

JORGE MANUEL LAGE FERNANDES

**UM ESTUDO DE GESTÃO DE PROJETOS COOPERADOS BASEADO EM PROJETOS
DE SOFTWARE LIVRE**

Florianópolis, 2009

UNIVERSIDADE DO ESTADO DE SANTA CATARINA – UDESC
PROGRAMA DE PÓS GRADUAÇÃO EM ADMINISTRAÇÃO
MESTRADO PROFISSIONAL EM ADMINISTRAÇÃO

JORGE MANUEL LAGE FERNANDES

**UM ESTUDO DE GESTÃO DE PROJETOS COOPERADOS BASEADO EM PROJETOS
DE SOFTWARE LIVRE**

Dissertação apresentada à Universidade do Estado de Santa Catarina como requisito para a obtenção do título de Mestre em Administração.

Orientador: Julio da Silva Dias

Florianópolis, 2009

JORGE MANUEL LAGE FERNANDES

**UM ESTUDO DE GESTÃO DE PROJETOS COOPERADOS BASEADO EM PROJETOS
DE SOFTWARE LIVRE**

Dissertação aprovada como requisito parcial para a obtenção do grau de Mestre em Administração, no curso de Mestrado profissional em Administração, do Programa de Pós-Graduação em Administração da Universidade do Estado de Santa Catarina.

Banca Examinadora:

Orientador:

Prof. Dr. Julio da Silva Dias
Universidade do Estado de Santa Catarina

Membro:

Prof. Dr. Carlos Roberto De Rolt
Universidade do Estado de Santa Catarina

Membro Externo:

Prof. Dr. Mario Antonio Ribeiro Dantas
Universidade Federal de Santa Catarina

Florianópolis, __/__/__

A Deus por me dar o dom da vida e as duas mulheres muito especiais na minha vida: Cláudia Elisa e Maria Eduarda.

AGRADECIMENTOS

A Deus, por fazer parte de todos os momentos nessa desafiadora caminhada.

À minha amada esposa, Cláudia, pela motivação para entrar no mestrado, pelo apoio e suporte constantes, pela compreensão, companheirismo, cumplicidade e amor.

A minha família, pelo apoio e expectativa do meu crescimento e sucesso. Em especial a minha querida mãe (*in memoriam*) a quem muito devo e de quem não poderia nunca me esquecer.

Ao meu orientador, Prof. Julio Dias por ter aceitado orientar este trabalho, pela dedicação e amparo que me prestou e pelas inúmeras supervisões e contribuições para o desenvolvimento desta pesquisa.

Ao Prof. De Rolt, pelo incentivo durante a disciplina “Tecnologias Organizacionais como Estratégia de Inovação” que serviu de inspiração para este trabalho de pesquisa.

Ao amigo e contribuidor especial para esta pesquisa Fernando Wuthstrack, que como sempre, não mediu esforços para prestar informações com boa vontade.

Aos colegas da YoungArts Tecnologia de Informação pela colaboração e compreensão nos bons e os maus momentos deste processo.

A Universidade do Estado de Santa Catarina, em especial aos professores do mestrado em administração que compartilharam seu conhecimento e sabedoria, sendo verdadeiros educadores.

RESUMO

As organizações estão vivenciando um momento de profundas transformações sociais e econômicas, proporcionadas principalmente pela possibilidade de cooperação baseada nas tecnologias de informação e comunicação. Neste cenário, esta pesquisa tem como objetivo a criação de um modelo conceitual para gestão de projetos cooperados através do estudo de projetos em rede baseados no desenvolvimento de software livre. Com a finalidade de coletar fontes de evidência que auxiliassem na compreensão do processo focalizado, o estudo analisou três projetos de desenvolvimento de software livre de diferentes características e portes. A pesquisa utilizou uma abordagem empírica e teórica, na qual além da revisão de literatura sobre o assunto, procurou-se verificar a coerência com a realidade, para conceber um conjunto de informações que servissem de referência para a construção do modelo de gestão de projetos cooperados. O trabalho de pesquisa foi apoiado por entrevistas estruturadas, realizadas a partir de um questionário pré-definido e a interpretação dos resultados foi realizada na percepção dos significados dos comportamentos dos sujeitos, coletados sistematicamente e reavaliados durante todo o processo de investigação. Os resultados de pesquisa auxiliaram para construir um modelo de gestão aplicável a este tipo de projetos e desta forma sugerem-se novos estudos que aprofundem o tema, e que forneçam a base técnica adequada para sua implantação.

Palavras-chave: Gestão de Projetos Cooperados. Software Livre. Modelos de Gestão. Organizações Virtuais.

ABSTRACT

The organizations are facing a moment of deep social and economics transformations caused by the cooperation possibility based on the information and communication technologies. In this scenery, this research has the main objective of building a cooperated project management conceptual model based on the study of free software network based projects. Using a purpose of collecting evidences that helped the comprehension of the focused process, the study has analyzed three free software development projects from different characteristics and sizes. The study was guided by a theoretical-empirical approach, which beyond the literature review, looked forward to verify the coherence with the reality to create a knowledge base for the cooperated project model construction. The research was bases on structured interviews, using a pre-defined questionnaire and the result interpretation that was made with the analysis of the subjects' behavior, systematically collected and revalued during the investigation process. The research results helped to build an applicable management model for this type of projects, thus new studies are suggested to supply adequate technical base for its implantation. Management

Keywords: Cooperated Project Management. Free Software. Management Models. Virtual Organizations.

LISTA DE ILUSTRAÇÕES

Figura	1	Tipologia de Redes de Empresas.....	28
Figura	2	Rede de Nós por Área de Especialidade.....	32
Figura	3	Ciclo de Vida de Uma Organização Virtual.....	35
Figura	4	Modelo Espiral de Boehm.....	39
Figura	5	Áreas de Atuação da Gerência de Projetos no Contexto de Software.....	40
Figura	6	Gráfico de Gantt.....	41
Figura	7	Redes PERT / CPM.....	42
Figura	8	Papel dos Membros X Nível Hierárquico.....	47
Figura	9	Diagrama Funcional do modelo CVS.....	49
Figura	10	Esquema Lógico do Funcionamento do CVS.....	50
Figura	11	Arquitetura de um sistema de controle de versões Subversion SVN.....	52
Figura	12	Quadro Esquemático das Motivações dos Projetos de Software Livre.....	60
Figura	13	Teoria das Necessidades do Ser Humano.....	62
Gráfico	14	Distribuição da Amostra da Pesquisa por Projeto.....	67
Figura	15	Topologia do Projeto Firebird.....	78
Gráfico	16	Fatores de Motivação dos Membros do Projeto Firebird.....	80
Gráfico	17	Gestão das Modificações do Projeto Firebird.....	83
Gráfico	18	Ferramentas e Tecnologias de Comunicação Adotadas pelos Membros do Projeto Firebird.....	84
Gráfico	19	Dificuldades Encontradas pelos Membros da Comunidade.....	87
Gráfico	20	Retorno dos Membros da Comunidade.....	88
Figura	21	Topologia do Projeto GNOME.....	93
Gráfico	22	Participação Geográfica dos Membros do Projeto GNOME.....	96
Gráfico	23	Fatores de Motivação dos Membros do Projeto GNOME.....	98
Gráfico	24	Gestão das Modificações do Projeto GNOME.....	100
Gráfico	25	Estratégias de Comunicação no Projeto GNOME.....	102
Figura	26	Exemplo de Sistema Baseado no Bugzilla.....	104
Gráfico	27	Dificuldades Encontradas pelos Membros da Comunidade.....	106
Gráfico	28	Retorno do Projeto GNOME.....	109
Figura	29	Topologia do Projeto TinyCobolBR.....	113
Gráfico	30	Fatores de Motivação do Projeto TinyCobolBR.....	114
Gráfico	31	Gestão das Modificações do Projeto TinyCobolBR.....	116
Gráfico	32	Estratégias de Comunicação do Projeto TinyCobolBR.....	118
Gráfico	33	Dificuldades Encontradas pelos Membros da Comunidade.....	120
Gráfico	34	Retorno dos Membros da Comunidade.....	121
Figura	35	Espaço de Cooperação.....	124
Figura	36	Espaço de Trabalho do Modelo Proposto.....	125
Figura	37	Áreas de Gestão do Modelo de Projetos Cooperados.....	126
Figura	38	Modelo de Gestão de Projetos Cooperados.....	135

LISTA DE TABELAS

Tabela	1	Aspectos econômicos: software proprietário e software livre.....	23
Tabela	2	Fases do Ciclo de Vida das Organizações Virtuais.....	36
Tabela	3	Hierarquia de Privilégios de Decisão dentro de um Projeto de Software Livre.....	45
Tabela	4	Elementos e Contexto na Organização Virtual.....	55
Tabela	5	Caracterização dos Projetos Estudados.....	63
Tabela	6	Compilação dos Resultados do Questionário de Pesquisa – Firebird.....	89
Tabela	7	Compilação dos Resultados do Questionário de Pesquisa – GNOME.....	109
Tabela	8	Compilação dos Resultados do Questionário de Pesquisa – TinyCobolBr..	122
Tabela	9	Necessidade Identificada X Ferramenta de TI Indicada.....	129

LISTA DE ABREVIATURAS E SIGLAS

API	Application Program Interface (Interface de Programação de Aplicativos)
CVS	Concurrent Version System (Sistema de Versões Concorrentes)
COBOL	Common Business Oriented Language
ESAG	Escola Superior de Administração e Gerência
GNOME	GNU Network Object Model Environment
GNU	Gnu is Not Unix (sistema operacional de código fonte aberto inspirado em Unix)
HP	Hewlett Packard
IRC	Internet Relay Chat
IBM	International Business Machines
LABGES	Laboratório de Tecnologias de Gestão
ONG	Organização do terceiro setor sem fins lucrativos
OV	Organização Virtual
PERT	Program Evaluation and Review Technique
SGBD	Sistema Gerenciador de Banco de Dados
SQL	Structured Query Language (Linguagem estruturada de consulta de bancos de dados)
SVN	Subversion (Sistema de Controle de Versão)
TIC	Tecnologia da Informação e Comunicação
UDESC	Universidade do Estado de Santa Catarina
USP	Universidade de São Paulo

SUMÁRIO

1	INTRODUÇÃO.....	14
1.1	Definição do Problema.....	16
1.2	Objetivos.....	17
1.2.1	<i>Objetivo Geral.....</i>	17
1.2.2	<i>Objetivos Específicos.....</i>	18
1.3	Justificativa.....	18
1.4	Organização do Estudo.....	21
2	FUNDAMENTAÇÃO TEÓRICA.....	22
2.1	Software Livre.....	22
2.2	Projetos Cooperados e Redes.....	26
2.3	Redes de Negócios.....	27
2.3.1	<i>Redes Sociais.....</i>	29
2.3.2	<i>Redes Burocráticas.....</i>	29
2.3.3	<i>Redes Proprietárias.....</i>	30
2.4	Organizações Virtuais.....	31
2.4.1	<i>A Força de Trabalho nas Organizações Virtuais.....</i>	33
2.4.2	<i>Ciclo de Vida das Organizações Virtuais.....</i>	35
2.4.3	<i>Benefícios das Organizações Virtuais.....</i>	37
2.5	A Metodologia de Desenvolvimento de Sistemas.....	38
2.6	Gerenciamento de Projetos.....	39
2.7	Ferramentas de Gerenciamento de Projetos.....	40
2.7.1	<i>Gráfico de Gantt.....</i>	41
2.7.2	<i>Redes Pert / CPM.....</i>	42
2.8	A Organização da Equipe de Desenvolvimento de Software Livre.....	43
2.8.1	<i>Hierarquia.....</i>	45
2.9	CVS: Ferramenta de Gerenciamento das Versões de Software.....	48
2.10	SVN: Ferramenta de Gerenciamento das Versões de Software.....	51
2.11	A Expansibilidade dos Projetos de Software Livre.....	53
2.12	A Produção de Software Livre como Organização Virtual.....	54
2.13	A Estratégia de Comunicação nesta Organização Virtual.....	56
2.14	O Controle das Organizações Virtuais de Software Livre.....	57
2.15	O Licenciamento de Software Livre.....	58
2.16	Motivação, Software Livre e a Cooperação de Projetos.....	59
2.17	Partindo para a Pesquisa de Campo.....	62
3	PROCEDIMENTOS METODOLÓGICOS.....	63
3.1	Caracterização do Estudo.....	64
3.2	Procedimento de Coleta de Dados.....	66
3.3	Desenvolvimento do Estudo.....	68
3.4	Validação do Questionário de Pesquisa.....	69

3.5	Limitações do Estudo.....	70
4	ESTUDO DE PROJETOS COOPERADOS EM SOFTWARE LIVRE.....	71
4.1	Análise do Projeto Firebird.....	71
4.1.1	<i>Sistemas Gerenciadores de Bancos de Dados.....</i>	71
4.1.2	<i>Histórico.....</i>	74
4.1.3	<i>Firebird no Brasil.....</i>	76
4.1.4	<i>A Fundação Firebird.....</i>	77
4.1.5	<i>Resultados da Pesquisa de Campo.....</i>	79
4.1.6	<i>Fatores de Motivação.....</i>	79
4.1.7	<i>Gestão das Modificações.....</i>	82
4.1.8	<i>Estratégias de Comunicação.....</i>	84
4.1.9	<i>Dificuldades no Projeto.....</i>	86
4.1.10	<i>Retorno do Projeto.....</i>	87
4.2	Análise do Projeto GNOME.....	90
4.2.1	<i>Histórico.....</i>	90
4.2.2	<i>Arquitetura do Projeto.....</i>	94
4.2.3	<i>Caracterização da Comunidade.....</i>	94
4.2.4	<i>Fatores de Motivação.....</i>	97
4.2.5	<i>Gestão das Modificações.....</i>	99
4.2.6	<i>Estratégias de Comunicação.....</i>	101
4.2.7	<i>Dificuldades no Projeto.....</i>	105
4.2.8	<i>Retorno do Projeto.....</i>	107
4.3	Análise do Projeto TinyCobolBr.....	111
4.3.1	<i>Histórico.....</i>	111
4.3.2	<i>Fatores de Motivação.....</i>	114
4.3.3	<i>Gestão das Modificações.....</i>	115
4.3.4	<i>Estratégias de Comunicação.....</i>	117
4.3.5	<i>Dificuldades no Projeto.....</i>	119
4.3.6	<i>Retorno do Projeto.....</i>	121
5	DISCUSSÃO DOS RESULTADOS E APRESENTAÇÃO DA PROPOSTA...	123
5.1	Modelo como Resposta para Dificuldades identificadas.....	123
5.2	Espaços de Cooperação.....	124
5.2.1	<i>O Espaço Comum de Desenvolvimento do Trabalho.....</i>	125
5.2.2	<i>A Área de Funções de Gestão do Projeto.....</i>	126
5.2.3	<i>A Plataforma de Tecnologia de Informação.....</i>	128
5.2.3.1	<i>Comunicação Interna.....</i>	129
5.2.3.2	<i>Planejamento, Acompanhamento e Administração de Recursos Humanos.....</i>	130
5.2.3.3	<i>Base de Conhecimento.....</i>	131
5.3	Os Atores Envolvidos no Processo.....	132
5.4	Modelo de Gestão de Projetos Cooperados.....	133
6	CONCLUSÕES E RECOMENDAÇÕES.....	136
6.1	Conclusões.....	136
6.2	Recomendações e Trabalhos Futuros.....	138

REFERÊNCIAS.....	139
GLOSSÁRIO.....	150
ANEXOS.....	152
ANEXO A – QUESTIONÁRIO DE PESQUISA 1.....	153
ANEXO B – QUESTIONÁRIO DE PESQUISA 2.....	155
ANEXO C – QUESTIONÁRIO DE PESQUISA 3.....	157

1 INTRODUÇÃO

A nova era da informação tem transformado a dinâmica das organizações, de maneira semelhante ao ocorrido na Revolução Industrial fez dois séculos atrás. Assim, a nossa sociedade está apenas começando a sentir os seus impactos. O computador representa, para a Revolução da Informação, o que a máquina a vapor representou na Revolução Industrial do século XV (DRUCKER, 2000).

Entretanto, se por um lado as tecnologias desenvolvem-se rapidamente, por outro as organizações precisam redesenhar os seus processos de negócios em função dos impactos da utilização das tecnologias de informação e comunicação, obrigando-as a entrar em um novo ambiente virtual de competição (DE ROLT, 1999).

Castells (1999), afirma que na sociedade interconectada, a fonte primária de criação de valor mudou a ênfase da produtividade para os relacionamentos, e a capacidade de colaborar precisa se tornar uma competência-chave para a organização.

Assim, como resposta a esse ambiente, o processo de cooperação entre as empresas deverá aumentar não apenas em relação a frequência, mas também em relação a sua importância estratégica. Por isso, o estabelecimento de alianças através do meio virtual deve ser algo cada vez mais presente ao longo dos próximos anos (DRUCKER, 2000).

É possível observar esta realidade pela franca expansão da utilização da Tecnologia da Informação na última década, que permitiu o surgimento das cooperações estabelecidas entre as organizações e pessoas numa dimensão global. Portanto, a Tecnologia da Informação permite que as corporações não vejam as barreiras de tempo e localização (O'LEARY; KUOKKA; PLANT, 1997).

Neste contexto, onde a visão estratégica da empresa contempla a identificação das necessidades de mercado, a previsão de futuro, a internação de metas e a inovação, ocorrem impactos na estrutura organizacional e operacional de uma organização competitiva (COSTA; DE ROLT; DIAS, 2008).

Castells (1999) afirma que essa nova realidade faz com que os mercados mudem, profissões sejam extintas, e que os paradigmas vigentes sejam modificados pelo atropelo da tecnologia da informação.

Por conseguinte, para enfrentar este e cenário de mudanças, torna-se necessário um arcabouço de informações que forneçam uma base de conhecimento sobre o tema. É nesta missão que esta dissertação tem sua finalidade principal, como um instrumento de embasamento para as futuras cooperações.

A elaboração deste trabalho de dissertação utilizou-se o método investigativo do estudo de caso, envolvendo três projetos virtuais de cooperação, segundo uma abordagem descritivo-avaliativa, que objetiva a modelagem dos processos de negócio relacionado com o objeto de pesquisa definido (MERRIAM, 1988).

Os projetos analisados neste estudo são projetos de cooperação de desenvolvimento de software livre de três níveis distintos de grandeza. Um projeto pequeno para atender uma necessidade específica (TinyCobolBR) , um projeto de médio porte que originou-se de um produto comercial (Firebird) e um projeto grande, com dimensões globais (GNOME), um dos ícones do movimento de software livre.

A escolha dos projetos foi motivada por dois fatores principais: A relevância do setor de software para a economia mundial e a característica virtual da gestão destes projetos.

Neste contexto, a indústria de *software* tem um papel muito significativo na economia do país, e em especial, o software livre como estrutura de produção colaborativa em rede, vem se tornando uma alternativa interessante, tanto para esfera pública ou privada.

Por outro lado, os conceitos da virtualidade do processo de desenvolvimento de software livre estão muito próximos da literatura de organizações virtuais e redes de negócios. Entretanto, a literatura existente está voltada principalmente para a análise da questão do software livre para o usuário final e na discussão sobre a adoção ou não de sua filosofia num contexto de movimento social (TRIANA, 2007). Logo, observa-se uma lacuna de estudos voltados para o software livre como negócio, mais especificamente na tecnologia do processo gerencial destas organizações.

A filosofia software livre central baseia-se na livre troca de conhecimentos e de pensamentos que podem tradicionalmente ser encontrado no meio científico. O desenvolvimento de software livre ocorre numa nova forma de cooperação das partes envolvidas, constituindo assim verdadeiras “redes dinâmicas de cooperação”, através da utilização das novas tecnologias de informação.

Assim, a proposição de modelos que facilitem e orientem a estruturação, organização e gestão de um projeto de cooperação, tendo como base modelos teóricos e práticas de sucesso, podem favorecer de maneira considerável no sucesso da cooperação das empresas envolvidas.

1.1 Definição do Problema

Se a gestão das organizações ditas “tradicionais” na era da informação e cooperação se torna algo novo a cada dia, a gestão desses projetos cooperados é um processo ainda mais complexo.

É comum que os membros de um projeto não tenham a noção da estrutura e hierarquia presente por trás deste ciberespaço colaborativo da qual fazem parte. No âmbito dos ambientes colaborativos de desenvolvimento do software, ocorre o encontro de diferentes individualidades que conseguem desenvolver um produto sem nunca se encontrar ou, sequer saber muitas vezes, o verdadeiro nome dos outros participantes (TRIANA, 2007).

Nos últimos anos, a discussão desta temática trouxe novos conceitos e práticas no campo da administração que visam o desenvolvimento de propostas de colaboração e cooperação organizacionais para o aumento da competitividade das organizações, a redução dos riscos e principalmente, a geração de valor para as organizações.

A própria característica informal das relações estabelecidas na rede colabora com este fato, visto que a gestão dos projetos é compartilhada e com isso a visão organizacional da rede torna-se bastante nebulosa.

Diante do exposto, têm-se as seguintes perguntas de pesquisa:

- Como ocorre a gestão das redes para produção de software livre?
- Qual é o modelo de gestão proveniente da observação dos projetos cooperados derivado da observação dos projetos de cooperação de software livre?

O Mestrado Profissional em Administração da UDESC/ESAG concentra-se na área de Gestão Estratégica das Organizações, onde uma das linhas de pesquisa é a de Organizações e

Tecnologias de Gestão. Essa linha de pesquisa visa realizar estudos e pesquisas, em organizações para promover mudanças organizacionais e das tecnologias de gestão.

Mais especificamente, vinculado ao Mestrado existe o LABGES (Laboratório de Tecnologias de Gestão) que busca analisar a aplicação e os impactos das tecnologias computacionais nas organizações, com o objetivo prático de desenvolver modelos de referência para a gestão de sistemas virtuais confiáveis.

Tendo em vista a existência desta linha de atuação de pesquisa e considerando a relevância das aplicações dos conceitos e dos resultados encontrados, as oportunidades de contribuição de fundamentação teórica para o meio acadêmico, este estudo focaliza o estabelecimento de um modelo de gestão para projetos de cooperação utilizando como base o modelo de desenvolvimento do software livre.

1.2 Objetivos

Utilizando como base a definição do problema da pesquisa, o objetivo geral e os específicos são os seguintes:

1.2.1 Objetivo Geral

Esta pesquisa tem como objetivo realizar o estudo de três casos de projetos em rede baseados no desenvolvimento de *software* livre para a criação do modelo de gestão de projetos cooperados, que contribua para o estabelecimento empírico e teórico de referências para futuros estudos e implantações de projetos de cooperação.

1.2.2 Objetivos Específicos

O objetivo geral da pesquisa é consolidado através da realização dos seguintes objetivos específicos:

- Realizar uma revisão bibliográfica sobre projetos cooperados, organizações virtuais e desenvolvimento de software livre como suporte teórico para esta pesquisa;
- Identificar na literatura os empecilhos e dificuldades para gestão de projetos de desenvolvimento de software livre;
- Realizar um estudo comparativo de três casos em redes de desenvolvimento de softwares livre (TinyCobolBR, Firebord e GNOME);
- Proposição de um modelo conceitual que possa ser utilizada levando em consideração as características dos projetos estudados;

1.3 Justificativa

O advento da era da informação possibilitou que as organizações buscassem novas formas de trabalho e de serviço, numa inovação. Processo contínuo de inovação voltado principalmente para o incremento da competitividade.

As tecnologias da informação emergentes desta realidade, ao mesmo tempo em que proporcionam inúmeros recursos inerentes a entrada no ciberespaço, por outro lado submetem a organização a um nível de competitividade global devido a velocidade em que ocorrem as mudanças e oportunidades.

Um dos maiores desafios para uma organização perante o atual ambiente dinâmico é manter a competitividade em todas as atividades de sua cadeia de valor. Mais especificamente, nas áreas onde as fontes de conhecimento estão largamente distribuídas e o avanço tecnológico é veloz, uma organização que se isola não consegue deter todas as habilidades necessárias para se manter competitiva.

Bin e Salle-Filho (2007) afirmam que este processo de inovação baseia-se na criação e apropriação social (via mercado ou não) de produtos, serviços, processos, métodos e sistemas que

não existiam anteriormente, ou contendo alguma característica nova e diferente da até então em vigor.

Neste contexto, empresas e formas de trabalhos tradicionais vêm sofrendo uma série de adaptações com a utilização destas tecnologias, principalmente através das redes de negócios. Estas tecnologias vêm possibilitando uma comunicação e troca de informações “*on-line*” com todo o mundo através de linhas de transmissão de dados digitais e da fibra ótica (TROPE, 1999).

O desenvolvimento tecnológico e a popularização da Internet proporcionaram uma interconexão das organizações e pessoas, criando um espaço comum de comunicação interativa. A este espaço denomina-se ciberespaço, nele é possível o compartilhamento da memória, da percepção, da imaginação. Isso resulta na aprendizagem coletiva e na troca de conhecimentos entre os grupos.

Segundo LEVY (1999), o ciberespaço é uma realidade irreversível, e cabe a nós explorar as potencialidades mais positivas desse espaço nos planos econômico, político, cultural e humano. Assim, as organizações vêm adotando esse espaço não só como uma forma de negócio, mas também como uma ferramenta para customizar e facilitar o trabalho cresce geometricamente a cada dia.

Para as organizações, a rede mundial de computadores (*Internet*), torna-se um dos principais catalisadores de transformações no mundo dos negócios, fazendo com que as empresas orientem-se para este meio como estratégia de crescimento, principalmente na área de vendas (DRUCKER, 2000).

Além disso, Miranda e Simeão (2003) destacam que a indústria da informação é o setor que mais emprega nos países avançados, sustentado pelos avanços da informática. Para o autor, outros aspectos da nova indústria da informação devem ser observados. Um deles seria que a automação e a robotização na agricultura e na indústria tendem a reduzir os níveis de emprego. O outro é que parte da mão-de-obra ociosa tende a ser absorvida pelos setores de serviços e da informação, mas somente na medida em que ela for sendo reciclada, preparada para assumir as novas funções.

É neste ponto que os desenvolvedores de *software* – em especial, de *software* livre – utilizam este espaço para formar uma organização virtual e com isso o tema vem ganhando muita atenção de várias empresas no mundo inteiro.

O mercado mundial de software demonstra-se extremamente relevante já que movimentou cerca US\$ 900 bilhões em negócios em 2008. O Brasil tem uma participação muito relevante, sendo atualmente o sexto maior, com vendas de US\$ 7,7 bilhões anuais, com 5.400 empresas, empregando 158 mil pessoas. Além disso, o setor é responsável por 1,5% do produto interno bruto brasileiro (VELOSO et al., 2003).

Aproximando ainda esta análise da realidade do pesquisador e da universidade promotora deste trabalho, observa-se ainda que Santa Catarina é um dos mais importantes pólos de desenvolvimento de software. Estão instaladas no estado 13,5% das empresas, sendo o segundo maior em número proporcionalmente e o primeiro no ranking brasileiro do segmento (QUEIROZ, 2003).

Além disso, a presença de outras diversas iniciativas de desenvolvimento de *software* livre demonstra a relevância das organizações estudadas para a economia do país. Diante deste fato, o presente estudo justifica-se pela necessidade de se discutir sobre o problema da gestão e operacionalização destas redes.

A emergência do *software* livre e o sucesso fenomenal dos seus pioneiros – como o sistema operacional GNU/Linux, o servidor de rede Apache, Perl, BIND – e muitos outros projetos têm nos forçado a realizar uma revisão no paradigma dominante sobre produção.” (BENKLER, 2002).

Assim, tratar desse tema é relevante na medida em que as informações geradas permitirão entender melhor essas redes, sugerir melhorias e identificar possíveis razões que dificultam o seu perfeito funcionamento.

Neste contexto, o mapeamento das características principais da gestão destes projetos, visando a consolidação de um modelo de cooperação com base nas repostas às perguntas de pesquisa, vem se configurando altamente relevantes, como grandes desafios para os estudos abordando este tipo de organizações.

Outro aspecto relevante é que o estudo dessa nova forma de produção aplicada ao desenvolvimento de *software* é relativamente recente e existe muito pouca literatura disponível. Com isso, a pesquisa pode fomentar o interesse de outros pesquisadores e organizações em geral, de em torno de alternativas de aperfeiçoamento dos processos gerenciais, aplicando o conhecimento oriundo da pesquisa aos seus contextos organizacionais.

1.4 Organização do Estudo

A presente dissertação apresenta uma descrição completa do domínio do problema, da metodologia de pesquisa utilizada, e dos resultados obtidos. A estrutura deste trabalho segue a seguinte seqüência:

No **primeiro capítulo** desta pesquisa, encontram-se apresentados: a apresentação do tema, a sua organização, a definição e a natureza do problema de pesquisa, apresentados de forma sucinta. Além disso, apresentam-se também os objetivos da pesquisa e a relevância do tema abordado.

O **segundo capítulo** apresenta a revisão da literatura, consultando para isso uma gama bastante expressiva de autores, abordando:

- Contextualização e caracterização de projetos de software livre;
- Organizações virtuais e redes;
- Os grupos de desenvolvimento de software livre como organizações virtuais;
- Gerência de projetos nas organizações virtuais de software livre.

Posteriormente, no **terceiro capítulo**, são apresentados os procedimentos metodológicos, iniciando pela caracterização da pesquisa, a apresentação das técnicas e instrumentos de coleta e análise dos dados, uma breve definição dos termos e variáveis envolvidos e as prováveis limitações da pesquisa.

No **quarto capítulo**, apresenta-se inicialmente a caracterização de cada uma das projetos de desenvolvimento de software livre estudados, descrevendo-se a sua forma de organização, seus aspectos estratégicos e o relacionamento destes com a revisão teórica do **segundo capítulo**. Após a caracterização destes projetos são apresentados os dados coletados na pesquisa, apresentando uma descrição narrativa e gráfica do processo estudado.

No **quinto capítulo** realizam-se as conclusões a respeito do trabalho apontando os resultados alcançados assim como apresentando recomendações para trabalhos futuros.

Finalmente, no **sexto capítulo**, é apresentada a referência bibliográfica utilizada na dissertação e também um glossário para consulta dos principais conceitos abordados no estudo.

Por fim, no **sétimo capítulo**, apresentamos as entrevistas, em forma de anexo, que serviram de coleta de dados para este trabalho.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo será apresentada uma descrição sucinta sobre assuntos pertinentes ao estudo dos projetos e a construção do modelo em questão. A compreensão destes tópicos faz-se necessária para uma melhor compreensão da relação entre os conceitos estudados.

Inicialmente busca-se apresentar e compreender as principais características do Software Livre, caracterizando-o como um processo de desenvolvimento de trabalho em rede. Posteriormente, são apresentados dentro deste contexto, os principais tipos de redes de cooperação. Estes conceitos são de vital interesse para a compreensão do conceito posterior de Organizações Virtuais.

Continuando o processo de fundamentação, o software livre é caracterizado também como um tipo de Organização Virtual. Este tema recebeu uma atenção especial no estudo devido fato dos projetos cooperados analisados possuírem características muito próximas a este tipo de organização.

Finalmente, são abordados aspectos inerentes do processo de desenvolvimento de software livre como: Metodologia específica de desenvolvimento, ferramentas de acompanhamento e gestão, licenciamento e liberação de versões, hierarquia e motivação. Todos esses fatores constituem-se em variáveis do cotidiano dos projetos de cooperação e servem como inspiração para o estudo dos projetos contemplado no capítulo 4.

2.1 Software Livre

O software livre pode ser definido como qualquer programa de computador que pode ser usado, copiado, estudado, modificado e redistribuído com algumas restrições (INSTITUTO NACIONAL DE TECNOLOGIA DE INFORMAÇÃO, 2005). Estas características, que são primordiais em qualquer projeto de software livre, se opõem diretamente ao conceito de software proprietário.

Este modelo de desenvolvimento de software surgiu há algumas décadas e seu uso vem crescendo no mundo, sendo aplicado em diversos setores por apresentar soluções econômicas e atender a necessidades de segurança de informações.

Para Bezerra (2008, p. 7), o software livre nasce de duas formas:

Pode ser fruto dos esforços de um programador individual que resolve, no momento de lançar o programa que criou, disponibilizar na internet seu código-fonte para modificação, estudo e distribuição. (...) Outras vezes pode surgir já como um projeto coletivo: um grupo de desenvolvedores se reúne para discutir os problemas computacionais a serem resolvidos pelo programa, as especificações básicas e os rumos do desenvolvimento. A partir dessas definições iniciais, o grupo começa a criar o software e publica seu código-fonte na internet, com a intenção expressa de mobilizar outros programadores a fim de dar continuidade ao projeto.

Algumas características importantes e os aspectos econômicos do modelo de negócios de software livre que o diferenciam o software proprietário são demonstrados na tabela 1:

Tabela 1: Aspectos econômicos: software proprietário e software livre

Aspectos Econômicos	Software Proprietário	Software Livre
Custos de desenvolvimento (<i>first copy costs</i>)	Maior	menor
Custo marginal de produção	Igual	igual
Economias de escopo na produção	Menor	maior
Efeitos de rede do lado da oferta	Igual	igual
Depreciação	Igual	igual
Efeitos de rede do lado da demanda	Igual	igual
Cumulatividade e efeitos de lock-in	Maior	menor
Não rivalidade no consumo	Igual	igual
Apropriabilidade	Maior	menor
Barreiras à entrada	Maior	menor
Ciclo do produto	Menor	maior
Taxa de inovatividade	Menor	maior
Criação de discontinuidades tecnológicas	Maior	menor

Fonte: Salles-Filho, 2006

Salles-Filho (2006, p. 1) afirma que o desenvolvimento de software livre gera economias de rede maiores que no software proprietário em decorrência das “externalidades positivas que surgem quando se desenvolve um produto com a participação de um conjunto maior (e aberto) de pessoas” sendo necessária uma coordenação eficaz para se “aproveitar o conhecimento novo que se gera da interação de grupos de especialistas.”

Além disso, o modelo produção de software tem resultado em produtos de excelente qualidade e de grande penetração e aceitação em vários setores do mercado mundial de software, com diversos casos de sucesso que estão em plena evolução pela colaboração em massa.

Tal como as idéias, os programas de computador não são tangíveis e podem ser copiados, trabalhados e até questionados. A sua distribuição é considerada a base de um processo de evolução que alimenta o desenvolvimento do pensamento.

Segundo Tirole e Lerner (2002), a criação e manutenção de software livre é um processo complexo que envolve desenvolvedores em diferentes localidades e organizações compartilhando o código fonte, para desenvolver e refinar os programas em um processo de colaboração em massa. Ainda segundo este autor, o crescente interesse no assunto deve-se à rápida difusão do software de fonte aberta, como por exemplo, o Apache servidor de internet, e o Linux , que já possui mais de vinte milhões de usuários em todo o mundo.

O fenômeno software livre começou a despertar atenção da comunidade técnica e de negócios nos últimos anos. Software Livre permite livre acesso ao código fonte de um programa. Código fonte são as instruções de um programa de software escrito em linguagem de programação. Qualquer desenvolvedor que conheça esta linguagem pode ler, entender e modificar o programa. Ter acesso ao código fonte é, portanto, a chave para abrir e modificar qualquer software (Taurion, 2004, p. 16).

O sucesso destes projetos inspirou um número cada vez maior de organizações que percebendo os benefícios desta colaboração em massa, adotaram o software livre como um novo modo de organização para substituir as estruturas empresarias tradicionais como o motor primário de criação e riqueza na economia (TAPSCOTT, 2007).

O software livre não é necessariamente, produto de um processo puramente colaborativo: pode ser fruto dos esforços de um programador individual que resolve, no momento de lançar o programa, disponibilizar livremente aos usuários o seu código-fonte. Nesse momento, este autor não está abrindo mão da autoria do software, mas disponibilizando a comunidade o seu trabalho para que esse possa modificação, estudo e distribuição livres.

O modelo do Software livre subverte a concepção que poderíamos ter de cooperação e inovação. É forçoso constatar que pela primeira vez estamos diante de uma cooperação maciça, planetária, altamente valorizável, assegurada conjuntamente por empresas, é verdade, mas sobretudo por um conjunto de sujeitos autônomos, ao mesmo tempo no coração do sistema capitalista e também fora dele, mesmo além. (MOINEAU; PAPATHÉODOROU, 2000, p. 115)

Neste contexto, para que a liberdade de fazer modificações, e de publicar versões aperfeiçoadas, tenha algum significado, deve-se ter acesso ao código-fonte do programa. Portanto, acesso ao código-fonte é uma condição necessária ao software livre.

A característica mais importante do software livre é a liberdade de uso, cópia, modificações e redistribuição. Esta liberdade é conferida pelos autores do programa e é efetivada através da distribuição do código fonte dos programas, o que os transforma em bens públicos, disponíveis para utilização por toda a comunidade e da maneira que seja mais conveniente a cada indivíduo (HEXSEL, 2002).

A grande maioria dos softwares proprietários não possui o código aberto ao usuário final. Portanto, não é possível realizar as quaisquer modificações no cerne do programa, verificar a qualidade do código, realizar melhorias no programa ou mesmo corrigir eventuais erros em a interferência de seu criador.

Essa liberdade para usar, copiar, modificar e redistribuir software livre lhe confere uma série enorme de vantagens sobre o software proprietário. A mais importante delas é disponibilidade do código fonte, porque isto evita que os usuários se tornem reféns de tecnologias proprietárias (HEXSEL, 2002).

Por outro lado, este conceito não necessariamente é oposto ao software comercial que visa à obtenção de lucro. Inclusive, o desenvolvimento comercial de software livre não é incomum.

Existem diversos softwares livres comerciais de código aberto, onde as empresas que desenvolvem estes softwares sobrevivem da comercialização do próprio software e de outras atividades relativas ao mesmo como: consultora, treinamento, implantação e customização.

Grandes corporações, dentre as quais a HP, IBM, e a SUN possuem projetos para desenvolver e usar software livre. Além disso, já existem companhias especializadas em comercializar Linux (como por exemplo, a Red Hat) e também outras companhias de fonte aberta como a Cobalt Networks, Collab.Net, Scriptics e Sendmail tem recebido grandes financiamentos públicos e privados. Logo, pode-se concluir que um software pode ser livre por natureza, mas visar algum tipo de lucro.

No Brasil é possível encontrar diversos casos de sucesso de projetos de cooperação para a produção de software livre, como por exemplo as iniciativas dos estados do Rio Grande do Sul, Paraná e Pernambuco. Estes estados adotaram em sua estrutura de tecnologia de informação este tipo de soluções.

O Governo brasileiro também em diversas áreas adota o Software Livre como uma alternativa interessante tanto no aspecto econômico como no social. Um exemplo é o Programa Sociedade da Informação que requer um uso maciço de software em diversas de suas ações.

Atualmente, o software livre já vem sendo usado e aprovado por diversas áreas na sociedade brasileira, como em empresas privadas, em empresas estatais, ONGs, Ministérios, Governos Estaduais e municipais, tanto no regime administrativo como também na educação.

2.2 Projetos Cooperados e Redes

Uma das principais características presente nestes projetos de software livre é a co-produção de diferentes competências de diversos profissionais em diversas partes do mundo.

Estes profissionais estabelecem redes para a produção de software que possibilitam o estabelecimento de cooperação entre as partes envolvidas, onde cada uma das partes colabora com as suas maiores e melhores competências. Essa forma de trabalho possibilita assim o aumento da competitividade e da abrangência das partes.

Segundo Neto e Olave (1998), estas redes de empresas são formadas inicialmente com o objetivo de reduzir incertezas e riscos, organizando atividades econômicas a partir da coordenação e cooperação entre empresas.

Com isso, as barreiras de distancia e tempo são eliminadas, possibilitando a criação de um projeto comum proveniente de diversas atuações distintas sem um envolvimento formal ou até mesmo pessoal.

Estas redes encaixam-se também no conceito de organizações virtuais, uma vez que as redes formadas estabelecem-se de forma temporária e podem envolver indivíduos, pequenas empresas ou partes de corporações maiores que se constituem com uma finalidade específica.

Entretanto, o que diferencia a maioria os projetos de Software Livre das organizações virtuais é que as relações estabelecidas entre os membros da rede podem tornar-se permanentes, enquanto em organizações virtuais quando é gerado o produto final da colaboração ocorre a dissolução da rede.

2.3 Redes de Negócios

Ao longo do histórico da humanidade e das organizações, sempre existiu uma divisão hierárquica, onde as diversas partes eram controladas por respectivos gestores. Às classes inferiores cabia um papel meramente coadjuvante sem qualquer participação no planejamento e no processo decisório dessas organizações.

Embora as hierarquias continuem sendo a base organizacional destas empresas, mudanças profundas na natureza da tecnologia, da demografia e da economia global estão fazendo emergir novos e modelos de produção baseados em comunidade, colaboração e auto-organização, sem hierarquia e controle rígidos. (TAPSCOTT, 2007).

Neste contexto, muitas organizações estão optando por estabelecer parcerias e fazer parte de organizações em rede, que as permitem concentrarem-se nas suas competências essenciais.

Powell (1990) afirma que diversos autores têm observado que existe uma nova forma de organização econômica e que muitos outros admitem que está emergindo uma nova forma de organização social. Para ele, as trocas econômicas estão envoltas em um contexto particular de estrutura social, dependentes de conexões, interesses mútuos e reputação e pouco guiadas por uma estrutura formal de autoridade.

A gestão compartilhada de inovação, baseada na colaboração em massa, é hoje uma tendência mundial e atinge as organizações. Pode-se inferir que esta tendência, cujos limites esboçam os contornos de uma nova lógica organizacional, altera o leque de estratégias e de estruturas vigentes, gerando um processo evolutivo que conduz a estruturas mais auto-organizadas.

Tapscott; Lowy e Ticoll (2000) nomeiam estas organizações definem as redes como uma rede elaborada de ofertantes, distribuidores, fornecedores de serviços comerciais e consumidores que conduzem as comunicações e transações relacionadas ao negócio na Internet e em outras mídias eletrônicas de forma a produzir valor para consumidores finais.

O estabelecimento de uma parceria ou aliança entre duas ou mais empresas é um processo que envolve a concretização de relacionamentos, que devem ser baseados principalmente na confiança nos membros da rede.

Segundo Wildelman (1999), uma organização procura parceiros estratégicos por três motivos: ou por estarem buscando competências complementares às suas, ou por terem identificado parceiros que poderão lhes fornecer acesso à um mercado específico ou por estarem visando o estabelecimento de uma estratégia de economia de escala.

Neste contexto, as organizações em redes possuem duas dimensões essenciais: controle econômico e grau de integração de valor. No que diz respeito ao controle econômico, elas podem ser auto-organizadas, onde nenhuma companhia dirige o conteúdo de suas transações ou o resultado econômico, ou hierárquicas, nas quais um “chefe” que controla o conteúdo, a política de preços e o fluxo das transações.

Com relação ao grau de integração de valor, as redes de negócios podem variar de baixo, quando elas não mudam a natureza dos produtos oferecidos, a alto, quando elas facilitam a criação, transformação e distribuição de produtos específicos que integram ou agregam componentes de múltiplas fontes.

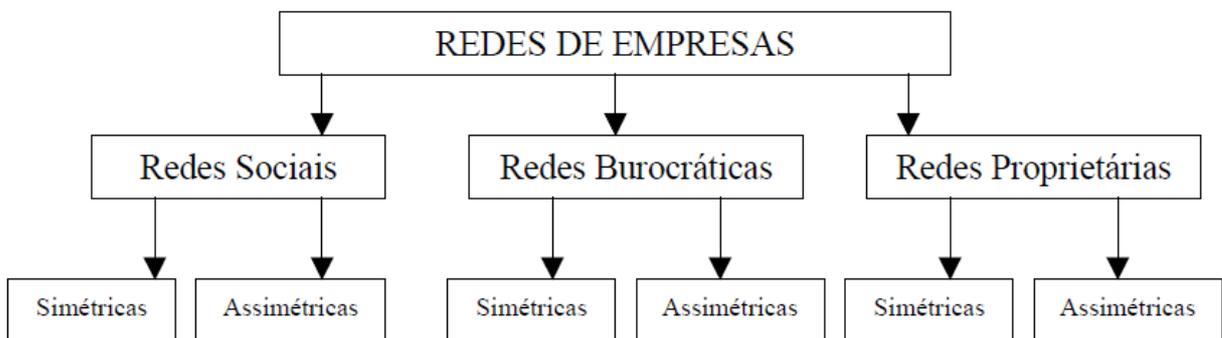


Figura 1 – Tipologia de Redes de Empresas.
Fonte: GRANDORI & SODA (1995)

Gandori e Soda apud Neto e Olave (1998), desenvolvem uma tipologia de redes de empresas baseada nos seguintes critérios: tipo de mecanismos de coordenação, grau de centralização da rede e grau de formalização da rede (Figura 1). A partir destes critérios, os autores identificaram três tipos básicos de redes:

2.3.1 Redes Sociais

As redes sociais têm por característica fundamental a informalidade nas relações inter-empresariais, isto é, prescindem de qualquer tipo de acordo ou contrato formal. Estas redes estão direcionadas para o intercâmbio da chamada “mercadoria social” (prestígio, status, e mobilidade profissional).

Caracterizam-se pela informalidade das relações entre as empresas. Podem ser simétricas, onde não existe um poder centralizado e todos os participantes têm a mesma capacidade de influência, ou assimétricas, em que existe a presença de um agente que coordena a rede.

As redes sociais simétricas se caracterizam pela inexistência de um poder centralizado, ou seja, todos os participantes desta rede compartilham a mesma capacidade de influência. São arranjos inter-organizacionais empregados em projetos de caráter mais exploratório, cujas informações são de alto potencial, porém de valor econômico desconhecido. São típicos os exemplos dos pólos e distritos de alta tecnologia, onde há uma intensa troca de informações e de conhecimentos entre as partes, sendo sua coordenação realizada através de mecanismos informais.

Já nas redes sociais assimétricas há a presença de um agente central, que tem por função primordial coordenar os contratos formais de fornecimento de produtos e/ou serviços entre as organizações que participam desta rede. A rede italiana de fornecimento da Benetton pode ser tomada como uma rede social assimétrica (GRANDORI; SODA, 1995).

2.3.2 Redes Burocráticas

Estas redes, em oposição às redes sociais, caracterizam-se pela existência de um contrato formal que define o relacionamento entre os membros da rede. A formalidade dos instrumentos desta rede e destina a regular não somente as especificações de fornecimento (de serviços e produtos), a própria organização da rede e as condições de relacionamento entre seus membros.

Assim como para as redes sociais, é possível subdividir as redes burocráticas em simétricas e assimétricas.

Na categoria de redes burocráticas simétricas encontram-se as associações comerciais que se caracterizam pelos cartéis, pelas federações e pelos consórcios. Neste último caso, estão presentes os mecanismos de coordenação e de divisão do trabalho entre empresas/organizações, assim como os sistemas de controles controle (utilizando o conceito de Foucault e Machado, 2004) para o monitoramento dos desempenhos e participações dos diversos membros deste consórcio.

As redes burocráticas assimétricas estão relacionadas às redes de agências, aos acordos de licenciamentos e aos contratos de franquias.

O licenciamento tende a envolver algumas cláusulas de natureza organizacional, como é o caso dos serviços de assistência técnica prestados por uma rede de concessionárias de automóveis. Por sua vez, os contratos de franquia ou “*franchising*” podem ser considerados a categoria mais completa de rede burocrática, dado que através dela são estabelecidos conjuntos de procedimentos formalizados, de resultados padronizados, de sistemas de contabilidade e de treinamento de pessoal padronizados.

2.3.3 Redes Proprietárias

As redes proprietárias têm como característica principal a formalização de acordos relativos ao direito de propriedade entre os acionistas de cada uma das empresas que a compõe. Também podem ser simétricas ou assimétricas.

As redes proprietárias assimétricas são comumente encontradas nas associações do tipo capital ventures, que relacionam o investidor de um lado e a empresa parceira de outro.

Por outro lado, o exemplo mais comum de rede proprietária simétrica são as joint ventures, geralmente empregadas na regulação das atividades de pesquisa e desenvolvimento, inovação tecnológica e de sistemas de produção de alto conteúdo tecnológico.

2.4 Organizações Virtuais

Uma organização virtual caracteriza-se por um novo modelo de corporação, sustentada por uma radical mudança dos conceitos clássicos de organização e divisão do trabalho.

Para Byrne (1993), o conceito de organização virtual pode ser entendido na como uma rede temporária de organizações independentes, ligadas pela tecnologia da informação, e que em sua forma mais pura se ligam a outras para formar uma cooperação, contribuindo apenas com o que considerar sua competência central.

Observando o conceito de redes de negócios, as organizações virtuais caracterizam-se como uma forma de cooperação entre empresas ou organizações constituindo assim em redes dinâmicas de cooperação, que se viabilizam através da utilização das novas tecnologias de informação.

É uma nova maneira de estruturação, uma união com um nível significativo de cooperação, de diferentes parceiros para tirar vantagem de uma oportunidade de negócios que é conseguida com o estabelecimento da cooperação entre os mesmos e que não seria tão viável por uma companhia individual agindo por seus próprios meios.

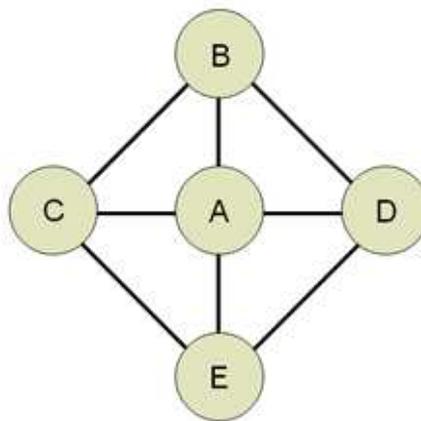
Tapscott e Williams (2007), saliente que o acesso crescente à tecnologia da informação disponibiliza as ferramentas necessárias para que os indivíduos possam colaborar criar valor e competirem em escala.

A organização virtual é uma forma distinta de organização. A OV não é somente uma propriedade de qualquer organização, mas uma forma pela qual a virtualidade é definida pelas partes envolvidas.

Uma organização virtual é definida pela combinação das melhores competências (core competency) essenciais dos parceiros legalmente independentes que cooperam entre si (HAMEL; PRAHALAD, 1995).

Essa competência central se refere a uma habilidade única para uma companhia, que não é facilmente reproduzida pelos seus competidores. Por isso, a OV tende a ser a melhor, pois reunirá as melhores competências de todas as organizações que a compõe. Alguns exemplos dessas competências compreendem habilidades organizacionais e tecnológicas, mas também podem incluir recursos humanos, conexões de redes e infra-estrutura.

Uma organização virtual é como uma rede onde os nós estão representados por uma área de especialidade. A especialidade pode ser contribuída por companhias ou pessoas. Todas as habilidades e conhecimento levados juntos fazem a empresa virtual completa para a tarefa que ela executa (TRÖGER, 1997).



Rede de Competências ou Especialidades

Figura 2: Rede de Nós por Área de Especialidade
Fonte: Strausak, 1998

Os parceiros são conectados através do uso de ferramentas de Tecnologia de Informação durante um período de tempo necessário para a realização de um objetivo específico de negócio (*business purpose*), sem considerar as fronteiras entre as mesmas.

A organização virtual é um espaço comum onde compartilham recursos, tecnologia, informação e mercado, como uma forma estratégica de aumentar a competitividade. Além disso, as organizações virtuais também dividem os riscos e os custos.

Assim sendo, nesta nova configuração de redes de empresas cada membro tem acesso aos recursos existentes em toda a rede. O risco de cada empreendedor, especialmente no caso de grandes projetos, é dividido entre os parceiros da rede.

Pelo lado do cliente final, ainda que ele visualize somente um fornecedor (não se importando quanto à forma de se construir a cadeia de valor), há a expectativa de receber produtos de menor preço e melhor qualidade, ter mais possibilidades de escolha e contar com melhores serviços.

Resumidamente, é possível definir uma Organização Virtual como uma rede temporária de instituições interdependentes, negócios ou indivíduos especializados, que trabalham juntos, de

um modo espontâneo, por meio da Tecnologia da Informação e Comunicação de forma a alcançar a ponta em uma competição existente. Eles integram-se verticalmente, unificam suas competências distintivas e funcionam como uma única organização ou unidade organizacional (STRAUSAK, 1998).

2.4.1 A Força de Trabalho nas Organizações Virtuais

Apesar do seu caráter puramente virtual, as organizações virtuais dependem, profundamente, de uma força de trabalho qualificada e treinada, capaz de tomar decisões e escolher direções. A organização virtual exige, ao mesmo tempo, competências especiais dos indivíduos que nelas trabalham e competências específicas das organizações que almejam fazer parte da rede de parcerias.

Haverá uma completa revolução nas competências dos trabalhadores, que deverão ser e estar preparados para absorver e sobreviver ao impacto dessa nova forma de organização. Muitas profissões estão fadadas ao desaparecimento e percebe-se, cada vez mais a necessidade de atualização e reabilitação profissional. É preciso estar habilitado a entender, gerenciar e responder a episódios do cotidiano do trabalho de diferentes naturezas e aplicações (BERTO, 1997).

- **Information Broker**

Principalmente na fase em que está acontecendo a procura de parceiros para a formação da empresa virtual existe uma grande quantidade de informação importante disponível. Essas informações podem ser sobre as competências principais das empresas, que são fundamentais para a escolha de parceiros apropriados.

Este contexto promove a emergência de uma nova profissão, o information broker. O information broker irá ajudar as empresas a encontrar seus parceiros mais adequados (ZIMMERMANN, 1996).

- **Business Architect**

O business architect é aquele que desenvolve os novos objetivos de negócio e que integra as pessoas e recursos para atingi-lo. O papel do business architect ganhou mais importância com

a emergência das empresas virtuais, pois, quando os objetivos de negócios eram dados pela tradicional hierarquia não necessitavam de freqüente consideração no momento em que passaram a ser dados pelas empresas virtuais, eles adquiriram uma constante redefinição voltada para qualquer oportunidade do mercado (KATZY, 1997).

O business architect descobre uma oportunidade de mercado e cuida de projetar e implementar a estrutura organizacional e industrial que seja apropriada para criar e capturar os valores do negócio. O business architect explora a mudança. O valor que ele cria para o cliente bem como para o fornecedor é gerar uma solução para a nova solicitação.

O novo nicho de mercado, os novos fornecedores ou uma nova tecnologia são a motivação para o business architect projetar e implementar uma nova empresa virtual, ou reimplementá-la, quando uma nova situação surgir.

- **Broker**

O broker em uma empresa virtual é o primeiro e mais adiantado membro da empresa. Ele age como um information broker, mas suas atividades vão além. Ele serve como um "iniciador", isto é, o impulsionador para a formação da empresa virtual. Ele procura e reconhece as oportunidades do mercado e procura e escolhe os parceiros competentes, apropriados e complementares. Este broker deverá auxiliar as empresas na busca de parceiros adequados e eventualmente coordenar as atividades de toda a empresa virtual.

O broker é o coordenador da empresa virtual, através da definição de assignments (papéis) e arranjos de negócios dos membros da empresa virtual. Ele atua como um moderador quando a empresa virtual se encontra na fase de execução/operação, resolvendo conflitos entre os parceiros, quando houver. Ele é o primeiro ponto de contato dos consumidores da empresa virtual. O papel do broker está continuamente envolvido em um número de atividades incluindo:

- Avaliar as competências dos parceiros antes e durante o desenvolvimento da empresa virtual;
- Agir como um encarregado do projeto;
- Construir e manter a confiança entre os membros da empresa virtual;
- Agir como o ponto de contato entre os membros da empresa virtual.

2.4.2 Ciclo de Vida das Organizações Virtuais

A organização virtual tipicamente dura tanto tempo quanto necessário para adquirir seus objetivos, sem compromissos de contato e/ou ligação após o término do projeto. A cada evento, a missão, as responsabilidades, as competências e os lucros são estabelecidos e compartilhados.

Quando a necessidade pela organização cessar, então ela é desmontada ou remontada, de forma a suprir outra demanda ou oportunidade do mercado, por isso em muitas definições pode ser encontrada a palavra temporária.

As fases do ciclo de vida das organizações virtuais são, de acordo com ZIMMERMANN, 1996 apud Mertens; Faisst, 1995, as seguintes:

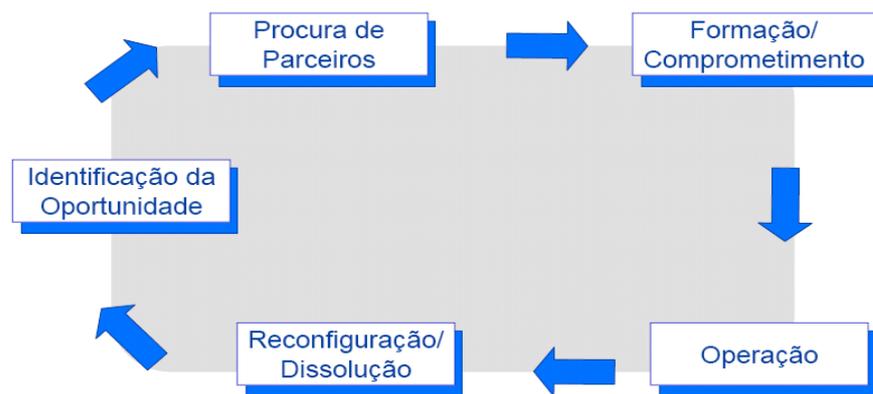


Figura 3: Ciclo de Vida de Uma Organização Virtual
Fonte: Mertens; Faisst, 1995

Tabela 2: Fases do Ciclo de Vida das Organizações Virtuais

Descrição	Características
Busca de Parceiros	A busca de parceiros deverá ser apoiada por catálogos da Internet, onde empresas apresentem suas competências essenciais. Esta fase é de fundamental importância, pois os parceiros potenciais devem ser selecionados de forma criteriosa e cuidadosa.
Contratação	A estrutura de acordos sobre a cooperação é negociada. São definidas regras para a divisão do trabalho, a designação de recursos e os procedimentos de operação, bem como a infra-estrutura necessária para a cooperação. As tarefas nesta fase são muito difíceis, porque o meio termo entre uma regulação demasiada, que reduz a flexibilidade e uma regulação superficial, que causa altos custos de coordenação, deve ser encontrado.
Operação	A fase de operação é a real fase de geração de valores da organização virtual. A fase de operação inclui coordenar a produção. Os acordos das fases de procura não são estáveis no decorrer do tempo e, por isso, devem estar permanentemente em revisão. Os parceiros individuais irão sofrer reorganização, a fim de manter o acerto com os parceiros. Se a proposta da fundação da empresa virtual é completada, a configuração corrente da empresa virtual mudará ou a empresa virtual se dissolverá.
Dissolução	<p>A organização virtual é criada para aproveitar uma oportunidade do mercado, tendo por isso um tempo de vida limitado. Uma vez atingidos os objetivos para os quais foi criada, procede-se, em geral à sua dissolução.</p> <p>A dissolução pode ocorrer na forma do final da sua existência, quando todas as relações entre os parceiros são desfeitas, não significando que os parceiros param as atividades entre eles. As relações normais entre os consumidores e fornecedores podem substituir as relações da organização virtual; ou, então, na forma do relançamento da organização virtual, significando que um número de parceiros da organização virtual está ainda convencido de que a forma cooperativa é o melhor veículo para compreender seus objetivos. Esses parceiros podem decidir mudar as regras, excluir parceiros, convidar novos parceiros ou mudar a estratégia.</p>
Reconfiguração	A Organização Virtual é por natureza dinâmica, isto é, deve poder facilmente suportar alterações de estrutura. Esta fase compreende a entrada e a saída de entidades, com a finalidade de alterar a estrutura da própria Empresa Virtual, ou simplesmente para substituir algumas das entidades participantes.

Fonte: Elaborado pelo Autor baseado em Mertens; Faisst, 1995

2.4.3 Benefícios das Organizações Virtuais

Berto (1997) apresenta algumas vantagens das empresas virtuais como a supressão de investimentos, custos fixos e variáveis de toda ordem: treinamento, manutenção de postos de trabalho em períodos sazonais de baixa demanda, subutilização da mão de obra, equipamentos, locações permanentes e estoques.

A construção modular de uma empresa virtual oferece muitas vantagens comparada com as tradicionais organizações. A flexibilidade é a palavra chave e é em áreas que demandam organizações adaptáveis que as tradicionais organizações serão deslocadas pelas corporações virtuais rápidas e oportunistas.

A cada empresa real é permitido concentrar-se na sua área de especialidade e esforçar-se para ser a melhor neste campo. Isso significa que uma corporação virtual poderia realmente adquirir as melhores competências disponíveis para cada função e, por conseguinte criar uma organização mais forte.

Uma corporação virtual pode ser extremamente flexível e adaptável. As habilidades ou funções que estiverem faltando podem ser facilmente obtidas através da adição de uma outra empresa que possua essas habilidades disponíveis, e é onde uma organização tradicional deveria ter de reconstruir fábricas para ajustar a produção para as demandas do mercado. Por outro lado, uma organização virtual poderia somente reestruturar sua organização através da contratação de um parceiro adequado.

O desenvolvimento da Internet e das tecnologias da comunicação trouxeram o conceito da não localidade, ou independência do local para o acesso à informação, encurtando distâncias e acelerando os processos de comunicação. O uso da moderna Tecnologia da Informação é muitas vezes crucial para os parceiros trabalharem harmoniosamente juntos.

A Tecnologia da Informação permite que o trabalho seja feito sobre grandes distâncias e isso irá beneficiar áreas rurais que são pobres em habitantes e em recursos. A liberdade do indivíduo é maximizada se ele puder fazer o trabalho de casa, ou seja, tele-trabalho.

Os parceiros que fazem parte de várias empresas virtuais são dependentes da Tecnologia da Informação para serem capazes de funcionar através do mundo. Isso se aplicará especialmente a corporações com foco em produtos não materialistas.

No caso de um nodo lidar com produtos físicos, o custo do transporte deterá relacionamentos de longas distâncias e dificultará a cobertura ao nível global.

Como resultado de não suportar funções periféricas, um nodo pode trabalhar em uma base eficiente de custos. Isso, por sua vez, assegura que empresa virtual aja como uma organização enxuta e bem sintonizada. Entretanto, custos adicionais para a Tecnologia da Informação devem ser considerados, bem como o emprego de tempo na coordenação pode significar perda de dinheiro.

Companhias menores podem desempenhar um papel maior e podem agir mais competitivamente quando parte de uma organização virtual. Como uma organização tradicional o custo de *overhead* para marketing e outros custos "periféricos" seriam grandes demais comparados aos lucros e certamente as companhias maiores sairiam vencedoras, como acontece hoje.

A constante mudança entre os parceiros e as corporações faz de dois competidores num dia, os parceiros do dia seguinte. O foco irá oscilar entre a competição e a procura por parceiros.

Paralelamente ao surgimento das organizações virtuais existe a evolução do trabalho, sendo que um influencia diretamente no outro. Por causa dessa co-evolução algumas profissões caem e outras novas, surgem. O surgimento dessas novas profissões pode ser considerado um benefício trazido pelas empresas virtuais, uma vez que significa uma nova oportunidade de trabalho.

2.5 A Metodologia de Desenvolvimento de Sistemas

A metodologia de desenvolvimento de sistemas envolvida no desenvolvimento de software livre é interativa e não difere muito da conhecida convencionalmente como Modelo Espiral (BOEHM, 1988). Cada fase inicia com um objetivo esperado e termina como uma revisão pelo cliente do progresso (que deve ser interna) e assim por diante.

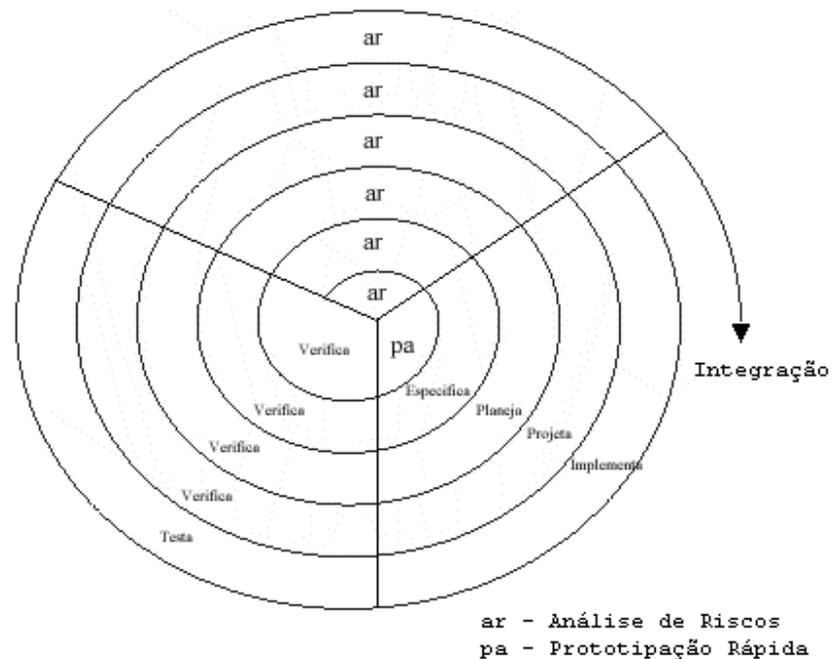


Figura 4: Modelo Espiral de Boehm
Fonte: Boehm, 1988

O processo compreende etapas como: difusão de código fonte, revisão de código e projeto, discussão pública de considerações de software, testes beta em massa, reuso generalizado de bibliotecas, organização, entre outras. A cada nova revisão e versão o processo é reiniciado, gerando assim uma espiral de desenvolvimento de software.

É importante ressaltar que a manutenção de um software utilizando este modelo de ciclo de vida é tratada da mesma forma que o desenvolvimento. Logo, a manutenção do software livre é gerada pela conformidade dos resultados dos testes e conforme a interação dos usuários com a equipe de desenvolvimento.

2.6 Gerenciamento de Projetos

O gerenciamento de projetos de *software* é uma tarefa de fundamental importância no processo de desenvolvimento de um produto, sendo definido como uma primeira camada deste processo. O gerenciamento de projeto não é visto como uma etapa clássica do processo de

desenvolvimento uma vez que ele acompanha a todas as etapas, da concepção à obtenção do produto (figura 5)

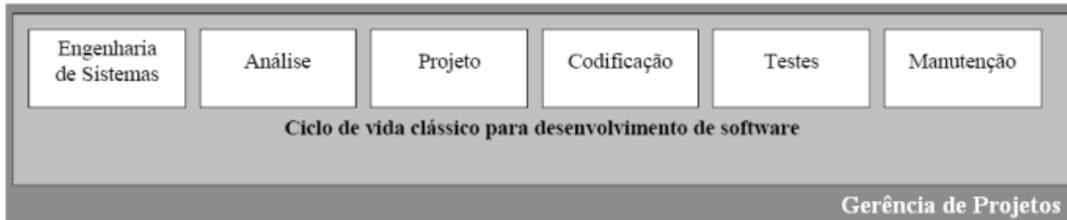


Figura 5: Áreas de Atuação da Gerência de Projetos no Contexto de Software

Para que um projeto de software seja bem sucedido, é necessário que alguns parâmetros sejam bem analisados, como por exemplo, o escopo do software, os riscos envolvidos, os recursos necessários, as tarefas a serem realizadas, os marcos de referência a serem acompanhados, os esforços (custos) aplicados e a sistemática a ser seguida.

A análise de todos estes parâmetros é a função típica do gerenciamento de projetos, função esta que se inicia antes do trabalho técnico e que prossegue à medida que o software vai se concretizando na forma de um produto.

2.7 Ferramentas de Gerenciamento de Projetos

Catapult (1995) apresenta o gráfico de Gantt e a rede PERT como ferramentas básicas que podem ajudar a obter respostas necessárias no decorrer do projeto. Com estas ferramentas, é possível observar acompanhar o desenvolvimento do projeto, através da visualização de determinadas informações através de gráficos e diagramas.

O gráfico de Gantt (Figura 5) informa para quando as tarefas estão programadas. A rede PERT ajuda a compreender as relações entre as tarefas e por que elas são programadas de determinada maneira. À medida que as necessidades de informação mudam no decorrer de um projeto, as ferramentas a serem utilizadas também mudam.

2.7.1 Gráfico de Gantt

Uma das ferramentas mais familiares para visualizar o andamento de um projeto é o Gráfico de Gantt. Como se pode ver na figura 6, o Gráfico de Gantt usa barras horizontais, cada uma representando uma única tarefa no projeto.

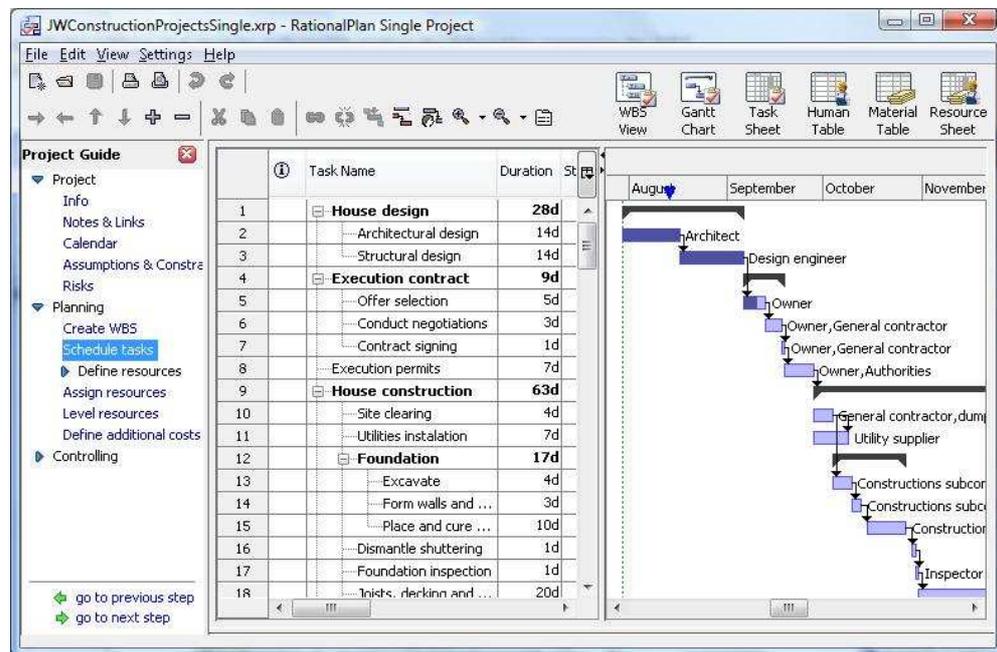


Figura 6: Gráfico de GANTT
Fonte: Adaptado de Catapult et al., 1995

Os gráficos de Gantt resumem e permitem o monitoramento de uma série de atividades que visam a consecução de um projeto, possibilitando a visualização gráfica do planejamento são, portanto o coroamento de todo o esforço precedente envolvendo as técnicas para controle de custo, desenvolvimento da programação, procura e compra de recursos e gerenciamento de riscos.

2.7.2 Redes PERT / COM

De acordo com Catapult (1995), quando é mais importante dirigir as atenções às relações das tarefas em um projeto entre si, a Rede PERT pode ser mais ilustrativo do que o Gráfico de Gantt. A Rede PERT exhibe a relação de interdependências entre as tarefas (Figura 7).

Cada tarefa é representada por um quadro, chamado nó, que contém informações básicas sobre as tarefas. As tarefas que dependem uma das outras para serem efetuadas ou que simplesmente ocorrem uma após a outra em uma seqüência de eventos são conectadas por linhas. A rede de PERT proporciona uma representação gráfica de como as tarefas são vinculadas entre si no projeto.

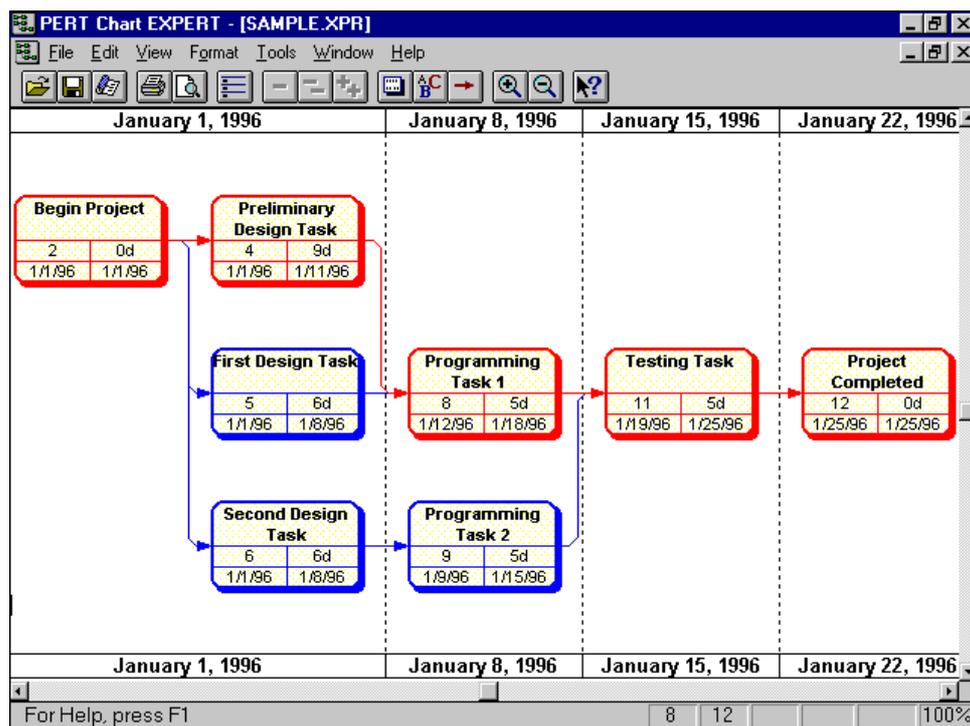


Figura 7: Redes PERT/CPM
Fonte: Adaptado de Catapult et al., 1995

As principais vantagens da utilização deste método são:

- Clareza e facilidade de compreensão das fases do projeto;
- Verificação das atividades;
- Mostrar a coerência técnica do projeto;

- Compreensão da lógica interna do projeto;
- Guia para execução e controle do projeto.

As redes PERT evidenciam relações de precedência entre atividades e permitem calcular o tempo total de duração do projeto bem como o conjunto de atividades que necessitam de atenção especial caso contrário os atrasos em sua execução causaram no projeto com um todo, este conjunto de atividades é denominado de Caminho Crítico (CPM).

2.8 A Organização da Equipe de Desenvolvimento de Software Livre

A organização de uma equipe de desenvolvimento de software livre ocorre de maneira bastante incremental e colaborativa, inicialmente de modo informal, mas que pode chegar a englobar uma estrutura bastante complexa dependendo da evolução do projeto.

O processo de desenvolvimento do software livre facilita a troca de experiências entre desenvolvedores, a evolução constante dos produtos e o intercâmbio de conhecimento entre os pares.

Apesar da imagem aparente de liberdade total e de um comprometimento parcial, não existe liberdade total no desenvolvimento. A organização do trabalho é extremamente importante, existindo hierarquia e delegação de responsabilidade.

A coordenação do trabalho dos membros da comunidade é utilizada ferramentas simples por meio da Internet, como: correio eletrônico, listas de discussão, controle de versão e acompanhamento de defeitos.

Segundo Arakaki e Turine (2005), as pessoas envolvidas em projetos de Software Livre podem ter sua participação classificada pelo tipo de papel que exercem:

- **Usuários não-participantes:** sem interesse em discutir ou ajudar no desenvolvimento do software. É nesta classe que se enquadram a maioria das pessoas. Por exemplo, fazem teste funcional do software, mas em geral não informam os erros encontrados;

- **Usuários participantes:** são os que ativamente contribuem para o projeto informando os problemas e discutindo as funcionalidades. Responsáveis por boa parte do teste funcionam provendo feedback para os autores em relação aos erros e inconsistências;
- **Desenvolvedores esporádicos:** usuários com conhecimento de programação e disposição para alterar código-fonte e produzir alterações. Normalmente essas alterações são de caráter localizado, reparando um pequeno defeito ou implementando uma pequena extensão em nível de funcionalidades;
- **Desenvolvedores ativos:** têm responsabilidade por módulos do código-fonte ou por implementar funcionalidade mais extensa ou mais complexa. Entre estes está o mantenedor central.

Tanto os desenvolvedores esporádicos como ativos participam também ativamente nas discussões, fazendo suporte ao usuário, e discutindo modificações com outros desenvolvedores sobre questões de projeto e implementação. Em projetos de dimensão um pouco maior existem listas e canais separados para discussão relacionada especificamente ao desenvolvimento. Desta forma, ficam mais isoladas as discussões entre usuários finais, que geralmente não estão interessados nas questões mais técnicas associadas ao desenvolvimento de software (REIS, 2003).

Estes membros da comunidade possuem uma cultura altamente racional, onde todas as suas ações são decididas metodicamente através de raciocínio lógico e justificativas tecnológicas. (YAMAUCHI, Y. et al., 2000).

Na grande maioria dos projetos cabe ao fundador da comunidade ou idealizador o papel de coordenação inicial do projeto, definindo as atribuições de cada um dos integrantes. Quanto maior a contribuição de um desenvolvedor da equipe, maior será sua reputação.

Com isso, no andamento normal e o eventual crescimento da equipe, surge então um conceito bastante singular presente na hierarquia do desenvolvimento de software livre: a meritocracia.

Neste conceito, a posição social/hierárquica de uma pessoa é determinada pela sua capacidade intelectual e esforço (YOUNG, 1958). As posições hierárquicas são conquistadas, em tese, com base no merecimento, e havendo uma predominância de valores associados à educação e à competência.

O grande problema do mérito do conhecimento técnico como mecanismo de distinção entre os membros num projeto de software livre é que os colaboradores mais avançados, visando manter o seu *status quo* perante a comunidade, muitas vezes acabam por dificultar a difusão de conhecimento com os novos membros (menos experientes). Nesse sentido observa-se um mecanismo de controle (FOUCAULT; MACHADO, 2004) que se imprime sobre os que não estão completamente inseridos no contexto da comunidade.

2.8.1 Hierarquia

Para fornecer uma visão hierárquica de maior organização de uma comunidade de software livre, Arakaki (2005), classifica em privilégios e níveis de poder os papéis exercidos pelos membros. Como parâmetros de referência para definir o nível em que cada papel se encontra na hierarquia de uma comunidade de software livre utilizamos: o poder de tomada de decisão e o acesso aos recursos do projeto (Tabela 3).

O poder de tomada de decisão indica quais participantes têm a competência de definir os rumos do projeto (competência de gerenciar na prática o projeto), para fornecer ou retirar responsabilidades a outros membros da comunidade.

A questão do acesso aos recursos é definida pela capacidade que os participantes possuem maiores ou menores privilégios de acesso aos recursos para o desenvolvimento. Estes recursos são: sistemas de gerência de configuração, sistemas de rastreamento e acompanhamento e gestão das mudanças.

A tabela abaixo resume s privilégios e poderes de decisão, em relação ao projeto, de cada nível hierárquico citado anteriormente:

Tabela 3: Hierarquia de Privilégios e Poderes de Decisão dentro de um Projeto de Software Livre

Nível	Privilégios	Poder
1	Todos	Todos
2	Acesso total ao repositório de dados do projeto,	Definir prioridades, decidir quais os

	administração nas listas de discussão, acesso de escrita aos arquivos da página do projeto, uso da ferramenta de rastreamento / acompanhamento de mudanças para envio de correções de bugs e modificação do estado da correção, acesso às ferramentas de geração de pacotes de versões, realização de construções do projeto no repositório principal do projetos	contribuições serão integradas ao código do projeto, definição de acesso de escrita aos recursos do projeto (repositório, página, entre outros), decisões de aspectos técnicos para evolução do projeto e participação de decisões junto à liderança do projeto.
3	Acesso total ao repositório de dados do projeto, postagem de mensagens nas listas, acesso de escrita aos arquivos da página do projeto, uso da ferramenta de rastreamento e acompanhamento de mudanças para envio de correções de bugs e modificação do estado da correção	Participar de votações (dependendo da forma de organização da comunidade) para definir prioridades no projeto (roadmaps)
4	Acesso de leitura ao repositório de dados do projeto, envio de mensagens nas listas, uso da ferramenta de rastreamento e acompanhamento de mudanças para envio de correções de bugs e modificação do estado da correção	Nenhum
5	Acesso de leitura ao repositório de dados do projeto, envio de mensagens nas listas de discussão, uso da ferramenta de rastreamento e acompanhamento de mudanças para envio de eventuais bugs	Nenhum
6	Acesso de leitura ao repositório de dados do projeto, envio de mensagens nas lista de discussão	Nenhum
7	Acesso de leitura ao repositório de dados do projeto (documentos públicos)	Nenhum

Fonte: Arakaki, 2005.

Na figura seguinte (Figura 8), apresentam-se os diversos participantes com seus devidos papéis dentro da comunidade: Quanto menor o nível em que o papel se encontra, maior o poder do indivíduo (ARAKAKI, 2005).

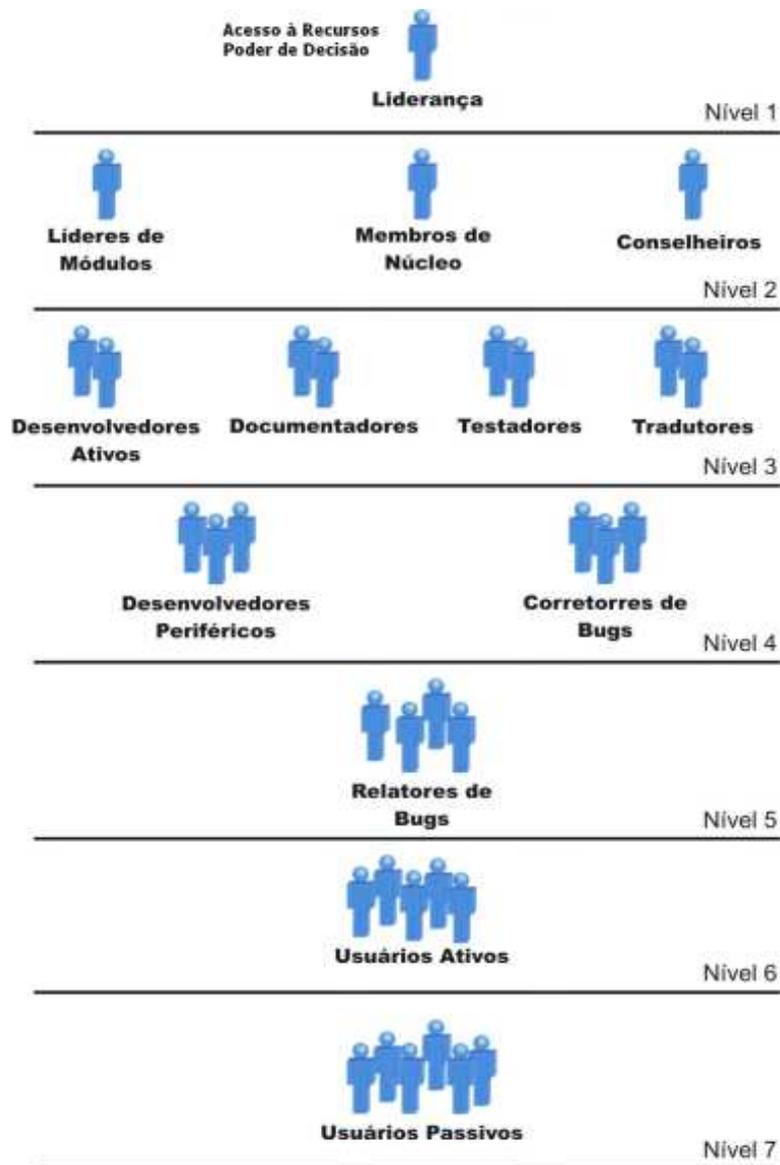


Figura 8: Papel dos Membros X Nível Hierárquico
 Fonte: Arakaki, 2005

O aspecto fundamental da hierarquia do papel dos membros da comunidade é que qualquer membro de um nível superior da hierarquia pode executar funções referentes a membros de nível inferior. Entretanto, a recíproca da execução das funções não é verdadeira.

Portanto, um membro de líder de módulo pode sugerir modificações, bem como ele mesmo implementá-las sem perder seus privilégios. O membro estará apenas executando uma função que é de responsabilidade de um outro papel. Por outro lado, os membros de nível inferior

não têm privilégios além daqueles rígidos pré-definidos para ele dentro de seu mérito dentro do projeto.

2.9 CVS: Ferramenta de Gerenciamento das Versões de Software

De nada adianta o controle hierárquico do papel dos membros do projeto ser eficiente, se não existirem mecanismos que possibilitem um controle das alterações realizadas no software, evitando assim que um membro ou mesmo uma equipe sobrescreva erroneamente o trabalho que foi feito por outra.

Neste contexto, a ferramenta normalmente utilizada pelas equipes no ciclo de desenvolvimento é o CVS (Concurrent Versions System). O CVS é uma ferramenta de controle de versão do software, que permite que os grupos de pessoas trabalhem simultaneamente no desenvolvimento dos módulos (partes do software) e outros produtos baseados em arquivos como, por exemplo, manuais, apostilas, dados de configuração, dentre outros (CEDERQVIST, 2006).

Para que seja possível o trabalho em equipe, o sistema de controle de versão pode possuir um subsistema de controle de usuários embutido ou pode utilizar algum outro sistema de autenticação separado. Assim, é possível identificar cada usuário, que geralmente fica protegido por uma senha pessoal, ou alguma senha criada pelo administrador de sistemas.

Como visto anteriormente, existe uma hierarquia que determina as atribuições de cada componente do projeto. A liberação das alterações/intervenções normalmente é feita pelo responsável do módulo trabalhado ou mesmo pelo administrador principal da versão gerada.

A ferramenta além de gerenciar o código-fonte possibilita o controle de versões para armazenar o histórico de alterações do código do projeto. Todo arquivo tem uma versão que é automaticamente definida pelo CVS.

Uma numeração intermediária ou nome simbólico é dado a uma determinada versão do projeto, pode ser usado para delimitar etapas do desenvolvimento de um projeto. A uma versão definitiva de todos os arquivos do projeto chamamos de release (liberação).

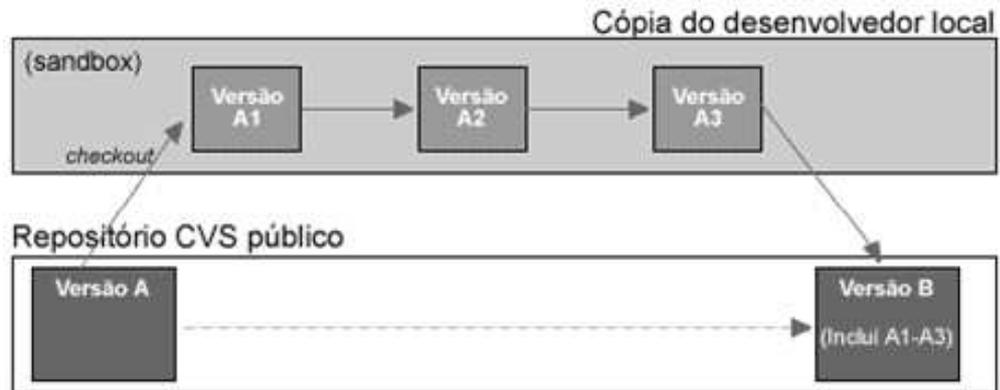


Figura 9: Diagrama funcional do modelo CVS: As versões A1-A3 existem apenas no sandbox (área local) do desenvolvedor, e ninguém mais terá acesso a elas até a liberação da versão com as implementações.

Cada desenvolvedor possui em sua máquina uma cópia local (*working copy*) somente da última versão de cada documento. Essa cópia local geralmente é feita num sistema de arquivos simples. A cada alteração relevante do desenvolvedor é necessário "atualizar" as informações do servidor submetendo (*commit*) as alterações.

O modelo básico de funcionamento do CVS permite o trabalho desconectado dos outros membros da rede, tornando possível que o desenvolvedor realize alterações em uma cópia local do código-fonte, sem necessitar estar conectado.

Além disso, o CVS não utiliza um mecanismo de exclusão mútua, o que permite que inúmeros contribuidores de um projeto modifiquem o código nas suas versões locais independentemente.

Neste contexto, existe um servidor que então guarda a nova alteração junto com todo o histórico mais antigo. Se o desenvolvedor quer atualizar sua cópia local é necessário atualizar as informações locais, e para isso é necessário baixar novidades do servidor (*update*).

Existe um repositório central com as versões mais recentes dos arquivos. Com isto, é possível criar uma cópia desses arquivos e, caso no futuro esses arquivos sejam atualizados, o CVS se encarregará de atualizar as cópias quando for necessário fazê-lo. Se novas versões destes arquivos forem colocadas no CVS nesse período de modificação, a atualização tentará o máximo possível combinar as alterações do repositório com as feitas localmente.

Cada arquivo modificado no repositório recebe uma versão numérica e seqüencial. Para cada versão é possível também criar apelidos que abstraem as versões de um conjunto de arquivos em um determinado instante do tempo. Com isso, caso seja encontrado algum erro é possível reverter alterações de arquivos individuais para versões anteriores.

Para o desenvolvedor membro da rede, o CVS permite facilmente a comparação das alterações feitas na sua cópia local (dentro do seu sandbox) com qualquer versão integrada no repositório. Isso torna possível reverter as alterações integradas no repositório central para versões anteriores.

Caso haja coincidência nas alterações e algum conflito seja criado pelas mudanças concorrentes, o CVS notificará o conflito e será necessária alguma intervenção humana para resolver este problema.

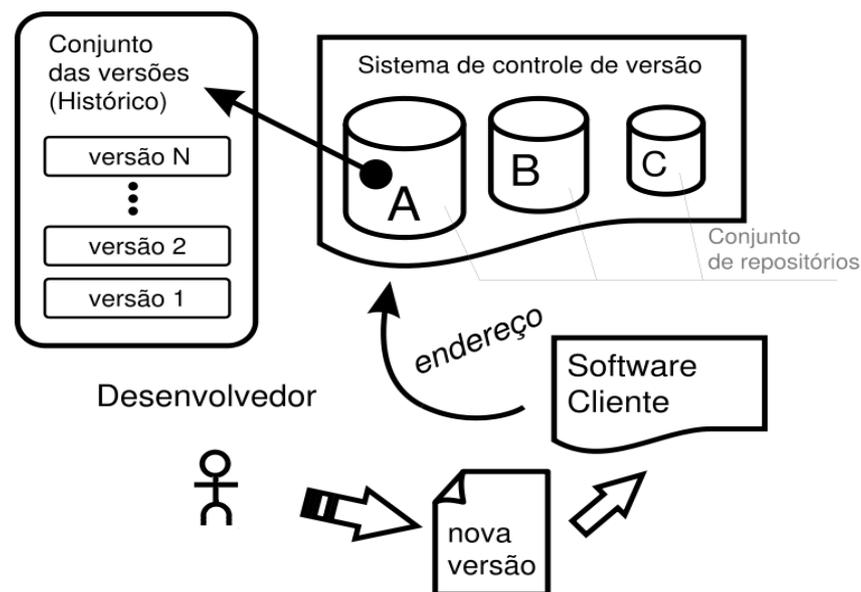


Figura 10: Esquema Lógico do Funcionamento do CVS

Fonte: <http://www-zeuthen.desy.de/dv/documentation/unixguide/infohtml/subversion/svn-book.html>

Contudo, o CVS possibilita que vários desenvolvedores executem o trabalho de modo simultâneo, resolvendo de modo muito satisfatório conflitos de versões através de diversos recursos como mesclagem e divisão de ramo do desenvolvimento (CEDERQVIST, 2006).

2.10 SVN: Ferramenta de Gerenciamento das Versões de Software

O Subversion ou SVN é um sistema de controle de versões mais moderno, que procura contornar as principais limitações do CVS. É constituído por um sistema controle de versão que permite armazenar documentos de qualquer natureza efetuando controle de acesso aos repositórios e mantendo as alterações através de um controle de revisões.

Atualmente, o SVN é uma ferramenta extensamente utilizada no desenvolvimento de projetos código aberto, tais como o Apache, Samba e GNOME. Trata-se de uma ferramenta gratuita e com código aberto que soluciona diversos problemas que antes eram comuns em outros sistemas. A ferramenta inclui todas as funcionalidades providas pelo CVS através de uma interface mais simples, facilitando a criação, edição e remoção de arquivos e diretórios.

O SVN é uma ferramenta essencial para controle de versões, que permite compartilhar arquivos, armazenar e catalogar as alterações, bem como manter um registro histórico de quem e quando as alterações foram feitas, evitar e resolver conflitos de edição. Devido ao mapeamento da estrutura do repositório, é possível (por exemplo) renomear e mover arquivos normalmente, mantendo um histórico das ações realizadas.

O elemento central do SVN é o sistema de gerenciamento e controle de versão de artefatos dispostos em um repositório. Este repositório compreende uma estrutura gerenciada pelo sistema servidor da solução.

Os documentos são recuperados do repositório e armazenados localmente, após os documentos serem alterados pode-se submetê-los ao servidor, essa atividade altera o número da revisão do repositório. O número de revisão é seqüencial iniciando em 1 e incrementada a cada alteração realizada nos documentos.

Cada documento tem armazenado no repositório sua última situação e um histórico da situação em cada uma das revisões efetuadas. Desta forma, se necessário, é possível reverter qualquer alteração ou comparar o documento em com uma revisão específica.

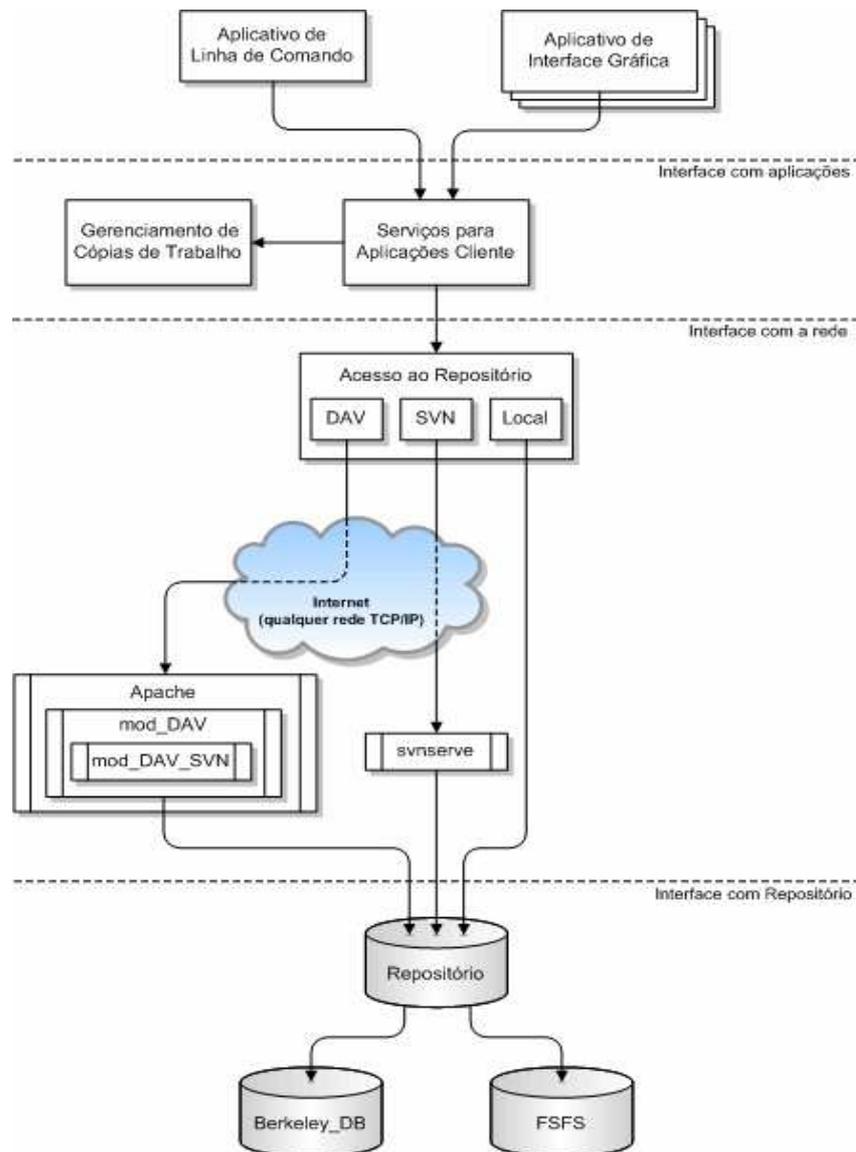


Figura 11: Arquitetura de um sistema de controle de versões Subversion (SVN)

Fonte: <http://subversion.tigris.org/faq.html>

O workflow de uso do SVN geralmente é o seguinte:

1. O repositório é criado
2. Os usuários e seus devidos níveis de acesso são determinados
3. Algum conteúdo inicial é colocado no repositório para ser gerenciado (*initial commit*)

4. Os usuários fazem uma cópia local dos arquivos para fins de consumo ou colaboração no desenvolvimento (*checkout*).
5. Os usuário com permissão para modificar o conteúdo efetivam suas alterações locais no repositório (*commit*). Esse passo pode envolver ainda diversas outras situações como verificar conflito entre o código enviado neste momento e alguma modificação realizada no repositório feita por outro usuário, entre o "*checkout*" e essa tentativa de "*commit*" das alterações locais.
6. Os demais usuários verificam modificações no repositório (*synchronization*) e atualizam suas versões locais (*update*). Se o usuário modificou algum arquivo, as situações de conflito entre modificações do conteúdo no repositório e conteúdo local do passo anterior podem ocorrer e o usuário terá como resolver as mesmas com instrumentos simples.

A arquitetura proposta pelo SVN é semelhante a um cliente-servidor, no servidor temos o repositório dos arquivos que são armazenados em um banco de dados e diversos clientes que são utilizados para acessar os documentos.

2.11 A expansibilidade dos Projetos de Software Livre

Segundo os autores Tiroler e Lerner (2002), as características favoráveis para a produção de software livre são sua modularidade, em que o projeto é dividido em partes menores e com tarefas bem definidas; e a existência de desafios a perseguir. Um projeto bem-sucedido de fonte aberta também requer um líder ou liderança confiável e uma organização consistente com a natureza do processo.

As novas tecnologias da informação, em especial a Internet, criaram uma nova maneira de elaborar o conhecimento, alterando as relações de espaço e tempo, agregando “em um mesmo ambiente de comunicação os estoques de itens de informação, as memórias, os meios de transferência e a realidade de convívio dos receptores de informação” (BARRETO, 2002, p. 6).

Neste contexto, o princípio do desenvolvimento de software livre é semelhante ao processo do desenvolvimento científico. Ao compartilhar o conhecimento, resultados e métodos,

estes podem ser justificados e reproduzidos. Outros podem dar contribuições, respondendo ou continuando o trabalho publicado, impulsionando o desenvolvimento da ciência.

De modo semelhante, ao escrever um software e o oferecer à comunidade, sua contribuição é revisada por outros, e se for boa o bastante, o autor consegue os créditos na economia de contribuição da comunidade de fonte aberta.

Portanto, um dos grandes argumentos dos entusiastas do software livre é a capacidade de expansão de um projeto devido a esse compartilhamento. Devido ao acesso ilimitado de agentes a recursos e a projetos e, em consequência, após algum tempo usando a Internet, os diversos agentes realizam uma auto-alocação mais eficiente dos recursos, com comunicação menos ineficiente e uma depuração mais robusta dos programas.

Segundo, BENKLER (2007), enquanto softwares livres / abertos são muito visíveis eles são de fato uns exemplos de um fenômeno sócio-econômico muito mais abrangente. Este modo de produção (do software livre/aberto) tem vantagens sistemáticas sobre as firmas e nos mercados por duas razões:

- Primeiro este modo é melhor na identificação e na delegação do capital humano à informação e ao processo de produção cultural.
- Segundo, há retornos incrementais substanciais em termos de eficiência de alocação ao permitir que grandes blocos de contribuidores potenciais interajam com grandes blocos de recursos e informação na busca por novos projetos e oportunidades de colaboração. A robustez é consequência direta da revisão dos programas por muitos e diferentes agentes.

2.12 A Produção de Software Livre como Organização Virtual

Analisando a definição de organizações virtuais como “uma rede (temporária) de organismos independentes, ligados através das tecnologias de informação, com vista a partilharem competências, recursos, custos e os espaços de intervenção de cada um” (MATOS, 1997, p. 23) na produção de um bem ou serviço (produto), observamos uma clara relação com a produção de software livre.

Segundo Reis (2003, p. 37),

Projeto de Software Livre é uma organização composta por um conjunto de pessoas que usa e desenvolve um único Software Livre, contribuindo para uma base comum de código-fonte e conhecimento. Este grupo terá à sua disposição ferramentas de comunicação e trabalho colaborativo, e um conjunto de experiências e opiniões técnicas que usam para discutir o andamento do projeto.

Na criação de Software Livre, cada um dos elementos presentes nestas definições pode ser caracterizado como:

Tabela 4: Elementos e Contexto na Organização Virtual

Elemento	Contexto do Software Livre
Rede Temporária	Rede de relacionamento (Comunidade)
Organismos Independentes	Diferentes desenvolvedores, projetistas e analistas de sistemas.
Competências	Conhecimento específico de cada membro da equipe.

Fonte: Elaborado pelo autor com base em Reis, 2003.

Nesse contexto, a produção de software livre poderia de uma estrutura que atenda essas condições, possibilitando a integração horizontal dos recursos, onde os recursos de hardware e software são integrados e gerenciados por meio de um sistema de grade de computadores (Grid) que faz alocação de recursos para determinada aplicação.

Como o gerenciamento desta aplicação e possui interfaces padronizadas para alocação de outras aplicações, gerando um melhor uso dos recursos disponíveis (menor custo).

O agrupamento de diversas competências necessárias ao redor de uma comunidade em rede, os conhecimentos pré-existentes disponíveis no código-fonte de software e a qualidade técnica como resultado final da junção destes ingredientes trazem maior probabilidade de acumulação futura do progresso técnico, fomentando inovações constantes e em seqüência (MENDES, 2006).

Ian Foster (2001) em seu artigo “The anatomy of the grid” define este Grid como sendo compartilhamento coordenado de recursos e solucionador de problemas em instituições ou organizações virtuais dinâmicas.

O grid torna-se então uma poderosa plataforma de suporte a estas organizações que podem assim melhorar dramaticamente sua qualidade e velocidade de produtos e serviços disponibilizados, enquanto os custos de Tecnologia de Informação são reduzidos por habilitar a colaboração transparente dos recursos compartilhados.

A rede temporária é estabelecida através deste grid utilizando uma ferramenta de controle de versões CVS para que as diversas competências sejam adicionadas ao projeto de forma organizada. Como visto anteriormente neste artigo, a ferramenta CVS permite que a equipe trabalhe simultaneamente no desenvolvimento nas partes do software.

2.13 A Estratégia de Comunicação nesta Organização Virtual

Nos projetos de Software Livre, as partes então se comunicam virtualmente utilizando a Tecnologia de Informação como ferramenta principal. Assim sendo, isso vem de encontro às Organizações Virtuais. Estas se apresentam como um novo modelo organizacional que utilizam a tecnologia para unir, de forma dinâmica, pessoas, bens e idéias sem, todavia, ser necessário reuni-las em um mesmo espaço físico e/ou ao mesmo tempo (BERTO, 1997).

A troca de experiências na comunidade é feita utilizando softwares de mensagens instantâneas e utilizando correio eletrônico. Estas ferramentas estão amplamente popularizadas e são utilizadas no dia-a-dia, como um meio de comunicação e troca de arquivos de forma ágil entre os membros.

Além disso, as comunidades se utilizam de ferramentas para apoio como listas de discussão. As listas servem como um meio de comunicação entre os membros do projeto. Através delas os membros de um projeto podem discutir sobre o desenvolvimento, aprimoramento, a utilização, o lançamento de novas versões e qualquer outro tópico relevante ao projeto.

Além disso, as listas de discussão possibilitam o arquivamento da comunicação para análise futura sendo uma fonte importante de documentação para os usuários, já que frequentemente apresentam problemas já resolvidos anteriormente.

2.14 O Controle das Organizações Virtuais de Software Livre

Se pensarmos na figura do *Broker*, que é o primeiro e mais adiantado membro da organização, fica clara que no contexto do software livre essa função é do idealizador ou fundador da comunidade. Ele executa essa função de controle do projeto, administrando o andamento do projeto através da ferramenta CVS.

Entretanto, é importante ressaltar que essa coordenação dos atores envolvidos ocorre por meio de sentimento de co-dependência, baseado em confiança mútua. Neste contexto, os projetos de software livre diferem de outros tipos de organizações virtuais pela importância colocada no controle social (FOUCAULT; MACHADO, 2004).

Especificamente, identificaram-se quatro grupos de mecanismos de controle que têm sido usados nos projetos de fonte aberta: A importância da reputação individual; A administração dos membros, que regulam quem deve participar; Regras e instituições; Sanções, ou ameaças de sanções, contra comportamentos indesejáveis (GALLIVAN, 2001).

Além disso, para Gallivan (2001) existem três tipos de confiança nos projetos de software livre:

- 1) Confiança na liderança, em que os colaboradores potenciais devem acreditar nos compromissos de longo prazo e na qualidade da liderança – confiança e fidelidade são aspectos importantes no crescimento de um projeto de fonte aberta;
- 2) Confiança no julgamento de outros, que é fundamental no processo de avaliação – o programador confia na comunidade de usuários que testam os códigos de software identificando possíveis problemas e apresentando relatórios de erros;
- 3) Confiança em uma comunidade de estranhos, isto é, nos desenvolvedores de software que contribuem com sua “propriedade intelectual” para projetos de fonte aberta e se beneficiam diretamente de seus trabalhos por também serem usuários desses produtos.

2.15 O Licenciamento de Software Livre

O tipo de licenciamento é um fator de grande importância no universo do software livre. Nesse universo existem as mais diferentes visões sobre o modelo ideal para o software livre e isso se reflete também nos mais diversos tipos de licenças, cada uma com suas particularidades individuais.

Entretanto, para que não se perca o foco desta pesquisa a forma de licenciamento aqui analisada é a GPL (General Public License) que além de ser o tipo de licença adotado nos projetos estudados é também o tipo licença de software livre mais utilizada no mundo.

A General Public License (Licença Pública Geral), é a especificação da licença para software livre idealizada no final dos anos 80 para GNU da Free Software Foundation. A característica primordial deste tipo de licença é que qualquer trabalho derivado de um software livre sob GPL, obrigatoriamente deve também permanecer livre.

Diferentemente das tradicionais licenças de software comerciais, as licenças livres são regidas por quatro liberdades principais (GNU, 2007):

- **Liberdade nº 0:** A liberdade de executar o programa, para qualquer propósito;
- **Liberdade nº 1:** A liberdade de estudar na totalidade como o programa funciona e adaptá-lo às suas necessidades. O acesso integral ao código-fonte é um pré-requisito básico do software;
- **Liberdade nº 2:** A liberdade de redistribuir cópias de modo que você possa ajudar ao seu próximo;
- **Liberdade nº 3:** A liberdade de aperfeiçoar o programa, e liberar os seus aperfeiçoamentos, de modo que toda a comunidade em geral possa também se beneficiar destas funcionalidades.

De acordo com estas liberdades, este tipo de licenciamento permite que os softwares distribuídos possam ser reaproveitados (evoluídos), mantendo, porém, os direitos do autor de forma a garantir que essa informação não seja usada de uma maneira que possa limitar as liberdades originais.

Este tipo de licença não permite, por exemplo, que o código fonte do software seja apoderado por outra pessoa, ou que sejam impostos sobre ele restrições que impeçam que este continue a ser distribuído da mesma maneira pela qual a licença original foi adquirida.

A designação mais comum a este tipo de licenciamento é de *copyleft*, que se consolida como uma forma de usar a legislação de proteção dos direitos autorais com o objetivo de retirar barreiras à utilização, difusão e modificação do código fonte, mas utilizando das normas de propriedade intelectual sob a ótica das quatro regras vistas anteriormente. O *copyleft* pode ser dividido em duas especificações, conforme suas liberdades:

- **Copyleft cheio:** Todas as partes de um trabalho (exceto a licença própria) podem ser modificadas por autores consecutivos;
- **Copyleft parcial:** Apenas algumas liberdades podem ser aplicadas ao trabalho, mas não todas. Por exemplo: Uma obra é licenciada com direitos de uso e distribuição mas sem os direitos de modificação.

Uma característica singular da GPL é que o texto original em inglês é atualmente a única forma aceita pela Free Software Foundation para que eventuais interpretações (ou erros) no processo de tradução não possam modificar de maneira alguma o sentido real desta forma de licenciamento (GNU, 2007).

2.16 Motivação, Software Livre e a Cooperação de Projetos

O desenvolvedor de software livre é movido principalmente pela vontade de compartilhar e cooperar (GNU, 2007). É através desse trabalho colaborativo e compartilhado que o software é desenvolvido. O processo de desenvolvimento de software livre envolve mais do que questões técnicas, envolve principalmente questões ideológicas (um quadro que resume as motivações do desenvolvedor de software livre pode ser observado na Figura 12).



Figura 12: Quadro de Esquemático das Motivações dos Projetos de Software Livre
 Fonte: Free Software Foundation, 2007.

Segundo Stallman (2002) muito esforço tem sido depreendido pelas comunidades de software livre para se conquistar mais usuários ou tornar os usuários mais ativos, porém não se vê o mesmo esforço para divulgar a ideologia que está por trás do software livre. O autor considera que muitos usuários têm utilizado softwares livres apenas por motivos técnicos. Entretanto, o autor alerta que os usuários devem estar conscientes da necessidade de defender sua liberdade frente aos desafios que o software livre precisa enfrentar.

De acordo com estudos a maioria dos participantes deste tipo de cooperação tem a participação na comunidade de software livre como uma atividade secundária, gastando menos de 10 horas por semana em trabalhos relacionados ao projeto (TAURION, 2004). Essa participação ocorre principalmente nos horários de folga da atividade que provê a subsistência deste indivíduo. Este baixo nível de participação introduz problemas de comunicação, já que muitos participantes têm dificuldade em acompanhar todos os desenvolvimentos realizados no projeto. Essas circunstâncias certamente dificultam o foco dos participantes no desenvolvimento do projeto.

O esforço dispensado em estar a par de todas as questões toma uma proporção que sobra pouco tempo para as contribuições efetivas. Muitos participantes perdem o interesse ou são absorvidos pelos comprometimentos externos, deixando muitos projetos pela metade. Assim, o trabalho é conduzido aos pedaços e finalizado apenas após um longo período como uma “colcha de retalhos”. Mas por que isso ocorre?

Quanto à motivação Abraham Maslow (1970), apresenta uma teoria que propõe um tipo de classificação ou hierarquia das necessidades inerentes do ser humano, com a qual colaborou para o conhecimento da própria natureza da motivação do trabalho humano (Figura 12).



Figura 13: Teoria das Necessidades do Ser Humano
Fonte: Maslow, 1970.

Nesse contexto, a motivação da pessoa é formada por cinco conjuntos de necessidades que são atendidas de maneira decrescente num homem saudável: fisiológicas, segurança, amor, estima e auto-realização.

Para o autor, esse comportamento motivacional depende do atendimento das denominadas necessidades humanas, que são forças internas que atuam sobre todos os indivíduos. Portanto, as outras prioridades que garantem a subsistência do membro da comunidade de software indivíduo acabam sobrepondo o “espírito do software livre” e isso afeta diretamente sua motivação, e conseqüentemente, a participação.

2.17 Partindo para a Pesquisa de Campo

A fundamentação teórica apresentada nos tópicos deste capítulo buscou principalmente fornecer o embasamento bibliográfico necessário à solidificação dos conceitos visando construção do modelo de gestão de projetos cooperados.

A partir deste momento, esta pesquisa passa então a abordar os procedimentos metodológicos adotados (capítulo 3) no estudo dos três casos (capítulo 4), que servirão como base (juntamente a fundamentação teórica) para construção do modelo proposto (capítulo 5).

3 PROCEDIMENTOS METODOLÓGICOS

O capítulo atual aborda a metodologia utilizada na preparação e no desenvolvimento deste trabalho, com o objetivo de dar-lhe um embasamento teórico e científico para o alcance dos objetivos propostos nesta pesquisa.

A caracterização de um trabalho científico é realizada através da utilização de metodologias. Estas têm como função nortear os rumos do projeto de forma a nortearem o desenvolvimento da pesquisa, sua sistematização, coleta de dados e avaliação de resultados.

Inicialmente, com o objetivo de realizar uma pesquisa englobando os diversos aspectos de projetos cooperados foi realizada a seleção dos projetos. O objetivo desta seleção foi escolher projetos de portes e objetivos distintos.

É importante ressaltar que a análise sobre a abrangência e o porte destes projetos são realizadas no capítulo 4. A nomenclatura e classificação básica dos projetos que foram selecionados podem ser observada na tabela 5:

Tabela 5: Caracterização dos Projetos Estudados

Projeto	Porte	Abrangência
Firebird Database	Médio (50 participantes)	Mundial
GNOME	Grande (Mais de 300 participantes)	Europa e Américas
TinyCobolBR	Pequeno (5 participantes)	Regional (Brasil)

Além disso, o processo de seleção dos projetos cooperados de software livre respeitou obviamente as limitações inerentes à pesquisa de campo segundo os seguintes critérios:

- A caracterização do projeto à proposta de pesquisa;

- A vocação do projeto (comercial, profissional ou pessoal);
- A viabilidade de acesso do pesquisador aos membros do projeto a serem entrevistados (ou disponibilidade de meio digital no projeto para tanto);
- A disponibilidade dos membros do projeto em fornecer outras informações necessárias ao pesquisador;
- A relevância dos projetos perante as comunidades de desenvolvimento de software livre.

3.1 Caracterização do Estudo

Conforme visto anteriormente no capítulo 1, esta pesquisa busca o estabelecimento de um modelo de gestão para projetos de cooperação utilizando como base o modelo de desenvolvimento do software livre.

Para tanto, esta pesquisa envolveu a obtenção de dados descritos sobre o ambiente da pesquisa (comunidades de software livre), sujeitos, e processos interativos pelo contato direto sobre a situação estudada, com o intuito de compreender os fenômenos segundo a perspectiva dos participantes da situação em estudo, com a escolha intencional deste universo estudado (MARCONI; LAKATOS, 1996).

Segundo os autores torna-se fundamental que o “pesquisador faça um levantamento dos temas e tipos de abordagens já trabalhados por outros estudiosos, assimilando os conceitos e explorando os aspectos já publicados”.

Portanto, quanto à natureza, esta pesquisa pode ser classificada como de perfil quali-quantitativo. Godoy (1995) demonstra que a pesquisa qualitativa surge de questões ou focos de interesses amplos, e que se delineiam de acordo com o desenvolvimento do estudo.

A parte quantitativa da pesquisa ocorreu através de questionários (instrumentos padronizados) as quais são mais adequadas para apurar opiniões e atitudes explícitas conscientes dos entrevistados (MARCONI; LAKATOS, 1996).

Assim, uma aplicação bastante usual das pesquisas exploratórias é o estudo de caso. Neste tipo de pesquisa, a fundamentação teórica e análise dos resultados são respaldadas na pesquisa bibliográfica.

Portanto, o método investigativo utilizado nesta dissertação foi o estudo de caso através de uma abordagem descritivo-avaliativa (MERRIAM, 1988), visando a modelagem dos processos de negócio relacionado aos projetos virtuais de cooperação estudados.

Ainda segundo Merriam (1988), este método é a uma estratégia de pesquisa que é pautada pelo exame de um fenômeno contemporâneo dentro de seu contexto e que busca descrever e analisar intensivamente na totalidade: Uma entidade, fenômeno ou unidade social. A coleta estruturada de dados compreende a utilização de um questionário formal que apresenta questões em uma ordem predeterminada.

Segundo Salomon (2001, p. 158) “as pesquisas exploratórias e descritivas têm por objetivo definir melhor o problema, proporcionar as chamadas intuições de solução, descrever comportamentos de fenômenos, definir e classificar fatos e variáveis”.

A pesquisa bibliográfica, segundo Barros e Lehfeld (2000, p. 70) é definida como aquela que “efetua tentando-se resolver um problema ou adquirir conhecimentos do emprego predominante de informações advindas de material gráfico, sonoro e informatizado.”

Do ponto de vista de procedimentos pode-se dizer que este trabalho de pesquisa está baseado no método de investigação do estudo de caso, envolvendo múltiplos casos baseando no referencial teórico da pesquisa bibliográfica. Yin (2003 p.33) menciona que a pesquisa de estudo de caso pode ser composta por estudos de caso único quanto de casos múltiplos.

O estudo de caso é uma indagação empírica que: investiga um fenômeno contemporâneo dentro do seu contexto de vida real; quando as fronteiras entre fenômeno e contexto não são claramente evidentes; e no qual múltiplas fontes de evidencia são utilizadas. (YIN, 1989, p. 23).

Segundo Merriam (1988), este método é a uma estratégia de pesquisa que realiza o exame de um fenômeno contemporâneo dentro de seu contexto e que busca descrever e analisar na totalidade uma entidade, fenômeno ou unidade social.

3.2 Procedimento de Coleta de Dados

A coleta de dados foi realizada no período de 10 de março a 4 de julho de 2009, utilizando um questionário estruturado destinado a provocar informações específicas dos entrevistados (*survey*).

Segundo Malhotra (2001, p. 179),

o método *survey* para obtenção de informações se baseia no interrogatório dos participantes, aos quais se fazem várias perguntas sobre o seu comportamento, intenções, atitudes, percepção, motivações e características demográficas e de estilo de vida.

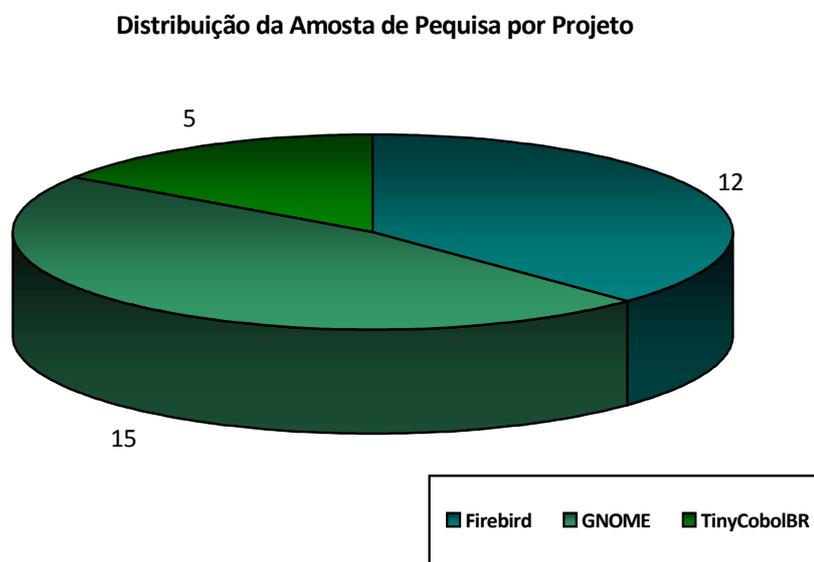


Gráfico 14: Distribuição da Amostra da Pesquisa por Projeto

Estes questionários foram enviados de forma eletrônica (e-mail) a 65 membros das comunidades estudadas. Desse número de questionários enviados, foram respondidos 32 formulários (totalizando um índice de respostas de 49,2%). A distribuição entre projetos dos questionários recebidos, e conseqüentemente avaliados pode ser observada no gráfico 14.

Em dois casos particulares, nos quais os coordenadores de projetos foram entrevistados, a entrevista ocorreu de forma direta através de um encontro pessoal do pesquisador com o entrevistado.

A primeira entrevista ocorreu no dia 10 de Junho de 2009 com o senhor Fernando Wuthstrack, atual coordenador do projeto TinyCobolBR, na cidade de Joinville, Santa Catarina.

A segunda entrevista com o senhor Vlad Khorsun, ocorreu no dia 4 de Julho de 2009, aproveitando o 6º Firebird Developers Day, onde o entrevistado veio da Rússia

para fazer uma participação especial como palestrante, ocorrido em Piracicaba, São Paulo.

Ambos os entrevistados, além de responderem ao questionário forneceram suas impressões pessoais (mais subjetivas) sobre os projetos e também contribuiriam com sugestões para o estudo apresentado por esta dissertação.

3.3 Desenvolvimento do Estudo

Para uma melhor compreensão e conseqüentemente uma análise efetiva dos dados optou-se por análise do conteúdo de discurso, conforme Bardin (1977). Conceitualmente essa análise de conteúdo é um conjunto de técnicas de análise visando, por procedimentos sistemáticos e objetivos de descrição do conteúdo das mensagens, obter indicadores (quantitativos ou não), que permitam a inferência de conhecimentos relativos às condições de produção ou recepção dessas mensagens.

A análise e interpretação dos resultados não foram baseadas em medidas quantitativas (ainda que para o objetivo final tenham sido usados também alguns dados quantitativos complementares), mas prioritariamente na percepção dos significados dos comportamentos dos sujeitos, coletados sistematicamente e reavaliados durante todo o processo de investigação.

O trabalho de pesquisa foi apoiado por entrevistas estruturadas, realizadas a partir de um questionário pré-definido, que está disponível na seção de apêndices desta dissertação.

3.4 Validação do Questionário de Pesquisa

A elaboração do questionário de pesquisa aplicado ocorreu por meio de questões fechadas, com a possibilidade de uma única ou múltiplas escolhas, e questões abertas, com o visando identificar questões mais subjetivas da a opinião do membro do projeto cooperado sobre um determinado assunto.

Estas questões foram elaboradas visando responder às questões levantadas pela temática e o problema de pesquisa. Além disso, para a elaboração das questões foi realizado um processo de pesquisa nas listas de discussão dos projetos para observar possíveis questões relevantes que não haviam sido previstas pelo pesquisados.

Após a elaboração das questões, foi realizado um teste preliminar com um membro dos projetos estudados (coordenador do TinyCobolBR), esse teste ocorreu com o intuito de validar o mesmo, com a identificação de possíveis problemas de compreensão e a verificação da adequação das questões aos objetivos da pesquisa.

O teste preliminar foi realizado utilizando-se o mesmo procedimento a ser aplicado no decorrer da pesquisa. Portanto, o envio do questionário ocorreu antecipadamente ao membro do projeto, sendo que para o envio e recebimento das questões utilizou-se como recurso o correio eletrônico (e-mail).

A após o recebimento das respostas do questionário, a entrevista seguiu com o questionamento junto ao especialista sobre as dificuldades encontradas para a interpretação das respostas, com o objetivo de realizar a validação do questionário. Este questionamento foi bastante proveitoso, pois apontou a necessidade da realização de algumas pequenas correções no formato de algumas questões.

3.5 Limitações do Estudo

Em se tratando de amostra intencional na estatística, a pesquisa fica limitada a escolha subjetiva dos entrevistados. Portanto, os procedimentos metodológicos aplicados foram um dos fatores limitantes da pesquisa, pois o grupo de amostra (32 membros dos projetos) foi um corpo de amostra reduzido, considerando também a subjetividade da respostas apresentadas por meio da entrevista semi-estruturada.

Esta condição poderá estabelecer um resultado mais consistente em estudos posteriores se puder ser aplicada a um maior número de membros de projetos cooperados de software. Portanto, as conclusões advindas do caso estudado só se aplicam a ele mesmo. A proposta apresentada, entretanto, pode servir como modelo de referência para outros estudos complementares.

Outro fator importante a ser destacado é que o método das entrevistas pode apresentar as seguintes desvantagens: gerar problemas de comunicação entre o entrevistador e entrevistado, atrasos na pesquisa devido ao tempo que o processo de entrevista consome e também este método pode gerar um custo mais elevado, o que normalmente conduz o pesquisador a utilização de amostras pequenas (MARCONI; LAKATOS, 1996).

Estas condições ocorreram principalmente em função do prazo de entrega e conclusão da dissertação ser reduzido para realizar todas as atividades mencionadas anteriormente num espectro tão amplo de amostra e também pela demora no recebimento das repostas por parte dos entrevistados.

Outro fator relevante diz respeito ao conhecimento limitado do pesquisador acerca da temática, da dinâmica geral do processo e do quadro teórico utilizado. Estes fatores constituem então uma limitação ao trabalho de pesquisa realizado.

4 ESTUDO DE PROJETOS COOPERADOS EM SOFTWARE LIVRE

O capítulo atual realiza a análise qualitativa dos projetos cooperados apresentando também a quantificação dos resultados da pesquisa de campo. Através desta análise realiza-se a caracterização do projeto para entender os mecanismos de cooperação utilizados.

Ao final de cada processo de análise, apresenta-se então o modelo conceitual do funcionamento do projeto, que servirá como base para criação do modelo proposto por esta pesquisa para gestão de projetos cooperados.

4.1 Análise do Projeto Firebird

O projeto Firebird constitui-se numa comunidade com um número pequeno de contribuidores, mas que está bastante ativa a julgar pelo número de versões e revisões do software apresentadas. O produto final da cooperação da comunidade é um sistema gerenciador de banco de dados utilizado em diversas instituições no mundo inteiro.

4.1.1 Sistemas Gerenciadores de Banco de Dados

A partir da década de 60, com o advento da aplicação comercial dos primeiros computadores digitais, as organizações passam a adotar o processo contínuo de digitalização dos seus dados operacionais.

Segundo Pereira (1996), computacionalmente, esses dados representam uma abstração de parte da realidade – representam algumas características selecionadas das entidades do mundo real, necessárias para a solução de um problema.

Inicialmente, os dados das corporações eram guardados em arquivos e a sua consulta e manipulação era uma tarefa bastante complexa. Além disso, os dados “brutos” necessitam de um processamento a partir de um conhecimento para que possam gerar informação.

Segundo Van Bommel (1999), o dado é a representação de observações ou conceitos apropriados para comunicação, interpretação e processamento por humanos ou máquinas. Dados interpretados formam as informações.

Por sua vez Teorey (1999), corrobora com a afirmação anterior, afirmando que o dado é a descrição de algum fenômeno do mundo real, já a informação é conhecimento adquirido pelo uso e interpretação do dado.

Completando estas afirmações Pereira (1996) afirma que o dado é aquilo que entra em um processo informatizado e informação é aquilo que é apresentado como resultado.

A informação registrada digitalmente adquire um papel decisivo nas corporações, pois como argumenta Robredo (2003, p. 9), esta informação pode ser:

[...] Registrada de diversas formas; duplicada e reproduzida; transmitida por diversos meios; conservada e armazenada em suportes diversos; medida e quantificada; adicionada a outras informações; organizada, processada e reorganizada segundo diversos critérios; recuperada quando necessário segundo regras preestabelecidas.

Isto posto, com o crescimento do volume destes dados e da necessidade da transformação dos mesmos em informação, surgem na década de década de 70, os primeiros sistemas responsáveis por organizar essa panacéia: Os Sistemas Gerenciadores de Bancos de Dados.

Para Teorey (1999, p. 10):

Um sistema gerenciador de banco de dados (SGBD) é um software genérico para manipulação de banco de dados. Um SGBD é capaz de tratar os aspectos de visão lógica e física dos dados, suportar linguagens para definição e manipulação de dados, bem como prover utilitários para o gerenciamento de transações e controle de concorrência.

Portanto, os sistemas gerenciadores de bancos de dados são sistemas de computação que possuem estruturas organizadas de dados que proporcionam formas ágeis de recuperação de informações.

Neste contexto, um sistema gerenciador de banco de dados é um software que permite a posterior recuperação das informações com as funções de disponibilizar operações de armazenamento e recuperação de dados, controle de acesso e de transações (VAN BEMMEL, 1999).

Os sistemas gerenciadores de bancos de dados são repositórios que não possuem a função específica de armazenagem; podem recuperar, efetuar *backup* em tempo real e manter a segurança dos dados muitas vezes críticos para os negócios das empresas (CANTU, 2005).

Completando esta definição, Korth e Silberschatz (1995), afirmam que o grande objetivo de um Sistema de Gerenciamento de Banco de Dados é prover os usuários com uma visão abstrata dos dados. Cabe ao SGBD omitir certos detalhes de como os dados são armazenados e mantidos para esconder a complexidade das estruturas de dados utilizadas internamente pelo sistema.

Além disso, estes autores afirmam que um sistema de banco de dados é projetado para armazenar grandes volumes de informações. O gerenciamento de informações implica a definição das estruturas de armazenamento destas e o fornecimento de mecanismos para sua manipulação.

Um sistema gerenciador de banco de dados (SGBD) é uma coleção programas que permite aos usuários criar e manter um banco de dados (...) é um software de propósito geral que facilita processos de definição, construção, manipulação e compartilhamento bancos de dados entre usuários e aplicações. (ELMASRI; NAVATHE, 2005, p. 5).

O sistema de banco de dados precisa também proporcionar segurança ao armazenamento de informações, diante de falhas do sistema ou acesso não autorizado. (KORTH; SILBERSCHATZ, 1995).

4.1.2 Histórico

Normalmente um projeto de software livre segue a filosofia “open source” desde sua gênese de duas formas: uma iniciativa individual que desperta o interesse de outros desenvolvedores e/ou um projeto coletivo.

Entretanto, o projeto Firebird é um caso bastante atípico deste tipo de projeto cooperado. O projeto se enquadra num projeto coletivo, mas com um foco inicial comercial – diferente da maioria dos projetos de software livre.

Em 1984, surge um protótipo de um sistema gerenciador de banco de dados comercial, denominado de “Groton Database System”, originalmente concebido e criado por um grupo de engenheiros de sistemas, funcionários da DEC (Digital Equipment Corporation).

O desejo inicial era produzir um sistema gerenciador de banco de dados inovador e que substancialmente oferecesse maiores benefícios que os outros sistemas até então existentes como, por exemplo, a compatibilidade com a linguagem estruturada de consultas (SQL).

Na época de sua concepção, muita tecnologia já existia como proposta de inovação, mas nenhum produto a oferecia como recurso ou característica disponível.

Em 1991 a companhia foi comprada pela Ashton Tate (responsável por produtos como o dBase) e o sistema passa a denominar-se Interbase Database. Logo após, a empresa foi adquirida pela empresa Borland, como parte de uma negociação de produtos.

Logo, a empresa passou a integrar o Interbase à suas ferramentas de desenvolvimento de software

Entre 1991 até 1999 a Borland responsabilizou-se pelo desenvolvimento do Interbase. Em 1999, a Borland resolveu abrir o código fonte do Interbase de forma que desenvolvedores em todo o mundo pudessem trabalhar no seu desenvolvimento.

O InterBase versão 6.0 tornou-se um produto de código aberto em Julho de 2000, depois de inúmeros conflitos entre o time de desenvolvimento do InterBase e a visão comercial dos executivos da Borland.

Surge então a InterBase Company, formada pelos idealizadores do InterBase e seus colaboradores iniciais, com o objetivo de dirigir o novo projeto de software livre com apoio da Borland.

Aparentemente arrependida com a opção de disponibilizar da opção de abertura do código fonte, a empresa coloca diversos obstáculos ao processo de abertura do código fonte (FREITAS; PARIS, 2007).

Em 2001, a Borland volta com versões comerciais, tendo então duas versões do Interbase 6, a open source e a comercial. Com isso, surge o Firebird, um projeto de um gerenciador de banco de dados poderoso baseado no código fonte do Interbase. A primeira versão do Firebird foi lançada em março de 2002 sendo licenciado sob a IPL (InterBase Public License), a qual tem os mesmos termos da Mozilla Public License 1.1. (FREITAS; PARIS, 2007).

Apesar de não mais receber apoio da Borland, que restabeleceu o desenvolvimento comercial do InterBase, a comunidade de software livre rapidamente organizou o projeto denominado Firebird, na conhecida plataforma de suporte aos projetos de código fonte – sourceforge.net¹, com o intuito de dar continuidade ao desenvolvimento baseado no código fonte do InterBase 6.0.

¹ O SourceForge é uma plataforma de tecnologia de informação de controle de desenvolvimento colaborativo de software. Fornece ferramentas que permitem a uma comunidade global de desenvolvedores de software o controle e manutenção de projetos open source, além de atuar como um repositório de código fonte. Possui atualmente mais de um milhão de usuários registrados na plataforma.

Nos últimos anos, o Firebird tem se mantido como uma excelente opção de banco de dados gratuito e de alto nível. Mais de 30 desenvolvedores colaboram constantemente na manutenção e crescimento do projeto.

Para administrar e garantir os rumos do projeto, foi criada a força motriz por trás do projeto Firebird: A Fundação Firebird SQL. É um grupo aberto de indivíduos e empresas que patrocinam o desenvolvimento do Firebird por meio de coleta de fundos e garantias para que os desenvolvedores ajudem a evoluir e continuar o projeto (COMUNIDADE FIREBIRD, 2009).

4.1.3 O Firebird no Brasil

No Brasil, desde a criação do projeto, existem diversas organizações que utilizam o Firebird como solução de banco de dados pela qualidade e confiabilidade do software.

Diversas empresas nacionais adotaram o Firebird em seus projetos, dentre as quais é possível destacar: a Caixa Econômica Federal, a Embrapa (Empresa Brasileira de Pesquisa Agropecuária), a Clínica do Leite/USP, e centenas de empresas de desenvolvimento de software que adotaram o Firebird como banco de dados de suas aplicações (CANTU, 2005).

Além disso, o país conta com diversos colaboradores ativos que atuam tanto no desenvolvimento das novas funcionalidades como também na manutenção do código fonte atual. Outro aspecto interessante é que o país conta também com membros votantes no comitê gestor do projeto.

4.1.4 A Fundação Firebird

A base de dados Firebird tem uma forte ligação com a sua comunidade de desenvolvimento e utilizadores. A Fundação Firebird é a organização desses indivíduos ou empresas de várias dimensões, trabalhando em conjunto como parte de uma organização sem fins lucrativos com a finalidade central de assegurar a continuação do projeto e dos produtos correlacionados (COMUNIDADE FIREBIRD, 2009).

Os assuntos da fundação são geridos por um Comitê de Gestão, que é eleito democraticamente durante a reunião anual de gestão. Os membros deste comitê são eleitos pelos membros com direito à voto associados ao projeto. Estes membros com direito a voto por sua vez são definidos conforme a participação dos membros no projeto.

O Comitê de Gestão governa de acordo com as regras definidas pela fundação, e tem ainda poderes de criar e delegar novas funções em sub-comitês, à medida que seja necessária a sua criação. Estes sub-comitês são responsáveis por funcionalidades específicas que não afetam diretamente o objetivo central do projeto (COMUNIDADE FIREBIRD, 2009).

O comitê mantém o fundo financeiro que sustenta o desenvolvimento do projeto, responsável inclusive pelo custeio dos desenvolvedores remunerados que trabalham exclusivamente na manutenção do projeto.

As reuniões da fundação são basicamente realizadas através de ferramentas de tecnologia de informação e são geralmente conduzidas num fórum no site do próprio projeto.

A topologia da rede de cooperação estabelecida e as relações entre os atores envolvidos na comunidade Firebird podem ser observados na Figura 15.

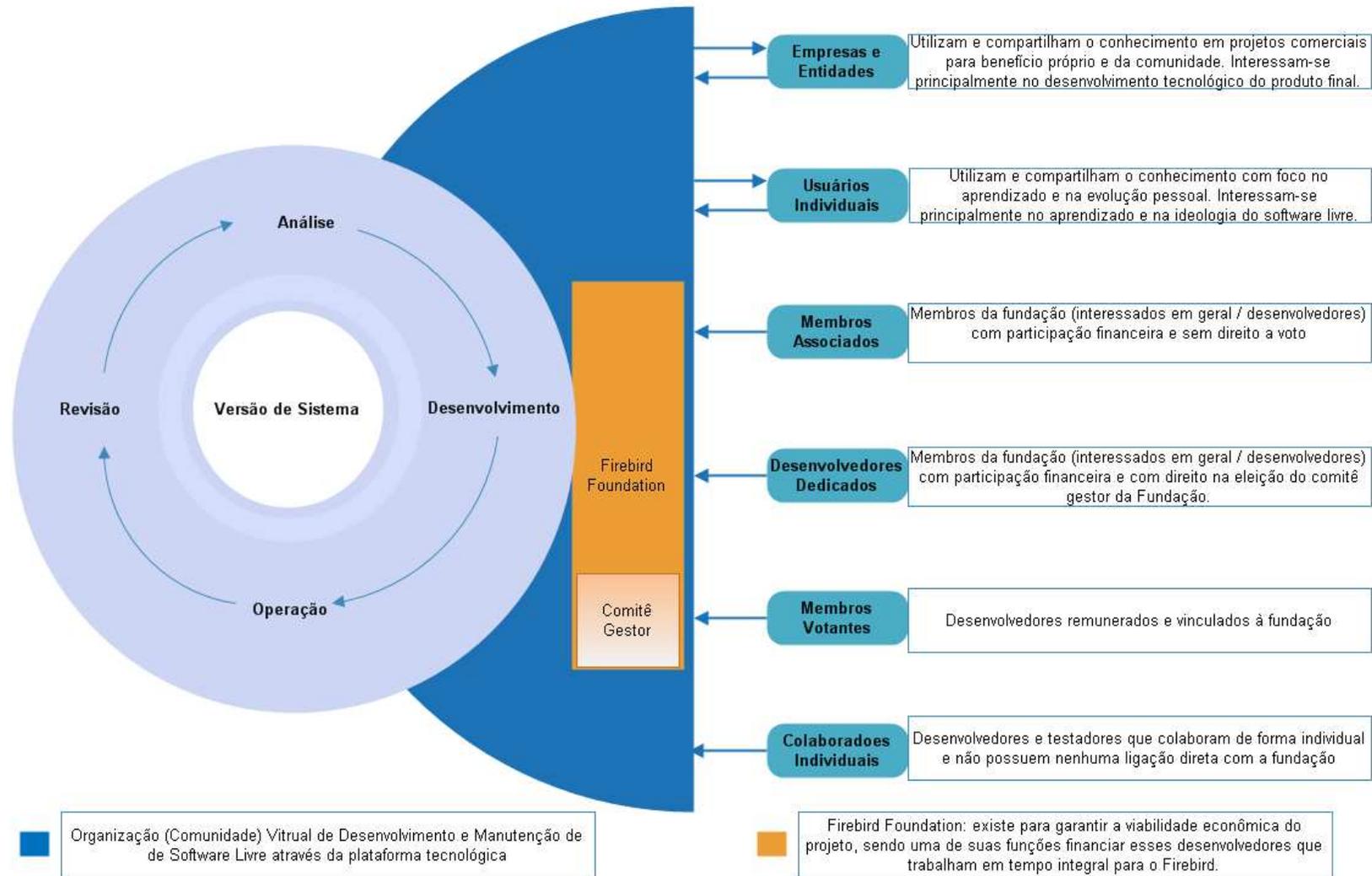


Figura 15: Topologia do Projeto Firebird

4.1.5 Resultados da Pesquisa de Campo

Durante o processo de pesquisa, foram entrevistados 15 participantes ativos da comunidade de desenvolvimento do projeto Firebird. Dentre os mesmos, foi possível entrevistar pessoalmente os dois dos mais respeitados líderes do projeto: Vlad Khorsun e Dmitry Yemanov.

Através da análise quantitativa destas repostas, foi possível traçar o perfil destes membros e também identificar como ocorre a gestão do projeto.

4.1.6 Fatores de Motivação

Inicialmente, para a melhor compreensão do perfil do desenvolvedor dentro do projeto de software livre Firebird, buscou-se identificar quais os fatores que motivam o desenvolvedor a participar da comunidade.

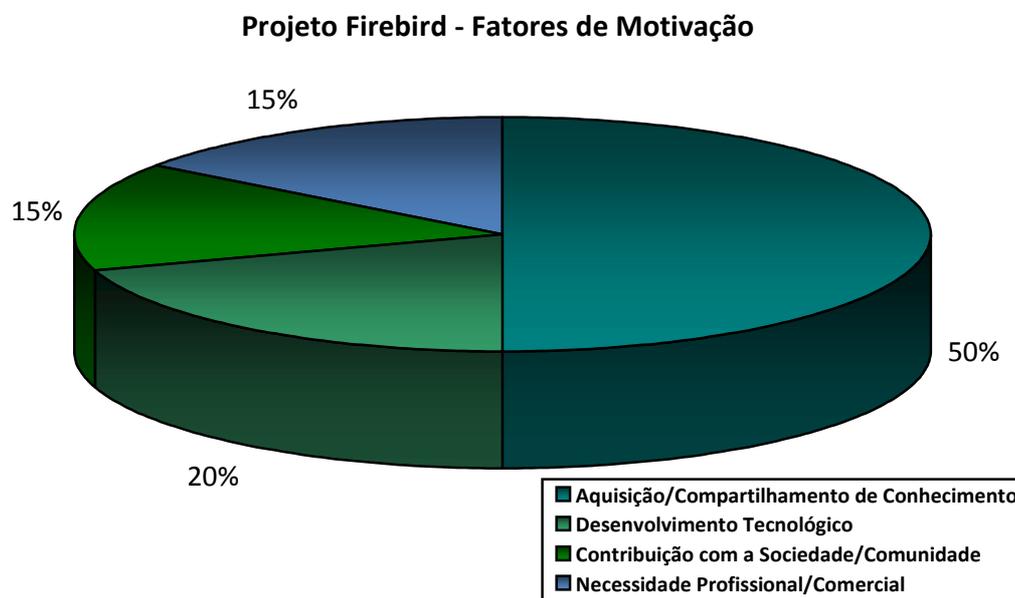


Gráfico 16: Fatores de Motivação dos Membros do Projeto Firebird

Para Chiavenato (1982, p. 414):

Para compreender o comportamento humano é fundamental o conhecimento da motivação humana. Motivo é tudo aquilo que impulsiona a pessoa a agir de determinada forma, isto é, tudo aquilo que dá origem a alguma propensão a um comportamento específico.

Segundo Silveira (2004, p. 41), o “trabalho colaborativo e em rede é a essência do desenvolvimento do software livre”. O desenvolvimento colaborativo traz a possibilidade de compartilhar e de multiplicar a informação e o conhecimento, bens não-rivais e não esgotáveis cuja circulação é restringida pelo modelo de apropriação das patentes e direito de autor utilizado pela indústria de software.

A adoção do modelo de desenvolvimento colaborativo, que envolve grande número de desenvolvedores voluntários, possibilita considerável qualidade técnica do software desenvolvido, o que por sua vez “atrai novos usuários, vários dos quais passam a agir como testadores e desenvolvedores do sistema. Esta atuação produz melhorias na qualidade do sistema, o que acaba por atrair novos usuários” (HEXSEL, 2002).

Neste contexto, a motivação é um fator imprescindível que se reveste de grande importância o processo gerencial do Firebird. É interessante destacar, que no caso do projeto Firebird, como se trata de um software bastante específico (aplicação de banco de dados), a motivação é um fator ainda mais importante, porque os resultados de uma colaboração muitas vezes não são visíveis à comunidade em geral.

O fator preponderante apontado por 50% dos entrevistados é que os mesmos participam da comunidade para que possam adquirir e compartilhar o seu conhecimento com os outros membros e com a sociedade em geral (Gráfico 16). Assim, é possível notar a influência dos fatores ideológicos do software livre, onde a questão da evolução do indivíduo e do compartilhamento dos seus conhecimentos se faz presente.

Este fator está diretamente relacionado com o fator de desenvolvimento tecnológico, que é apontado por 20% dos membros pesquisados como fator preponderante da sua participação. Para estes membros, o sentimento de estar participando do desenvolvimento de uma tecnologia é sua grande realização. Entretanto, foi possível observar que os membros do projeto entrevistados, acabam de certa forma confundindo o próprio projeto.

Segundo 15% dos entrevistados, a sua participação no projeto é motivada pelo desejo de contribuir com a sociedade através de um espírito altruísta de colaboração em massa por um objetivo maior.

Assim, o fator social é a sua principal motivação já que o software livre constrói um patrimônio comum de toda sociedade na forma de conhecimento. Esse patrimônio comum permite, por exemplo, que o conhecimento seja assimilado de forma muito mais fácil pelos agentes sociais. Com isso, a possibilidade de inovação torna-se acessível a todos e não apenas àqueles que controlam privadamente determinado rol de conhecimentos (INSTITUTO NACIONAL DE TECNOLOGIA DE INFORMAÇÃO, 2005)

Finalmente, o fator profissional como motivação está presente e, 15% do universo dos membros. Estes membros conheceram o projeto profissionalmente e acabaram

participando deste para aprimorarem a ferramenta com que estão trabalhando. Normalmente esta participação ocorre na busca de uma funcionalidade que necessitavam e que não estava presente, ou mesmo para melhorar alguma característica do software.

É interessante destacar que esta participação só é possível pela própria característica básica do software livre. Se ao invés de um projeto “open source”, o desenvolvedor estivesse utilizando uma solução de código fonte fechado, a única alternativa seria solicitar a modificação aos mantenedores do código fonte.

Assim, esta necessidade só poderia ser atendida se existisse algum interesse comercial ou pessoal do criador do projeto nesta modificação, independente da opinião deste ou demais utilizadores da solução.

4.1.7 Gestão das Modificações

Conforme visto anteriormente, a participação do indivíduo na comunidade é uma das principais características do desenvolvimento de software livre. O software produzido é um resultado de inúmeras contribuições e re-contribuições de membros de todas as partes do planeta (HEXSEL, 2002).

Para compreender como ocorrem estas modificações no projeto Firebird, foi realizado também, um questionamento aos membros de como ocorre o processo de gestão das modificações e atualizações a serem realizadas no projeto (Gráfico 17).

Assim, o resultado encontrado, apresenta-se coerente com esta realidade, visto que a para a grande maioria dos membros do projeto (60%), as decisões ocorrem de forma comunitária. Essas decisões são discutidas através dos fóruns de discussão com a participação aberta a todos os membros.

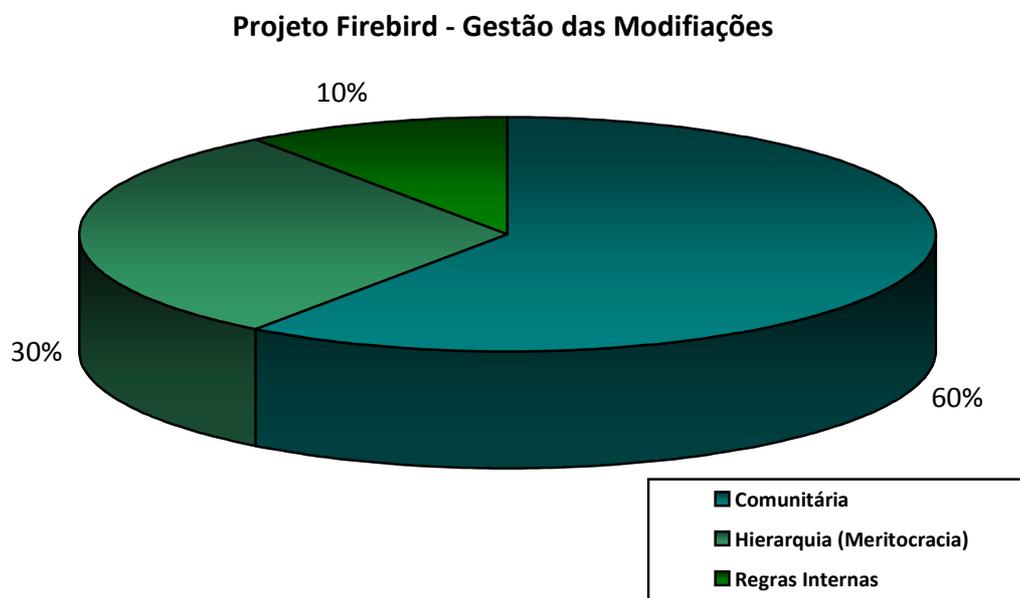


Gráfico 17: Gestão das Modificações do Projeto Firebird

Entretanto, foi possível identificar que uma parte considerável dos entrevistados (30%) apontou que existe certo grau de hierarquia na decisão final pela modificações.

Essa hierarquia ocorre principalmente pela importância que um determinado membro possui no círculo de discussão da modificação corrente. Ou seja, as modificações são discutidas em grupo, mas a em alguns casos a “palavra final” pode ser dada por um membro mais antigo ou que já possui algum mérito dentro da comunidade (meritocracia).

4.1.8 Estratégias de Comunicação

O desenvolvimento das tecnologias de informação e comunicação levou a um processo nunca outrora vislumbrado de interconexão das pessoas e organizações, possibilitando um espaço comum de colaboração em massa (TAPSCOTT, 2007).

Mais especificamente a popularização da Internet proporcionou a consolidação da difusão e troca de conhecimentos entre grupos de trabalho. Neste contexto, os membros da comunidade Firebird foram questionados sobre quais são as tecnologias e ferramentas de comunicação utilizadas (Gráfico 17).

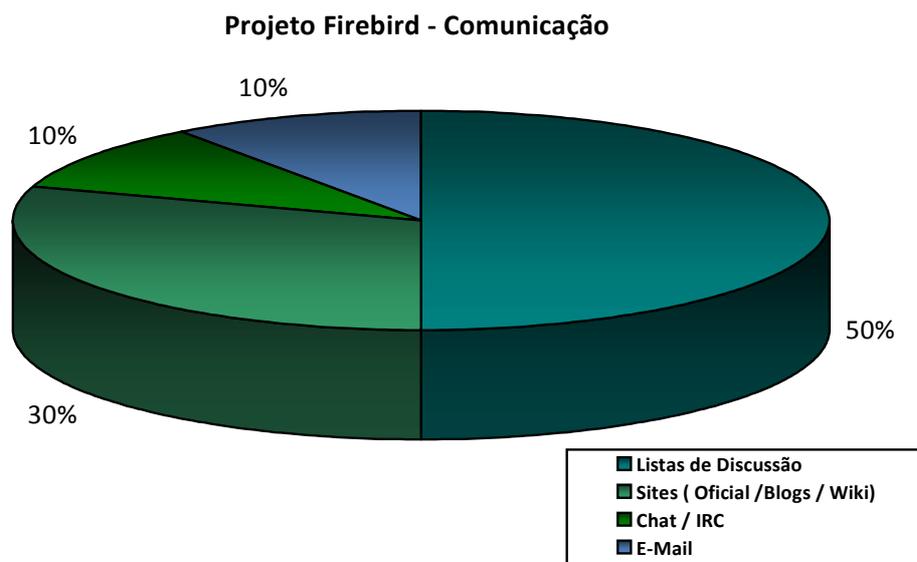


Gráfico 18: Ferramentas e Tecnologias de Comunicação Adotadas pelos Membros do Projeto Firebird

O recurso tecnológico mais relevante adotado na comunidade Firebird segundo os membros entrevistados foi a lista de discussão (50%). Esta foi apontada como um espaço democrático onde os membros podem além de discutir os rumos das modificações e inovações. Segundo os entrevistados, esta ferramenta exerce um papel preponderante por

possibilitar um registro histórico dessas ponderações que pode ser acessado a qualquer momento, servindo como uma importante base de conhecimento.

Outro aspecto relevante da utilização da lista de discussão é que os membros da comunidade procuram desenvolver uma linguagem textual que tenta representar a emoção e a realidade. Assim, estas formas são utilizadas para trazer riqueza de sentido às falas. Ruoso (2006, p. 25), no seu trabalho etnográfico das metodologias de gestão do projeto Debian, ressalta este aspecto afirmando que:

Uma vez que no mundo virtual o corpo não pode mais ser utilizado como meio de comunicação, outras formas são encontradas para trazer riqueza de sentido às falas. E mesmo que se pudesse utilizá-lo, hoje já se estabeleceu uma forma complexa de comunicação que não seria possível de se representar com voz ou mesmo com vídeo. A simplicidade do texto permitiu que a habilidade de utilizar as listas de discussão se desenvolvesse com características que não seriam possíveis na interação face-a-face.

Além da lista de discussão, para 30% dos entrevistados, os sites abertos à comunidade, (na forma de página oficial do projeto, blog e Wiki) são também relevantes e possibilitam um canal de comunicação não tão flexível quanto a lista de discussão, mas com um papel similarmente importante.

Finalmente, 10% membros entrevistados apontaram as tradicionais formas de comunicação como Chat (conversa on-line) e E-mail como os meios de comunicação mais relevantes. De certa forma, o E-mail é utilizado também para as listas de discussão do projeto, entretanto, quando foi questionado sobre o recurso E-mail, buscou se a sua utilização como forma de comunicação mais direta.

4.1.9 Dificuldades no Projeto

A afirmação de Vlad Khorsun², um dos membros mais ativos do projeto Firebird, “[...] a demanda por modificações e funcionalidades é muito grande, porém o número de pessoas capacitadas que contribuem efetivamente para a continuidade do projeto é muito baixo para esta demanda [...]” resume perfeitamente a compilação dos resultados encontrados.

A pouca “mão de obra” (no sentido de software livre) e os recursos limitados são apontados pela grande maioria (80%) dos membros entrevistados como as principais dificuldades encontradas (Gráfico 19).

Segundo Taurion (2004), a maioria dos projetos de software livre depende de poucas pessoas já que estatisticamente 10% dos participantes contribuem com 78% do código; o segundo décimo, com 12%, o terceiro, com 3%; os outros, com menos de 1% cada.

Além disso, o baixo comprometimento com o projeto corrobora com a afirmação citada anteriormente, já que este é um fator apontado por 10% dos membros entrevistados.

² Essa declaração foi concedida por Vlad Khorsun ao pesquisador durante entrevista realizada no dia 04/07/2009 durante o 6º Firebird Developers Day em Piracicaba/SP.

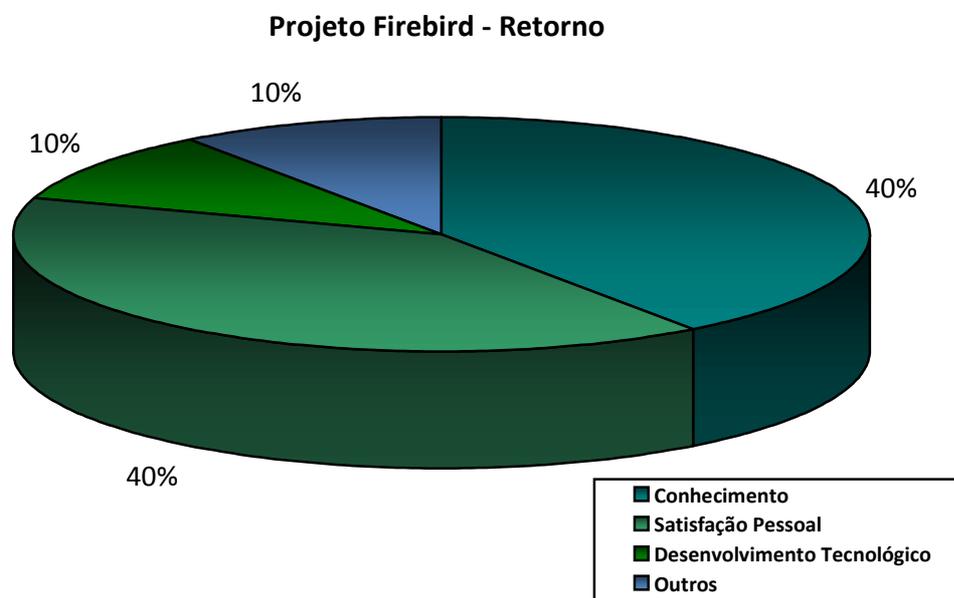


Gráfico 19: Dificuldades Encontradas pelos Membros da Comunidade

Para Taurion (2004), nota-se um maior interesse da comunidade de desenvolvedores em participar de projetos de software básico ou em projetos com grande visibilidade, de softwares que tem como usuários o grande público (em que se incluem os desenvolvedores), tais como softwares de multimídia, editores de texto e navegadores.

Finalmente, os conflitos internos (10%), como por exemplo, a discordância de metas a serem alcançadas, também são apontados como dificuldades encontradas.

4.1.10 Retorno do Projeto

Acima de qualquer outro valor, foi possível observar nos depoimentos obtidos, que os valores inerentes ao software livre estão presentes no espírito de cada membro atuante na comunidade. Por isso, foi possível notar que o retorno obtido no projeto é

acima de tudo um retorno pessoal inerente a motivação pessoal do membro da comunidade.

Portanto, uma questão relevante abordada foi a questão de qual é o retorno que o membro recebe pela sua participação na comunidade de desenvolvimento de software livre (Gráfico 20).

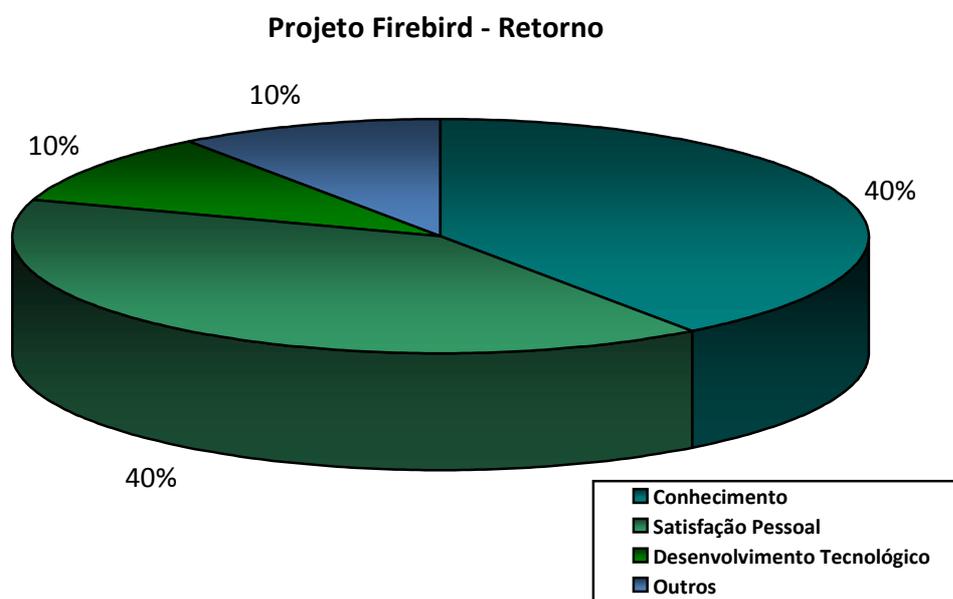


Gráfico 20: Retorno dos Membros da Comunidade

Os resultados da pesquisa demonstram que o retorno obtido pelo membro do projeto Firebird está relacionado ao benefício intelectual do próprio indivíduo. Assim, 40% dos entrevistados indicaram que o conhecimento obtido é um fator de retorno relevante.

Além disso, para outros 40% dos entrevistados, a satisfação pessoal é o principal retorno obtido. Cada desafio enfrentado torna-se uma missão pessoal do contribuidor e observa-se que cada solução encontrada é uma vitória que atribui um mérito especial a ele perante a comunidade. Para estes membros a satisfação pessoal está relacionada a poder contribuir em algo que venha ser utilizado por todos.

Segundo 10% dos entrevistados, o desenvolvimento tecnológico como produto final da cooperação justifica e recompensa o contribuidor. O membro da comunidade se satisfaz em poder criar algo totalmente novo ou por melhorar alguma solução pré-existente. Esta nova tecnologia é algo que passa a recompensar a contribuição do indivíduo principalmente pelo pioneirismo que este pode usufruir numa nova tecnologia deste mundo virtual.

Finalmente, outros motivos como recompensa profissional, vocação, fatores pessoais e até mesmo aceitação pela comunidade constituem-se como relevantes para 10% dos entrevistados.

Tabela 6: Compilação dos Resultados do Questionário de Pesquisa – Firebird

Motivação	<ul style="list-style-type: none"> - Cooperativismo - Participar de algo que será útil a diversas pessoas no mundo inteiro - Desafio - Desenvolvimento do conhecimento
Decisões	<p>As decisões de nível operacional são tomadas nas listas de discussões antes do trabalho ser realizado. Geralmente busca-se um consenso, abrindo mão de alguns pontos e adotando outros.</p> <p>Participantes das decisões: Os líderes de desenvolvimento, com o feedback dos desenvolvedores, a equipe de administração, a comunidade e recentemente um Technical Task Group com representantes de patrocinadores da Firebird Foundation.</p>
Meio de Comunicação	<ul style="list-style-type: none"> - Fórum de Discussão - Encontros de Desenvolvedores (Firebird Developers Day)
Gestão dos Recursos	<p>A Firebird Foundation administra os recursos e oferece as garantias para que os desenvolvedores ajudem a evoluir e continuar o projeto.</p>
Hierarquia	<p>Dependendo da intensidade da participação do membro (meritocracia), mais ou menos influência este terá no Software e na Comunidade.</p>
TICs Utilizadas	<ul style="list-style-type: none"> - Fórum de Discussão - Website - E-mail - Ferramentas de mensagens instantâneas

Desenvolvimento Do Software	Todos os membros podem se envolver no desenvolvimento de soluções para qualquer parte do software – Sem entraves burocráticos para modificar qualquer parte do software
Difusão do Conhecimento	Como a maioria das discussões são públicas, todos podem aprender. Os desenvolvedores de hoje são pessoas que apenas observavam o que era discutido no passado.
Ferramentas de Gestão	Apenas ferramentas do SourceForge.net como o CVS (Controle de versões e modificações), Bugtracker (jira).

4.2 Análise do Projeto GNOME

O projeto GNOME é uma comunidade com um número grande de contribuidores, e que está muito ativa a julgar pelo número de versões e revisões do software apresentadas recentemente. O produto final da cooperação da comunidade é um excelente ambiente gráfico de trabalho completo para computadores de todos os tipos.

4.2.1 Histórico

O GNOME (GNU Network Object Model Environment) é um projeto internacional de colaboração para o desenvolvimento de um software livre que envolve mais de 400 contribuintes diretos das mais diversas nacionalidades.

O Projeto originou-se em agosto de 1997, como parte do projeto de um sistema operacional totalmente livre (GPL³). Neste tipo de sistema, qualquer pessoa obtém o direito de usar, modificar e redistribuir, o programa, seu código fonte, desde que garantindo para todos a mesma filosofia de direitos.

Seguindo a vocação dos projetos cooperados, o anúncio oficial da sua criação ocorreu através de uma ferramenta de tecnologia de informação. Assim, esta comunicação inicial ocorreu através das principais listas de discussão dos principais projetos de software livre da época tais como: Debian, Gimp e KDE.

Em pouco tempo, o projeto atraiu a atenção de desenvolvedores, usuários e até mesmo hackers. Assim, poucos meses após sua gênese, o Gnome se consolidava como um dos projetos de software livre mais badalados da Internet, sendo que em dezembro de 1997, ocorreu lançamento público da sua primeira versão (versão 0.10).

A partir de 1998, além de usuários isolados, diversas empresas começaram a investir no projeto visando a criação de uma interface gráfica que facilitasse o uso de sistemas Unix e requisito básico para criar uma alternativa viável em relação aos sistemas operacionais comerciais. O sucesso do projeto ocorre de tal forma, que diversos desenvolvedores passam a se dedicar exclusivamente na melhoria contínua do sistema.

Até 2000, o projeto sofre um crescimento exponencial em participação e importância, com dezenas de softwares e centenas de colaboradores. Isto leva ao aumento do interesse de grandes empresas de tecnologia, que vislumbram uma oportunidade grande oportunidade de negócios.

Entretanto, esta cobiça começou a ameaçar a filosofia e as metas fundamentais do Gnome como projeto de software livre. Em resposta a esta realidade, os membros do projeto decidiram criar uma Fundação (Gnome Foundation), com o objetivo de garantir a orientação ideológica inicial do projeto.

³ A General Public Licence (GPL) é um tipo de licença que garante ao usuário o direito de copiar ou modificar um trabalho.

A Fundação é responsável por definir as metas gerais do projeto, gerenciar e organizar o processo de desenvolvimento, fazer o papel de porta-voz do projeto para a imprensa e interface institucional às outras organizações.

É composta por todos os membros colaboradores efetivos do Gnome e administrada por sete diretores (Board of Directors) que são eleitos democraticamente por votação direta dos membros. Além disso, a Fundação conta com um quadro de organizações parceiras (Advisory Board). Este quadro é composto por empresas e organizações sem fins lucrativos que de maneira direta ou indireta apóiam o projeto. O Brasil tem uma participação muito expressiva na fundação com 11 membros ativos no conselho.

Como forma de amenizar alguns interesses comerciais, foi determinada que as organizações que fazem o Advisory Board - dentre elas diversas organizações de grande porte - não tenham nenhum poder deliberativo dentro da Fundação. Entretanto, a opinião destas organizações pode ser levada em consideração através do papel de um conselho paralelo (conselheiros).

A partir do ano 2000 em diante, diversas empresas, fornecedoras do sistema operacional Unix, como a Sun Microsystems, a IBM e a HP se unem à Fundação Gnome de fortalecê-lo com padrão.

A partir 2004, em plena era da mobilidade, a Nokia inicia um forte investimento em tecnologias para dispositivos moveis baseados em GNOME e GNU/Linux para serem utilizadas em seus produtos. Assim, a inserção do GNOME neste amplo contexto, garante que o papel do projeto na democratização do uso de sistemas operacionais alcance uma relevância ainda maior no campo das tecnologias de informação nos anos seguintes.

A topologia da rede de cooperação estabelecida e as relações entre os atores envolvidos na comunidade GNOME (incluindo a participação do quadro consultivo) podem ser observados em detalhes na Figura 21.

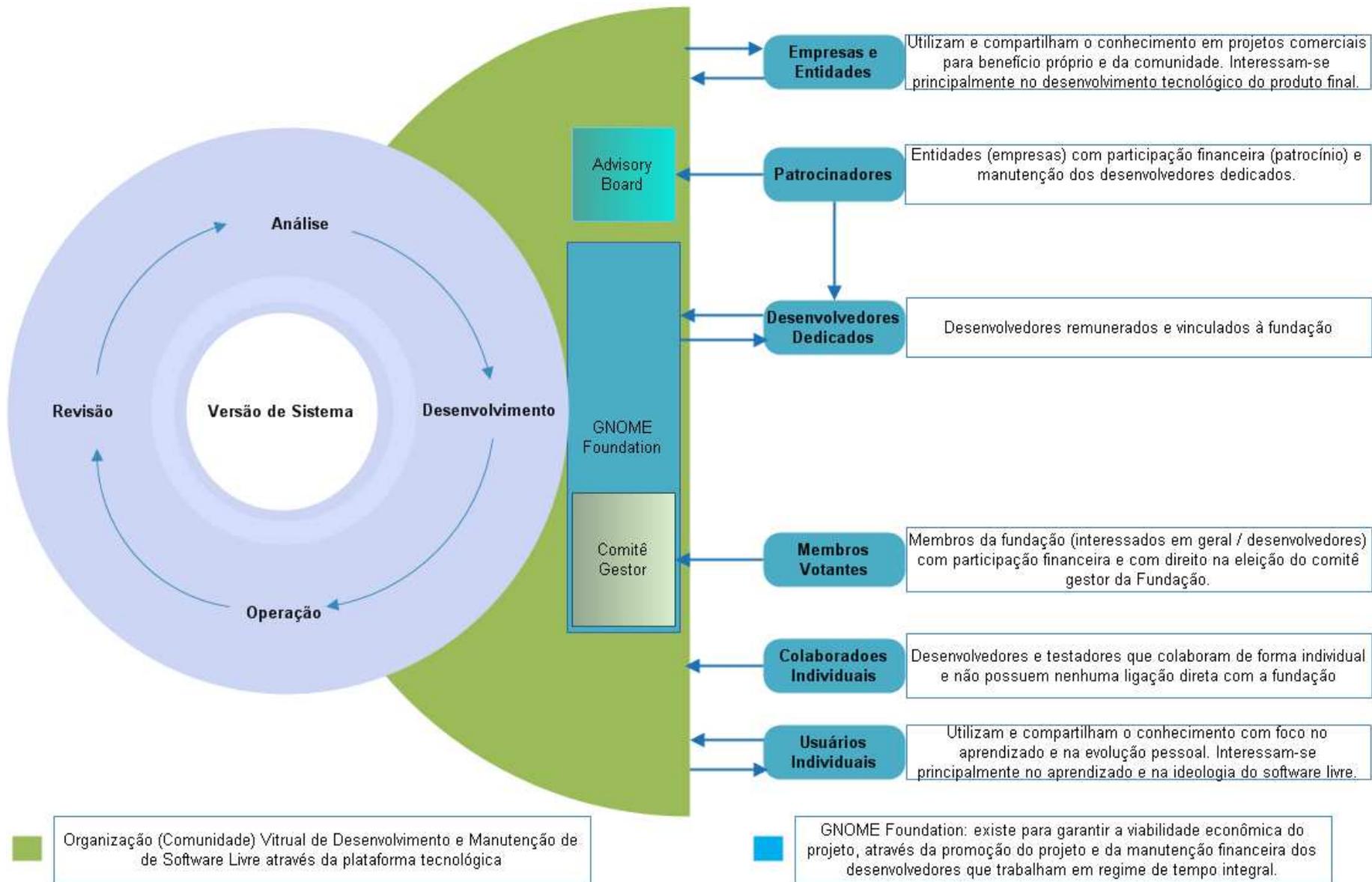


Figura 21: Topologia do Projeto GNOME

4.2.2 Arquitetura do Projeto

O Projeto GNOME divide-se em duas aplicações principais: O desktop e a plataforma de desenvolvimento. Estas aplicações são compostas por mais de 100 softwares (de código e licença livre), entre bibliotecas e aplicativos.

O desktop é a interface com o usuário final, um ambiente de interface gráfica (Graphical User Interface – GUI) e de aplicativos que cria uma camada de interação entre “homem x máquina”, facilitando assim a interação deste usuário.

Além desta interface, o projeto tem outra aplicação com o papel de fornecer uma base para o crescimento do desktop: A plataforma de desenvolvimento. Esta plataforma provê uma Interface de Programação de Aplicativos (API) poderosa e diversas ferramentas de programação, que são utilizadas por todos os envolvidos no projeto para fomentar mais ainda o crescimento do mesmo.

4.2.3 Caracterização da Comunidade

Partindo desta realidade, é importante frisar que o processo de desenvolvimento de softwares livres em comunidades dessa natureza é singular em relação a um aspecto pragmático do modo de vida dos participantes desta comunidade.

Enquanto “a última revolução industrial marcou a transição para uma sociedade que contava com os resultados científicos, na era da informação, mais importante que os resultados científicos é o modelo de produção entre pares, que permite obtenção desses resultados” (HIMANEM, 2001, p. 73).

Esta forma de trabalho denomina-se “bazar” e ocorre de maneira descentralizada, gerenciada pela estrutura de poder em rede horizontalizada, e com participação de voluntários (RAYMOND, 2001).

Qualquer pessoa pode voluntariar-se a começar um novo projetos ou mesmo participar com contribuições em projetos existentes ou até mesmo participar ativamente de decisões.

Entretanto, peso de uma contribuição para a tomada de decisão na comunidade depende de vários fatores como: histórico de contribuições do membro no projeto, habilidade para especificar suas contribuições, urgência da proposição em relação a objetivos atuais e atividades da comunidade e a existência de pessoal disponível para a execução da sugestão.

No caso do Projeto GNOME, observa-se uma imensa rede de colaboração que envolve mais de 400 pessoas de mais de 50 países de todos os continentes do planeta.

A distribuição destes contribuidores concentra-se na Europa e América do Norte, conforme pode ser observado no gráfico 22. O Brasil, e em especial, a América do Sul tem uma participação relativamente relevante, com uma participação de 5% do total dos contribuidores.

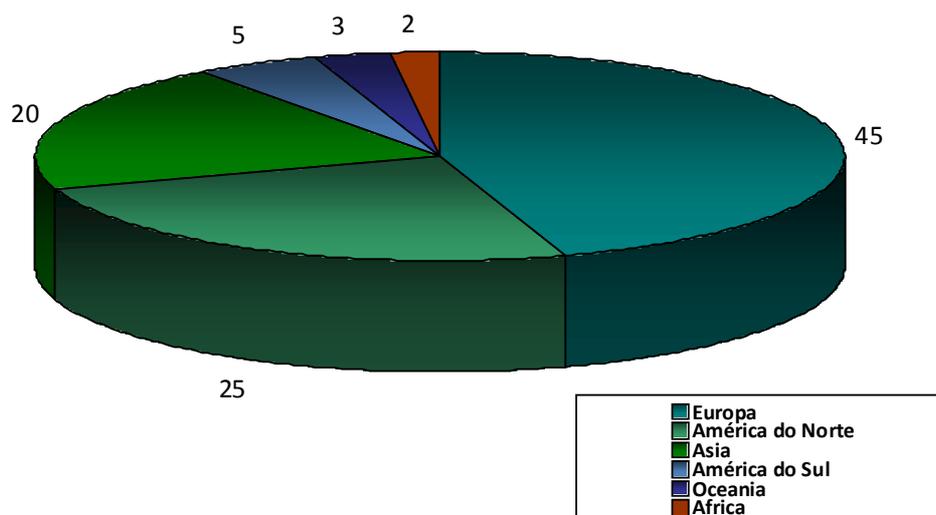
Participação Geográfica dos Membros do Projeto GNOME (%)

Gráfico 22: Participação Geográfica dos Membros do Projeto GNOME
Fonte: Elaborado pelo autor com base em GNOME, 2009.

No levantamento quantitativo de informações realizado na presente dissertação foi possível encontrar 610 contribuidores ativos, sendo que a grande maioria (mais de 90%) pertence ao sexo masculino. Estes contribuidores atuam em diversas funções do projeto como: desenvolvedores, testadores tradutores e colaboradores em geral.

A compreensão da visão destes membros perante a gestão do projeto e dinâmica de trabalho interno torna-se relevante para compreender os fatores de sucesso deste projeto e para o levantamento do modelo de gestão utilizado em redes de cooperação de projetos de grande porte.

O projeto GNOME torna-se especialmente interessante para esta dissertação devido ao reconhecimento de sua agilidade operacional no dito “mundo digital”. Hoje o projeto constitui-se numa estrutura de cooperação extremamente eficiente, conseguindo

realizar o lançamento periódico de atualizações e semestralmente a liberação de uma versão totalmente nova.

4.2.4 Fatores de Motivação

A motivação constitui um importante fator de compreensão da natureza humana e da explicação do comportamento humano, que impulsiona a pessoa a agir de determinada forma ou mesmo que dá origem a um comportamento específico (CHIAVENATO, 2005).

O projeto GNOME conforme Raymond (2001) pode ser caracterizado como um grande “bazar descentralizado” que envolve a cooperação de centenas de pessoas visando um objetivo comum. Partindo deste pressuposto, nota-se que a motivação de cada contribuidor é extremamente relevante, pois o produto final gerado é um resultado direto da dedicação e colaboração de cada um destes.

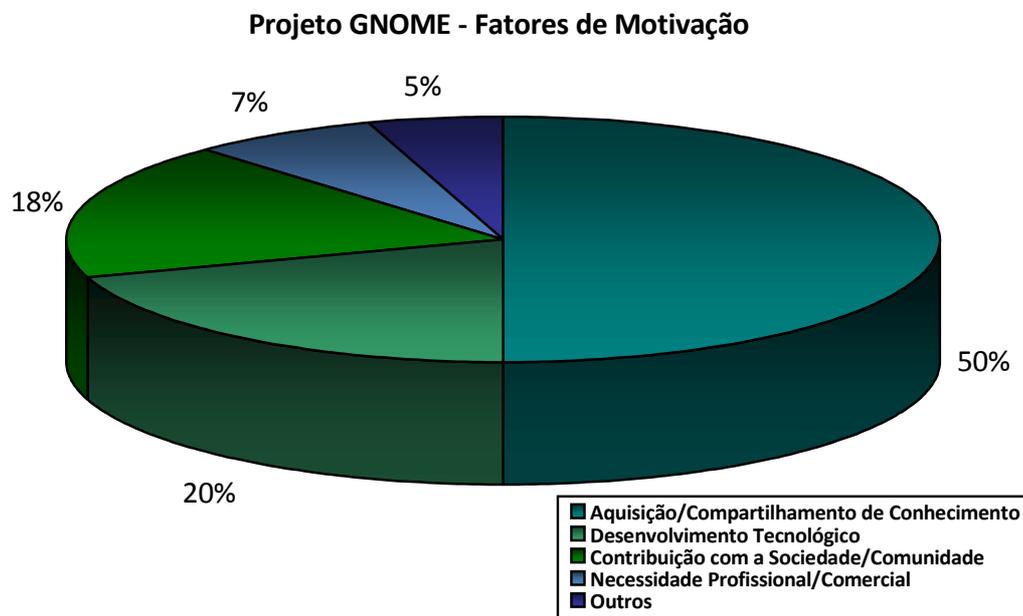


Gráfico 23: Fatores de Motivação dos Membros do Projeto GNOME

Os resultados da pesquisa demonstram 13 dos entrevistados (50%) colaboram com o projeto GNOME para que possam compartilhar e adquirir conhecimento. Este caráter educativo da rede, onde o contribuidor busca o aprimoramento pessoal ou mesmo profissional denota-se até mesmo nas notas textuais de liberação de versões do sistema (*release notes*):

Contribuidores habilidosos e motivados do GNOME podem encontrar novas portas e oportunidades abertas para eles na forma de reconhecimento, palestras e empregos pagos (GNOME, 2009).

Além disso, os entrevistados apontaram que a sua motivação está em auxiliar no desenvolvimento tecnológico da plataforma gerada (20%). Este fator está diretamente relacionado com o aspecto de colaboração com a comunidade (18%), pois a sociedade em geral é a maior beneficiada com o advento ou aprimoramento de uma nova tecnologia (Gráfico 23).

A necessidade profissional também foi apontada (7%), principalmente por membros da comunidade que participam de alguma implementação que utilizam em sua atividade profissional.

Essas necessidades geralmente são muito específicas e seguem a necessidade de alguma aplicação comercial da empresa onde o colaborador trabalha. Por exemplo é possível citar a empresa americana Red Hat, que comercializa e fornece suporte à sua distribuição do sistema operacional Linux. A empresa constantemente necessita de algumas funcionalidades do GNOME específicas adaptadas ao seu sistema operacional. Para tanto, a empresa incentiva e custeia alguns “contribuidores” no projeto para que desenvolvam estas funcionalidades.

Finalmente, outras motivações para a atuação como contribuidores do projeto foram apresentadas pelos entrevistados (5%) como: desafio pessoal, vocação, gosto pessoal, entre outros. É interessante destacar também, que alguns dos entrevistados afirmaram que todos os fatores citados anteriormente são relevantes e, portanto, estes não poderiam destacar uma motivação central.

4.2.5 Gestão das Modificações

Como visto anteriormente, o projeto GNOME é composto por centenas de pessoas, visando atender a necessidades de pessoas de diversas realidades do mundo todo. Portanto, as decisões centrais dos rumos do projeto devem seguir esta característica de um grande “bazar” da comunidade.

Os resultados da pesquisa demonstram esta característica principal da comunidade, onde a grande maioria dos entrevistados (70%) apontou que observaram que as decisões de gestão das modificações do projeto são realizadas de forma comunitária.

É interessante destacar que uma parcela considerável (20%) dos entrevistados apontou que o fator hierarquia (meritocracia) se faz presente. Numa análise mais subjetiva dos questionamentos, foi possível observar que normalmente esta noção de hierarquia está associada a figura do comitê gestor do projeto

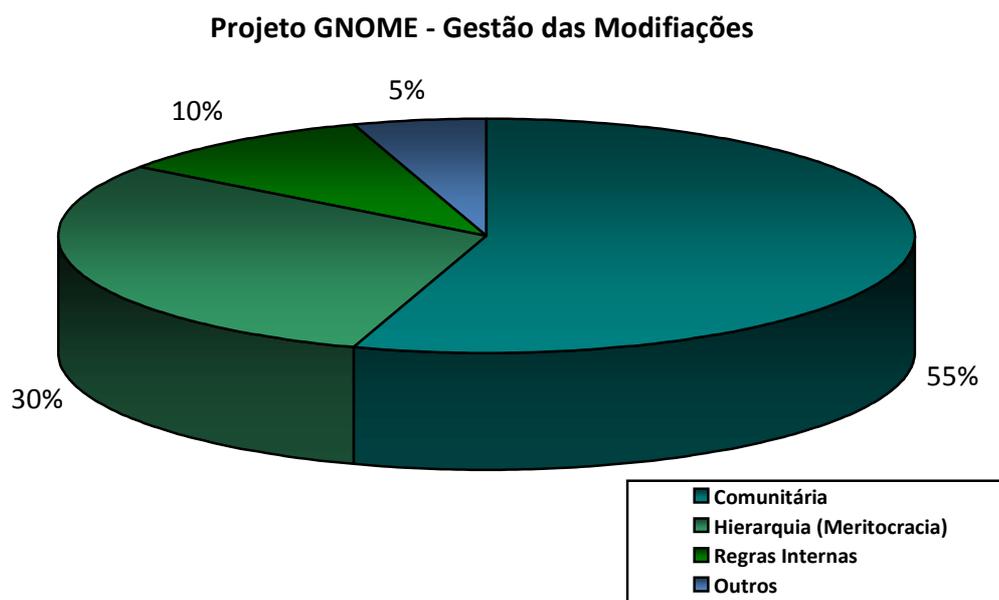


Gráfico 24: Gestão das Modificações do Projeto GNOME

No modelo de coordenação por comitê gestor existe um grupo de pessoas responsáveis por tomar as decisões do projeto. Este comitê é formado por desenvolvedores do projeto eleitos internamente na comunidade, ou pode ser formado pelos fundadores ou mantenedores do projeto.

O comitê gestor do projeto GNOME é composto membros do mundo inteiro, sendo alguns deles brasileiros. A participação de um membro no comitê está diretamente associada ao histórico do membro na comunidade e também sua participação.

Este estilo de gerência exige uma maior complexidade de interação se comparado ao tradicional modelo de centralização de liderança (REIS, 2003). Nesse modelo, é

necessário um meio democrático (como uma eleição, por exemplo) para definir quais os participantes da comunidade farão parte do comitê gestor.

Uma vez formado, o comitê realiza as mesmas funções de um líder, mas as atribuições são divididas o grupo formado, o que demanda uma política específica de tomada de decisões.

4.2.6 Estratégias de Comunicação

A estratégia de comunicação dentro da comunidade GNOME está totalmente baseada na utilização de ferramentas apoiadas pelas tecnologias de informação e comunicação – TICs (Gráfico 24).

Nesse contexto, para 40% dos entrevistados, o meio mais utilizado é a lista de discussão. Os membros da comunidade GNOME, assim como os membros da comunidade Firebird, destacaram as listas de discussão como um espaço democrático onde os membros podem além de discutir os rumos das modificações e inovações que serve como repositório histórico e de conhecimento de tudo que foi decidido.

Além da lista de discussão, para 25% dos entrevistados, os sites abertos à comunidade, (na forma de página oficial do projeto, blog e Wiki) são também fatores preponderantes de comunicação, pois se constituem num canal de comunicação importante e de caráter mais oficial.

Um aspecto singular identificado neste item da pesquisa foi a grande utilização das ferramentas de conversação (*chat on-line*) como meio de comunicação. Um quarto, ou seja, 25% dos entrevistados apontaram estas ferramentas como a principal ferramenta utilizada para comunicação.

Inclusive, é comum que ocorram reuniões periódicas da equipe de desenvolvimento do projeto utilizando a ferramenta IRC (Internet Relay Chat). O IRC é

utilizado basicamente como uma sala de “bate-papo” e troca de arquivos, permitindo a conversa em grupo ou privada.

Esta ferramenta é um protocolo de comunicação que foi bastante utilizado na Internet na década de 90, mas que aos poucos passou a ser menos utilizado pelo público em geral. Entretanto, os membros do projeto GNOME (principalmente os membros mais antigos) ainda a utilizam a ferramenta e arquivam no site do projeto o histórico das reuniões realizadas.

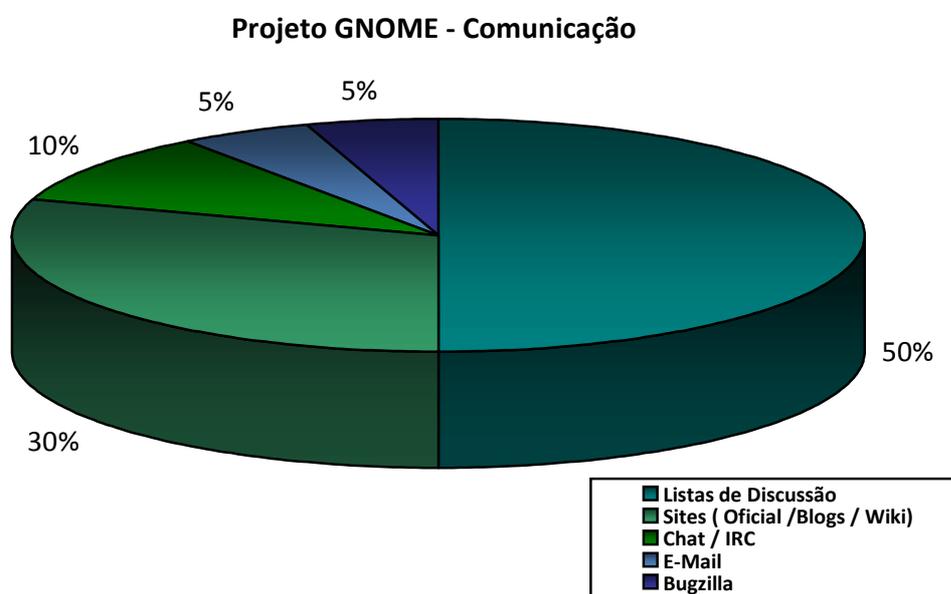


Gráfico 25: Estratégias de Comunicação no Projeto GNOME
Fonte: Elaborado pelo autor, 2009

Além das ferramentas vistas anteriormente, 5% membros entrevistados apontaram a utilização de mensagens eletrônicas (E-mail) como o meio de comunicação mais relevante. Como no caso do projeto Firebird, o E-mail também é utilizado para as listas de discussão do projeto, mas quando do questionamento sobre o recurso E-mail, utilizou-se o conceito da ferramenta como forma de comunicação direta e não pública.

O email é bastante utilizado por ser um bom denominador comum entre plataformas e línguas e geralmente é possível a comunicação com clareza através deste meio.

Finalmente, 5% dos entrevistados apontaram a utilização da plataforma Bugzilla como uma estratégia de comunicação utilizada. Essa plataforma também foi desenvolvida através de um projeto de software livre, assim incorpora também toda a metodologia de desenvolvimento deste tipo de produção.

O Bugzilla é uma ferramenta baseada na Internet que através de e-mail fornece suporte ao desenvolvimento de um projeto software. A ferramenta permite um registro histórico das alterações, e gerência das alterações em aberto e com isso permite unificar o processo de desenvolvimento do navegador de Internet Mozilla, suportando múltiplos aspectos do processo.

A maioria dos canais de comunicação do GNOME são públicos, isto ocorre por causa do grande número de pessoas que contribuem entre desenvolvedores e usuários-contribuidores. Não seria possível que todos esses envolvidos participassem ativamente na comunidade se os canais fossem restritos.

Além disso, não seria possível gerir o processo de desenvolvimento de um projeto tão complexo quanto o GNOME utilizando um processo de comunicação restrito, ainda que incluindo todas essas pessoas.

64 bugs found.

ID	Sev	Pri	Pit	Owner	State	Result	Summary
111909	wis	NOR	uns	kmail-devel kde org	NEW		printing multiple messages at once
111910	nor	NOR	Com	coolo kde org	NEW		Can't open links in whatsthis help
111911	wis	NOR	uns	kopete-devel kde org	NEW		support for drag and drop of attachment files in kopete
111912	nor	NOR	uns	linuxphreak gmx co uk	UNCO		wrong display of disk usage
111915	wis	NOR	SuS	konq-bugs kde org	UNCO		no crystalsvg key_enter.png available in 32x32
111916	nor	NOR	uns	andrew chant utoronto ca	UNCO		Number of enemy balls is always one no matter what it is ...
111917	nor	NOR	Gen	coolo kde org	UNCO		am_edit is tagging comment lines in Makefile.in
111918	nor	NOR	SuS	kdevelop-bugs kdevelop org	UNCO		setAttribute on form elements does not work
111919	nor	NOR	uns	kopete-devel kde org	UNCO		Contact protocol status icons shown offline with "plug"
111920	cra	NOR	SuS	digikam-devel kde org	UNCO		crash with signal 11 when sending album to trash
111922	wis	NOR	uns	konq-bugs kde org	NEW		file search dialog - default search period too long

Figura 26: Exemplo de Sistema Baseado no Bugzilla (KDE bug Tracking System)

Fonte: <http://bugs.kde.org>

A ferramenta possibilita que os usuários de um determinado sistema possam reportar deficiências, atuando no rastreamento dessas falhas e servindo também como plataforma para pedidos de recursos (Figura 26).

4.2.7 Dificuldades no Projeto

Conforme visto anteriormente, o GNOME é um projeto de grande porte, com isso a necessidade de recursos para manutenção do projeto também se tornam elevadas. Conforme visto anteriormente estes recursos são geridos pela Fundação GNOME, e são primordiais para manutenção de diversas atividades de apoio do projeto.

Neste contexto, a principal dificuldade apontada pelos entrevistados foi a falta de recursos disponíveis sendo apontada por 40% dos entrevistados como a principal dificuldade encontrada (Gráfico 27).

Esta constatação justifica-se pela atual conjuntura econômica mundial, onde diversas empresas enfrentam ou enfrentaram um período de severa crise. Nesse contexto, John Palmieri, principal tesoureiro do projeto fez a seguinte afirmação em carta pública à comunidade GNOME:

[...] Estamos ainda projetando que sem um fluxo de entrada considerável de contribuidores constantes nós não conseguiremos manter um Diretor Executivo ativo sem fazer cortes no orçamento [...] Enquanto a economia permanecer estas montanha russa de altos e baixos, a Fundação está se adaptando aos impactos, tentando servir da melhor maneira possível aos seus membros. (PALMIERI, 2009).

Além disso, segundo 30% dos entrevistados, os conflitos internos e de interesses dentro do projeto são fatores que dificultam a cooperação dentro do projeto. Estes conflitos que surgem naturalmente em todo contexto humano e toda comunidade de software livre, observam-se por inúmeras *flamewars* nas listas de discussão (Flamewar é uma situação de conflito onde um determinado assunto gera uma discussão não usual e possivelmente com ânimos alterados).

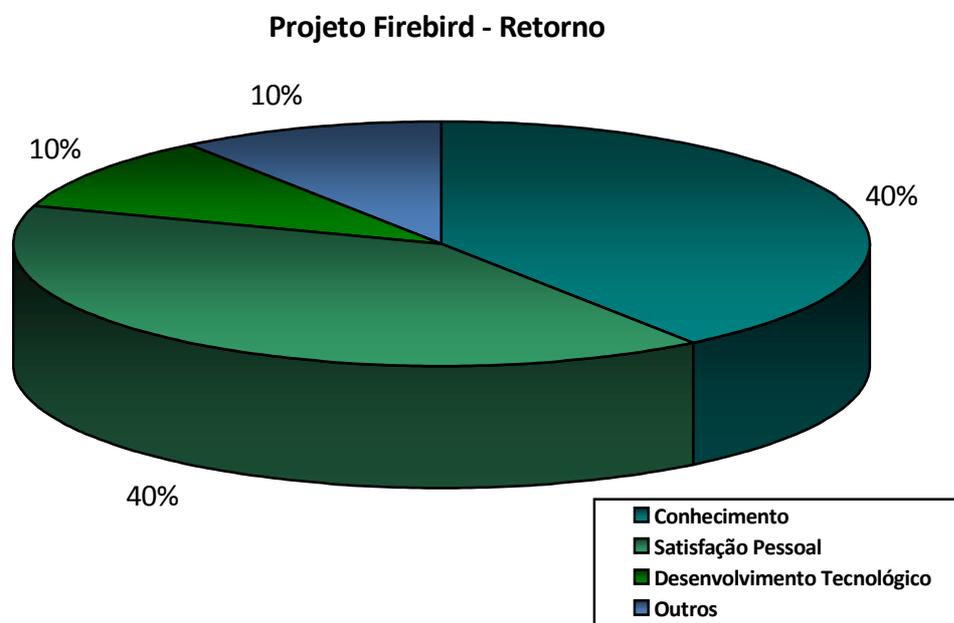


Gráfico 27: Dificuldades Encontradas pelos Membros da Comunidade

Esses conflitos devem-se principalmente pela heterogeneidade das pessoas e pela diferença da metodologia de trabalho que cada uma destas pessoas está acostumada a adotar no seu dia a dia.

A própria caracterização do processo de trabalho (em paralelo) dificulta a escolha conjunta de alternativas por parte dos usuários, o que pode terminar em conflitos de posicionamento no mercado, gerando discussões apaixonadas, quase religiosas, com potencial de provocar “rachas” e enfraquecer o movimento de disseminação do software livre (BOSTON, 2002).

Por outro lado, uma comunidade como a GNOME (conforme discutido anteriormente) atrai pessoas de todo o mundo e com diversas competências. Entretanto, este “bazar” muitas vezes sofre do problema do comprometimento dos colaboradores segundo 30% dos entrevistados

Corroborando com este resultado um estudo apresentado pelo Boston Consulting Group apontou que cerca de 70% dos participantes de projetos de software livre gastam 10 horas ou menos por semana em trabalhos relacionados ao projeto e essas horas são, em sua maioria, gastas durante à noite ou apenas nos finais de semana.

Assim, este baixo nível de participação pode introduzir problemas de comunicação, dado que muitos participantes têm dificuldade em acompanhar todos os desenvolvimentos realizados no projeto. Essas circunstâncias certamente dificultam o foco dos participantes no desenvolvimento do projeto (BOSTON, 2002).

Estes colaboradores normalmente necessitam de uma outra atividade para subsistência, pois seu trabalho é voluntário e não recompensado (direta ou indiretamente) financeiramente pelo trabalho no projeto.

Além disso, um outro fator é que os colaboradores são motivados a trabalhar no projeto pelo desafio intelectual que representa e muitas vezes após almejam esse objetivo acabam diminuindo o seu grau de participação nos projetos (BOSTON, 2002).

Finalmente, para 10% dos entrevistados, a pouca mão de obra disponível é um fator que dificulta o desenvolvimento do projeto, isso se deve principalmente ao nível de conhecimento necessário para se analisar e entender um código-fonte de um sistema grande que requer treinamento e experiência. O esforço em adquirir esse conhecimento não é algo efetuado pela maioria dos participantes.

4.2.8 Retorno do Projeto

O desenvolvimento de software livre é um processo de aprendizado contínuo, onde as partes envolvidas aprendem e contribuem para comunidade (BOSTON, 2002). Dentro desta comunidade alguns indivíduos participam ativamente buscando este

conhecimento e a satisfação pessoal através de um trabalho voluntário nos espaços de produção (LEHMAN, 2004).

Os resultados encontrados (Gráfico 28) confirmam essas afirmações, onde se observa que 40% dos entrevistados apontam que o retorno que o Projeto GNOME lhes fornece é o conhecimento adquirido através da atuação dentro do trabalho exercido na rede.

Os entrevistados afirmaram que este conhecimento é utilizado principalmente na atividade principal que exercem já que a atuação no projeto normalmente é uma atividade que exercem nas “horas de folga”.

Por outro lado, outros 40% dos entrevistados, apontaram que encontram a satisfação pessoal como principal retorno obtido. Esse fator relaciona-se diretamente aos princípios ideológicos do Software Livre, onde o contribuidor participa da comunidade pura e simplesmente buscando realização pessoal em colaborar com a comunidade ou com o surgimento de uma nova tecnologia.

Lakhani e Wolf (2005) afirmam que nos projetos de software livre, os programadores em especial, são motivados pelo fato daquele código gerado ser necessário para os utilizadores daquele software.

Diretamente relacionado com o a satisfação pessoal está o fator desenvolvimento tecnológico que foi apontado por 10% dos entrevistados como principal retorno obtido. O membro da comunidade sente-se motivado em poder participar no desenvolvimento de algo novo e que possa ser utilizado por todos.

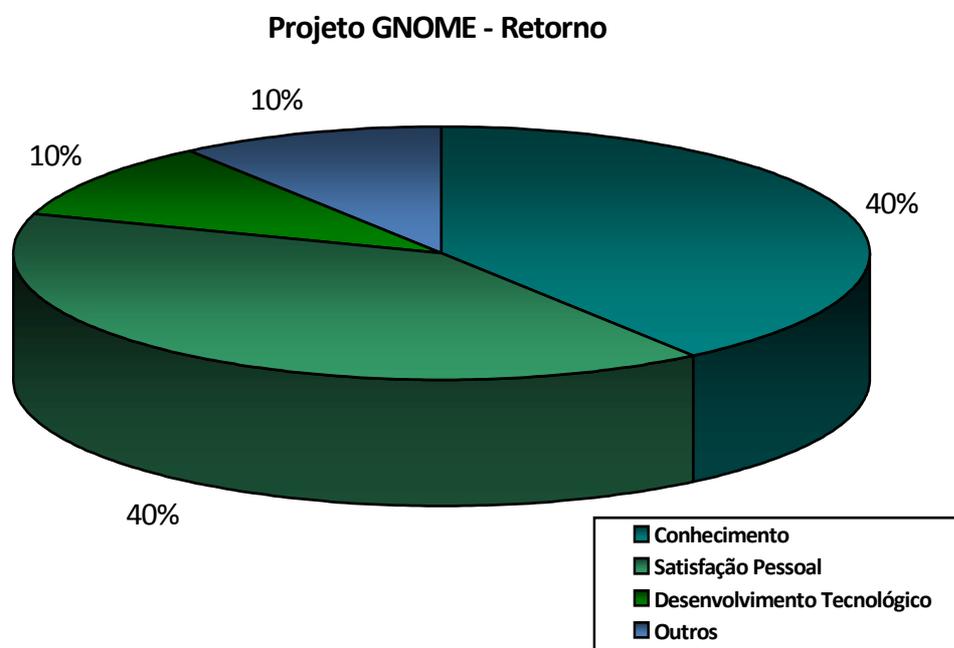


Gráfico 28: Retorno do Projeto GNOME

Concluindo esta análise foi possível observar que para 10% dos entrevistados existem outros fatores podem ser apontados como retorno do projeto GNOME. Dentre estes fatores foram apontados: realização de amizades, complementação profissional, curiosidade e outros.

Tabela 7: Compilação dos Resultados do Questionário de Pesquisa - GNOME

Motivação	<ul style="list-style-type: none"> - Disseminação do Espírito do Software Livre - Desafio - Desenvolvimento do conhecimento
Decisões	<p>As decisões de nível operacional são tomadas nas listas de discussões antes do trabalho ser realizado. Geralmente busca-se um consenso, abrindo mão de alguns pontos e adotando outros.</p> <p>Participantes das decisões: Os líderes de desenvolvimento, com o feedback dos desenvolvedores, a equipe de administração, a comunidade .</p>

Meio de Comunicação	<ul style="list-style-type: none"> - Listas de Discussão - E-Mail - Chat / IRC
Gestão dos Recursos	<p>A Gnome Foundation administra os fundos fazendo a ponte entre os patrocinadores (empresas e pessoas) e os desenvolvedores.</p> <p>A fundação tem como papel principal oferecer as garantias para que os desenvolvedores ajudem a evoluir e continuar o projeto.</p>
Hierarquia	<p>Dependendo da intensidade da participação do membro (meritocracia) , mais ou menos influência este terá no Software e na Comunidade.</p>
TICs Utilizadas	<ul style="list-style-type: none"> - Fórum de Discussão - Website - E-mail - Ferramentas de mensagens instantâneas
Desenvolvimento Do Software	<p>Todos os membros podem se envolver no desenvolvimento de soluções para qualquer parte do software – Sem entraves burocráticos para modificar qualquer parte do software</p>
Difusão do Conhecimento	<p>Como a maioria das discussões são públicas, todos podem aprender. Os desenvolvedores de hoje são pessoas que apenas observavam o que era discutido no passado.</p>
Ferramentas de Gestão	<p>Apenas ferramentas do SourceForge.net como o CVS (Controle de versões e modificações), Bugtracker (jira).</p>

4.3 Análise do Projeto TinyCobolBr

O projeto TinyCobolBr é constituído por uma comunidade com um número muito pequeno de contribuidores, e que está ativa a julgar pelos depoimentos colhidos pelo pesquisador junto à comunidade. O produto final da cooperação da comunidade é uma ferramenta de desenvolvimento de software orientada a negócios utilizada por diversas empresas no Brasil.

4.3.1 Histórico

O COBOL (abreviação de *Commom Busines Oriented Language*) é uma linguagem de computador orientada a programação estruturada que surgiu na década de 50 com o propósito de trazer a solução de problemas de programação do governo e das forças armadas americanas.

No fim dos anos 90 o Gartner Group, uma empresa de pesquisa na área de processamento de dados, estimou que dos 300 bilhões de linhas de código-fonte existentes no mundo, 80% - ou cerca de 240 bilhões de linhas - eram em COBOL. Eles também reportaram que mais de metade dos novos aplicativos de missão crítica ainda estavam sendo desenvolvidos usando o COBOL.

Os programas COBOL continuam em uso na maioria das empresas comerciais em todo o mundo, principalmente em bancos, uma vez que as regras que comandam o uso da linguagem a fazem aplicável a problemas comerciais.

Assim, o nome COBOL está diretamente associada a imagem de grandes bancos. Por este motivo, a MicroFocus (Inglaterra), fabricante do compilador COBOL mais utilizado mundialmente, pratica preços bastante elevados para suas ferramentas, demonstrando a completa falta de interesse no mercado de pequenas desenvolvedoras.

Devido a esses altos custos, muitas empresas buscavam um compilador COBOL viável para produção de seus programas.

Atendendo a essa demanda foi que surgiu em 2005 o projeto TinyCobol, visando oferecer uma alternativa de alto nível em software livre como ferramenta de desenvolvimento e compilação para a linguagem COBOL.

Em 2008 foi criado um projeto paralelo, ou *fork* do projeto, denominado TinyCOBOLBR, composto por toda equipe brasileira de desenvolvimento do TinyCOBOL que contribuía com o projeto visando atender às necessidades locais.

Hoje o projeto é composto por cinco membros que executam as funções de especificação, planejamento, desenvolvimento, suporte e manutenção do sistema. A ferramenta Cobol "nacionalizada" é utilizada por dezenas de programadores por todo o Brasil e mais de 50 sistemas são resultado da cooperação deste projeto.

O TinyCOBOLBR é utilizado para produzir aplicações para diversos segmentos, como Indústrias, Universidades, grandes Varejos, Prefeituras e Empresas de Serviços. Além disso, seguindo a vocação do COBOL em aplicações comerciais robustas estes sistemas são soluções que atendem missões críticas como: Controle de Contas a Pagar, Gestão de Contabilidade, Controle de Almoxarifado, Sistema de Compras, Emissão de Cupom Fiscal, entre outros.

A topologia da rede de cooperação estabelecida e as relações entre os atores envolvidos na comunidade TinyCobolBr podem ser observados em detalhes na Figura 29.

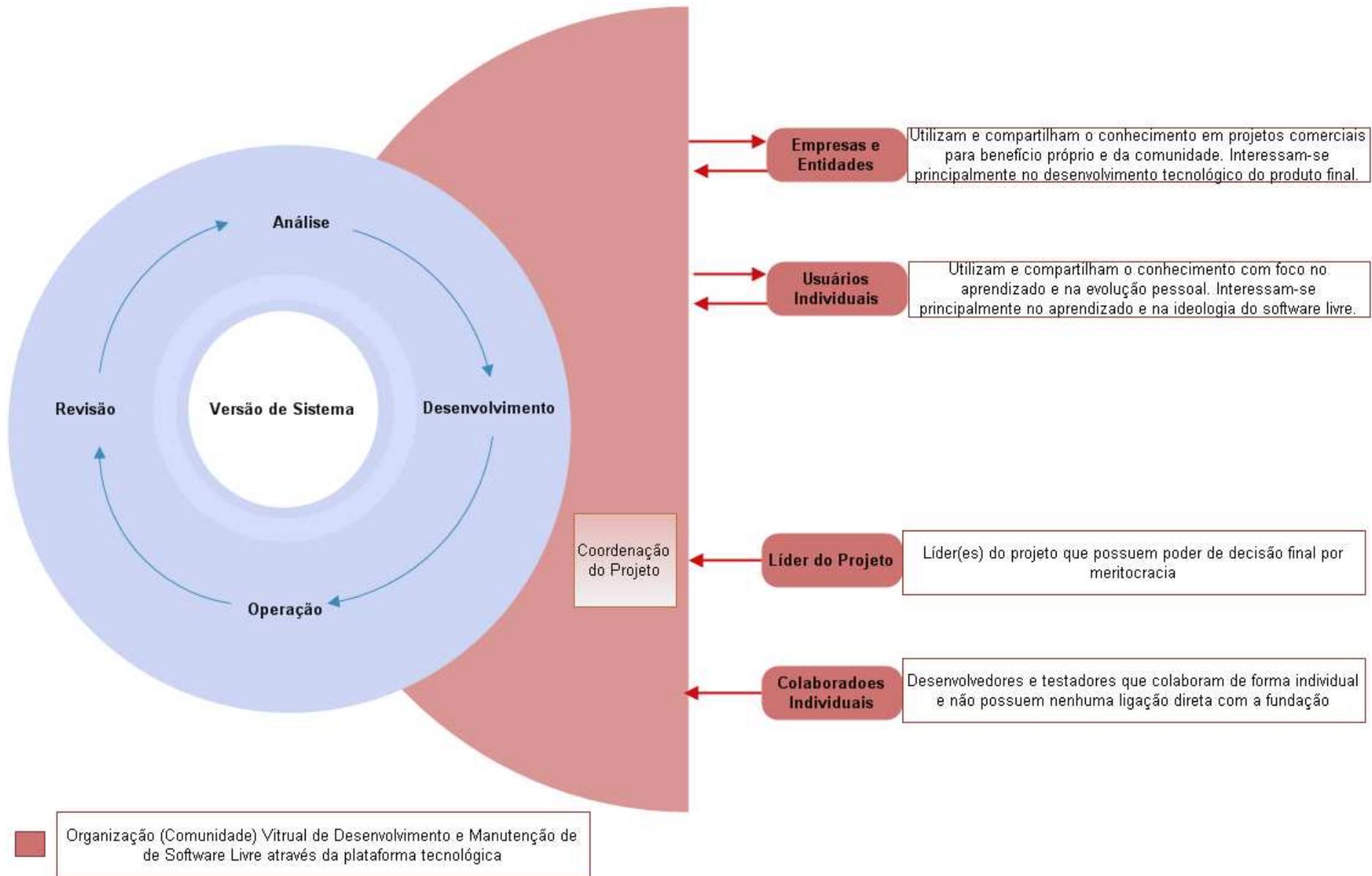


Figura 29: Topologia do Projeto TinyCobolBR

4.3.2 Fatores de Motivação

As pessoas são motivadas, acima de tudo, pela necessidade de realização (MINTZBERG; AHLSTRAND; LAMPEL, 2000). Como visto anteriormente, a comunidade do TinyCobolBR surgiu da missão de consolidar uma ferramenta de desenvolvimento comercial.

Portanto, ao identificar quais os fatores que motivam o membro da comunidade, foi possível constatar que neste projeto a grande “mola motora” são as variáveis profissionais e comerciais, que foram apontadas por 50% dos entrevistados como maior motivação da participação do projeto (Gráfico 30).

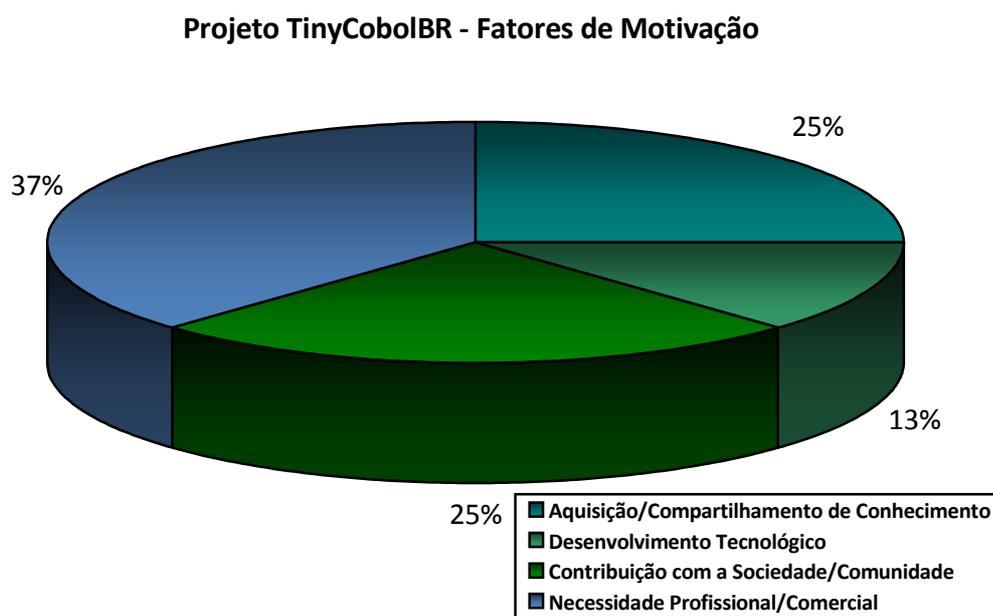


Gráfico 30: Fatores de Motivação do Projeto TinyCobolBR

Por outro lado, a questão da contribuição com a sociedade em geral, tão presente no espírito do software livre, é apontada por 30% dos entrevistados como um fator importante.

Isto demonstra que apesar do perfil substancialmente comercial do projeto, existe também a motivação social. Esta afirmação é confirmada pelos outros dois resultados encontrados como fatores de motivação: aquisição e compartilhamento de conhecimento e o puro desenvolvimento tecnológico/científico. Estes fatores foram igualmente apontados por 10% dos entrevistados como fatores relevantes.

4.3.3 Gestão das Modificações

Pelos resultados obtidos, nota-se que para a grande maioria dos membros do projeto (60%), a gestão das modificações ocorre de forma hierárquica (Gráfico 31). Observa-se então que a liderança do projeto é centralizada respeitando uma hierarquia verticalizada, onde o mérito histórico na comunidade de criador ou maior contribuidor do projeto é que define quem toma as decisões dentro da rede.

Porém, uma parte substancial dos entrevistados (30%) apontam que essas decisões são discutidas através dos fóruns de discussão com a participação aberta a todos os membros e por isso a gestão das modificações também ocorre de forma participativa.

Consultando as listas de discussão do grupo foi possível também encontrar a participação de regras internas como meio de controle e gestão, este fator também foi apontado por 10% dos entrevistados como relevante.

Os resultados encontrados justificam-se principalmente pelo tamanho e pela abrangência do projeto. Como visto anteriormente, o projeto TinyCobolBR é composto por poucas pessoas, visando atender às necessidades de uma comunidade mais restrita. Portanto, as decisões centrais dos rumos do projeto são tomadas de forma semelhante à

uma pequena organização tradicional, através da centralização do poder pelos dois autores e idealizadores originais.

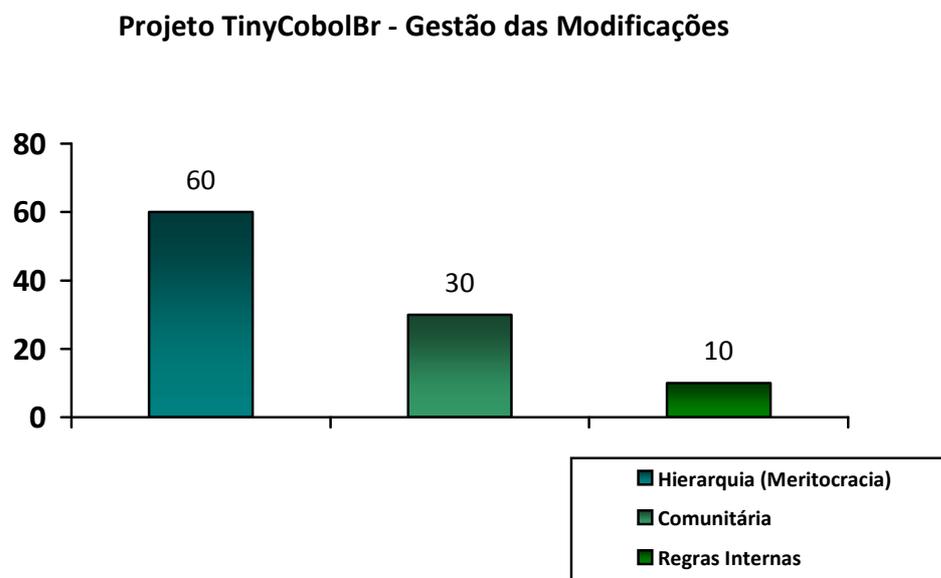


Gráfico 31: Gestão das Modificações do Projeto TinyCobolBR

Em uma organização neste estilo, onde existe uma única pessoa ou um pequeno grupo seletivo de pessoas que define os rumos do projeto, é possível observar a presença do chamado “ditador benevolente”. Este tipo de líder centraliza todas as decisões e comanda as atividades realizadas no sentido de evoluir o projeto.

Esse líder com poder de delegação de responsabilidades a desenvolvedores específicos da equipe possui acesso irrestrito aos recursos do projeto e é responsável pelo funcionamento da comunidade.

Entretanto, essa “ditadura” diz-se “benevolente” porque a liderança é compartilhada em parte através do compartilhamento de algumas atribuições para membros específicos do projeto (delegação de poderes) sobre subsistemas específicos do software.

É interessante destacar que a maior parte de projetos de software livre de pequeno e médio porte utiliza este tipo de liderança (REIS, 2003), pois não exige meios complexos de execução da atividade gerencial.

4.3.4 Estratégias de Comunicação

Os resultados da pesquisa de campo mostram que a estratégia de comunicação dentro do projeto TinyCobolBR é bastante simples. Isso se deve principalmente ao número de contribuidores ativos. A comunicação ocorre de maneira basicamente informal, utilizando as Tecnologias de Informação e Comunicação tradicionais da Internet: E-mail e chat conversa on-line).

Para 40% dos entrevistados, a comunicação ocorre de maneira direta, através da utilização de softwares de conversa *on-line* como o Microsoft Network Messenger (MSN). Segundo os entrevistados, este tipo de conversação ocorre normalmente para questões mais relativas ao desenvolvimento de sistemas e para o agendamento de eventuais encontros virtuais.

Ainda segundo outros 40% dos entrevistados citaram o e-mail como estratégia de comunicação utilizada. O email é utilizado principalmente para questões que devem ser repassadas para outros membros do projeto.

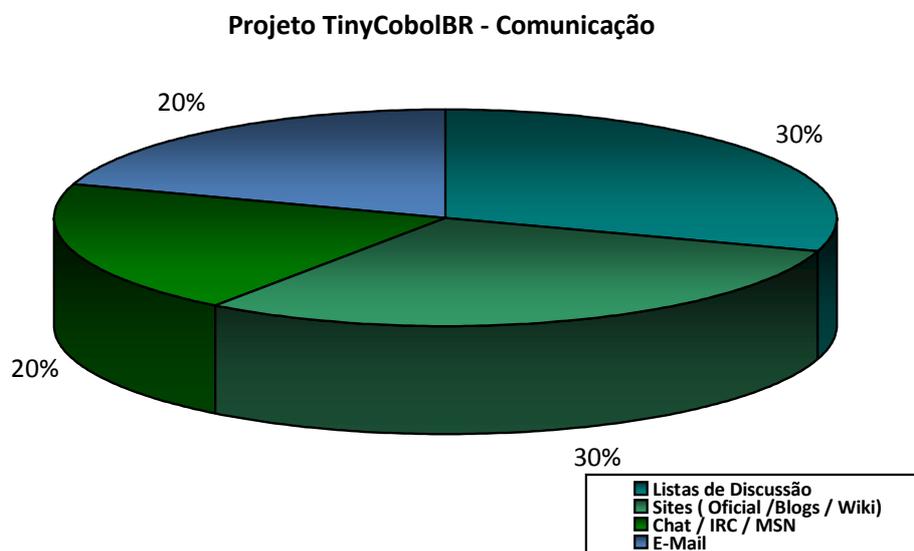


Gráfico 32: Estratégias de Comunicação do Projeto TinyCobolBR

Os entrevistados apontaram ainda a lista de discussão do grupo (10%) e o site oficial (10%) como estratégias de comunicação utilizadas. Entretanto, realizando uma análise qualitativa destes recursos foi possível observar que os mesmos estão sendo muito pouco utilizados para gestão e desenvolvimento do projeto. A utilização destes recursos está ocorrendo mais um meio de apoio à promoção e divulgação do projeto, buscando novos membros para o grupo.

Além disso, é interessante destacar que foi possível estabelecer uma relação direta entre os resultados obtidos (gráfico 32) e as conclusões do artigo "Collaboration with Lean Media: How Open Source Succeeds", de Yamauchi et al. (2000) que analisam os mecanismos de comunicação utilizados em projetos cooperados de software livre.

O artigo faz também uma análise das estratégias de comunicação em projetos de software livre, concentrando-se em uma análise das comunicações feitas entre os desenvolvedores dos projetos GCC e FreeBSD.

Na análise realizada pelo artigo, os autores confirmam os resultados obtidos nesta parte da pesquisa, concluindo que apesar do surgimento de novas ferramentas complexas

de trabalho colaborativo, os projetos de software livre (em especial os projetos de pequeno) porte sobrevivem do especialmente uso de ferramentas e meios de comunicação extremamente simples como: Correio eletrônico, E-mail.

4.3.5 Dificuldades no Projeto

Apesar do perfil substancialmente comercial do projeto, os resultados encontrados no gráfico 32 demonstram que o TinyCobolBr sofre da falta de contribuições financeiras (patrocinadores). Assim, os recursos limitados são apontados por metade (50%) dos membros entrevistados como a principal dificuldade encontrada.

Além disso, os resultado evidenciam que os desenvolvedores têm pouca disponível, uma vez que a baixa disponibilidade de tempo é apontada por 30% dos membros como um problema relevante.

Neste contexto, Fernando Wuthstrack⁴ (2009), administrador do projeto TinyCobolBR, resume assim esta dificuldade:

Se tivéssemos mais recursos, poderíamos contar com o desenvolvimento full-time ou part-time de um dos dois desenvolvedores, o que já iria resolver uma boa parte dos nossos problemas [...] Infelizmente não podemos cobrar mais comprometimento dos membros da comunidade. Como o desenvolvimento não é subsidiado, ninguém pode obrigar os desenvolvedores a utilizarem seu tempo em livre para as implementações necessárias.

⁴ Essa declaração foi concedida por Fernando Wuthstrack no dia 10/06/2009 em visita à Infocont Sistemas Integrados em Joinville/SC.

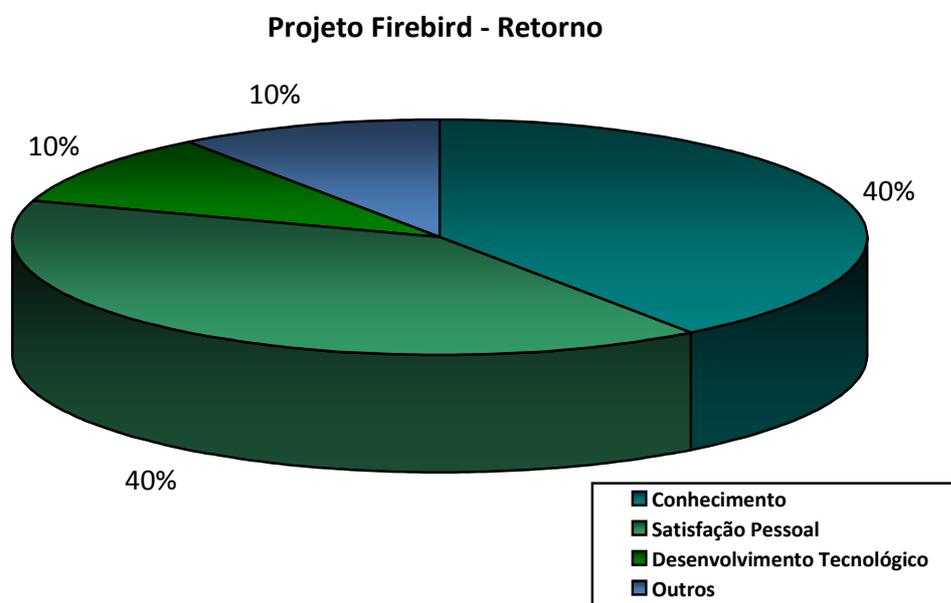


Gráfico 33: Dificuldades Encontradas pelos Membros da Comunidade

Além disso, os conflitos internos ocorrem como fator prejudicial ao funcionamento da rede de cooperação já que este é um fator apontado por 10% dos membros entrevistados. Estes conflitos ocorrem principalmente nas questões relacionadas com as diferentes metodologias de desenvolvimento de sistemas adotadas e normalmente são geridas pelo administrador do projeto.

Finalmente, 15% dos entrevistados apontam que existe escassez de mão de obra qualificada. Isso se evidencia no trabalho de Taurion (2004) que demonstra que os projetos de software livre concentram em poucas pessoas o conhecimento necessário para a execução do trabalho e nos projetos de software livre de pequeno porte este fator torna-se ainda mais preponderante.

4.3.6 Retorno do Projeto

Conforme visto nas motivações do projeto TinyCobolBR, observa-se que o desenvolvimento do projeto deve-se principalmente a uma necessidade profissional dos membros do projeto.

Esta afirmação pode ser confirmada pelos resultados obtidos visto que a maioria dos entrevistados (60%) apontou o desenvolvimento tecnológico na forma do software final gerado como principal retorno obtido (Gráfico 34).

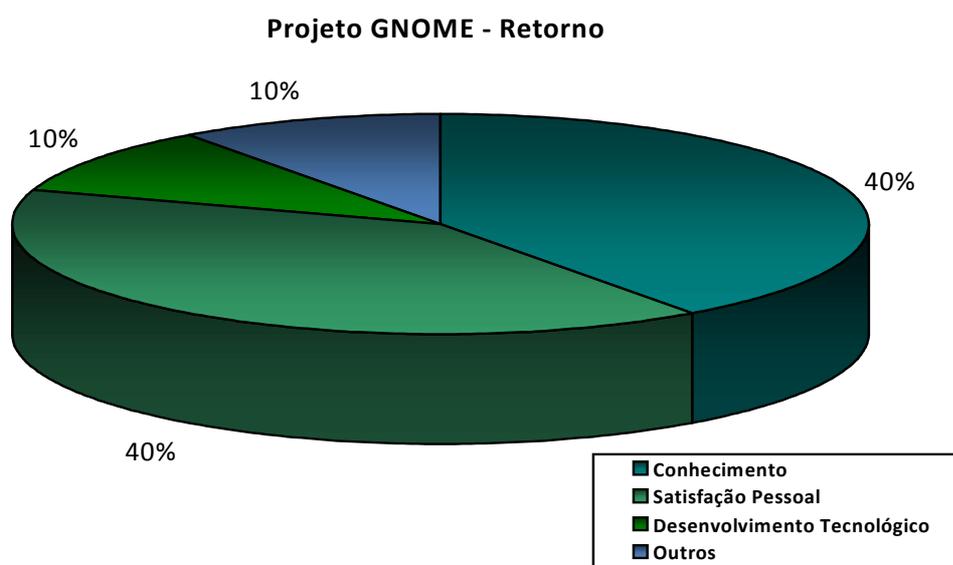


Gráfico 34: Retorno dos Membros da Comunidade

Além disso, foi possível também observar que a satisfação pessoal (10%) e a contribuição para aquisição de conhecimento (10%) também são relevantes num sentido complementar ao processo do desenvolvimento do TinyCobolBR.

Contudo, é interessante destacar também que os resultados obtidos demonstram uma relação muito próxima da pesquisa de Lakhani e Wolf (2005), onde os autores

concluíram que a mais forte motivação dos entrevistados era o grau de criatividade envolvido no projeto, ao lado do estímulo intelectual e do desafio de aperfeiçoar as funcionalidades dos softwares desenvolvidos.

Tabela 8: Compilação dos Resultados do Questionário de Pesquisa – TinyCobolBr

Motivação	- Necessidade de Desenvolvimento de Tecnologia (Profissional) - Aquisição e Compartilhamento de Conhecimento - Contribuição com a Sociedade
Decisões	As decisões de nível operacional são tomadas em conjunto pelos membros da comunidade, cabendo o poder de veto aos dois criadores da comunidade.
Meio de Comunicação	- Fórum de Discussão - Encontros de Pessoais dos Desenvolvedores
Gestão dos Recursos	Administra os fundos e oferece as garantias para que os desenvolvedores ajudem a evoluir e continuar o projeto. Definir as “roadmaps” do projeto.
Hierarquia	Dependendo da intensidade da participação do membro (meritocracia) e tempos de participação, mais ou menos influência este terá no Software e na Comunidade.
TICs Utilizadas	Utilização de ferramentas de comunicação mais diretas como :E-mail e ferramentas de mensagens instantâneas.
Desenvolvimento Do Software	Todos os membros podem se envolver no desenvolvimento de soluções desde que haja aprovação da versão gerada pelos membros – Sem entaves burocráticos para modificar qualquer parte do software.
Difusão do Conhecimento	Através de contato membro para membro (via ferramentas de TIC), da análise do código fonte e da documentação.
Ferramentas de Gestão	Nenhuma ferramenta formal de gestão, apenas uma lista de tarefas.

5 DISCUSSÃO DOS RESULTADOS E APRESENTAÇÃO DA PROPOSTA

A interpretação dos resultados obtidos na análise dos projetos cooperados abordados por este trabalho possibilita uma visão mais ampla do processo de gestão de projetos utilizada pelas comunidades de software livre. Assim, dando continuidade a pesquisa, o capítulo atual discute esses resultados e fornece um modelo inspirado nas características principais das redes de cooperação estabelecidas nesses projetos.

É importante ressaltar que a proposta aqui apresentada foi concebida para aplicação em qualquer tipo de projeto cooperado, apesar da análise do estudo ter utilizado como base em projetos de cooperação virtual.

5.1 Modelo como resposta para dificuldades identificadas

Um modelo, por tratar-se de uma representação simplificada, tem a função de servir como referência, a partir de determinados parâmetros obtidos no estudo, devidamente sistematizados.

A análise crítica dos modelos de processos deve resultar em uma listagem de pontos entendidos enquanto problemas e oportunidades de melhoria que devem ser tratados de forma adequada (SANTOS, 2007).

A análise das questões abertas da pesquisa (anexos A, B e C) permite apresentar os seguintes obstáculos mais comuns apontados pelos entrevistados:

- Falta de metodologias específicas e eficazes para a gestão de organizações virtuais;
- Questões de conflitos (sócio/culturais) que dificultam o trabalho cooperado;
- Falta de técnicas de técnicas de gestão na rede.

Uma constatação muito relevante que é que estas dificuldades são muito próximas àquelas identificadas nas organizações virtuais de Jagers, Jansen e Steenbakkers (1998). Dentre as dificuldades encontradas nas organizações virtuais citadas por esses autores estão presentes também algumas dificuldades inerentes a insuficiência de infra-estrutura básica e de tecnologia para integrar a estrutura virtual. Nesse sentido, o modelo poderá fornecer uma resposta positiva, por trabalhar sob uma arquitetura que utiliza a tecnologia de informação como base operacional.

5.2 Espaços de Cooperação

Na análise realizada nos projetos de software foi possível identificar três domínios de cooperação: (1) espaço comum de desenvolvimento do trabalho, (2) a área de funções de gestão do projeto e (3) a plataforma de tecnologia de informação. Nestes espaços é que ocorre o processo de interação e cooperação entre os membros do projeto (Figura 35). A relação entre os membros e cada um desses espaços é vista mais detalhadamente ao final deste capítulo.

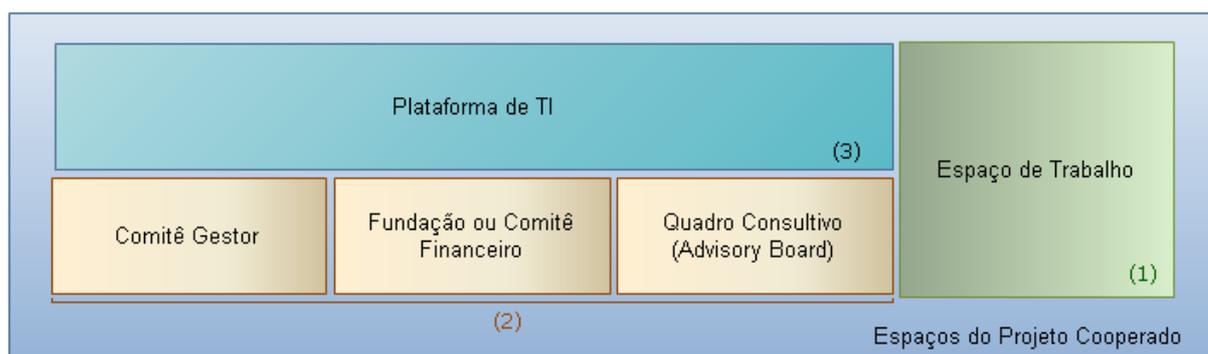


Figura 35: Espaços de Cooperação

5.2.1 O Espaço Comum de Desenvolvimento do Trabalho

O espaço comum de desenvolvimento do trabalho é o local real ou virtual onde a execução das atividades do projeto é realizada. Como exemplo de espaço comum de desenvolvimento de trabalho é possível destacar: laboratórios, unidades produtivas, obras de engenharia ou mesmo uma sala de aula.

No contexto mais específico do software livre esse espaço está distribuído virtualmente, sendo que cada computador onde atua um desenvolvedor torna uma fração do espaço de trabalho.

Porém, apesar do espaço parecer pulverizado, ele está interconectado utilizando a internet através da plataforma de tecnologia de informação (Figura 36). Sob a plataforma será administrado o projeto (circunferência central verde) e suas partes ou módulos no caso de software (na figura essas partes são circunferências verdes com as letras de A até H).

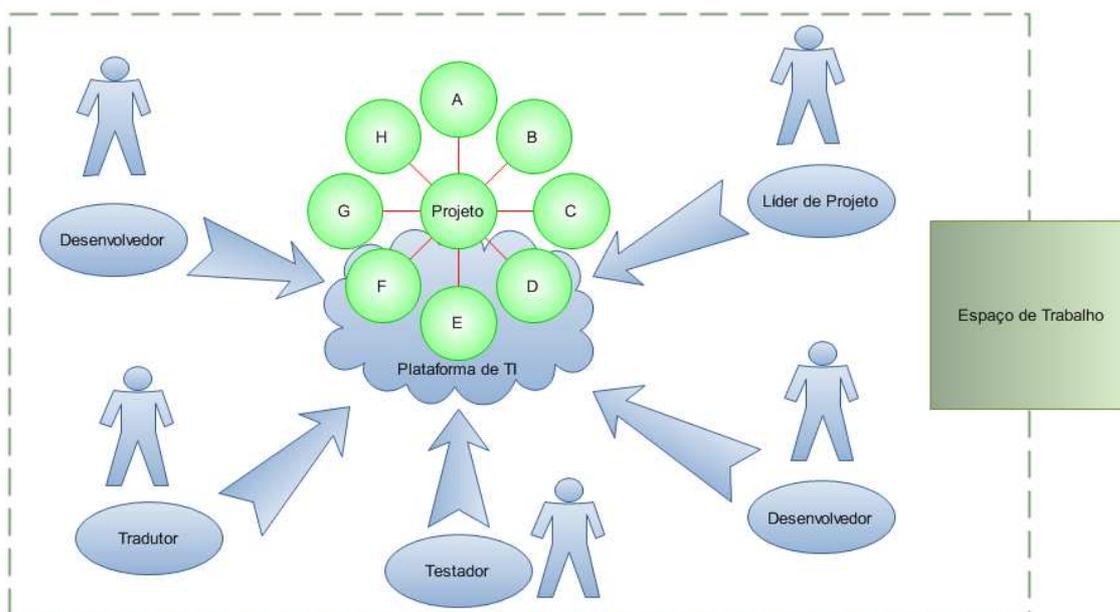


Figura 36: Espaço de Trabalho do Modelo Proposto

Os usuários têm acesso às partes e a documentação do projeto de acordo com a hierarquia pré-estabelecida pelos líderes da cooperação e o controle deste acesso é feito também através da plataforma de tecnologia de informação.

Em especial, é importante destacar a importância das ferramentas de controle de versões como CVS e SVN nesse contexto. É através destas ferramentas que cada parte do trabalho é unificada para formar o produto final da cooperação: a versão do sistema.

Nesse espaço ocorre a interação entre os membros do projeto e dessa interação surgem os produtos finais da cooperação como resultado dos “novos e poderosos modelos de produção baseados em comunidade, colaboração e auto-organização” (TAPSCOTT; WILLIAMS, 2007, p. 9).

5.2.2 A Área de Funções de Gestão do Projeto

A área de funções de gestão do projeto é responsável pelas atribuições de administração da colaboração através das funções de planejamento e controle. As funções deste domínio têm um papel de liderança no projeto com a “finalidade de influenciarem as pessoas de forma que os objetivos planejados sejam alcançados.” (ARAÚJO, 2004, p. 170).

As funções de planejamento envolvem a determinação no presente do que se espera para o futuro da cooperação, através das decisões estratégicas, para que os objetivos do projeto sejam alcançados.

Por sua vez, as funções de controle são similares àquelas executadas por um gestor executivo numa organização inerente ao acompanhamento das atividades, para garantir a execução do planejado e a correção de possíveis desvios (ARAÚJO, 2004).

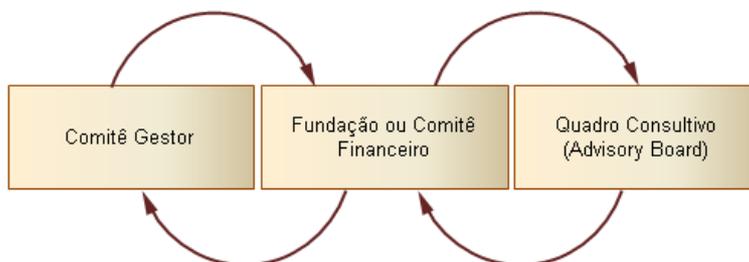


Figura 37: Áreas de Gestão do Modelo de Projetos Cooperados

Este domínio pode ser dividido em três áreas distintas (Figura 37) inspiradas no modelo do software livre e são apresentadas a seguir:

- **Comitê Gestor:** É composto por membros com direito a voto dentro da cooperação e conseqüentemente com poder de decisão. O próprio projeto definirá as regras da vigência deste comitê de acordo com as características do mesmo. Os membros do comitê são definidos por meritocracia, escolhidos dentre os demais, ou atuam na forma representativa de um determinado grupo. Para exemplificar, pode-se citar o caso de um membro que pode fazer parte do comitê gestor para representar uma organização ou até mesmo um outro projeto cooperado que tem participação no projeto em questão.
- **Fundação ou Comitê Financeiro:** É composto por membros escolhidos dentre os diversos membros da comunidade e dotados de capacitação para a função. O próprio projeto definirá as regras da vigência deste comitê de acordo com as características do mesmo. Os membros do comitê também são definidos por meritocracia ou escolhidos dentre os demais de forma representativa. O comitê financeiro deve ser constituído por tantos membros quantos forem indicados, sendo obrigatória a designação de no mínimo, um presidente e um tesoureiro. Este domínio é responsável gerir e recolher os fundos angariados e distribuir esses mesmos fundos dentro do projeto provendo assim a infra-estrutura necessária para que os objetivos da cooperação sejam alcançados. Além disso, é responsável pela

prestação de contas junto à comunidade e pela ligação (captação de recursos) com eventuais patrocinadores.

No caso de um projeto não comercial (como no caso do software livre) ou do terceiro setor é possível constituir este domínio como uma fundação autônoma (com personalidade jurídica e patrimônio próprios) administrada segundo as determinações de fundamentos determinados pelo comitê gestor do projeto.

- **Quadro Consultivo:** O quadro consultivo será formado por personalidades de destaque na cooperação para contribuir no desenvolvimento das atividades e na função de um conselho paralelo que poderá exercer também fiscalização sobre as outras áreas de gestão. A experiência do projeto GNOME sugere que organizações com possíveis interesses alheios ao foco principal do projeto participarão do quadro consultivo, mas não tem participação no comitê gestor do projeto.

Nos três projetos estudados, é possível também observar a mobilidade de membros dentro dentre destas funções. Além disso, é importante destacar que estas funções são exercidas por processos interdependentes, mas que podem e devem interagir entre si.

5.2.3 A Plataforma de Tecnologia de Informação

Na plataforma aqui abordada utiliza-se a abordagem mais ampla do conceito de Tecnologia da Informação incluindo desde as questões relativas ao fluxo de trabalho, pessoas e informações, incluindo os sistemas de informação, software e hardware, telecomunicações, automação, recursos multimídia, utilizados pelas organizações para fornecer dados, informações e conhecimento (LUFTMAN; LEWIS; OLDACH, 1993).

A função básica da plataforma é prover ferramentas que possibilitem a operacionalização do projeto cooperado. As necessidades atendidas pela plataforma e a ferramenta indicada para cada uma dessas necessidades, foram identificadas a partir da pesquisa dos projetos cooperados do capítulo anterior. Um quadro que resume estas necessidades versus a ferramenta de TI indicada pode ser observado na tabela 9:

Tabela 9: Necessidade Identificada X Ferramenta de TI Indicada

Necessidade Identificada	Ferramenta de TI Indicada
Comunicação Interna	Chat / E-mail / Fórum de Discussão
Planejamento / Acompanhamento	Ferramenta PERT/CPM e Controle de Recursos Lista de Tarefas (ToDoList / Tracker)
Base de Conhecimento	Wiki / Fórum de Discussão
Administração de Recursos Humanos	Ferramenta PERT/CPM e Recursos

5.2.3.1 Comunicação Interna

Primeiramente, o requisito básico a ser atendido pela plataforma é a de comunicação interna. Os espaços de cooperação necessitam de um ambiente que permita a comunicação síncrona ou assíncrona entre os membros do projeto.

Para a comunicação assíncrona, o e-mail é a opção mais trivial e usual, entretanto, assim como nos projetos de software livre, surge a necessidade de uma forma de comunicação com os membros que possibilite um registro histórico e permita um nível maior de colaboração. É neste contexto que a lista de discussão é a aplicação mais indicada.

A grande vantagem da utilização de uma lista de discussão é que este tipo de ferramenta permite a formação de comunidades virtuais de aprendizagem, onde os membros do projeto são estimuladas à busca do conhecimento e ao compartilhamento de

idéias. As listas são talvez o recurso mais importante de uma comunidade de aprendizagem colaborativa à distância (SOARES; AMARAL, 2004).

Para a comunicação síncrona entre os membros sugere-se a utilização de ferramentas de conversação *on-line* (mensagens instantâneas) de acordo com a disponibilidade e necessidade da comunidade. Além disso, a plataforma pode disponibilizar um ambiente para reuniões via *chat* que possam ficar registradas e acessíveis à comunidade.

5.2.3.2 Planejamento, Acompanhamento e Administração de Recursos Humanos

O planejamento e acompanhamento do projeto são funções primordiais sugeridas na plataforma e no modelo de gestão de projetos cooperados apresentado por este trabalho de pesquisa.

A plataforma deve disponibilizar então uma ferramenta integrada que disponibilize ferramentas para geração do cronograma, utilização de redes PERT / CPM (caminho crítico), gráfico de Gantt (Figura 5) e o controle dos recursos alocados de cada projeto. Com isso será possível controlar custos, prazos e alocação de recursos para cada etapa do projeto.

Um outro tipo de ferramenta dentro do contexto de planejamento e controle indicada é um controle (lista) de tarefas urgentes. Esta lista armazena informações sobre tarefas em aberto, possíveis falhas encontradas no decorrer do processo e dados sobre a execução das mesmas.

Dentre os projetos de software livre, apenas o GNOME possui uma estrutura de projeto mais definida utilizando parcialmente este tipo de soluções. É interessante constatar que a gestão dos recursos humanos é uma dificuldade comum dentre os projetos pesquisados.

Hoje, existem inúmeras opções de soluções que contemplam estas necessidades, e, portanto, esta pesquisa não irá apontar nenhuma em especial. Dentre estas inúmeras soluções, o maior requisito é que a ferramenta seja totalmente acessível pela internet. Permitindo a colaboração do membro independentemente de onde ele esteja. Sugere-se então a utilização de uma aplicação para internet (via navegador, por exemplo).

5.2.3.3 Base de Conhecimento

Uma das necessidades mais evidentes observadas nos projetos estudados foi a de manter uma base de conhecimento. Nonaka e Takeuchi (1997, p. 10), corroboram com essa constatação afirmando que “as atividades criadoras do conhecimento realizadas dentro do sistema de negócios e os níveis de equipe de projeto são captados e recontextualizados na base de conhecimento da organização como um todo”.

Nesse contexto, as ferramentas de gestão do conhecimento têm como objetivos desenvolver, disseminar e compartilhar conhecimentos. Assim, a ferramenta adotada pelo modelo é um tipo especial de site colaborativo que possibilita a contribuição e autoria de diversas pessoas voltada para a construção coletiva de conteúdos: o wiki.

O wiki torna possível a geração de documentação do projeto de forma colaborativa e que pode ser acessada e atualizada constantemente. Além disso, a criação do conhecimento, conforme entendida por Nonaka e Takeuchi (1997), supõe um processo de aprendizagem individual em que o aprendizado seja resultado de reflexão, de criatividade, de questionamento, logo de natureza construtivista.

É neste sentido que outra ferramenta já citada anteriormente, a lista de discussão, possibilita o estabelecimento de uma base de conhecimento. Além de uma excelente ferramenta de comunicação contribui substancialmente para construção de uma base de

conhecimento. Trata-se de um registro histórico completo da interação e reflexão dos indivíduos envolvidos sobre diversos temas relacionados ao projeto.

5.3 Os Atores Envolvidos no Processo

Para construção do modelo proposto por este trabalho, além dos três domínios de cooperação apresentados nos tópicos anteriores, será necessário compreender o papel de cada um dos atores envolvidos no processo.

É importante destacar que uma mesma pessoa pode exercer mais de um papel dentro da comunidade, de forma semelhante ao “bazar” do software livre (RAYMOND, 2001), caracterizado pela estrutura de poder horizontalizada, colaboração e auto-iniciativa.

Portanto, o papel de cada um destes atores envolvidos no modelo proposto são apresentadas a seguir:

- **Membros Votantes:** São responsáveis pelas decisões de nível gerencial para questões macro do projeto. A votação dos membros tem como objetivo possibilitar que os votantes manifestem sua vontade ou opinião sobre essas questões para que as decisões sejam tomadas. Segundo Rezende (2003) um processo de votação pode ser realizado de várias formas, como por exemplo, concordando ou discordando sobre alguma questão, ou escolhendo uma dentre várias opções. O direito de abstenção, normalmente, é dado ao votante.
- **Empresas e Entidades:** Participam no desenvolvimento do projeto através de patrocínio, manutenção de membros com dedicação total ao projeto e podem atuar junto ao comitê financeiro para fiscalização da utilização dos recursos. Podem designar representantes para que atuem como membros votantes, mas não têm participação direta no comitê gestor. Isso ocorre para que estas organizações não

utilizem o seu poder de controle econômico (no sentido de Foucault, 2004) para obterem o comando ou vantagens exclusivas dentro do projeto cooperado. Além disso, as empresas e entidades têm um papel importante no quadro consultivo, utilizando a experiência organizacional em favor do espaço cooperado.

- **Membros Dedicados:** São membros da comunidade do projeto que possuem dedicação profissional total e remunerada. A presença desses membros assegura uma evolução contínua das atividades propostas. A sua remuneração é responsabilidade do comitê financeiro.
- **Membros Esporádicos:** São responsáveis por contribuições espontâneas ou contratadas junto às atividades do projeto. A participação destes membros justifica-se também em questões relativas a pesquisas, consultorias e treinamentos oferecidas aos demais membros do projeto cooperado.

5.4 Modelo de Gestão de Projetos Cooperados

Os resultados obtidos com o alcance dos objetivos específicos da pesquisa fundamentaram, naturalmente, a construção do modelo de gestão de projetos cooperados apresentado pela Figura 38.

O modelo proposto determina o processo de gestão dos projetos cooperados, definindo a forma como este deve ser operacionalizado nas suas fases do ciclo de vida. É através do modelo de gestão que se definem: a hierarquia e as responsabilidades, papéis e posturas, critérios de avaliação e a forma como ocorre a interação do cliente do projeto com a rede formada.

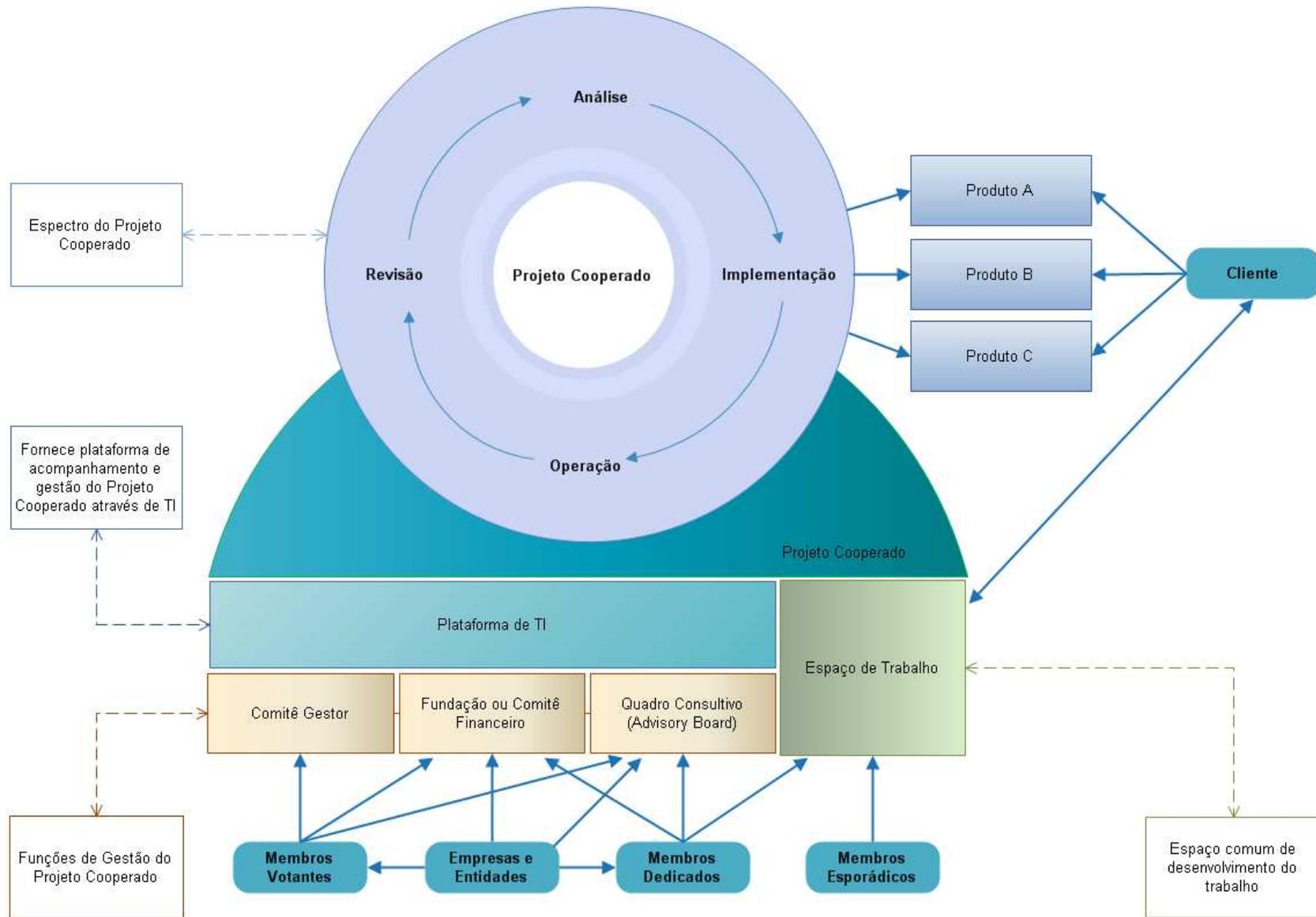


Figura 38: Modelo de Gestão de Projetos Cooperados

O modelo é constituído pela junção dos espaços de cooperação, com as pessoas envolvidas (atores) e pelo espectro do projeto cooperado. Num nível superior é possível observar a execução do projeto através das etapas de análise, implementação, operação e revisão. O resultado deste processo são os produtos finais (por exemplo: softwares, serviços, projetos, equipamentos, entre outros) gerados pela cooperação em rede estabelecida.

A plataforma de tecnologia está propositalmente situada ao centro do modelo (Figura 38). É a plataforma que proporciona a interconexão dos parceiros (assim como nas organizações virtuais) para que possam combinar seus esforços e recursos num processo conjunto de criação de valor. Outro aspecto importante e sugerido por este modelo, é que a plataforma seja reutilizável, já que a cada projeto realizado uma base de conhecimento é gerada.

Com isso, as atividades criadoras do conhecimento realizadas nos projetos podem ser captadas e recontextualizadas gerando valor para a organização (NONAKA; TAKEUCHI, 1997). Assim, as informações coletadas externamente são adaptadas de forma que orientem os futuros projetos estrategicamente.

No modelo, existe também a figura do cliente que é a representação das pessoas que serão beneficiadas pelo projeto. O cliente interage com os outros atores do projeto através do espaço de trabalho que, por sua vez, está baseada na plataforma de tecnologia de informação. É importante ressaltar que assim como no software livre um membro do projeto pode também ser um cliente do mesmo.

Contudo, este modelo é mutável de acordo com novas necessidades, estando apto a transformações e melhorias, assim como a tecnologia de informação alterou a forma das redes de cooperação do desenvolvimento de software livre (BENKLER 2007). Estas questões serão abordadas mais detalhadamente no capítulo subsequente.

6 CONCLUSÕES E RECOMENDAÇÕES

Neste capítulo são apresentadas as conclusões, limitações e recomendações para futuros trabalhos. A formulação da conclusão foi extraída do presente trabalho, tendo como base os resultados encontrados e objetivos estabelecidos. As recomendações presentes foram elaboradas no intuito de contribuir para a evolução do modelo proposto nesta dissertação.

6.1 Conclusões

Costa, de Rolt e Dias (2008) afirmam que a formação de novos modelos de gestão faz-se necessária para que as organizações sejam competitivas através da utilização do desenvolvimento tecnológico e científico, atendendo assim as demandas atuais e preparando as bases para novas demandas.

Neste contexto, o ponto de partida para o estabelecimento das conclusões deste trabalho foram os objetivos que nortearam todo o desenvolvimento da pesquisa. Assim, a proposição do modelo conceitual utilizado com base na pesquisa de campo e na literatura pertinente foi a principal premissa adotada, servindo como suporte e referencial para a conclusão desta dissertação.

Portanto, é possível concluir, que o modelo proposto (apesar da origem específica) traz uma solução aberta e adaptável a qualquer tipo de projeto de cooperação, podendo ainda ser considerado uma metodologia abrangente, na medida em que mantém uma visão dos principais atores e processos envolvidos no desenvolvimento de um projeto cooperado.

Mesmo com as limitações inerentes a um trabalho de pesquisa desta natureza, como o tamanho da amostra, a abrangência do tema de pesquisa, caráter abstrato dos projetos estudados e principalmente a originalidade do modelo, é possível afirmar os objetivos propostos foram plenamente atingidos.

Neste contexto, a aplicabilidade do modelo se justifica por estar baseada em soluções de cooperação de software livre já utilizadas (mesmo que informalmente) em diversos projetos por todo mundo.

É interessante destacar que toda a plataforma de tecnologia de informação adotada por estes projetos já existe e pode ser totalmente utilizada pelo modelo proposto. Além disso, esta tecnologia segue a linha ideológica principal dos projetos estudados, ou seja, são soluções em plena evolução e que não exigem licenciamento comercial (software livre).

O modelo proposto não tem a ambição de estar totalmente completo, ou mesmo de ser uma panacéia para todos os projetos cooperados, estando sujeito a transformações e melhorias. Neste sentido, o avanço da tecnologia da informação, o surgimento de novas pesquisas, conceitos e os resultados de sua aplicação prática, são exemplos de situações que podem vir gerar a necessidade de mudança no modelo proposto. Assim, como o próprio processo de desenvolvimento científico, o modelo também deve seguir o caminho da melhoria contínua, alternando a sua evolução entre teoria e prática.

Outra constatação interessante obtida é que apesar da literatura sobre projetos cooperados e organizações virtuais ser relativamente recente (TRÖGER, 1997) é possível observar-se que os conceitos destas teorias (mesmo que empiricamente) já faziam parte da realidade do software livre desde sua gênese há mais de 30 anos.

Vale ressaltar que alguns aspectos extraordinários foram encontrados e merecem maior atenção em estudos futuros. Dentre estes é possível destacar: as questões jurídicas para a gestão das cooperações, as questões de direitos autorais dos produtos provenientes da cooperação e a administração dos conflitos inerentes a qualquer processo de cooperação.

6.2 Recomendações e Trabalhos Futuros

Para futuros trabalhos relacionadas à gestão de projetos, esta dissertação propõe validar o modelo de gestão de projetos cooperados através da aplicação do modelo em projetos de cooperação virtuais ou não, criando assim a possibilidade de evolução do modelo proposto.

Além disso, recomenda-se que sejam realizados trabalhos de pesquisa analisando as ferramentas de apoio à gestão do projeto, através do método *benchmarking*, para aprofundar a pesquisa realizada, verificando possíveis aplicações que possam ser utilizadas na plataforma de tecnologia de informação proposta.

Com relação a outros trabalhos futuros que estejam focados no software livre, recomenda-se o estudo da melhoria dos processos deste tipo de produção. Apesar dos excelentes produtos resultantes destes tipos de cooperação, ainda existem limitações e problemas a serem solucionados com relação à conflitos, ao processo e à comunidade. Apesar de não ser o foco principal desta pesquisa, foi possível identificar algumas dessas deficiências, que poderão ser utilizadas como ponto de partida.

Por fim, espera-se que os novos estudos e iniciativas provenientes desta pesquisa possam contribuir para o desenvolvimento de uma base de conhecimento científica sobre o tema abordado. Assim, poderão surgir novos modelos (evoluções do modelo atual ou complementares) e experimentos que possibilitarão assim, o aperfeiçoamento da gestão de projetos cooperados.

REFERÊNCIAS

AMATO NETO, João. Redes de organizações/empresas virtuais na economia global. In: ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO, 17., Gramado - RS, 1997. **Anais...** Porto Alegre: UFRGS.PPGEP, 1997.

ARAKAKI, A. A.; TURINE, M. A. S. Gerência de Projetos de Software Livre no Framework SAFE: Conceitos e Ferramentas. In: WTD - Workshop de Teses e Dissertações do Webmedia 2005 - Simpósio Brasileiro de Sistemas Multimídia e Web, 2005, Poços de Caldas - MG. Anais do WTD - Workshop de Teses e Dissertações do Webmedia 2005 - Simpósio Brasileiro de Sistemas Multimídia e Web, 2005.

ARAÚJO, Luis César G. **Teoria Geral da Administração:** aplicação e resultados nas empresas brasileiras. São Paulo: Ed Atlas, 2004.

BARDIN, Laurence. **Análise de conteúdo.** Lisboa: Edições 70, 1977.

BARRETO, A. A. A condição da informação. **São Paulo em perspectiva**, São Paulo, v. 16, n. 3, p. 67-74, 2002.

BARROS, Aidil Jesus da Silveira; LEHFELD, Neide Aparecida de Souza. **Fundamentos de metodologia científica:** um guia para a iniciação científica. 2. ed. ampl. São Paulo: Makron Books, 2000.

BENKLER, Yochai. **Coases Penguin, or Linux and the Nature of the Firm, 112 Yale Law Journal 369.** Disponível em: <<http://www.benkler.org/CoasesPenguin.htm>>. Acesso em: 05 maio 2007.

BERTO, Rosa Maria Villares de Souza. Organizações Virtuais: Revisão Bibliográfica e Comentários. In: Encontro Nacional de Engenharia de Produção. 1997, Gramado, RS. **Anais...** Porto Alegre: UFRGS. PPGEP, 1997.

BEZERRA, C. E. L. M. Interação, cooperação e comunicação no desenvolvimento de programas computacionais: Determinando o campo de ação e visão. In: COLÓQUIO EM COMUNICAÇÃO E SOCIABILIDADE, 1., **Anais...** UFMG, Belo Horizonte, 2008.

BIN, Adriana; SALLES-FILHO, Sergio. Contributions to a Conceptual Framework of Technology and Innovation Planning at the Micro Level. In: ANNUAL MEETING ON SOCIO-ECONOMICA – SASE, 19., 2007, **Anais...** Copenhagen. NETWORK F Knowledge, Technology, and Innovation, 2007. v. 1. p. 1-12.

BOEHM, Barry W. A Spiral Model of Software Development and Enhancement. **TRW Defense**. Systems Group. IEEE Magazine. maio, 1988. p. 61-72.

BOSTON Consulting Group. Hacker Survey release 0.73, Disponível em: <http://flosscom.net/index.php?option=com_docman&task=doc_view&gid=45>. Acesso em: 02 abr. 2009.

BYRNE, J. A. The virtual corporation. **Business Week**, Feb. 1993.

CANTU, Carlos Henrique. **Firebird Essencial**. Rio de Janeiro: Editora Ciência Moderna Ltda, 2005.

CAPRA, F. **A teia da vida**: uma nova compreensão científica dos sistemas vivos. São Paulo: Cultrix, 1996.

CASAROTTO FILHO, Nelson, PIRES, Luis H. **Redes de pequenas e médias empresas e desenvolvimento local**: estratégias para a conquista da competitividade global com base na experiência italiana. São Paulo: Atlas, 1998. 148p.

CASTELLS, M. **A sociedade em rede**. 2 ed. São Paulo: Paz e Terra, 1999.

CASTELLS, M., Hacia el Estado Red?. Globalización económica e instituciones políticas en la era de la información. In: SEMINÁRIO INTERNACIONAL SOCIEDADE E A REFORMA DO ESTADO, 1998, São Paulo. **Anais...** São Paulo: [s.n.], 1998. p. 1-16.

CATAPULT, INC. **Microsoft Project 4 for windows: passo a passo.** São Paulo: Makron Books, 1995. 419 p.

CEDERQVIST, P. **Version Management with CVS.** Network Theory Ltd. 2006.

CHIAVENATO, I. **Administração de empresas: uma abordagem contingencial.** São Paulo: McGraw-Hill, 1982.

CHIAVENATO, I. **Introdução á teoria geral da administração.** 6. ed. Rio de Janeiro. Campus, 2000.

CHIAVENATO, I. **Gestão de pessoas.** 2. ed. rev. e atual. Rio de Janeiro: Elsevier, 2005. 529 p.

COLLINS-SUSSMAN Ben, FITZPATRICK Brian W., PILATO, C. Michael. **Version Control with Subversion For Subversion 1.4.** Disponível em: <<http://www-zeuthen.desy.de/dv/documentation/unixguide/infohtml/subversion/svn-book.html>>. Acesso em: 11 mai. 2009.

COMUNIDADE FIREBIRD. Disponível em: <http://www.comunidade-firebird.org/cflp/html_docs/001_FacSheet/Firebird%201_5%20Factsheet.htm>. Acesso em: 10 abr. 2009.

COSTA, Tiago da; DE ROLT, Carlos Roberto; DIAS, Júlio da Silva. O Conceito de Serviços Compartilhados como Fundamento para a Inovação dos Serviços Públicos Delegados no Brasil. In: SIMPÓSIO DE GESTÃO DA INOVAÇÃO TECNOLÓGICA, 25., 2008, Brasília. **Anais...** Brasília, 2008.

DE ROLT, C. R. **O desenvolvimento da comunidade virtual**: uma proposta para a melhoria da qualidade e da comercialização de software. 1999. 167 f. Tese. (Doutorado em Engenharia da Produção) – Universidade Federal de Santa Catarina, Florianópolis. 1999.

DRUCKER, P. **A nova sociedade das organizações**. In: HOWARD, R. (Org.) *Aprendizado organizacional*. Rio de Janeiro: Campus, 2000. p. 1-7.

ELSMARI, Ramez; NAVATHE, Shamrant B. **Sistemas de Bancos de Dados**. São Paulo: Addison Wesley, 2005.

FOSTER I., Kesselman C.; Tuecke S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. **International J. Supercomputer Applications**, v. 3, n. 15, 2001. Disponível em: <<http://hpc.sagepub.com/cgi/content/abstract/15/3/200>>. Acesso em: 28 ago. 2009.

FOUCAULT, Michel; MACHADO, Roberto. **Microfísica do poder**. 19. ed. São Paulo: GRAAL, 2004. 295 p.

FREITAS, Gustavo André de. PARIS, Riliane Alpoim. **FIREBIRD**. Linhares. 2007

GALLIVAN, M. J. Striking a balance between trust and control in a virtual organization: a content analysis of open source software case studies. **Information Systems Journal**, Blackwell Science Ltd., p. 277-204, 2001.

GNOME, Release Notes. **Notas de Liberação de Versão**. Disponível em: <<http://library.gnome.org/misc/release-notes/2.14/index.html.pt>>. Acesso em: 13 abr. 2009.

GNU Lesser General Public Licence. Versão 3, jun. 2007. Free Software Foundation Inc, USA. Disponível em: <<http://www.gnu.org/licenses/lgpl.txt>>. Acesso em: 02 jun. 2009.

GODOY, A. S. Introdução à pesquisa qualitativa e suas possibilidades. **Revista de Administração de Empresas**, São Paulo, v. 35, n. 2, p. 57-63, abr. 1995.

GRANDORI, A.; SODA, G. Inter-firm Network: antecedents, mechanisms and forms. **Organization Studies**, v. 2, n. 6, p. 183-214, 1995.

HAMEL, Gary; PRAHALAD, C. K. **Competindo pelo futuro: estratégias inovadoras para obter o controle do seu setor e criar os mercados de amanhã**. 7. ed Rio de Janeiro: Campus, c1995. 377p.

HEXSEL, Roberto A. **Propostas de Ações de Governo para Incentivar o Uso de Software Livre**. Relatório Técnico do Departamento de Informática da UFPR. Curitiba, 2002. 53 p.

HIMANEM, Pekka. **A ética dos hackers e o espírito da era da informação: a importância dos exploradores da era digital**. Rio de Janeiro: Campus, 2001.

INSTITUTO NACIONAL DE TECNOLOGIA DE INFORMAÇÃO. **Estudo sobre software livre**. Fundação Getúlio Vargas, Rio de Janeiro, 2005.

JÄGERS, H.; JANSEN, W.; STEENBAKKERS, W. **Characteristics of Virtual Organizations**. In: SIEBER, P.; GRIESE, J. Organizational virtualness and electronic commerce. Proceedings of the 2nd VoNet - workshop, Bern: Simowa Verlag, Zurich, p. 65-76, sep. 1998.

KATZY, Bernhard. The Business Architect: The Concept of Enterprise Integration Revisited. In: European Conference on Virtual Enterprises and Networked Solutions - **New Perspectives on Management**, communication and Information Technology. Paderbon, Germany, Abr. 1997.

KDE.Bug Tracking System. Disponível em: < <http://bugs.kde.org/>>. Acesso em: 10 mar. 2009.

KORTH, Henry F; SILBERSCHATZ, Abraham. **Sistema de banco de dados**. 2. ed. rev. Rio de Janeiro: Makron Books, c1995. 754 p.

LAKHANI, R. K.; WOLF, R. G. Why hackers do what they do: understanding motivation ad effort in free/open source Software projects. 2005. Disponível em: <<http://freeSoftware.mit.edu/papers/lakhaniwolf.pdf>>. Acesso em: 01 abr. 2009

LEHMANN, F. **Floss developers as social formation**. First Monday, 2004. Disponível em: <<http://www.firstmonday.org/issues/issue9.11/lehmann/>>. Acesso em: 02 abr. 2009.

LEVY, Pierre. **Cibercultura**. São Paulo: 34. ed. 1999. 260 p.

LUFTMAN, J. N.; LEWIS, P. R.; OLDACH, S. H. **Transforming The Enterprise: The Alignment Of Business And Information Technology Strategies**. IBM Systems Journal, v. 32, n. 1, p. 198-221, 1993.

MALHOTRA, Naresh K. **Pesquisa de marketing: uma orientação aplicada**. 3. ed. Porto Alegre: Bookman, 2001. 719 p.

MARCONI, Marina de Andrade; LAKATOS, Eva Maria. **Técnicas de pesquisa: planejamento e execução de pesquisas, amostragens e técnicas de pesquisa, elaboração, análise e interpretação de dados**. 3. ed. rev. e ampl. São Paulo: Atlas, 1996, c1985. 231p.

MASLOW, A. H. **Motivation and a Personality**. New York: Harper, 1970.

MATOS, Luis M. Camarinha. **Organizações virtuais**. Lisboa: Mimeo, Universidade Nova de Lisboa, 1997. 35 p.

MENDES, C. I. C. **Software Livre e Inovação Tecnológica: uma Análise sob a Perspectiva da Propriedade Intelectual**. 297 f. Dissertação (Mestrado). Instituto de Economia. Universidade Estadual de Campinas. Campinas, 2006.

MERRIAM, S. B. **Qualitative research and case study applications in education.** Jossey-Bass Publishers: San Francisco, 1988.

MERTENS, Peter; FAISST, Wolfgang. Virtuelle Unternehmen - eine Organisationsstruktur für die Zukunft? **Technologie & management**, v. 44, n. 2, 1995, p. 61-68.

MINISTÉRIO da Ciência e Tecnologia. **Pesquisa nacional de qualidade e produtividade no setor de software brasileiro.** Brasília: 2003.

MINTZBERG, Henry; AHLSTRAND, Bruce; LAMPEL, Joseph. **Safari de estratégia: um roteiro pela selva do planejamento estratégico.** Porto Alegre: Bookman, 2000. viii, 299 p.

MIRANDA, Antonio; SIMEÃO, Elmira. **Ciência da informação: teoria e metodologia de uma área em expansão.** Brasília, DF, Thesaurus, 2003. 212 p.

MOINEAU, L.; PAPATHÉODOROU, A. Cooperação e produção imaterial em softwares livres – elementos para uma leitura política do fenômeno GNU/LINUX. **Revista Lugar Comum**, Rio de Janeiro, n. 11, 2000.

NASCIMENTO, Ana Lúcia do. **A indústria brasileira de software.** Informativo **Softex**. Campinas, 2001.

NETO, José Amaro. OLAVE, Maria Elena León. **Uma Análise de Redes de Cooperação nas Pequenas e Médias Empresas do Setor das Telecomunicações.** Dissertação (Mestrado). Engenharia de Produção da Escola Politécnica da Universidade de São Paulo, São Paulo, 1998.

NEVES, José Luis. **Pesquisa qualitativa: características, usos e possibilidades.** FEA-USP. São Paulo, v. 1. n. 3, 1996.

NONAKA, Ikujiro; TAKEUCHI, Hirotaka. **Criação de conhecimento na empresa:** como as empresas japonesas geram a dinâmica da inovação. 5. ed Rio de Janeiro: Campus, 1997. 358 p.

O'LEARY, Daniel E.; KUOKKA, Daniel; PLANT, Robert. Artificial Intelligence and Virtual Organizations. **Communications of the ACM**, New York, v. 40, n. 1. p. 52-59, Jan. / 1997. Disponível em: <<http://portal.acm.org/citation.cfm?id=242871>>. Acesso em: 27 ago. 2009.

PALMIERI John. Current Year Budget - GNOME. Disponível em: <<http://mail.gnome.org/archives/foundation-announce/2009-April/msg00002.html>>. Acesso em: 20 jun. 2009.

PEREIRA, S. L. **Estruturas de Dados Fundamentais:** Conceitos e Aplicações. São Paulo: Érica, 1996.

POWELL, Walther W. Neither Market for Hierarchy network forms of organization. Research in Organizational Behavior. 1990 Disponível em: <http://www.stanford.edu/~woodyp/papers/powell_neither.pdf>. Acesso em: 27 ago. 2009.

PRESSMAN, Roger. **Engenharia de software**, Makron Books, 1995.

QUEIROZ, Luiz. Cresce a demanda pelo software local. 2003. Disponível em: <<http://www.softwarelivre.mt.gov.br/destaque.php?BASE=destaque&ID=69>>. Acesso em: 28 ago. 2009.

RAYMOND, Eric. **The Cathedral and the Bazaar:** Musings on Linux and Open Source by an Accidental Revolutionary. Sebastopol: O Reilly Media, 2001.

REIS, C. R. **Caracterização de um Processo de Software para Projetos de Software Livre.** 2003. 195 f. Dissertação (Mestrado) - Instituto de Ciências Matemáticas e de Computação de São Carlos, Universidade de São Paulo, 2003.

REZENDE, J. L. **Aplicando Técnicas de Conversação para a Facilitação de Debates no Ambiente AulaNet**. 2003. 170 f. Dissertação (Mestrado) - Departamento de Informática, PUC-Rio, 2003.

ROBREDO, Jaime. **Da ciência da informação revisitada aos sistemas humanos de informação**. Brasília (DF): Thesaurus Editora: SSRR Informações Consultoria e Projetos Ltda, 2003. 245 p.

RUOSO, Daniel. **A construção e a desconstrução da legitimidade de pessoas e de idéias na Comunidade Software Livre - um estudo sobre a Debian**. Universidade Federal do Ceará. Fortaleza, Ceará 2006.

SALLES-FILHO, Sergio. **Comparativo econômico software livre e software proprietário**. Instituto de Geociências, Universidade Estadual de Campinas – Unicamp. Campinas: 2006.

SALOMON, Delcio Vieira. **Como fazer uma monografia**. 10. ed. Sao Paulo: Martins Fontes, 2001. 412 p.

SANTOS, R. P. C. **As tarefas para a gestão de processos**. 2007. 471 f. Tese (Doutorado em Ciências) - Universidade Federal do Rio de Janeiro. Rio de Janeiro. 2007.

SILVEIRA, Sérgio Amadeu da. **Software Livre: a luta pela liberdade do conhecimento**. São Paulo: Editora Fundação Perseu Abramo, 2004.

SOARES, S. B. C.; AMARAL, S. F. Comunidades de aprendizagem colaborativa a distância via internet e bibliotecários de referência de bibliotecas universitárias brasileiras. In: SEMINÁRIO NACIONAL DE BIBLIOTECAS UNIVERSITÁRIAS, 13.; SIMPÓSIO DE DIRETORES DE BIBLIOTECAS UNIVERSITÁRIAS DA AMÉRICA LATINA E DO CARIBE, 3., **Anais...** 2004.

STALLMAN, R. Free software, free society: selected essays of Richard Stallman Boston. **GNU Press**. 2002. Disponível em: <<http://www.gnu.org/philosophy/fsfs/rms-essays.pdf>>. Acesso em: 28 ago. 2009.

STRAUSAK, Nicole. Resumée of VoTalk. In: SIEBER, Pascal, GRIESE, Joachim (eds). **Organizational Virtualness**. Proceedings of the VoNet - Workshop, Apr. 27-28, 1998. Bern, Simona Verlag Bern, 1998, p. 9-24.

TAPSCOTT, Don, LOWY, Alex; TICOLL, David. **Capital Digital**. São Paulo: Makron Books, 2000.

TAPSCOTT, Don; WILLIMS, Anthony D. **Wikinomics**: como a colaboração em massa pode mudar o seu negócio. Rio de Janeiro: Nova Fronteira, 2007. 367 p.

TAURION, Cezar. **Software Livre**: Potencialidades e Modelos de Negócios. Brasport, 2004.

TEOREY, T. J. **Database modeling & design**. San Francisco: Morgan Kaufman, 1999.

TIGRIS, Subversion FAQ. **Perguntas Mais Frequentes Sobre Subversion**. Disponível em: <<http://subversion.tigris.org/faq.html>>. Acesso em: 23 jun. 2009.

TIROLE, Jean; LERNER, Josh. Some Simple Economics of Open Source. **The Journal of Industrial Economics**, v. 50, n. 2, p. 197-234, jun. 2002.

TRIANA, Yago Quiñones. O estudo do fenômeno do Software Livre como um Movimento Social. In: SEMINÁRIO NACIONAL: MOVIMENTOS SOCIAIS, PARTICIPAÇÃO E DEMOCRACIA, 2., 2007, Florianópolis, **Anais...**, Florianópolis: NPMS, 2007. v. 3. p. 715-729. Disponível em: <www.sociologia.ufsc.br/npms/yago_quinones_triana.pdf>. Acesso em: 27 ago. 2009.

TRÖGER, Ane. Um Estudo sobre Organizações Virtuais. T.I n. 692. 1997. CPGCC-UFRGS.

TROPE, Alberto. **Organização Virtual: Impactos do teletrabalho nas organizações**. Rio de Janeiro: Qualitymark Editora, 1999. 104 p.

VAN BEMMEL, J. H. **Handbook of Medical Informatics**. 1999.

VELOSO Francisco, et al. **Slicing the knowledge-based economy in Brazil, China and India: a tale of 3 software industries**. Massachusetts Institute of Technology, 2003. Disponível em: <http://en.brazilny.org/images/secomfiles/Brazil_-_MIT_Study.pdf>. Acesso em: 27 ago. 2009.

WILDEMAN, Leo. Organização virtual. **Revista HSM Management**, São Paulo: HSM do Brasil, n. 15, ano 3, jul./ago, p.74-80, 1999.

YAMAUCHI, Y. et al. Collaboration with Lean Media: How Open Source Succeeds. **Proceedings of CSCW**, 2000. ACM Press. p. 329–337.

YIN, Robert K. . **Estudo de caso: planejamento e métodos**. 2. Ed. Porto Alegre: Bookman, 2001. 205 p.

YOUNG, Michael. **The Rise of the Meritocracy**. Londres, Thames and Hudson, 1958.

ZIMMERMANN, Frank. Structural and Manageral Aspects of Virtual Enterprises. **Business Information Systems**, Germany. 1996.

GLOSSÁRIO

Chat On-line – Software que permite a comunicação por meio do computador entre pessoas mediante mensagens instantâneas utilizando para tal uma rede específica ou a Internet.

Ciberespaço – Rede informacional construída pelo entrelaçamento de meios de telecomunicação e informática, utilizando os tanto digitais quanto analógicos, em escala global ou regional. É um novo meio de comunicação que surge da interconexão mundial dos computadores que nomeia não apenas a infra-estrutura material da comunicação digital, e também os seres humanos que navegam e alimentam esse universo (LÉVY, 1999).

Comunidades de Software Livre – Organização virtual que utiliza de redes dinâmicas de cooperação, através da utilização das novas tecnologias de informação para o desenvolvimento de um projeto (normalmente de software).

Fork – Termo utilizado nas comunidades de Software Livre para determinar um grupo de pessoas que participavam de um projeto e que, a partir de um determinado momento, desvincularam-se do projeto inicial para fazer sua própria versão do projeto, aproveitando parte do código original.

Organizações virtuais – Rede temporária de organizações independentes, ligadas pela tecnologias de informação e comunicação, e que em sua forma mais pura se ligam a outras para formar uma cooperação, contribuindo com o que considerar sua competência central (BYRNE, 1993).

Processo - O processo é um conjunto de atividades sequenciais (conectadas), relacionadas e lógicas que tomam um *input* com um fornecedor, acrescentam valor a este e produzem um *output* para um consumidor (Harrington, 1993).

Sistema Desktop – Sistema de informação que utiliza-se de interface gráfica e de aplicativos para estabelecer uma camada de interação entre com o usuário com o computador.

ANEXOS

ANEXO A - QUESTIONÁRIO DE PESQUISA 1

MODELO DE GESTÃO DE PROJETOS COOPERADOS BASEADO NO SOFTWARE LIVRE

Foco de Estudo: Projeto Firebird

Objetivo:

Este questionário destina-se à obtenção de dados para subsidiar Dissertação de Mestrado na Universidade do Estado de Santa Catarina (UDESC). As informações aqui prestadas são absolutamente sigilosas e os resultados serão apresentados sempre na forma percentual, não havendo identificação do respondente.

Caso deseje receber um resumo dos resultados, preencha o campo disponível abaixo. Ao término do preenchimento, por favor, devolva o presente questionário via e-mail para o endereço do remetente.

O objetivo deste questionário são os seguintes:

- Identificar as características dos colaboradores atuantes no Projeto Firebird;
- Observar as percepções destes colaboradores da tecnologia de gestão adotada por esta rede;
- Buscar a compreensão do fluxo de informações dentro da rede, bem como a importância das ferramentas de tecnologias de informação adotadas;
- Estabelecer um paralelo entre a plataforma virtual de gestão da rede com a teoria de Organizações Virtuais.

Deseja Receber um Resumo dos Resultados: () Sim () Não

Caso Positivo, especifique um endereço eletrônico para a remessa:

Questões

1. Qual a sua motivação para participar da comunidade?
 - Aquisição/Compartilhamento de Conhecimento
 - Desenvolvimento Tecnológico
 - Contribuição com a Sociedade/Comunidade
 - Necessidade Profissional/Comercial
2. Quem aprova as alterações realizadas?
3. Como funciona a comunicação entre os desenvolvedores do projeto?
4. Como ocorre a difusão do conhecimento através da rede de desenvolvedores do Firebird?
5. Quantos são os desenvolvedores que trabalham exclusivamente no Projeto Firebird?
6. Quem define os roadmaps do projeto?
7. Existe algum tipo de broker (uma ou mais pessoas que coordenam a rede de cooperação) definido no Projeto Firebird ?
8. Existe algum tipo de hierarquia no projeto? Se existir, em que está baseada esta hierarquia?
9. Qual o retorno que você recebe do projeto?
 - Conhecimento Satisfação Pessoal Desenvolvimento Tecnológico
 - Outro: _____
10. Quais são as tecnologias de informação e comunicação utilizadas para a cooperação no projeto?
11. Existe alguma ferramenta de gestão do projeto? Qual é esta ferramenta? Como você avalia a eficiência desta ferramenta?
12. Quais são as dificuldades que você observa no projeto?

ANEXO B - QUESTIONÁRIO DE PESQUISA 2

MODELO DE GESTÃO DE PROJETOS COOPERADOS BASEADO NO SOFTWARE LIVRE

Foco de Estudo: Projeto GNOME

Objetivo:

Este questionário destina-se à obtenção de dados para subsidiar Dissertação de Mestrado na Universidade do Estado de Santa Catarina (UDESC). As informações aqui prestadas são absolutamente sigilosas e os resultados serão apresentados sempre na forma percentual, não havendo identificação do respondente.

Caso deseje receber um resumo dos resultados, preencha o campo disponível abaixo. Ao término do preenchimento, por favor, devolva o presente questionário via e-mail para o endereço do remetente.

O objetivo deste questionário são os seguintes:

- Identificar as características dos colaboradores atuantes no Projeto GNOME;
- Observar as percepções destes colaboradores da tecnologia de gestão adotada por esta rede;
- Buscar a compreensão do fluxo de informações dentro da rede, bem como a importância das ferramentas de tecnologias de informação adotadas;
- Estabelecer um paralelo entre a plataforma virtual de gestão da rede com a teoria de Organizações Virtuais.

Deseja Receber um Resumo dos Resultados: () Sim () Não

Caso Positivo, especifique um endereço eletrônico para a remessa:

Questões

1. Qual a sua motivação para participar da comunidade?
 - Aquisição/Compartilhamento de Conhecimento
 - Desenvolvimento Tecnológico
 - Contribuição com a Sociedade/Comunidade
 - Necessidade Profissional/Comercial
2. Quem aprova as alterações realizadas?
3. Como funciona a comunicação entre os desenvolvedores do projeto?
4. Como ocorre a difusão do conhecimento através da rede de desenvolvedores do GNOME?
5. Quantos são os desenvolvedores que trabalham exclusivamente no Projeto GNOME?
6. Quem define os roadmaps do projeto?
7. Existe algum tipo de broker (uma ou mais pessoas que coordenam a rede de cooperação) definido no Projeto GNOME ?
8. Existe algum tipo de hierarquia no projeto? Se existir, em que está baseada esta hierarquia?
9. Qual o retorno que você recebe do projeto?
 - Conhecimento Satisfação Pessoal Desenvolvimento Tecnológico
 - Outro: _____
10. Quais são as tecnologias de informação e comunicação utilizadas para a cooperação no projeto?
11. Existe alguma ferramenta de gestão do projeto? Qual é esta ferramenta? Como você avalia a eficiência desta ferramenta?
12. Quais são as dificuldades que você observa no projeto?

ANEXO C - QUESTIONÁRIO DE PESQUISA 3

MODELO DE GESTÃO DE PROJETOS COOPERADOS BASEADO NO SOFTWARE LIVRE

Foco de Estudo: Projeto TinyCobolBR

Objetivo:

Este questionário destina-se à obtenção de dados para subsidiar Dissertação de Mestrado na Universidade do Estado de Santa Catarina (UDESC). As informações aqui prestadas são absolutamente sigilosas e os resultados serão apresentados sempre na forma percentual, não havendo identificação do respondente.

Caso deseje receber um resumo dos resultados, preencha o campo disponível abaixo. Ao término do preenchimento, por favor, devolva o presente questionário via e-mail para o endereço do remetente.

O objetivo deste questionário são os seguintes:

- Identificar as características dos colaboradores atuantes no Projeto TinyCobolBR;
- Observar as percepções destes colaboradores da tecnologia de gestão adotada por esta rede;
- Buscar a compreensão do fluxo de informações dentro da rede, bem como a importância das ferramentas de tecnologias de informação adotadas;
- Estabelecer um paralelo entre a plataforma virtual de gestão da rede com a teoria de Organizações Virtuais.

Deseja Receber um Resumo dos Resultados: () Sim () Não

Caso Positivo, especifique um endereço eletrônico para a remessa:

Questões

1. Qual a sua motivação para participar da comunidade?
 - Aquisição/Compartilhamento de Conhecimento
 - Desenvolvimento Tecnológico
 - Contribuição com a Sociedade/Comunidade
 - Necessidade Profissional/Comercial
2. Quem aprova as alterações realizadas?
3. Como funciona a comunicação entre os desenvolvedores do projeto?
4. Como ocorre a difusão do conhecimento através da rede de desenvolvedores do TinyCobolBR?
5. Quantos são os desenvolvedores que trabalham exclusivamente no Projeto TinyCobolBR?
6. Quem define os roadmaps do projeto?
7. Existe algum tipo de broker (uma ou mais pessoas que coordenam a rede de cooperação) definido no Projeto TinyCobolBR ?
8. Existe algum tipo de hierarquia no projeto? Se existir, em que está baseada esta hierarquia?
9. Qual o retorno que você recebe do projeto?
 - Conhecimento Satisfação Pessoal Desenvolvimento Tecnológico
 - Outro: _____
10. Quais são as tecnologias de informação e comunicação utilizadas para a cooperação no projeto?
11. Existe alguma ferramenta de gestão do projeto? Qual é esta ferramenta? Como você avalia a eficiência desta ferramenta?
12. Quais são as dificuldades que você observa no projeto?