

**UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS
Departamento de Engenharia de Produção**

TEOBALDO RIVAS

**Objetos de aprendizagem no contexto das comunidades virtuais
auto-organizadas para a produção de software livre e de código
aberto**

**São Carlos
2009**

Livros Grátis

<http://www.livrosgratis.com.br>

Milhares de livros grátis para download.

TEOBALDO RIVAS

Objetos de aprendizagem no contexto das comunidades virtuais auto-organizadas para a produção de software livre e de código aberto

Tese apresentada à Escola de Engenharia de São Carlos da Universidade de São Paulo, para a obtenção do título de Doutor em Engenharia de Produção.

Área de Concentração: Economia, Organizações e Gestão do Conhecimento.
Orientador: Prof. Dr. Edson Walmir Cazarini.

**São Carlos
2009**

AUTORIZO A REPRODUÇÃO E DIVULGAÇÃO TOTAL OU PARCIAL DESTE TRABALHO, POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

Ficha catalográfica preparada pela Seção de Tratamento
da Informação do Serviço de Biblioteca – EESC/USP

R618o Rivas, Teobaldo
Objetos de aprendizagem no contexto das comunidades virtuais auto-organizadas para a produção de software livre e de código aberto / Teobaldo Rivas ; orientador Edson Walmir Cazarini. -- São Carlos, 2009.

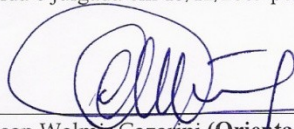
Tese (Doutorado-Programa de Pós-Graduação em Engenharia de Produção e Área de Concentração em Economia, Organizações e Gestão do Conhecimento) -- Escola de Engenharia de São Carlos da Universidade de São Paulo, 2009.

1. Aprendizagem tecnológica. 2. Objetos de aprendizagem. 3. Software livre e de código aberto. 4. Comunidades virtuais auto-organizadas. 5. Educação virtual. 6. Gestão do conhecimento. I. Título.

FOLHA DE JULGAMENTO

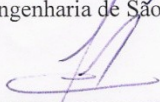
Candidato: Bacharel TEOBALDO RIVAS.

Tese defendida e julgada em 15/12/2009 perante a Comissão Julgadora:



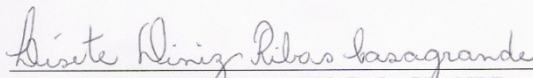
Prof. Dr. Edson Waldir Cazarini (**Orientador**)
(Escola de Engenharia de São Carlos/USP)

Aprovado



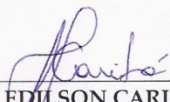
Prof. Dr. **JOSÉ DUTRA DE OLIVEIRA NETO**
(Faculdade de Economia, Administração e Contabilidade de Ribeirão Preto/USP)

APROVADO



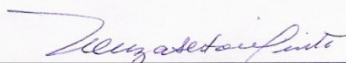
Prof^a. Dr^a. **LISETE DINIZ RIBAS CASAGRANDE**
(Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto/USP)

Aprovado



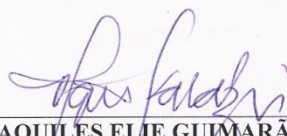
Prof. Dr. **EDILSON CARLOS CARITÁ**
(Universidade de Ribeirão Preto/UNAERP)

APROVADO



Prof^a. Dr^a. **NEUSA BERTONI PINTO**
(Pontifícia Universidade Católica/PUC/Campus do Paraná)

APROVADO



Prof. Associado **AQUILES ELIE GUIMARÃES KALATZIZ**
Coordenador do Programa de Pós-Graduação em
Engenharia de Produção



Prof. Titular **GERALDO ROBERTO MARTINS DA COSTA**
Presidente da Comissão da Pós-Graduação da EESC

2012
Serviço de Pós-Graduação
15/12/09
Procedimento 4

DEDICATÓRIA

À minha Mãe Honorfía e ao meu Pai Edison (*in memorian*), os quais foram os grandes incentivadores dos meus estudos, para a busca do conhecimento como forma de progresso moral, pessoal e profissional.

À minha querida esposa Noeli, que me estimulou e me incentivou a pesquisar sobre educação e tecnologia e, sobretudo, a continuar os meus estudos na pós-graduação *strictu-sensu*.

À minha irmã Isamar e ao meu irmão Ricardo, pelo amor e carinho que sempre tiveram por mim.

Ao Professor Amazílio (*in memorian*), que foi o meu primeiro Professor, do curso primário, que correu o risco de me aceitar como aluno, mesmo sem eu ter a idade mínima para o ingresso no curso.

AGRADECIMENTOS

Nos dias atuais, não é possível desenvolver qualquer trabalho isoladamente, sobretudo quando se trata de pesquisa de caráter científico. Dezenas de pessoas, direta ou indiretamente, colaboraram, ajudaram, motivaram e incentivaram, para que esta tese pudesse assim ser realizada. Portanto, agradeço:

A Deus por permitir a minha existência.

Ao Professor Dr. Wilson Kendy Tachibana, que me abriu as portas da Escola de Engenharia de São Carlos da USP.

Ao Professor Dr. Renato Vairo Belhot, que me aceitou como orientando do curso de Mestrado e me estimulou a iniciar pesquisas na área da educação aplicada à engenharia.

Ao Professor Dr. Edson Walmir Cazarini, sempre muito paciente e acolhedor, que me estimulou a continuar a pós-graduação com o curso de Doutorado.

À Professora Dra. Lisete Ribas Casagrande, como membro da banca de qualificação, pelas valiosas considerações efetuadas sobre a metodologia da pesquisa.

À Professora Dra. Neuza Bertoni Pinto, como membro da banca de qualificação, pelas relevantes sugestões para o aprimoramento da pesquisa.

Ao Sr. José Luiz Chiaretto, pelo zelo e dedicação em me manter sempre informado sobre os prazos, regras e demais orientações do Programa.

Ao *United Nations Development Programme* – UNDP da Guiné Equatorial, por me disponibilizar, quando lá trabalhava, todos os recursos possíveis, para que eu iniciasse a escrita desta tese, superando todas as restrições e dificuldades locais.

EPÍGRAFE

“O amor, o conhecimento e o trabalho são as fontes da nossa vida. Deveriam também governá-las.”

Wilhelm Reich

RESUMO

RIVAS, T. (2009). **Objetos de aprendizagem no contexto das comunidades virtuais auto-organizadas para a produção de software livre e de código aberto.** 327 f. Tese (Doutorado) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2009.

A presente tese tem como objetivo produzir subsídios para a construção de um referencial teórico-metodológico sobre a utilização dos objetos de aprendizagem no contexto das comunidades virtuais auto-organizadas para a produção de software livre e de código aberto. Utiliza-se como metodologia a etnografia virtual (HINE, 2000) em conjunto com a teoria analítica da ação mediada (WERTSCH, 1991) e a análise de conteúdo (BARDIN, 2000). A coleta de dados foi efetuada, em duas fases, nos fóruns de discussão de quatro (4) comunidades, além de uma comunidade incubadora de desenvolvimento de projetos de software livre e de código aberto. Constata-se que os objetos de aprendizagem mediam o processo de solução de problemas, uma vez que 100% dos problemas da amostra analisada foram resolvidos, apesar da ausência de um padrão de conformidade desses objetos. Esta limitação é superada em razão do perfil específico imperante dos membros ativos da comunidade, pois estes possuem elevado nível de comportamento colaborativo/cooperativo, iniciativa voluntária, desprendimento, obstinação, capacidade de autoaprendizagem, autonomia, independência, disciplina, responsabilidade e comprometimento com prazos, qualidade dos produtos e outras exigências estabelecidas pela comunidade. Outro fator de relevância é que as comunidades são ricas em interação humana, o que qualifica o processo do significado da mediação e o ambiente de colaboração, nas ações referentes à localização, montagem e contextualização dos objetos de aprendizagem. Os significativos resultados atingidos por estas comunidades têm impactado, sobremaneira, as grandes organizações de produção de software, levando-as a rever suas estratégias corporativas, boas práticas de desenvolvimento, gestão de pessoas, equipes e projetos. Por outro lado, infere-se que a sustentabilidade de tais comunidades não pode estar assentada somente em atributos e habilidades pessoais, principalmente pelo fato de que a localização dos objetos de aprendizagem para a solução de problemas baseia-se no conhecimento tácito de seus membros. Necessário se faz agregar inovações na forma e funcionalidade de tais comunidades (padrão de conformidade, métodos e ferramentas tecnológicas), com vistas a possibilitar uma efetiva e universal acessibilidade do conhecimento produzido para a solução mais eficiente dos problemas, bem como incorporar membros com comportamentos e habilidades diversos. Os resultados desta pesquisa contribuem para as inovações futuras, no campo teórico e prático, na definição de um padrão de conformidade para a especificação, indexação, uso, combinação e avaliação dos objetos de aprendizagem, além de motivar a mudança de comportamento, cultura e forma de aprender.

Palavras-chave: Aprendizagem tecnológica. Comunidades virtuais auto-organizadas. Software livre e de código aberto. Objetos de aprendizagem. Educação virtual. Gestão do conhecimento.

ABSTRACT

RIVAS, T. (2009). **Learning objects in the context of self-organized virtual communities for the development of free and open source software.** 327 f. Tese (Doutorado) – Escola de Engenharia de São Carlos, universidade de São Paulo, São Carlos, 2009.

The present thesis aims to produce subsidies for the development of a theoretical-methodological referential on the use of learning objects in the context of self-organized virtual communities for the development of free and open source software. The methodologies used were the virtual ethnography (HINE, 2000), together with the mediated action analytical theory (WERTSCH, 1991) and content analysis (BARDIN, 2000). Data was collected in two phases: from the discussion forums of (4) four communities, and from an incubator community for the development of free and open source software projects. The learning objects appear to mediate the problem solution processes, since all the problems of the analyzed sample were solved, despite the lack of a standard of compliance of those objects. This limitation is overcome due to the specific dominant profile of the active members of the community, who display a high level of collaborative behavior, voluntary initiative, detachment, obstinacy, auto-learning capacity, autonomy, independence, discipline, responsibility and commitment to deadlines, product quality and other requirements established by the community. Another relevant aspect is that the communities are abundant in human interaction, what qualifies the mediation significance process and the collaborative environment in the actions referring to location, assembling and contextualization of the learning objects. The meaningful results obtained by those communities have led great software production organizations to review their corporate strategies, good developmental practices, staff, people and project management. On the other hand, it is inferred that the sustainability of those communities can not be maintained only by personal attributes and abilities, specially because the location of the learning objects for problem solution is based on the tacit knowledge of their members. It is necessary to aggregate innovation into the nature and functionality of those communities (standard compliance, technological methods and tools), to not only enable an effective and universal accessibility to knowledge leading to more efficient problem solution, but also to incorporate members with diverse behavior and abilities. The results of the present research contribute to future innovation in both theoretical and practical fields in the definition of a pattern of conformity for the specification, indexation, use, combination and evaluation of learning objects, in addition to motivating a change of behavior, culture and way of learning.

Keywords: Technological learning. Self-organized virtual communities. Free and open source software. Learning objects. Virtual education. Knowledge management.

LISTA DE QUADROS

Quadro 1 - Relação de softwares livre e de código aberto	45
Quadro 2 - Enfoque e categorias das teorias de aprendizagem	77
Quadro 3 – Princípios e estratégias para a formação de comunidades virtuais.....	97
Quadro 4 - Seleção e qualificação das CVAO e respectivos endereços URL.....	165
Quadro 5 - Período de coleta de dados por comunidade	166
Quadro 6 - Composição do código para o processo de discussão	167
Quadro 7 - Abreviação dos nomes das comunidades	167
Quadro 8 - Tipo de problema, respectivo encaminhamento da solução e situação - Comunidade Linux	169
Quadro 9 - Tipo de problema, respectivo encaminhamento da solução e situação - Comunidade Apache.....	170
Quadro 10 - Tipo de problema, respectivo encaminhamento da solução e situação - Comunidade MySQL.....	171
Quadro 11 - Tipo de problema, respectivo encaminhamento da solução e situação - Comunidade PHP	172
Quadro 12 - Tipo de problema, encaminhamento da solução e situação - Comunidade Sourceforge.....	173
Quadro 13 - Análise taxonômica dos fóruns das CVAO	176

LISTA DE ILUSTRAÇÕES

Figura 1- Critérios para adoção de produtos e serviços Linux	52
Figura 2- Elementos das comunidades virtuais	96
Figura 3 - Demonstração esquemática da atividade mediada.....	111
Figura 4 - Dialogicidade em Bakhtin: enunciado	115
Figura 5 - Fundamentos metodológicos do estudo da língua	120
Figura 6 - Pentagrama das telas terminísticas da ação humana.....	121
Figura 7 - Modelo topológico fundado na teoria da ação mediada	125
Figura 8 - Mediação e artefatos culturais na perspectiva de Cole.....	130
Figura 9 - Representação gráfica da metodologia utilizada	145
Figura 10- Relação entre a proposta metodológica e o contexto investigado	155
Figura 11 - Equivalência do modelo no contexto do fórum de discussão	157
Figura 12 – Figura esquemática da arquitetura LAMP	164
Figura 13 - Demonstração gráfica – a relação da ação mediada com o fórum de discussão .	189
Figura 14 - Discussão sob o código 1.4.SFG	199
Figura 15 - Interação 1.8.LNX	204

LISTA DE TABELAS

Tabela 1 - Problemas e interações da comunidade Linux – Resumo de Ocorrências	169
Tabela 2 - Problemas e interações da Comunidade Apache – Resumo de Ocorrências	170
Tabela 3 – Problemas e interações da Comunidade MySQL – Resumo de Ocorrências	171
Tabela 4 – Problemas e interações da Comunidade PHP – Resumo de Ocorrências.....	172
Tabela 5 – Problemas e interações da Comunidade Sourceforge – Resumo de Ocorrências	173
Tabela 6 - Tabela resumo analítico quantitativo – interações dos fóruns e listas de discussão e solução de problemas	174
Tabela 7 - Tabela resumo - situação dos problemas apresentados por comunidade	176

LISTA DE SIGLAS

ADL	Advanced Distributed Learning Network
AICC	Aviation Industry CBT Committee
ARIADNE	Alliance of Remote Instructional Authoring & Distribution Networks for Europe
ASF	The Apache Software Foundation
BSD	Berkeley Software Distribution
CVAO	Comunidades Virtuais Auto-Organizadas
FAQ	Frequent Answers and Questions
FOSS	Free and Open Source Software
GNU	GNU is not Unix
IEEE	Institute of Electrical and Electronics Engineers
LINUX	Contração de Linus e Unix
LMS	Learning Management System
LTSC	Learning Technology Standards Committee
MH	Método de Harvard
NCSA	National Center for Supercomputer Applications
OA	Objetos de Aprendizagem
OSI	Open Source Initiative
OSSOS	Online Self-Organizing Social Systems
PBL	Problem Based Learning
RIO	Reusable Information Objects
RLO	Reusable Learning Objects Strategy
SCO	Sharable Content Objects
SCORM	Sharable Content Object Reference Model

SLCA	Software Livre e de Código Aberto
UA	Unidade de Análise
UC	Unidade de Contexto
UR	Unidade de Registro
ZDP	Zona de Desenvolvimento Proximal

SUMÁRIO

INTRODUÇÃO	29
Objetivo	33
Contribuição	33
Procedimentos metodológicos	34
Organização da tese	35

CAPÍTULO I

1 REVISÃO DA LITERATURA	39
1.1 Software livre e de código aberto	39
1.1.1 O movimento do software livre	40
1.1.2 O movimento do software de código aberto	41
1.1.3 Principais produtos de software livre e de código aberto	44
1.1.4 O processo de criação e desenvolvimento	47
1.1.5 Um novo modelo de negócio	50
1.1.6 A necessidade de um padrão de conformidade	55
1.2 Objetos de aprendizagem (OA)	56
1.2.1 Os enfoques de objetos de aprendizagem e as pesquisas atuais de ensino e aprendizagem	60
1.2.2 Outras questões relacionadas com os enfoques de objetos de aprendizagem	64
1.2.3 Em direção a uma nova teoria	66
1.3 SCORM – <i>Sharable content object reference model</i>	67
1.3.1 O SCORM e outros padrões de conformidade	70
1.4 <i>Fóruns de discussão</i>	72
1.4.1 Conceituação	73
1.4.2 Taxonomias	74
1.5 <i>Teorias de aprendizagem</i>	76
1.5.1 Enfoque colaborativo	77
1.5.2 Enfoque baseado na experiência	81
1.5.3 Enfoque baseado na solução de problemas	85
1.6 <i>Comunidades virtuais</i>	94
1.7 <i>Comunidades virtuais de aprendizagem auto-organizadas</i>	99
1.7.1 O fenômeno da auto-organização	99
1.7.2 Sistemas sociais online auto-organizados	100

1.8 <i>Mediação, dialogicidade, artefatos culturais e vozes da mente: conceitos que se entrecruzam na perspectiva de Vigotsky, Bakhtin, Wertsch e Cole</i>	101
1.8.1 Vigotsky e os conceitos de mediação e operações de pensamento.....	101
1.8.2 A dialogicidade em Bakhtin.....	112
1.8.3 Wertsch: Aproximação sociocultural da mente, linguagens sociais e ação.....	120
1.8.4 Aportes críticos de Cole: mediação e os artefatos culturais.....	129

CAPÍTULO II

2 METODOLOGIA	145
2.1 <i>Etnografia Virtual</i>	145
2.2 <i>Teoria analítica da ação mediada como método de análise</i>	155
2.3 <i>Análise documental</i>	157
2.4 <i>Critérios de seleção das comunidades virtuais</i>	159

CAPÍTULO III

3 RESULTADOS E DISCUSSÃO	163
3.1 <i>Seleção das comunidades virtuais</i>	163
3.1.1 Comunidades selecionadas.....	163
3.2 <i>Levantamento de dados</i>	166
3.3 <i>Análise dos dados</i>	167
3.3.1 <i>Análise de conteúdo e categorização</i>	167
3.3.2 <i>Análise etnográfica</i>	177
3.3.2.1 <i>Comunidade Linux (kernel)</i>	179
3.3.2.2 <i>Comunidade Apache</i>	182
3.3.2.3 <i>Comunidade MySQL</i>	184
3.3.2.4 <i>Comunidade PHP</i>	185
3.3.3 <i>Análise etnográfica das comunidades: sínteses</i>	186
3.3.4 <i>Análise da ação mediada</i>	187
3.4 <i>Análise da integração entre objetos de aprendizagem (OA) e as comunidades virtuais de aprendizagem auto-organizadas (CVAO)</i>	196
4 CONSIDERAÇÕES FINAIS	206

REFERÊNCIAS	217
--------------------------	------------

ANEXOS	231
<i>Anexo A – Home Page da Comunidade Linux – Kernel.....</i>	<i>231</i>
<i>Anexo B – Home Page da Comunidade Apache.....</i>	<i>233</i>
<i>Anexo C – Home Page da Comunidade MySQL.....</i>	<i>234</i>
<i>Anexo D – Home Page da Comunidade PHP.....</i>	<i>235</i>
<i>Anexo E – Home Page da Comunidade SourceForge.....</i>	<i>236</i>
<i>Anexo F - Discussões da Comunidade Linux.....</i>	<i>237</i>
Interação 1.8.LNX.....	237
Interação 2.30.LNX.....	244
<i>Anexo G – Discussões da Comunidade Apache.....</i>	<i>294</i>
Interação 1.7.APH.....	294
Interação 2.2.APH.....	301
Interação 3.3.APH.....	302
<i>Anexo H – Discussões da Comunidade MySQL.....</i>	<i>304</i>
Interação 1.4.SQL.....	304
Interação 2.4.SQL.....	305
<i>Anexo I – Discussões da Comunidade PHP.....</i>	<i>307</i>
Interação 1.9.PHP.....	307
Interação 2.12.PHP.....	313
<i>Anexo J – Discussões da Comunidade SourceForge.....</i>	<i>326</i>
Interação 1.4.SFG.....	326
Interação 2.2.SFG.....	327
Interação 3.2.SFG.....	327
Interação 4.1.SFG.....	327
Interação 5.1.SFG.....	327

INTRODUÇÃO

A sociedade contemporânea tem sido influenciada por significativas mudanças no campo social, econômico, cultural, político e antropológico, e nessa perspectiva são consideráveis seus efeitos no desenvolvimento dos sujeitos. Uma dessas mudanças ocorre de maneira especial na área da tecnologia, que não se restringe apenas aos novos usos de determinados equipamentos e produtos, mas afeta, significativamente, a maioria das atividades socioculturais da humanidade, entre as quais a ciência, a arte e o poder político (LÉVY, 1993).

A evolução social do homem confunde-se com as tecnologias desenvolvidas e empregadas em cada época, refletindo-se na economia, na política e na divisão social do trabalho. A globalização, por exemplo, eminentemente econômica, constitui-se como um conjunto de relações sociais que se traduzem na intensificação das interações transnacionais, sejam elas práticas interestatais, práticas capitalistas globais ou práticas sociais e culturais transnacionais (SOUSA SANTOS, 2001, p. 90). Nesse duplo processo, entre entidades ou fenômenos dominantes globalizados acontecem transformações externas e internas que afetam a humanidade.

Dessa forma, ferramentas tecnológicas (tudo o que utilizamos em nossa vida diária, pessoal e profissional) e técnica (forma de utilização de cada ferramenta para realizar determinada ação) constituem a *tecnologia* (CASTELLS, 2007; KENSKI, 2003; LÉVY, 1993). A *tecnologia*¹ é um agente de mudança que resulta em uma revolucionária mudança de paradigma, assentada no uso crescente de códigos, simulações, algoritmos, internet e programas que mediatizam a interação humana. A rede mundial de computadores, mais

¹ Tecnologia “é o estudo dos processos técnicos de um determinado ramo de produção industrial ou de mais ramos” (ABBAGNANO, N. Dicionário de Filosofia, 1982, p. 906. A técnica (...) “compreende todo conjunto de regras aptas a dirigir eficazmente uma atividade qualquer. (...) Seu campo estende-se tanto quanto o das atividades humanas” (idem, p.904).

conhecida como Internet, exerce significativa influência sobre a maneira como as pessoas aprendem, vivem e trabalham. Entretanto, Lévy (1993) alerta-nos que um dos riscos que se correm com o uso das tecnologias atuais é o de considerar os seres humanos simples sistemas de tratamento de informações, deixando de lado suas dimensões essencialmente humanas. O autor usa a expressão “*ecologia cognitiva*” para se referir às várias dimensões técnicas e coletivas da cognição (inteligência), visto que somos seres pensantes que se relacionam por meio de múltiplas linguagens, dentre as quais se destacam a escrita e a fala. Lévy (1994, p.24) argumenta:

(...) Novas maneiras de pensar e de conviver estão sendo elaboradas no mundo das telecomunicações e da informática. As relações entre os homens, o trabalho, a própria inteligência dependem, na verdade, da metamorfose incessante de dispositivos informacionais de todos os tipos. (...) O pensamento se dá em uma rede na qual os neurônios, módulos cognitivos humanos, instituições de ensino, línguas, sistemas de escrita, livros e computadores se interconectam, transformam e traduzem as representações. Não se pode mais conceber a pesquisa científica sem uma aparelhagem complexa que redistribui as antigas divisões entre experiência e teoria.

Emerge neste contexto o impacto das tecnologias de informação e das comunicações na construção/apropriação do conhecimento, nas formas de ensino e aprendizagem, nos novos tempos e espaços educativos. Segundo Kenski (2009, p.239) “o ensino mediado pelas tecnologias digitais pode alterar essas estruturas verticais (professor-aluno) e lineares de interação com as informações e com a construção individual e social do conhecimento”. Ambientes virtuais de ensino surgem onde se situavam as antigas formas “desgrudadas da geometria aprisionada de tempo, espaço e reações hierarquizadas de saber existentes nas estruturas escolares tradicionais” (KENSKI, p.239).

Esse cenário apontado pela autora corrobora com a ideia de que coexistem vários tipos de interação e comunicação, quando se usa a Internet para fins didáticos e de aprendizagem, que podem ser unidirecionais e refletirem um ensino baseado em trocas e desafios (KENSKI, 2009). Trabalhos cooperativos desenvolvidos nas redes incluem mecanismos de comunicação

que permitem às pessoas ver, ouvir e enviar mensagens uma às outras. Para Kerckhove (1999 *apud* KENSKI, 2009, p.242),

(...) a megaconvergência do hipertexto, multimídia, realidade virtual, redes neurais, agentes digitais e vida artificial está mudando modos de comunicação, entretenimentos e trabalho. (...) A Internet nos dá acesso a um entorno real (...) trata-se de uma nova condição cognitiva a que eu chamo de *webness* ou *inteligência em conexão*.

Fazendo um recorte para o nosso objeto de pesquisa, percebemos que os recursos educacionais estão sendo projetados, desenvolvidos e integrados, para serem disponibilizados e compartilhados em comunidades virtuais de aprendizagem e de trabalho colaborativo. Tal cenário tem motivado o desenvolvimento de pesquisas relacionadas à tecnologia aplicada à educação, o que pressupõe novas formas de utilização da tecnologia da informação e das comunicações como suporte efetivo ao processo de ensino e aprendizagem e do trabalho colaborativo, notadamente em ambientes virtuais. Na última década do século XX, a utilização desses recursos permitiu o acesso, a partir de qualquer lugar e a qualquer momento, a conteúdos educacionais, consolidando assim a aplicação dessas tecnologias aos processos educacionais.

A temática enfocada neste trabalho tem por base os meus estudos desenvolvidos na dissertação de mestrado “As Tecnologias Colaborativas em Ambiente Distribuído de Aprendizagem como um Recurso de Formação Continuada do Engenheiro de Produção” (RIVAS, 1998), no qual se apresenta uma proposta fundamentada em novos conceitos de ensino e aprendizagem e suportados por recursos tecnológicos, que foram divididos em três categorias:

1. tecnologias de distribuição;
2. tecnologias interativas;
3. tecnologias colaborativas.

Essas tecnologias categorizadas foram relacionadas com os respectivos modelos instrucionais e com os objetivos de aprendizagem, dando-se ênfase ao aprendizado colaborativo por meio dos fóruns de discussão, recurso este que tem sido o principal sustentáculo para que ocorra a aprendizagem nas comunidades virtuais que oferecem suporte ao trabalho colaborativo voltado para a solução de problemas.

A presente investigação constitui-se de um estudo empírico e de análise etnográfica das comunidades de aprendizagem virtual (HINE, 2000), aplicando o significado da intermediação com a ação pessoal como unidade de análise, principalmente a partir dos estudos de Cole (1997); Vigotsky (2007); Wertsch (1991); Wiley (2002); e análise documental (BARDIN, 2000).

Considerando-se a visão da existência dos objetos de aprendizagem como um artefato descontextualizado e independente da interação dos estudantes para com estes, o método analítico de Wertsch (1991) permite focar a pesquisa nas formas pela quais os recursos mediam a consecução efetiva da aprendizagem no ambiente virtual, dentro de sua fronteira histórica, cultural e contexto institucional. Nesse sentido, este trabalho investigativo é orientado para a verificação do uso dos objetos de aprendizagem no contexto de funcionamento das comunidades virtuais, especificamente, no que diz respeito ao processo de aprendizagem virtual para a solução de problemas.

Diante do exposto, problematiza-se a seguinte questão que servirá como elemento fulcral nesta pesquisa: *Como são utilizados os objetos de aprendizagem pelos membros das comunidades virtuais auto-organizadas para a solução de problemas relacionados à produção de software livre e de código aberto, tendo em vista a não existência de padrões de conformidade para a indexação, combinação, uso e avaliação dos objetos de aprendizagem?*

Objetivo

O objetivo geral desta pesquisa é produzir subsídios para a construção de um referencial teórico-metodológico sobre a utilização dos objetos de aprendizagem no contexto das comunidades virtuais auto-organizadas para a produção de software livre e de código aberto, que sirva de base para o desenvolvimento de um padrão de conformidade.

Definido o objeto geral, propõem-se as seguintes questões norteadoras:

1. Como as comunidades virtuais auto-organizadas são estruturadas e quais são seus mecanismos de funcionamento?
2. Como os objetos de aprendizagem são utilizados pelos membros das comunidades virtuais auto-organizadas?
3. Quais são os critérios de categorização dos problemas relacionados à produção de software livre e de código aberto, adotados pela comunidade virtual auto-organizada?
4. Existem padrões replicáveis evidentes dos objetos de aprendizagem selecionados para uso mais frequente que outros?

Contribuição

A contribuição desta pesquisa consubstancia-se na produção de subsídios para a construção de um referencial de análise, ou seja, num instrumental teórico-prático a ser utilizado no campo da tecnologia da informação e educação. Mais especificamente, contribui para a definição de um padrão de conformidade que utilize a especificação, indexação, uso, combinação e avaliação dos objetos de aprendizagem em comunidades virtuais auto-

organizadas, voltadas para a solução de problemas originados no trabalho colaborativo, aplicados à produção e ao uso de software livre e de código aberto – SLCA.

A pesquisa poderá também preencher um hiato existente, que é o de melhorar as condições de aprendizagem, por meio dos objetos de aprendizagem dispostos nos fóruns de discussão, que ainda são pouco elaborados, de difícil recuperação, indexação, e de qualidade variada.

O padrão de conformidade SCORM², já existente para o *design* e uso de objetos de aprendizagem foi definido para uso em ambientes instrucionais estruturados, e a sua concepção foi baseada em teorias de aprendizagem que não se coadunam com essa realidade.

Outras contribuições dizem respeito à oferta de referenciais, que poderão subsidiar o desenvolvimento de novos recursos relacionados à tecnologia da informação, aplicados ao ensino e aprendizagem em contextos virtuais.

Procedimentos metodológicos

Os procedimentos metodológicos adotados são os seguintes:

1. Efetuar a pesquisa bibliográfica com as últimas tendências investigativas acerca da produção de software livre e de código aberto, objetos de aprendizagem, teorias de aprendizagem e comunidades virtuais auto-organizadas;
2. Identificar e selecionar, no mínimo, quatro comunidades virtuais de usuários e desenvolvedores de software livre e de código aberto, que possuem outros projetos derivados em desenvolvimento na incubadora virtual *Sourceforge* e que já estão sendo largamente utilizados no mercado;

² SCORM – Sharable Content Object Reference Model (Modelo de Referência de Objeto de Conteúdo Compartilhável). É um padrão de conformidade para objetos de aprendizagem.

3. Identificar e descrever as principais características das comunidades virtuais, objeto de investigação;
4. Eleger um período de tempo e volume de dados relevantes referente às discussões contidas nos fóruns e listas de discussão das comunidades;
5. Efetuar estudo etnográfico das CVAO selecionadas, segundo o método de HINE (2000).
6. Demonstrar a equivalência do modelo de análise de diálogos com o modelo de análise de fóruns de discussão, segundo WERTSCH (1991).
7. Analisar e sistematizar os dados levantados (análise de conteúdo), segundo BARDIN (2000).
8. Consolidar resultados para a construção de um conhecimento que responda aos questionamentos deste trabalho de investigação;
9. Apresentar os resultados finais da pesquisa e com eles estabelecer a base do desenvolvimento de uma estrutura teórica que defina um padrão de conformidade para a utilização dos objetos de aprendizagem no contexto das comunidades virtuais auto-organizadas, visando à solução de problemas da produção de software livre e de código aberto.

Organização da tese

Esta tese está organizada da seguinte forma: na Introdução consta o objetivo, a problematização e contribuição da pesquisa. O Capítulo I versa sobre a revisão de literatura relacionada a Software livre e de código aberto, Objetos de Aprendizagem (OA), SCORM, Fóruns de Discussão, Teorias de Aprendizagem, Comunidades Virtuais e conceitos que tratam da mediação, dialogicidade, artefatos culturais e operações de pensamento. O Capítulo II

contempla a Metodologia consubstanciada na Etnografia Virtual, Teoria Analítica da ação mediada e análise de conteúdo. O Capítulo III aborda os Resultados e Discussão, por meio do levantamento, análise de dados e resultados. A síntese e os desdobramentos são apresentados nas Considerações Finais. Os Anexos abrangem todo o universo de dados coletados nos Fóruns de Discussão referentes às Comunidades Virtuais Auto-Organizadas (CVAO) selecionadas para a presente investigação.

Capítulo I

1 REVISÃO DA LITERATURA

Inicialmente, a proposta fundamental da revisão da literatura é (1) deixar transparente um contexto geral, identificar os principais elementos teóricos necessários para o desenvolvimento da pesquisa, bem como (2) proporcionar condições de integração entre esses elementos, obtendo desse modo uma condição básica de suporte e de sustentação teórica, no âmbito do pensamento de diversos autores sobre os temas abordados.

1.1 Software livre e de código aberto

O movimento do SLCA é mundialmente conhecido por meio da sigla em inglês – *FOSS – Free and Open Source Software*. Vale ressaltar que a estratégia utilizada pelo movimento do software livre é similar ao do software de código aberto, mas o seu significado é diferente. Para efeito desta pesquisa, serão considerados os dois movimentos conjuntamente, não se fazendo nenhuma distinção de significado ou de ideologia entre eles.

O movimento SLCA foi motivado por desenvolvedores de software (especialistas) que não estavam satisfeitos com a sistemática de proteção de software sob o regime corrente de *copyright*³, sistema ao qual o licenciamento para uso de software estava vinculado até então.

Este grupo vem crescendo rapidamente nos últimos anos, pela adesão de diferentes sujeitos ao movimento, e seu desempenho em diferentes papéis. Alguns deles são simples usuários finais de software, outros são desenvolvedores amadores e/ou profissionais, empresas ou mesmo agências governamentais, que proveem determinados projetos com fundos financeiros específicos.

³ *Copyright* é um sistema de proteção da obra autoral em si, seja literária, técnica, artística ou científica, dando ênfase à vertente econômica, à exploração patrimonial das obras através do direito de reprodução.

1.1.1 O movimento do software livre

O Software Livre é um movimento que pertence à *Free Software Foundation*, (STALLMAN, 1998). A filosofia básica do software livre é proporcionar o código das aplicações para distribuição, o que não implica a sua gratuidade nem a cessão dos direitos autorais. O software desenvolvido sob esta condição deve estar sujeito a uma das seguintes condições:

1. A licença de *copyleft*⁴ em oposição ao *copyright*. Nesta opção, o autor conserva todos os direitos de autoria do software criado, mas permite a modificação e a redistribuição sob certas condições, assegurando a adequação do desenvolvimento e a obrigatoriedade de que todas as aplicações derivadas sejam também livres;
2. A licença de domínio público, em que o autor prescinde de seus direitos e não exerce nenhum controle sobre o software por ele criado;
3. A licença *BSD*⁵, em que o autor mantém o direito autoral, bem como o direito sobre todas as aplicações derivadas, mesmo que não tenham sido desenvolvidas por ele. Esta forma de licença é mais conhecida por *GNU*⁶.

⁴ *Copyleft* é um trocadilho com o termo *copyright*. É uma forma oposta ao *copyright* de proteção aos direitos autorais de uma obra literária, artística, cultural ou científica. Este termo nasceu com o movimento de software livre e de código aberto, o qual foi popularizado por Richard Stallman. Disponível em: <<http://pt.wikipedia.org/wiki/copyleft>>. Acesso em: 15 jan. 2009.

⁵ *BSD* um tipo de licença que pertence ao grupo de licenças de software livre. A sigla *BSD* significa em inglês Berkeley Software Distribution. .

⁶ *GNU* foi um projeto iniciado por Richard Stallman em 1984, com o objetivo de criar um sistema operacional livre, que qualquer pessoa teria direito de usar, modificar e redistribuir, tanto o programa quanto o seu código-fonte, desde que garantido para todos os mesmos direitos. Este sistema operacional GNU deveria ser compatível com o sistema operacional UNIX, porém não deveria utilizar-se do código-fonte do UNIX. Stallman escolheu o nome **GNU** porque este nome, além do significado original do mamífero Gnu, é um acrônimo recursivo de: **GNU is Not Unix** (em português: GNU não é Unix). A partir de 1984, Stallman e vários programadores, que abraçaram a causa, vieram desenvolvendo as peças principais de um sistema operacional, como compilador de linguagem C, editores de texto, etc. Em 1991, o sistema operacional já estava quase pronto, mas faltava o principal, que era o seu núcleo, que significa em inglês *kernel*. Em 1991, o finlandês Linus Torvalds havia criado um *kernel* que poderia usar todas as peças do sistema operacional GNU. Este *kernel* ficou conhecido como Linux, contração de Linus e Unix. Atualmente, o sistema operacional GNU com o *kernel* Linux é conhecido como GNU/Linux, que é como o projeto solicita aos utilizadores que se refiram ao sistema completo, embora a maioria das pessoas se refere ao sistema apenas como Linux por uma questão de comodidade. Disponível em: <http://pt.wikipedia.org/wiki/gnu>. Acesso em 15 jan. 2009.

Como exemplo de softwares livres, licenciado sob esta filosofia, pode ser citado: Linux⁷, PHP⁸, Python⁹, Open Office¹⁰, Mozilla¹¹, SendMail¹², Apache¹³, MySQL¹⁴, além de muitos outros.

Segundo Stallman (1998), um software pode ser considerado livre, se:

1. Tiver a liberdade de ser executado para qualquer propósito. Esta liberdade é chamada de “liberdade 0”;
2. Tiver a liberdade para estudar, analisar o software e modificá-lo. Esta liberdade é chamada de “liberdade 1”;
3. Tiver a liberdade de copiar o software, de maneira que seja útil para qualquer pessoa. Esta liberdade é chamada de “liberdade 2”;
4. Tiver a liberdade de aprimorar, bem como tornar públicas as melhorias e inovações, de forma a beneficiar toda a comunidade que utiliza ou virá a utilizar o software. Esta liberdade é chamada de “liberdade 3”.

1.1.2 O movimento do software de código aberto

Software de código aberto – *open source* – é um movimento que foi criado por Raymond e Perens (1998). O objetivo principal do movimento não foi, necessariamente, promover o benefício da distribuição do software gratuitamente, mas sim a busca de questões de natureza prática e técnica no processo de desenvolvimento de software, com ênfase na procura dos

⁷ Disponível em: <http://www.kernel.org/>. Acesso em: 21 jan. 2008.

⁸ Disponível em: <http://www.php.org/>. Acesso em: 21 jan. 2008.

⁹ Disponível em: <http://www.phyton.org/>. Acesso em: 21 jan. 2008.

¹⁰ Disponível em: <http://www.openoffice.org/>. Acesso em: 21 jan. 2008.

¹¹ Disponível em: <http://www.mozilla.org/>. Acesso em: 21 jan. 2008.

¹² Disponível em: <http://www.sendmail.org/>. Acesso em: 21 jan. 2008.

¹³ Disponível em: <http://www.apache.org/>. Acesso em: 21 jan. 2008.

¹⁴ Disponível em: <http://www.mysql.com/>. Acesso em: 21 jan. 2007.

benefícios do trabalho colaborativo. Os softwares de código aberto devem seguir preceitos e regras pré-estabelecidas. Cabe, no entanto, destacar que nem todos os softwares de código aberto, desenvolvidos e liberados para distribuição, permitem a sua modificação ou adaptação, além do que devem estar sob a licença expedida pela OSI – *Open Source Initiative*¹⁵.

Para que um software desenvolvido possa ser realmente considerado *aberto*, há que ser ratificado de acordo com os pré-requisitos do processo de licenciamento, conforme segue (OSI, 1998):

1. Livre distribuição:

Em nenhuma hipótese, para o processo de licenciamento, deve-se restringir a sua venda ou a sua entrega, juntamente ou como componente de algum outro software. O processo de licenciamento não pode requerer nenhum tipo de pagamento de *royalties*¹⁶ ou obrigações de cotas de venda.

2. Código-Fonte:

O pacote de software deve incluir o *código-fonte*¹⁷, e deve permitir a sua distribuição tanto com o código-fonte quanto com o *código compilado*¹⁸. Nos casos em que o código-fonte não for entregue juntamente com o software, este deve ser disponibilizado em um local de acesso conhecido e sem custo algum para consegui-lo. O meio mais usual é a sua disponibilização em um diretório de arquivos acessível na Internet, mediante o processo de *downloading*. Vale ressaltar que o código não deve

¹⁵ Disponível em: <http://www.opensource.org/>. Acesso em: 20 jan. 2008.

¹⁶ *Royalties* é uma forma de cobrança em valor monetário do direito de uso de uma marca ou produção de um bem. Usualmente, define-se um percentual sobre um determinado valor fixo ou variável.

¹⁷ Código-fonte é a versão do programa em linguagem de computador não compilado. Quando o código está nessa condição, existe a possibilidade de alteração do programa por outros programadores. Basicamente, é o conjunto de palavras escritas de forma ordenada, com instruções de uma linguagem de programação específica. Existem linguagens que são compiladas e as que são interpretadas. As linguagens compiladas, após ser compilado o código-fonte, transforma-se em software, ou seja, em programas executáveis.

¹⁸ Código compilado é resultado da compilação do código-fonte. Quando o código está nesta condição, não existe a possibilidade de alteração do programa.

ser confuso, para não dificultar ou confundir os futuros programadores responsáveis pela modificação dos programas.

3. Desenvolvimento derivados:

O processo de licenciamento deve permitir que modificações possam ser efetuadas no código-fonte e assim derivar novos módulos, subsistemas ou até mesmo novos softwares derivados, mas desde que estes sejam distribuídos sob as mesmas condições da licença do software original.

4. Integridade do código-fonte do autor:

A licença somente pode restringir a distribuição de código-fonte modificado, se for distribuído juntamente com *patch files*¹⁹, código-fonte original, com o propósito de modificar o software durante o processo de desenvolvimento. A licença pode exigir que todos os softwares derivados, ou que sofreram modificações, sejam identificados por um número de versão que seja distinto do número de versão do software original.

5. Não identificar pessoas ou grupos:

No processo de licenciamento do software, não pode haver qualquer citação do nome de pessoas, equipe ou grupo de trabalho que tenham participado do processo de desenvolvimento.

6. Não mencionar campos ou áreas de aplicação:

O processo de licenciamento do software não pode restringir o seu uso à determinada área específica de aplicação, como, por exemplo, delimitar a utilização em certas áreas específicas de negócio, de pesquisa, ou em outras áreas de aplicação.

¹⁹ *Patch files* é um termo técnico em inglês da área de computação que denomina um pequeno programa de computador cuja função é alterar um software já instalado, para corrigir um erro ou falha de execução.

7. Distribuição da licença:

O direito de licença deve ser aplicado a todas as pessoas a quem o software foi concedido, sem a necessidade de obter qualquer licença adicional.

8. A licença não deve ser específica de um produto:

O direito de licença estende-se ao software como um todo, ou seja, tem que permanecer igual ao das partes, ou dos módulos, se estas forem distribuídas separadamente ou se tornarem parte de outras distribuições. A licença das partes tem que ser a mesma do programa original do qual foi originada.

9. A licença não deve restringir o uso de outros softwares:

A licença não pode restringir o uso de outros softwares na mesma mídia de distribuição, ou exigir que os outros softwares também sejam livres ou de código aberto.

10. A licença deve ser tecnologicamente neutra:

A licença não pode privilegiar nenhuma tecnologia em específico ou estilo de interface.

1.1.3 Principais produtos de software livre e de código aberto

Os principais e mais conhecidos projetos de SLCA incluem iniciativas em várias categorias de aplicação tais como: sistemas operacionais, sistemas gerenciadores de banco de dados, linguagens de programação, gerenciadores de servidores *Web*, servidores de impressão, ambientes gráficos, etc.

Logo abaixo, apresenta-se um quadro descritivo de algumas iniciativas com uma abordagem sucinta das funcionalidades do software, bem como o endereço de acesso, na

Internet, da *Web Site* institucional e respectiva comunidade virtual para acesso dos usuários e desenvolvedores (TAURION, 2004).

Quadro 1 - Relação de softwares livre e de código aberto

<i>Software</i>	<i>Descrição</i>	<i>Web Site</i>
PostgreSQL	Banco de dados relacional e orientado a objetos, derivado dos projetos Ingres, desenvolvido no período de 1977 a 1985, e Postgres (1986-1994), ambos da Universidade da Califórnia em Berkeley.	www.postgresql.org
Firedbird	O projeto iniciado em julho de 2000 baseia-se no código-fonte da Interbase, liberado pela Borland.	www.sourceforge.net/projects/firebird
MySQL	Gerenciador de Banco de Dados Relacional criado em 1994, por meio de uma empresa sueca chamada MySQL AB. Este banco de dados é disponibilizado na modalidade GPL e também sob o licenciamento proprietário.	www.mysql.com
Apache	O desenvolvimento deste software começou em 1994 no NCSA ²⁰ . É um software de gerenciamento de servidores Web.	www.apache.org
OpenOffice	É uma alternativa de software livre para o Microsoft Office. ²¹	www.openoffice.org
Perl	É uma linguagem de programação interpretada desenvolvida em 1987. É muito utilizada para aplicações da Internet, principalmente na construção de <i>scripts</i> para o servidor Apache.	www.perl.org
Python	Linguagem de programação interpretada que é comparada com o Perl. Existe uma versão integrada com Java (Jpython) que permite executá-la em computador equipado com máquina virtual Java.	www.python.org
Mozilla	Projeto de browser iniciado pela Netscape, como resposta à ação da Microsoft em distribuir gratuitamente o <i>browser Internet</i>	www.mozilla.org

²⁰ NCSA – National Center for Supercomputer Applications. Disponível em: <http://www.ncsa.uiuc.edu/>. Acesso em: 20 jan. 2008.

²¹ Microsoft Office – Disponível em: <http://office.microsoft.com/pt-br/default.aspx>. Acesso em: 20 jan. 2008.

<i>Software</i>	<i>Descrição</i>	<i>Web Site</i>
	<i>Explorer.</i>	
Gnome e KDE	Conjunto de ferramentas voltadas para o ambiente de interface gráfica em desktop para Linux.	www.gnome.org e www.kde.org
Php	PHP é uma linguagem de programação interpretada, projetada originalmente para criação de páginas dinâmicas da web. PHP é um acrônimo que significa <i>Hypertext Pre-processor</i> . Esta linguagem foi criada por Rasmus Lerdorf em 1994. A continuidade do desenvolvimento da linguagem foi efetuada pela PHP Group, sendo distribuída como software livre.	www.php.net
Sendmail	O projeto foi iniciado no final da década de 1970, com uma versão mais restrita, denominada <i>Delivermail</i> e, em 1981, surgiu uma versão mais abrangente, que passou a se chamar <i>Sendmail</i> .	www.sendmail.org
FreeBSD	É uma versão aberta do Unix, que começou a ser desenvolvida em 1993 pela Universidade de Berkeley, na Califórnia.	www.freebsd.org
Compiere	É um software tipo ERP (enterprise resourcing planning), escrito em Java, e voltado para pequenas empresas. É licenciado pelas regras do <i>Mozilla Open Source Licence</i> . O Projeto foi iniciado na Goodyear da Alemanha, que patrocinou o projeto, para uso próprio. Seu autor, o alemão Jorg Janke, tornou o produto mundialmente conhecido quando abriu o código-fonte na Internet.	www.compiere.org
GNUE	É um software de gestão empresarial (ERP). Atualmente está ainda em estágio de desenvolvimento.	www.gnue.org

1.1.4 O processo de criação e desenvolvimento

De maneira geral, os projetos de software livre e de código aberto são desenvolvidos por membros que trabalham colaborativamente, por meio de comunidades. Todo processo de comunicação entre os membros destas comunidades é efetuado por meio da Internet, com a utilização de diversos recursos, tais como: e-mails, listas de discussão, fóruns de discussão, chats, blogs, etc.

O uso do telefone é mínimo, principalmente pelo fato de os desenvolvedores não se conhecerem e estarem em diferentes países. Por outro lado, existe a preocupação com custos, em evitar custos diretos ou variáveis, tendo em vista que essa atividade é voluntária e não remunerada.

Na maioria das vezes, o projeto é organizado em torno de um desenvolvedor responsável, que geralmente é o idealizador do software. O idealizador é chamado de mantenedor, enquanto os demais participantes do projeto são chamados de contribuidores.

O ciclo de vida de um software livre e de código aberto começa quando o autor programa a versão inicial e publica na Internet o seu código-fonte. Se o software despertar interesse, outros desenvolvedores começam a experimentá-lo e a corrigir-lhe os erros, propondo modificações e melhorias. Tais propostas são enviadas ao autor, que as analisa, avalia, e, caso forem válidas, incorpora as melhores ao software, o que gera uma nova versão que é novamente publicada. Esse ciclo se repete continuamente, e, de acordo com a popularidade do software, o projeto vai atraindo mais voluntários desenvolvedores. Se o projeto não se torna popular entre a comunidade, há grande chance de o projeto fracassar e desaparecer.

De acordo com Taurion (2004), podem-se propor, resumidamente, as seguintes atividades principais desenvolvidas pelos membros da comunidade:

1. O mantenedor libera a primeira versão do software e o respectivo código-fonte. Divulga o fato nos fóruns e/ou listas de discussão. A partir desta iniciativa, começam a surgir os primeiros interessados;
2. Os contribuidores voluntários acessam a *Web Site* da comunidade incubadora do projeto, fazem o *download* do software e de seu código-fonte e começam a colaborar;
3. Experimentam e testam o software, identificam erros e propõem correções devidas;
4. Na maioria das vezes, escrevem os próprios códigos, programando as correções e modificações, e as enviam para o Mantenedor;
5. As correções e modificações são discutidas pelos membros da comunidade nos fóruns de discussão. O Mantenedor analisa e avalia as propostas e elege as melhores, o que também é discutido com os membros da comunidade. Nesse momento, há um processo intenso de colaboração no sentido de melhorar em conjunto tudo o que foi proposto e de criar códigos de qualidade. Em alguns momentos, pode até surgir algum tipo de competição, mas o Mantenedor deve dissipar esse comportamento e fazer prevalecer sempre o espírito de colaboração;
6. Após exaustiva discussão, as melhores propostas são incorporadas ao software pelo mantenedor;
7. O Mantenedor libera nova versão à comunidade e o ciclo se repete.

Na realidade, o processo de desenvolvimento é efetuado em rede, por uma comunidade de desenvolvedores voluntários, sem vínculo entre si, em uma organização informal. A comunicação é efetuada por meio da Internet, virtualmente sem fronteiras

geográficas, e apenas alguns princípios regulam o trabalho. A liderança do projeto pode ser definida de maneira prévia e formal, mas pode também emergir naturalmente pelos méritos de determinado membro da comunidade de voluntários.

Os códigos são revisados pelos próprios pares e geralmente o melhor código é selecionado. O ritmo de desenvolvimento é direcionado pela disponibilidade de tempo e de dedicação dos desenvolvedores voluntários. Esse modelo de desenvolvimento cria uma forte tendência para gerar código de alta qualidade. O código é lido e analisado por diversos membros, às vezes por centenas de desenvolvedores, o que acelera o processo de depuração e correção de erros. Simplesmente pelo fato de não haver pressão por prazos, é possível ocorrer mais ciclos de depuração. A decisão de liberação do código é resultado de consenso do grupo e não da imposição de alguém ou de alguma área, como ocorre em projetos de caráter comercial.

O líder principal do projeto é quem se encarrega de coordenar o processo de liberação de uma versão do software. Os ciclos de testes e revisões são constantes e o processo de realimentação é contínuo. Todos, a qualquer momento, podem interceder e comentar sobre o código, uma vez que este é livre e está disponível para todos.

Um dos fatores críticos de sucesso do SLCA é a frequência das primeiras versões. Caso as primeiras versões apresentarem falhas e erros, ou se o software apresenta o potencial necessário para gerar interesse na comunidade, as correções e modificações serão postadas muito rapidamente. Diante do exposto, o processo de liderança e coordenação, bem como a rápida inserção dessas correções são essenciais para garantir que o interesse se mantenha ativo, estimulando os voluntários, principalmente nas fases iniciais do ciclo de vida de desenvolvimento do software.

Os projetos de SLCA são tipicamente caracterizados por um grau pequeno de coordenação. Os desenvolvedores são completamente livres para introduzir modificações ou

correções que desejam. Cabe ao líder do projeto selecionar e integrar as colaborações que combinam qualidade e funcionalidade. Esse papel é de extrema importância. O líder do projeto deve conquistar a confiança e admiração dos membros da comunidade, para garantir que suas decisões sejam acatadas e que não surjam versões alternativas do software impulsionadas pelos descontentes com aquelas determinações.

Os princípios de desenvolvimento de SLCA, por meio do trabalho colaborativo, executado por voluntários, ainda não são totalmente conhecidos, além de outros fenômenos derivados.

Um dos fenômenos de extrema relevância é como os membros das comunidades realmente aprendem, solucionam problemas, negociam e resolvem conflitos. São aspectos de natureza sócio-comportamental que precisam ser observados, analisados e estudados, dentro do rigor científico, para o esboço dos modelos teóricos iniciais sobre o assunto.

1.1.5 Um novo modelo de negócio

O movimento SLCA é uma realidade indiscutível. O *Linux*²² é um sistema operacional cuja base instalada tem aumentado constantemente. O *Apache*²³ é líder na categoria de servidores *Web*. A linguagem de programação *PHP*²⁴, bem como o *Sendmail*²⁵, o gerenciador de banco de dados *MySQL*²⁶, e dezenas de outros softwares já fazem parte do portfólio de recursos da maioria das empresas do mundo. Grandes corporações e governos estão utilizando o sistema operacional *Linux* extensivamente, e, a partir de um olhar mais global, a maioria das empresas é usuária do Sistema Operacional *Linux*. Essas empresas estão adaptando o *Linux* às suas necessidades, usando-o em seus próprios servidores ou em

²² *Linux* é um sistema operacional.

²³ *Apache* é um software de gerenciamento de hospedagem de sites para a web.

²⁴ *PHP* é uma linguagem de programação.

²⁵ *Sendmail* é um recurso do *PHP* para receber e enviar e-mail.

²⁶ *MySQL* é um gerenciador de banco de dados relacional.

servidores de *data centers*²⁷, e estão atentas ao uso de tecnologias de código aberto e como essas tecnologias podem se encaixar na sua estratégia de tecnologia da informação, como suporte aos seus processos de negócios.

Do ponto de vista corporativo, as grandes empresas estão em constante preocupação para se manter à frente, no que se refere a apreender uma determinada tecnologia e usá-la como recurso para se destacar no mercado de sua área de atuação. Além disso, essas empresas apoiam fortemente o uso de tecnologias que estimulem o uso de padrões abertos e que tragam maior agilidade aos seus processos. Também executam em seus computadores uma série de sistemas de informação para gerir e fornecer suporte aos seus negócios, que precisam agora ser operacionalizados cada vez mais com segurança nos diversos níveis. Essas organizações demandam um sistema operacional que possa servir de base para todos esses sistemas de informação, providos de recursos como virtualização, segurança em camadas de aplicação, etc. Assim, a indústria em geral, especialmente as de maior porte, tem considerado os sistemas baseados em código aberto e padrões abertos como um meio de acelerar a inovação dentro de suas próprias organizações, mas também como uma maneira de ter mais escolha e maior flexibilidade na sua infraestrutura de tecnologia da informação. Diante de tal realidade, a migração para os sistemas de código aberto evidencia ser uma tendência irreversível.

Logo abaixo, a demonstração gráfica dos critérios que as empresas têm adotado para a utilização de produtos e serviços baseados no *Linux*.

²⁷ *Data Center* é um tipo de negócio em que a empresa presta serviços de infraestrutura de tecnologia da informação e comunicação para outras empresas. Os serviços típicos de um *data center* são o armazenamento de dados, hospedagem de web sites, serviço de correio eletrônico, etc.

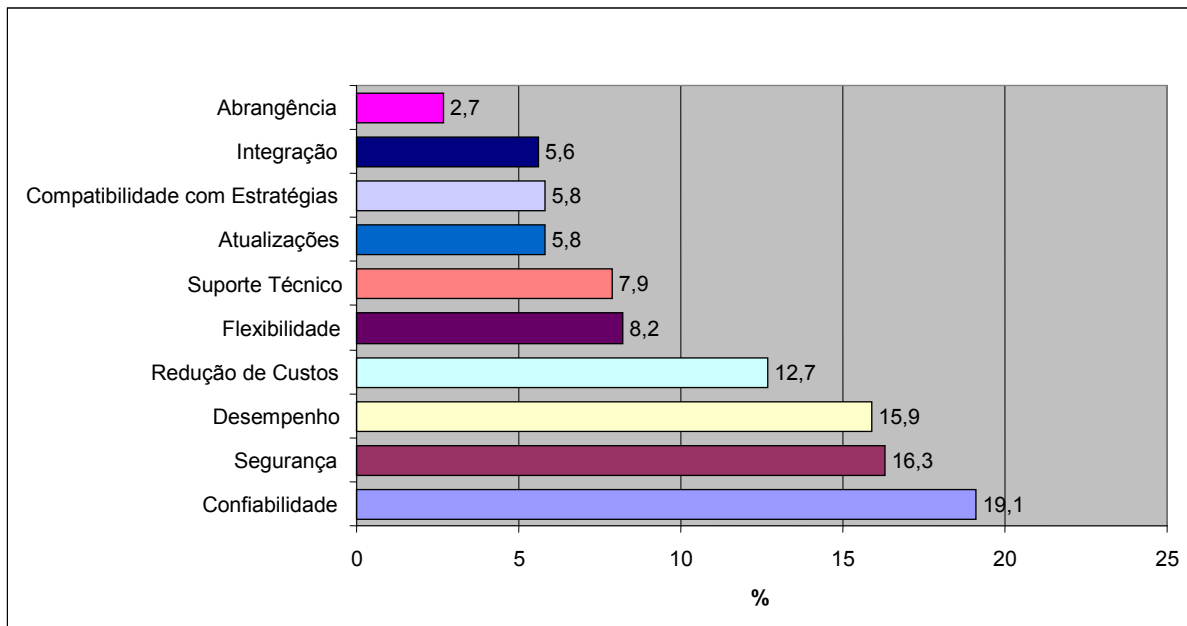


Figura 1- Critérios para adoção de produtos e serviços Linux

Fonte: Revista Linux Magazine

A economia gerada pelo uso de sistemas de código aberto não é mais o principal motivo para a sua adoção. Confiabilidade, segurança e desempenho foram a motivação de mais de 50% dos 500 executivos de TI entrevistados em 2005 pela Chadwick Martin Bailey, segundo a Revista Linux Magazine.

A incubadora virtual de projetos de desenvolvimento de SLCA, como o *Sourceforge*, já armazena mais de 160 mil projetos, com mais de 1,6 milhão de usuários registrados, que fazem parte da comunidade²⁸.

A essência da filosofia de desenvolvimento de software livre e de código aberto não é o software em si (o produto), mas o seu processo de desenvolvimento, que é colaborativo e em que os usuários colaboram entre si para criar determinado software. Nesse processo, não existe a necessidade de altos investimentos prévios. No modelo tradicional, o desenvolvedor, na maioria das vezes, precisa vender as licenças do seu software para recuperar o investimento e a obtenção de lucros financeiros.

²⁸ Disponível em: <http://www.sourceforge.net/>. Acesso em: 20 jan. 2008.

No modelo de desenvolvimento em forma de trabalho colaborativo, tanto o esforço quanto o custo de desenvolver o software são compartilhados. Dessa forma, não existem maiores investimentos prévios, e conseqüentemente não existe a necessidade da venda de produtos, ou seja, as licenças. Pode-se auferir receita a partir da venda de serviços. O *Kernel* 2.6²⁹ tem mais de 3,2 milhões de linhas de código em *linguagem C*³⁰ e quase 180.000 linhas de código em linguagem *assembler*³¹. Se esse sistema operacional fosse desenvolvido por uma única empresa, tal volume de trabalho custaria centenas de milhões de dólares, além da necessidade de contar com uma escala de mercado muito ampla e de retorno rápido, o suficiente para se manter viável. O modelo de desenvolvimento de software de código aberto dilui os custos por meio do trabalho colaborativo de dezenas de milhares de colaboradores, na sua maioria voluntária.

Os custos fixos de pesquisa e desenvolvimento não são alocados a nenhum centro de custo ou atividade. Este modelo permite a sustentabilidade do produto mesmo quando a escala é pequena. A sua sobrevivência é mais viável que a de um software desenvolvido sob o modelo tradicional com vistas à comercialização de suas licenças.

Existem diferenças significativas entre modelos de negócio baseados na venda de licenças e na venda de serviços. O modelo de negócio baseado em venda de licenças consegue obter resultados financeiros muito mais elevados, principalmente quando se considera software de venda em massa. Estes tipos de softwares são desenvolvidos uma única vez e podem ter as suas licenças vendidas dezenas de milhares de vezes a custos marginais (como o preço da mídia que o embala, tipo CD), ou a custos próximos a zero, quando estes são distribuídos, via *download*³², com pequenos investimentos adicionais em suporte e consultoria. Por sua vez, os modelos de negócios baseados em serviços demandam mais

²⁹ Kernel é o núcleo básico do sistema operacional.

³⁰ *Linguagem C* é um tipo de linguagem de programação que foi desenvolvido por Stallman.

³¹ *Assembler* é uma linguagem de programação de baixo nível.

³² *Download* é uma forma de descarga eletrônica, de software, documento ou outro tipo de objeto em formato digital.

recursos humanos e a lucratividade é significativamente menor e obtida em maior prazo. A demanda por consultores e técnicos especializados aumenta à medida que a base instalada é ampliada.

Os impactos causados pelo modelo de negócio baseado no desenvolvimento de software de código aberto começam a causar sérios transtornos e preocupações para as empresas que são extremamente dependentes de receitas financeiras advindas da venda de licenças, pois as empresas de software de código aberto, que possuem ações na bolsa de valores, são continuamente e constantemente monitoradas pelos investidores. Tais empresas são forçadas a melhorar o seu desempenho, por meio de um crescimento mais rápido do que as expectativas dos seus acionistas. Crescer no mesmo ritmo do mercado apenas evita a queda no preço das ações, mas não gera ganhos. Contudo, esse novo cenário torna o crescimento dessas empresas mais desafiador, gerando reações agressivas de mercado, na tentativa de evitar uma eventual percepção negativa por parte dos investidores.

A consequência desse novo paradigma de desenvolvimento de software sob a forma de trabalho colaborativo caracteriza-se como um fenômeno que vem sendo objeto de muitos estudos e pesquisas, pois há muito a ser investigado em todas as suas dimensões.

Usualmente, formam-se as Comunidades Virtuais de Aprendizagem Auto-Organizadas - CVAO a partir do momento em que é lançada uma *versão beta*³³ do software no mercado, que até então continuava sob o abrigo da incubadora virtual *Sourceforge*. Essa comunidade passa a conter diversas informações sobre o software e torna-se um ambiente de troca de informações e de conhecimento entre os seus usuários. O compartilhamento do conhecimento é feito por meio dos grupos de discussão, em que um usuário efetua a postagem do seu problema e outros usuários postam uma forma de solução. Normalmente, vários usuários apresentam diversas formas de solução para o problema apresentado.

³³ Versão beta é uma versão teste do software.

As perguntas mais frequentes (popularmente conhecidas como *FAQ*³⁴) são eleitas para comporem uma seção da comunidade. As perguntas são colocadas em ordem alfabética com as respectivas respostas, as quais vão desde procedimentos descritivos bem detalhados até cópias completas de tutoriais, informando passo a passo o que deve ser feito para a solução do problema. Diversos *Web links*³⁵ de outras comunidades relacionadas também são utilizados.

O principal problema dessas comunidades é que, por serem auto-organizadas e sem um padrão previamente definido, a organização, disposição e acesso ao conhecimento não são estruturado nem facilmente acessíveis. A forma mais prática de se obter uma solução para o problema encontrado é postar uma pergunta correspondente ao grupo de discussão. Na maioria das vezes, essa pergunta já foi anteriormente efetuada, mas pela dificuldade de recuperação de discussões, já arquivadas, a forma mais prática e rápida é efetuar-la novamente. O que ocorre, usualmente, é que aqueles usuários mais experientes, simplesmente “copiam e colam” a resposta novamente, já respondida em outras ocasiões e a reutilizam, para o usuário que está efetuando a pergunta, naquele momento. Todo esse conjunto de perguntas e respostas pode ser chamado de *objetos de aprendizagem*, que estão sendo reutilizados.

1.1.6 A necessidade de um padrão de conformidade

Os membros e usuários dessas comunidades virtuais, que suportam o trabalho colaborativo, bem como a aprendizagem necessária para a solução de problemas, têm utilizado os objetos de aprendizagem como um recurso desses ambientes, apesar de estes usuários não terem a consciência de que o resultado de suas perguntas e respostas postadas nos fóruns pode ser, tecnicamente, denominado de objetos de aprendizagem. Tais recursos não possuem entre si nenhum tipo de padrão de conformidade quanto à *indexação*,

³⁴ FAQ, sigla derivada do idioma inglês, que significa *frequent answers and questions*.

³⁵ Web link é recurso de acesso, na forma de hipertexto, que é muito utilizado nas páginas formatadas para Web.

combinação, uso e avaliação. Uma situação assim compromete a qualidade e, sobretudo, a efetividade dos ambientes para com os seus usuários, pois ainda existe um descompasso entre o que os objetos de aprendizagem podem oferecer e os novos conceitos teóricos de aprendizagem.

1.2 Objetos de aprendizagem (OA)

Há uma variedade de denominações destes recursos educacionais, tais como:

1. Componentes de software educacional;
2. Objetos de conteúdo compartilhável (ADL, 2001);
3. Objetos de conhecimento (MERRIL, 2001);
4. Objetos educacionais (SPHORER, 2001);
5. Objetos de aprendizagem (LTSC/IEEE, 2000).

Tais recursos educacionais podem ser chamados, genericamente, de OA, de acordo com a terminologia adotada pelo *Learning Technology Standards Committee* (LTSC) of the *Institute of Electrical and Electronics Engineers* (IEEE), que é a seguinte:

Objetos de aprendizagem são definidos como uma entidade, digital ou não digital, que pode ser usada e reutilizada ou referenciada durante um processo de suporte tecnológico ao ensino e aprendizagem. Exemplo de tecnologia de suporte ao processo de ensino e aprendizagem inclui sistemas de treinamento baseados em computador, ambientes de aprendizagem interativa, sistemas instrucionais assistidos por computadores inteligentes, sistemas de educação a distância e ambientes de aprendizagem colaborativa. Exemplos de Objetos de Aprendizagem incluem conteúdos de aplicações multimídia, conteúdos instrucionais, objetivos de aprendizagem, ferramentas de software e software instrucional, pessoas, organizações ou eventos referenciados durante o processo de suporte da tecnologia ao ensino e aprendizagem (LOM, 2000, p.23).

Wiley (2000) propôs uma definição mais específica, visto que, segundo ele, o descrito acima seria basicamente todo o conjunto de coisas existentes no universo, não sendo assim

uma definição devidamente apropriada para o contexto educacional. Para Wiley (2000, p.7) “objeto de aprendizagem é qualquer recurso digital que pode ser reutilizado para dar suporte a um processo de aprendizagem”.

Entretanto, existe uma necessidade premente de novas estruturas teóricas para o desenvolvimento e uso desses recursos. Há um consenso, entre os pesquisadores da área, de que os enfoques atuais para a utilização de objetos de aprendizagem são descontextualizados e isolados, tendo falhado, sobremaneira, no sentido de proporcionar um efetivo resultado educacional como fruto da aplicação de tais objetos de aprendizagem, isto é, a busca de um nível elevado de qualidade dos ambientes de aprendizagem que suportam tais processos em sua totalidade (WILEY, 2000).

Enquanto os pesquisadores em tecnologia educacional, educação e psicologia educacional têm-se engajado no desenvolvimento de estratégias de ensino e aprendizagem considerando esse novo contexto, poucos métodos existem para a criação de uma teoria de *design* e uso de objetos de aprendizagem que criem uma composição concreta, para se conseguir resultados de qualidade e se obter assim um conjunto de objetivos de aprendizagem necessários para a formação de uma competência. De acordo com Rivas (1998, p.17) esse conjunto de objetivos compreende:

1. Transferência de conhecimento;
2. Desenvolvimento de novas habilidades;
3. Formação de novas atitudes.

Para a consecução de cada um desses objetivos, seria necessário um conjunto de objetos de aprendizagem dispostos, contextualizados e devidamente formatados em uma sequência específica e suportados por diferentes ferramentas tecnológicas, mas encapsulados em um ambiente integrador que pudesse realmente fazer com que os estudantes aprendessem, ou seja,

tivessem a competência necessária para compreender conceitos, desenvolver projetos, solucionar problemas e saber comportar-se diante de situações específicas desafiadoras e conflituosas. Basicamente, existe a necessidade de formulação de uma teoria que possa orientar todo esse processo de *design* e uso articulado dos objetos de aprendizagem em novos contextos.

Há uma diversidade de nomenclaturas para referenciar objetos de aprendizagem, bem como sua utilização. Para melhor ilustração e compreensão do que está sendo citado, serão apresentadas duas iniciativas, sendo uma do setor privado e outra do segmento militar, no sentido de fundamentar teoricamente o problema em questão.

Devido à demanda interna de treinamento e desenvolvimento de pessoal, a Cisco Systems, Inc.³⁶, empresa americana que atua no desenvolvimento de tecnologia e fornecimento de produtos voltados para redes de comunicação de computadores desenvolveu, por meio de seus pesquisadores, estudos no campo do design instrucional e aplicação de objetos de aprendizagem. Foi desenvolvido a *Cisco's Reusable Learning Objects Strategy – RLO* (BARRIT; LEWIS, 2001). O processo *RLO* começa por analisar a tarefa a partir das partes que a constituem, e os objetivos de aprendizagem são desenvolvidos pelas partes identificadas destas tarefas. Foi também desenvolvido um novo conceito relacionado à informação denominado de *Reusable Information Objects – RIO*, produzidos por necessidade de ensinar um objetivo bem específico. A prática dessas estratégias e as avaliações são desenvolvidas no sentido de mensurar o nível de domínio do estudante sobre o assunto. Essa metodologia forneceu significativos benefícios para os desenvolvedores de cursos em ambiente virtual, permitindo então a reutilização dos recursos educacionais, e assemelhando-se muito à forma como a engenharia de software reutiliza os objetos no paradigma de programação orientada a objetos.

O método desenvolvido pela Cisco System, Inc. e outros métodos conhecidos são surpreendentemente similares, tendo em vista que são baseados nas mesmas referências

³⁶ Disponível em: <http://www.cisco.com/>. Acesso em: 20 jan. 2008.

teóricas das décadas de 1970 e 1980, incluindo Bloom (1956); Clark (1989); Mager (1975) e Merrill (1983), bem como outros autores que desenvolveram seus trabalhos sob o paradigma instrucional behaviorista.

Para citar outra iniciativa, o Departamento de Defesa dos Estados Unidos da América, por intermédio da *Advanced Distributed Learning Network* desenvolveu o *Sharable Contents Objects Reference Model* (SCORM; ADL, 2001, p.17-18), que efetuou a seguinte declaração:

Estudos empíricos têm levantado o interesse nacional no emprego de tecnologias de educação e treinamento que são baseadas no aumento de poder, acessibilidade e disponibilidade das tecnologias de computação e redes de comunicação. Estes estudos sugerem que se realize a promessa de melhoria da eficiência da aprendizagem através do uso de tecnologias instrucionais – assim como a instrução baseada em computador, instrução baseada na multimídia interativa e sistemas tutores inteligentes - para uma melhor adequação a cada estilo de estudante, seus objetivos e necessidades – dependendo da habilidade destas tecnologias em costurar as necessidades instrucionais dos indivíduos. Em contraste com a aprendizagem em sala de aula, estes enfoques permitem o ritmo, a sequência, e método de instrução para melhorar e adequar ao estilo de aprendizagem, objetivos e desejos (...) o dilema apresentado pela instrução personalizada para o atendimento individual é inviável economicamente. Com poucas exceções, um instrutor para cada estudante, apesar das vantagens, não é possível. Por outro lado a tecnologia instrucional promete fornecer muitas vantagens ao encontro das instruções individualizadas com custos acessíveis, e mensuráveis, além de uma alta qualidade de seus conteúdos.

Quatro suposições podem ser extraídas da declaração citada, que são:

1. O emprego de tecnologias instrucionais e sistemas tutores inteligentes auxiliam o sujeito em seu processo de aprendizagem;
2. O modelo instrucional individual é preferido em relação a outros modelos;
3. A interação humana em ambientes de aprendizagem de larga escala é economicamente inviável;
4. A automação dos sistemas instrucionais inteligentes é a única solução viável para fornecer a aprendizagem virtual sem limites de tempo e espaço.

A importância dessas suposições não pode ser extrapolada como a única forma de oferta efetiva de ensino e aprendizagem em ambiente virtual. Como se pode verificar, as

iniciativas expostas anteriormente estão voltadas para a promoção da aprendizagem em ambientes estruturados e totalmente controlados, especialmente em contextos corporativos.

1.2.1 Os enfoques de objetos de aprendizagem e as pesquisas atuais de ensino e aprendizagem

Enquanto tais propostas se conciliam com as pesquisas de educação, sob o enfoque tecnicista e behaviorista das décadas de 1970 e 1980, as necessidades atuais de OA contemplam outros enfoques, com referenciais na área socioconstrutivista ou sociointeracionista. Três das principais fraquezas dos atuais enfoques de aprendizagem virtual (descontextualização linearizada, falta de intermediação e o processo de automação) em larga escala são descritos abaixo:

1. Aprendizagem contextualizada/descontextualizada:

No processo de desenvolvimento de um curso em ambiente virtual, os objetos de aprendizagem são movimentados para a sua descontextualização. Tal fato é verdadeiro no sentido de ser inversamente proporcional o relacionamento entre o tamanho do OA e o seu potencial para reutilização, ou seja, quanto maior for o objeto, menor será a probabilidade de ser reutilizado.

Rechker e Gibbons (2000) procuram demonstrar que o uso de um objeto de aprendizagem é mais bem descrito como se a **contextualização** fosse o seu sinônimo, isto é, o uso de um objeto de aprendizagem por si só já deveria ser contextualizado. Significa que tanto um desenvolvedor de cursos virtuais como os sistemas automatizados, ao usarem um OA estão na verdade colocando o objeto num contexto instrucional.

O relacionamento entre o contexto interno do OA e o externo, no qual ele está sendo colocado, determina se o objeto deve ou não pertencer àquele do OA, pois quanto menos específico for este, maior será a probabilidade de aderência do objeto ao conjunto instrucional no qual está sendo colocado. Por outro lado, quanto mais específico for o OA, menor a probabilidade de aderência desse objeto ao contexto instrucional, ou seja, menor será a probabilidade de reutilização dele. A título de exemplo, e para ilustrar o que foi acima descrito, pode-se mencionar que a imagem de uma molécula tem mais probabilidade de ser utilizada em diferentes contextos instrucionais do que em uma lição completa sobre unificação molecular. Os desenvolvedores de OA costumam remover os contextos internos dos referidos objetos no sentido de maximizar a reutilização destes. Diante disso, há um paradoxo que se produz pelo fato de as teorias modernas de ensino e aprendizagem atribuírem maior ênfase à preeminência do contexto da aprendizagem, usando a linguagem como **contexto social** (VIGOTSKY, 1981); **cultural, histórico e institucional** (WERTSCH, 1991), e **situacional** (JONASSEN, 1991; LAVE; WENGER, 1990), pois, enquanto o alvo de um processo de ensino e aprendizagem é a contextualização social, histórica, cultural e institucional, estes são fatores cruciais que devem ser considerados no *design* de um ambiente virtual de aprendizagem e nos respectivos OA. O simples alinhamento e/ou sequenciamento desses OA descontextualizados não irá produzir um significativo contexto para que a aprendizagem realmente se concretize.

2. O objeto de aprendizagem como instrumento de intermediação:

Os OA são geralmente dispostos em partes de conteúdos ou contêineres de informação. Tendo em vista que alguns teóricos da educação focam a aprendizagem contextualizada, o mundo real, a autenticidade e a utilização de metodologias de ensino e aprendizagem, surgiram como resposta algumas estratégias, tais como: cenários baseados em casos (SCHANK; BERMAN; MACPHERSON, 1999) ou o aprendizado baseado em problemas (ALBANESE; MITCHELL, 1993; VERNON; BLACK, 1993).

Quando o aprendizado acontece no contexto da solução de um problema, os OA e outros recursos são alterados em seu contexto, pois de conhecimento inerte, extraído de especialistas, é assimilado pelo estudante, por meio de mediações, um conhecimento com caráter significativo e contextualizado. Para ilustrar, descrevem-se três situações em que isso ocorre, ou seja, do ponto de vista da semiótica, esse conhecimento é intermediado e formatado pela ação dos estudantes (WERTSCH, 1991), por intermédio de cartões (VIGOTSKY, 1981) e, por fim, pela interpretação das tarefas coloridas (LEONTIEV, 1932). Neste último experimento de cores proibidas, diversos temas eram perguntados para descrever determinados elementos, que poderiam ter o nome de qualquer cor, mais de uma vez. Várias proposições foram fornecidas com os cartões correspondentes às cores, para o uso durante o experimento. Ao aplicar o uso dos cartões em temas mais atuais, o resultado não alcançou a expectativa dos propositores, mas os mais antigos puderam usar os cartões como instrumento para intermediar o desempenho de sua tarefa, por exemplo, virando a face do cartão para baixo uma vez que aquela cor já tivesse sido usada. Esse aspecto instrumental de OA, em outras palavras, é a maneira como um OA faz a intermediação para

a execução de atividades com vistas à solução de problemas. Wertsch (1991, p.68) propõe a pesquisa da ciência social na ação da intermediação, sugerindo que nenhum estudante, que está aprendendo em um ambiente virtual, com a utilização dos recursos daqueles ambientes, pode estudar de forma isolada.

Ao invés de estudar os OA fora do contexto, a unidade de análise da pesquisa deve focar o uso atual dos objetos dentro do contexto da aprendizagem. O autor lembra que “somente sendo parte da ação faz com que o significado da intermediação venha ao encontro da representação do seu papel. Eles não têm o mágico poder neles próprios”.

3. Automação dos objetos de aprendizagem:

Muitos pesquisadores que investigam o tema OA, com o propósito de estabelecer comunicação, em qualquer lugar e a qualquer hora, por meio da produção automatizada pelo computador dos objetos de aprendizagem personalizados (ADL, 2000; HODGINS, 2000; LTSC/IEEE, 2000; 2002; MARTINEZ, 2001), fazem-no em busca de um *design* instrucional de baixo custo. A consequência desta automação é similar ao modelo de instrução baseado em computador da década de 1970, cujo enfoque isolacionista representa uma disparidade se comparado às modernas teorias de ensino e aprendizagem que enfatizam a importância da colaboração (NELSON, 1999), a aprendizagem cooperativa (JOHNSON; JOHNSON, 1997); (SLAVIN, 1990), as comunidades de aprendizagem (BROWN, 1994), a negociação social (DRISCOLL, 1994) e o empreendimento do seu próprio aprendizado (ROGOFF, 1990).

1.2.2 Outras questões relacionadas com os enfoques de objetos de aprendizagem

A condição de reutilização é o primeiro objetivo para o *design* de um OA diante de uma variedade muito ampla de contextos de aprendizagem. Esses OA são geralmente projetados de maneira altamente descontextualizada. Autores como Reigeluth e Nelson (1997), South e Monson (2002), defendem que, quando se trabalha com uma mídia instrucional de qualquer tipo, os desenvolvedores desmontam os materiais dentro das partes dos componentes, para depois remontar a mídia de acordo com as suas necessidades individuais.

Entretanto, Wiley, Reckher e Gibbons (2001) sustentam que a mídia altamente descontextualizada é, na verdade, mais difícil de ser utilizada em um desenvolvimento instrucional, pela:

1. Dificuldade de indexá-la para a descoberta e uso humano;
2. Inabilidade dos computadores em identificar o significado e por conseguinte combinar a mídia primitiva dentro de uma unidade, com o devido significado instrucional.

No campo da semiótica, OA e outros recursos educacionais são na verdade sinais que podem ser textos, gráficos, áudios, animações, etc. A tarefa principal dos usuários de OA é combinar os recursos individuais dentro de lições de significado instrucional. O que pode ser similar é a tarefa de um orador combinar palavras individuais e a sua respectiva pronúncia que estejam dentro de um significado para o processo de comunicação. Tal fato é tanto verdade que, ao verificar o **influxo do senso** dito por (VIGOSTSKY, 2001), pode ser perfeitamente aplicado a questão da montagem de OA.

No estudo da linguagem, o significado das palavras e sentenças que avançam seguido por uma palavra individual, por exemplo, a palavra **senso** na sentença seguinte “*dá cor, vida ou sentido*” ao significado da palavra. Em outras palavras, a entonação altera, significativamente, o significado de uma palavra.

A esse respeito Vigotsky (2001, p. 398 - 401) explicita:

O significado da palavra é uma unidade indecomponível de ambos os processos e não podemos dizer que ela seja um fenômeno da linguagem ou um fenômeno de pensamento. A palavra desprovida de significado não é palavra, é um som vazio. Logo, o significado é um traço constitutivo indispensável da palavra. (...) Consequentemente, estamos autorizados a considerar o significado da palavra como um fenômeno do pensamento. A descoberta da mudança dos significados das palavras e do seu desenvolvimento é a nossa descoberta principal, que permite, pela primeira vez, superar definitivamente o postulado da constância e da imutabilidade do significado da palavra, que servirá de base a todas as teorias anteriores do pensamento e da linguagem. (...) É natural que, do ponto de vista do desenvolvimento do aspecto fonético da fala, o desenvolvimento do significado das palavras se torne inexplicável e impossível. O senso da diferença do fluxo das palavras dentro de outra – literalmente influencia outra palavra – então a que está contida antes, modifica a quem vem depois. Assim, a palavra mantém a recorrência em um livro ou poema. Algumas vezes absorve toda a variedade de senso contida nela e torna-se, de uma forma, equivalente ao seu próprio funcionamento. (...) No funcionamento do pensamento discursivo de um homem maduro não podemos descobrir nada senão um movimento linear contínuo, que transcorre em uma superfície por vias associativas e a palavra. A compreensão da linguagem consiste numa cadeia de associações, que surgem na mente sob a influência das imagens semióticas das palavras.

Ao criar uma expressão vocal significativa, cria-se uma ação na qual as palavras e outras expressões vocais se sobrepõem ao significado do contexto de absorção que colocados todos juntos são misturados para criar o significado. Retornando à questão dos OA, a combinação ou sequenciamento de recursos educacionais cria um contexto nos quais os recursos destacam cor e absorvem sentido dos outros significados. Este significado complexo de elaboração pertence ao ser humano, pois todo pensamento tem um movimento, um fluxo, um desdobramento, no qual o pensamento executa tarefas e atividades. Na mais básica das aplicações instrucionais, os computadores não têm condições de se engajar em um tipo de

significado complexo que requer criar novos significados para toda a instrução baseando-se em um OA. Isto implica a condição e a habilidade do ser humano, exclusivamente, para elaborar e montar OA por meio de suas próprias mãos, por mais rudimentar que seja o conteúdo instrucional. De acordo com Vigotsky (2001, p.409), “*todo pensamento procura unificar alguma coisa, estabelecer uma relação entre coisas*”.

1.2.3 Em direção a uma nova teoria

Em resumo, existe uma grande lacuna que precisa ser preenchida no que diz respeito ao desenvolvimento de uma abordagem teórica, para o *design* e a utilização de OA em comunidades virtuais visando solucionar problemas na execução do trabalho colaborativo.

Para efeito desta pesquisa, existem enfoques que devem ser reexaminados à luz das atuais pesquisas no campo da aprendizagem, levando-se em conta as dificuldades com os atuais enfoques que são discutidos neste texto. A unidade de análise pode não ser também o usuário ou membro de um ambiente virtual ou mesmo os OA em si, mas deve ser o significado da relação da intermediação com a ação das pessoas, ou seja, como se efetiva o processo da aprendizagem, pela interação do membro ou usuário de uma comunidade virtual que usa tais recursos educacionais, primordialmente como apoio ao processo de solução de problemas.

Na construção de uma nova teoria, deve-se atentar para descrever o ontológico e funcional aspecto dos OA. Deve-se criar uma taxonomia dos tipos de OA disponíveis para os desenvolvedores instrucionais, alinhada para a real utilização de tais objetos em um domínio complexo. De forma crítica deve ser analisado o uso da metáfora de OA, com os blocos

*LEGO*³⁷, e sugerir uma metáfora mais robusta, talvez constituída ao redor da cadeia molecular, a qual se aproxima mais das noções de contextualização e descontextualização desses objetos de aprendizagem.

1.3 SCORM – Sharable content object reference model

SCORM é uma sigla que significa “Sharable Content Object Reference Model”, (Modelo de Referência de Objeto de Conteúdo Compartilhável). Modelo de referência é um recurso que mostra qual tipo de trabalho será necessário para resolver um problema em específico, como esses problemas podem ser agrupados, qual o padrão mais adequado para a sua solução e como esse padrão pode ser utilizado e operacionalizado.

O SCORM é um padrão de conformidade que corresponde a um conjunto de regras para especificação e desenvolvimento dos OA, que são chamados de Objetos de Conteúdos Compartilháveis (Sharable Content Objects – SCO), bem como das funções dos Sistemas de Gerenciamento de Aprendizagem (Learning Management Systems - LMS)³⁸, onde os OA são armazenados, organizados, sequenciados e operacionalizados de acordo com funções pré-definidas de cada objeto.

O referido modelo oferece três condições básicas de referência, que são:

1. Oferta de um guia de orientação voltado para os desenvolvedores de conteúdos de aprendizagem em ambientes virtuais;

³⁷ LEGO é um conjunto de blocos pequenos que permitem múltiplas formas de encaixe, sendo permitido o seu uso para montar tipos diferentes de objetos.

³⁸ LMS é um ambiente de gestão de conteúdos educacionais baseado na tecnologia *Web*, em que é possível armazenar os objetos de aprendizagem, bem como executá-los de forma sequenciada, de acordo com os propósitos e objetivos de aprendizagem. É possível também disponibilizar avaliações de aprendizagem.

2. Garantia de que o modelo de referência deve ser compreendido, adotado e utilizado pela maioria dos envolvidos, em especial pelos desenvolvedores de conteúdos e seus usuários;
3. Garantia de que o modelo viabilize o mapeamento da proposta do desenvolvedor, com o objetivo de demonstrar como o conteúdo pode ser operado, considerando-se o modelo de referência.

Com a finalidade de estimular os desenvolvedores a utilizarem o padrão SCORM, como modelo de referência para os seus produtos, alguns fundamentos conceituais e características são estabelecidos, a saber:

1. Acessibilidade:

É a característica de localizar e acessar os componentes instrucionais de um local remoto e distribuí-lo para outros locais;

2. Adaptabilidade:

É a característica necessária de se projetar atividades instrucionais voltadas para as necessidades individuais e/ou organizacionais;

3. Custo:

É a característica de aumentar a eficiência e produtividade para a redução de tempo e custos do desenvolvimento e disponibilização do conteúdo instrucional;

4. Durabilidade:

É a característica de garantir o funcionamento diante da evolução tecnológica e das mudanças, sem o custo de um novo desenvolvimento;

5. Interoperabilidade:

É a característica de dispor de componentes instrucionais desenvolvidos para determinado ambiente, que possam operar em outros ambientes;

6. Reusabilidade:

É a flexibilidade para incorporar componentes instrucionais em múltiplas e diferentes aplicações educacionais e contextos.

Muito em razão do surgimento e popularidade dos ambientes baseados na *Web*, outros fundamentos conceituais são acrescentados com o objetivo de maximizar o acesso e reuso dos conteúdos de aprendizagem. As principais razões fundamentam-se no seguinte:

1. As tecnologias e infraestruturas baseadas na *Web* são expandidas com rapidez e têm sido a base para as novas tecnologias de aprendizagem;
2. Os padrões das tecnologias baseadas na web ainda não estão bem definidos e disseminados;
3. Os conteúdos baseados na web podem ser distribuídos usando o próprio meio ou outros meios, tais como: CD-ROM, DVD, outros sistemas que funcionam sem conexão com a Internet (off line) ou os ambientes conectados em rede.

Considerando a rápida evolução e tendências dos ambientes de aprendizagem baseados na *Web*, o padrão SCORM tem acompanhado tal evolução oferecendo e viabilizando o seu processo, permitindo as seguintes possibilidades:

1. A condição de um LMS operar um conteúdo de aprendizagem que foi desenvolvido por ferramentas de diferentes fornecedores e possibilitar a troca de dados entre esses ambientes;
2. A condição de um LMS operar um conteúdo de aprendizagem que foi desenvolvido por ferramentas de diferentes fornecedores e possibilitar a troca de dados e informações em tempo real.

O funcionamento dos conteúdos, ou seja, os OA em ação, por intermédio das funcionalidades disponibilizadas por um LMS, são constituídos por três modelos:

1. Agregação de conteúdo;
2. Ambiente de execução do conteúdo formado pelos OA;
3. Disposição sequencial dos OA e critérios de navegação.

1.3.1 O SCORM e outros padrões de conformidade

A ADL tem adaptado as especificações, os padrões e os guias de referências desenvolvidos por outras organizações para a formação de um modelo de referência integrado, de modo compatível aos objetivos do modelo SCORM.

Dentre muitas organizações que têm por objetivo estabelecer padrões de conformidade voltados para o desenvolvimento de conteúdos de aprendizagem e suas funcionalidades, quatro dessas tem especial destaque no processo de parceria e colaboração concernente à evolução e integração dos padrões existentes e propostos:

1. ARIADNE – Alliance of Remote Instructional Authoring & Distribution Networks for Europe;³⁹
2. AICC – Aviation Industry CBT Committee;⁴⁰
3. IEEE – Institute of Electrical and Electronics Engineers – LTSC – Learning Technology Standards Committee;⁴¹
4. IMS Global Learning Consortium, Inc.⁴²

³⁹Disponível em: <http://www.ariadne-eu.org> . Acesso em: 9 jan. 2008.

⁴⁰Disponível em: <http://www.aicc.org/> Acesso em: 9 jan. 2008.

⁴¹Disponível em: <http://ieeeltsc.org/> .Acesso em: 9 jan. 2008.

⁴²Disponível em: <http://imglobal.org/>.Acesso em: 9 jan. 2008.

O compromisso de manter o relacionamento com tais organizações para assegurar a integração entre os diferentes padrões de conformidade tem como principal objetivo assegurar que os conteúdos de aprendizagem, desenvolvidos de acordo com o padrão SCORM, possam ser executados em todas as suas funcionalidades por múltiplos LMS, baseados na *Web*, além do acesso aos repositórios de dados e objetos.

De acordo com a ADL, a função-chave de um LMS é a capacidade de gerenciar a execução dos objetos de aprendizagem mediante a experiência de aprendizagem dos usuários.

Gonzalez (2005, p. 123), ao analisar o padrão SCORM afirma que:

O modelo SCORM especifica um formato comum para os objetos de tipo instrução, avaliação e prática individual, que são os objetos utilizados pelos ambientes de aprendizagem orientados ao modelo de aprendizagem individual. Para as atividades de aprendizagem em grupo não foram encontradas definições para os recursos instrucionais envolvidos, nem especificações para a criação e uso desses objetos, tanto para o formato da estrutura estática da aplicação como especificações para o formato do comportamento dinâmico da sua execução. Aqui, infere-se que é necessário descrever as atividades em grupo de forma geral e simples, de modo que seja possível reconhecer os recursos instrucionais que participam nessas atividades e do que oferecem apoio à realização efetiva da aprendizagem colaborativa.

O modelo SCORM é um padrão de conformidade muito eficaz para a utilização em projetos instrucionais estruturados e com objetivos de aprendizagem bem definidos. Os OA que são desenvolvidos de acordo com este padrão possuem escopo específico, ou seja, são “artificialmente” produzidos ou fabricados por profissionais especializados, por meio de metodologia e tecnologia apropriadas. No caso específico dos OA que são derivados dos fóruns de discussão das CVAO, são produzidos de forma espontânea como resultado da interação entre os sujeitos participantes de um processo de troca de informações e conhecimento, para a busca da solução de problemas, e não possuem qualquer característica ou semelhança com os OA produzidos de acordo com o modelo SCORM.

Rudzki e Jonson , Von Krogh et al. e Sowe et al. apud Nascimento (2009, p.144) observaram que o ciclo da gestão do conhecimento (captura, representação, armazenamento e reutilização de conhecimento) nem sempre apresenta suporte ou estrutura adequada para o acesso, busca e/ou recuperação do conhecimento nestes ambientes. Acrescentam ainda que (...) “a dificuldade de acessar o conhecimento gerado em interações passadas para entender o resultado de uma discussão, e poder reutilizá-lo, é um problema identificado nas comunidades”.

Nesta direção, Nascimento e Santoro (2009) afirmam que este problema é causado em parte por falta de estruturação no conteúdo das interações em fóruns. Tal constatação corrobora que com a ausência de um padrão de conformidade, o conhecimento gerado é armazenado de forma não estruturada, dificultando a busca e recuperação para a sua reutilização.

Neste sentido, há que se desenvolver um padrão de conformidade para os OA construídos por meio das interações dos fóruns de discussão, considerando-se o contexto sociocultural no qual estão inseridos.

1.4 Fóruns de discussão

Os fóruns de discussão têm sido uma das principais fontes de informação virtual, para a coleta de dados dos usuários conectados na Internet. As formas de comunicação mediadas por computador têm sido a base para a análise da interação e socialização dos sujeitos que se comunicam por esse meio. Os citados fóruns têm sido alvo de estudos não só nos aspectos comportamentais de seus participantes, mas também nos produtos gerados por tais interações e relações, já que, por ser um meio virtual, há registros concretos sobre as ações no entorno virtual.

1.4.1 Conceituação

Hine (2000, p.30) afirma que “os estudos do enfoque naturalista de entorno virtual tomam nota da riqueza e complexidade presentes no uso das comunicações mediadas por computador como ponto de partida para analisar comportamentos”. Nessa linha de pensamento, Muñoz (2007) aprofunda essa abordagem, afirmando que convém referenciar uma taxonomia para a efetiva análise dessa questão. Afirma ainda, que é necessária a diferenciação da comunicação **síncrona** (quando os usuários da rede se comunicam dentro do mesmo espaço temporal, como exemplo pode ser citado os “chats” ou popularmente conhecido como “bate papo”) e a comunicação **assíncrona** (quando os usuários da rede se comunicam em tempos diferentes, simplesmente postando as mensagens e aguardando as respostas em outros momentos, como exemplo pode ser citado os fóruns de discussão).

No presente estudo optou-se pela comunicação assíncrona dos fóruns de discussão, como forma de coleta de informação sobre o comportamento e conteúdo das discussões, na configuração de objetos de aprendizagem. A comunicação assíncrona emerge no âmbito da internet de forma completamente oposta às formas de comunicação tradicional, porque a comunicação síncrona apenas representa a comunicação tradicional entre os sujeitos, mediada por recurso tecnológico, possibilitando o compartilhamento da discussão, ao mesmo tempo com os sujeitos em diferentes regiões geográficas, mas sofrendo dos efeitos de rompimento da comunicação, como resultado de falhas técnicas ou outros motivos relacionados. Por outro lado, a comunicação assíncrona promove a interação entre os sujeitos, mesmo que não ao mesmo tempo, mas os discursos são mais reflexivos e gerados na ausência da instantaneidade. É possível afirmar que a eliminação do marco temporal compartilhado ameniza a pressão de dar uma resposta instantânea para o interlocutor, além de possibilitar o acesso a um tipo de interação social mais estruturado e introspectivo que em muitas ocasiões se sobrepõe a um

simples manifesto, sem importância ou relevância, para o assunto em debate. Na verdade, é um espaço virtual em que são propiciadas condições e estímulo da postagem de discursos textuais que se apresentam de forma estruturada e acessível, por meio da interação social entre os sujeitos participantes. Nessa perspectiva, o fórum de discussão pode ser assim conceituado:

Um espaço virtual o qual permite que um determinado número de usuários na rede, com a finalidade de conhecer, produzir e trocar conhecimentos, colaborar e/ou colocar os seus pontos de vista individuais sobre um tema de forma compartilhada (MUÑOZ, 2007, p.7).

O fórum de discussão geralmente é conduzido por um moderador, que é responsável pela supervisão e dinamização das atividades, notadamente no que se refere ao funcionamento e conduta dos sujeitos participantes do grupo. A dinâmica e operacionalização do fórum de discussão são bastante simples, pois cada sujeito acessa o ambiente, por meio da *web site*, introduzindo a sua identificação: nome (ID) e senha. O usuário escolhe o tema que pretende participar e inicia a sua colaboração. As discussões, para uma melhor visualização e compreensão da sua evolução, são apresentadas de forma endentada (*threads*).

1.4.2 Taxonomias

Muñoz (2007) sugere uma classificação taxonômica com o intuito de dar seguimento aos conceitos básicos sobre os fóruns de discussão. Para efeito dessa classificação taxonômica, apresenta-se o seguinte:

1. **Suporte comunicativo:** É a forma ou modo com que os usuários se comunicam entre si, que pode ser assim diferenciado:
 - a. **Fóruns textuais:** a comunicação virtual acontece por meio de texto, pois todas as interações se articulam de forma escrita.

- b. Fóruns multimídia:** a comunicação virtual acontece por meio de conteúdos em forma de multimídia (imagens, sons, vídeo, animações, etc.). Neste tipo de fórum, o texto tem apenas uma função complementar.
- 2. Temática:** Geralmente, o fórum de discussão estrutura-se com base em determinado tema. Os usuários interagem entre si, postando mensagens na forma textual ou multimídia, sobre o tema do fórum. A seguinte tipologia pode ser definida:
- a. Fóruns gerais de entretenimento e notícias:** são os fóruns que apresentam a sua temática direcionada para diversão, cultura geral, notícias e atualidades, como exemplo tem-se: os fóruns de notícias, cinema, hobbies, esportes, viagens, etc.
- b. Fóruns colaborativos:** são os fóruns onde os participantes prestam ajuda mútua, recíproca e desinteressada para a busca de solução de problemas, cuja origem pode estar associada à discussão em si, ou a alguma atividade pela qual o fórum seja um recurso. Isso ocorre normalmente quando o fórum é parte de uma comunidade virtual de aprendizagem auto-organizada.
- 3. Interação usuário-ferramenta:** um aspecto relevante que incide diretamente sobre o desenvolvimento e funcionamento de qualquer fórum de discussão é relativo à utilidade que tem para os participantes. Nesse sentido, é relevante diferenciar o seguinte:
- a. Uso passivo:** é quando o usuário acessa um fórum de discussão apenas para tomar conhecimento das informações ali disponibilizadas, com o objetivo precípuo de satisfação do seu interesse particular. Há um termo específico para esse tipo de usuário, que usualmente é chamado de “*lurkers*”. Na verdade, é um usuário observador do fórum, que não participa efetivamente das discussões, mas usufrui da dinâmica e dos resultados do fórum.

- b. Uso ativo:** é quando o usuário acessa o fórum de discussão como um participante ativo dos debates e discussões, colaborando, interagindo, contribuindo e construindo conhecimentos com os demais participantes.
- 4. Função investigadora:** basicamente, é quando o fórum assume a função de disponibilidade para o acesso a pesquisadores, informações produzidas, análise e pesquisa. Sob esta perspectiva, os fóruns podem ser assim definidos:
- a. Fóruns primários:** são aqueles cujo formato da estrutura e das discussões são projetadas sob um propósito explícito e específico, ou seja, é um fórum exclusivamente criado para atender a uma demanda de pesquisa pré-determinada.
- b. Fóruns secundários:** são aqueles cujo formato da estrutura e das discussões é projetado sob um tema proposto e aberto ao público em geral, e que foi originado de forma espontânea. Geralmente são os fóruns que fazem parte do entorno das comunidades virtuais de aprendizagem auto-organizadas. Nesse caso, as informações são utilizadas por pesquisadores, de forma que estes não interferem no funcionamento e direcionamento do fórum. Geralmente o pesquisador atua nestes fóruns como um “*lurcker*”.

1.5 Teorias de aprendizagem

Para efeito deste trabalho serão categorizadas algumas teorias de aprendizagem relacionadas à solução de problemas no processo do trabalho colaborativo, por meio dos fóruns de discussão.

Nelson (1998) categorizou as seguintes teorias e enfoques de aprendizagem, dirigidas para este tema:

Quadro 2 - Enfoque e categorias das teorias de aprendizagem

<i>Colaborativo</i>	<i>Experiência</i>	<i>Solução de Problemas</i>
Aprendizagem cooperativa	Aprendizagem baseada na experiência	Aprendizagem baseada em problemas
Aprendizagem cognitiva	Aprendizagem ativa	Aprendizagem baseada em projetos
		Aprendizagem baseada em casos

1.5.1 Enfoque colaborativo

O enfoque de aprendizagem colaborativa, por definição, é utilizado pelos pequenos grupos de trabalho colaborativo como meio educacional para o suporte ao processo de solução de problemas. A colaboração pressupõe a realização de atividades de forma coletiva, ou seja, a tarefa de um complementa o trabalho dos outros. Essa interdependência exige aprendizados complexos de interação permanente. O enfoque colaborativo desencadeado nas CVAO não dilui os participantes em um “coletivo de iguais” nem tem por objetivo alcançar o idealizado nível máximo de aprendizagem igual para todos.

O enfoque colaborativo de aprendizagem compreende dois tipos:

1. Aprendizagem cooperativa:

Este tipo de aprendizagem tem recebido força e estímulo para o seu uso em processo de colaboração no desenvolvimento de atividades em grupo em salas de aula e no desempenho acadêmico e competências sociais dos alunos nas escolas secundárias americanas (HUMPHREYS; JOHNSON; JOHNSON, 1982; JOHNSON; JOHNSON, 1989; PHELPS; DANSON, 1989; SHARAN, 1994; SHARAN; HERTZ-LAZAROWITZ, 1980; SHARAN; SHARAN, 1984; SLAVIN, 1990a). Geralmente as tarefas propostas são estruturadas de forma a proporcionar atribuições específicas e

contribuições individuais de cada membro do grupo. No entanto, cada membro é responsável não só pela sua aprendizagem individual, mas também por ajudar os seus colegas a se desenvolver cognitivamente e socialmente num ambiente de interação no grupo. Sharan (1984) assinala que, de todos os métodos cooperativos mais usados destacam-se: a) O *Jigsaw*, utilizado como material narrativo, durante o qual cada membro do grupo fica responsável pela aprendizagem de uma parte específica de determinado tópico, juntamente com representantes do restante do grupo. A avaliação final é individual e abrange todas as partes do tópico em estudo. b) As investigações em grupo (SHARAN; SHARAN, 1984) centram-se no desenvolvimento de competências de pensamento como a análise, síntese e a avaliação e se realizam mediante projetos coletivos. Segundo Phelps e Damon (1989), a colaboração entre pares constitui uma estratégia útil no ensino de conceitos cognitivamente complexos. Várias investigações revelam alguma eficácia da aprendizagem cooperativa, principalmente no que se refere ao aumento do desempenho e desenvolvimento de competências cognitivas sociais. No entanto, a heterogeneidade dos grupos nem sempre se tem revelado decisiva na obtenção desses resultados. Num estudo realizado por Humphreys et al. (1982), o desempenho dos alunos em aprendizagem cooperativa revelou-se significativamente superior ao dos restantes. Portanto, a aprendizagem colaborativa oferece um conjunto de orientações de como organizar os grupos de aprendizagem e sugere atividades específicas para que as próprias experiências de aprendizagem possam ser estruturadas. As contribuições que circulam nas CVAO expressam o somatório das individualidades, percepções e racionalidades

que contribuem para a “constituição rizomática dos saberes” (KENSKI, 2008, p. 113), em constante transformação, bem como contribuem para o desenvolvimento de competências cognitivas sociais.

2. Aprendizagem cognitiva:

A aprendizagem é uma função da atividade dentro do contexto e cultura em que ela ocorre (LAVE; WENGER, 1990; MCLELLAN, 1995). A aprendizagem acontece não só pelo engajamento do sujeito na execução de atividades educativas, mas também por meio da construção social do conhecimento (WERTSCH, 1985). A interação social é um componente crítico do aprendizado situado – aprendizes tornam-se envolvidos em uma "comunidade de prática" que incorpora certas convicções e comportamentos a serem adquiridos. À medida que o iniciante se movimenta da periferia dessa comunidade para o seu centro, ele se torna mais ativo e envolvido dentro da cultura. A partir daí assume um papel de especialista ou de mais antigo, o que Lave e Wenger (1991) denominam de processo de "participação periférica legítima".

Além disso, o aprendizado situado é normalmente não-intencional, em vez de deliberado. Esse tipo de aprendizagem fornece ao estudante a oportunidade de ser orientado para a participação juntamente com outros estudantes, os quais podem suportar e alavancar as suas capacidades para conseguir novas compreensões e habilidades. Aprendizagem significativa na perspectiva de Brown, Collins e Duguid, 1989; Rogoff, (1990, p.161) “sustenta o aprendizado em um campo onde os sujeitos adquirem, desenvolvem e usam as ferramentas cognitivas em uma autêntica atividade

de domínio.” O aprendizado avança por meio de uma interação social, colaboradora e da construção social de conhecimento.

Os autores também enfatizam a necessidade de uma nova epistemologia para o aprendizado – uma que ressalte a percepção ativa sobre conceitos e representação. Suchman (1988) explora a estrutura do aprendizado situado no contexto de inteligência artificial. O aprendizado situado tem antecedentes no trabalho de Gibson (teoria dos recursos) e Vigotsky (aprendizado social). Além disso, a teoria de Schoenfeld sobre a resolução de problemas matemáticos incorpora alguns dos elementos críticos da estrutura do aprendizado situado. O aprendizado situado é uma teoria geral de aquisição de conhecimento, que foi aplicada em atividades de aprendizado baseadas na tecnologia para escolas, que focalizam as habilidades na resolução de problemas (COGNITION & TECHNOLOGY GROUP AT VANDERBILT, 1993).

McLellan (1995) fornece uma revisão de artigos que descrevem várias perspectivas sobre a teoria. Os princípios que embasam este conceito são:

- a.** O conhecimento deve ser apresentado em um contexto autêntico, isto é, em situações e aplicações que normalmente envolvem o conhecimento;
- b.** O aprendizado requer interação e colaboração social.

1.5.2 Enfoque baseado na experiência

O enfoque baseado na experiência é o entendimento de que os eventos de aprendizagem ocorrem muito próximos e associados às experiências do mundo real, os quais podem fornecer aos estudantes uma oportunidade de aprendizagem.

1. Aprendizagem baseada na experiência:

Este tipo de aprendizagem acarreta discussão qualificada, vivência e diálogo. Quanto mais os participantes possam relatar suas experiências e o que já sabem no contexto online, mais entenderão o que aprendem. O processo de conectar as experiências cotidianas na aprendizagem não apenas confere uma sensação de maior importância aos participantes, mas também os valoriza como pessoas que têm vivência, conhecimento e podem aplicá-los em outros contextos e situações. Dewey (1994, p.10), filósofo norte-americano, em sua obra *Democracia e Educação* situa a capacidade de experienciar a natureza, de autorrenovação e de uso de energias do ambiente para sua conservação e crescimento como a característica mais emblemática da vida. Para o autor

A mais notável distinção entre seres vivos e seres inanimados é que os primeiros mantêm-se a si próprios através da renovação. Uma pedra quando recebe um choque resiste. Se a resistência é maior que a força do choque, a pedra permanece inalterada. Em caso contrário, fragmenta-se em pequenos pedaços. (...) Apesar de que o ser vivo pode facilmente ser esmagado por uma força superior, ele, entretanto, tenta transformar as energias que agem sobre ele em meios de perpetuação de sua própria existência.

Nem toda atividade humana constitui experiência, configurando-se como tal somente as ações acompanhadas da percepção consciente da mudança produzida no ambiente, e do seu reflexo. Dessa forma, o fluxo da ação e da onda reflexa de suas consequências ganha significado, torna-se uma

aprendizagem. “Aprender da experiência”, afirma Dewey (1994), “é estabelecer conexões para diante e para trás entre as coisas que fazemos e o que apreciamos ou sofremos das coisas em consequência” Dewey (1981, p.495-96). Nessa direção, o fazer torna-se um experimento com o mundo e o que dele sofremos torna-se aprendizagem, a descoberta da conexão das coisas. O valor cognitivo de uma experiência repousa, portanto, na percepção de relacionamento ou continuidade a que ela induz.

Na visão do autor, não há aprendizagem genuína em processos divorciados da experiência, em que se memorizam fatos sem perceber relacionamentos, gerando um conhecimento superficial e destituído de significado pessoal para o ser que aprende. Portanto, a aquisição de significado torna-se, assim, o elemento vital para que uma experiência seja educativa. E para que isso ocorra, o ato de reflexão deve, necessariamente, estar presente.

Uma experiência pode limitar-se à fase de tentativa e erro até que se encontre uma forma de agir que funcione e é adotada como padrão. Ou pode ir além, num estágio em que a função reflexiva descobre conexões entre ações e consequências, percebendo os detalhes das relações de causa e efeito. O produto final da experiência reflexiva é a capacidade de controlar as condições objetivas, de propiciar os fatores que facilitam a produção de determinados efeitos e de eliminar outros que produzem resultados indesejáveis. A crença de que uma teoria pode ser apreendida com base em uma fórmula verbal nasce da concepção de que primeiro percebemos as coisas, as ideias, para depois percebermos suas relações, como se fossem itens separados do conhecimento a serem ligados pela função do julgamento.

Dewey estabelece dois princípios interligados para a avaliação do significado educacional das experiências: *o continuum experiencial e a interação*. O *continuum experiencial* baseia-se na noção de que cada experiência modifica o sujeito da ação, alterando a qualidade das experiências subsequentes. A cada ação realizada, um novo sujeito emerge, exibindo novas atitudes emocionais e intelectuais, com uma sensibilidade diferente para responder futuras situações-problema. A *interação* leva em conta que a experiência não se realiza unicamente na esfera da subjetividade, mas acontece em um mundo de pessoas e coisas que representam a cultura acumulada pela sociedade.

Nossa aprendizagem se processa em um tecido social, numa tentativa de conquistar uma compreensão e um agir comuns. Importa, por conseguinte, no processo educativo, desenvolver uma intimidade experiencial com a comunidade e suas condições físicas, sociais, econômicas e históricas, para que a aprendizagem possa ganhar significância. O ambiente com o qual o sujeito interage possui o potencial de evocar as respostas desejadas e determina, em grande parte, a riqueza da experiência educativa.

Em se tratando de comunidades virtuais, pode-se inferir que a aprendizagem experiencial pressupõe um significado de unidade vida-formação-desenvolvimento pessoal, com capacidade potencial para lidar com situações desconhecidas e problematizadoras, bem como compartilhar experiências num processo permanente de construção/reconstrução do conhecimento.

2. Aprendizagem ativa:

Atualmente, crianças e jovens conseguem acessar, com muita rapidez, várias formas de mídias (de videogames à internet) e buscam de forma mais ativa o conhecimento e o entretenimento. Porém, apesar da lacuna tecnológica entre gerações, participantes das comunidades virtuais criam ativamente o conhecimento e o significam por meio da experimentação, exploração, manipulação, colaboração, num processo de aprendizagem ativa e colaborativa. Jonassen et al. (1991) analisam o resultado dessa forma de aprender e concluem que a facilitação proporcionada pelos ambientes de aprendizagem virtual estimula a construção de sentido pessoal, a construção do conhecimento e do significado de interações nesses ambientes. Em outras palavras, o processo educativo está centrado no sujeito, o que determina seu ritmo e direção do processo de aprendizagem.

Palloff e Pratt (2002), em um estudo realizado na década de 1990, indicam que muitos estudantes são concreto-ativos, isto é, aprendem melhor por intermédio de experiências concretas, nas quais seus sentidos estejam envolvidos. Em vista disso, muitos professores estão procurando melhorar sua prática, e os resultados de aprendizagem de seus alunos começam a incorporar estratégias de ensino que viabilizem a aprendizagem ativa, tais como, o trabalho colaborativo, a participação em pequenos grupos de discussão, em projetos, estudo de caso, dramatizações e o uso de simulações.

1.5.3 Enfoque baseado na solução de problemas

1. Aprendizagem baseada em problemas:

Aprendizagem baseada em problemas - *Problem-based Learning* – (PBL) é a aprendizagem que resulta do processo de trabalho orientado para a compreensão ou resolução de um problema. Schmidt (1993) estabelece seis componentes para a aprendizagem no PBL:

- a. O problema;
- b. Os grupos tutoriais;
- c. O estudo individual;
- d. A avaliação do estudante;
- e. Os blocos ou as unidades por meio dos quais se estrutura o currículo.

O problema é o ponto de partida e o fio condutor do processo de aprendizagem. Os estudantes trabalham em torno do problema, organizados em grupos tutoriais assumidos como um dos componentes fundamentais para o PBL. O tutor estimula o processo de aprendizagem e ajuda o grupo a conduzir o ciclo de atividade no PBL, utilizando-se de diversos meios, entre os quais a apresentação de perguntas ou questionamentos sobre um problema central. Após a identificação dos objetivos de aprendizagem, realizada durante encontro do grupo tutorial para análise do problema, os estudantes devem conduzir uma fase de estudo individual. Nessa fase, eles são responsáveis pela identificação do material bibliográfico e mediático, pela decisão e sistematização dos novos conhecimentos para apresentação subsequente ao grupo.

Outro elemento considerado por Schmidt (1993) está ligado à avaliação progressiva, que constitui um meio adequado para compatibilizar a responsabilidade institucional de certificar a qualificação dos estudantes, com a necessária liberdade que deve ser assegurada aos estudantes, no PBL, para selecionar os aspectos ou tópicos relacionados aos problemas a que se propuseram. Por fim, uma característica organizacional central do PBL é a estruturação do currículo em unidades ou blocos, cada um deles tendo um único tema específico como base, focado mediante uma série de problemas inter-relacionados.

O esforço colaborativo, com o qual um ajuda ao outro a esclarecer as questões, é um componente central (WEBB, 1982), que conduz a um número de elementos que não podem ser elucidados com a base de conhecimento existente. Uma lacuna entre o que é conhecido e compreendido e o que não é compreendido será experimentada. Nesse sentido, o PBL pode ser adequado como estratégia de ensino às comunidades virtuais, pois essa fenda entre o que é compreendido e o que ainda precisa ser compreendido induz a uma forte motivação intrínseca da aprendizagem, ou como afirma (VIGOTSKY, 2001), essa etapa pode ser denominada de “*zona de desenvolvimento proximal*”. A colaboração entre sujeitos com conhecimentos diferentes potencializa a aprendizagem e o desenvolvimento. A diferença nos ajuda a compreender que somos sujeitos com particularidades, com experiências próprias, constituídas nos processos coletivos de que participamos; posta em diálogo, essa colaboração enriquece a ação

pedagógica, relacionada à diversificação dos instrumentos mediadores e à ampliação dos modos de sua utilização.

Por conseguinte, o PBL pode ser considerado como forma colaborativa de aprendizagem e instrução, direcionada para a construção de modelos mentais coerentes dos problemas apresentados, bem como forma de aprendizagem contextual ou situacional.

Schmidt (1993), em seus estudos na Universidade de Maastricht (Holanda), afirma que o PBL produz, no processo de aprendizagem do estudante, os seguintes efeitos cognitivos:

- a. Ativação do conhecimento prévio: a análise inicial do problema estimula a recuperação de conhecimentos adquiridos previamente e facilita a compreensão das novas informações;
- b. A elaboração sobre os conhecimentos prévios por meio da discussão em pequenos grupos, tanto antes quanto depois que os novos conhecimentos tenham sido adquiridos, enriquece as redes semânticas desenvolvidas, o que facilita a recuperação do conhecimento;
- c. A reestruturação do conhecimento de modo a se encaixar no problema apresentado contribui para a construção de uma teoria rica em rede semântica;
- d. Aprendizagem contextual: o problema serve como estrutura para armazenagem de pistas que podem auxiliar a recuperação de conhecimentos relevantes, quando necessários para problemas similares posteriormente encontrados;
- e. O estudante percebe a relevância do tema a ser aprendido;

- f. A motivação intrínseca é ampliada pela discussão do problema, pois quando as pessoas se tornam envolvidas com um tema, elas querem saber mais sobre o assunto.

De acordo com o Schmidt (1993, p.108), a Teoria da Aprendizagem Baseada em Problemas contempla:

- a. A ativação e a elaboração do conhecimento prévio para análise de um problema em pequenos grupos;
- b. A construção de modelos mentais orientados por problema na forma de redes semânticas, incluindo chaves contextuais derivadas de problemas profissionalmente relevantes;
- c. A estimulação da curiosidade epistêmica: uma nova classe de abordagens instrucionais que pode ser descrita como construtivista colaborativa e contextual.

2. Aprendizagem baseada em projetos:

Opondo-se às sociedades tradicionais, nossa cultura tecnológica exprime cada vez mais a aprendizagem baseada em projetos. A pedagogia de projetos estimula a introdução de atividades mais dinâmicas na relação ensino-aprendizagem, por meio de atividades cooperativas, baseadas no diálogo, em que professores e estudantes interagem no processo permanente de construção de conhecimentos. O trabalho com projetos proporciona ao estudante situações em que ele é gradativamente estimulado a desenvolver qualidades próprias ao trabalho independente. Vários autores têm contribuído com seus estudos sobre a importância da pedagogia de projetos em direção às aprendizagens significativas e ao processo de ação

colaborativa (BOUTINET, 2002; HERNÁNDEZ, 2000; HERNÁNDEZ; VENTURA, 1998; LEITE 1996; LÜDKE, 2004).

Lüdke (2004) destaca algumas características básicas dos projetos que contribuem para o processo de aprendizagem. São elas:

- a. Ruptura com o esquema tradicional de ensino por disciplinas;
- b. Possibilidade de reunir o que já foi aprendido pelo estudante e o que pode vir a sê-lo, nos vários campos do conhecimento;
- c. Participação ativa, dinâmica dos alunos, desencadeando forças em geral apassivadas no modelo escolar tradicional;
- d. Construção de conhecimento pela investigação própria dos estudantes; articulação entre trabalho individual e coletivo e valorização de atitudes e comportamentos sociais;
- e. Combinação entre o trabalho escolar e o de várias outras instituições e agências.

Embora não devam ser considerados como panacéia ou solução salvadora para os problemas vividos em nossas instituições educacionais, os projetos representam uma via interessante para conjugar esforços e contribuir para autonomia do estudante em direção ao processo de aprendizagem significativa.

O projeto não é uma metodologia didática, mas uma forma de entender o sentido da escolaridade baseado no ensino para a compreensão. (...) Compreender é uma virtude cognoscitiva e experiencial de tradução-revelação entre um original, isto é, uma informação, um problema, e o conhecimento pessoal e grupal relacionado a ela. A compreensão consiste em poder realizar uma variedade de ações de compreensão que mostrem uma interpretação do tema e, ao mesmo tempo, um avanço sobre o mesmo (HERNÁNDEZ, 2000, p.183).

O estímulo para aprender nas comunidades virtuais é um desafio permanente, pois a dinâmica dessas comunidades está orientada de acordo com o princípio do benefício de todos os seus membros. Um tema interessante, gerado na pedagogia de projetos, agrega membros para uma comunidade e, por sua vez, esses membros geram novos conteúdos, que ampliam e enriquecem todo o grupo. Portanto, o processo de compreensão para a aprendizagem baseada em projetos passa pelo pertencimento a um grupo com interesses comuns e pelo questionamento problematizado.

3. Aprendizagem baseada em casos:

O modelo de aprendizagem baseado em casos é construído com situações (casos) que buscam a resolução de problemas sejam de ordem técnica, social, etc. Os casos são mais usados para ensinar habilidades que visem à tomada de decisão no campo profissional. Esse recurso pedagógico utiliza problemas tão próximos quanto possível das condições reais dos estudantes, levando-os desse modo a evoluir e desenvolver sua capacidade na resolução de problemas.

Moreira (2005) aponta algumas vantagens e desvantagens na aplicação desse recurso. As principais vantagens são: consolidação e aplicação de conhecimentos, bem como questionamento de comportamentos; exercitação da criatividade, iniciativa e poder de decisão; discussão construtiva, ouvir e compreender pontos de vista diferentes. As desvantagens, levantadas pelo autor são: inaplicabilidade à transmissão do conhecimento puro ou de aptidões psicomotoras; tempo requerido para a sua aplicação; insegurança perante a ausência de situação ideal.

Pode-se destacar, dentre os modelos de aprendizagem baseado em casos, o Método de Harvard - MH, o Método de Aprendizagem baseado em Problemas - PBL ou ABP e o Método de Aprendizagem baseado em Projetos.

O Método de Harvard surge por volta dos anos de 1920, na Harvard Business School (CHRISTENSEN; HANSEN, 1987) e tem sido adotado por diferentes áreas do saber. Resumidamente, ele é composto pelos seguintes elementos:

- a. Diálogo permanente e sistemático entre todos os envolvidos visando aprendizagens;
- b. A comunicação recai sobre casos ou situações reais;
- c. O conceito é o mesmo de que o processo de decisão;
- d. Aprender não se resume a ouvir, mas implica descobertas, compreensão e transmissão.

O MH não é um relatório de soluções, mas proporciona ao estudante a confrontação entre a sua maneira de pensar e a sua forma de reagir perante um fato ou acontecimento, estimula o autoconhecimento, o protagonismo e valoriza opiniões divergentes.

Moreira (2005) faz uma análise das potencialidades e limites na aplicação desse método, como segue:

- a. **Potencialidades:** melhor assimilação e ampliação de conhecimentos, capacidade de trabalhar em equipe, estudo de situações reais, abordagem de fenômenos complexos, desenvolvimento de competências de formulação de perguntas,

melhoria na capacidade de decisão e aumento do nível de criatividade;

- b. **Limites:** Lentidão na aprendizagem e reduzida exploração das capacidades individuais.

O *ensino com casos investigativos* (WATERMAN, 1998) pode ser organizado de várias formas, desde pares lendo um minicaso em sala de aula, pequenos grupos estudando um caso por período prolongado, um caso sendo discutido em laboratório, ou numa comunidade virtual de aprendizagem. Ressalta-se que um dos elementos mais importante dessa estratégia diz respeito à discussão colaborativa dos temas do caso. Dessa forma, os sujeitos podem identificar o que eles já sabem e o que necessitam descobrir a fim de compreender o caso e formular um problema para investigar.

Waterman (1998) sugere um roteiro para trabalhar com *ensino com casos investigativos*:

a. **Formulação do Problema:**

- Reconhecer temas em potencial;
- Levantar as conexões e definir o espaço do problema;
- Identificar o material a ser aprendido;
- Formular perguntas específicas;
- Definir e especificar o foco;
- Definir outros temas a serem investigados por consulta aos colegas.

b. **Resolução do Problema:**

- Obter referências/fontes adicionais;

- Gerenciar informações;
- Definir melhor o problema (compartilhando pontos de vista/informações);
- Projetar e conduzir informações com softwares de simulações (modelos), com métodos de campo/laboratório, com novas fontes (referências, pesquisas diversas, uso de recurso de mídias), etc.

c. Persuasão de Colegas:

- Apresentar conclusões das investigações;
- Desenvolver análises ou relatórios para persuadir os colegas;
- Conduzir debates de pontos de vista ou resultados opostos;
- Produzir outros materiais que demonstrem conclusões e novas possibilidades de investigação.

A Aprendizagem com Casos parece promissora como mais uma ferramenta a ser utilizada nas CVAO. Essa estratégia incorpora vários elementos identificados nos métodos ativos. Os membros das CVAO trabalham colaborativamente. Iniciam sua aprendizagem com perguntas baseadas no que eles já sabem. Enquanto eles trabalham em equipes, desenvolvem habilidades de comunicação e gerenciamento de informações necessárias para o seu aprendizado, usam de habilidades de caráter científico e apresentam resultados de suas investigações.

As teorias de aprendizagem apresentadas, dentre diversas outras, são as que mais se adaptam à realidade estudada, apesar de ficar evidente sempre a figura do professor como o condutor ou mediador do processo de ensino e aprendizagem. Considerando o objeto aqui

investigado, por ser uma realidade relacionada a um produto de um trabalho e que a aprendizagem é a base para a geração de conhecimento, para que uma ação seja executada e, por conseguinte um produto seja desenvolvido, no caso o SLCA, não existe a figura de um professor, tutor ou alguém que desempenhe um papel semelhante. De qualquer forma, os sujeitos participantes das comunidades possuem características especiais, que fazem com que a aprendizagem, a geração de conhecimento e aplicação desse conhecimento ocorra, de forma eficaz, para a efetiva solução dos problemas apresentados nas situações reais em que estão inseridos.

1.6 Comunidades virtuais

Um dos inúmeros resultados da revolução causada pela Internet foi o surgimento das **comunidades virtuais**. Tais comunidades são formadas por um conjunto de pessoas chamadas de **membros**, que interagem entre si baseadas em um interesse comum, independentemente da sua localização geográfica (YOO; SUH; LEE, 2002).

Martin (1999) apontou as comunidades virtuais como um dos sete fatores que irão mudar a forma como os negócios serão feitos no século XXI. Por outro lado, Woods (1999) argumenta que as comunidades virtuais serão um dos fatores-chave para o sucesso de negócios baseados em B2B⁴³.

O termo **comunidade virtual** foi usado pela primeira vez por Rheingold (1992, p.5), que assim a definiu:

É uma agregação social que emerge no ambiente da rede Internet quando uma quantidade suficiente de pessoas torna pública uma discussão por um longo período de tempo, demonstrando o sentimento humano, com o objetivo de formar uma rede de relacionamento pessoal no ciberespaço.

⁴³ B2B é uma sigla em inglês que significa *business to business*. É uma nova forma de fazer negócio em que fornecedor e clientes se comunicam por meio de sistemas automatizados, via Internet.

Hagell III e Armstrong (1997) descrevem as comunidades virtuais como uma diferente forma de um grupo de consumidores unirem-se por um interesse comum.

Willians e Cothrel (2000) as definem como um grupo de pessoas engajadas em um tipo de interação online de muitos-para-muitos.

Baseando-se nas definições acima apresentadas pode-se concluir que uma comunidade virtual possui quatro elementos-chave:

1. **Objetivo comum:**

É a principal motivação e o interesse pelo qual as pessoas se reúnem para formar uma comunidade virtual;

2. **Interatividade:**

Existem basicamente dois níveis de interação: um é o processo de relacionamento entre os técnicos que atuam no suporte do ambiente tecnológico com os membros, e o outro é o relacionamento entre os próprios membros da comunidade;

3. **Continuidade:**

Refere-se ao tempo de permanência da interação em atividade entre os membros da comunidade, quanto aos temas e assuntos objeto de discussão;

4. **Ciberespaço:**

Inclui todos os aspectos e recursos relacionados à infraestrutura tecnológica e logística que ampara e suporta todo o processo operacional da comunidade virtual.

Os elementos essenciais de uma comunidade virtual são representados por meio de uma figura ilustrativa, logo abaixo:

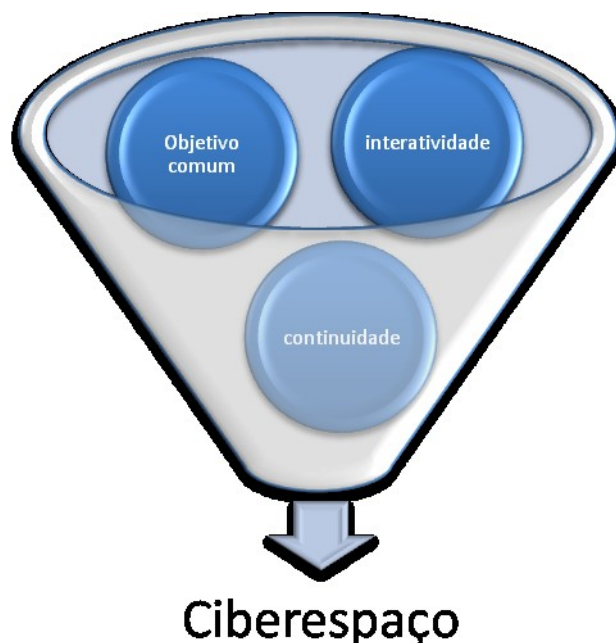


Figura 2- Elementos das comunidades virtuais

Apesar da ainda restrita pesquisa empírica sobre esse assunto, Hagel III e Armstrong (1997) argumentam que as pessoas se agregam em uma comunidade virtual com o objetivo de preencher as suas necessidades e encontrar informações, relacionamentos, conhecimento e até mesmo fantasias. Assim as comunidades virtuais devem satisfazer tais necessidades pela disponibilização de conteúdos, comunicação e comercialização de produtos e serviços, quando for o caso.

Kim (2000) focaliza as suas pesquisas na parte operacional e de *design* das comunidades virtuais, sugerindo nove princípios para a construção de comunidades virtuais, observadas por meio da sua experiência, que são os seguintes:

1. Definir e articular o **propósito** da comunidade;
2. Construir um flexível e extensivo **local** para a agregação de pessoas;
3. Criar um significativo e envolvente ambiente para atrair membros com necessidade e **perfis** afins;

4. Prover o máximo de informações sobre os diferentes **papéis** que podem ser desempenhados pelos membros da comunidade;
5. Definir programas de promoção das **lideranças** que podem emergir no âmbito da comunidade;
6. Definir um consistente regulamento com as **regras** de conduta e netiqueta, de forma clara e objetiva;
7. Organizar e promover **eventos** cíclicos entre os membros da comunidade;
8. Integrar a comunidade com o **mundo real**;
9. Facilitar a criação de **subgrupos** por parte dos membros da comunidade.

O quadro 3 demonstra a relação entre os princípios descritos e as estratégias para a sua consecução:

Quadro 3 – Princípios e estratégias para a formação de comunidades virtuais

<i>Princípios</i>	<i>Estratégias</i>
<i>Propósitos</i>	<ul style="list-style-type: none"> • Construir a declaração da missão. • Criar um ambiente com forte personalidade e identidade da comunidade.
<i>Local</i>	<ul style="list-style-type: none"> • Prover uma clara visão geral do ambiente da comunidade. • Incluir as características de comunicação e interação bem como as ferramentas disponíveis.
<i>Perfil</i>	<ul style="list-style-type: none"> • Disponibilizar o perfil dos membros de forma clara e divertida. • Manter o perfil dos membros atualizado e envolvente.
<i>Papel</i>	<ul style="list-style-type: none"> • Criar mecanismos para oferecer alguns privilégios para os membros mais ativos e participantes. • Criar mecanismos para identificar e recrutar líderes e mentores entre os membros da comunidade.
<i>Liderança</i>	<ul style="list-style-type: none"> • Preparar infraestrutura para um rápido crescimento. • Oferecer suporte online aos membros da comunidade.

<i>Regra</i>	<ul style="list-style-type: none"> • Criar código de conduta. • Definir critérios para o aprimoramento do código de conduta.
<i>Evento</i>	<ul style="list-style-type: none"> • Promover eventos regularmente. • Fazer frequentes pesquisas entre os membros da comunidade.
<i>Mundo real</i>	<ul style="list-style-type: none"> • Promover eventos que promovam a identidade social dos membros. • Reconhecer os promotores de eventos da própria comunidade.
<i>Subgrupo</i>	<ul style="list-style-type: none"> • Prover condições que facilitem a formação de pequenos grupos dentro da comunidade. • Criar contextos para a formação de subgrupos.

Williams e Cothrel (2000) apontam três elementos-chave para a sustentabilidade de uma comunidade virtual:

1. Desenvolvimento de novos membros:

A comunidade necessita de um número suficiente de membros para que seja criada uma massa crítica, com o intuito de manter os propósitos da comunidade ativos e vigorosos, a fim de que a atenção dos membros seja sempre despertada;

2. Gestão do patrimônio da comunidade:

O patrimônio intangível de uma comunidade virtual geralmente é formado pelo conhecimento gerado pelas interações entre os membros, conteúdos diversos, relatos de experiências e alianças com outras comunidades. Quanto ao patrimônio tangível, é a infraestrutura tecnológica que suporta a operação da comunidade, e corresponde a um conjunto de hardware, software, redes de comunicação, etc.

3. Relacionamento:

É a capacidade de a comunidade executar estratégias para atrair visitantes e fazê-los novos membros, bem como a capacidade de reter os que já fazem parte da comunidade.

Nesse sentido, os autores Hagel III e Armstrong (1997) desenvolveram conceitos básicos enquanto Kim (2000) e Willians e Cothrel (2000) propuseram questões de ordem mais operacional e concreta sobre as comunidades virtuais.

1.7 Comunidades virtuais de aprendizagem auto-organizadas

As comunidades virtuais de aprendizagem auto-organizadas - CVAO podem também ser chamadas de sistemas sociais auto-organizados em ambiente virtual – *online self-organizing social systems* – OSSOS. A estrutura dessas comunidades permite que uma significativa quantidade de indivíduos possa se auto-organizar, de maneira altamente descentralizada e distribuída no espaço e no tempo, buscando conhecimento e aprendizado para o equacionamento e solução de problemas, para questões de inter-relacionamento e processos que visem à pesquisa e aprendizagem.

1.7.1 O fenômeno da auto-organização

Aparentemente é pouco provável imaginar que pessoas possam se unir para estudar, aprender ou solucionar problemas, sem necessariamente ter alguma liderança ou alguma ação de autoridade. Ainda mais quando se trata de formar uma comunidade em ambiente virtual.

Maeterlinck (1927) aborda o assunto, sem se reportar a pessoas, mas sim às formigas brancas. Muitas espécies de formigas, abelhas e outros tipos de insetos armazenam recursos e

suprimentos com o objetivo de prover as necessidades da sua comunidade, no tempo e lugar certos, selecionando os alimentos de melhor qualidade, construindo ninhos, abrigos e resolvendo uma diversidade enorme de problemas de natureza geométrica, econômica e de engenharia.

Os modelos auto-organizados têm sido aplicados às comunidades humanas há muitas décadas. Jacobs (1961) os tem utilizado para o planejamento urbano, baseando-se na similaridade de como os insetos se movimentam e comparando-os com a movimentação das pessoas nas cidades, trafegando pelas ruas, calçadas e residindo em bairros ou zonas especializadas tais como comércio, serviços, etc.

Diversos pesquisadores têm dado continuidade, de forma frutífera, à aplicação dos modelos auto-organizados para os sistemas de organização humana na área da economia (KRUGMAN, 1996). Mais recentemente, Eriksson e Wulf (1999) começam a explorar o vínculo existente entre os sistemas auto-organizados e o trabalho colaborativo mediado por computador. Wulf (1999) tem examinado os modos pelos quais atuam as equipes virtuais de forma auto-organizada.

As recentes pesquisas sobre o assunto trazem à tona o ponto em que o potencial do processo auto-organizacional do sistema social humano tem sido reconhecido e documentado, e muitas pesquisas estão sendo iniciadas, levando-se em consideração as tecnologias da informação e da comunicação. Tais iniciativas facilitam consideravelmente o desenvolvimento das atividades auto-organizadas por indivíduos que estão dispersos e distantes geograficamente.

1.7.2 Sistemas sociais online auto-organizados

Estes sistemas são facilitados por infraestrutura de software, hardware e redes de comunicação, ou seja, são possibilitados pelo suporte de recursos disponibilizados pelas tecnologias da informação e comunicação. Geralmente esses sistemas são baseados na tecnologia *web* e caracterizados por elevado grau de descentralização do seu gerenciamento. Os *blogs* são exemplos típicos desses sistemas. Dia após dia, notícias e comentários são postados, acumulando dessa forma conteúdo expressivo de informações e de conhecimento. A validação da sua qualidade emerge naturalmente do grupo de participantes. Esse processo pode variar desde um controle centralizado que valida o conteúdo, previamente, até um nível totalmente sem nenhum controle prévio à postagem do conteúdo, permitindo que o grupo se manifeste livremente nessas condições.

Mesmo sem a intenção prévia, esse ambiente acaba se tornando um ambiente de aprendizagem colaborativa para a solução de problemas.

1.8 *Mediação, dialogicidade, artefatos culturais e vozes da mente: conceitos que se entrecruzam na perspectiva de Vigotsky, Bakhtin, Wertsch e Cole*

1.8.1 Vigotsky e os conceitos de mediação e operações de pensamento

A teoria sociocultural de Vigotsky⁴⁴ enfatiza a atividade humana enquanto processo e resultado das vivências em atividades socioculturais compartilhadas, mais do que nas questões de conhecimento e apropriação da cultura social. Compreende as práticas de aprendizagem como atividade sempre situada em um contexto de cultura, de relações, de

⁴⁴ As obras de Vigotsky, a partir da década de 1980, ganham importância na área da educação, fomentada pelo desenvolvimento da teoria histórico-cultural da atividade, que tem suas origens nos trabalhos dos psicólogos russos na tradição de Vigotsky. Descreve os processos através dos quais o conhecimento é construído como resultado da experiência pessoal e subjetiva de uma atividade. Considera que a atividade precede o conhecimento, que é mediada por signos culturais (linguagem, utensílios, tecnologias, meios de comunicação, convenções, etc.), e que as próprias tecnologias são artefatos de atividade prática. A medida que esses artefatos mudam, muda também a atividade e, com ela, a consciência dos participantes, num continuum que envolve ciclos de aprendizagem (WERTSCH, 1991).

conhecimento. Enfatiza que a inteligência humana provém da nossa sociedade ou cultura, e que ocorre em primeiro lugar pela interação com o ambiente social (ponto de vista interpessoal).

A aprendizagem resulta da interação sujeito-objeto, em que a ação do sujeito sobre o meio é socialmente mediada, atribuindo-se peso significativo à cultura e às relações sociais. A atividade do sujeito supõe a ação entre sujeitos, no sentido de uma relação com o outro, com seus parceiros. Mais especificamente, as funções mentais superiores (linguagem, atenção voluntária, memória, abstração, percepção, capacidade de comparar, analisar, etc.) são ações interiorizadas de algo socialmente mediado, com base na cultura constituída (DANIELS, 2003).

Cole e Scribner (2007) em prefácio à obra de Vigotsky, *A Formação Social da Mente*, ressaltam que nessa obra há uma influência do materialismo dialético, afirmando que ela explora o conceito de ferramenta de um modo que encontra antecedentes diretos em Friedrich Engels, sobre o papel crítico do trabalho humano e o uso de instrumentos como os meios pelos quais o homem modifica a natureza e, ao fazê-lo, transforma a si mesmo. “O instrumento simboliza especificamente a atividade humana, a transformação da natureza pelo homem: a produção.”, ou seja, “a especialização da mão – que implica o instrumento, e o instrumento implica a atividade humana específica, a reação transformadora do homem sobre a natureza” (ENGELS, 1979 *apud* COLE; SCRIBNER, 2007, p.7).

Os mesmos autores afirmam que a teoria marxista da sociedade (materialismo histórico) teve papel fundamental no pensamento de Vigotsky. De acordo com Marx (1979), mudanças históricas na sociedade e na vida material produzem mudanças no homem (consciência e comportamento). Portanto, esse pensamento incorpora a compreensão do papel ativo da história no desenvolvimento psicológico humano.

Os aspectos particulares da existência social humana refletem-se na cognição humana, ou seja, um indivíduo tem a capacidade de expressar e compartilhar com outros membros de seu grupo social o entendimento que ele tem da experiência comum ao grupo. Tendo em vista que os instrumentos de trabalho sofrem mudanças no decorrer do tempo, também os instrumentos do pensamento se transformam historicamente. Ainda nessa direção Vigotsky (2007, p.167) argumenta que o

efeito do uso de instrumentos sobre os homens é fundamental não apenas porque os ajuda a se relacionar mais eficazmente com seu ambiente como também (...) pelos efeitos que o uso de instrumentos tem sobre as relações internas e funcionais no interior do cérebro humano. (...) As estruturas sociais como as estruturas mentais têm de fato raízes históricas muito definidas, sendo produtos bem específicos de níveis determinados do desenvolvimento dos instrumentos.

O conceito de mediação é fundamental no estudo das CVAO, pois para Vigotsky (2007), a utilização de artefatos, que são social e culturalmente construídos, tem efeitos sobre a mente do utilizador e sobre o contexto. A inclusão de uma nova ferramenta, ela própria portadora de uma carga cultural anterior que conduziu à sua concepção e construção, num processo de comportamento, introduz diversas funções relacionadas com o uso dela e com o seu controle. Além disso, faz abolir, por terem passado a ser desnecessários, alguns processos naturais cuja caducidade resulta da utilização da ferramenta, enquanto altera todo o conjunto dos traços individuais (intensidade, duração, sequência, etc.) de todos os processos mentais que fazem parte do ato instrumental, substituindo algumas funções por outras. Assim, a utilização de artefatos deve ser reconhecida como transformadora do funcionamento da mente, e não apenas como meio de facilitar processos mentais já existentes.

Cole e Scribner (2007, p.7), a propósito do conceito de mediação explicam:

De maneira brilhante, Vigotsky estendeu esse conceito de mediação na interação homem-ambiente pelo uso de instrumentos ao uso de signos. Os sistemas de signos (a linguagem, a escrita, o sistema de números) assim como o sistema de instrumentos, são criados pelas sociedades ao longo do curso da história humana e mudam a forma social e o nível de seu

desenvolvimento cultural. Vigotsky acreditava que a internalização dos sistemas de signos produzidos, culturalmente provoca transformações comportamentais e estabelece um elo entre as formas iniciais e tardias do desenvolvimento individual. Assim, para Vigotsky, na melhor tradição de Marx e Engels, o mecanismo de mudança individual ao longo do desenvolvimento tem sua raiz na sociedade e na cultura.

O desenvolvimento da estrutura cognitiva no organismo é concebido como produto das modalidades de interação entre o organismo e o seu meio ambiente: a exposição direta às fontes de estímulo e de aprendizagem mediada.

A experiência de aprendizagem mediada acontece quando os estímulos enviados pelo ambiente são transformados por um agente mediador. Esse agente mediador guiado por suas intenções, sua cultura e seu investimento emocional, seleciona e organiza o mundo dos estímulos. Os três componentes da interação mediada são: o organismo receptor, o estímulo e o mediador. O efeito da experiência de aprendizagem mediada é a criação nos receptores de uma disposição direta aos estímulos. Basicamente, tudo isso pode ser mais bem entendido como mediar para ensinar a aprender.

Uma interação que leva à aprendizagem mediada, necessariamente, inclui uma intenção por parte do mediador (docente) de transcender as necessidades imediatas ou as preocupações do receptor a ir além do aqui e agora e do tempo e espaço. Qualquer antecipação de resultados é uma construção interna da realidade, que depende de uma representação e também de um pensamento inferencial por parte do educando.

Vigotsky (2007) distingue duas classes de instrumentos mediadores, em razão do tipo de atividade que possibilita: a ferramenta e os símbolos. Uma ferramenta modifica o entorno materialmente, enquanto os símbolos são um constituinte da cultura e atuam como mediadores em nossas ações. Existem muitos sistemas de símbolos que nos permitem atuar sobre a realidade, tais como: a linguagem, utensílios, tecnologias, meios de comunicação, convenções, os sistemas de mediação, a cronologia, a aritmética, etc. Esse conceito de

atividade mediada foi investido do mais amplo significado em Hegel, que viu nele um aspecto característico da razão humana. Para o autor,

(...) a razão é tão engenhosa quanto poderosa. A sua engenhosidade consiste principalmente em sua atividade mediadora, a qual, fazendo com que os objetos ajam e reajam uns sobre os outros, respeitando sua própria natureza e, assim, sem nenhuma interferência direta no processo, realiza as intenções da razão (HEGEL, 1995, *apud* COLE; SCRIBNER, 2007, p.54).

Essa análise fornece base sólida para que se assinale o uso de signos à categoria de atividade mediada, uma vez que os homens são afetados em seu comportamento pelos signos, pois a atividade cognitiva não se limita apenas ao uso de instrumentos ou signos. No plano lógico da relação entre os dois conceitos (signo e instrumentos) há linhas divergentes da atividade mediada.

A função do instrumento é servir como condutor da influência humana sobre o objeto da atividade e constitui-se um meio pelo qual a atividade humana externa é dirigida para o controle e domínio da natureza. Além disso, ele é orientado externamente e deve, necessariamente, conduzir para a mudança nos objetos. O signo, por outro lado, orientado internamente, constitui-se em um meio de atividade dirigido para o controle do próprio indivíduo e não modifica em nada o objeto da operação psicológica. Essas atividades são tão diferentes que a natureza dos meios por elas utilizados não pode ser a mesma. Diferente da ferramenta, os sinais ou símbolos não modificam materialmente o estímulo, mas sim modificam a pessoa que o utiliza como mediador e, de forma definitiva, atua sobre a interação de uma pessoa com o seu entorno. Os meios de ajudar a execução são: modelagem, manejo de contingências, instrução, perguntas e estruturação cognitiva. Modelagem, manejo de contingências e retroalimentação são os principais mecanismos para ajudar os aprendizes através das Zonas de Desenvolvimento Proximal (ZDP). Os meios de ajuda na execução especificamente linguísticos (sinais ou símbolos) são: instruir, perguntar e estruturar cognitivamente.

O uso dos meios artificiais – a transição para a atividade mediada – modifica, basicamente, todas as operações psicológicas, assim como o uso de instrumentos amplia infinitamente a gama de atividades em cujo interior as novas funções psicológicas podem operar. “Nesse contexto, podemos assinalar que a combinação entre os instrumentos e o signo na atividade psicológica corresponde à internalização de formas culturais de comportamentos, ou seja, envolve a reconstrução da atividade psicológica tendo como base as operações com signo” (VIGOTSKY, 2007, p. 58).

As operações da atividade mediada passam por diversas fases. A instrução (por exemplo) requer ações específicas. Perguntar solicita especificamente respostas linguísticas. A estruturação cognitiva solicita uma resposta específica, mas também provê uma estrutura para organizar os elementos, que se relacionam entre si. A estruturação cognitiva se refere a prover uma estrutura para o pensamento e a ação. Pode ser uma estrutura de crenças, de operações mentais ou de compreensão, que abrange ações de avaliação, agrupamentos, sequenciamento de atividades que envolvem percepção, memória e atividades. Enquanto para o comportamento mediacional, os estímulos (E) e as respostas mediadoras (R) são, segundo o princípio da correspondência, meras cópias não observáveis dos estímulos e respostas externas, os mediadores vigotskianos não são réplicas das associações E-R externas, nem um conector a mais das cadeias associativas. Os mediadores são instrumentos que transformam a realidade em vez de imitá-las. A sua função não é adaptar-se passivamente às condições do meio, mas sim modificá-los ativamente.

O conhecimento não é um objeto que se passa de um para outro, mas sim algo que se constrói por meio de operações e habilidades cognitivas que se intercambiam no contexto e na interação social. O desenvolvimento intelectual do indivíduo não pode entender-se como independente do meio social em que está imersa a pessoa, e o desenvolvimento das funções

psicológicas superiores desenvolve-se no plano social e depois no nível individual (VIGOTSKY, 2007).

A transmissão e aquisição de conhecimentos e padrões culturais são possíveis quando a interação – plano *interpsicológico* – chega a internalização – plano *intrapsicológico*. As relações entre os indivíduos estão na base de todas as funções superiores. A natureza cultural, histórica e institucionalmente situada e contextualmente específica das funções psicológicas humanas, decorre do fato de os artefatos que interferem nessas funções psicológicas serem igualmente situados e dependentes do contexto em que decorre a sua utilização. O complexo processo de passagem do interpessoal para o intrapessoal é denominado de *internalização*, ou seja, *a reconstrução interna de uma operação externa* (VIGOTSKY, 2007, p.56). Nessa perspectiva, o autor afirma que todas as funções psicológicas superiores são relações sociais internalizadas, pois esse processo consiste numa série de transformações, a saber:

1. *Uma operação que inicialmente representa uma atividade externa é reconstruída e começa a ocorrer internamente*, ou seja, para o desenvolvimento dos processos mentais superiores é necessária a transformação da atividade que utiliza signos, pelo desenvolvimento da inteligência prática, da atenção voluntária e da memória;
2. *Um processo interpessoal é transformado num processo intrapessoal*. Qualquer função presente no desenvolvimento cultural da criança aparece duas vezes e em dois planos diferentes. Em primeiro lugar surge no plano social, para que seja feito logo no plano psicológico. Em princípio surge entre as pessoas como uma categoria interpsicológica, para logo aparecer na criança, que é o sujeito da aprendizagem como uma categoria intrapsicológica;
3. *A transformação de um processo interpessoal num processo intrapessoal é o resultado de uma longa série de eventos ocorridos ao longo do desenvolvimento*.

Entretanto, esses eventos adquirem o caráter de processos internos como resultados de um desenvolvimento prolongado (VIGOTSKY, 2007, p.57-58).

Vigotsky recusa os enfoques que reduzem a psicologia e a aprendizagem a uma simples acumulação de reflexos e associações entre estímulos e respostas. Existem características especificamente humanas que não são reduzidas a associações tais como a consciência e a linguagem e que não podem ser alheias à Psicologia. O autor não nega a importância da aprendizagem associativa (à diferença de outras posições de Gestalt e Piaget), porém a considera insuficiente e procurou uma abordagem abrangente que possibilitasse “a descrição e a explicação das funções psicológicas superiores, em termos aceitáveis para as ciências naturais” (COLE; SCRIBNER, 2007, p.14).

Outro conceito da teoria de Vigotsky (2007) é a ideia de que o potencial para o desenvolvimento cognitivo está limitado a uma determinada "zona de desenvolvimento proximal" (ZDP). Trata-se de um estágio do processo de aprendizagem em que o estudante consegue desenvolver as suas atividades de aprendizagem de forma individual ou com a colaboração do professor, isto é, pode ser dispensada a mediação do professor. Na ótica de Vigotsky, esse “fazer em colaboração” não anula, mas destaca a participação criadora do aprendiz e serve para medir o seu nível de desenvolvimento intelectual, sua capacidade de discernimento, de tomar iniciativa, de começar a fazer sozinho o que antes só fazia acompanhado, sendo, ainda, um valiosíssimo critério de verificação e eficácia do processo de ensino-aprendizagem.

Bezerra (2001) define esse conceito como a distância entre o nível real e atual de conhecimentos de uma pessoa, determinado pela resolução independente de problemas e o nível de desenvolvimento potencial determinado pela resolução de problemas sob a orientação de adultos ou em colaboração com companheiros mais capacitados.

A ZDP pode compor-se de diferentes níveis de experiência individual (alunos e professores) e inclui artefatos tais como livros, programas para computadores e materiais de caráter científico, etc. A finalidade principal da ZDP é a de oferecer suporte à aprendizagem intencional.

Em suma:

1. Em contraste com Piaget, Vigotsky não fala de assimilação mas sim de apropriação (em termos de contextos culturais);
2. Influência predominante do meio cultural: refere-se à origem social dos processos psicológicos superiores (linguagem oral, jogo simbólico, leitura e escrita). A maioria deles não se forma sem a intervenção educativa.
3. Mediação Semiótica: é o papel central dos instrumentos de mediação na constituição do psiquismo: ferramentas e símbolos. As ferramentas estão orientadas para os objetos físicos, enquanto os símbolos permitem organizar o pensamento, ou seja, são orientados para o interior e o exterior de um sujeito, produzindo mudanças nos outros.
4. Zona de Desenvolvimento Proximal (ZDP): é a distância entre o nível de desenvolvimento, determinado pela capacidade de resolver independentemente um problema e o nível de desenvolvimento potencial, determinado pela resolução de um problema sob a orientação de um adulto ou em colaboração com um par mais capacitado. Uma vez que vê o aprendizado como um processo profundamente social, enfatiza o diálogo e as diversas funções da linguagem na instrução e no desenvolvimento cognitivo mediado. Segundo Cole e Scribner (2007, p.164), para implementar o conceito de zona de desenvolvimento proximal na instrução, Vigotsky ressalta a colaboração de psicólogos e educadores na análise dos processos internos de desenvolvimento, os quais contribuem para o aprendizado.

A facilidade com que o conceito ZDP se presta para o uso com fins práticos é decorrente das condições em que o conceito foi formulado originalmente por Vigotsky: como alternativa frente às provas padronizadas no campo educativo. Essas provas conduzem ao enfoque da atenção nas habilidades ou capacidades já constituídas e dominadas pelo educando, sem dizer nada sobre o que está em processo de aprendizagem, mas que por um só momento pode se realizar conjuntamente com outra pessoa que é mais especializada na tarefa ou no problema em questão. Este último é o que mais interessava a Vigotsky e o que pensava que era o ponto de partida do esforço educativo. Em qualquer caso, essa mesma finalidade prática, por assim dizer, do conceito, tem impedido que uma análise mais acurada dos problemas esteja na base da sua elaboração.

A concepção de desenvolvimento humano em Vigotsky tem contribuído para a investigação psicológica contemporânea. Embora o autor tenha dedicado a maior parte de seus esforços ao estudo da criança, sua ênfase no estudo do desenvolvimento contribui para desvendar os processos humanos complexos. Portanto, sua preocupação está voltada para as implicações da atividade humana, uma vez que esta transforma tanto a natureza como a sociedade. Nesse aspecto, podem-se encontrar elementos fecundos na obra de Vigotsky (uma teoria abstrata não é suficiente para captar os momentos críticos da transformação), seja em relação à gênese dos conceitos científicos seja em relação ao pensamento e linguagem (COLE; SCRIBNER, 2007).

Essas formulações teóricas permitem aproximações com as CVAO no que diz respeito aos instrumentos de trabalho e do pensamento (ex: questão problematizadora, resolução de problemas, motivação, cooperação, etc.), bem como os conceitos de mediação e zonas de desenvolvimento.

A figura 3, logo a seguir, demonstra de forma ilustrada e esquemática, a identificação dos dois instrumentos mediadores: a ferramenta e os símbolos (signos) (VIGOTSKY, 2007).



Figura 3 - Demonstração esquemática da atividade mediada

Portanto, a atividade humana é mediada pelo uso de ferramentas, que estão para a evolução cultural como os genes para evolução biológica (COLE, 1994). As ferramentas são criadas e modificadas pelos seres humanos como forma de se ligarem ao mundo real e de regularem o seu comportamento e as suas interações com o mundo e com os outros. Cada indivíduo alcança a consciência pela atividade mediada por ferramentas (meio pelo qual a atividade externa humana domina a natureza), as quais unem a mente com o mundo real dos objetos e dos acontecimentos. Os signos e ou símbolos (meio de atividade interna, empenhada no domínio do próprio sujeito) não provocam alteração no objeto da operação psicológica, pois são orientados internamente.

Alguns aspectos da teoria de Vigotsky são fundamentais para entendermos os processos de funcionamento das CVAO quanto à interação, mediação, uso dos sistemas de signos e ZDP. A importância da interação prende-se ao fato de que as mais elevadas funções mentais do indivíduo emergem de fenômenos sociais. Nas CVAO a interatividade e a continuidade (WILLIANS; COTHREL, 2000) são elementos-chave para o seu funcionamento. A mediação pode ser representada por seus membros, ativos e responsáveis (KIM, 2000), que se constituem em massa crítica, com o intuito de manter os propósitos da comunidade em *um tipo de interação de muitos – para - muitos* (WILLIANS; COTHREL, 2000). O uso de sistemas e signos pode ser representado pelo ciberespaço (HAGEL III;

ARMSTRONG, 1997), pela infraestrutura tecnológica e pelo conhecimento gerado pelas interações entre os membros. (KIM, 2000). A aproximação sociocultural de Vigotsky à aprendizagem e, muito em particular, os conceitos de mediação e ZDP podem contribuir para o processo de apreensão e internalização de conceitos discutidos e fomentados pelas CVAO, tendo em vista que operam com instrumentos (externo) e signos (interno). Além disso, essas formulações teóricas permitem aproximações com as CVAO relativamente aos instrumentos de trabalho e de pensamento (ex: questão problematizadora, resolução de problemas, motivação, cooperação, etc.).

1.8.2 A dialogicidade em Bakhtin

Os processos mediáticos e culturais advindos das redes de comunicação e informação têm contribuído para novas formas de interação e estilos de linguagem, ou seja, um novo *ethos* caracterizado por uma linguagem verbal e não verbal utilizada por usuários, principalmente jovens (em sua maioria) e adultos em ambientes virtuais. Essas experiências e as respectivas aprendizagens revelam novas relações entre o campo oral e escrito no gênero discursivo, seja pela aproximação ao texto oral, abreviação, seja pela associação da escrita a aspectos emocionais ou adaptada às necessidades ou características desse tipo de comunicação (SIQUEIRA FILHO; BORTOLETO, 2006).

Esse novo *ethos* pressupõe uma linguagem peculiar, na qual encontramos em Bakhtin (1995, p.279) ⁴⁵ a forma como determinados sujeitos sócio-históricos apropriam-se e utilizam-se dessas linguagens. Para o autor

⁴⁵ Mikhail Bakhtin, em sua obra *Marxismo e Filosofia da Linguagem* (1929), utiliza a fundamentação teórico-metodológica a partir da matriz marxista e elabora uma abordagem antropológica e linguística que epistemologicamente estabelece para as ciências sociais um campo interfático semiótico fundamentado no dialogismo e na alteridade que se objetiva na produção de sentidos dos discursos e nas formas de sua construção (FARACO; TEZZA; CASTRO, 1996; HENRIQUES, 2007).

A utilização da língua efetua-se em forma de enunciados (orais e escritos), concretos e únicos, que emanam dos integrantes duma ou doutra esfera da atividade humana. O enunciado reflete as condições específicas e as finalidades de cada uma dessas esferas, não por seu conteúdo temático e por seu estilo verbal, ou seja, pela seleção operada nos recursos da língua (...), mas, sobretudo, por sua construção composicional. Estes três elementos (conteúdo temático, estilo e construção composicional) fundem-se indissolivelmente no todo do enunciado, e todos eles são marcados pela especificidade de uma esfera de comunicação. Qualquer enunciado considerado isoladamente é, claro, individual, mas cada esfera de utilização da língua elabora seus tipos relativamente estáveis de enunciados, sendo isso que denominamos de gêneros de discurso. (...) Um gênero discursivo não é uma forma de linguagem, mas sim uma forma típica de enunciado; como tal, o gênero também inclui uma determinada classe típica de expressão que é inerente.

A noção do enunciado está vinculada à voz da personalidade do sujeito que fala. Sabe-se quem fala e/ou para quem fala, devido ao gênero discursivo que ele emprega. Um enunciado só pode existir se é produzido por uma voz, pois a *enunciação* é o produto da interação de dois indivíduos socialmente organizados. Para o autor, a palavra é o signo ideológico por excelência. Para falar, de acordo com Bakhtin (1995, p.301) utiliza-se sempre dos “gêneros do discurso, em outras palavras, todos os nossos enunciados dispõem de uma forma padrão e relativamente estável de estruturação de um todo. Possuímos um rico repertório dos gêneros do discurso orais (e escritos)”. O enunciado estabelece uma relação complexa entre o campo formal e não formal, entre os distintos gêneros do discurso implicando um processo em que outros enunciados entram em contato e também se confrontam.

De acordo com Bakhtin (1995, p.285), são três propriedades fundamentais ou características do enunciado: limites, finalização e forma genérica.

(...) Apesar da variedade que pode apresentar os enunciados em função de sua longitude, seu conteúdo e sua estrutura de composição como unidade da comunicação verbal têm em comuns traços estruturais, sobretudo, limites que são muito claramente marcados... Os limites de cada enunciado concreto como unidade da comunicação verbal estão determinados por uma mudança dos sujeitos que se comunicam, é dizer, uma mudança do emissor. Todo enunciado – desde uma curta réplica (mesmo que de uma só palavra) em um diálogo cotidiano, até um extenso artigo científico – tem, por assim dizer,

um início absoluto e um fim absoluto: o seu início está precedido pelos enunciados dos outros, e o seu fim está seguido de resposta dos outros (a resposta pode ser a compreensão ativa do outro, ou finalmente, uma ação baseada nesta compreensão). A pessoa que comunica finaliza o seu enunciado para deixar espaço para o outro, ou para ceder lugar para a resposta referente à compreensão ativa do outro. O enunciado não é uma unidade convencional, mas sim uma unidade real, claramente definida pela mudança do sujeito que comunica.

O autor adverte, como primeira característica, que a forma particular que adota esse traço estrutural varia com os diferentes tipos de discurso, ou seja, em sua natureza, nas diferentes formas das esferas heterogêneas da vida e da atividade humana, dependendo das funções da linguagem, das condições e situações da comunicação. Entre a lista de tipos de pares do enunciado-réplica que foi elaborada, incluem pergunta e resposta, afirmação e objeção, afirmação e um parecer favorável, sugestão e aceitação, ordem e execução.

A segunda característica de um enunciado como unidade da comunicação verbal é a sua finalização. Essa característica está ligada à primeira, que pode ser considerada como “o lado interior da mudança dos sujeitos que comunicam”. Tal mudança (de sujeitos ou vozes) pode ocorrer somente porque o sujeito disse (ou escreveu) tudo o que desejava dizer em um momento determinado ou sob determinadas circunstâncias. Normalmente, quando um sujeito escuta ou lê, sente claramente o final do enunciado, como se ouvisse o sujeito emissor concluindo o seu pensamento. Essa finalização é específica e determinada por critérios especiais que caracterizaram as diferentes linguagens sociais e gêneros discursivos ou esferas da atividade humana.

A terceira e mais importante característica do enunciado é a sua forma genérica. Segundo Bakhtin (1995), a escolha de um gênero discursivo está determinada pela natureza específica de uma esfera dada a partir da comunicação verbal, por considerações semânticas (temáticas), pela situação concreta da comunicação verbal e pela composição pessoal dos participantes. O autor, como sempre, enfatiza que existem limites estritos no grau de individualização que pode apresentar o enunciado: quando o plano discursivo do emissor da

mensagem, com toda a sua individualização e sua subjetividade, aplica-se e adapta-se a um gênero que fora eleito, adota a sua forma e desenvolve-se dentro de determinada forma genérica.

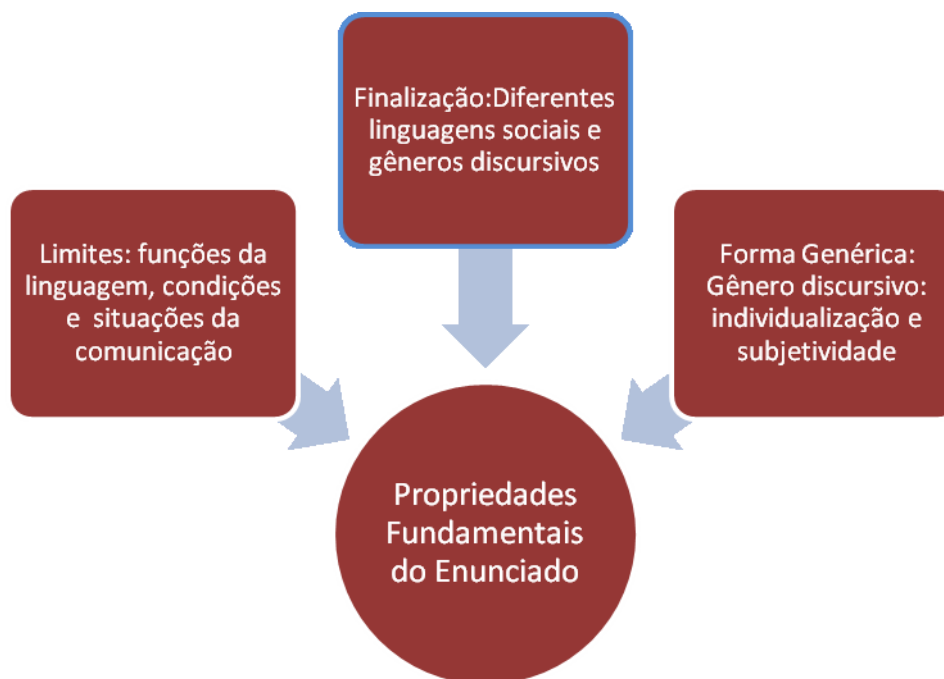


Figura 4 - Dialogicidade em Bakhtin: enunciado

Fonte: Adaptado de Bakhtin (1995)

Em sua discussão sobre o papel dos gêneros discursivos na conformação dos enunciados, o autor afirma que um aspecto muito importante do enunciado é a sua relação com o emissor da mensagem ou enunciado (o autor mesmo) e com todos os outros que participam da comunicação verbal. Diante desse contexto, o autor identifica duas questões básicas:

1. o feito de que cada enunciado se caracteriza principalmente por determinado conteúdo semântico referencial; e

2. que deve ser levado em conta o aspecto relacionado à forma de expressão e emoção de cada emissor da mensagem no que concerne ao conteúdo semântico referencial.

O *conteúdo semântico referencial* de um enunciado pode variar, como, por exemplo, sobre aspectos das relações comerciais de certo país, discussões sobre a teoria da relatividade ou até mesmo sobre uma peça de quebra-cabeça de criança. As técnicas semânticas tradicionais em linguística interessam-se por esses aspectos, mas o foco do autor é nos enunciados, e não nas orações ou outras classes de símbolos. Assim, para o autor o enunciado é uma conexão entre a cadeia de comunicação verbal e o conteúdo semântico referencial (tema) que dependerá do seu lugar em relação aos outros enunciados, enquanto o conteúdo semântico referencial (tema) de uma oração é dependente de tais fatores contextuais (GRILLO, 2006).

Na linguagem utilizada nos fóruns, há uma linguagem própria adaptada às necessidades e características dessa comunicação, bem como a efetivação de experiências centradas na educação não formal, devido à circularidade de informações disponibilizada e possibilidades de interação.

Levy (1994, p.7) afirma que “novas maneiras de pensar e de conviver estão sendo elaboradas no mundo das telecomunicações e da informática (...) escrita, leitura, visão, audição, criação, aprendizagem são capturados por uma informática mais avançada”. Nesse aspecto, a mistura de códigos orais, escritos, icônicos, “*smileys*”⁴⁶ caracterizam a linguagem virtual do estilo web de escrever e ampliam o sentido particular de determinada palavra ou expressão no todo do enunciado, só estabelecido com o diálogo.

⁴⁶ Grafismos utilizados em correio eletrônico, como forma de acentuação da informalidade do meio e compensação pela inexistência de contato físico. Os mais populares são: satisfeito, insatisfeitos e os acrônimos (conjunto de siglas) (SCHNEIDER, 2007). As repercussões da escrita eletrônica no desenvolvimento da escrita manual na sala de aula e as consequências no código verbal. Disponível em: www.educacaoonline.pro.br. Acesso em: 23 de setembro de 2009.

No enunciado do texto virtual é que se estabelece uma relação com a dialogia bakhtiniana, ou seja, a preocupação em se fazer entender da maneira mais apropriada faz com que o *locutor* lance mão de todos os recursos disponíveis neste meio (neste caso, as comunidades virtuais auto-organizadas - CVAO) para atingir seu objetivo, que é a comunicação/diálogo com o interlocutor. Nas palavras de Bakhtin (1992, p.113) “(...) toda palavra comporta duas faces, pois ela é determinada tanto pelo fato de que procede de alguém, como pelo fato de que se dirige para alguém. Ela constitui justamente o produto da interação do locutor e do ouvinte.”.

Considerando a linguagem usada nos fóruns de discussão e a posição de Bakhtin acerca do discurso, evidencia-se que esse discurso que é construído, não é mais discurso de um sujeito autônomo, porque no enunciado fala a voz de outrem pelo menos em dois sentidos, no sentido da voz do interlocutor presente, cuja influência o discurso reflete, e no sentido da voz do ouvinte, cuja influência se manifesta em argumentos, pontos de vista e réplicas antecipadas (HARVEY, 1992; SANTAELLA; NÖTH, 2004; SCHAFF, 1991).

No entanto, semelhante situação ao ser transposta para o ambiente das CVAO, pressupõe uma linguagem impregnada pelos discursos alheios, códigos e sua estreita relação para com quem se fala. Esse tipo de linguagem tem sentido, função e possibilita a interação entre os membros das CVAO, pois esses usuários compartilham de maneira consensual desse novo gênero (e de seu sistema significativo), o que possibilita o diálogo. Bakhtin (1992, p.44) a este respeito especifica

(...) que todo signo, como sabemos, resulta de um consenso entre indivíduos socialmente organizados no decorrer de um processo de interação. Razão pela qual as formas são condicionadas tanto pela organização social de tais indivíduos como pelas condições em que a interação acontece.

Faraco (1988, p.24) observa que o dialogismo tratado por Bakhtin estuda o discurso interior, o monólogo, a comunicação diária, os vários gêneros do discurso, a literatura e outras manifestações culturais. E acrescenta (...)

a dialógica em Bakhtin é toda sua cosmovisão: a ele parecem interessar todas as vozes; quer ouvi-las com atenção e quer com elas interagir, não um diálogo ingênuo e esquemático que se esgota na substituição sucessiva de locutores, mas numa interação em que a meta não é nem a composição dogmática de uma voz, nem o relativismo duma coexistência acrítica de todas as vozes, mas a síntese dialética de vozes contrárias.

Em obra mais recente, Faraco (1996, p.122), embasado nos conceitos de Bakhtin, argumenta que “o caráter dialógico é o fator unificador de todas as atividades linguageiras, havendo também uma rica dialogia entre os gêneros do discurso”. Apesar dos gêneros serem formas ágeis, em constante re-elaboração, para o sujeito falante eles não deixam de ter um valor normativo, pois os gêneros são dados, informações e não é ele que os cria. “Se não existissem os gêneros do discurso e se não os dominássemos, (...) a comunicação verbal seria quase impossível” (Bakhtin, 1995, p.302).

O caráter normativo dos gêneros percebidos pelos sujeitos que lhes confere a relativa estabilidade é um princípio definidor dos gêneros (BENTES, 2005). Pode-se concluir que os gêneros discursivos preparam a fala para que a comunicação possa acontecer entre os sujeitos, historicamente situados, pois o uso da língua se concretiza na forma de enunciados (orais e escritos), vinculada ao contexto linguístico real e alicerçado no diálogo. Nessa perspectiva, o diálogo constitui uma das formas mais importantes da interação verbal, e pode-se entendê-lo num sentido mais amplo, isto é, não apenas como a comunicação em voz alta, de pessoas colocadas face a face, mas toda comunicação verbal, independente de seu tipo.

As ideias de Bakhtin têm importantes implicações para uma aproximação vigostskyana da ação mediada e pelo fenômeno social da interação verbal. Dessa forma, no processo de interação verbal as palavras não são neutras, mas trazem consigo sentidos (visões

de mundo). Bakhtin (1995, p.124) conclui que a ordem metodológica para o estudo da língua ocorre conforme a seguinte configuração:

1. As formas e os tipos de interação verbal têm ligação com as condições concretas em que se realiza.
2. As funções da linguagem são como um condutor que transfere corporeamente pensamentos de uma pessoa para outra.
3. Ao escrever e falar, as pessoas inserem seus pensamentos e sentimentos nas palavras.
4. Uma transferência contém pensamentos, sentimentos que os conduzem para os outros.
5. Ao escutar ou ler, as pessoas extraem uma vez mais das palavras os seus sentimentos e emoções.

As formas dos atos de fala evoluem como resultado da interação verbal, e o processo de evolução reflete, enfim, na mudança das formas da língua. A palavra internamente persuasiva é em parte nossa e em parte do outro, permitindo assim a interação dialógica. A sua criatividade e produtividade consiste precisamente em que tal palavra desperta palavras novas e independentes, bem como organizam um conjunto de nossas palavras interiores, e não permanecem na condição estática e isolada. Sintetizando a contribuição de Bakhtin no que diz respeito a uma das principais categorias de seu pensamento, ou seja, o *dialogismo*, pode-se inferir que o autor considera todos os diálogos e busca a síntese dialética.

Bakhtin (1995) historiciza também a linguagem, contextualizando-a histórica e socialmente, pois a significação dos objetos ultrapassa as suas materialidades, traduzindo-se em signos com funções sociais e que se determinam na organização da vida material em geral (HENRIQUES, 2007).

Em se tratando da contribuição, objeto desta pesquisa, pode-se depreender que na dialogicidade da linguagem, a convergência dos meios torna possível a convergência entre as pessoas para a partilha de informações e aprendizagem em conjunto nos múltiplos espaços virtuais (BAUME; BERTOLUS, 1995; KENSKI, 2008). Nessas questões reside a potencialidade e atualidade de Bakhtin. A relação dialógica apontada pelo autor adquire

também nas CVAO uma especificidade devido às condições materiais de transmissão e necessidades postas pelos usuários (tempo real, respostas imediatas, entonações e emoções), enfim, uma forma peculiar de interação.

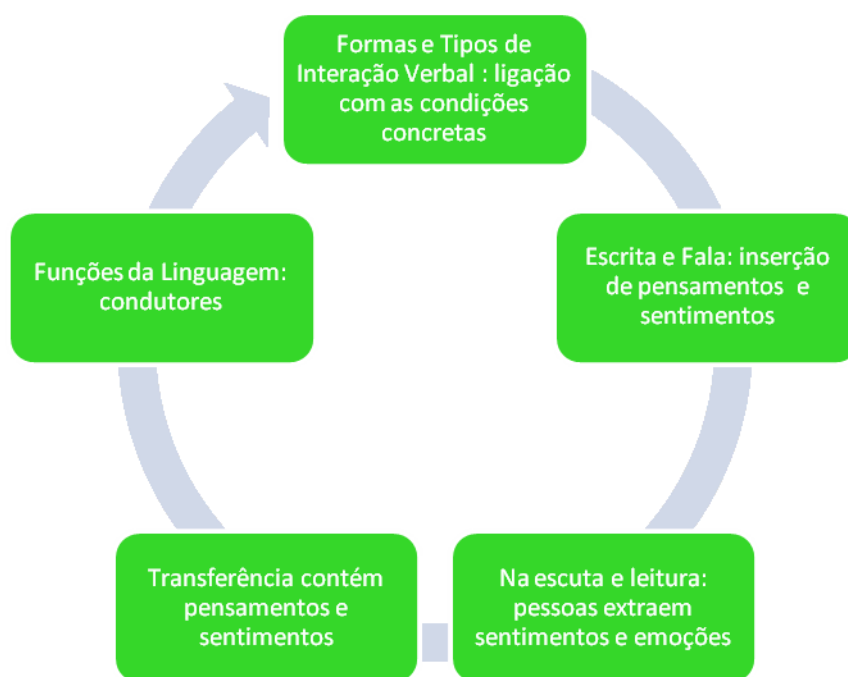


Figura 5 - Fundamentos metodológicos do estudo da língua

Fonte: Adaptado de Bakhtin (1995, p.124)

1.8.3 Wertsch: Aproximação sociocultural da mente, linguagens sociais e ação

O objetivo básico da aproximação sociocultural da mente (cognição) é elaborar uma explicação dos processos mentais humanos que reconheça a relação essencial entre esses processos e seus cenários culturais, históricos e institucionais. Wertsch (1991, 2002), vinculado à tradição sociocultural, apóia-se em estudiosos como Vigotsky (1981, 1978) para tratar do processo de internalização, de mediação dos símbolos na atividade humana; em

Bakhtin (1992) para discutir dialogia e gêneros de discurso, e principalmente em Burke (1973) para focar sobre as múltiplas perspectivas da ação humana.

Wertsch busca em Burke uma aproximação analítica para a ação humana, que este desenvolveu para descrever o dramatismo. Na análise das ações humanas, o autor sugere a existência de duas categorias de perguntas, uma de natureza ontológica e outra de natureza metodológica. Perguntar sobre *o quê* e *por quê* se refere ao ontológico. Perguntar sobre *quem*, *como* e *quando* se refere ao metodológico. Na perspectiva de Burke (1973), essas questões são elaboradas em algumas áreas específicas de conhecimento e costumam prender-se a determinados elementos constituintes da ação humana. Utiliza-se o modelo “pentagrama das telas terminísticas” elaborado pelo autor para analisar as ações humanas relacionadas às CVAO.

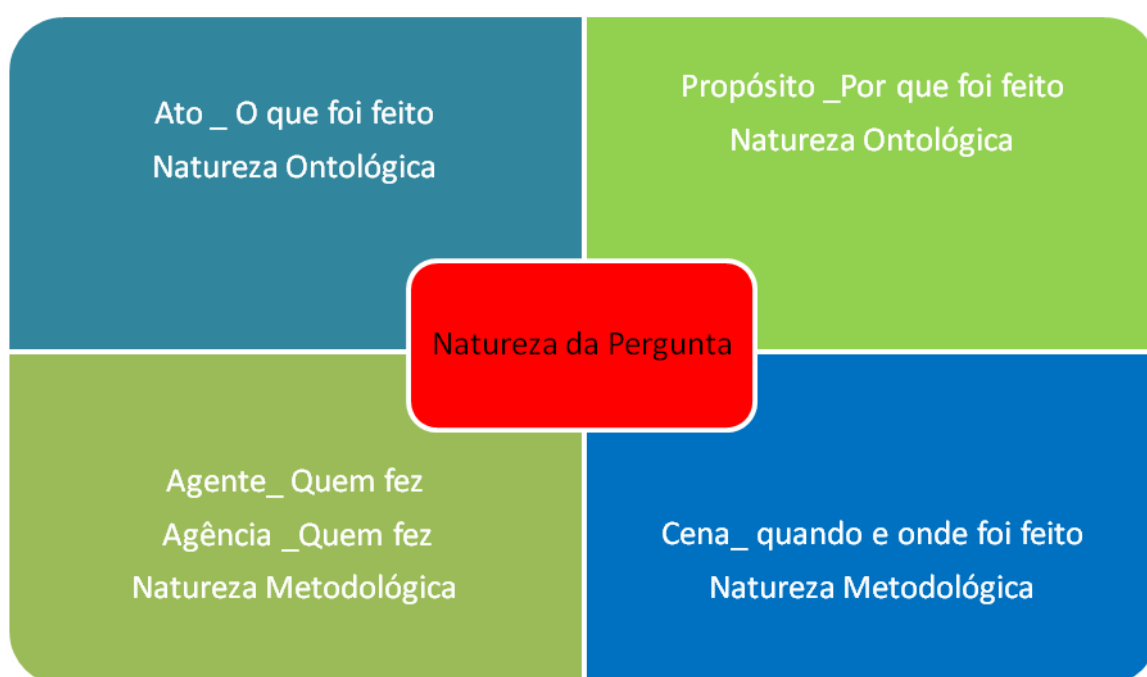


Figura 6 - Pentagrama das telas terminísticas da ação humana

Fonte: Adaptado de Burke (1973, p.68)

Assim, em uma perspectiva de análise das CVAO, o estudo da ação humana orienta-se por uma dada janela terminística e por sua terminologia, que selecionam fragmentos parciais da realidade. Burke critica essa conduta analítica de focar a realidade por meio de telas terminísticas e sugere uma abordagem capaz de vincular as perspectivas, sem reduzi-las, umas às outras. Essa abordagem está expressa na conduta metodológica de tratar os elementos do pentagrama como construtos hipotéticos, considerando suas possibilidades de transformação e faixas de combinação, e então verificar como esses construtos explicitam as ações humanas, o que implica rejeitá-los como simples reflexos da realidade. Nesse sentido, a investigação da ação humana passaria a ser conduzida por uma ferramenta, o pentagrama (Burke, 1973, p. 68), que ao vincular os cinco elementos permitiria uma análise multifocada e integrada da ação humana e de seus motivos: Natureza da Pergunta, O que foi feito (Natureza Ontológica), Propósito - Por que foi feito (Natureza Ontológica), Agente - Quem fez e Agência - Como ele fez (Natureza Metodológica), Quando e onde foi feito (Natureza Metodológica).

Certamente, a complexidade desta análise multifocada poderia inviabilizar o estudo sobre as CVAO, com o risco ainda de torná-lo superficial e relativizado. É nesse sentido que o enfoque na *ação mediada* torna-se atrativo, pois ao se considerar a tensão *agentes-agindo-com ferramentas- culturais* (agências na designação de Burke) como a unidade de análise, pode-se manter comprometido com o princípio de investigar a ação, situando-a em seu contexto cultural e institucional. Para tanto, é preciso reconhecer que a tensão irreduzível ‘agentes - ferramentas culturais’ pode ser representativa da ação mediada e ser adotada como uma unidade de análise capaz de explicar satisfatoriamente ações humanas diversificadas, como, por exemplo, aquelas que se realizam por meio dos fóruns.

Sob este enfoque, para saber quem executa a ação ou quem fala em um diálogo é preciso considerar não apenas o sujeito isolado, mas também a ferramenta cultural que ele emprega para agir ou falar. É diante da indissociabilidade entre agente e ferramenta cultural

que se considera a elaboração de significados participantes (agentes), e como eles se apropriam de ferramentas culturais, reconhecendo que tanto elaboração de significados como apropriação de ferramentas culturais são processos acoplados capazes de ser explicados na perspectiva da ação mediada.

Wertsch (1991, p.24), ao discorrer sobre o conceito de ação mediada, alicerçado pela corrente sociocultural, focaliza sua análise no agente e em suas ferramentas culturais, os mediadores da ação, ou melhor, no agente-agindo-com-ferramentas-culturais. Dessa forma, o autor toma como unidade de análise a conexão entre a ação e seu contexto cultural, institucional e histórico. Isso significa que toda ação está situada em um contexto e a tensão entre agente e ferramenta cultural pode ser adotada como unidade de análise da ação humana.

Para o autor, a compreensão sobre como e por que a introdução de novas formas de mediação se relaciona à subversão das interações pode contribuir para interpretar as mudanças de caráter intra e intermental que ocorre nos sujeitos.

Considerando as CVAO como organismo social com cultura e identidade próprias, na qual se realizam ações entre pessoas com diferentes visões de mundo, recomenda-se a teoria de ação mediada como construção teórica para interpretar os modelos de organização dessas comunidades. Nessa direção, analisam-se as comunidades virtuais baseando-se em um novo arranjo do pentagrama, tendo em vista que:

1. as ações são realizadas por meio de *atividades estruturadas*, que supõem formas de interação entre os participantes (agentes), o propósito da atividade e o ato em si (disponibilização de conteúdos, comunicação e comercialização de produtos e serviços);
2. as ações mediadas nos fóruns têm como objetivo permitir que um número significativo de indivíduos possa se auto-organizar, de maneira altamente descentralizada e distribuída no espaço e no tempo, visando a

busca de conhecimentos e equacionamento de problemas (pesquisa e aprendizagem). É preciso reconhecer que essas formas de pensamento são como ferramentas culturais (*conceitos*) que são usadas pelos participantes das comunidades em um ‘cenário’ convenientemente construído;

3. a ampliação da visão de mundo ocorre porque o sujeito se depara com realidades distintas daquelas de seu cotidiano e essa ampliação ocorre devido ao embate entre as ferramentas culturais usadas no cotidiano e aquelas usadas nos cenários construídos nos fóruns, então é importante considerar a *tematização* como um terceiro elemento capaz de aproximar os participantes de horizontes conceituais (visões de mundo).

Dessa forma, o modelo topológico fundado na teoria da ação mediada (BURKE, 1973; WERTSCH, 1991) propõe a atividade, o conceito e a tematização como três eixos organizadores que podem auxiliar na análise das CVAO. Os eixos organizadores são fundamentados em algumas propriedades da ação mediada e se traduzem em:

1. situacionalidade (contexto e continuidade);
2. historicidade (narrativa; são características da narrativa: organizada em torno da temporalidade; ter um assunto central com começo, meio e fim bem marcados e uma voz narrativa identificável; atinge um fechamento, uma conclusão, uma resolução);
3. materialidade (dos instrumentos aos dispositivos de pensamento);
4. mediação (fala e representação).

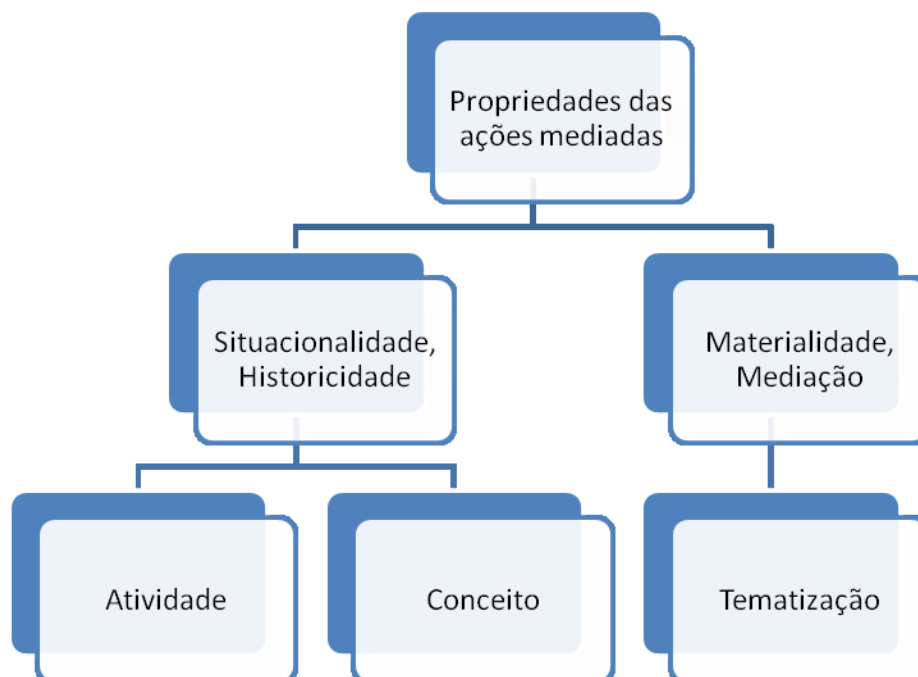


Figura 7 - Modelo topológico fundado na teoria da ação mediada

Fonte: Adaptado de Burke (1973, p.68)

Em seu livro *Vozes da Mente*, Wertsch (1991, p.13) explica o significado do título do livro: O termo “vozes” está descrito num contexto, em que “existem inúmeras formas de representar a realidade ao se defrontar com um problema que possibilita várias abordagens de análise”. O enfoque sociocultural da mente é explicar como se localiza a ação humana em âmbitos culturais, históricos e institucionais. A chave para tal explicação é o uso da noção da *ação mediada* como unidade de análise e pessoal que atua com os instrumentos mediadores como descrição adequada do agente dessa ação, pois os instrumentos mediadores estão inerentemente relacionados à ação. O termo *mente* em vez de *cognição* (habitualmente mais utilizado em âmbitos psicológicos) reflete o desejo de integrar um amplo leque de fenômenos psicológicos.

Wertsch (1991), tanto quanto Vigotsky e Bakhtin (1992) acreditam que é sem sentido isolar diversos aspectos dos processos mentais para analisá-los separadamente. Por outro lado, boa parte da sua obra é relativa aos processos cognitivos, com o desejo de que a discussão não termine e sim que resulte em algo relevante, para quem interessa por outros aspectos da vida

humana como, por exemplo, pelas emoções. Ressalta que elegeu o termo “sociocultural” com a finalidade de reconhecer as importantes contribuições de diversas disciplinas e escolas do pensamento e do estudo da ação mediada. O autor salienta que são reconhecidas as contribuições de Vigotsky e outros autores (embora eles utilizem o termo sócio-histórico mais do que o termo sociocultural, para descrever as suas posições), para muitos contemporâneos estudiosos da cultura.

Uma aproximação cultural da mente começa com o pressuposto de que a ação está mediada, e que não pode ser separada do meio em que se dá o suporte para a conclusão do processo. Wertsch (1991) aponta três temas básicos que podem ser reconhecidos em uma aproximação sociocultural vigotskiana e ação mediada.

1. A confiança na análise genética e evolutiva;
2. A afirmação de que as funções mentais superiores do indivíduo derivam da vida social;
3. A ideia de que a ação humana, tanto no plano individual como no plano social, está mediada por ferramentas e sinais ou símbolos.

Ao centrar-se nos instrumentos mediadores, a mente vai muito mais além, sobretudo em outro sentido: o agente da ação mediada é concebido como o indivíduo ou os indivíduos que atuam em conjunção com os instrumentos mediadores, podendo estes estar constituídos por ferramentas e/ou sinais. Nas práticas discursivas das quais participam, inclusive nos fóruns de discussão, as pessoas estão constantemente construindo e reconstruindo o mundo e a si mesmas, dentro “de um contexto cultural, histórico e institucional, pois suas ações e interações não podem ser separadas do espaço sociocultural no qual elas ocorrem (WERTSCH, 1991, p.9). Não podemos considerar “a natureza ou os indivíduos isoladamente” (WERTSCH, 1991, p.10).

Os seres humanos são vistos como estando sempre em contato com, ou criando, seus arredores, assim como a si mesmos, por meio das práticas nas quais se engajam. Como as pessoas não existem de forma isolada e não estão situadas em um vácuo social – estão inseridas no mundo social, pois a “ação é mediada e (...) não pode ser separada do meio social em que está inserida” (WERTSCH, 1991, p.18) – podemos chamar suas ações e interações de práticas sociais, e o discurso faz parte dessas práticas sociais.

A noção de *heterogeneidade apesar da hierarquia genética*, discutida por Wertsch (1991) também fornece um argumento psicológico importante para o fato de que a aquisição de um conceito mais complexo num perfil conceitual não implica o desaparecimento das ideias anteriores. O fenômeno de heterogeneidade lida com a mesma ideia geral de que numa dada cultura ou num indivíduo possam existir diferentes formas de pensar. A heterogeneidade, apesar da hierarquia genética, assume que essas diferentes formas podem ser colocadas numa hierarquia genética, mas isso não implica que as formas posteriores sejam mais poderosas. Wertsch (1991, p.31) argumenta que "essa posição pode ser resumida dizendo-se que, embora algumas formas de funcionamento mental apareçam depois de outras, elas não são inerentemente melhores". O reconhecimento de que certas formas de pensar, certos mediadores ou certas linguagens sociais (Bakhtin, 1992) sejam mais apropriados e eficazes em certos contextos envolve uma atitude psicológica de reconhecer esses contextos e *privilegiar* determinados mediadores. O reconhecimento dos tipos de escolha que os participantes da CVAO fazem quando tentam resolver questões nos fóruns pode ser um importante dado empírico para entender o mecanismo envolvido no reconhecimento do contexto e no privilégio dado a certos mediadores.

Para Wertsch (2002, p.13) a influência de Marx e Engels nas obras de Vigotsky evidencia-se quando este discorre sobre o uso das ferramentas nas atividades de trabalho. A contribuição principal de Vigotsky derivou da sua “concepção das ferramentas psicológicas,

em contraste com as técnicas”. Graças ao seu interesse continuado pelos complexos processos da ação semiótica humana, resultou em buscar com grande complexidade o rol de um sistema de sinais, tais como a linguagem humana, os diagramas e a aritmética. Contrastando com muitas análises contemporâneas da linguagem, com a sua estrutura posta nos sistemas de sinais, com independência de qualquer ação mediadora que poderiam cumprir, Vigotsky encarou a linguagem e outros sistemas de sinais como parte e como mediadores da ação humana (daí a sua associação com o termo “ação mediada”).

A explicação vigotskyana da mediação como um conjunto de supostos próximos da natureza de determinadas funções mentais superiores, mais especificamente, a sua concepção de que o pensamento, a atenção voluntária e a memória lógica formam um sistema de “relações interfuncionais” (WERTSCH, 2002, p.15). Em sua última obra *Pensamento e Linguagem*, Vigotsky (2001, p.23) assinalou que o “estudo do pensamento e a linguagem são uma das áreas da psicologia que resulta particular importância de uma compreensão clara das relações interfuncionais”. De qualquer forma, essa obra está totalmente dedicada à questão de como a fala e o pensamento chegam a encontrar-se completamente entremeados na vida humana. Segundo Wertsch (2002, p 17), ao examinar

“as relações entre linguagem e pensamento, Vigotsky colocou o acento principal e como se relacionam diferentes formas de linguagem com as mais diferentes formas de pensamento. (...) Outro pressuposto não menos importante também foi ressaltado: os mediadores verbais deveriam ser usados o mais amplo e frequente possível”.

Dessa forma, o privilégio diz respeito ao fato de que um instrumento mediador se concebe como o mais apropriado ou eficaz que outros em determinado cenário sociocultural. Basicamente esse elemento está vinculado ao fato de que os usuários supõem que certos instrumentos mediadores são apropriados e inclusos na única alternativa possível, embora em princípio existam outras possibilidades. Aplicando esses princípios às CVAO, sem cair num reducionismo, podemos concluir (não em definitivo, dado o caráter histórico) que a teoria de

Wertsch (2002, 1981) sobre o conceito de ação mediada pressupõe que é a ação, e não apenas os seres humanos ou a natureza considerados isoladamente, que constitui o ponto de partida para a análise do conceito de ação mediada.

1.8.4 Aportes críticos de Cole: mediação e os artefatos culturais

Em diversos trabalhos, Cole (1985, 1994) faz uma síntese do processo que levou a encontrar na obra de Vigotsky uma alternativa teórica aos problemas práticos que teve que enfrentar como profissional de psicologia.⁴⁷ O autor distingue noções de contexto definido como aquilo que entrelaça, retomando a obra de Bronfenbrenner (1979) sobre a ecologia do desenvolvimento humano, que retrata camadas de contextos em círculos concêntricos. A forma como indivíduos ou grupos usam os artefatos realmente transforma o modelo dos contextos que existem num dado momento num cenário particular. Segundo Cole (1994, p.84) “o que mais me preocupa no contexto dessa discussão é que a cultura foi por demais tratada como um pacote de variáveis independentes, não como um meio e não foi o objeto direto da análise”. O autor faz importante distinção entre habilidades e atividades no contexto de abrangente discussão sobre o conteúdo que pode ser considerado no currículo e no papel da cultura.

Indaga a si próprio como é possível que os instrumentos científicos projetados para a compreensão das pessoas produzam contradições, ou seja, que simultaneamente levem a ver as pessoas como muito prontas e também muito incapazes. Tal fato conduz à investigação sobre a origem dos instrumentos utilizados no exame das capacidades cognitivas das pessoas,

⁴⁷ Michael Cole é um dos psicólogos norte-americanos que sem dúvida têm contribuído para o avanço a aplicação das obras de Vigotsky. De forma sintética, depois de uma formação experimental rigorosa e especializada no ensino das ciências matemáticas, o autor obteve uma experiência de trabalho na Libéria. Naquele país o autor detectou que os estudantes (crianças e adultos) eram incapazes de resolver problemas aparentemente simples e triviais associados a tarefas experimentais (como, por exemplo: montar quebra-cabeças e classificar figuras) enquanto podiam dar mostras de capacidades intelectuais surpreendentes em atividades da vida cotidiana (COLE, 1994; COLE; ENGESTROM, VASQUEZ, 1997).

a qual por sua vez remete a um estudo dos antecedentes históricos das suposições contemporâneas sobre a simbiose entre mente e cultura. Fazendo uma ligeira busca na antropologia norte-americana do século XIX mostra que suposições como a citada já estavam presentes na psicologia cognitiva, a saber:

1. Que a mente e a sociedade são aspectos do mesmo fenômeno;
2. Que as sociedades se caracterizam por níveis de desenvolvimento cultural-mental;
3. Que os níveis de cultura ou graus de civilização são uniformes dentro das sociedades;
4. Que a mudança é o resultado de fatores mentais-culturais endógenos.



Figura 8 - Mediação e artefatos culturais na perspectiva de Cole

Fonte: Adaptado de Cole (1994)

Excetuando-se a primeira suposição, a psicologia cognitiva tem conservado as outras três, da seguinte maneira: a conduta individual pode-se caracterizar pelo seu nível de desenvolvimento (que só pode ser um), pelo funcionamento do indivíduo todo e, por último, que o desenvolvimento é a dicotomia entre a cultura e a mente como campos especializados

da antropologia e da psicologia, respectivamente, em que caberia à antropologia descrever e catalogar os produtos culturais, enquanto à psicologia caberia associar os processos cognitivos individuais com os ditos produtos culturais (COLE 1987, 1994).

A experiência da pesquisa transcultural serve para o autor concluir que muito pouco se avança quando se relaciona cultura e mente em pressupostos de que a primeira especifica as variáveis independentes, enquanto a segunda produz as variáveis dependentes. Reconhece ainda a necessidade de se desenvolver uma maneira de superar conceitualmente a dicotomia existente entre cultura e mente, acreditando encontrar no conceito de Zona de Desenvolvimento Proximal - ZDP (VIGOTSKY, 1978, 1981; WOOD; BRUNER; ROSS, 1976) uma boa alternativa para as tentativas.

COLE (1994) busca o mecanismo da mudança individual na interação entre os indivíduos, os quais ao mesmo tempo constituem a sociedade por meio das suas interações no nível mais evidente, organizando com elas nos contextos que formam e mudam a aprendizagem, o seu desenvolvimento e os seus esquemas. De acordo com essa perspectiva, o conceito ZDP permite compreender o seguinte:

1. Que as crianças podem participar no desenvolvimento de atividades que não entendem completamente e que são incapazes de realizá-las individualmente;
2. Que as situações reais de solução de problemas, não possuem passos pré-determinados para a solução e nem papéis fixos dos participantes. Basicamente a solução está distribuída entre os participantes e que é a mudança na distribuição da atividade com respeito à tarefa que constitui a aprendizagem;
3. Que nas ZDP's reais o adulto não atua somente de acordo com a sua própria definição da situação, mas sim se baseando na interpretação dos gestos e fala da criança como indicadores da definição da situação por parte desta;

4. Que as situações que são novas para a criança não são da mesma maneira para os outros presentes e que o conhecimento faltante para a criança decorre de um ambiente organizado socialmente;
5. Que o desenvolvimento está intimamente relacionado com o contexto que pode ser negociado por um indivíduo ou por um grupo social.

O autor adota o conceito de ZDP basicamente no sentido com que havia proposto por Wood, Bruner e Ross (1976) para entender o tipo de apoio proporcionado por um adulto que ajuda uma criança a realizar uma tarefa que não poderia realizar sozinha. Nesse caso específico, não existe reflexão em relação às condições de elaboração do conceito por Vigotsky e nem sobre a forma de sua recuperação. Entretanto, é evidente que o horizonte da reflexão desdobra qualquer campo temático específico, pois há que se repensar a relação mente-cultura. Por esse dado, podem ser colocadas em discussão várias linhas de reflexão.

Em tal contexto, o autor ainda considera necessário ter cautela a respeito do uso do conceito de ZDP por diversos pesquisadores norte-americanos (COLE; GRIFFIN, 1983). Primeiramente, assinala que a fonte-padrão das discussões sobre ZDP havia sido uma versão recortada da obra de Vigotsky *Linguagem e Pensamento*, que foi traduzida para o inglês em 1962. Vale a pena chamar também a atenção sobre a necessidade de contextualizar historicamente o surgimento do conceito e, nesse sentido, salienta que Vigotsky o desenvolveu enquanto era diretor do Instituto de Paidologia, com o intento de neutralizar a difusão das ideias provenientes da Europa ocidental. Por fim, fez-se a distinção das modalidades na forma em que se havia retomado o conceito de ZDP pelos norte-americanos:

1. A versão do “passo seguinte”, segundo o qual o desenvolvimento da criança poderia aumentar se o ambiente proporcionava a quantidade justa de discrepâncias entre os resultados precedentes e as demandas atuais;

2. A versão dos “andaimes” que sustentam a intervenção tutorial do adulto é inversamente proporcional ao nível da competência da criança na tarefa em questão.

O autor também critica essas interpretações da ZDP. A primeira versão por sua limitação estreita na maneira de se conceber uma tarefa. O conceito ZDP implica vários níveis de tarefas simultâneas, e nisso segue os adultos - sobre todos os cenários da vida real – que creem e apoiam diversos níveis de participação das crianças. A segunda versão é criticada porque sugere mais uma variação quantitativa do que qualitativa – enquanto as mudanças no apoio que proporciona o adulto à criança apontam para as diferenças de qualidade – e , especificamente, porque deixa abertas as perguntas sobre a criatividade da criança durante o processo.

Cole (1997), após as críticas citadas, mostra a necessidade de um conceito de ZDP que não mais seja apenas uma metáfora espacial, mas que se reintroduza o componente temporal da construção, assim como a importância de levar em conta tarefas de forma sequenciada menos marcada, em que os adultos tenham tanto um papel como uma capacidade de superar a ambiguidade.

Fundamentando-se nos trabalhos de Bogoyavlensky e Menchinskaya (1991), resgata o conceito de sistema funcional, o qual se distingue por romper com uma versão simplista da relação entre estrutura e função e propõe que os sistemas funcionais não somente se caracterizem pela complexidade da sua estrutura, mas sim pela flexibilidade dos papéis que jogam os participantes. O autor, baseando-se nesse princípio, sugere que os componentes da ZDP constituam um sistema funcional: materiais, tarefas, adultos, crianças, modelos do futuro e modelos do passado funcionem de modo conjunto para tornar possível o desenvolvimento.

Norteando-se também nos trabalhos de Leontiev (1932, 1991), o autor retoma o conceito de atividade-guia, que foi concebida como a totalidade que não se constitui mediante o agregado de ações separadas, mas caracteriza-se como uma forma de relação definida da criança com a realidade e com seu nível de desenvolvimento (jogo, aprendizagem formal, relação com os pares, trabalho, etc.). Essa ideia serve para destacar três aspectos, segundo o autor:

1. O que constitui uma alternativa à estratégia de definição das etapas internas e individuais;
2. O que proporciona um marco para vincular os aspectos diversos do desenvolvimento;
3. O que sugere maneiras de experimentar com o reordenamento das progressões, com fins teóricos e práticos.

Alicerçado nessa reelaboração conceitual, Cole e Griffin (1983) expuseram a sua utilidade prática em duas situações de aprendizagem: uma, com um programa de computação projetado como um jogo para aprender sobre o traçado numérico de uma reta, e outra por meio de uma situação mais estruturada quanto ao aspecto da aprendizagem em que se retratava uma oficina de verão oferecida às crianças com dificuldades de aprendizagem em suas respectivas escolas. Os autores não discutem as implicações dessa nova conceitualização de ZDP no contexto da relação cultura e mente, porém no que se concentra mais adequadamente nas lições das duas situações de aprendizagem estudadas. Dessa maneira, é assinalado que o papel do adulto não é apenas padronizado e sim um papel variado nos distintos sistemas funcionais da atividade. Por exemplo, nem sempre proporciona apoio para uma progressão no grau ascendente de complexidade nem assume as funções executivas ou de ordem superior. Na utilização do jogo de computação, a própria criança demandava e

regulava a ajuda adulta necessária para a sua própria atividade mental, além de também salientar que o saber do adulto proporciona uma teleologia para o desenvolvimento da criança, e que a organização social da atividade-guia oferece um espaço para que se possam realizar análises criativas.

Embora os autores não retomem a discussão a qual iniciaram em 1983 – e nesse sentido parecem não explorar as possibilidades de análise realizadas – é evidente que a nova conceitualização de ZPD agrega algumas especificações aos pontos b e c citados acima (principalmente no relacionado ao papel da atividade e conhecimento do adulto no processo). E, por último, porém não menos importante, oferece uma demonstração empírica da utilidade prática do conceito revisado de ZPD. Essa última constitui não só um espaço de demonstração do que foi refletido teoricamente, mas um verdadeiro terreno de diálogo em que se construíram ou reconstruíram os conceitos.

No ano seguinte, Cole (1985) volta a investigar a mesma questão. Por princípio, chama a atenção para a ideia de que a ZPD é o lugar em que a cultura e a cognição se criam mutuamente, daí se segue que o conceito pode permitir a reintegração dos trabalhos dos antropólogos e dos psicólogos. O autor enfatiza a preocupação de Vigotsky (1981) em romper com a dicotomia entre indivíduo e ambiente social e a intenção de encontrar uma forma de vê-los como mutuamente constitutivos. Não obstante, também salienta que pode haver sido uma prova crucial, pois os estudos realizados junto aos “uzbekos” foram limitados ao incluir conteúdos culturais locais dentro dos formatos de tarefas padronizadas projetadas para descobrir um suposto funcionamento mental universal.

Foi enfatizada também por Vigotsky (2001) a natureza interativa das mudanças do desenvolvimento e uma maneira de caracterizá-lo foi quanto às variações no controle da responsabilidade. Como já vimos antes, tal concepção aplicou-se primeiramente no contexto educativo e das provas psicológicas, sendo este o campo predominante de uso. Cole (1985)

advoga considerá-lo como o núcleo de uma concepção mais geral: a ZDP como estrutura da atividade conjunta em qualquer contexto no qual existem participantes exercendo responsabilidades diferenciadas devido a seu nível de conhecimento distinto. O autor ainda faz um pequeno levantamento das pesquisas de natureza psicológica e antropológica que descrevem as zonas de desenvolvimento proximal em atividades organizadas culturalmente: o uso de provérbios entre os “Kpelle”, a aprendizagem sobre os tecidos e, entre os “zinacantecos”, a aprendizagem sobre a alfaiataria.

Dessa forma, conclui que podem considerar como estabelecidos os seguintes pontos:

1. A ZDP é uma unidade básica comum ao analisar as culturas e os processos psicológicos;
2. A unidade consiste em um indivíduo implicado em uma atividade dirigida a uma meta (atividade, tarefa, evento) respeitando as restrições convencionadas;
3. As atividades estão povoadas por outros; principalmente, no caso das crianças, por adultos;
4. A aquisição da conduta culturalmente apropriada é um processo de interação entre as crianças e adultos, e estes guiam a conduta daqueles como elementos essencial do processo.

Até esse ponto, Cole (1985) destacou as seguintes questões afins ao conceito de ZDP:

1. Permite entender a possibilidade mesmo de alguém em que a atividade, em sentido estrito, é incapaz de realizar por si só (motivo pelo qual, desde outras perspectivas, afirmar-se-ia a impossibilidade da dita experiência). Tal participação pressupõe a outra pessoa com especialidade e responsabilidade diferenciada na atividade;

2. Como a ZDP é um produto da interação, entende-se que não implique uma sequência predeterminada de ações e nem papéis fixos para os participantes (em particular, o papel das ações e conhecimento do adulto);
3. Como produto interativo, a ZDP salienta que é inadequado tomar somente a perspectiva do adulto ou somente a perspectiva da criança na análise do processo que ocorre entre eles, é basicamente dizer que se supõe um significado específico da interação irreduzível à soma das perspectivas isoladas dos participantes;
4. Como produto da interação, a ZDP não implica uma dimensão temporal irreduzível do aqui e agora, sinal este que sintetiza a relação do presente com o passado e o futuro, síntese que realiza um plano predeterminado;
5. Permite repensar o desenvolvimento como uma ramificação complexa intimamente vinculada a uma faixa de contextos que pode negociar com uma pessoa ou grupo, em vez de com um escalão ou etapa homogênea dentro de uma progressão que permeia a totalidade das possibilidades do indivíduo;

O problema sobre a definição de cultura não é abordado pelo autor quando analisa o conceito de ZDP. Essas abordagens referentes à cultura estão mais claramente orientadas em seus trabalhos mais recentes, o que significa uma mudança no papel desempenhado pelo conceito de ZDP. Cole (1994) ressalta em sua obra os níveis de desenvolvimento estudados por Vigotsky (1978, 1991). O desenvolvimento real é o que já foi consolidado pelo indivíduo, de forma a torná-lo capaz de resolver situações valendo-se de seu conhecimento de forma autônoma. O desenvolvimento potencial é aquele que o sujeito poderá construir com o auxílio de outro, e esta capacidade de realizar tarefas de forma independente de nível de

desenvolvimento real é que determina até onde a criança já chegou, ou seja, as etapas já conquistadas pela criança. Porém, Cole (1994) centrou grande parte de sua análise no conceito de ZDP, no esforço de ampliá-lo superando sua centralidade.

Na discussão sobre a existência de fontes de uma psicologia cultural, Cole (1997) realizou um deslocamento das categorias centrais sobre os problemas que foram assinalados. Com base em Vigotsky, Luria e Leontiev e outros (1991), o autor salienta que os processos psicológicos humanos são mediados culturalmente, são desenvolvidos historicamente e surgem com a atividade prática. Esse deslocamento significa que o conceito de ZDP não pode ser o eixo da reflexão, pois os três recursos específicos dos processos psicológicos humanos não se destacam imediatamente ou de forma evidente nele mesmo.

No entanto, a psicologia cultural como parte do reconhecimento das capacidades especiais dos seres humanos em modificar o seu ambiente criando artefatos e transmitindo as modificações acumuladas, para as gerações futuras, evidencia que o conceito de artefato cultural é perfilado como núcleo conceitual da reflexão do qual são imediatamente aparentes a mediação, a historicidade e a vinculação com a atividade prática. O autor introduz, embora sem desenvolvê-la, a ideia de que é necessária uma concepção de cultura como o meio único da existência humana que funcione simultaneamente como restrição e como ferramenta da ação.

Os artefatos cumprem a função básica de coordenar os seres humanos com o mundo físico e entre si. São ao mesmo tempo materiais e ideias (aspectos conceituais e simbólicos), assim, enquanto está agindo sobre o mundo também pode trabalhar de forma significativa.

Como resultado da natureza matéria/ideia dos sistemas de artefatos integrantes do ambiente cultural, os seres humanos vivem em um mundo duplo, simultaneamente material e artificial ou virtual (COLE, 1985, p. 285).

Uma vez que os artefatos são criação humana, é evidente que possuem uma história que não pode ser negligenciada como parte da constituição dos processos psicológicos. Como corolário do que foi acima citado, e se a atividade se constitui pelo uso de artefatos com a história, é indispensável estudar os processos psicológicos com base nas formas historicamente específicas de atividades práticas nas quais estão implicadas as pessoas.

Dando continuidade ao pensamento, deve-se dar ênfase na especificidade contextual dos processos mentais, suas origens sociais e a necessidade de uma análise genética (o que não deve ser entendido como somente um estudo do desenvolvimento da ontogenia, mas como a necessidade de compreensão do processo em seu movimento, considerando os diversos domínios genéticos, a saber: filogenia, história, ontogenia e microgênese) (COLE, 1987, 1994; NEWMAN; GRIFFIN; COLE, 1989).

O autor ainda propõe colocar em ação um novo marco analítico pertinente à aquisição da alfabetização. Para isso considera os níveis filogenético, histórico (nos quais se discutem as origens da escolaridade formal), ontogenético (revisando a relação entre a escolaridade e o desenvolvimento das operações lógicas e a alfabetização sem escolaridade) e o microgenético (relação com uma experiência empírica – que eventualmente obteve sucesso – de aquisição da leitura por crianças com problemas escolares, no início dos estudos) em que concede especial atenção à forma como tem que ser o repensar:

1. A estrutura da atividade mediada para incluir o tempo;
2. O modo específico de articulação criança-adulto-texto-mundo que faz tornar possível o desenvolvimento de uma nova relação criança-mundo, mediada pelo texto.

É no microgenético que fica associada a preocupação prévia pela ZDP, cujo conceito já não é mais utilizado por Cole nessa discussão. Nesse sentido, Cole (1997, p.328) constata

que há evidência de que as crianças interiorizam o roteiro da atividade que participam com os outros, daí a análise é chamada em termos de mediação, como segue:

A leitura (...) é um processo emergente de construção de significado que ocorre quando a informação topicalizada pelo texto sintetiza com o conhecimento prévio como parte de um processo geral da interação mediada com o mundo. A especificidade desta aproximação (...) está na ênfase e no papel social do professor e no ajuste das condições que coordenam os sistemas de mediação preexistentes em um só sistema de atividades subordinado à meta da compreensão.

Basicamente o abandono do conceito de ZDP como eixo da reflexão permite indicar explicitamente os problemas que antes ficavam implícitos, compreendidos e vagamente sugeridos. Tal é o caso da formulação de uma concepção explícita da cultura, que embora seja breve serve para apontar as elaborações a serem realizadas posteriormente. Ao demonstrar a clareza do caráter histórico dos artefatos, também se torna urgente a necessidade de visualizar a ação como uma síntese de distintos planos de temporalidade (ou para dizer negativamente, a impossibilidade de reduzir ao aqui e agora). Contudo, inclui o conceito de artefatos culturais como exposto de forma esquemática e é clara a necessidade de um exame mais profundo.

Em três recentes trabalhos sobre o tema, Holland e Cole (1995) desenvolveram o conceito de cultura como o conceito de artefato cultural.

Levando em consideração tais estudos, a cultura em si é pensada como mediadora:

1. As pessoas não se relacionam com as condições biológicas ou ambientais de maneira direta, mas sim dentro de um meio único que é a cultura;
2. A cultura constitui-se e transforma-se por meio dos artefatos das gerações precedentes, porque não é um conjunto aleatório, mas possui estruturação (COLE et al. (1971). Os autores assinalam duas fontes de estruturação: a modularidade da mente e os contextos culturais da ação.

Embora à primeira vista possa parecer que cada uma dessas fontes aponta para os polos da dicotomia tradicional entre filogenia e cultura, o autor afirma ainda que seja um erro supor que determinado dado (input) filogenético atuaria de pronto, na modificação cultural. Pelo mesmo autor ainda é sustentada a ideia de uma coevolução dessas duas fontes de estruturação: nem as predisposições congênicas são constituídas à margem de contextos culturais, bem como nem esses últimos são formados independentemente das disposições dos seres humanos. Esse entrelaçamento está presente não somente a nível filogenético, mas também no histórico incluindo o microgenético.

O conceito de artefato cultural tem agregado a especificação de sua natureza dual: são os ideais contendo em forma codificada as interações que foram previamente parte e as que fazem a mediação no momento presente. São materiais consagrados como artefatos materiais (COLE, 1994, p. 10). A dimensão ideal dos artefatos é a que faz possível projetar o passado cultural em um futuro imaginado e logo o retrocede ao presente em forma de crenças que organizam e restringem o desenvolvimento. A mediação “sempre ocorre como parte de uma unidade mais ampla de estruturação sociocultural referida como o contexto, situação, atividade, etc.” Assim, pode-se entender melhor a definição mais recente do artefato como “um aspecto do mundo material com o uso recordado coletivamente” (COLE, 1994, p. 34).

É evidente que ambos os conceitos se complementam mutuamente. Somente nessa relação é possível deixar de pensar a cultura como forma de interpretar a realidade e relacionar-se com ela, o que é coerente e intemporal e que, por isso, é compartilhada por todas as pessoas independentemente de sua posição social. Da mesma maneira, é oportuno apresentar um enfoque sobre as formas circunscritas da atividade habitual e de interpretação coletiva da atividade social, assim como uma análise mais ampla das formas culturais particulares, a sua especificidade histórica, os detalhes de sua circulação e as particularidades de seu significado para a subjetividade. O aspecto material e ideal dos artefatos culturais nos

permite reconhecer que estão em jogo os sistemas de significados compartilhados, todavia sem esquecer que as interações sempre estão situadas no tempo e no espaço e, por isso, coloca-se em jogo um posicionamento dos participantes que implica relações de poder.

A contribuição do autor no estudo das CVAO permite visualizar que a mudança notável constitui o deslocamento do conceito de artefatos culturais para o centro da reflexão. Tal conceito permite abordar de modo novo a relação entre ação e contexto, pois, tendo uma dupla natureza (material/ideal), os artefatos são ao mesmo tempo “coisas que estão aí”, entretanto só podem constituir-se como ferramentas (ou restrições) de uma atividade para os que podem responder à sua dimensão significativa.

Capítulo II

2 METODOLOGIA

Esta pesquisa de cunho qualitativo utiliza o método da etnografia virtual, proposto por HINE (2000) e a teoria analítica da ação mediada na perspectiva de WERTSCH (1991). Para a análise das inferências acerca de dados sobre o funcionamento das CVAO, utiliza-se a técnica da Análise de Conteúdo proposto por BARDIN (2000). A figura 9 representa a síntese de instrumentos utilizados na metodologia.

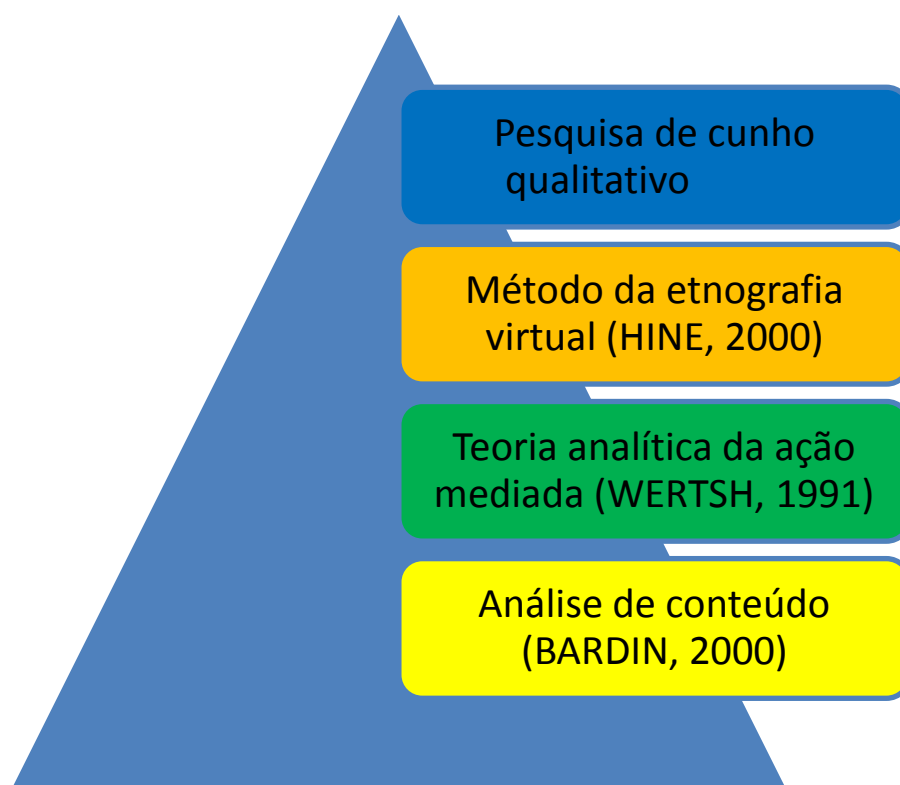


Figura 9 - Representação gráfica da metodologia utilizada

2.1 *Etnografia Virtual*

A etnografia é uma metodologia utilizada para a investigação antropológica, por meio da técnica de observação participante, em que o pesquisador se integra aos processos sociais

que investiga, para obter informação sob a perspectiva de membro do grupo, com o objetivo de compreender as estruturas de significados.

Por etnografia virtual entende-se uma forma do fazer etnográfico nos espaços virtuais suportados pela Internet.

A etnografia virtual possui os mesmos riscos metodológicos da etnografia tradicional, pois é possível fazer a etnografia dos coletivos mediados pelas tecnologias da informação e comunicação, como a Internet, utilizando as técnicas básicas.

A adaptação da metodologia etnográfica às propriedades dos fenômenos que se desenvolvem no mundo virtual implica repensar muito os seus conceitos básicos e planejamento metodológico. A ideia de campo e de entrada em campo, o conceito de observação participante e da identidade do pesquisador, são componentes que podem ser repensados e que apresentam consideráveis diferenças com respeito às etnografias realizadas com grupos humanos, face a face.

Os estudos etnográficos demonstram que os ambientes virtuais são realmente agrupamentos sociais com características e fenômenos que ainda precisam ser investigados, portanto o foco deve ser na interação social e cultural desses grupamentos, e não somente na tecnologia como o recurso ou componente que determina e condiciona as mudanças. A esse respeito Grint e Woolgar⁴⁸ (1997, *apud* HINE, 2000, p.1) argumentam:

Discordamos da suposição implícita de que algumas propriedades ou características inerentes à tecnologia expliquem o impacto da tecnologia em nossas vidas. Propomos então que muitos outros aspectos da nossa relação com a tecnologia devam ser levados em conta se pretendemos adquirir um entendimento de suas consequências. Estes outros aspectos incluem: nossas atitudes em relação à tecnologia, nossas concepções do que a tecnologia pode ou não pode fazer, nossas expectativas e suposições sobre as possibilidades de mudança tecnológica e as várias maneiras nas quais a tecnologia está representada na mídia e nas organizações. Objetivamos fornecer uma exploração crítica da visão de que estes aspectos da tecnologia são consequências da maneira pela qual organizamos nossas atividades de

⁴⁸ GRINT, K.; WOOLGAR, S. (1997). *The Machine at work: technology, work and organization*. London: Cambridge.

trabalho, instituições, lazer e aprendizado. Este argumento requer o entendimento de diferentes maneiras de pensar e representar a tecnologia, pelo menos tanto como as diferenças na tecnologia por si própria. De fato, argumentamos para a necessidade de tratar a ideia de “tecnologia por si própria” com considerada cautela.

A citação acima ressalta que iniciativas de estudos sistemáticos e aprofundados sejam empreendidas para se obter e usufruir de uma melhor compreensão da tecnologia como agente de mudança.

Os ambientes virtuais formados no contexto da Internet tornam-se campo fértil para este estudo. Hine (2000) afirma que a utilização da metodologia etnográfica deve ser o ponto de partida para o início de estudos dessa natureza. A metodologia etnográfica pode ser utilizada para explorar as complexas relações existentes entre as pessoas e as novas tecnologias, e nos mais diferentes espaços sociais, tais como: residências, local de trabalho, mídia de massa, bem como ambientes educacionais e acadêmicos.

Um estudo etnográfico da Internet pode oferecer um olhar, em detalhe, da forma como a tecnologia é vivenciada no cotidiano dos atores sociais. O pesquisador etnográfico pode, por meio dos instrumentos metodológicos, agir em forma de imersão nesse campo, tomando notas sobre os relacionamentos, atividades e buscar compreender fenômenos que ocorrem nesse ambiente social formado no espaço virtual.

Em tal contexto, Kitchin (1998) categoriza da seguinte forma os fenômenos do espaço virtual da Internet:

1. A mudança do papel de tempo e espaço;
2. A mudança do processo de comunicação entre os atores e da comunicação de massa;
3. O questionamento dos dualismos como o real e o virtual, a verdade e a ficção, o autêntico e o artificial, a tecnologia e a natureza;
4. A representação e a realidade.

A proposta de uma etnografia virtual assume, como ponto de partida, duas perspectivas, segundo Hine (2000): a Internet como cultura e como artefato cultural. Nessa dupla abordagem, alguns conjuntos de questões impõem-se ao pesquisador, a começar pela necessidade de um deslocamento em relação ao entendimento da *etnografia situada* no espaço e da temporalidade como ferramenta fundamental da antropologia.

O processo de imersão e engajamento são as duas condições básicas do processo etnográfico, sem as quais o trabalho de campo se revela como pura descrição ou imposição.

Com base em uma redefinição do trabalho etnográfico, a observação *participante virtual* passa necessariamente pela consideração do contexto no qual se desenvolvem novas formas de sociabilidade. A autora resumiu em uma série de princípios a sua proposta de etnografia virtual:

1. O deslocamento da noção de *campo* para a noção de *campo de relações*;
2. A exploração da constituição de fronteiras e de conexões, especialmente entre o *virtual* e o *real*, com o auxílio de ferramentas de análise discursiva e pragmática;
3. O envolvimento intenso e não extensivo com a interação social prática e mediada, com base no entendimento da dimensão pragmática das práticas de linguagem em ambientes virtuais;
4. O caráter parcial e intersticial da etnografia virtual: diferente da pretensão da etnografia tradicional, que postula a descrição de informantes, lugares e culturas delimitadas no espaço e no tempo, e apreendidas em sua totalidade.

Quanto à importância do estudo da dinâmica social que ocorre dentro e fora da Internet, Hine (2000, p.60) afirma que

Um complemento útil para os estudos online que tratam a Internet como uma esfera cultural separada seria conduzir um estudo contextual das formas através das quais a Internet é articulada internamente e transforma relacionamentos off-line. Isto permitirá um sentido muito mais rico dos usos da Internet e das formas nas quais os relacionamentos locais moldam o seu uso como tecnologia e como contexto cultural.

Logo a seguir, os dez princípios da etnografia virtual, proposta por Hine (2000):

1. A presença prolongada do pesquisador no campo, combinada com o envolvimento intenso com o cotidiano dos habitantes do local de coleta de dados, para fazer o tipo especial de conhecimento, é o que pode ser chamado de fazer etnográfico. O pesquisador é capaz de usar essa interação sustentada para reduzir a confusão que as pessoas de outras formas de vida podem evocar (GEERTZ, 1993, p.16). Ao mesmo tempo, a etnografia pode ser um dispositivo para induzir perplexidade que, mesmo deslocando-se o embotamento no sentido de familiaridade com o qual o mistério da própria capacidade de tolerância do homem, em relação à perspicácia do outro, está escondido em cada um. Etnografia virtual é usada como um dispositivo para tornar o uso da Internet um problema: ao invés de ser inerentemente sensível, ela adquire a sua sensibilidade em uso. O estatuto da Internet como meio de comunicação, como objeto dentro da vida das pessoas e como local para a comunidade com formações, é alcançado e sustentado nas formas em que é utilizado, interpretado e reinterpretado.
2. A mídia interativa fornece um desafio e uma oportunidade para a etnografia, pelo fato de colocar em questão o local de interação. O

ciberespaço não é para ser pensado como espaço independente de qualquer ligação à vida real e presencial. Tem ricas e complexas conexões com os contextos em que ela é usada. Também depende de tecnologias que são utilizadas e entendidas tanto como cultura quanto artefato cultural. A concentrar-se em um ou outro aspecto que redundando na exclusão do outro, leva a uma visão empobrecida.

3. O crescimento da interação mediada torna desnecessário para a etnografia de ser pensada como localizada em determinados lugares, ou mesmo em multilocalis. A investigação do fazer e refazer do espaço por meio de interações mediadas é uma grande oportunidade para a abordagem etnográfica. Pode-se pensar na etnografia da interação mediada como móvel, em vez de multilocal.
4. Como consequência, o conceito do local do campo de investigação é colocado em questão. Se a cultura e a comunidade não são muito bem localizadas em determinado local, então não é etnografia. O objeto de investigação etnográfica pode ser útil para ser remodelado, concentrando-se no fluxo e conectividade, em vez de localização e de fronteira como o princípio organizador.
5. Fronteiras não são assumidas a priori, mas exploradas pelo curso da etnografia. O desafio da etnografia virtual é explorar a tomada de limites e a realização de conexões, nomeadamente entre o virtual e o real. Junto com isso vai o problema de saber quando parar, ou seja, definir os limites. Se o conceito de etnografia (e/ou cultura) como tendo limites naturais é abandonado para fins de análise, também se pode abandonar a ideia de uma etnografia de todo determinado objeto. Interromper a etnografia torna-se

uma decisão pragmática. O objeto etnográfico, propriamente dito, pode ser reformulado com cada decisão, quer seja continuar em direção a outra conexão quer retroceder ao ponto anterior. Praticamente é limitado por restrições do pesquisador incorporadas em tempo, espaço e engenhosidade.

6. Junto com o deslocamento espacial vem o deslocamento temporal. Engajamento com os contextos mediados é intercalado com interações em outras esferas e com outros meios de comunicação. Etnografia virtual é intersticial, uma vez que se encaixa em outras atividades de ambos os pesquisadores e assuntos. Imersão nas configurações só é conseguida de forma intermitente.
7. Etnografia virtual é necessariamente parcial. Uma descrição holística de um informante, localização ou cultura é impossível de alcançar. A noção de pré-existência, isolamento e descrição do informante, lugares e culturas é anulada. Os dados podem ser baseados em ideias de relevância estratégica, em vez de representação fiel da realidade objetiva.
8. Etnografia virtual implica envolvimento intenso com interação mediada. Esse tipo de engajamento acrescenta uma nova dimensão para a exploração do uso do meio no contexto. Envolvimento do pesquisador com o meio é fonte valiosa de discernimento. Etnografia virtual pode ser útil se recorrer ao pesquisador como informante e abraçar a dimensão reflexiva. A formação de interações com informantes pela tecnologia faz parte da etnografia, como são as interações do pesquisador com a tecnologia.
9. Novas tecnologias de interação tornam-se possíveis tanto para os informantes que se ausentam quanto para torná-los presentes na etnografia. Da mesma forma, o pesquisador é ao mesmo tempo ausente e presente

diante dos informantes. A tecnologia permite que esse relacionamento seja fugaz ou sustentado e sua relação perpassa a dimensão temporal e espacial. Todas as formas de interação são etnograficamente válidas, não apenas a presencial. A concepção do objeto etnográfico é possível por meio das tecnologias disponíveis. Essa etnografia ocorre *dentro e por meio do* virtual.

10. Etnografia virtual não é apenas virtual, no sentido estrito da palavra. A virtualidade também carrega uma conotação de realidade. Etnografia Virtual é adequada para a finalidade prática de explorar as relações de interação mediada, mesmo que não exatamente no contexto real em termos puramente metodológicos; é uma adaptação da etnografia estabelecida para se adequar às condições em que se encontra.

Em se tratando do princípio 10, a etnografia sempre foi adaptada para as condições em que se encontra. Isso pode ajudar a explicar a relutância tradicional de pesquisadores para dar aconselhamento aos outros pesquisadores prestes a iniciar o trabalho de campo. Não há um conjunto de regras a seguir a fim de realizar a etnografia de forma perfeita, bem como é inútil qualquer tentativa de definir os elementos fundamentais da abordagem etnográfica. O foco da etnografia tem de ser sobre a questão da adaptação às exigências da cultura e das possibilidades contidas nos pressupostos anteriores. Na etnografia virtual, a adaptação da metodologia à circunstância suscita questões que os princípios de 1 a 9 contemplam.

Diante do exposto, parece haver uma contradição. Se houver uma adesão ao princípio 10, então os outros nove princípios são minados, uma vez que para ser adaptativa e adequada ao objetivo se torna um problema. Há um deslocamento temporal nessa situação. A maioria dos leitores da etnografia irá reconhecer o produto escrito de um estudo etnográfico como uma construção depois da ocorrência do evento, o produto de uma sobreposição, mas em

grande parte linear no processo de planejamento, coleta de dados, análise e escrita. O produto escrito raramente reflete essa sequência de eventos e considerações metodológicas que surgiram durante a fase da coleta de dados, que podem ser apresentados como precedentes e até mesmo justificar as decisões que lhes deram origem.

Nesse sentido, o princípio 10 é o princípio fundamental subjacente aos demais, tornando-os possíveis. A adaptação e a interrogação mantêm a etnografia contextualizada e relevante.

Miller e Slatter (2000) apresentam outra forma de trabalho de campo atual e virtual, em seus estudos realizados em cibercafés em Trinidad Tobago. Para os autores, o problema principal é o do estabelecimento de uma etnografia que assuma como foco a incorporação da Internet à cultura material de uma dada coletividade.

Ao contrário de Hine, os autores criticam a distinção entre online e off-line, argumentando que se trata de uma reificação analítica. Para recolocar o contexto no estudo da Internet, afirmam os autores, é preciso superar dualidades, relacionando contexto e fenômeno de forma recíproca.

Para uma etnografia do ciberespaço, Miller e Slater (2000, p.47) argumentam que:

O problema não é só a falta generalizada de envolvimento contextualizado e em longo prazo. Nem é simplesmente a fascinação com o virtual e o ciberespaço, que tem levado tantos pesquisadores a conduzir estudos inteiramente online, sobre os modos de interação e relacionamento específicos ao cenário online. O problema, ao contrário, é a falta de atenção às formas em que o objeto e o contexto precisam ser definidos em relação um ao outro para projetos etnográficos específicos. Às vezes, o uso da Internet parece constituir virtualidades, às vezes não.

A forma como os autores elaboram a crítica remete a duas ordens de problemas concernentes à etnografia virtual. A primeira relaciona-se com a imposição de modelos que partem do domínio teórico para o empírico, sem o cuidado particular de uma transposição ou,

em outros termos, do uso da sensibilidade antropológica no trabalho sobre as categorias nativas, no que elas podem vir a informar as escolhas teórico-metodológicas.

A segunda ordem de problemas emerge da proposta de simetria entre sujeitos e objetos, postulada pelos autores, com origem nos trabalhos de LATOUR; WOOLGAR, 1986; STRATHERN, 1992. A antropologia simétrica é evocada com o objetivo de sustentar a crítica à separação entre a tecnologia e o seu contexto social, implicando uma desconsideração das especificidades do virtual, enquanto dimensão potencializadora de uma nova ordem de experiências intersubjetivas. A distinção entre o virtual e o atual, entre o online e o off-line, pode ser mais bem expressa a partir tanto das definições de Hine como, em parte, das críticas dirigidas por Miller e Slater.

No tocante à aplicação desse método de investigação nas CVAO, podem-se assumir como virtual⁴⁹ as distintas formas, que são geradas com o estabelecimento de outros laços sociais. A insistência de Miller e Slater (2000, p.43) em um emprego ortodoxo da etnografia como “observação participante relacionada com o tempo passado na comunidade ou no espaço social estudado” revela-se ineficaz, dada a incontornável característica da agência social na rede de computadores: flexível, inconstante e de grande mobilidade.

Herring (1996) afirma que a participação ativa na comunidade virtual permite que se observem as atitudes dos participantes na busca de solução de problemas, no sentido de experienciar a efetividade da comunidade. Essa é uma visão particular do autor, porém não pode ser adotada como única, tendo em vista que Hine (2000) afirma que o pesquisador pode

⁴⁹ No sentido filosófico é aquilo que existe em potência e não em ato, o campo de forças e de problemas que tende a resolver-se em uma atualização. O virtual encontra-se antes da realização efetiva ou formal – a árvore está virtualmente presente no grão. O virtual não se opõe ao real, mas sim ao atual: virtualidade e atualidade são apenas dois modos diferentes da realidade. (...) Acrescentamos que as atualizações de uma mesma entidade virtual podem ser bastantes diferentes uma das outras, e que o atual nunca é completamente determinado pelo virtual. Assim, de um ponto de vista acústico e também semântico, nenhuma atualização da palavra se parece exatamente com nenhuma outra, e há pronúncias – nascimento de novas vozes – ou sentidos – invenção de novas frases – imprevisíveis que, no entanto, podem sempre aparecer. O virtual é uma fonte indefinida de atualizações (LÉVY, 1997, p. 47)

desenvolver a etnografia virtual, como um “*lurker*”, ou seja, como um participante observador, já que a Internet é considerada cultura e um artefato cultural.

2.2 Teoria analítica da ação mediada como método de análise

Considerando os fóruns de discussão das CVAO como unidade de análise e sua dinâmica no processo de discussão, verifica-se o papel do **aprendiz agindo (aprendendo) com o objeto de mediação**, ou seja, o conjunto **pessoa - ação de aprendizagem – objeto de mediação** (WERTSCH, 1991).

Adequando mais especificamente o conceito proposto ao objeto de investigação, tem-se: **membro da comunidade virtual – solução de problemas – objeto de aprendizagem como instrumento**.

Abaixo, figura esquemática que demonstra como ocorre a relação proposta por WERTSCH (1991) com o contexto investigado, que envolve: membros da comunidade, solução de problemas e objeto de aprendizagem:

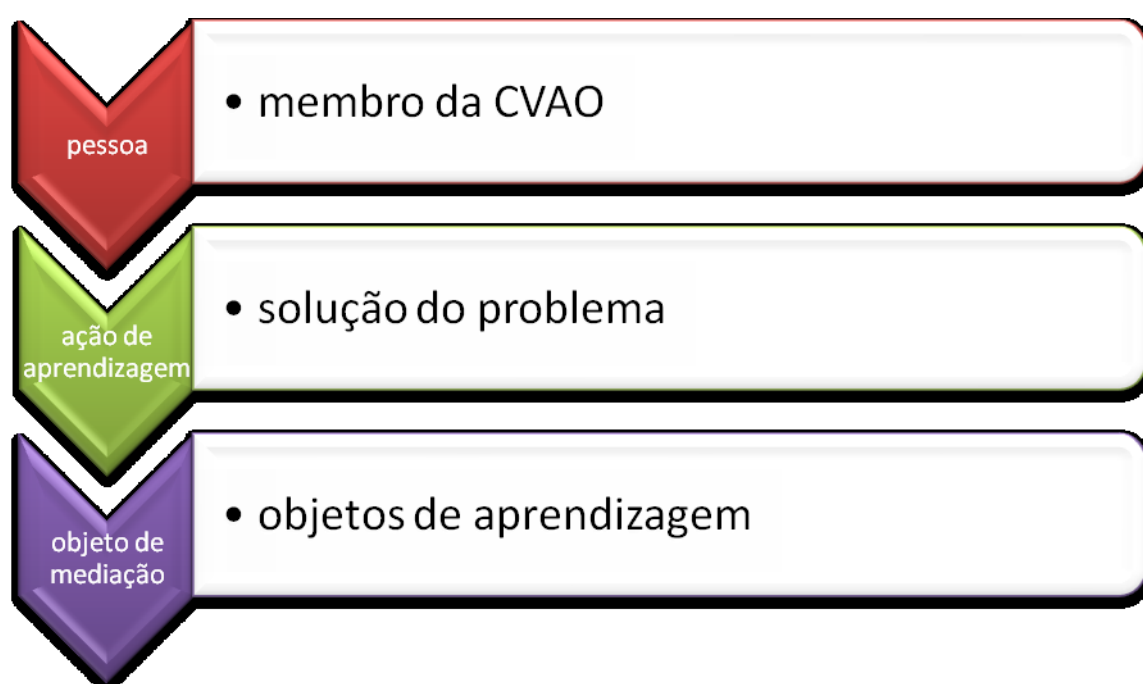


Figura 10- Relação entre a proposta metodológica e o contexto investigado

Wertsch (1991) ao analisar um diálogo, identifica as pessoas que estão dialogando entre si. A apresentação do diálogo de forma escrita é o objeto de mediação, que é todo o diálogo escrito com a tentativa, por meio do uso das palavras e das pontuações, de demonstrar com a maior fidedignidade possível as emoções, sentidos e intenções que foram expressas pelas pessoas no momento da ocorrência daquele diálogo na forma verbal.

O diálogo pode também existir somente na forma escrita sem nunca ter ocorrido na forma oral. O leitor ao ler o diálogo na forma escrita, utilizando-se de todos os recursos gramaticais e linguísticos, pode construir em sua mente uma imagem real e concreta daquele diálogo na forma oral.

Nesses termos, procura-se utilizar o mesmo modelo referencial de análise do diálogo proposto por WERTSCH (1991), para a análise dos fóruns de discussão, os quais não deixam de ser uma forma de diálogo, na forma escrita, mas de forma interativa, por ser suportado pelas tecnologias da informação e comunicação – TIC. Tal tipo de diálogo (fóruns de discussão), que é um dos principais instrumentos das CVAO, para a solução de problemas, pode ser analisado da mesma forma que os diálogos pessoais foram analisados pelo autor.

Uma das diversas contribuições desta pesquisa é também fazer a equivalência do modelo, proposto pelo autor para análise de diálogos, com a análise dos fóruns de discussão. A figura abaixo demonstra, de forma esquemática, a equivalência construída.

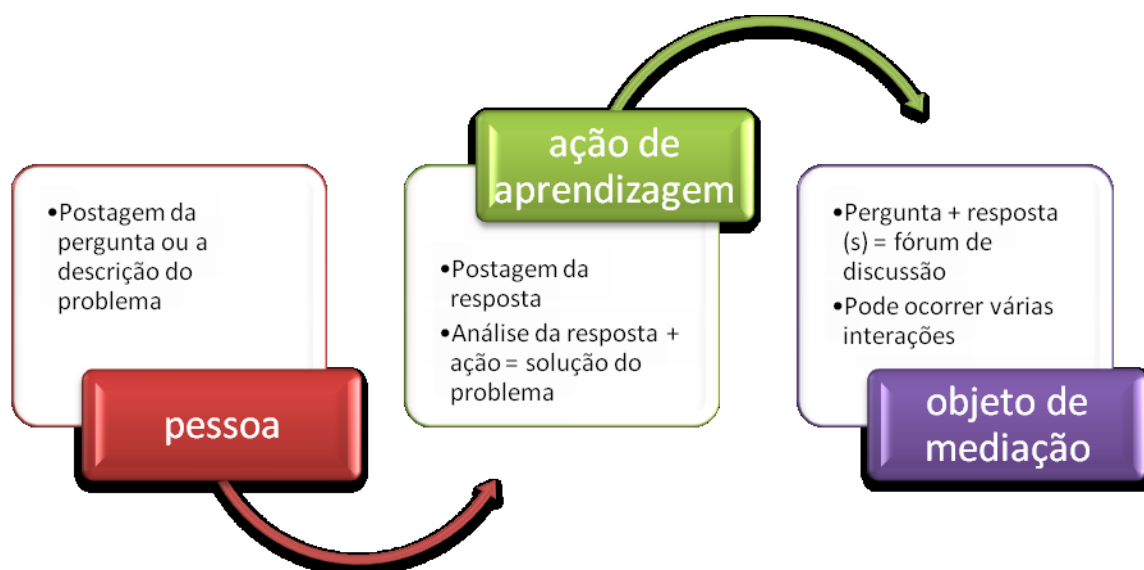


Figura 11 - Equivalência do modelo no contexto do fórum de discussão

2.3 *Análise documental*

A interação intermediada da análise qualitativa dos dados é examinada por padrões ou categorias (BARDIN, 2000). Para a análise de conteúdo referente às discussões documentadas em formato digital utiliza-se a técnica categorial temática como forma de interpretar e desvelar os conteúdos manifestos e simbólicos nos enunciados. As categorias são construídas, conectadas e verificadas também quanto à sua intensidade, sempre como fator analítico dos dados que estão sendo analisados.

O ponto de partida da **Análise de Conteúdo (AC)** é a mensagem que expressa um significado e um sentido. De acordo com Bardin (2000, p.9-10), a análise de conteúdo pode ser entendida

Como um conjunto de instrumentos metodológicos cada vez mais sutis em constante aperfeiçoamento, que se aplicam a discursos (conteúdos e continentes) extremamente diversificados. O fator comum destas técnicas múltiplas e multiplicadas é uma hermenêutica controlada, baseada na dedução: a inferência. Enquanto esforço de interpretação, a análise de conteúdo oscila entre dois polos do rigor da objetividade e da fecundidade da

subjetividade. Absolve e cauciona o investigador por esta atração pelo escondido, o latente, o não-aparente, o potencial do inédito (do não-dito), retido por qualquer mensagem. (...) Ao desempenharem o papel de técnicas de ruptura face à intuição aleatória e fácil, os processos de análise de conteúdo obrigam à observação de um intervalo de tempo entre o estímulo-mensagem e a reação interpretativa. Se este intervalo de tempo é rico e fértil então, há que recorrer à análise de conteúdo.

Nesse sentido, a AC assenta-se nos pressupostos de uma concepção crítica e dinâmica da linguagem, entendida como *“uma construção real de toda a sociedade e como expressão da existência humana que, em diferentes momentos históricos, elabora e desenvolve representações sociais de dinamismo interacional que se estabelece entre linguagem, pensamento e ação.”* (FRANCO, 2005, p.14). Esses pressupostos fogem da concepção formalista da linguagem. O significado de um objeto, que se manifesta nas representações sociais, cognitivas, valorativas e emocionais contextualizadas, pode ser compreendido e analisado pelo seu *corpus* de significação e pelas suas características definidoras.

A organização da análise contempla três fases: a pré-análise, a exploração do material e o tratamento dos resultados, a inferência e a interpretação. A AC como procedimento de pesquisa contribui para indagação e análise de mensagens que produzam *inferências* e relevância teórica sobre os elementos básicos da comunicação (fonte emissora, processo codificador, detector ou recipiente da mensagem). Ou seja, no caso desse estudo, a AC representa um instrumento importante para o tratamento cuidadoso das mensagens, nos aspectos de descrição e interpretação dos elementos que compõem o processo de utilização e apropriação de objetos de aprendizagem pelos membros das comunidades virtuais auto-organizadas no ambiente virtual, em um contexto histórico, cultural e institucional.

Outra questão importante refere-se à definição da **Unidade de Análise (UA)**, a qual segundo Bardin (2000) divide-se em **Unidades de Registro (UR)** e **Unidades de Contexto (UC)**.

A **UR** é a menor parte do conteúdo, cuja ocorrência é registrada de acordo com as categorias levantadas e é representada pelos seguintes tipos: “a palavra”, “o tema”, “o objeto ou referente”, “o personagem”, “o acontecimento” e “o documento”. Franco (2005) alerta que os tipos de UR, embora apresentados como *unidades estanques possam ser utilizadas de forma compartilhada e inter-relacionadas*. (p.42).

Pela característica do presente estudo, optou-se pela **Unidade de Contexto (UC)**, que é

a parte mais ampla do conteúdo a ser analisado, porém, é indispensável para a necessária análise e interpretação dos textos a serem decodificados (tanto do ponto de vista do emissor, quanto do receptor) e principalmente, para que se possa estabelecer a necessária diferenciação, resultante dos conceitos de significado e de sentido, os quais devem ser respeitados, quando da análise e interpretação das mensagens disponíveis (FRANCO, 2005, p.43).

As **UC** incorporam as **UR** e nesse trabalho são relatadas sob a forma de um conjunto de tabelas, que representam a ação dos membros das comunidades virtuais auto-organizadas e sua relação com os objetos de aprendizagem, a categorização dos problemas, o encaminhamento e a situação.

Define-se *categoria* como “uma operação de classificação de elementos constitutivos de um conjunto, por diferenciação e, seguidamente, por reagrupamento segundo o gênero (analogia), com os critérios previamente definidos.” (BARDIN, 2000, p.117).

As categorias analisadas emergiram da análise dos discursos dos participantes das CVAO, citadas nos fóruns e listas de discussões coletados, que estão representadas no item 3.3.1.

2.4 Critérios de seleção das comunidades virtuais

Para a seleção das comunidades virtuais foram criados seis quesitos que compõem os critérios de seleção das CVAO.

Os critérios adotados para a seleção são os seguintes:

1. Que sejam comunidades que suportam SLCA já adotados e consagrados pelo mercado;
2. Que o conjunto de SLCA forme uma arquitetura integrada, composta por:
 - a. Um sistema operacional;
 - b. Um gerenciador de banco de dados;
 - c. Um gerenciador de servidor Web;
 - d. Uma linguagem de programação;
3. Que o conjunto de SLCA ofereça todos os recursos e funcionalidades para o desenvolvimento de praticamente qualquer tipo de aplicação de sistema de informação baseado na Web;
4. Que as comunidades virtuais auto-organizadas estejam em funcionamento por um período mínimo de dois anos;
5. Que as comunidades virtuais auto-organizadas tenham disponíveis, para livre acesso, todos os arquivos históricos das discussões, para consulta e análise;
6. Que as comunidades virtuais auto-organizadas não tenham, de forma explícita, qualquer tipo de restrição à intervenção e uso dos dados publicados para finalidades de investigação científica.

Capítulo III

3 RESULTADOS E DISCUSSÃO

O desenvolvimento da pesquisa segue os procedimentos descritos na página 34.

3.1 *Seleção das comunidades virtuais*

Após a aplicação dos seis quesitos que compõem os critérios de seleção das CVAO, procede-se a análise das comunidades selecionadas.

3.1.1 **Comunidades selecionadas**

As comunidades selecionadas para o propósito desta investigação formam a chamada arquitetura LAMP⁵⁰ (TAURION, 2004), que é composta pelos seguintes softwares livres e de código aberto: Linux, Apache, MySQL e a combinação das linguagens PHP, Python e Perl. A comunidade Sourceforge também é alvo de investigação, tendo em vista a sua relevância como incubadora e diretório dos projetos de software livre e de código aberto.

A arquitetura LAMP é composta por um sistema operacional (Linux), por um gerenciador de banco de dados (MySQL), por um gerenciador de servidor Web (Apache) e por linguagens de programação (PHP, Python e Perl). Esse conjunto de softwares preenche todos os requisitos constantes nos critérios para seleção das comunidades citados no item 3.4.

Logo a seguir, demonstra-se esquematicamente a arquitetura LAMP. Com esses quatro recursos, é possível desenvolver qualquer aplicação de um sistema de informação, com acesso e operação de suas funções, por meio da Web.

⁵⁰ LAMP é uma sigla formada pelas primeiras letras dos softwares: Linux, Apache, MySQL, PHP/Python,Perl.

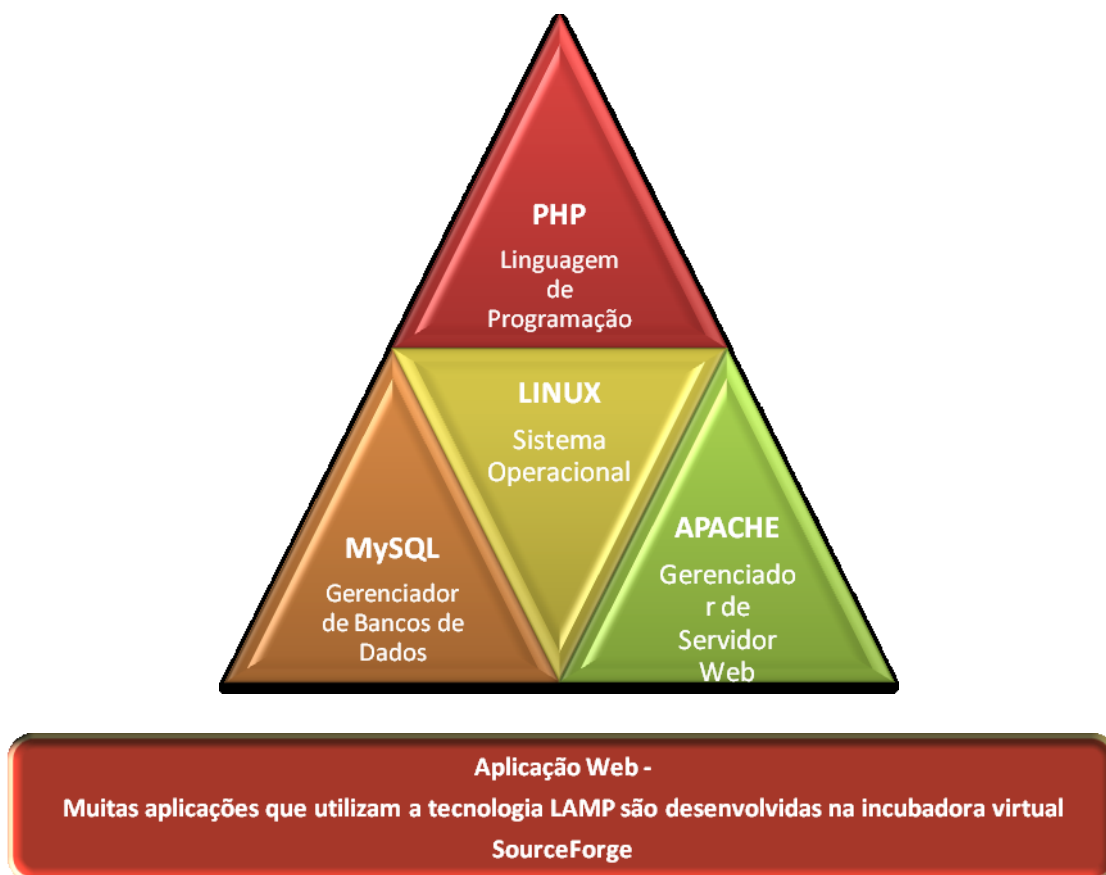


Figura 12 – Figura esquemática da arquitetura LAMP

Vale ressaltar que tais comunidades não são comunidades virtuais de aprendizagem no sentido *stricto* do termo, ou seja, vinculadas a atividades acadêmicas formais, mas sim comunidades de aprendizagem voltadas para a solução de problemas relacionados ao desenvolvimento e uso de software que formam a arquitetura LAMP, inseridas em um contexto do mundo real, mediado por um conjunto de recursos da tecnologia da informação e comunicação.

A seleção e qualificação das CVAO, segundo os critérios definidos, são demonstradas no quadro 4.

Quadro 4 - Seleção e qualificação das CVAO e respectivos endereços URL

Comunidades	Linux	Apache	MySQL	PHP
Critérios de Seleção				
a. Que sejam comunidades que suportam SLCA já adotadas e consagradas pelo mercado.	SIM	SIM	SIM	SIM
b. Que o conjunto de SLCA forme uma arquitetura integrada por: b.1 Sistema Operacional b.2 Gerenciador de Servidor Web b.3 Gerenciador de Banco de Dados b.4 Linguagem de Programação	X	X	X	X
c. Que o conjunto de SLCA ofereça todos os recursos e funcionalidades para o desenvolvimento de praticamente qualquer tipo de aplicação de sistema de informação baseado na Web.	SIM	SIM	SIM	SIM
d. Que as CVAO estejam em funcionamento por um período superior a dois anos.	SIM	SIM	SIM	SIM
e. Que as CVAO tenham disponíveis, para livre acesso, todos os arquivos históricos das discussões para consulta e análise.	Todos os arquivos disponíveis	Todos os arquivos disponíveis	Todos os arquivos disponíveis	Todos os arquivos disponíveis
f. Que as CVAO não tenham, de forma explícita, qualquer tipo de restrição à intervenção e uso dos dados publicados para finalidades de investigação científica.	Sem nenhuma restrição	Sem nenhuma restrição	Sem nenhuma restrição	Sem nenhuma restrição
g. Endereço da URL da Comunidade	http://www.kernel.org http://tux.org/lklm/	http://apache.org/	http://www.mysql.com/	http://www.php.net
h. Característica do instrumento de discussão	Lista de discussão	Lista de discussão	Fórum de discussão	Lista de discussão

3.2 Levantamento de dados

O levantamento de dados foi efetuado nos fóruns e listas de discussões das CVAO selecionadas, conforme quadro 5.

O período de coleta de dados foi realizado em duas fases, sendo um período no início e final do ano de 2008 e início de 2009, conforme quadro 5.

Quadro 5 - Período de coleta de dados por comunidade

Comunidade	Período	
	Início	Término
Linux	31/01/2008	06/02/2008
Apache	08/01/2008	15/04/2008
MySQL	16/02/2009	01/03/2009
PHP	24/12/2008	26/02/2009
SourceForge ⁵¹	01/2008	01/2008

Ressalta-se que as interações ocorridas, por meio dos fóruns e listas de discussão, foram extraídas de forma aleatória, no período selecionado (2008 /2009). Tendo em vista a utilização da etnografia virtual para observação, acompanhamento e análise do funcionamento das CVAO, assume-se a postura de participante ativo no desenvolvimento do processo de ação e movimento dos sujeitos, procurando apreender o sentido, o significado, a mediação, a comunicação, o comportamento, a aprendizagem e a solução de problemas no contexto das CVAO.

⁵¹ A Comunidade SourceForge não apresenta nas discussões dos fóruns a data e horário da postagem da mensagem, apenas coloca a informação em tempo relacionado ao dia em que a postagem foi efetuada, p.e. 1 mês atrás, 6 meses atrás, 1 ano atrás. É uma limitação grave, pois se o fórum for analisado fora do contexto da web site, fica impossível precisar a data em que a postagem fora efetuada.

3.3 *Análise dos dados*

3.3.1 *Análise de conteúdo e categorização*

As unidades de conteúdo – UC (BARDIN, 2000) são classificadas, por comunidade, usando a seguinte codificação, conforme quadro seguinte:

Quadro 6 - Composição do código para o processo de discussão

9	Número sequencial das interações do fórum relacionado ao tipo de problema.
99	Quantidade de interações que ocorreram na discussão.
XXX	Abreviação da comunidade, conforme tabela 7

Composição do código: 9.99.XXX – Exemplo: **1.20.LNX** – O significado desse código é: o primeiro conjunto de interações, composto por (20) vinte interações de discussão ocorridas na comunidade Linux.

Abaixo quadro 7 que demonstra a abreviação das comunidades:

Quadro 7 - Abreviação dos nomes das comunidades

Abreviação	Comunidade
LNX	Linux
APH	Apache
SQL	MySQL
PHP	PHP
SFG	SourceForge

Os quadros de classificação das CVAO são compostas pela unidade de contexto – UC devidamente identificada por um código e associada à relação dos fóruns de discussão apresentados nos anexos de F a J, contemplando os seguintes elementos: período da coleta dos dados, tipo de problema apresentado na discussão, encaminhamento da solução e a situação (*status*) do problema.

Os resultados da análise de contexto das CVAO são apresentados nos quadros de 8 a 13, que revelam tipo de problema, respectivo encaminhamento da solução, período de coleta de dados e situação. Para esta análise foram criadas categorias que explicitam os dados da unidade de contexto – UC (composta por: ordem da interação, quantidade de interações ocorridas e designação codificada da CVAO). As tabelas de 1 a 6 contêm resumo quantitativo de ocorrências das discussões por comunidade. A tabela 6 apresenta um resumo analítico quantitativo das interações dos fóruns e listas de discussão, bem a solução de problemas. A tabela 7 apresenta um resumo quantitativo da situação dos problemas apresentados por comunidade.

Quadro 8 - Tipo de problema, respectivo encaminhamento da solução e situação - Comunidade Linux

UC	Período	Tipo de problema	Encaminhamento da solução	Situação
1.8. LNX	31/01/2008 a 06/02/2008	Obter informações sobre fóruns, cujos membros discutem, exclusivamente, problemas sobre software de código aberto.	Membro da comunidade responde.	Resolvido
2.30. LNX	31/01/2008 a 06/02/2008	Obter orientações de como fazer a instalação de um determinado tipo de software.	Membro da comunidade responde.	Resolvido

Tabela 1 - Problemas e interações da comunidade Linux – Resumo de Ocorrências

Problemas	Quantidade	%
Resolvidos	02	100
Pendentes	00	0
Descontinuado	00	0
Total de problemas postados		02
Total de interações		38
Média de interações		16
Membros envolvidos		08

Quadro 9 - Tipo de problema, respectivo encaminhamento da solução e situação - Comunidade Apache

UC	Período	Tipo de problema	Encaminhamento da solução	Situação
1.7. APH	02/04/2008 a 17/04/2008	Discussão e orientação sobre a definição do tipo de uso de licença de software	Membro da comunidade	Resolvido
2.2. APH	15/04/2008 a 15/04/2008	Troca de ideias sobre treinamento	Membro da comunidade	Concluído
3.3. APH	08/01/2008 a 10/01/2008	Troca de ideias sobre um artigo	Membro da comunidade	Concluído

Tabela 2 - Problemas e interações da Comunidade Apache – Resumo de Ocorrências

Problemas	Quantidade	%
Resolvidos	03	100
Pendentes	00	0
Descontinuado	00	0
Total de problemas postados		03
Total de interações		12
Média de interações		04
Membros envolvidos		12

Quadro 10 - Tipo de problema, respectivo encaminhamento da solução e situação - Comunidade MySQL

UC	Período	Tipo de problema	Encaminhamento da solução	Situação
1.4. SQL	24/02/2009 a 01/03/2009	Configuração do banco de dados	Membro da comunidade	Resolvido
2.4. SQL	16/02/2009 a 25/02/2009	Remoção e re-instalação do banco de dados	Membro da comunidade	Resolvido

Tabela 3 – Problemas e interações da Comunidade MySQL – Resumo de Ocorrências

Problemas	Quantidade	%
Resolvidos	02	100
Pendentes	00	0
Descontinuado	00	0
Total de problemas postados		02
Total de interações		08
Média de interações		04
Membros envolvidos		04

Quadro 11 - Tipo de problema, respectivo encaminhamento da solução e situação - Comunidade PHP

UC	Período	Tipo de problema	Encaminhamento da solução	Situação
1.9. PHP	24/12/2008 a 25/12/2008	Compartilhamento de memória	Membro da comunidade	Resolvido
1.12. PHP	21/02/2009 a 26/02/2009	Atualização de nova versão de desenvolvimento, para correção de um erro	Membro da comunidade	Resolvido

Tabela 4 – Problemas e interações da Comunidade PHP – Resumo de Ocorrências

Problemas	Quantidade	%
Resolvidos	02	100
Pendentes	00	0
Descontinuado	00	0
Total de problemas postados		02
Total de interações		21
Média de interações		10.5
Membros envolvidos		08

Quadro 12 - Tipo de problema, encaminhamento da solução e situação - Comunidade Sourceforge

UC	Data	Tipo de problema	Encaminhamento da solução	Situação
1.4. SFO		Obter informações sobre fóruns, cujos membros discutem, exclusivamente, problemas sobre software de código aberto.	Membro da comunidade indica relação de web sites, que discutem o assunto.	Resolvido.
2.2. SFO		Obter orientações de como fazer a instalação de um determinado tipo de software.	Membro da comunidade responde, solicitando mais informações, para poder ofertar a orientação, mas o solicitante não voltou a interagir.	Descontinuado.
3.2. SFO		Obter orientações sobre os procedimentos de como abrir um determinado arquivo.	Membro da comunidade responde por meio de uma orientação específica.	Resolvido.
4.1. SFO		Obter informações sobre licenças de um tipo específico de software (Clmg).	Nenhum membro da comunidade respondeu a pergunta.	Pendente.
5.1. SFO		Obter informações sobre os procedimentos técnicos de como fazer a remoção de dependências entre dois tipos específicos de software (Clmg e ImageMagick).	Nenhum membro da comunidade respondeu à pergunta.	Pendente.

Tabela 5 – Problemas e interações da Comunidade Sourceforge – Resumo de Ocorrências

Problemas	Quantidade	%
Postados	05	100
Resolvidos	02	40
Pendentes	02	40
Descontinuado	01	20
Total de problemas postados		05
Total de interações		10
Média de interações		2
Membros envolvidos		6

A seguir demonstra-se, por meio da tabela 6, um resumo analítico de caráter quantitativo referente às interações dos fóruns e listas de discussão e solução de problemas, por comunidade.

Tabela 6 - Tabela resumo analítico quantitativo – interações dos fóruns e listas de discussão e solução de problemas

Comunidade	Quantidade						
	Problemas	%	Interações	%	Média	Membros	%
Linux	02	14.3	38	42.7	16	08	21.5
Apache	03	21.4	12	13.5	04	12	31.6
MySQL	02	14.3	08	9	04	04	10.5
PHP	02	14.3	21	23.6	10.5	08	21.5
SourceForge	05	35.7	10	11.2	02	06	15.8
Total	14	100	89	100	6,35	38	100

Como pode ser observado na tabela acima, são coletados 14 problemas, durante a primeira e a segunda fase. Desse total, a Comunidade SourceForge representa 35.7% dos total de problemas coletados, seguida da Comunidade Apache com 21.3%, e as demais comunidades com 14.3%. Já no que se refere à quantidade de interações por problema a Comunidade Linux, com apenas 02 (dois) problemas postados, apresenta 38 interações, o que representa 42.7% do total de interações de todas as outras comunidades. A Comunidade PHP apresenta 23.6%, ficando em segundo lugar em relação à quantidade total de interações das 5 (cinco) comunidades analisadas. Como resultado, a Comunidade Linux apresenta uma média de 16 interações por problema, contrastando com a Comunidade SourceForge que apresenta uma média de apenas 2 (duas) interações por problema. A Comunidade PHP apresenta uma média de 10.5 interações por problema, mantendo uma coerência com o indicador anterior. Pode-se concluir, baseando-se nesses dados, que o nível de complexidade dos problemas

postados na Comunidade Linux é mais elevado do que dos problemas postados na Comunidade SourceForge.

No que se refere à participação de diferentes membros da comunidade envolvidos na discussão, os dados reforçam a análise, acima citada, pois enquanto 08 (oito) membros produziram 38 (trinta e oito) interações na Comunidade Linux e 21 (vinte e um) na Comunidade PHP, 6 (seis) membros da Comunidade SourceForge produziram 10 (dez) interações e 12 (doze) membros produziram 12 (doze) interações na Comunidade Apache. Indiscutivelmente, a Comunidade Linux apresenta-se como a CVAO mais ativa e discute problemas de natureza mais complexa, seguida pela Comunidade PHP.

Considerando o total de 14 problemas postados, durante o período da coleta de dados, dispostos na tabela 19, verifica-se que 78.6% dos problemas postados são resolvidos, ou seja, obtiveram êxito no resultado da aplicação da solução apresentada por membros da comunidade. A CVAO SourceForge apresenta resultado diferente das demais, pois apresenta um índice de 40% de problemas resolvidos, 40% de problemas pendentes e 20% descontinuados.

Diante dos dados apresentados, a Comunidade *SourceForge*, por ser incubadora de desenvolvimento de software, possui nível diversificado de problemas postados, que são em sua maioria oriundos de potenciais usuários dos softwares produzidos por meio da incubadora. Observa-se que, em razão da diversificação do interesse dos usuários, a quantidade de problemas em um mesmo período de tempo é maior que nas outras comunidades, mas, por sua vez, o nível de interesse em solucionar o problema é aparenta ser menor na amostra coletada.

Abaixo a tabela 7 demonstra o resumo da situação (status) dos problemas apresentados:

Tabela 7 - Tabela resumo - situação dos problemas apresentados por comunidade

Comunidade	Problemas	Resolvido	Quantidade				
			%	Pendente	%	Descontinuado	%
Linux	02	02	100	-	-	-	-
Apache	03	03	100	-	-	-	-
MySQL	02	02	100	-	-	-	-
PHP	02	02	100	-	-	-	-
SourceForge	05	02	40	02	40	01	20
Total	14	11	78.6	02	14.3	01	7.1

Diante do exposto, infere-se que os fóruns de discussão mais específicos, focados e uma comunidade mais ativa e motivada apresentam um comprometimento maior em solucionar problemas do que os fóruns genéricos e que possuem membros diversificados.

Com o objetivo de caracterizar os fóruns das CVAO analisadas e de acordo com o item 2.4.2, em que se descreve a taxonomia dos fóruns de discussão, é imperativo demonstrar a classificação dos fóruns das CVAO, conforme demonstrado no quadro 13:

Quadro 13 - Análise taxonômica dos fóruns das CVAO

Tipos	Classificação	
Suporte comunicativo	Fórum textual	X
	Fórum multimídia	
Temática	Fórum geral	
	Fórum colaborativo	X
Interação usuário-ferramenta	Uso passivo	X
	Uso ativo	X
Função investigadora	Fórum primário	
	Fórum secundário	X

Em resumo, os fóruns das CVAO analisados sob o ponto de vista da taxonomia (MUÑOZ, 2007) possuem suporte comunicativo textual, prevalecendo a temática colaborativa no processo de discussão. Quanto ao quesito interação usuário-ferramenta os fóruns possuem membros com comportamento ativos e passivos, ou seja, membros que colaboram ativamente e oferecem condições para que os sujeitos tenham acesso ao conhecimento gerado. Quanto à

função investigadora, prevalece a de natureza secundária por ser de fóruns de discussão pertencentes a comunidades que surgem espontaneamente.

3.3.2 Análise etnográfica

A maioria dos softwares que compõem a arquitetura LAMP surgiu em meados da década de 1990. Foi uma época muito profícua para as iniciativas de desenvolvimento de softwares abertos e livres. Hoje, muitos projetos de software começam o seu desenvolvimento no ambiente de uma incubadora virtual, sendo a mais popular a *SourceForge*.

A incubadora virtual oferece à equipe de desenvolvedores a infraestrutura necessária, para o desenvolvimento de software, sobretudo quanto ao gerenciamento do projeto, gerenciamento de tarefas, controle de versões, repositório de documentação, ferramentas para *download*, por parte dos potenciais usuários, fóruns e/ou listas de discussão para que os membros da equipe possam se relacionar, discutindo, trocando informações e conhecimento, voltados para a solução de problemas.

Uma vez o software na versão beta⁵² é liberado para uso do público em geral, ou seja, os interessados naquele tipo específico de software efetuam o *download*, fazem a instalação do software no ambiente desejado e começam a utilizá-lo. Esse usuário, ao utilizar tal versão, desempenha papel importante, pois o software ainda está em forma de protótipo, ou seja, não é efetivamente um produto acabado, testado em todas as suas funcionalidades.

No decorrer da utilização do software, o usuário pode encontrar diversos erros (*bugs*), que podem ser reportados pelo correio eletrônico, fóruns ou listas de discussão disponíveis na *web site* da incubadora. Essas informações são de extrema valia, pois apontam diversos problemas, erros ou inconsistências do software, o que facilita os desenvolvedores a corrigi-

⁵² Versão beta é a última versão de teste, antes da primeira versão.

los e aprimorá-los, cada vez mais. A incubadora publica estatística de erros reportados e corrigidos, por meio da sua *web site*. Dessa forma, existe uma correlação de forças orientadas para o processo colaborativo, ocorrendo uma intensa colaboração entre a equipe de desenvolvedores e os usuários do software, que ao utilizarem-no testam no mundo real.

É lançada a versão 1.0 quando o software atinge certa estabilidade. O primeiro dígito significa a versão do software e o segundo dígito a *release*⁵³. O número da versão é incrementado, quando ocorrem significativas mudanças na estrutura e funcionalidade do software. O número do release é incrementado quando incidem pequenas correções e/ou alterações em sua funcionalidade.

Ao longo do tempo, quando o software começa a ter certa popularidade, este inaugura uma comunidade própria, para os seus usuários. Os usuários fazem dessa comunidade o ponto de encontro para a discussão e solução de problemas, obtenção de informações ou até mesmo uma simples troca de ideias. O ponto forte da comunidade é o fórum ou lista de discussão, em que os usuários postam questões, para a solução de problemas diversos, que na maioria das vezes estão relacionados à operação do software, instalação, integração com outras tecnologias, etc. Quanto aos desenvolvedores, usualmente, utilizam o fórum da incubadora para a solução de problemas ligados ao desenvolvimento dele mesmo ou o do fórum específico da sua comunidade. É nesse ponto que as comunidades começam a operar, a se desenvolver e conseqüentemente a amadurecer.

Ao observar a *web site* das comunidades selecionadas, pode-se verificar que a comunidade *Sourceforge* e a comunidade *MySQL* possuem algumas características semelhantes no que se refere à:

1. Qualidade da interface gráfica, pois se verifica que é um web site bem elaborado;

⁵³ *Release* significa uma pequena adição de funcionalidades ou a correção de erros reportados.

2. A ferramenta de discussão é baseada em fóruns, ou seja, há facilidade na leitura das discussões, sobretudo pela forma de endentação das mesmas;
3. Ambas possuem uma versão comercial, que é apresentada em outro ambiente por meio de uma nova URL.

Com relação à web *site* das comunidades *Linux*, *PHP* e *Apache*, diz-se delas que possuem algumas características semelhantes no seguinte:

1. A interface da web *site* é simples e sem sofisticação em relação à interface gráfica, funcionalidade e diversidade de recursos;
2. A ferramenta de discussão é baseada em listas de discussão, pois as discussões são trocadas, ainda por e-mail. Torna-se difícil a leitura dos documentos produzidos pela discussão, pelo fato de estes apresentarem formas sequenciais e lineares, ao contrário da forma endentada dos fóruns de discussão.
3. A orientação aos seus usuários a procurarem as respostas nas FAQs antes de fazer qualquer pergunta.

Descrevem-se, a seguir, as principais características, funcionalidades e um breve histórico das CVAO, objeto da pesquisa, considerando os princípios da análise etnográfica virtual.

3.3.2.1 Comunidade Linux (kernel)

A comunidade objeto desta análise pode ser acessada por meio da URL <http://www.kernel.org/>⁵⁴, sendo que na URL <http://marc.info/?l=linux-kernel>⁵⁵ estão os arquivos das listas de discussões da comunidade.

⁵⁴ Acesso em: 15 mar. 2009.

⁵⁵ Acesso em: 15 mar 2009.

Linus Tovuards é o proprietário da marca Linux. Tal fato implica que, para a liberação de qualquer versão do Kernel do Linux, é necessário a aprovação pessoal do proprietário. Uma vez liberada uma versão, apenas quando esta atinge o nível de estabilidade necessário para o uso no mercado, é iniciado novo ciclo de desenvolvimento em que a comunidade discute e decide quais são as novas funcionalidades a ser desenvolvidas e incorporadas no Kernel.

Segundo Taurion (2004), a primeira versão do Linux foi a 0.01, sendo disponibilizado pelo seu criador Linus Tovuards, no ano de 1991. A primeira versão tinha somente 7.500 linhas de código no kernel. A versão 1.0, disponibilizada no ano de 1994, já possuía em torno de 158.000 linhas de código. Em 1996, com o lançamento da versão 2.0, o Kernel do Linux era composto de 649.000 linhas de código. A última versão, lançada em 2003, já tinha mais de 4.200.000 linhas de código no seu kernel.

Ao analisar a comunidade que congrega os desenvolvedores do Kernel do Linux, observa-se que existe somente uma página *web* (Home Page), de extrema simplicidade, sem nenhuma sofisticação gráfica ou de interatividade. É apenas uma página textual com orientações básicas e objetivas associadas com diversos *hiperlinks*, que direcionam o usuário para outras páginas de igual simplicidade. O formato da página pode ser visto no anexo A.

Existem (11) onze subtítulos dispostos na Home Page, que são os seguintes:

1. Os endereços URL dos arquivos de acesso público;
2. A discriminação das últimas versões do kernel;
3. Últimas notícias;
4. Informações sobre os *web sites* espelho (mirror);
5. Informações sobre o que é o Linux;
6. Informações sobre o que há de novo no Linux;
7. Endereço URL, para que os erros (bugs) sejam reportados;

8. Endereço das listas de discussões;
9. Árvores de desenvolvimento;
10. Informações sobre criptografia;
11. Comentários.

Basicamente, não existe informação sobre as características da comunidade, dados históricos da sua criação, quantidade de membros, etc. É verdadeiramente um ambiente simples em que somente consta o mínimo necessário. Pode-se afirmar que os membros da comunidade estão focados única e exclusivamente no desenvolvimento do kernel do Linux, não dispensando esforços extras, para qualquer sofisticação do *web site*, que abriga a comunidade.

Os arquivos das listas de discussão são acessados a partir da URL <http://marc.info/?l=linux-kernel>. Os arquivos das discussões estão disponibilizados em pastas ou diretórios, por assunto. A primeira pasta contém as discussões sobre o desenvolvimento do kernel do Linux, desde 01/03/1999. Foram trocadas, naquela data, 4921 mensagens. A média aproximada de mensagens nos últimos doze meses foi de onze mil mensagens. Isso implica um crescimento ao redor de quase duzentos por cento, de crescimento na quantidade de interações entre os membros desenvolvedores da comunidade, em relação à data de 15/03/2009, ou seja, dez dias depois a quantidade de mensagens existente na pasta é de 6.608.

Foi observado que os membros da comunidade são ativos, demonstrando um esforço de colaboração intenso, na busca de um objetivo bem definido.

A troca de informações por meio das interações das listas de discussão, em sua maioria, é de natureza técnica, na busca da solução de problemas relacionados ao desenvolvimento de programas de alta complexidade. Existem discussões com cerca de

aproximadamente 30 interações, implicando uma verdadeira discussão associada à troca de conhecimento, para a solução de problemas, de forma colaborativa.

3.3.2.2 Comunidade Apache

O desenvolvimento deste software começou em 1994 no NCSA⁵⁶. É um software de gerenciamento de servidores *Web*. Os projetos da Comunidade Apache sempre foram caracterizados pelo desenvolvimento baseados na colaboração, na meritocracia e no consenso, com o forte desejo de criar softwares de alta qualidade no seu segmento.

Em 1998, é criada a *The Apache Software Foundation - ASF*, cuja missão precípua é a de oferecer o suporte necessário às ações da Comunidade Apache. Esse suporte consiste no apoio organizacional, financeiro, legal e na manutenção de um ambiente estimulante e motivador para o desenvolvimento de software.

A *ASF* é uma organização sem fins lucrativos, composta por membros que, voluntariamente, se inscrevem para participar do projeto. Uma vez os membros eleitos, pelos membros do conselho da fundação, passam a fazer parte da comunidade de desenvolvedores. Para a escolha dos membros, não só se valoriza a questão da atitude voluntária do candidato, mas quais são as suas verdadeiras razões, para a efetiva participação da comunidade.

A *ASF*, periodicamente, elege os membros da diretoria da fundação, para a gestão das questões administrativas, financeiras e operacionais. Existe um regimento específico, que regula o processo, bem como as responsabilidades de cada integrante.

A *Web Site* da *ASF* é simples, sem sofisticação gráfica, mas possui as informações necessárias desde a sua fundação, organização, conselho, diretoria, membros, projetos, e

⁵⁶ NCSA – National Center for Supercomputer Applications. Disponível em: <http://www.ncsa.uiuc.edu/>. Acesso em: 20 jan. 2008.

equipes responsáveis, bem como a localização geográfica da residência dos membros da equipe.

É disponibilizada na primeira página do *web site*, a relação dos projetos em desenvolvimento, com informações detalhadas sobre cada projeto. Existe uma distinção entre os projetos Apache, ou seja, os projetos técnicos relacionados ao ambiente Apache e os projetos institucionais relacionados à Fundação.

Observa-se, em uma das opções do menu de informações, a opção “como funciona”, na qual é descrita como os projetos são desenvolvidos e a filosofia empregada. É evidente a preocupação da comunidade em expressar a sua filosofia de trabalho e de participação entre os membros.

Em uma das opções do menu, encontra-se acesso aos arquivos das discussões, baseado em troca de e-mails, por meio de listas de discussão. Os arquivos estão disponíveis desde o início de cada projeto. Efetuou-se uma busca aleatória em dezenas de diretórios e subdiretórios das listas de discussão, nos quais são encontrado arquivos de discussão, desde o ano de 2001.

Esses arquivos estão dispostos, por projeto e assunto, dentro de pastas nomeadas por ano e meses. Ao acessar os arquivos das listas de discussão, verifica-se que estão em formato de e-mail. Todas as interações podem ser identificadas pelas conexões agrupadas por discussão. A leitura desses documentos é difícil, fato este inerente às discussões baseadas em listas.

3.3.2.3 Comunidade MySQL

O Sistema Gerenciador de Banco de Dados Relacional – SGDB foi criado em 1994, por uma empresa sueca chamada MySQL AB. Ele é disponibilizado na modalidade GPL⁵⁷ e também sob o licenciamento proprietário, ou seja, há uma versão aberta e outra versão sob o licenciamento comercial tradicional. O *web site* é acessado por meio da URL <http://www.mysql.com>, sendo a parte comercial da MySQL.

A comunidade original pode ser acessada por meio de uma opção do menu denominada de *forge*⁵⁸, com a seguinte URL <http://forge.mysql.com/>.

O projeto começa com uma proposta sob a filosofia de código aberto e livre, criando depois um ramo comercial do produto final. A versão comercial é robusta, com funcionalidades que não fazem parte da versão aberta. A comunidade continua participando de forma voluntária no desenvolvimento da versão livre e aberta. Em nenhum da *web sites* é acessível informações sobre a história do projeto e as razões da decisão de se criar uma versão comercial do produto. Em janeiro de 2008, o produto é comprado pela *Sun Microsystems*⁵⁹ e após pela *Oracle Corporation*⁶⁰.

No *web site* da comunidade de desenvolvimento, verifica-se a importância dada aos membros desenvolvedores, que possuem projetos bem posicionados, demonstrados por estrelas, ou seja, os projetos mais bem avaliados, segundo critérios específicos, são ranqueados com cinco estrelas e dispostos na primeira página da comunidade. Os novos projetos também são listados.

O aspecto do *web site* é profissional, com uma apresentação gráfica sofisticada, oferecendo diversas opções, para que se tenha acesso às funcionalidades do produto, aos

⁵⁷ GPL – Gnu

⁵⁸ Forge – vocábulo do idioma inglês que significa, neste contexto, avançar. É um termo muito utilizado nas comunidades e incubadoras de software de código aberto e livre.

⁵⁹ *Sun Microsystems* pode ser acessada pela seguinte URL: <http://www.sun.com/>

⁶⁰ *Oracle Corporation* pode ser acessada pela seguinte URL: [HTTP://www.oracle.com/](http://www.oracle.com/)

parceiros comerciais, revendedores e tipo de serviços ofertados, como consultoria de instalação, treinamento e hospedagem⁶¹, efetuar cadastramento no *web site* para proposta de projetos. Uma vez o projeto aprovado, o proponente poderá iniciar o processo de desenvolvimento.

Há disponível também um fórum para discussão dos usuários do produto. O formato do fórum é de fácil compreensão e acesso, pois as discussões são apresentadas de forma endentada, permitindo assim uma fácil leitura das interações de cada discussão.

3.3.2.4 Comunidade PHP

PHP é uma linguagem de programação interpretada, projetada originalmente para criação de páginas dinâmicas da web. PHP é um acrônimo que significa *Hypertext Pre-processor*. Foi criada por Rasmus Lerdorf em 1994. A continuidade do desenvolvimento da linguagem é efetuada pela PHP Group, sendo distribuída como software livre.

O *web site* da comunidade PHP é simples, sem nenhuma sofisticação gráfica. Apresenta as informações básicas necessárias, para que a comunidade possa trabalhar, colaborar e compartilhar informações. É dada importância às novas versões e atualizações do software. Observa-se um destaque especial para os eventos que ocorrem ao redor do mundo voltados para os grupos de usuários.

No menu localizado no cabeçalho da Home Page é disponibilizado a opção de acesso à lista de discussão. Ao acessar essa opção, verifica-se que todas as discussões são devidamente categorizadas por assunto. A forma de discussão é por e-mail, ou seja, a troca de informações é efetuada por listas de discussão. Na apresentação das diversas listas de discussão

⁶¹ Utiliza-se o termo inglês, *hosting* para esta modalidade de serviço.

disponíveis é informado se a lista é moderada ou não, bem como o acesso ao grupo de notícias. Existem arquivos de discussão disponíveis desde o ano de 1999.

3.3.3 Síntese da análise etnográfica das comunidades

As ferramentas sociais que suportam as comunidades analisadas são de extrema simplicidade e de baixa complexidade. Surgem há mais de uma década e estão obsoletas. A forma de armazenagem é baseada em arquivos sequenciais organizados cronologicamente, não apresentando indexação dos assuntos discutidos nos fóruns, para acesso com maior rapidez, em eventuais pesquisas.

A CVAO MySQL, que criou a versão comercial, é a única que possui um fórum de discussão, com a apresentação de forma indentada, ou seja, permite ver a hierarquia das interações de forma visual e com maior facilidade. As demais CVAO estão utilizando listas de discussão, que nada mais são que uma simples troca de e-mails entre seus membros.

A CVAO do Kernel do Linux apresenta-se como comunidade mais ativa, paradoxalmente é a que possui as ferramentas sociais mais obsoletas de todas as CVAO analisadas.

No que tange aos membros participantes da CVAO, verifica-se que muitos deles não obedecem à caracterização dos tópicos de discussão, ou seja, fazem perguntas técnicas ou não em qualquer grupo. Eles parecem ser os que frequentam pouco a comunidade, possuindo apenas curiosidade pelos temas discutidos pelos membros do grupo, não tendo uma participação efetiva. Observa-se que a minoria de membros mantém a CVAO ativa, pelo fato de responderem muitas perguntas. Por outro lado, observa-se um forte relacionamento entre alguns de seus membros, criando até certa cumplicidade na condução das soluções dos problemas apresentados.

Mesmo que as comunidades apresentem centenas ou milhares de interações, durante anos, é visível identificar, em determinado período de tempo, os sujeitos ativos,

comprometidos e realmente engajados nos propósitos da comunidade. Fica evidente a percepção do nível de participação, contribuição e colaboração dos participantes.

Pode-se concluir, por meio da observação etnográfica, que desafios e dificuldades são enfrentados pelas CVAO, apesar dessas comunidades proporcionarem ambiente fértil para o aprendizado em larga escala para a solução de problemas. Essas dificuldades se relacionam a:

1. Imposição de padrões aos membros da comunidade;
2. Avaliação dos resultados da aprendizagem, por parte de seus membros, como indicador do nível de qualidade das respostas apresentadas para as questões postadas;
3. A solicitação imediata de *feedbacks* pode não ser atendida, provocando certa insatisfação ou frustração;
4. A construção da identidade e de um relacionamento confiável entre os membros da comunidade pode demandar muito tempo para ser estabelecida.

3.3.4 Análise da ação mediada

A apresentação gráfica, por meio da figura 12, demonstra o modelo proposto por Wertsch (1991) e a respectiva adaptação esquemática para a análise da ação mediada dos fóruns de discussão. Como exemplo, apresenta-se uma discussão extraída da CVAO MySQL descrita no anexo H, sob o código 1.4.SQL.

O problema postado na citada comunidade diz respeito à dificuldade de um de seus membros, chamado Allison Loeb, com a instalação do banco de dados MySQL. Allison Loeb postou o problema no dia 24/02/2008, às 05h41min da manhã. No mesmo dia, às 08h18min da manhã, o colega Jay Alverson solicita ao Allison um complemento de informação. No dia 25/02/2009, às 02h56min da manhã, Allison atende à solicitação de Jay fazendo o complemento

de informações relativo ao problema postado. Logo, às 08h05min da manhã, Jay posta a resposta. No dia 01/03/2009, às 07h00min da manhã, Allison informou a Jay que o problema já fora solucionado, mas complementa dizendo que estava cometendo outro equívoco.

A ação da aprendizagem ocorre desde a segunda interação, ou seja, no momento em que Jay solicita mais detalhes sobre o problema, Allison percebe que a descrição do problema não está completa, para que um colega possa apresentar uma solução. No momento em que Allison complementa a pergunta principal, imediatamente Jay posta uma possível solução respondendo: “Try loggin in as ´root´”. Finalmente, Allison, de posse da orientação, executa o procedimento e obtém êxito na instalação do bando de dados.

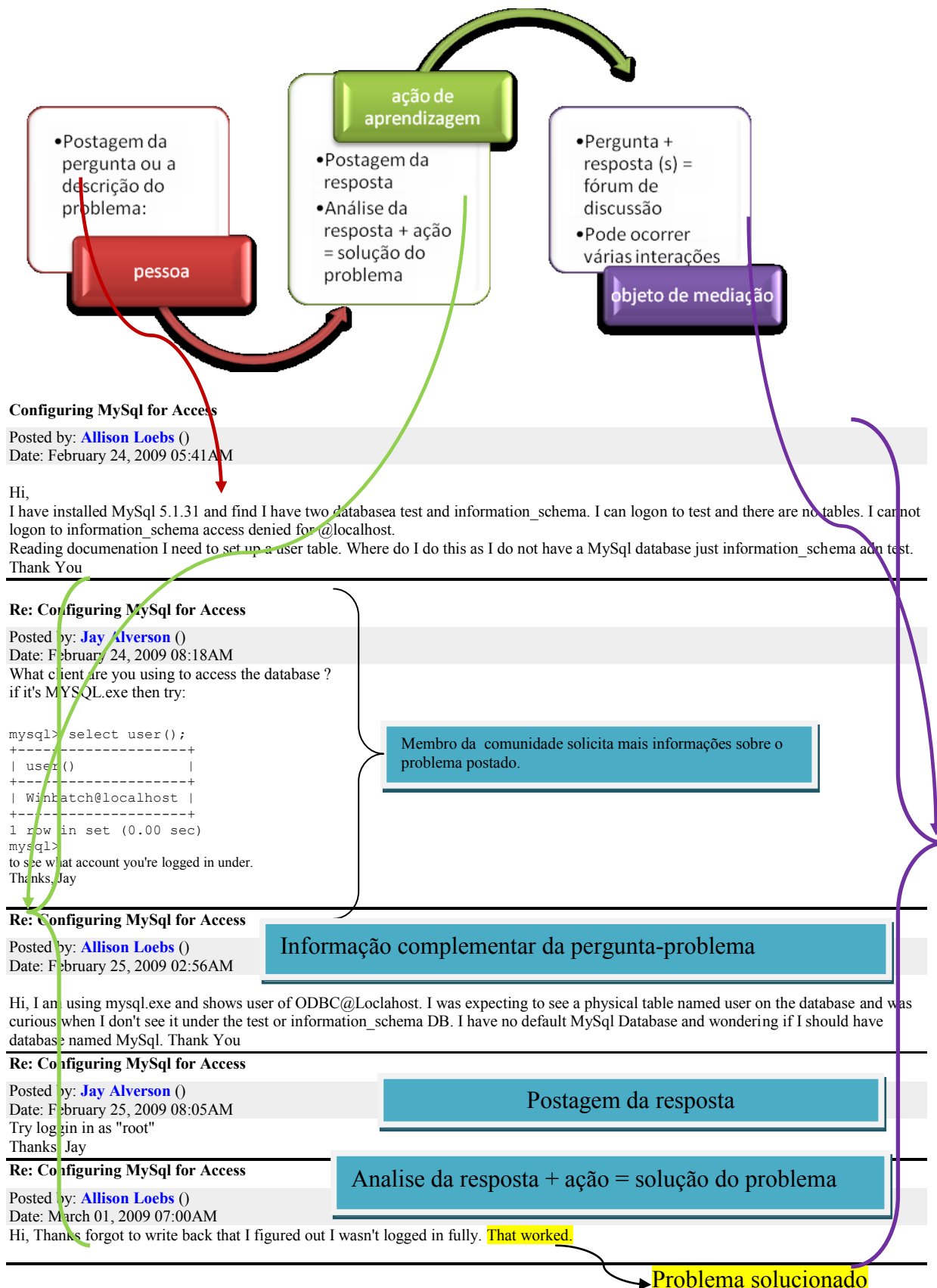


Figura 13 - Demonstração gráfica – a relação da ação mediada com o fórum de discussão

O conjunto de interações (quatro) desde a pergunta até o *feedback* com a afirmativa do sucesso da solução do problema, postada por Allison, pode ser chamado de objeto de mediação ou, para efeito desta pesquisa, de objeto de aprendizagem. Esse conjunto de interações, se fosse armazenado e indexado, poderia em outras ocasiões ser reutilizado, por meio da busca por palavras-chave, mas, como os recursos utilizados pelas CVAO não oferecem condições para tal, isso não tem sido possível.

Para melhor exemplificar a complexidade da situação, pode-se citar a discussão da CVAO Linux apresentada, no anexo F, sob o código 2.30.LNX. Essa discussão é composta por trinta (30) interações. A primeira postagem do problema ocorre no dia 31/01/2008, às 23h41min, e a última postagem, com a realimentação da solução, ocorre no dia 06/02/2008, às 00h04min. No exemplo acontece uma forte interação entre diversos participantes discutindo vários assuntos relacionados ao problema. Na discussão sob o código 1.8. LNX, verifica-se a participação ativa do próprio Linus Torvalds, criador do Sistema Operacional Linux. (Ver anexo A)

Levando em consideração outras pesquisas realizadas nesse campo, Björck (2001), ao apresentar os resultados da sua pesquisa feita com os estudantes do curso denominado CLEA (termo sem significado específico em sueco e em inglês), na modalidade a distância, que tinha por objetivo estimular o desenvolvimento da economia social no âmbito da União Européia – cujos dados foram coletados no período de outubro de 1996 a maio de 1998 –, efetua algumas considerações, após analisar o conteúdo dos fóruns de discussão (neste contexto chamado de objetos de aprendizagem) para a solução de problemas. O autor afirma que:

Quando os estudantes tornam-se engajados no processo de solução de problemas, eles se tornam parte ativa da responsabilidade pelo seu aprendizado, por meio da criação necessária de artefatos usados no método. Ao apropriar formas de levar adiante a busca da solução, os estudantes gradualmente estabelecem um acordo mútuo entre o grupo usando uma abordagem altamente conversacional. A apropriação do método não é limitada aos estudantes – estabelecer o nível do grupo é fundamental para o domínio do método. Quando os estudantes comentam aos outros estudantes

sobre o seu trabalho, estes se engajam muito mais na pesquisa dialógica. Este processo inclui escrever os comentários bem como a troca de opinião. Comentar cada outro comentário constrói a comunidade de forma a promover a reciprocidade e um maior engajamento (BJÖRCK, 2001, p.11).

As características identificadas nos estudantes, na citação acima, corroboram com as características identificadas nos participantes das CVAO, analisadas nessa pesquisa. A diferença é que os participantes do curso fazem as atividades de forma orientada, ao passo que os participantes das CVAO as fazem de forma voluntária e independente.

Cameron (1999) cita em seus estudos que a ocorrência de mudanças de papel do facilitador do fórum e troca de opiniões entusiásticas entre os participantes – assim como nas citações de Wertsch (2002) sobre a reciprocidade entre professor e estudante na troca de ideias –, influencia positivamente o comportamento dos membros da comunidade e, por conseguinte, os resultados do fórum.

Em ambos os estudos é verificada a necessidade da ocorrência de várias sessões para que os estudantes comecem a participar de forma envolvente e confiável. A partir da perspectiva da ação mediada, isso significa que em ambos os casos, eventualmente, os alunos passaram a utilizar um conjunto de ferramentas culturais sem a necessidade de continuar com o apoio externo.

Estes resultados sugerem que os grupos parecem se organizar em torno de tarefas e processos (...). Na continuidade das atividades, estes níveis sofisticados de interação podem ser referidos como a construção da comunidade nos termos de Rheingold (1992). Se a construção da comunidade é estabelecida, como no C-grupo, atividades bem sucedidas podem continuar mesmo após o término formal do curso (Björck, 2001, p. 12).

Wertsch (2002) comenta que na década de 1930, Vigotsky não podia, naturalmente, ter considerado o hardware e o software como recursos para o processo de comunicação, mas nos dias atuais é amplamente reconhecido que suas abordagens se aplicam às formas de comunicação, por meio desses recursos. Daí a necessidade de entender como novas formas

de comunicação a distância por computador e de aprendizagem mediada podem "alterar todo o fluxo e a estrutura das funções mentais." E dado que Vigotsky estava preocupado com a forma como essas funções mentais ocorrem nessa prática, isso significa que se deve preocupar com a forma, com todo o fluxo e estrutura de processos de comunicação e colaboração, bem como com os processos mentais individuais, que podem passar por essa transformação.

De uma forma ou de outra, diversos autores reconhecem que uma nova maneira de mediação está envolvida nos fenômenos associados às CVAO, principalmente no que tange aos fóruns de discussão. Björck (2001); McConnell (2004); Orril (2001) e Ronteltape e Eurelings (2002) relatam os resultados de seus experimentos sobre o uso das ferramentas assíncronas como suporte aos fóruns de discussão, para a solução de problemas. Os termos sistema de conferência assíncrona, ferramentas de discussão assíncrona, fórum assíncrono e ferramentas de assíncrona são termos usado em seus relatos. Isso já demonstra certa uniformização desses termos. Tal fato sugere que tais autores estão conscientes de que os processos colaborativos que estão sendo alvo de suas investigações, de algum modo diferem do processo de comunicação presencial, ou seja, face-a-face. Por outro lado, nota-se alguma ambivalência não intencional a essa questão. Os autores frequentemente fazem comentários sugerindo que a mediação do computador é simplesmente um detalhe, ou algo que proporciona um incremento quantitativo de eficácia, mas nenhuma mudança fundamental para os processos envolvidos.

Diante desse contexto, vale considerar a possibilidade de que seja necessário um realinhamento mais radical da abordagem analítica utilizada pelos autores. Sendo assim, ao invés de ver o intróito de uma nova ferramenta cultural, tal como fazer um formulário de registro de ação mais fácil ou mais eficiente, pode ser importante considerar como ela introduz uma mudança fundamental – às vezes a tal ponto que se pode questionar se a mesma forma de ação é envolvida em tudo. De qualquer forma, mesmo que se façam inovações nas

ferramentas culturais, Wertsch (2002) opina de que a maneira como acontecem as ações e os processos mentais é frequentemente imprevista.

Em muitos casos, essa nova ferramenta é desenvolvida com a intenção de influenciar a mudança nos processos sociais ou psicológicos, contudo é precisamente em tais áreas que se pode encontrar o seu legado mais poderoso e duradouro, ou seja, a resistência do sujeito à mudança. Relativamente a esse ponto, pode-se observar que as CVAO continuam a utilizar as mesmas ferramentas há mais de uma década, isso demonstra certa contradição, visto que utilizam o ambiente para a solução de problemas de natureza simples ou complexa, para a construção sofisticada de softwares, todavia não demonstram interesse e esforço para inovar as ferramentas de suporte ao processo, ou seja, continuam com o mesmo comportamento de uma década passada.

No entanto, pode ser importante levar em consideração a possibilidade de que não se pode simplesmente adicionar as ferramentas de comunicação assíncrona em um mix existente de processos sociais e psicológicos sem alterá-los de maneira fundamental (não intencional) e que isso pode ser um dos aspectos mais interessantes a se considerar no processo de solução de problemas mediante fóruns de discussão.

Alguns autores têm já avançado um pouco nessa direção, como demonstram citações em seus trabalhos científicos, “os participantes em nossos experimentos tem vivenciado um novo método de comunicação que afetou o seu nível de atividade” (RONTELTAP; EURELINGS, 2002, p.10). Björck (2001) escreve sobre a existência do chamado “meta-comentário” que parece emergir de forma única no contexto dos fóruns. Ao mesmo tempo, o autor argumenta que “os alunos estão dominando o processo de solução de problemas usando abordagem altamente conversacional”. Comenta que “iniciar uma discussão on-line em um formato mais rico” demonstra indícios de que a mediação por meio do computador é simples extensão da interação tradicional presencial.

Levando em consideração as afirmações dos autores citados, sobre como essa nova forma de mediação pode levar à transformação qualitativa do fluxo e da estrutura dos processos sociais e psicológicos, Orril (2001) escreve sobre as limitações críticas das discussões encadeadas (fóruns). De acordo com ele, um dos problemas é quando não existe uma clara realimentação permitindo ao sujeito emissor de uma opinião discernir se os pontos referidos estão realmente claros ou não. Steinkuehler et al. (2002) argumentam ser difícil a orquestração para que todas as “vozes” das discussões assíncronas se façam ouvir.

Um ponto que merece destaque é refletido por algumas transcrições de Björck (2001), quando aborda o envolvimento na pesquisa dialógica. O autor cita o caso de uma participante de um fórum de discussão que faz um extenso comentário sobre vários outros levantados por seus interlocutores. O autor ressalta que o caso pode ser considerado um exemplo de "meta-comentário", uma prática que parece acrescentar uma dimensão de processamento cognitivo reflexivo. É concebível que a citada participante fez, nesse caso, com que também ocorresse algum tipo de interação presencial, mas a menos que o “gênero do discurso” (BAKHTIN, 1992; WERTSCH, 1991) fosse altamente restrito, assim como um extenso comentário feito por um único participante seria muito incomum.

Tudo indica que pelo fato de o fórum ser mediado por computador possibilita ao participante fazer um comentário longo e reflexivo sobre os comentários de outros participantes. São oportunidades que surgem e em grande parte devidas ao fator de que o meio envolvido permite a um único participante do fórum explicitar a sua opinião calmamente, uma vez que não se sente pressionado pelas limitações de tempo da interação verbal presencial. Em outras palavras, algumas cruciais novas características de comunicação social – e, presumivelmente, também os processos mentais – surgiram com o uso dessa nova forma de mediação.

Basicamente, o que foi relatado não quer dizer que tanto as ferramentas culturais como as suas consequências são inerentemente positivas ou negativas, diante da prática presencial. Em suma, os autores citados tendem a vê-las como uma oferta benéfica de novas oportunidades para o desenvolvimento de comunidades de aprendizagem e crescimento cognitivo, e há boas razões para acreditar que isso seja verdade. Como acontece com qualquer ferramenta cultural, no entanto, as novas formas de mediação por computador têm restrições, bem como também promessas a seu respeito (WERTSCH, 2002). O autor ainda afirma que é muito cedo para ter certeza sobre as promessas, para não falar das limitações associadas a essa nova forma de mediação. Tal consideração é feita com base nas ideias emanadas de Bakhtin (1992, 1995) sobre a natureza do diálogo humano. Ambos os autores viram a dialogicidade como propriedade essencial do ser humano, que o motiva a levantar questões de como o diálogo difere do monólogo e como várias formas de diálogo podem ser distinguidas.

Uma das dimensões básicas, ao longo do qual eles queriam diferenciar as formas de diálogo, tem a ver com a intensidade do dar e receber envolvidos quando as “vozes” entram em contato. Semelhante processo pode acontecer por meio de troca rápida de enunciados abreviados entre os interlocutores que já se conhecem entre si ou por declarações monológicas entre estranhos, que podem durar alguns minutos ou horas ou até mesmo deixar de produzir. Bakhtin (1992) considera que a fala humana e a consciência ainda são inerentemente dialógicas, mas a dinâmica é fundamentalmente diferente.

Independentemente dessas abordagens, é claramente possível reconhecer a distinção entre o dar e receber em um jogo rápido de troca de enunciados em diálogo presencial, em que as expressões são muitas vezes fragmentadas e interrompidas; por outro lado, quando o processo se torna contínuo, o sentido pode ficar claro desde que, obviamente, quando o processo é mediado por ferramentas culturais, de forma assíncrona, para a solução de problemas colaborativamente. Em suma, o uso de novos sistemas de comunicação assíncrona

pode ser a introdução de um elemento de mudança na solução de problemas, em ambiente colaborativo, baseado nos fóruns de discussão e suas consequências para a cognição, identidade e colaboração. Por certo, as diversas abordagens citadas e, de diferentes autores, demonstram um avanço importante nos resultados obtidos na solução de problemas em fóruns de discussão mediados por computador; de qualquer forma o tema merece uma atenção relevante à medida que as ferramentas de mediação seguem o caminho da evolução.

3.4 *Análise da integração entre objetos de aprendizagem (OA) e as comunidades virtuais de aprendizagem auto-organizadas (CVAO)*

Pode-se ressaltar, diante dos dados analisados sob o ponto de vista qualitativo e quantitativo, que todas as CVAO analisadas construíram, ao longo do tempo de existência, a declaração de sua missão, ou seja, qual a razão da existência da comunidade, quais são efetivamente seus propósitos, sua finalidade específica e seus valores, ficando evidente e explícita, a personalidade e identidade de cada uma das comunidades.

As CVAO apresentam claramente os recursos e ferramentas disponíveis de comunicação e interação, bem como as suas características, para que não se tenha nenhuma dúvida no acesso aos seus recursos. Em algumas comunidades, pela simplicidade do ambiente, essa realidade é difusa, tendo em vista que existem ambiguidades de informações e dificuldades de navegação nas opções disponíveis, ficando difícil a localização de opções e recursos disponíveis. A navegabilidade na CVAO Linux apresenta-se como a mais crítica nesse quesito. O membro dessa comunidade precisa se familiarizar, previamente, com o ambiente, para melhor se localizar quanto aos recursos e ferramentas.

A disponibilização do perfil dos membros não é um recurso usual nas CVAO analisadas. Aparentemente, a preocupação em não se apresentar de forma explícita o perfil dos membros não é pela questão de manutenção da privacidade, mas por não ser uma

prioridade na divulgação deles mesmos. Fica evidente que os membros atuam de forma voluntária e altruísta, não se preocupando, em nenhum momento, com a autopromoção, mas sim em apresentar trabalho que contribua para o desenvolvimento dos produtos relacionados à CVAO. É importante salientar que muitos membros utilizam apelidos (nicknames) ininteligíveis, ou seja, são praticamente anônimos.

A CVAO Apache define a atuação de seus membros, mesmo porque é mantida por uma Fundação organizada, com a definição de condutas, responsabilidades, e atribuições de sua Diretoria e Conselheiros. São explicitados, de forma clara e formal, os critérios dos diversos papéis desempenhados pelos seus membros, bem como as estratégias de estímulo e motivação.

Todas as CVAO, sem exceção, de forma mais evidente ou não, apresentam suas regras de conduta, sempre associadas aos seus valores, o que assinala a personalidade da CVAO. Há uma prioridade em estabelecer regras de conduta e atitudes de seus membros. Por outro lado, não se observa muita relevância à punição de membros que desrespeitam regras, ou apresentação de listas de membros suspensos, ou algo semelhante.

Todos estão focados em estabelecer as regras atitudinais e de conduta, mas com a pressuposição de que irão segui-las e não violá-las. Em nenhum momento observou-se qualquer iniciativa ou manifestação de membros em violar quaisquer das regras estabelecidas.

As CVAO analisadas demonstram interesse em criar espaços para divulgação dos eventos envolvendo os seus produtos ou mesmo os temas afins. A CVAO PHP e MySQL possuem um espaço muito bem definido e institucionalizado para a divulgação de seus eventos.

Por se tratar de comunidades virtuais evidencia-se a correlação com casos de sucessos, notícias e discussões voltadas à criação de um vínculo claro e concreto com o mundo real, o que não fica evidente na CVAO Linux.

Diante da preocupação em simplificar a complexidade que a CVAO vai adquirindo ao longo do seu tempo de existência, verifica-se a criação de subgrupos de estudos e discussão, na busca de especialização e foco na solução de problemas, para o desenvolvimento sistemático e sustentável da CVAO.

No que tange especificamente aos objetos de aprendizagem, constata-se que estes não são empregados na forma tradicional, em que são utilizados de maneira sequenciada com vistas a propiciar a consecução de um objetivo de aprendizagem específico, seguindo métodos instrucionais bem definidos. Nesses casos, os objetos são colocados de forma livre e não de forma artificial como nos cursos online devidamente planejados.

Diante de um questionamento de um membro da comunidade, outro membro efetua a postagem de um objeto direcionado a atender àquela demanda. O objeto é colocado de acordo com o contexto do ambiente. Essa modalidade de utilização de OA sugere definição, já citada anteriormente: *“qualquer recurso digital que pode ser reutilizado para facilitar o aprendizado através de dispositivos digitais usado para a mediação do aprendizado”* (WILEY, 2000, p.7). O foco do processo passa a ser na mediação (WERTSCH, 1991) e na distinção, com base na facilitação, para ser significativo (COLE, 1997).

Em razão de o OA mediar as atividades dos indivíduos dentro da comunidade, em algumas situações pode ser suscetível às mesmas fraquezas presentes nos métodos tradicionais de uso de objetos de aprendizagem. Entretanto, as comunidades são ricas em interação humana, as quais podem utilizar recursos de forma arbitrária e de forma eficiente, tornando, num ambiente colaborativo, de alta qualidade o processo de significação da mediação (BAKHTIN, 1995; COLE, 1997).

O mais importante enfoque de uso dos OA nessas comunidades quanto ao uso tradicional é o pressuposto de contar com pessoas para localizar, montar e contextualizar os recursos.

Semelhante questão pode ser demonstrada pela interação identificada na CVAO *SourceForge*, conforme figura a seguir:

Posted 1 year ago by [rameshch45](#) [1]

I have a few questions about using different licenses of open source software and other related topics.

What are some of the famous online discussion forum websites where people join in and discuss about open source?

Membro da comunidade posta pergunta no fórum.

Posted 1 year ago by [silverfang](#) [2]

Here (of course). You can ask me the question and I'll do my best to answer it.

Um colega, cordialmente, se propõe a responder.

Posted 1 year ago by [rameshch45](#) [3]

Sure. I wondered and asked that because I did not find many posts in the forums that I can search/read, essentially learning from earlier users... So, I thought maybe this is a site specific placeholder and there could be some other extensively exclusive forum to discuss open-source.

Thank you.

Membro da comunidade justifica-se, pois tentou procurar a resposta nos arquivos do fórum, mas não conseguiu encontrar. **Esta é a principal fraqueza dos fóruns disponibilizados**, não possui forma eficiente de busca indexada. A única forma é postar uma pergunta no fórum, mesmo que esta pergunta já tenha sido feita no passado.

Posted 1 year ago by

http://developer.kde.org/documentation/licensing/licenses_summary.html

<http://www.fsf.org>
<http://www.opensource.org>
<http://www.freebsd.org>

Apesar do membro da comunidade sob o “nickname” silverfang se colocar a disposição para responder a pergunta, um outro colega, logo, posta a resposta para o colega rameshh45.

Figura 14 - Discussão sob o código 1.4.SFG

Entretanto, a tragédia dos comuns (HARDIN, 1968) sugere que a colaboração voluntária não é sustentável ao longo do tempo à emergência da Internet e, mais especificamente, o movimento do software livre e de código aberto tem demonstrado que a habilidade do uso da tecnologia da informação e comunicação tem colocado os membros da

comunidade em um processo simbiótico de relacionamento da forma como segue: “*you respond to my questions that I will respond to yours*” (HIMANEN, 2001; RAYMOND, 1999).

Quando os OA são considerados como fator de mediação, significa que os membros da comunidade o empregam para a solução de um problema ou outro tipo de atividade para responder às questões postadas. Aparentemente partes heterogêneas de conteúdos digitais, avaliações, simulações e aplicações giram para dentro de um processo de mediação, proporcionando um modelo conceitual para um novo método de indexação e descoberta, combinação, uso e avaliação dos recursos educacionais em formato digital, conforme segue:

1. Indexação e descoberta:

Os OA não são catalogados com os atributos de metadata ao serem inseridos no repositório. Os membros da comunidade sabem da existência dos objetos e sua localização. Na maioria das vezes, tais objetos estão armazenados em seus próprios computadores;

2. Combinação:

Os OA não são automaticamente armazenados ou demonstrados em formulários ou formatos padronizados. Estes são associados a textos que orientam sua localização ou simplesmente os descrevem em formato escrito e textual;

3. Uso:

Os OA são empregados em razão de uma demanda, por meio de uma questão que está orientada à solução de um problema específico.

4. Avaliação:

Os objetos de aprendizagem não são avaliados de forma isolada ou fora do contexto o qual está sendo utilizado. O membro da comunidade apenas o

avalia dentro do contexto da pergunta e da resposta associada ao auxílio parcial ou total para a solução do problema. O OA está associado ao contexto em que está sendo inserido e dele é totalmente dependente.

Infere-se que futuramente as CVAO oferecerão muitos subsídios para o processo de ensino e aprendizagem formal, principalmente porque essas comunidades podem ser formadas por centenas de milhares de membros cuja escalabilidade passa a ser uma qualidade e não uma restrição.

Os membros dessas comunidades são providos por um significativo conjunto de recursos que promovem o seu aprendizado sem as restrições de tempo e espaço e com um riquíssimo processo de interação humana. Os OA são encapsulados em um significativo contexto de aprendizagem, mas o acesso e a contextualização deles são feitos naturalmente pelos próprios usuários, de acordo com as suas reais necessidades.

Quanto aos critérios de categorização dos problemas da produção de SLCA, não se observou nenhum tipo de categorização dos problemas discutidos na CVAO. O único recurso que é disponibilizado é a *FAQ*⁶², forma bastante limitada de acesso à informação, diante do volume de conhecimento gerado pelos fóruns. A CVAO Linux enfoca a busca de informações na seção *FAQ*, antes de o usuário fazer qualquer pergunta, por meio do fórum. Existe também o campo de busca (search), mas como não há indexação dos fóruns, no formato de objetos de aprendizagem, a busca também se torna limitada, na maioria das vezes muito frustrante para o usuário, como pode ser constatado na figura 13.

No que concerne a esse ponto, vale ressaltar que pelo fato de as CVAO possuírem um grupo de membros muito ativos, estes já possuem um conhecimento tácito sobre o funcionamento da comunidade e o local onde está o conhecimento. À medida que as

⁶² FAQ – frequent answers and questions – perguntas e respostas mais frequentes.

discussões são armazenadas, o conhecimento tácito torna-se conhecimento explícito, todavia o problema passa a ser a indexação e organização desse saber. Diante de tal situação, o que realmente acaba importando em tais comunidades é quem possui o conhecimento tácito.

Os membros iniciantes ou usuários que pretendem obter algum tipo de informação, como um usuário qualquer, ou seja, não participa efetivamente como um desenvolvedor, encontra dificuldades de localizar a informação necessária para solução do seu problema. A única forma é acessar o fórum e postar a pergunta sobre o problema existente. Observou-se que até o usuário chegar a esse ponto encontra muitas barreiras, no entanto como esse usuário não é um membro ativo da comunidade e sabe das limitações do recurso, não posta nenhum retorno com as críticas pertinentes, talvez por não se sentir com poder ou interesse de criticar ou fazer sugestões no modelo de funcionamento da CVAO. Já o membro que participa ativamente demonstra certo nível de acomodação, pois já sabe os caminhos e os meandros, além de fazer parte de uma equipe fiel e comprometida uns com outros, as limitações não são percebidas.

No que diz respeito à existência de padrões replicáveis de OA adotados por parte dos membros da CVAO, constata-se que não existe consciência evidente sobre o uso desses objetos. Um conjunto de interações, desde a pergunta inicial até a resposta final dando por solucionado o problema, forma os objetos de aprendizagem, os quais ficam armazenados em pastas, em ordem cronológica e não são indexados sob nenhum critério.

Para os conteúdos desses diretórios, em todas as CVAO, não existe recurso ou mecanismo de busca específica. Os arquivos formados pelas listas de discussão efetivamente são um conjunto de e-mails arquivados inclusive em ordem cronológica. A consulta é extremamente complicada e trabalhosa, uma vez que o mecanismo de busca é o mesmo que encontrar fisicamente um documento dentro de uma pasta que está dentro de determinada gaveta em determinado arquivo, dentre diversos outros.

Por outro lado, constata-se em algumas situações, a replicação de objetos tais como: biblioteca, listas de comandos, conjunto de procedimentos, funções, etc. que os membros possuem arquivados em seus próprios computadores; usualmente eles “copiam e colam” no corpo do e-mail associado com as explicações direcionadas à solução do problema em pauta. Todavia, houve uma única evidência de replicação observada. Na figura 14, apresenta-se exemplo⁶³ de listas de comandos que, por certo, foi replicado de outras discussões, ou extraído de algum manual técnico de programação:

```
List: linux-kernel
Subject: Re: Pull request: TASK KILLABLE
From: Matthew Wilcox <matthew \(\) wil ! cx>
Date: 2008-01-31 23:45:05
Message-ID: 20080131234504.GB25967 \(\) parisc-linux ! org
[Download message RAW]
```

```
On Tue, Jan 29, 2008 at 10:41:24PM +1100, Linus Torvalds wrote:
> On Mon, 28 Jan 2008, Matthew Wilcox wrote:
> > > I'd like you to pull the task_killable branch of
> > git://git.kernel.org/pub/scm/linux/kernel/git/willy/misc.git
> > Well, Andrew already pointed out some of this, but I do want more
> information. There's a "git request-pull" script that even comes with git
> and that generates a shortlog, a proper url+branch description, and a
> diffstat of the changes.
> > So that is already much more descriptive, but in addition to that, if this
> isn't one of the "regular" pulls (which it obviously isn't), I really want
> a human description too so that I know what I'm pulling and what it's all
> about.
> > I do *not* want to be in the position of having to fetch first, just to be
> able to look at what I'm fetching. By the time I pull a branch, I want to
> basically know (a) what I'm pulling and (b) why I _want_ to pull it.
OK. Hopefully this satisfies you:
Introduce TASK_KILLABLE.
The basic idea was Linus', back in 2002. We can't normally return a
short read because it breaks applications. But if a task has received
a fatal signal, it doesn't have a chance to notice that it didn't get
all the data it asked for.
To allow tasks to be interrupted by fatal signals, we introduce a new
TASK_* bit; TASK_WAKEKILL. We also add a predicate fatal_signal_pending;
the counterpart of signal_pending(). Then we add killable versions
of lock_page(), mutex_lock(), schedule_timeout(), wait_event(), and
wait_for_completion(). Finally, we can make the NFS 'intr' mount option
a no-op.
At the end of this patch set, you can now kill a task in readdir(), read()
and write(), at least most of the time. There is still much more work
that can (and should) be done, such as implementing down_killable(),
lock_kernel_killable() and even spin_lock_killable().
While my main motivation has been to fix problems I have with flaky
wireless networks and/or NFS servers, people like Ingo and Nick Piggin
want to see this infrastructure for helping them improve the OOM killer.
The following changes since commit 8af03e782cae1e0a0f530ddd22301cdd12cf9dc0:
  Linus Torvalds (1):
    Merge branch 'for-2.6.25' of git://git.kernel.org/.../paulus/powerpc
are available in the git repository at:
```

⁶³ O exemplo é parte da interação código 1.8.LNX.

Exemplo de uma possível replicação ou reutilização de um objeto

```
git://git.kernel.org/pub/scm/linux/kernel/git/willy/misc.git task_killable
```

```
Liam R. Howlett (2):
```

```
  Add mutex_lock_killable
  Use mutex_lock_killable in vfs_readdir
```

```
Matthew Wilcox (20):
```

```
  Use wake_up_locked() in eventpoll
  Add macros to replace direct uses of TASK_flags
  perfmon: Use task_is_*
  proc/array.c: Use TASK_REPORT
  proc/base.c: Use task_is_*
  wait: Use TASK_NORMAL
  power: Use task_is_*
  ptrace: Use task_is_*
  sched: Use task_contributes_to_load, TASK_ALL and TASK_NORMAL
  signal: Use task_is_*
  exit: Use task_is_*
  Add TASK_WAKEKILL
  Add fatal_signal_pending
  Add lock_page_killable
  Use lock_page_killable
  Add schedule_timeout_killable
  Add wait_event_killable
  Add wait_for_completion_killable
  NFS: Switch from intr mount option to TASK_KILLABLE
  Remove commented-out code copied from NFS
```

```
arch/ia64/kernel/perfmon.c | 4 +-
fs/eventpoll.c | 11 +----
fs/nfs/client.c | 6 +--
fs/nfs/direct.c | 10 +----
fs/nfs/inode.c | 6 +--
fs/nfs/mount_clnt.c | 2 +-
fs/nfs/nfs3proc.c | 7 +--
fs/nfs/nfs4proc.c | 27 +-----
fs/nfs/nfsroot.c | 3 -
fs/nfs/pagelist.c | 18 +-----
fs/nfs/read.c | 5 --
fs/nfs/super.c | 4 --
fs/nfs/write.c | 7 +---
fs/proc/array.c | 7 +---
fs/proc/base.c | 2 +-
fs/readdir.c | 5 +-
fs/smbfs/request.c | 2 +-
include/linux/completion.h | 1 +
include/linux/mutex.h | 5 ++
include/linux/nfs_fs.h | 9 +----
include/linux/nfs_mount.h | 2 +-
include/linux/pagemap.h | 14 ++++++
include/linux/sched.h | 36 ++++++++
include/linux/sunrpc/clnt.h | 2 -
include/linux/sunrpc/sched.h | 2 -
include/linux/wait.h | 52 ++++++++
kernel/exit.c | 88 ++++++++
kernel/mutex.c | 36 ++++++++
kernel/power/process.c | 6 +-
kernel/ptrace.c | 8 +---
kernel/sched.c | 28 ++++++
kernel/signal.c | 19 ++++++
kernel/timer.c | 7 +++
kernel/wait.c | 2 +-
mm/filemap.c | 25 ++++++++
net/sunrpc/clnt.c | 47 +-----
net/sunrpc/rpcb_clnt.c | 3 +-
net/sunrpc/sched.c | 15 +-----
net/sunrpc/sunrpc_syms.c | 2 -
39 files changed, 281 insertions(+), 254 deletions(-)
```

```
-- Intel are signing my paycheques ... these opinions are still mine
"Bill, look, we understand that you're interested in selling us this
operating system, but compare it to ours. We can't possibly take such
a retrograde step."
```

Figura 15 - Interação 1.8.LNX

4 CONSIDERAÇÕES FINAIS

Considerando-se os resultados da análise dos dados sob a metodologia utilizada nesta pesquisa, constata-se que o processo de solução de problemas, a comunicação entre os participantes e notadamente a forma e funcionalidade das CVAO são realmente eficazes, ou seja, 100% dos problemas da amostra da pesquisa, postados nos fóruns de discussão das CVAO da tecnologia LAMP, foram resolvidos. Por outro lado, o que era indagação, agora se torna constatação, tendo em vista que os ambientes virtuais, recursos e ferramentas utilizados pelas CVAO não permitem a classificação e indexação das discussões resultantes das interações nos fóruns. E, portanto, não oferecem a possibilidade de busca automática para a solução de problemas, já discutidos e solucionados em interações anteriores.

A prática comum constatada foi a de que os novos membros da CVAO encontram mais facilidade de obter resposta para a solução de um problema postando perguntas nos fóruns. Diante da evidência do não questionamento dos membros ativos das CVAO sobre esse assunto, não existe a iniciativa de criar inovadoras ferramentas de automação, armazenagem, indexação e busca dos conteúdos dos fóruns. Infere-se ainda que devido à inexistência de um padrão de conformidade para a postagem das perguntas e, por pressuposto, postagem das respostas, o comportamento continua o mesmo, ou seja, não há mudanças, pois o formato de funcionamento e apresentação dos fóruns e/ou listas de discussão continua praticamente intacto desde a sua criação.

Em face disso, pode-se argumentar que a sustentabilidade das CVAO pesquisadas, que já ultrapassam mais de uma década desde a sua criação, decorre das seguintes razões:

1. Extraordinário equilíbrio entre o conhecimento tácito e explícito no âmbito da CVAO;

2. Elevado nível de comportamento colaborativo, cooperativo e iniciativa voluntária, por parte dos membros ativos;
3. Elevado nível de desprendimento, obstinação e altruísmo em prol dos resultados almejados pela CVAO;
4. Elevada capacidade de autoaprendizagem, independência e autonomia, na condução da solução dos problemas encontrados;
5. Elevado nível de disciplina, autoexigência, responsabilidade e comprometimento com prazos, qualidade e outras exigências estabelecidas pela comunidade;
6. Poder hegemônico dos membros com perfil similar;
7. Exclusão de membros sem o perfil predominante.

O resultado desta pesquisa pretende colaborar no preenchimento de um hiato ainda existente, ou seja, melhorar e facilitar as condições de aprendizagem, por meio de aprimoramento dos objetos de aprendizagem dispostos nos fóruns de discussão, ainda pouco elaborados, de difícil recuperação, indexação, e também de qualidade variada.

Pressupõe-se que uma vez agregados novos recursos tecnológicos e admitido um padrão de conformidade, os membros das CVAO, ainda que não necessariamente, precisarão ter inúmeras habilidades e um comportamento de cooperação e colaboração destacado e singular, pois espera-se que o uso de tais recursos fará o universo de participantes ainda maior e com grupos cujo comportamento e habilidades serão mais aperfeiçoados e diversos dos que hoje imperam neste tipo de comunidade.

Outras constatações de alta relevância também foram identificadas, que são arroladas a seguir:

1. Cobertura: os fóruns de discussão como parte das CVAO são acessíveis por meio da rede Internet, recurso este ainda não

- universalizado, porquanto está condicionado à existência de infraestrutura pertinente. De qualquer forma, o sujeito participante tem que estar em uma zona em que o acesso é disponibilizado;
2. Habilidade: o sujeito participante tem que ter a habilidade necessária para atuar de forma ativa no contexto das discussões, para que haja efetiva contribuição em direção aos resultados da comunidade. Verifica-se que a maioria das CVAO é efetivamente suportada por uma quantidade mínima de participantes ativos; muitos fazem parte da comunidade contribuindo minimamente, ou seja, poucos colaboram muito e muitos colaboram pouco (VIGOTSKY, 1991, 2001);
 3. Identidade: a real identidade no âmbito dos fóruns ainda tem sido uma incógnita, apesar da inferência de que esses sujeitos são desprendidos e avessos a qualquer tipo de reconhecimento, visto que a maioria deles ainda utiliza apelidos (nicknames) para a sua identificação, evitando expor a sua real identidade, comportamento esse herdado dos fóruns com objetivos de entretenimento ou interação apenas social. No caso dos fóruns voltados para a solução de problemas, a falta da identidade real não corrobora para um descompromisso com o resultado da proposta de solução colocada, conforme os resultados das análises demonstradas nesta pesquisa;
 4. Comunicação não-verbal: centrando a atenção na dinâmica da comunicação estabelecida no âmbito dos fóruns de discussão, convém ter-se a consciência de que a comunicação se articula principalmente por meio do texto, que implica ausência de

elementos circundantes ao processo comunicativo. É claro que a entonação ou forma da escrita, implicitamente, pode demonstrar alguns traços de comportamento do sujeito emissor da mensagem (BAKHTIN, 1995, 1992). É óbvio que, com a constante inovação das ferramentas de comunicação para este fim, em um futuro não previsível, poderá conter outras formas de expressar a comunicação não-verbal, valendo o desenvolvimento de novas pesquisas para analisar o impacto dessas supostas inovações.

5. Redução de custos: indiscutivelmente, a redução de custos é uma realidade absoluta. Um dos diversos fatores que favoreceram o crescimento e expansão exponencial das CVAO é o custo praticamente insignificante de sua manutenção, com a efetiva redução de custos de banda de comunicação para o acesso à rede Internet, servidores, computadores, armazenamento, etc.;
6. Acessibilidade: permite que um sujeito interessado em participar de uma CVAO e possuindo os meios necessários de comunicação, integre-se por meio de um cadastro mínimo e comece a colaborar. A participação efetiva ou não na comunidade escolhida irá depender única e exclusivamente dele, pois não há cobranças ou supervisão, visto que o comportamento imperante é o de colaboração, cooperação e voluntariado;
7. Efeito desinibidor: na suposição de o sujeito não necessariamente ter que se identificar completamente, inclusive expondo o seu perfil e não tendo uma exposição física e pessoal, cria-se um ambiente que de certa forma acolhe aqueles que são tímidos ou têm alguma

dificuldade de exposição de ideias, opiniões e argumentações nas situações presenciais. Os fóruns, marcadamente pela sua característica assíncrona, permitem que o sujeito faça a sua contribuição de forma tranquila, bem pensada, podendo até mesmo revisar o texto por várias vezes, não correndo o risco de expor contribuições, de forma impulsiva ou emocional. O ambiente torna-se acolhedor e reservado, proporcionando um terreno fértil para a participação de sujeitos com personalidade mais acanhada ou tímida.

É notório, entre a comunidade científica que desenvolve pesquisas no âmbito da Internet, que as CVAO emergem em um contexto de interação e socialização paralelo, ainda que vinculado à realidade. Embora com certas singularidades específicas, tais como a atemporalidade, a ausência de espaço físico e a presença da virtualidade, elas formam em última instância a necessidade do desenvolvimento de metodologias específicas para a compreensão dos fenômenos que ocorrem ao seu largo, bem como as sucessivas transformações das formas de comunicação que, irremediavelmente irão condicionar o desenvolvimento da teoria e prática da pesquisa no seu entorno virtual.

Levamos em consideração que o instrumento principal, alvo para a coleta de dados e informação no âmbito das CVAO, são os fóruns de discussão, como comunicação assíncrona que constitui, neste momento histórico, parte dos pilares básicos para canalizar a interação e a socialização das CVAO. “O fórum é um espaço onde é difícil discernir onde começa o meramente técnico e o cultural” (GALVEZ et al., 2003, p.6). Tendo em vista semelhante afirmação, e compreendendo melhor o entorno virtual em termos de marco técnico-social, é

necessário verificar prós e contras que representam, na atualidade, o uso desse instrumento na pesquisa aplicada, para a compreensão dos fenômenos derivados da dinâmica das CVAO.

Cabe considerar que, se o fórum de discussão se torna efetivamente o repositório dos OA derivados das discussões – em que o conhecimento é compartilhado para a efetiva solução de problemas –, são prementes sugestões de aprimoramento e inovação em sua forma, funcionalidade e acessibilidade. Nessa direção, citamos alguns pesquisadores que também têm identificado a mesma problemática, objeto desta investigação, tais como:

- a. Li, Dong e Huang (2009) propõem um *framework* para o desenvolvimento de fóruns semânticos, com o objetivo de facilitar o acesso ao seu conteúdo baseado em palavras chaves ou associação entre palavras, porém o modelo proposto só se aplica aos fóruns propriamente dito, não se aplicando às listas de discussão. É uma proposta tecnológica que vislumbra resolver, em parte, o problema de acesso ao conhecimento armazenado nos repositórios históricos;
- b. Jones (2009, p.26) ao entrevistar o fundador da empresa Internet.com, Alan Meckler, afirma que a Web 3.0 poderá ser chamada de “Web Semântica”, ou seja, todo o seu conteúdo será indexado, de forma a facilitar, sobremaneira, o acesso ao conteúdo procurado;
- c. Nascimento e Santoro (2009) ao pesquisar sobre a limitação de acesso e recuperação do conhecimento gerado pelos fóruns de discussão das CVAO de software livre e de código aberto efetuaram uma revisão de literatura sobre os modelos de argumentação (IBIS, 1970; TOULMIN, 1959; ColaOnto, 2008). As autoras, baseando-se nos dois primeiros modelos de argumentação propõem um modelo denominado ArFoLi (argumentação para fórum de software livre) com o objetivo de “estruturar a discussão do problema, organizar as respostas inseridas e por meio da estruturação facilitar a busca do conhecimento pelos

participantes, ou seja, pela identificação de cada etapa da discussão e atuação do participante”. (p. 148).

Outro aspecto a considerar é que as CVAO tornaram-se um ambiente que, pela sua caracterização e constituição, é de certa forma excludente, ou seja, o sujeito que não possui o perfil citado anteriormente nestas considerações, não terá sucesso na participação efetiva. Tais sujeitos poderão, sim, fazer parte da CVAO, no entanto como membro passivo, observador ou meramente figurativo.

Pode-se afirmar que, se as CVAO inovarem não só em sua forma e funcionalidade, poderão oportunizar acesso a esses sujeitos, hoje em posição passiva, para que possam se tornar ativos e contribuir nos resultados da comunidade, promovendo assim a inclusão destes. Sugere-se ainda, que novas estratégias sejam adotadas, para que delas surjam outros ambientes de aprendizagem e trabalho, que permitam a participação de sujeitos com perfil diferente do predominante.

A hipótese básica do enunciado desta pesquisa incide na constatação de que a ausência de um padrão de conformidade para os OA nos fóruns de discussão dificulta o acesso ao conhecimento explícito gerado pelas discussões, e isso é verdadeiro; entretanto, positivamente tal limitação não tem afetado o desempenho e os resultados das CVAO. Estas continuam vigorosas e cumprindo o seu verdadeiro papel. Todavia, outra constatação ficou evidente, a qual assinala a ampla e irrestrita necessidade de socialização do conhecimento explícito gerado e a sua plena acessibilidade mediante modernas ferramentas de indexação, busca e acesso. Para tanto, há que se estabelecer um padrão de conformidade para os OA, considerando esse tipo de comunidade, o perfil dos geradores do conhecimento, a solução de problemas e a efetivação da aprendizagem, por meio da ação mediada, aqui demonstrada pelos autores BAKHTIN (1995); COLE (1994); VIGOSTSKY (2007); e, WERTSCH (1991).

Obviamente, para se chegar a um padrão de conformidade, há muito que percorrer, porém os resultados e conclusões obtidos nesta pesquisa poderão contribuir para que uma teoria sobre o assunto seja desenvolvida, cujo padrão possa ser criado, proposto, testado, e por fim adotado pelas CVAO existentes e as vindouras.

O fenômeno pesquisado, por meio do olhar da etnografia virtual, da análise das unidades de contexto, da ação mediada e solução de problemas poderá colaborar no processo, para uma melhor compreensão das formas pelas quais os membros e usuários das comunidades virtuais efetivamente usam os objetos de aprendizagem, para o desenvolvimento de atividades relacionadas à solução de problemas. Além disso, poderá também contribuir criando um valor significativo à educação, ciência e tecnologia, servindo como referência para as pesquisas e processos de inovação no campo da aprendizagem em ambientes virtuais.

Com base nos resultados obtidos, decorrentes da investigação proposta, sugere-se apropriar-se dos resultados da investigação de Nelson (1998), que efetuou uma revisão e síntese dos métodos existentes para a formulação de uma teoria baseada nos estudos de PATTERSON, 1977; REIGELUTH, 1983; RICHEY, 1986; SNELBECKER, 1974; SNOW, 1971.

Da experiência de estudos de caso, Nelson (1998) criou (13) treze passos para o processo de construção de uma teoria associada ao processo de aprendizagem para a solução de problemas em comunidades virtuais de pequenos grupos.

Alicerçado nessa teoria, o próximo passo sugerido para futuro prosseguimento dos trabalhos de investigação será a elaboração de um esboço do padrão de conformidade de objetos de aprendizagem voltado para o uso em comunidades virtuais auto-organizadas, conforme propõe NELSON (1998):

1. Definição da proposta da teoria:

Delinear o propósito da teoria, bem como a sua abrangência e a especificação do tipo de problema que ela pretende resolver.

2. Seleção do paradigma da teoria:

Apresentar os autores, conceitos e a base referencial que a teoria utiliza como fundamentação para suporte e sustentação da proposta.

3. Determinação do domínio específico, situação, ou escopo da teoria:

Determinar os limites e fronteiras da teoria, ou seja, qual o seu escopo e o que não faz parte desse escopo. Demonstrar quais são os fundamentos da teoria e quais as condições em que a teoria deverá ser utilizada.

4. Identificação do processo ótimo para o modelo da teoria:

Criar e demonstrar um modelo mental, se possível de forma gráfica, que sintetize o processo de desenvolvimento da teoria.

5. Desenvolvimento de critérios gerais para os objetivos, métodos e condições:

Desenvolver os critérios para avaliar os objetivos, métodos e condições de aplicação da teoria.

6. Desenvolvimento dos objetivos da teoria:

Definir claramente os objetivos pelos quais a teoria está sendo desenvolvida.

7. Desenvolvimento dos métodos da teoria:

Formular os métodos apropriados e estratégias necessárias para a realização dos objetivos da teoria.

8. Identificação das condições da teoria:

Criar um guia de orientações para que os usuários possam utilizar a teoria de acordo com os propósitos especificados.

9. Criação de uma variável taxonômica para teoria:

Criar uma variável que represente o grupo de objetivos e métodos e seus relacionamentos.

10. Finalização do protótipo da teoria:

Apresentar uma proposta de protótipo da teoria para as aplicações prévias e validações.

11. Formalização da pesquisa do protótipo da teoria:

Refinar o protótipo baseando-se em aplicações simuladas, propondo melhorias para que a teoria possa ter maior congruência entre seus objetivos e métodos.

12. Revisão dos objetivos, métodos e condições:

Uma vez que a teoria tenha sido verificada em sua usabilidade e testada conceitualmente, os objetivos, métodos e condições também deverão ser revistos.

13. Descrição da teoria:

Após as fases acima, a teoria, finalmente, deverá ser formalizada e disseminada, para exame e crítica de outros pesquisadores ou usuários.

O conteúdo desta pesquisa possui referenciais teórico-metodológicos para o desenvolvimento da teoria relativa ao processo de aprendizagem para a solução de problemas em comunidades virtuais de pequenos grupos, baseada na metodologia proposta por NELSON (1998), além de explicitar direções futuras, que poderão ser investigadas.

Os resultados que estão sendo atingidos pelas CVAO são significativos, ou seja, as grandes organizações estão revendo as suas estratégias e as suas boas práticas de desenvolvimento e produção de software, bem como as suas relações de poder e gestão de equipes e pessoas voltadas para o trabalho colaborativo, em razão do sucesso alcançado por essas comunidades.

Por certo, quando do desenvolvimento e liberação de um padrão de conformidade para os objetos de aprendizagem derivados dos fóruns de discussão das CVAO interessadas na

produção de SLCA, uma nova história começará a ser escrita. Nesse contexto, haverá um salto significativo do conhecimento, que vem sendo acumulado ao longo de mais de uma década, e poderá ser ampliada a sua acessibilidade, desde que novas teorias, conceitos e ferramentas inovadoras sejam desenvolvidos. Com a adoção, apropriação e uso desses recursos, é fatal a exponencialização dos resultados, que, por sua vez, influenciarão inevitavelmente o comportamento, cultura e formas de aprendizagem.

REFERÊNCIAS

- ABBAGNANO, N. (1982). **Dicionário de filosofia**. 2.ed. São Paulo: Mestre Jou.
- ADVANCED DISTRIBUTED LEARNING NETWORK WEBSITE. (2000). Disponível em:<<http://www.adlnet.org>>. Acesso em: 23 Jan. 2008.
- ADVANCED DISTRIBUTED LEARNING NETWORK WEBSITE. (2001). **Sharable content objects reference model**. Disponível em:<<http://www.adlnet.org>>. Acesso em: 31 Jan. 2009.
- ALBANESE, M.A.; MITCHELL, S. (1993). Problem-based learning: a review of literature on its outcomes and implementation issues. **Academic Medicine**, Philadelphia, v.68, p.52-81.
- AUSUBEL, D.P. (1968). **Educational psychology: a cognitive view**. New York: Holt, Rinehart and Winston.
- BAKHTIN, M. (1992). **Estética da criação verbal**. São Paulo: Martins Fontes.
- _____. (1995). **Marxismo e filosofia da linguagem**. 7.ed. São Paulo: Hucitec.
- BARDIN, L. (2000). **Análise de conteúdo**. Lisboa: Ed.70.
- BARRIT, C.; LEWIS, D. (2001). **Reusable learning object strategy**. Disponível em:<http://www.cisco.com/warp/public/10/wwtraining/elearning/learn/whitepaper_docs/rlo_strategy_v3-1.pdf>. Acesso em: 23 July 2005.
- BAUME, R.; BERTOLUS, J. (1995). **Os Novos senhores do mundo: a louca história dos multimídia**. Lisboa: Teorema.
- BENTES, A.C. (2005). Gêneros e ensino: algumas reflexões sobre a produção de materiais didáticos para a educação de jovens e adultos. In: KARWOSKI, A.M. et al. (Org.). **Gêneros textuais: reflexão e ensino**. Palmas e União da Vitória: Kaygangue.
- BERNSTEIN, B. (2003). A Pedagogização do conhecimento: estudos sobre recontextualização. **Cadernos de Pesquisa**, São Paulo, n.120, p.75-110, nov.

BEZERRA, P. (2001). Prefácio. In: VIGOTSKI, L.S. **A Construção do pensamento e da linguagem**. Tradução de Paulo Bezerra. São Paulo: Martins Fontes.

BJÖRCK, U. (2001). Distributed problem-based learning in social economy - a study of the use of a structured method for education. In: ANNUAL MEETING OF THE AMERICAN EDUCATIONAL RESEARCH ASSOCIATION, 2001, Seattle. **Proceedings** Washington: American Educational Research Association. p.85-103.

BLOOM, B.S. (1956). **Taxonomy of educational objectives, handbook 1: cognitive domain**. New York: Longmans Green.

BOEHM, B. (1981). **Software engineering economics**. Englewood Cliffs: Prentice-Hall.

BOGOYAVLENSKY, N; MENCHINS- KAYA, N.A. (1991). A Psicologia da aprendizagem, de 1900 a 1960. In: LURIA, A.R.; LEONTIEV, A.N.; VIGOTSKY, L.S. **Psicologia e pedagogia**. 2.ed. Lisboa: Estampa.

BOUTINET, J.P. (2002). **Antropologia do projeto**. 5.ed. Porto Alegre: Artmed.

BRONFENBRENNER, U. (1979). **A Ecologia do desenvolvimento humano: experimentos naturais e planejados**. Porto Alegre: Artes Médicas.

BROWN, A.L. (1994). The Advancement of learning. **Educational Researcher**, Washington, v.23, n.3, p.4-12.

BROWN, J.S.; COLLINS, A.; DUGUID, P. (1989). Situated cognition and the culture of learning. **Educational Researcher**, Washington, v.18, n.1, p.32-42.

_____. (1990). Anchored instruction and situated cognition revisited: cognition & technology group at Vanderbilt. **Educational Researcher**, Washington, v.19, n.6, p.2-10.

BURKE, K. (1973). Self- regulated learning: implications for the design of instructional media. **Contemporary Educational Psychology**, Duluth, v.11, n.4, p.404-427, Oct.

CAMERON, T.; BARROWS, H.S.; CROOKS, S.M. (1999). **Distributed problem-based learning at southern Illinois University School of Medicine**. Paper presented at the Computer supported Collaborative Learning Conference (CSCL'99), Stanford, 1999.

CASTELLS, M. (2000). **O Fim do milênio**. São Paulo: Paz e Terra.

_____. (2007). **A Sociedade em rede**. São Paulo: Paz e Terra. v.1.

CHRISTENSEN, R.; HANSEN, A.J. (1987). **Teaching and the case method**. Boston: Harvard Business School Press.

CLARK, R. (1989). **Developing technical training**: a structured approach for the development of classroom and computer-based instructional materials. [S.l.]: Performance Technology.

COLE, M. (1985). The Zone of proximal development: where culture and cognition create each other. In: WERTSCH, J.V. (Org.). **Culture, communication and cognition: vygotskian perspectives**. Cambridge: Cambridge University Press.

_____. (1987). **The Theory of A.R. Luria, functions of spoken language in the development of higher mental processes**. Hillsdale: Lawrence Erlbaum.

_____. (1994). A Conception of culture for a communication theory of mind. In: VOCATE, D. (Org.). **Intrapersonal communication: different voices, different minds**. Hillsdale: Lawrence Erlbaum.

_____. (1997). **Using new information technologies in the creation of sustainable after school literacy activities**: from intervention to maximising the potential. San Diego: LCHC. Terceiro Relatório anual, julho de 1996-julho de 1997, submetido a Andrew W. Mellon Foundation.

COLE, M.; MEANS, B. (1986). **Cognición y pensamiento**. Buenos Aires: Paidós.

COLE, M.; SCRIBNER, S. (1977). **Cultura y pensamiento**. Relación de los procesos cognoscitivos con la cultura. México: Limusa.

COLE, M.; ENGSTRÖM, Y.; VASQUEZ, O. (Org.). (1997). **Mind, culture and activity: seminal papers from the laboratory of comparative human cognition**. Cambridge: Cambridge University Press.

COLE, M. et al. (1971). **The Cultural context of learning and thinking: an exploration in experimental anthropology**. New York: Basic Books.

COSTA, R. (2005). Comunidades virtuais. **Informática na Educação: Teoria & Prática**, Porto Alegre, v.2, n.2, p.55-73, jul./dez.

CRICHTON, S.; KINASH, S. (2003). Virtual ethnography: interactive interviewing online as method. **Canadian Journal of Learning and Technology**, v.29, n.2. Disponível em: <http://www.cjlt.ca/content/vol29.2/cjlt29-2_art-5.html>. Acesso em: 15 May 2009.

DANIELS, H. (Org.). (2002). **Uma Introdução a Vygotsky**. São Paulo: Loyola.

_____. (2003). **Vygotski e a pedagogia**. São Paulo: Loyola.

DEWEY, J. (1959). **Democracia e educação**: introdução à filosofia da educação. 4.ed. Tradução de Godofredo Rangel e Anísio Teixeira. São Paulo: Nacional.

_____. (1981). **The Philosophy of John Dewey**. Chicago: The University of Chicago Press.

_____. (1994). **Democracy and education**: an introduction to the philosophy of education. New York: The Free Press.

DOMINGOS, A.M. et al. (1986). **A teoria de Bernstein**: em sociologia da educação. Lisboa: Fundação Calouste Gulbenkian.

DRAFT standard for learning technology - learning object metadata. (2002). Disponível em: <http://ltsc.ieee.org/doc/wg12/LOM_1484_12_1_v1_Final_Draft.pdf>. Acesso em: 18 Sept. 2007.

ENGELS, F. (1979). **A Dialética da natureza**. 6.ed. São Paulo: Paz e Terra.

ERIKSSON, D.M.; WULF, V. (1999). Self-organising social systems: a challenge to computer supported cooperative work. **Cybernetics and Human Knowing**, Charlottesville, v.6, n.2, p.1-4.

FARACO, C.A. (1996). O Dialogismo como chave de uma antropologia filosófica. In: FARACO, C.A.; TEZZA, C.; CASTRO, G. (Org.). **Diálogos com Bakhtin**. Paraná: UFPR.

_____. (1988). Bakhtin: a invasão silenciosa e a má leitura. In: FARACO, C.A. et al. **Uma Introdução a Bakhtin**. Curitiba: Hatier.

FETTERMAN, D. (1989). **Ethnography**: step by step. Newbury Park: Sage.

FRANCO, M.L. (2005). **Análise de conteúdo**. 2.ed. Brasília: Liber Livro. (Série Pesquisa, 6).

GÁLVEZ, A.M. et al. (2003). **Los Espacios de interacción virtual como dispositivos sociotécnicos**. Disponível em: <http://cv.uoc.edu/~grc0_000199_web/pagina_personal/Dispositivo.pdf>. Acesso em: 12 ago. 2009.

GARNIER, C. et al. (1996). **Após Vygotsky e Piaget**. Porto Alegre: Artes Médicas.

GEERTZ, C. (1993). **A Interpretação das culturas**. Rio de Janeiro: LTC.

GONZALEZ, L.A. (2005). **Um Modelo conceitual para aprendizagem colaborativa baseada na execução de projetos pela web**. São Paulo: USP.

GRILLO, S.V.C. (2006). A Metalinguística: por uma ciência dialógica da linguagem. **Horizontes**, Bragança Paulista, v.24, n.2, p.121-128, jul./dez.

GRINT, K.; WOOLGAR, S. (1977). **The Machine at work**: technology, work and organization. London: Cambridge.

HAGEL III, J.; ARMSTRONG, A. (1997). **Net gain**. Boston: Harvard Business School Press.

HARDIN, G. (1968). The Tragedy of the commons. **Science**, Washington, v.162, p.1243-1248.

HARVEY, D. (1992). **Condição pós-moderna**. 5.ed. São Paulo: Loyola.

HEGEL, G.W.F. (1995). **A Razão na história**. Lisboa: Ed.70.

HENRIQUES, M.J.R. (2007). As Raízes marxistas de Mikhail Bakhtin. **Estudos Linguísticos**, São Paulo, v.36, p.357-362, set./dez.

HERNÁNDEZ, F. (2000). **Cultura visual, mudança educativa e projeto de trabalho**. Porto Alegre: Artmed.

HERNÁNDEZ, F.; VENTURA, J. (1998). **A Organização do currículo por projetos de trabalho**: o conhecimento é um caleidoscópio. Porto Alegre: Artmed.

HERRING, S. (1996). **Computer mediated communication**: linguistic, social and cross-cultural perspectives. Amsterdam: John Benjamins.

HIMANEN, P. (2001). **The Hacker ethic**. New York: Random House.

HINE, C. (2000). **Virtual ethnography**. London: SAGE.

HODGINS, H.W. (2000). **Into the future: a vision paper**. Commission on Technology and Adult Learning. Disponível em: <<http://www.learnativity.com/download/MP7.pdf>>. Acesso em: 26 Apr. 2005.

HOLLAND, D.; COLE, M. (1995). Between discourse and schema: reformulating a cultural-historical approach to culture and mind. **Anthropology and Education Quarterly**, Washington, v.26, n.4, p.475-490.

HUMPHREYS, B.; JOHNSON, R.T.; JOHNSON, D.W. (1982). Effects of cooperatives competitive and individualistic learning on students achievement in science class. **Journal of Research in Science Teaching**, New York, v.19, n.5, p.351-356, May.

JACOBS, J. (1961). **The Death and life of great american cities**. New York: Vintage Books.

JOHNSON, D.W.; JOHNSON, R.T. (1997). **Joining together: group theory and group skills**. 6thed. Boston: Allyn & Bacon.

_____. (1989). **Cooperation and competition: theory and research**. Edine: Interaction.

JOHNSON, S. (2001). **Cultura da interface**. Rio de Janeiro: Jorge Zahar.

JONASSEN, D. et al. (1991). Objectivism versus constructivism: do we need a new philosophical paradigm?. **Educational Technology, Research and Development**, Washington, v.39, n.3, p.5-14.

JONES, B. L. (2009) **WEB 2.0 Heroes**. São Paulo: Digerati Books.

KENSKI, V.M. (2003). **Tecnologias e ensino presencial e a distância**. Campinas: Papirus.

_____. (2008). Educação e comunicação: interconexões e convergências. **Educação & Sociedade**, Campinas, v.29, n.104, 647-665, out.

_____. (2009). Novos processos de interação e comunicação no ensino mediado pelas tecnologias. In: PIMENTA, S.G.; ALMEIDA, M.I. (Org.). **Pedagogia universitária**. São Paulo: EDUSP.

KERCKHOVE, D. (1999). **Connected intelligence: the arrival of the WEB society**. Toronto: Somerville House Books.

KIM, A.J. (2000). **Community building on the Web**. Berkeley: Peachpit.

KITCHIN, R. (1998). **Cyberspace**: the world in the wires. Chichester: Wiley.

KRUGMAN, P. (1996). **The Self-organizing economy**. Oxford: Blackwell.

LATOUR, B.; WOOLGAR, S. (1986). **Laboratory life**: the construction of scientific facts. New Jersey: Princeton University.

LAVE, J. (1988). **Cognition in Practice**: mind, mathematics, and culture in everyday life. Cambridge: Cambridge University Press.

LAVE, J.; WENGER, E. (1990). **Situated learning**: legitimate peripheral participation. Cambridge: Cambridge University Press.

LEARNING objects metadata standard (current draft). (2001). Disponível em: <<http://ltsc.ieee.org>>. Acesso em: 18 Sept. 2007.

LEARNING objects metadata working group homepage. (2000). Disponível em: <<http://ltsc.ieee.org/wg12/index.html>>. Acesso em: 11 Oct. 2009.

LEARNING technology standards committee website. (2000). Disponível em: <<http://ltsc.ieee.org/>>. Acesso em: 11 out. 2009.

LEITE, H.A. (1996). Pedagogia de projetos: intervenção no presente. **Presença Pedagógica**, Belo Horizonte, v.2, n.8, p.24-33, mar./abr.

LEMO, A.; PALACIOS, M. (Org.). (2001). **Janelas da cibercultura**. Porto Alegre: Sulina.

LEONTIEV, A.N. (1932). Studies on the cultural development of the child. **Journal of Genetic Psychology**, Worcester, v.40, p.52-83.

LÉVY, P. (1993). **Cibercultura**. São Paulo: Ed.34.

_____. (1994). **As Tecnologias da inteligência**. Rio de Janeiro: Ed.34.

_____. (1996). **O Que é o virtual**. Rio de Janeiro: Ed.34.

LÉVY, P. (2001). **A Conexão planetária**: o mercado, o ciberespaço, a consciência. São Paulo: Ed.34.

_____. (2004). **As Tecnologias da inteligência: o futuro do pensamento na era da informática**. 13.ed. São Paulo: Ed.34.

LI, Y.; MINGKAI, D.; HUANG, R. (2009) Toward a semantic forum for active collaborative learning. **Journal of Educational Technology & Society**, Athabasca, v.12, n.4, p.71-86.

LINUX MAGAZINE. (2006). São Paulo, n.20, jun.

LOM. (2000) LOM Working Draft v. 4.1. Disponível em:
<<http://ltsc.ieee.org/doc/wg12/LOMv4.1.htm>> Acesso em: 14 Set. 2007.

LÜDKE, M. (2004). O Trabalho com projetos e a avaliação da educação básica. In: ESTEBAN, M.; HOFFMANN, J. (Org.). **Práticas avaliativas e aprendizagens significativas**. Porto Alegre: Mediação.

LURIA, A.R. (1988). Vigotskii. In: VYGOTSKY, L.S.; LURIA, A.R.; LEONTIEV, A.N. **Linguagem, desenvolvimento e aprendizagem**. São Paulo: Ícone; EDUSP.

LURIA, R.; LEONTIEV, N.; VIGOTSKY, L.S. (1991). **Psicologia e pedagogia**. 2.ed. Lisboa: Estampa. 2v.

MAETERLINCK, M. (1927). **The Life of the white ant**. London: Geroge Allen and Unwin.

MAGER, B.F. (1975). **Preparing instructional objectives**. Belmont: David S. Lake.

MARTIN, C. (1999). **Net future**. New York: McGraw-Hill.

MARTINEZ, M. (2001). Designing learning objects to mass customize and personalize learning. In: WILEY, D.A. **The Instructional use of learning objects**. Bloomington: Association for Educational Communications and Technology.

McCONNELL, D. (2002). Action research and distributed problem-based learning in continuing professional education. **Distance Education**, Toowoomba, v.23, n.1, p.59-83.

_____. (2004). **Designing for distributed networked learning communities**. Disponível em:<<http://www.formare.erickson.it/>>. Acesso em: 13 Aug. 2009.

McLELLAN, H. (1995a). How many people can wield one pen?. **American Medical Writers Association Journal**, Rodville, v.10, n.1, p.11-13.

MCLELLAN, H. (1995b). **Situated learning perspectives**. Englewood Cliffs: Educational Technology.

MERRIL, M.D. (1983). Component display theory. In: REIGELUTH, C.M. **Instructional design theories and models**. Hillsdale: Erlbaum.

_____. (2001). **Instructional design theory**. Englewood Cliffs: Educational Technology.

MILLER, D.; SLATER, D. (2000). **The Internet: an ethnographic approach**. London: Berg.

MOREIRA, A.F.B. (2005). **Currículo: questões atuais**. Campinas: Papirus.

MUNÕZ, R.A. (2007). Nuevos medios o nuevas de indagación? Una propuesta metodológica para la investigación social on-line a través del foro de discusión. **FQS Forum: qualitative social research**, Canada, v.8, n.3, Art.37, Sept.

NASCIMENTO, L.C.; SANTORO, F.M. et al. (2009). Um modelo de argumentação para apoiar as comunidades virtuais de software livre. In: **VI SIMPÓSIO BRASILEIRO DE SISTEMAS COLABORATIVOS**, 2009, Fortaleza. Anais. Fortaleza: IEEE Computer Society, 2009. p. 144-153.

NELSON, L.M. (1998). **Collaborative problem solving: an instructional theory for learning through small group interaction**. 142 f. Tesis (Doctor) - Department of Instructional Systems Technology, Indiana University, Indiana, 1998.

_____. (1999). Collaborative problem solving. In: REIGELUTH, C.M. **Instructional design theories and models: a new paradigm of instructional theory**. Hillsdale: Lawrence Erlbaum. p.241-267.

NEWMAN, A.; GRIFFIN, P.; COLE, M. (1989). **The Construction zone: working for cognitive change in school**. New York: Cambridge University Press.

ORRILL, C.H. (2001). Supporting online PBL: design considerations for collaborative problem - solving communication tools. In: ANNUAL MEETING OF THE AMERICAN EDUCATIONAL RESEARCH ASSOCIATION, 2001, Seattle. **Proceedings**. Washington: American Educational Research Association.

PALLOFF, R.M.; PRATT, K. (2002). **Construindo comunidades de aprendizagem no ciberespaço**. Porto Alegre: Artmed.

PATTERSON, C. (1977). **Foundation for a theory of instruction and educational psychology**. New York: Harper & Row.

PETERSON, T.L.L.; MONSON, D.W. (1999). Three common properties of efficient online instructional support systems. **The ALN Magazine**, v.3, n.2. Disponível em: <http://www.aln.org/alnweb/magazine/vol13_issue2/wiley.htm>. Acesso em: 21 Jan. 2008.

PHELPS, E.; DAMON, W. (1989). Problem solving with equals: peer collaboration on a context for learning mathematics and spatial concepts. **Journal of Educational Psychology**, Baltimore, v.81, n.4, p.639-646, Dec.

PRESSMAN, R.S. (1992). **Software engineering: a practitioner's approach**. New York: McGraw-Hill.

RATNER, C. (1995). **A Psicologia socio-histórica de Vygostsky: aplicações contemporâneas**. Porto Alegre: Artes Médicas.

RAYMOND, E. (1999). **The Cathedral & the bazaar: musings on linux and open source by an accidental revolutionary**. Monterey: O'Reilly.

RAYMOND, E.; PERENS, B. (1998). **Open sources: voices from the open source revolution**. Disponível em: <<http://onlamp.com/onlamp/2008/02/12/a-look-back-at-10-years-of-osi.html>>. Acesso em: 13 Oct. 2009.

REIGELUTH, C.M. (1983). Instructional design. What is it and why is it?. In: _____. **Instructional design theories and models: an overview of their current status**. Hillsdale: Lawrence Erlbaum.

REIGELUTH, C.M.; NELSON, L.M. (1997). A New paradigm of ISD?. In: BRANCH, R.C.; MINOR, B.B. **Educational media and technology yearbook**. Englewood: Libraries Unlimited. v.22, p.24-35.

RHEINGOLD, H. (1992). **A Slice of life in my virtual community**. Disponível em: <<http://www.communities.com/paper/settlmnt.html>>. Acesso em: 13 May 2006.

RICHEY, R.C. (1986). **The Theoretical and conceptual bases of instructional design**. London: Kogan Page.

RIVAS, T. (1998). **As Tecnologias colaborativas em ambiente distribuído de aprendizagem como um recurso de formação continuada do engenheiro de produção**. 109 f. Dissertação (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 1998.

ROGOFF, B. (1990). **Apprenticeship in thinking**: cognitive development in social context. New York: Oxford University Press.

RONTELTAP, F.; EURELINGS, A. (2002). Activity and interaction of students in an electronic learning environment for problem-based learning. **Distance Education**, Toowoomba, v.23, n.1, p.11-22.

ROWN, A.L. (1994). The Advancement of learning. **Educational Researcher**, Washington, v.23, n.3, p.4-12.

ROYCE, W.W. (1987). Managing the development of large software systems: concepts and techniques. In: ANNUAL IEEE WESCON CONVENTION, 9., 1987, Monterey. **Proceedings...** New York: IEEE Computer Society. p.322-338.

RUÉ, J. (2007). **Enseñar en la Universidad**: El EEES como reto para la educación superior. Madrid: NARCEA.

SANTAELLA, L.; NÖTH, W. (2004). **Comunicação e semiótica**. São Paulo: Hacker.

SANTOS, B.S. (2001). **Pela mão de Alice**: o social e o político na pós-modernidade. 2.ed. São Paulo: Cortez.

SCHAFF, A. (1991). **A Sociedade informática**. 2.ed. São Paulo: Brasiliense.

SCHANK, R.C.; BERMAN, T.R.; MACHPHERSON, K.A. (1999). Learning by doing. In: REIGHELUTH, C.M. **Instructional design theories and models**: a new paradigm of instructional theory. Hillsdale: Lawrence Erlbaum.

SCHNEIDER, M.B.D. (2009). **As Repercussões da escrita eletrônica no desenvolvimento da escrita manual na sala de aula e as consequências no código verbal**. Disponível em:<www.educacaoonline.pro.br>. Acesso em: 23 set. 2009.

SCHMIDT, H.G. (1993). Foundations of problem-based learning: some explanatory notes. **Educational Psychology Review**, New York, v.5, n., p.205-219.

SIQUEIRA FILHO, V.; BORTOLETO, M. (2006). **A Interação na internet**: o gênero discursivo na rede mundial de computadores e suas implicações na relação entre indivíduo e aprendizagem. Disponível em:<www.educacaoonline.pro.br>. Acesso em: 23 set. 2009.

SILVA, F.P. (2006). Alforria para a sua empresa. **Revista Linux Magazine**, São Paulo, n. 20, p.10-14, junho, 2006.

SLAVIN, R.E. (1990a). Research on cooperative learning: consensus and controversy. **Educational Leadership**, Washington, v.47, n.4, p.52-54, Dec./Jan.

_____. (1990b). **Cooperative learning: theory, research and practice**. Englewood Cliffs: Prentice Hall.

SHARAN, S. (1994). **Handbook of cooperative learning methods**. Westport: Greenwood.

SHARAN, Y.; SHARAN, S. (1994). Group investigation in the cooperative classroom. In: SHARAN, S. **Handbook of cooperative learning methods**. Westport: Greenwood.

_____. (1984). **Cooperative learning in the classroom: research in desegregated schools**. Hillsdale: Erlbaum.

SHARAN, Y.; SHARAN, S.; HERTZ-LAZAROWITZ, R. (1980). A Group investigation method of cooperative learning in the classroom. In: SHARAN, P. et al. **Cooperation in education**. Provo: Brigham Young University Press.

SLOTNICK, H.B. (1996). How doctors learn: the role of clinical problems across the medical school-to-practice continuum. **Academic Medicine**, Philadelphia, v.71, p.28-34.

SNELBECKER, G.E. (1974). **Learning theory, instructional theory, and psychoeducational theory**. New York: McGraw-Hill.

SNOW, R.E. (1971). Theory construction for research on teaching. In: TRAVERS, R.M.W. **Second handbook of research on teaching**. Chicago: Rand McNally.

SOUTH, J.B.; MONSON, D.W. (2002). A University-wide system for creating, capturing, and delivering learning objects. In: WILEY, D. **The Instructional use of learning objects**. Bloomington: Association for Educational Communications and Technology.

SPOHRER, J. (2001). Educational authoring tools and the educational object economy: introduction to the special issue from the east/west group. **Journal of Interactive Media in Education**. Disponível em: <<http://www-jime.open.ac.uk/>>. Acesso em: 12 Oct. 2009.

STALLMAN, R. (1998). **The GNU project**. Disponível em: <<http://www.gnu.org/gnu/thegnuproject.html>>. Acesso em: 13 Oct. 2009.

STEINKUEHLER, C.A. et al. (2002). Cracking the resource nut with distributed problem-based learning in secondary teacher education. **Distance Education**, Toowoomba, v.23, n.1, p.23.39.

STRATHERN, M. (1992). **After nature**. Cambridge: Cambridge University Press.

SUCHMAN, L. (1988). **Plans and situated actions: the problem of human/machine communication**. Cambridge: Cambridge University Press.

TAURION, C. (2004). **Software livre: potencialidades e modelos de negócios**. Rio de Janeiro: Braspost Livros e Multimídia.

VERNON, D.T.A.; BLAKE, R.L. (1993). Does problem-based learning work? A meta analysis of evaluative research. **Academic Medicine**, Philadelphia, v.68, p.550-563.

VIGOTSKY, L.S. (1978). **Mind in society**. Cambridge: Harvard University Press.

_____. (1981a). The Development of higher mental functions. In: WERTSCH, J.V. **The Concept of activity in soviet psychology**. Nova York: Sharpe.

_____. (1981b). The Genesis of higher mental functions. In: WERTSCH, J.V. **The Concept of activity in soviet psychology**. New York: Sharpe.

_____. (2001a). **A Construção do pensamento e da linguagem**. Tradução de Paulo Bezerra. São Paulo: Martins Fontes.

_____. (2001b). **Psicologia pedagógica**. Tradução de Paulo Bezerra. São Paulo: Martins Fontes.

_____. (2007). **A Formação social da mente: o desenvolvimento dos processos psicológicos superiores**. Tradução de José Cipolla Neto et al. São Paulo: Martins Fontes.

WATERMAN, P. (1998). **Globalization, social movements and the new internationalisms**. London: Mansell.

_____. (1999). **The Fires of capitalism and the phoenix of internationalism**. Global Solidarity Dialogue. Disponível em: <<http://www.antenna.nl/~waterman/fires.html>>. Acesso em: 15 June 2008.

WEBB, N.M. (1982). Student interaction and learning in small groups. **Review of Educational Research**, Washington, v.52, p.421-445.

WERTSCH, J.V. (1981). **The Concept of activity in soviet psychology**. Nova York: Sharpe.

_____. (1985). **Cultural, communication and cognition: Vygotskian perspectives.** Cambridge: Harvard University Press.

_____. (1991). **Voices of the mind: a sociocultural approach to mediated action.** Cambridge: Harvard University Press.

_____. (2002). Computer mediation, PBL, and dialogicality. **Distance Education,** Toowoomba, v.23, n.1, p.105-108.

WILLIAMS, R.L.; COTHREL, J. (2000). Four smart ways to run online communities. **Sloan Management Review,** Cambridge, v.41, n.4, p.81-91.

WILEY, D.A. (2000). **Learning object design and sequencing theory.** 131 f. PhD. Thesis (Doctor) - Department of Instructional Psychology and Technology, Brigham Young University, Brigham, 2000.

_____. (2002). **The Instrutlional use of learning objects.** Bloomington: Association for Educational Communications and Technology.

WILEY, D.A.; RECKER, M.M.; GIBBONS, A. (2000). **In Defense of the by-hand assembly of learning objects.** Disponível em: <<http://wiley.ed.usu.edu/docs/axiomatic.pdf>>. Acesso em: 12 Oct. 2009.

WILEY, D.A.; RECKER, M.M.; GIBBONS, A. (2001). **The Reusability paradox.** Disponível em: <<http://relt.usu.edu/whitepapers/paradox.html>>. Acesso em: 11 Oct. 2009.

WILSON, B.; ATKINSON, M. (2005). Rave and straightedge, the virtual and the real. **Youth and Society,** Newbury Park, v.36, n.3, p.276-311.

WON-SEOK, Y.; KIL-SOO, S.; MOON-BONG, L. (2002). Exploring the factors enhancing member participation in virtual communities. **Journal of Global Information Management,** Hrshey, v.10, n.3, p.55-71.

WOOD, D.; BRUNER, J.; ROSS, G. (1976). The Role of tutoring in problem solving. **Journal of Child Psychology and Psychiatry,** New York, v.17, n.2, p.89-100, Jan.

WULF, V. (1999). Evolving cooperation when introducing groupware: a self-organization perspective. **Cybernetics and human knowing,** Charlottesville, v.6, n.2, p.55-75.

YOO, W.S.; SUH, K.S.; LEE, M.B. (2002). **Exploring the factors enhancing member participation in virtual communities.** Hershey: Idea.

ANEXOS

Anexo A – Home Page da Comunidade Linux – Kernel

The Linux Kernel Archives

Welcome to the Linux Kernel Archives. This is the primary site for the Linux kernel source, but it has much more than just Linux kernels: [Frequently Asked Questions](#)

Protocol	Location
HTTP	http://www.kernel.org/pub/
FTP	ftp://ftp.kernel.org/pub/
RSYNC	rsync://rsync.kernel.org/pub/

The latest stable version of the Linux kernel is: [2.6.29](#) 2009-03-23 23:30 UTC [E](#) [V](#) [VI](#) [C](#) [ChangeLog](#)

The latest [snapshot](#) for the stable Linux kernel tree is: [2.6.29-git4](#) 2009-03-29 00:01 UTC [B](#) [V](#) [VI](#) [C](#) [ChangeLog](#)

The latest 2.4 version of the Linux kernel is: [2.4.37](#) 2008-12-02 08:13 UTC [E](#) [V](#) [VI](#) [C](#) [ChangeLog](#)

The latest 2.2 version of the Linux kernel is: [2.2.28](#) 2004-02-25 00:28 UTC [E](#) [V](#) [C](#) [ChangeLog](#)

The latest [prepatch](#) for the 2.2 Linux kernel tree is: [2.2.27-rc2](#) 2005-01-12 23:55 UTC [B](#) [V](#) [C](#) [ChangeLog](#)

The latest [-mm patch](#) to the stable Linux kernels is: [2.6.28-rc2-mm1](#) 2008-10-29 06:29 UTC [V](#)

F = full source, B = patch baseline, V = view patch, VI = view incremental, C = current [changesets](#)
 Changesets are provided by the kernel authors directly. Please don't write the webmaster about them.
[Customize the patch viewer](#)

Site News

- December 31, 2008: [patchwork.kernel.org](#) is now available for general use. It is currently only monitoring the Linux Kernel mailing-list, but should be useful to kernel developers in dealing with patches flying across the wire.
- September 19, 2008: mirrors.kernel.org has been flipped over to using our new GeoDNS based bind server (named-geodns). This means that, at the dns query level, our servers will attempt to direct you to the nearest/fastest kernel.org mirror for your request. This means that you no longer have to use mirrors.us.kernel.org or mirrors.eu.kernel.org to generally route you to the right place. This does mean a change to mirrors.kernel.org no longer explicitly pointing at mirrors.us.kernel.org. Additional information on named-geodns will be forth coming, check back here for an addendum soon.
- September 12, 2008: Thanks to a [fundraising effort](#) by the [Linux Foundation](#), John "warthog" Hawley will be joining kernel.org full time as administrator starting Monday, September 15. John has already been the primary administrator on a volunteer basis for the past three years, and will now be able to devote all his time toward kernel.org.
- March 24, 2008: After far too long of a bringup, [mirrors.eu.kernel.org](#) is now available. Again thanks to [Hewlett-Packard](#), [ISC](#) and [Umeå Universitet](#), we now have a full complement of mirror servers in Amsterdam and Umeå. Huge thanks to HP for donating two additional DL585 G5's for this effort, and huge thanks to ISC and UMU for volunteering to host our servers!
- Aug 17, 2007: We have been recognized by the IRS as a [501\(c\)3 nonprofit corporation](#).
- Jun 30, 2007: [European servers!](#) Thanks to [Hewlett-Packard](#), [ISC](#) and [Umeå Universitet](#), we are introducing new kernel.org servers at ISC in Amsterdam and Umeå in Sweden. These servers are available as a round-robin as [www.eu.kernel.org](#), [ftp.eu.kernel.org](#), and [rsync.eu.kernel.org](#). Additionally, the hostnames [www.all.kernel.org](#), [ftp.all.kernel.org](#), and [rsync.all.kernel.org](#) are worldwide round-robins among all four servers. We hope to have Geo- or BGP-based IP directing in the future, and hope to soon provide additional services in Europe. Huge thanks to HP for donating two additional DL380 G5's for this effort, and huge thanks to ISC and UMU for volunteering to host our servers!
- The bandwidth graphs have now moved [here](#).
- May 20, 2007: [New servers!](#) We have received two new [DL380 G5](#) servers from [Hewlett-Packard](#). These servers are dual quad-core Xeon E5335 servers with 16 GB RAM per server, and 15000 RPM SAS disks. These servers have replaced the old servers for all local-contents services (www, ftp, and rsync); the old servers are now dedicated to running [mirrors.kernel.org](#). Once again, thanks to HP for your continued support of kernel.org!
- Mar 13, 2007: [10 years of kernel.org!](#) The Linux Kernel archives were [announced to the world on March 13, 1997](#). Thanks to everyone that has made kernel.org possible over the years, our sponsors, staff volunteers and of course the Linux community for making it a worthwhile effort in the first place!
- Feb 4, 2006: [Hewlett-Packard](#) has been kind enough to replace our internal master server with a brand new [ProLiant DL385](#) with a dual-core Optron 275 and 4 GB of RAM. Once again, huge thanks to HP!
- Sep 30, 2005: By popular demand, there is now a [kernel.org FAQ](#).
- Aug 21, 2003: Please don't use [finger.kernel.org](#) for any sort of automatic monitoring. The number of automatic bots hitting this port is causing the finger daemon to shut down more often than not. The same information is available from [http://www.kernel.org/kdist/finger_banner](#).

The Linux Kernel Archives Mirror System

To improve access for everyone, a number of sites around the world have provided mirrors of this site, which may be faster to use than the master archive itself. Please see <http://www.kernel.org/mirrors/> for information about how to connect to a participating mirror site.

To guard against Trojan mirror sites, all files originating at the Linux Kernel Archives are [cryptographically signed](#). If you are getting a message that the verification key has expired, please see [this link](#).

What is Linux?

Linux is a clone of the operating system Unix, written from scratch by Linus Torvalds with assistance from a loosely-knit team of hackers across the Net. It aims towards POSIX and [Single UNIX Specification](#) compliance.

It has all the features you would expect in a modern fully-fledged Unix, including true multitasking, virtual memory, shared libraries, demand loading, shared copy-on-write executables, proper memory management, and multistack networking including IPv4 and IPv6.

Although originally developed first for 32-bit x86-based PCs (386 or higher), today Linux also runs on (at least) the [Alpha AXP](#), [Sun SPARC](#), [Motorola 68000](#), [PowerPC](#), [ARM](#), [Hitachi SuperH](#), [IBM S/390](#), [MIPS](#), [HP PA-RISC](#), [Intel IA-64](#), [AMD x86-64](#), [AXIS CRIS](#), [Renesas M32R](#), [Atmel AVR32](#), [Renesas H8/300](#), [NEC V850](#), [Tensilica Xtensa](#), and [Analog Devices Blackfin](#) architectures, for many of these architectures in both 32- and 64-bit variants.

Linux is easily portable to most general-purpose 32- or 64-bit architectures as long as they have a paged memory management unit (PMMU) and a port of the GNU C compiler (gcc) (part of [The GNU Compiler Collection, GCC](#)). Linux has also been ported to a number of architectures without a PMMU, although functionality is then obviously somewhat limited. See the [µClinux project](#) for more info.

The Linux Kernel Archives

ES Español (Español, internacional)

http://www.kernel.org/ RSS Google

Apple Yahoo! Google Maps YouTube Wikipedia Noticias (21) Populares

New to Linux?

If you're new to Linux, you don't want to download the kernel, which is just a component in a working Linux system. Instead, you want what is called a *distribution* of Linux, which is a complete Linux system. There are numerous distributions available for download on the Internet as well as for purchase from various vendors; some are general-purpose, and some are optimized for specific uses. We currently have mirrors of several distributions available at mirrors.kernel.org, as well as a small collection of special-purpose distributions at <http://www.kernel.org/pub/dist/>.

Note, however, that most distributions are very large (several gigabytes), so unless you have a [fast Internet link](#) you may want to save yourself some hassle and purchase a CD-ROM with a distribution; such CD-ROMs are available from a number of vendors.

[The Linux Installation HOWTO](#) has more information how to set up your first Linux system.

More Information

There is much [information about Linux on the web](#).

Reporting Linux Kernel bugs

Please see <http://www.kernel.org/pub/linux/docs/kml/reporting-bugs.html> if you want to report a Linux kernel bug. *Bug reports sent to the kernel.org administrators will be ignored.*

There is now a bugzilla setup at bugzilla.kernel.org. Currently this is for reporting kernel version 2.6 bugs only.

Mailing list

The Linux kernel is discussed on the *linux-kernel* mailing list at vger.kernel.org. The FAQ is available at <http://www.kernel.org/pub/linux/docs/kml/>, please read the FAQ before subscribing.

Although there is no official archive site, unofficial archives of the list can be found at:

- <http://marc.info/?l=linux-kernel>
- <http://www.ussg.iu.edu/hypermall/linux/kernel/>

Development trees

The Linux kernel, as well as several other pieces of software, are maintained using the [git](#) source code control system. For a more user-friendly wrapper around git, see [Cogito](#).

See <http://git.kernel.org/> for a list of git repositories on *kernel.org*.

The Linux Kernel Archives

ES Español (Español, internacional)

http://www.kernel.org/ RSS Google

Apple Yahoo! Google Maps YouTube Wikipedia Noticias (21) Populares

<http://www.ussg.iu.edu/hypermall/linux/kernel/>

Development trees

The Linux kernel, as well as several other pieces of software, are maintained using the [git](#) source code control system. For a more user-friendly wrapper around git, see [Cogito](#).

See <http://git.kernel.org/> for a list of git repositories on *kernel.org*.

Cryptographic Software

Due to U.S. Exports Regulations, all cryptographic software on this site is subject to the following legal notice:

This site includes publicly available encryption source code which, together with object code resulting from the compiling of publicly available source code, may be exported from the United States under License Exception "TSU" pursuant to 15 C.F.R. Section 740.13(e).

This legal notice applies to *cryptographic software only*. Please see the [Bureau of Industry and Security](#) for more information about current U.S. regulations.

Our servers are located in Corvallis, Oregon, USA; Palo Alto and San Francisco, California, USA; Amsterdam, Netherlands; and Umeå, Sweden. Use in violation of any applicable laws is prohibited.

Comments

Before emailing kernel.org, please take a look at our [kernel.org FAQ](#).

For comments about *the web pages*, please send mail to webmaster@kernel.org.

For comments about *the file archive*, please send mail to ftpadmin@kernel.org.

Please do not send general Linux questions or bug reports to these addresses. We do not have the resources to reply to them. If your message does not relate to the operation of the Linux Kernel Archives, it will be deleted without action. Because of the volume of unsolicited email we receive, please add [KORG] to the subject to avoid accidental deletion. Additionally, any Content-Type other than text/plain will most likely be deleted without action.

This site is operated by the Linux Kernel Organization, Inc., a [501\(c\)3 nonprofit corporation](#), with support from the following sponsors.

Hosted @ ISC

IBM

hp invent

THE LINUX FOUNDATION

OSL opensourcelab

Linux

W3C XHTML 1.0

Anexo B – Home Page da Comunidade Apache

Welcome! - The Apache Software Foundation

ES Español (España, internacional)

http://www.apache.org/ RSS Google

Apple Yahoo! Google Maps YouTube Wikipedia Noticias (21) Populares

Foundation Projects People Get Involved Support Apache Download ASF Blog

The Apache Software Foundation

Celebrating a Decade of Open Source Leadership.

The Apache Software Foundation provides support for the Apache community of open-source software projects. The **Apache projects** are characterized by a collaborative, consensus based development process, an open and pragmatic software license, and a desire to create high quality software that leads the way in its field.

We consider ourselves not simply a group of projects sharing a server, but rather a *community of developers and users.*

Latest News

If you would like to keep up with news and announcements from the foundation and all its projects, you can subscribe to the [Apache Announcements List](#).

Celebrating a Decade of Open Source Leadership

Exactly ten years ago, the highly dedicated members of the Apache Group decided to secure the future of the Apache HTTP Server by forming The Apache Software Foundation. [Read more](#) about this special anniversary in a letter by Jim Jagielski, current Chairman of the foundation.

ASF Foundation Blog

Tune in to all the latest coming out of Amsterdam, Blog posts daily recapping whats gone and what's coming up. Check out the new [Foundation Blog](#)

Free Live Video Streaming from ApacheCon Europe 2009

ApacheCon Europe 2009 will be held at the Movenpick Hotel, Amsterdam, between the 23rd and 27th March. The official user conference of the Apache Software Foundation features over a dozen training workshops from Industry Experts, over sixty one-hour sessions, free BarCampApache, free project MeetUps, and the Apache Hackathon! If you can't make it to Amsterdam, check out the [free live video streams](#) of all our keynotes

10 YEARS OF The ASF
Celebrating a Decade of Open Source Leadership
<http://www.apache.org>

LiveVideo STREAMING

ApacheCon Europe 2009
23-27 March
Amsterdam

Apache Projects

- HTTP Server
- Abdera
- ActiveMQ
- Ant
- APR
- Archiva
- Beehive
- Buildr
- Camel
- Cayenne
- Cocoon
- Commons
- Continuum
- CouchDB
- CXF
- DB
- Directory
- Excalibur
- Felix
- Forrest
- Geronimo
- Gump
- Hadoop

Foundation

- FAQ
- Licenses
- News
- Press Inquiries
- Public Records
- Sponsorship
- Donations
- Thanks
- Contact

Foundation Projects

- Conferences
- Infrastructure
- JCP
- Legal Affairs
- Public Relations
- Security
- Travel Assistance

How it works

- Introduction
- Meritocracy
- Structure

o Wednesday, 8:00 UTC: Jim Jagielski, *State of the Feather*

o Wednesday, 8:30 UTC: Raghu Ramakrishnan, *Data Management in the Cloud*

o Wednesday, 12:00 UTC: Lars Eilebrecht, *Behind the Scenes of The Apache Software Foundation*

o Thursday, 12:00 UTC: Jim Jagielski, *Sponsoring the ASF the Corporate and Individual Level*

o Thursday, 16:30 UTC: James Governor, *Open sourcing the analyst business...*

o Friday, 10:30 UTC: Apache Pioneer's Panel, *10 years of The Apache Software Foundation*

o Friday, 12:00 UTC: J Aaron Farr, *The Apache Way*

The following select technical tracks are available for a fee:

- Wednesday, 9:30-17:00 UTC: **Hadoop Geeks for Geeks track**
- Thursday, 8:00-16:00 UTC: **Tomcat for Developers and Administrators**
- Friday, 8:00-16:00 UTC: **HTTP Server Administration**

Press Conference for the Tenth Anniversary of The Apache Software Foundation

Globally Recognized Open Source Organization Celebrates a Decade of Leadership

The Apache Software Foundation (ASF) will hold a press conference to commemorate a decade of innovation in Open Source software and community development at ApacheCon Europe 2009, its official user conference. ASF representatives will also be available to answer the media's questions and explain the various aspects of ApacheCon; a photo opportunity (pool for broadcast media) will be arranged after the press conference. See the [full press release](#) for more details.

The Apache Software Foundation Names Qpid a Top-Level Project

Influential community-driven foundation promotes Open Source messaging implementation built on the AMQP standard

The Apache Software Foundation (ASF) today announced the graduation of the Qpid project from the Apache Incubator as a Top-Level Project (TLP), signifying that the Project's community and products have been well-governed under the ASF's meritocratic process and principles.

Apache Qpid is an Open Source messaging implementation built on the Advanced Messaging Queuing Protocol (AMQP) specification, the first open standard for enterprise messaging. Qpid provides transaction management, queuing, clustering, federation, security, management, and support across multiple operating systems and platforms. See the [full press release](#) for more details.

HiveMind

- HttpComponents
- iBATIS
- Incubator
- Jackrabbit
- Jakarta
- James
- Labs
- Lenya
- Logging
- Lucene
- Maven
- Mina
- MyFaces
- ODE
- OFBiz
- OpenEJB
- OpenJPA
- Perl
- POI
- Portals
- Qpid
- Roller
- Santuario
- ServiceMix
- Shale
- SpamAssassin
- STDCXX
- Struts
- Synapse
- Tapestry
- TCL
- Tiles
- Tomcat
- Turbine
- Tuscany
- Velocity
- Wicket
- Web Services
- Xalan
- Xerces

Collaboration

- Infrastructure
- Incubator
- Public Relations Committee
- Other entities
- Glossary
- Voting

Get Involved

- Mailing Lists
- Version Control
- Developer Info

Download

- from a mirror

Community

- Memorials

Related Sites

- ApacheCon
- Feathercast
- PlanetApache

Anexo C – Home Page da Comunidade MySQL

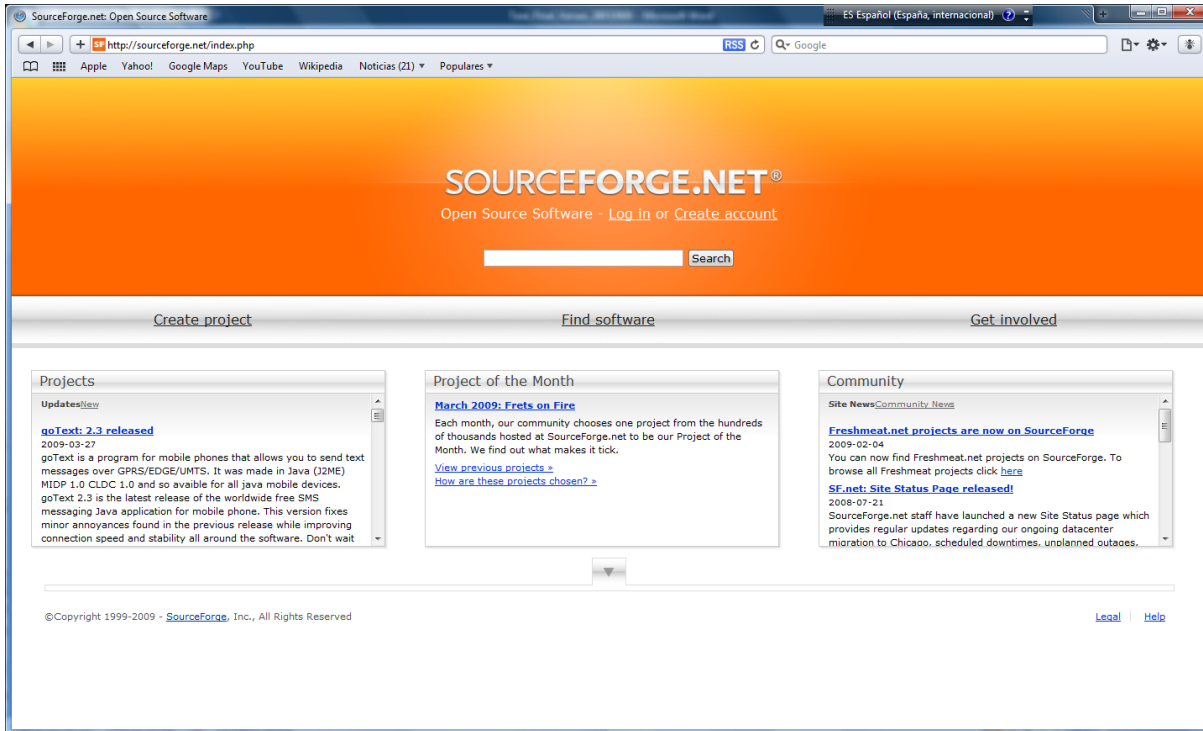
The screenshot shows the MySQL website home page in Spanish. The browser address bar displays 'http://www.mysql.com/'. The page features a navigation menu with categories like 'Products', 'Downloads', 'Documentation', 'Services', 'Customers', 'Why MySQL?', 'News & Events', and 'How to Buy'. A prominent banner at the top right encourages users to 'Test Drive the new MySQL Query Analyzer Free for 48 Hours!'. Below this, there are several columns of links and news items, including 'Discover', 'Test Drive', 'Learn', and 'What's New'. A 'MySQL Newsletter' subscription box is also visible. At the bottom, there are promotional banners for 'MySQL is Hiring!', 'MySQL Enterprise Unlimited MySQL Servers for one low price!', 'Open Source in Government', and 'REGISTER NOW!'. The footer contains copyright information and links to 'Online Shop', 'Site Map', 'About MySQL', 'Legal', 'Privacy Policy', 'Contact Us', and 'Job Opportunities'.

Anexo D – Home Page da Comunidade PHP

The screenshot displays the PHP website's home page, which is organized into several sections:

- What is PHP?**: A brief introduction to PHP as a widely-used general-purpose scripting language.
- Upcoming conferences**: Lists events like [php|tek 2009](#).
- Google Summer of Code 2009**: Announces PHP's involvement in the project, encouraging students to propose ideas.
- PHP 5.3.0RC1 Release Announcement**: Details the first release candidate of PHP 5.3.0, highlighting new features like namespaces, performance improvements, and various syntax additions.
- Stable Releases**: Shows the current stable version (5.2.9) and historical versions (4.4.9).
- Release Candidates**: Lists the current release candidate (5.3.0RC1).
- Upcoming Events**: A calendar of events for the month of March.
- Conferences**: Lists upcoming conferences such as OSSCamp Delhi March 2009.
- User Group Events**: Lists various user group meetings across different cities.
- Community**: A section for community news and updates.
- Syndication**: Information about news feeds.
- 5.2.9-1 (for Windows) released**: Announces a new Windows build of PHP 5.2.9, focusing on security fixes.
- PHP 5.2.9 Released!**: Announces the immediate availability of PHP 5.2.9, focusing on stability and security.
- Security Enhancements and Fixes in PHP 5.2.9**: Lists specific security fixes, such as issues with `imagecolorat()`, `zip` files, and `json_decode()`.

Anexo E – Home Page da Comunidade SourceForge



Anexo F - Discussões da Comunidade Linux

Interação 1.8.LNX

List: [linux-kernel](#)
Subject: [Re: Pull request: TASK KILLABLE](#)
From: [Matthew Wilcox <matthew \(\) wil ! cx>](#)
Date: [2008-01-31 23:45:05](#)
Message-ID: [20080131234504.GB25967 \(\) parisc-linux ! org](#)
[\[Download message RAW\]](#)

On Tue, Jan 29, 2008 at 10:41:24PM +1100, Linus Torvalds wrote:

> On Mon, 28 Jan 2008, Matthew Wilcox wrote:

> >

> > I'd like you to pull the task_killable branch of

> > git://git.kernel.org/pub/scm/linux/kernel/git/willy/misc.git

>

> Well, Andrew already pointed out some of this, but I do want more

> information. There's a "git request-pull" script that even comes with git

> and that generates a shortlog, a proper url+branch description, and a

> diffstat of the changes.

>

> So that is already much more descriptive, but in addition to that, if this

> isn't one of the "regular" pulls (which it obviously isn't), I really want

> a human description too so that I know what I'm pulling and what it's all

> about.

>

> I do *not* want to be in the position of having to fetch first, just to be

> able to look at what I'm fetching. By the time I pull a branch, I want to

> basically know (a) what I'm pulling and (b) why I want to pull it.

OK. Hopefully this satisfies you:

Introduce TASK_KILLABLE.

The basic idea was Linus', back in 2002. We can't normally return a short read because it breaks applications. But if a task has received a fatal signal, it doesn't have a chance to notice that it didn't get all the data it asked for.

To allow tasks to be interrupted by fatal signals, we introduce a new TASK_* bit; TASK_WAKEKILL. We also add a predicate fatal_signal_pending; the counterpart of signal_pending(). Then we add killable versions of lock_page(), mutex_lock(), schedule_timeout(), wait_event(), and wait_for_completion(). Finally, we can make the NFS 'intr' mount option a no-op.

At the end of this patch set, you can now kill a task in readdir(), read() and write(), at least most of the time. There is still much more work that can (and should) be done, such as implementing down_killable(),

lock_kernel_killable() and even spin_lock_killable().

While my main motivation has been to fix problems I have with flaky wireless networks and/or NFS servers, people like Ingo and Nick Piggin want to see this infrastructure for helping them improve the OOM killer.

The following changes since commit
8af03e782caele0a0f530ddd22301cdd12cf9dc0:

Linus Torvalds (1):
Merge branch 'for-2.6.25' of
git://git.kernel.org/.../paulus/powerpc

are available in the git repository at:

git://git.kernel.org/pub/scm/linux/kernel/git/willy/misc.git
task_killable

Liam R. Howlett (2):
Add mutex_lock_killable
Use mutex_lock_killable in vfs_readdir

Matthew Wilcox (20):
Use wake_up_locked() in eventpoll
Add macros to replace direct uses of TASK_ flags
perfmon: Use task_is_*
proc/array.c: Use TASK_REPORT
proc/base.c: Use task_is_*
wait: Use TASK_NORMAL
power: Use task_is_*
ptrace: Use task_is_*
sched: Use task_contributes_to_load, TASK_ALL and TASK_NORMAL
signal: Use task_is_*
exit: Use task_is_*
Add TASK_WAKEKILL
Add fatal_signal_pending
Add lock_page_killable
Use lock_page_killable
Add schedule_timeout_killable
Add wait_event_killable
Add wait_for_completion_killable
NFS: Switch from intr mount option to TASK_KILLABLE
Remove commented-out code copied from NFS

arch/ia64/kernel/perfmon.c		4 +-
fs/eventpoll.c		11 ++---
fs/nfs/client.c		6 +--
fs/nfs/direct.c		10 +----
fs/nfs/inode.c		6 +--
fs/nfs/mount_clnt.c		2 +-
fs/nfs/nfs3proc.c		7 +--
fs/nfs/nfs4proc.c		27 +++-----
fs/nfs/nfsroot.c		3 -
fs/nfs/pagelist.c		18 ++-----
fs/nfs/read.c		5 --
fs/nfs/super.c		4 --
fs/nfs/write.c		7 +---
fs/proc/array.c		7 +---
fs/proc/base.c		2 +-
fs/readdir.c		5 ++-
fs/smbfs/request.c		2 +-
include/linux/completion.h		1 +


```

include/linux/mutex.h      |    5 ++
include/linux/nfs_fs.h    |    9 +----
include/linux/nfs_mount.h |    2 +-
include/linux/pagemap.h   |   14 ++++++
include/linux/sched.h     |   36 ++++++
include/linux/sunrpc/clnt.h |    2 -
include/linux/sunrpc/sched.h |    2 -
include/linux/wait.h      |   52 ++++++
kernel/exit.c             |   88 ++++++
---
kernel/mutex.c            |   36 ++++++
kernel/power/process.c    |    6 +-
kernel/ptrace.c           |    8 +--
kernel/sched.c            |   28 ++++++
kernel/signal.c           |   19 ++++++
kernel/timer.c            |    7 +++
kernel/wait.c             |    2 +-
mm/filemap.c              |   25 ++++++
net/sunrpc/clnt.c         |   47 +-----
net/sunrpc/rpcb_clnt.c    |    3 +-
net/sunrpc/sched.c        |   15 +----
net/sunrpc/sunrpc_syms.c  |    2 -
39 files changed, 281 insertions(+), 254 deletions(-)

```

--

Intel are signing my paycheques ... these opinions are still mine
"Bill, look, we understand that you're interested in selling us this
operating system, but compare it to ours. We can't possibly take such
a retrograde step."

List: [linux-kernel](#)
Subject: [Re: Pull request: TASK KILLABLE](#)
From: [Linus Torvalds <torvalds \(\) linux-
foundation ! org>](#)
Date: [2008-02-01 0:48:44](#)
Message-ID: [alpine.LFD.1.00.0802011146180.3198 \(\) www !
l ! google ! com](#)
[\[Download message RAW\]](#)

On Thu, 31 Jan 2008, Matthew Wilcox wrote:

```

>
> To allow tasks to be interrupted by fatal signals, we introduce a new
> TASK_* bit; TASK_WAKEKILL. We also add a predicate fatal_signal_pending;
> the counterpart of signal_pending(). Then we add killable versions
> of lock_page(), mutex_lock(), schedule_timeout(), wait_event(), and
> wait_for_completion(). Finally, we can make the NFS 'intr' mount option
> a no-op.

```

Ok, I've pulled this. It conflicted with some NFS changes, but I did what
appeared to be the obvious conflict resolution, and it all at least
compiles and it looked sane to me. But I don't have NFS enabled normally,
and I certainly don't have a good way to test with my laptop while in
Melbourne, so Trond/Matthew, can you please double- and triple-check my
merge?

Linus

List: [linux-kernel](#)
Subject: [Re: Pull request: TASK KILLABLE](#)
From: [Trond Myklebust <Trond.Myklebust \(\) netapp ! com>](#)
Date: [2008-02-01 1:45:04](#)
Message-ID: [1201830304.3087.2.camel \(\) heimdal ! trondhjem ! org](#)
[\[Download message RAW\]](#)

On Fri, 2008-02-01 at 11:48 +1100, Linus Torvalds wrote:

```

>
> On Thu, 31 Jan 2008, Matthew Wilcox wrote:
> >
> > To allow tasks to be interrupted by fatal signals, we introduce a new
> > TASK_* bit; TASK_WAKEKILL. We also add a predicate
fatal_signal_pending;
> > the counterpart of signal_pending(). Then we add killable versions
> > of lock_page(), mutex_lock(), schedule_timeout(), wait_event(), and
> > wait_for_completion(). Finally, we can make the NFS 'intr' mount
option
> > a no-op.
>
> Ok, I've pulled this. It conflicted with some NFS changes, but I did what
> appeared to be the obvious conflict resolution, and it all at least
> compiles and it looked sane to me. But I don't have NFS enabled normally,
> and I certainly don't have a good way to test with my laptop while in
> Melbourne, so Trond/Matthew, can you please double- and triple-check my
> merge?

```

Hmm... The current code won't compile as a module. We're at least going to require something like the attached patch.

Cheers
Trond

[\["linux-2.6.24-000-export fatal signal pending.dif" \(linux-2.6.24-000-export fatal signal pending.dif\)\]](#)

From: Trond Myklebust <Trond.Myklebust@netapp.com>
Date: Thu, 31 Jan 2008 20:40:29 -0500
Subject: Ensure that we export __fatal_signal_pending()
Message-Id: <1201830304.3087.3.camel@heimdal.trondhjem.org>
Mime-Version: 1.0

It may be used by the modules nfs.ko and sunrpc.ko

Signed-off-by: Trond Myklebust <Trond.Myklebust@netapp.com>

```

kernel/signal.c |    1 +
1 files changed, 1 insertions(+), 0 deletions(-)

```

```

diff --git a/kernel/signal.c b/kernel/signal.c
index 8054dd4..1f2d4e1 100644
--- a/kernel/signal.c
+++ b/kernel/signal.c
@@ -998,6 +998,7 @@ int fastcall __fatal_signal_pending(struct task_struct
*tsk)
{

```

```

        return sigismember(&tsk->pending.signal, SIGKILL);
    }
+EXPORT_SYMBOL_GPL(__fatal_signal_pending);

/*
 * Must be called under rcu_read_lock() or with tasklist_lock read-held.

```

List: [linux-kernel](#)
Subject: [Re: Pull request: TASK KILLABLE](#)
From: [Linus Torvalds <torvalds \(\) linux-foundation ! org>](#)
Date: [2008-02-01 1:54:46](#)
Message-ID: [alpine.LFD.1.00.0802011252450.3198 \(\) www ! l ! google ! com](#)
[\[Download message RAW\]](#)

On Thu, 31 Jan 2008, Trond Myklebust wrote:

```

>
> Hmm... The current code won't compile as a module. We're at least going
> to require something like the attached patch.

```

Ahh, ok, I obviously only tested compiling it in.

No need to make that one-liner function be a GPL-only export, it's not like any of the other regular sigpending functions we export are GPL-only either (dequeue_signal() is, but that is because we *really* don't want people using that, and it's really about internal signal handling behavior).

Linus

List: [linux-kernel](#)
Subject: [Re: Pull request: TASK KILLABLE](#)
From: [Andrew Morton <akpm \(\) linux-foundation ! org>](#)
Date: [2008-02-01 2:02:05](#)
Message-ID: [20080131180205.20e7dece.akpm \(\) linux-foundation ! org](#)
[\[Download message RAW\]](#)

On Thu, 31 Jan 2008 20:45:04 -0500 Trond Myklebust <Trond.Myklebust@netapp.com> wrote:

```

>
> On Fri, 2008-02-01 at 11:48 +1100, Linus Torvalds wrote:
> >
> > On Thu, 31 Jan 2008, Matthew Wilcox wrote:
> > >
> > > To allow tasks to be interrupted by fatal signals, we introduce a new
> > > TASK_* bit; TASK_WAKEKILL. We also add a predicate
> > > fatal_signal_pending;
> > > the counterpart of signal_pending(). Then we add killable versions

```

```

> > > of lock_page(), mutex_lock(), schedule_timeout(), wait_event(), and
> > > wait_for_completion(). Finally, we can make the NFS 'intr' mount
option
> > > a no-op.
> >
> > Ok, I've pulled this. It conflicted with some NFS changes, but I did
what
> > appeared to be the obvious conflict resolution, and it all at least
> > compiles and it looked sane to me. But I don't have NFS enabled
normally,
> > and I certainly don't have a good way to test with my laptop while in
> > Melbourne, so Trond/Matthew, can you please double- and triple-check my
> > merge?
>
> Hmm... The current code won't compile as a module. We're at least going
> to require something like the attached patch.
>
> ...
> +EXPORT_SYMBOL_GPL(__fatal_signal_pending);
>

```

No such export was needed in the patches which I added to -mm. So something changed between then and now.

And going back through the mailing list all I can find is a series of five patches in October - it's unclear where and when the other 17 were reviewed, if they were.

Sorry, but I don't think this work has been suitably reviewed and tested.

List: [linux-kernel](#)
Subject: [Re: Pull request: TASK KILLABLE](#)
From: [Trond Myklebust <Trond.Myklebust \(\) netapp ! com>](#)
Date: [2008-02-01 2:03:53](#)
Message-ID: [1201831433.3087.14.camel \(\) heimdal ! trondhjem ! org](#)
[\[Download message RAW\]](#)

On Fri, 2008-02-01 at 12:54 +1100, Linus Torvalds wrote:

```

> No need to make that one-liner function be a GPL-only export, it's not
> like any of the other regular sigpending functions are we export are
> GPL-only either (dequeue_signal() is, but that is because we *really*
> don't want people using that, and it's really about internal signal
> handling behavior).

```

No. It was more in the spirit of "let's not export a function that others may implement trivially using established interfaces". However if you don't care, then here's a replacement.

Cheers
Trond

```

From: Trond Myklebust <Trond.Myklebust@netapp.com>
Date: Thu, 31 Jan 2008 20:40:29 -0500
Subject: Ensure that we export __fatal_signal_pending()
Message-Id: <1201831343.3087.12.camel@heimdal.trondhjem.org>
Mime-Version: 1.0

```

It may be used by the modules nfs.ko and sunrpc.ko

Signed-off-by: Trond Myklebust <Trond.Myklebust@netapp.com>

```
kernel/signal.c |    1 +
1 files changed, 1 insertions(+), 0 deletions(-)

diff --git a/kernel/signal.c b/kernel/signal.c
index 8054dd4..4333b6d 100644
--- a/kernel/signal.c
+++ b/kernel/signal.c
@@ -998,6 +998,7 @@ int fastcall __fatal_signal_pending(struct task_struct
*tsk)
{
    return sigismember(&tsk->pending.signal, SIGKILL);
}
+EXPORT_SYMBOL(__fatal_signal_pending);

/*
 * Must be called under rcu_read_lock() or with tasklist_lock read-held.
```

List: [linux-kernel](#)
Subject: [Re: Pull request: TASK KILLABLE](#)
From: [Matthew Wilcox <matthew \(\) wil ! cx>](#)
Date: [2008-02-06 2:19:42](#)
Message-ID: [20080206021942.GA21509 \(\) parisc-linux ! org](#)
[\[Download message RAW\]](#)

On Thu, Jan 31, 2008 at 06:02:05PM -0800, Andrew Morton wrote:
> No such export was needed in the patches which I added to -mm. So
> something changed between then and now.

Not sure about that problem -- still on holiday, so just checking ym
mail quickly.

> And going back through the mailing list all I can find is a series of
five
> patches in October - it's unclear where and when the other 17 were
> reviewed, if they were.

A large number of these patches are just a resplit of the patches sent
back in October -- you complained they weren't split up enough. So I
resplit them. And sent them to you. Asking if this was how you
preferred it. Which you didn't reply to.

--

Intel are signing my paycheques ... these opinions are still mine
"Bill, look, we understand that you're interested in selling us this
operating system, but compare it to ours. We can't possibly take such
a retrograde step."

List: [linux-kernel](#)
Subject: [Re: Pull request: TASK KILLABLE](#)

From: [Andrew Morton <akpm \(\) linux-foundation ! org>](mailto:akpm@linux-foundation.org)
Date: [2008-02-06 2:33:51](#)
Message-ID: [20080205183351.200d6480.akpm \(\) linux-foundation ! org](#)
[\[Download message RAW\]](#)

On Tue, 5 Feb 2008 19:19:42 -0700 Matthew Wilcox <matthew@wil.cx> wrote:

> > And going back through the mailing list all I can find is a series of
 > five
 > > patches in October - it's unclear where and when the other 17 were
 > > reviewed, if they were.
 >
 > A large number of these patches are just a resplit of the patches sent
 > back in October -- you complained they weren't split up enough. So I
 > resplit them. And sent them to you. Asking if this was how you
 > preferred it. Which you didn't reply to.

Well, I apologise if that's the case, but... I can find no record of the
 later patch series. Maybe an MTA ate them?

Interação 2.30.LNX

List: [linux-kernel](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) gumranet ! com>](mailto:andrea@gumranet.com)
Date: [2008-01-31 23:41:01](#)
Message-ID: [20080131234101.GS7185 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Thu, Jan 31, 2008 at 03:09:55PM -0800, Christoph Lameter wrote:

> On Thu, 31 Jan 2008, Christoph Lameter wrote:
 >
 > > > pagefault against the main linux page fault, given we already have
 > all
 > > > needed serialization out of the PT lock. XPMEM is forced to do that
 > >
 > > pt lock cannot serialize with invalidate_range since it is split. A
 > range
 > > requires locking for a series of ptes not only individual ones.
 >
 > Hmmm.. May be okay after all. I see that you are only doing it on the pte
 > level. This means the range callbacks are taking down a max of 512
 > entries. So you have a callback for each pmd. A callback for 2M of
 memory?

Exactly. The point of `_pages` is to reduce of an order of magnitude (512, or 1024 times) the number of needed `invalidate_page` calls in a few places where it's a straightforward optimization for both KVM and GRU. Thanks to the PT lock this remains a totally obviously safe design and it requires zero additional locking anywhere (nor linux VM, nor in the mmu notifier methods, nor in the KVM/GRU page fault).

Sure you can do `invalidate_range_start/end` for more than 2M(/4M on 32bit) max virtual ranges. But my approach that averages the fixed `mmu_lock` cost already over 512(/1024) ptes will make any larger "range" improvement not strongly measurable anymore given to do that you have to add locking as well and `_surely_` decrease the GRU scalability with tons of threads and tons of cpus potentially making GRU a lot slower `_especially_` on your numa systems.

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-01 1:37:21](#)
Message-ID: [Pine.LNX.4.64.0801311733140.24297 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Fri, 1 Feb 2008, Andrea Arcangeli wrote:

> I appreciate the review! I hope my entirely bug free and
 > straightforward #v5 will strongly increase the probability of getting
 > this in sooner than later. If something else it shows the approach I
 > prefer to cover GRU/KVM 100%, leaving the overkill mutex locking
 > requirements only to the mmu notifier users that can't deal with the
 > scalar and finegrined and already-taken/trashed PT lock.

Mutex locking? Could you be more specific?

I hope you will continue to do reviews of the other mmu notifier patchset?

List: [linux-kernel](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-01 1:37:21](#)
Message-ID: [Pine.LNX.4.64.0801311733140.24297 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Fri, 1 Feb 2008, Andrea Arcangeli wrote:

> I appreciate the review! I hope my entirely bug free and
 > straightforward #v5 will strongly increase the probability of getting
 > this in sooner than later. If something else it shows the approach I
 > prefer to cover GRU/KVM 100%, leaving the overkill mutex locking
 > requirements only to the mmu notifier users that can't deal with the
 > scalar and finegrined and already-taken/trashed PT lock.

Mutex locking? Could you be more specific?

I hope you will continue to do reviews of the other mmu notifier patchset?

List: [linux-mm](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-01 1:37:21](#)
Message-ID: [Pine.LNX.4.64.0801311733140.24297 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Fri, 1 Feb 2008, Andrea Arcangeli wrote:

> I appreciate the review! I hope my entirely bug free and
 > straightforward #v5 will strongly increase the probability of getting
 > this in sooner than later. If something else it shows the approach I
 > prefer to cover GRU/KVM 100%, leaving the overkill mutex locking
 > requirements only to the mmu notifier users that can't deal with the
 > scalar and finegrained and already-taken/trashed PT lock.

Mutex locking? Could you be more specific?

I hope you will continue to do reviews of the other mmu notifier patchset?

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-01 1:44:24](#)
Message-ID: [Pine.LNX.4.64.0801311738570.24297 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Fri, 1 Feb 2008, Andrea Arcangeli wrote:

> GRU. Thanks to the PT lock this remains a totally obviously safe
 > design and it requires zero additional locking anywhere (nor linux VM,
 > nor in the mmu notifier methods, nor in the KVM/GRU page fault).

Na. I would not be so sure about having caught all the issues yet...

> Sure you can do invalidate_range_start/end for more than 2M(/4M on
 > 32bit) max virtual ranges. But my approach that averages the fixed
 > mmu_lock cost already over 512(/1024) ptes will make any larger
 > "range" improvement not strongly measurable anymore given to do that
 > you have to add locking as well and surely decrease the GRU
 > scalability with tons of threads and tons of cpus potentially making
 > GRU a lot slower especially on your numa systems.

The trouble is that the invalidates are much more expensive if you have to send theses to remote partitions (XPmem). And its really great if you can simple tear down everything. Certainly this is a significant improvement over the earlier approach but you still have the invalidate_page calls in ptep_clear_flush. So they fire needlessly?

Serializing access in the device driver makes sense and comes with additional possiblity of not having to increment page counts all the time.

So you trade one cacheline dirtying for many that are necessary if you always increment the page count.

How does KVM insure the consistency of the shadow page tables? Atomic ops?

The GRU has no page table on its own. It populates TLB entries on demand using the linux page table. There is no way it can figure out when to drop page counts again. The invalidate calls are turned directly into tlb flushes.

List: [linux-mm](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-01 1:44:24](#)
Message-ID: [Pine.LNX.4.64.0801311738570.24297 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Fri, 1 Feb 2008, Andrea Arcangeli wrote:

> GRU. Thanks to the PT lock this remains a totally obviously safe
 > design and it requires zero additional locking anywhere (nor linux VM,
 > nor in the mmu notifier methods, nor in the KVM/GRU page fault).

Na. I would not be so sure about having caught all the issues yet...

> Sure you can do invalidate_range_start/end for more than 2M(/4M on
 > 32bit) max virtual ranges. But my approach that averages the fixed
 > mmu_lock cost already over 512(/1024) ptes will make any larger
 > "range" improvement not strongly measurable anymore given to do that
 > you have to add locking as well and surely decrease the GRU
 > scalability with tons of threads and tons of cpus potentially making
 > GRU a lot slower especially on your numa systems.

The trouble is that the invalidates are much more expensive if you have to send theses to remote partitions (XPmem). And its really great if you can simple tear down everything. Certainly this is a significant improvement over the earlier approach but you still have the invalidate_page calls in ptep_clear_flush. So they fire needlessly?

Serializing access in the device driver makes sense and comes with additional possibility of not having to increment page counts all the time. So you trade one cacheline dirtying for many that are necessary if you always increment the page count.

How does KVM insure the consistency of the shadow page tables? Atomic ops?

The GRU has no page table on its own. It populates TLB entries on demand using the linux page table. There is no way it can figure out when to drop page counts again. The invalidate calls are turned directly into tlb flushes.

List: [linux-kernel](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)

Date: [2008-02-01 1:44:24](#)
Message-ID: [Pine.LNX.4.64.0801311738570.24297 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Fri, 1 Feb 2008, Andrea Arcangeli wrote:

> GRU. Thanks to the PT lock this remains a totally obviously safe
 > design and it requires zero additional locking anywhere (nor linux VM,
 > nor in the mmu notifier methods, nor in the KVM/GRU page fault).

Na. I would not be so sure about having caught all the issues yet...

> Sure you can do invalidate_range_start/end for more than 2M(/4M on
 > 32bit) max virtual ranges. But my approach that averages the fixed
 > mmu_lock cost already over 512(/1024) ptes will make any larger
 > "range" improvement not strongly measurable anymore given to do that
 > you have to add locking as well and surely decrease the GRU
 > scalability with tons of threads and tons of cpus potentially making
 > GRU a lot slower especially on your numa systems.

The trouble is that the invalidates are much more expensive if you have to send theses to remote partitions (XPmem). And its really great if you can simple tear down everything. Certainly this is a significant improvement over the earlier approach but you still have the invalidate_page calls in ptep_clear_flush. So they fire needlessly?

Serializing access in the device driver makes sense and comes with additional possibility of not having to increment page counts all the time. So you trade one cacheline dirtying for many that are necessary if you always increment the page count.

How does KVM insure the consistency of the shadow page tables? Atomic ops?

The GRU has no page table on its own. It populates TLB entries on demand using the linux page table. There is no way it can figure out when to drop page counts again. The invalidate calls are turned directly into tlb flushes.

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Robin Holt <holt \(\) sgi ! com>](#)
Date: [2008-02-01 2:23:22](#)
Message-ID: [20080201022321.GZ26420 \(\) sgi ! com](#)
[\[Download message RAW\]](#)

On Thu, Jan 31, 2008 at 05:37:21PM -0800, Christoph Lameter wrote:

> On Fri, 1 Feb 2008, Andrea Arcangeli wrote:

>
 > > I appreciate the review! I hope my entirely bug free and
 > > strightforward #v5 will strongly increase the probability of getting
 > > this in sooner than later. If something else it shows the approach I
 > > prefer to cover GRU/KVM 100%, leaving the overkill mutex locking
 > > requirements only to the mmu notifier users that can't deal with the
 > > scalar and finegrined and already-taken/trashed PT lock.

>
 > Mutex locking? Could you be more specific?

I think he is talking about the external locking that xpmem will need to do to ensure we are not able to refault pages inside of regions that are undergoing recall/page table clearing. At least that has been my understanding to this point.

Thanks,
Robin

List: [linux-kernel](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Robin Holt <holt \(\) sgi ! com>](#)
Date: [2008-02-01 2:23:22](#)
Message-ID: [20080201022321.GZ26420 \(\) sgi ! com](#)
[\[Download message RAW\]](#)

On Thu, Jan 31, 2008 at 05:37:21PM -0800, Christoph Lameter wrote:

> On Fri, 1 Feb 2008, Andrea Arcangeli wrote:

>

> > I appreciate the review! I hope my entirely bug free and
 > > straightforward #v5 will strongly increase the probability of getting
 > > this in sooner than later. If something else it shows the approach I
 > > prefer to cover GRU/KVM 100%, leaving the overkill mutex locking
 > > requirements only to the mmu notifier users that can't deal with the
 > > scalar and finegrained and already-taken/trashed PT lock.

>

> Mutex locking? Could you be more specific?

I think he is talking about the external locking that xpmem will need to do to ensure we are not able to refault pages inside of regions that are undergoing recall/page table clearing. At least that has been my understanding to this point.

Thanks,
Robin

--

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-01 2:26:36](#)
Message-ID: [Pine.LNX.4.64.0801311824360.25839 \(\)](#)
[schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Thu, 31 Jan 2008, Robin Holt wrote:

> > Mutex locking? Could you be more specific?

>

> I think he is talking about the external locking that xpmem will need
 > to do to ensure we are not able to refault pages inside of regions that
 > are undergoing recall/page table clearing. At least that has been my
 > understanding to this point.

Right this has to be something like rw spinlock. Its needed for both GRU/XPmem. Not sure about KVM.

Take the read lock for invalidate operations. These can occur concurrently. (Or a simpler implementation for the GRU may just use a spinlock).

The write lock must be held for populate operations.

Lock can be refined as needed by the notifier driver. F.e. locking could be restricted to certain ranges.

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) qumranet ! com>](#)
Date: [2008-02-01 12:00:28](#)
Message-ID: [20080201120028.GW7185 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Thu, Jan 31, 2008 at 05:37:21PM -0800, Christoph Lameter wrote:

> On Fri, 1 Feb 2008, Andrea Arcangeli wrote:

>

> > I appreciate the review! I hope my entirely bug free and
 > > strightforward #v5 will strongly increase the probability of getting
 > > this in sooner than later. If something else it shows the approach I
 > > prefer to cover GRU/KVM 100%, leaving the overkill mutex locking
 > > requirements only to the mmu notifier users that can't deal with the
 > > scalar and finegrined and already-taken/trashed PT lock.

>

> Mutex locking? Could you be more specific?

Robin understanding was correct on this point. Also I think if you add start,end to range_end (like suggested a few times by me and once by Robin) I may get away without a lock thanks to the page pin. That's one strong reason why start,end are needed in range_end.

The only one that will definitely be forced to add a new lock around follow_page, in addition to the very _scalable_ PT lock, is GRU.

> I hope you will continue to do reviews of the other mmu notifier patchset?

Sure. I will also create a new KVM patch to plug on top of your versions (I already did for V2/V3 even if it never worked, but that might have a build problem, see kvm-devel for details). To be clear, as long as anything is merged that avoids me to use kprobes to make KVM swap work, I'm satisfied. I thought my approach had several advantages in simplicity, and being more obviously safe (in not relaying entirely on the page pin and creating a window of time where the linux pte writes to a page and the sptes writes to another one, remember populate_range), and it avoids introducing external locks in the GRU follow_page path. My approach was supposed to be in addition of the range_start/end needed by xpmem that can't possibly take advantage of the scalar PT lock and it definitely requires external lock to serialize xpmem fault against linux pagefault (not the case for GRU/KVM).

List: [linux-kernel](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)

From: [Andrea Arcangeli <andrea \(\) qumranet ! com>](mailto:andrea () qumranet ! com)
Date: [2008-02-01 12:00:28](#)
Message-ID: [20080201120028.GW7185 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Thu, Jan 31, 2008 at 05:37:21PM -0800, Christoph Lameter wrote:

> On Fri, 1 Feb 2008, Andrea Arcangeli wrote:

>

> > I appreciate the review! I hope my entirely bug free and
 > > straightforward #v5 will strongly increase the probability of getting
 > > this in sooner than later. If something else it shows the approach I
 > > prefer to cover GRU/KVM 100%, leaving the overkill mutex locking
 > > requirements only to the mmu notifier users that can't deal with the
 > > scalar and finegrained and already-taken/trashed PT lock.

>

> Mutex locking? Could you be more specific?

Robin understanding was correct on this point. Also I think if you add start,end to range_end (like suggested a few times by me and once by Robin) I may get away without a lock thanks to the page pin. That's one strong reason why start,end are needed in range_end.

The only one that will definitely be forced to add a new lock around follow_page, in addition to the very _scalable_ PT lock, is GRU.

> I hope you will continue to do reviews of the other mmu notifier patchset?

Sure. I will also create a new KVM patch to plug on top of your versions (I already did for V2/V3 even if it never worked, but that might have a build problem, see kvm-devel for details). To be clear, as long as anything is merged that avoids me to use kprobes to make KVM swap work, I'm satisfied. I thought my approach had several advantages in simplicity, and being more obviously safe (in not relaying entirely on the page pin and creating a window of time where the linux pte writes to a page and the sptes writes to another one, remember populate_range), and it avoids introducing external locks in the GRU follow_page path. My approach was supposed to be in addition of the range_start/end needed by xpmem that can't possibly take advantage of the scalar PT lock and it definitely requires external lock to serialize xpmem fault against linux pagefault (not the case for GRU/KVM).

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) qumranet ! com>](mailto:andrea () qumranet ! com)
Date: [2008-02-01 12:09:55](#)
Message-ID: [20080201120955.GX7185 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Thu, Jan 31, 2008 at 05:44:24PM -0800, Christoph Lameter wrote:

> The trouble is that the invalidates are much more expensive if you have to

> send theses to remote partitions (XPmem). And its really great if you can
 > simple tear down everything. Certainly this is a significant improvement
 > over the earlier approach but you still have the invalidate_page calls in
 > ptep_clear_flush. So they fire needlessly?

Dunno, they certainly fire more frequently than yours, even _pages fires more frequently than range_start,end but don't forget why!

That's because I've a different spinlock for every 512 ptes/4k-grub-tlbs that are being invalidated... So it pays off in scalability. I'm unsure if gru could play tricks with your patch, to still allow faults to still happen in parallel if they're on virtual addresses not in the same 2M naturally aligned chunk.

> Serializing access in the device driver makes sense and comes with
> additional possibility of not having to increment page counts all the
time.
> So you trade one cacheline dirtying for many that are necessary if you
> always increment the page count.

Note that my #v5 doesn't require to increase the page count all the time, so GRU will work fine with #v5.

See this comment in my patch:

```
/*
 * invalidate_page[s] is called in atomic context
 * after any pte has been updated and before
 * dropping the PT lock required to update any Linux pte.
 * Once the PT lock will be released the pte will have its
 * final value to export through the secondary MMU.
 * Before this is invoked any secondary MMU is still ok
 * to read/write to the page previously pointed by the
 * Linux pte because the old page hasn't been freed yet.
 * If required set_page_dirty has to be called internally
 * to this method.
 */
```

invalidate_page[s] is always called before the page is freed. This will require modifications to the tlb flushing code logic to take advantage of _pages in certain places. For now it's just safe.

> How does KVM insure the consistency of the shadow page tables? Atomic ops?

A per-VM mmu_lock spinlock is taken to serialize the access, plus atomic ops for the cpu.

> The GRU has no page table on its own. It populates TLB entries on demand
> using the linux page table. There is no way it can figure out when to
> drop page counts again. The invalidate calls are turned directly into tlb
> flushes.

Yes, this is why it can't serialize follow_page with only the PT lock with your patch. KVM may do it once you add start,end to range_end only thanks to the additional pin on the page.

List: [linux-mm](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) gumranet ! com>](#)
Date: [2008-02-01 12:09:55](#)
Message-ID: [20080201120955.GX7185 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Thu, Jan 31, 2008 at 05:44:24PM -0800, Christoph Lameter wrote:

> The trouble is that the invalidates are much more expensive if you have to
 > send these to remote partitions (XPmem). And its really great if you can
 > simple tear down everything. Certainly this is a significant improvement
 > over the earlier approach but you still have the invalidate_page calls in
 > ptep_clear_flush. So they fire needlessly?

Dunno, they certainly fire more frequently than yours, even _pages fires more frequently than range_start,end but don't forget why! That's because I've a different spinlock for every 512 ptes/4k-grub-tlbs that are being invalidated... So it pays off in scalability. I'm unsure if gru could play tricks with your patch, to still allow faults to still happen in parallel if they're on virtual addresses not in the same 2M naturally aligned chunk.

> Serializing access in the device driver makes sense and comes with
 > additional possiblity of not having to increment page counts all the time.
 > So you trade one cacheline dirtying for many that are necessary if you
 > always increment the page count.

Note that my #v5 doesn't require to increase the page count all the time, so GRU will work fine with #v5.

See this comment in my patch:

```
/*
 * invalidate_page[s] is called in atomic context
 * after any pte has been updated and before
 * dropping the PT lock required to update any Linux pte.
 * Once the PT lock will be released the pte will have its
 * final value to export through the secondary MMU.
 * Before this is invoked any secondary MMU is still ok
 * to read/write to the page previously pointed by the
 * Linux pte because the old page hasn't been freed yet.
 * If required set_page_dirty has to be called internally
 * to this method.
 */
```

invalidate_page[s] is always called before the page is freed. This will require modifications to the tlb flushing code logic to take advantage of _pages in certain places. For now it's just safe.

> How does KVM insure the consistency of the shadow page tables? Atomic ops?

A per-VM mmu_lock spinlock is taken to serialize the access, plus atomic ops for the cpu.

> The GRU has no page table on its own. It populates TLB entries on demand
 > using the linux page table. There is no way it can figure out when to
 > drop page counts again. The invalidate calls are turned directly into tlb
 > flushes.

Yes, this is why it can't serialize follow_page with only the PT lock with your patch. KVM may do it once you add start,end to range_end only thanks to the additional pin on the page.

List: [linux-kernel](#)

Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) gumranet ! com>](#)
Date: [2008-02-01 12:09:55](#)
Message-ID: [20080201120955.GX7185 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Thu, Jan 31, 2008 at 05:44:24PM -0800, Christoph Lameter wrote:
 > The trouble is that the invalidates are much more expensive if you have
 > to
 > send theses to remote partitions (XPmem). And its really great if you can
 > simple tear down everything. Certainly this is a significant improvement
 > over the earlier approach but you still have the invalidate_page calls in
 > ptep_clear_flush. So they fire needlessly?

Dunno, they certainly fire more frequently than yours, even _pages
 fires more frequently than range_start,end but don't forget why!
 That's because I've a different spinlock for every 512
 ptes/4k-grub-tlbs that are being invalidated... So it pays off in
 scalability. I'm unsure if gru could play tricks with your patch, to
 still allow faults to still happen in parallel if they're on virtual
 addresses not in the same 2M naturally aligned chunk.

> Serializing access in the device driver makes sense and comes with
 > additional possiblity of not having to increment page counts all the
 > time.
 > So you trade one cacheline dirtying for many that are necessary if you
 > always increment the page count.

Note that my #v5 doesn't require to increase the page count all the
 time, so GRU will work fine with #v5.

See this comment in my patch:

```
/*
 * invalidate_page[s] is called in atomic context
 * after any pte has been updated and before
 * dropping the PT lock required to update any Linux pte.
 * Once the PT lock will be released the pte will have its
 * final value to export through the secondary MMU.
 * Before this is invoked any secondary MMU is still ok
 * to read/write to the page previously pointed by the
 * Linux pte because the old page hasn't been freed yet.
 * If required set_page_dirty has to be called internally
 * to this method.
 */
```

invalidate_page[s] is always called before the page is freed. This
 will require modifications to the tlb flushing code logic to take
 advantage of _pages in certain places. For now it's just safe.

> How does KVM insure the consistency of the shadow page tables? Atomic
 ops?

A per-VM mmu_lock spinlock is taken to serialize the access, plus
 atomic ops for the cpu.

> The GRU has no page table on its own. It populates TLB entries on demand
 > using the linux page table. There is no way it can figure out when to
 > drop page counts again. The invalidate calls are turned directly into tlb

> flushes.

Yes, this is why it can't serialize `follow_page` with only the PT lock with your patch. KVM may do it once you add `start,end` to `range_end` only thanks to the additional pin on the page.

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-01 19:23:57](#)
Message-ID: [Pine.LNX.4.64.0802011118060.18163 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Fri, 1 Feb 2008, Andrea Arcangeli wrote:

> Note that my #v5 doesn't require to increase the page count all the
 > time, so GRU will work fine with #v5.

But that comes with the cost of firing `invalidate_page` for every page being evicted. In order to make your single `invalidate_range` work without it you need to hold a refcount on the page.

> `invalidate_page[s]` is always called before the page is freed. This
 > will require modifications to the tlb flushing code logic to take
 > advantage of `_pages` in certain places. For now it's just safe.

Yes so your `invalidate_range` is still some sort of dysfunctional optimization? Gazillions of `invalidate_page`'s will have to be executed when tearing down large memory areas.

> > How does KVM insure the consistency of the shadow page tables? Atomic ops?

>

> A per-VM `mmu_lock` spinlock is taken to serialize the access, plus
 > atomic ops for the cpu.

And that would not be enough to hold of new references? With small tweaks this should work with a common scheme. We could also redefine the role of `_start` and `_end` slightly to just require that the refs are removed when `_end` completes. That would allow the KVM page count ref to work as is now and would avoid the individual `invalidate_page()` callouts.

> > The GRU has no page table on its own. It populates TLB entries on demand

> > using the linux page table. There is no way it can figure out when to
 > > drop page counts again. The `invalidate` calls are turned directly into tlb

> > flushes.

>

> Yes, this is why it can't serialize `follow_page` with only the PT lock
 > with your patch. KVM may do it once you add `start,end` to `range_end`
 > only thanks to the additional pin on the page.

Right but that pin requires taking a refcount which we cannot do.

Frankly this looks as if this is a solution that would work only for KVM.

List: [linux-mm](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-01 19:23:57](#)
Message-ID: [Pine.LNX.4.64.0802011118060.18163 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Fri, 1 Feb 2008, Andrea Arcangeli wrote:

> Note that my #v5 doesn't require to increase the page count all the
> time, so GRU will work fine with #v5.

But that comes with the cost of firing `invalidate_page` for every page being evicted. In order to make your single `invalidate_range` work without it you need to hold a refcount on the page.

> `invalidate_page[s]` is always called before the page is freed. This
> will require modifications to the tlb flushing code logic to take
> advantage of `_pages` in certain places. For now it's just safe.

Yes so your `invalidate_range` is still some sort of dysfunctional optimization? Gazillions of `invalidate_page`'s will have to be executed when tearing down large memory areas.

> > How does KVM insure the consistency of the shadow page tables? Atomic
ops?
>
> A per-VM `mmu_lock` spinlock is taken to serialize the access, plus
> atomic ops for the cpu.

And that would not be enough to hold of new references? With small tweaks this should work with a common scheme. We could also redefine the role of `_start` and `_end` slightly to just require that the refs are removed when `_end` completes. That would allow the KVM page count ref to work as is now and would avoid the individual `invalidate_page()` callouts.

> > The GRU has no page table on its own. It populates TLB entries on
demand
> > using the linux page table. There is no way it can figure out when to
> > drop page counts again. The `invalidate` calls are turned directly into
tlb
> > flushes.
>
> Yes, this is why it can't serialize `follow_page` with only the PT lock
> with your patch. KVM may do it once you add `start,end` to `range_end`
> only thanks to the additional pin on the page.

Right but that pin requires taking a refcount which we cannot do.

Frankly this looks as if this is a solution that would work only for KVM.

List: [linux-kernel](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-01 19:23:57](#)

Message-ID: [Pine.LNX.4.64.0802011118060.18163 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Fri, 1 Feb 2008, Andrea Arcangeli wrote:

> Note that my #v5 doesn't require to increase the page count all the
 > time, so GRU will work fine with #v5.

But that comes with the cost of firing `invalidate_page` for every page being evicted. In order to make your single `invalidate_range` work without it you need to hold a refcount on the page.

> `invalidate_page[s]` is always called before the page is freed. This
 > will require modifications to the tlb flushing code logic to take
 > advantage of `_pages` in certain places. For now it's just safe.

Yes so your `invalidate_range` is still some sort of dysfunctional optimization? Gazillions of `invalidate_page`'s will have to be executed when tearing down large memory areas.

> > How does KVM insure the consistency of the shadow page tables? Atomic
 ops?
 >
 > A per-VM `mmu_lock` spinlock is taken to serialize the access, plus
 > atomic ops for the cpu.

And that would not be enough to hold of new references? With small tweaks this should work with a common scheme. We could also redefine the role of `_start` and `_end` slightly to just require that the refs are removed when `_end` completes. That would allow the KVM page count ref to work as is now and would avoid the individual `invalidate_page()` callouts.

> > The GRU has no page table on its own. It populates TLB entries on
 demand
 > > using the linux page table. There is no way it can figure out when to
 > > drop page counts again. The `invalidate` calls are turned directly into
 tlb
 > > flushes.
 >
 > Yes, this is why it can't serialize `follow_page` with only the PT lock
 > with your patch. KVM may do it once you add `start,end` to `range_end`
 > only thanks to the additional pin on the page.

Right but that pin requires taking a refcount which we cannot do.

Frankly this looks as if this is a solution that would work only for KVM.

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) qumranet ! com>](#)
Date: [2008-02-03 2:17:04](#)
Message-ID: [20080203021704.GC7185 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Fri, Feb 01, 2008 at 11:23:57AM -0800, Christoph Lameter wrote:

> Yes so your `invalidate_range` is still some sort of dysfunctional
 > optimization? Gazillions of `invalidate_page`'s will have to be executed

> when tearing down large memory areas.

I don't know if gru can flush the external TLB references with a single instruction like the cpu can do by overwriting %cr3. As far as vmx/svm are concerned, you got to do some fixed amount of work on the rmap structures and each spte, for each 4k invalidated regardless of page/pages/range/ranges. You can only share the cost of the lock and of the memslot lookup, so you lookup and take the lock once and you drop 512 sptes instead of just 1. Similar to what we do in the main linux VM by taking the PT lock for every 512 sptes.

So you worry about gazillions of invalidate_page without taking into account that even if you call invalidate_range_end(0, -1ULL) KVM will have to mangle over 4503599627370496 rmap entries anyway. Yes calling 1 invalidate_range_end instead of 4503599627370496 invalidate_page, will be faster, but not much faster. For KVM it remains an O(N) operation, where N is the number of pages. I'm not so sure we need to worry about my invalidate_pages being limited to 512 entries.

Perhaps GRU is different, I'm just describing the KVM situation here.

As far as KVM is concerned it's more sensible to be able to scale when there are 1024 kvm threads on 1024 cpus and each kvm-vcpu is accessing a different guest physical page (especially when new chunks of memory are allocated for the first time) that won't collide the whole time on naturally aligned 2M chunks of virtual addresses.

> And that would not be enough to hold of new references? With small tweaks
> this should work with a common scheme. We could also redefine the role
> of _start and _end slightly to just require that the refs are removed
when
> _end completes. That would allow the KVM page count ref to work as is now
> and would avoid the individual invalidate_page() callouts.

I can already only use _end and ignore _start, only remaining problem is this won't be 100% coherent (my patch is 100% coherent thanks to PT lock implicitly serializing follow_page/get_user_pages of the KVM/GRU secondary MMU faults). My approach give a better peace of mind with full scalability and no lock recursion when it's the KVM page fault that invokes get_user_pages that invokes the linux page fault routines that will try to execute _start taking the lock to block the page fault that is already running...

> > > The GRU has no page table on its own. It populates TLB entries on demand
> > > using the linux page table. There is no way it can figure out when to
> > > drop page counts again. The invalidate calls are turned directly into tlb
> > > flushes.
> >
> > Yes, this is why it can't serialize follow_page with only the PT lock
> > with your patch. KVM may do it once you add start,end to range_end
> > only thanks to the additional pin on the page.
>
> Right but that pin requires taking a refcount which we cannot do.

GRU can use my patch without the pin. XPMEM obviously can't use my patch as my invalidate_page[s] are under the PT lock (a feature to fit GRU/KVM in the simplest way), this is why an incremental patch adding invalidate_range_start/end would be required to support XPMEM too.

List: [linux-kernel](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) qumranet ! com>](#)
Date: [2008-02-03 2:17:04](#)
Message-ID: [20080203021704.GC7185 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Fri, Feb 01, 2008 at 11:23:57AM -0800, Christoph Lameter wrote:
 > Yes so your invalidate_range is still some sort of dysfunctional
 > optimization? Gazillions of invalidate_page's will have to be executed
 > when tearing down large memory areas.

I don't know if gru can flush the external TLB references with a single instruction like the cpu can do by overwriting %cr3. As far as vmx/svm are concerned, you got to do some fixed amount of work on the rmap structures and each spte, for each 4k invalidated regardless of page/pages/range/ranges. You can only share the cost of the lock and of the memslot lookup, so you lookup and take the lock once and you drop 512 sptes instead of just 1. Similar to what we do in the main linux VM by taking the PT lock for every 512 sptes.

So you worry about gazillions of invalidate_page without taking into account that even if you call invalidate_range_end(0, -1ULL) KVM will have to mangle over 4503599627370496 rmap entries anyway. Yes calling 1 invalidate_range_end instead of 4503599627370496 invalidate_page, will be faster, but not much faster. For KVM it remains an O(N) operation, where N is the number of pages. I'm not so sure we need to worry about my invalidate_pages being limited to 512 entries.

Perhaps GRU is different, I'm just describing the KVM situation here.

As far as KVM is concerned it's more sensible to be able to scale when there are 1024 kvm threads on 1024 cpus and each kvm-vcpu is accessing a different guest physical page (especially when new chunks of memory are allocated for the first time) that won't collide the whole time on naturally aligned 2M chunks of virtual addresses.

> And that would not be enough to hold of new references? With small tweaks
 > this should work with a common scheme. We could also redefine the role
 > of _start and _end slightly to just require that the refs are removed
 > when
 > _end completes. That would allow the KVM page count ref to work as is now
 > and would avoid the individual invalidate_page() callouts.

I can already only use _end and ignore _start, only remaining problem is this won't be 100% coherent (my patch is 100% coherent thanks to PT lock implicitly serializing follow_page/get_user_pages of the KVM/GRU secondary MMU faults). My approach give a better peace of mind with full scalability and no lock recursion when it's the KVM page fault that invokes get_user_pages that invokes the linux page fault routines that will try to execute _start taking the lock to block the page fault that is already running...

> > > The GRU has no page table on its own. It populates TLB entries on demand
 > > > using the linux page table. There is no way it can figure out when to
 > > > drop page counts again. The invalidate calls are turned directly into tlb
 > > > flushes.
 > >

> > Yes, this is why it can't serialize follow_page with only the PT lock
 > > with your patch. KVM may do it once you add start,end to range_end
 > > only thanks to the additional pin on the page.
 >
 > Right but that pin requires taking a refcount which we cannot do.

GRU can use my patch without the pin. XPMEM obviously can't use my patch as my invalidate_page[s] are under the PT lock (a feature to fit GRU/KVM in the simplest way), this is why an incremental patch adding invalidate_range_start/end would be required to support XPMEM too.

List: [linux-mm](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) gumranet ! com>](#)
Date: [2008-02-03 2:17:04](#)
Message-ID: [20080203021704.GC7185 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Fri, Feb 01, 2008 at 11:23:57AM -0800, Christoph Lameter wrote:
 > Yes so your invalidate_range is still some sort of dysfunctional
 > optimization? Gazillions of invalidate_page's will have to be executed
 > when tearing down large memory areas.

I don't know if gru can flush the external TLB references with a single instruction like the cpu can do by overwriting %cr3. As far as vmx/svm are concerned, you got to do some fixed amount of work on the rmap structures and each spte, for each 4k invalidated regardless of page/pages/range/ranges. You can only share the cost of the lock and of the memslot lookup, so you lookup and take the lock once and you drop 512 sptes instead of just 1. Similar to what we do in the main linux VM by taking the PT lock for every 512 sptes.

So you worry about gazillions of invalidate_page without taking into account that even if you call invalidate_range_end(0, -1ULL) KVM will have to mangle over 4503599627370496 rmap entries anyway. Yes calling 1 invalidate_range_end instead of 4503599627370496 invalidate_page, will be faster, but not much faster. For KVM it remains an O(N) operation, where N is the number of pages. I'm not so sure we need to worry about my invalidate_pages being limited to 512 entries.

Perhaps GRU is different, I'm just describing the KVM situation here.

As far as KVM is concerned it's more sensible to be able to scale when there are 1024 kvm threads on 1024 cpus and each kvm-vcpu is accessing a different guest physical page (especially when new chunks of memory are allocated for the first time) that won't collide the whole time on naturally aligned 2M chunks of virtual addresses.

> And that would not be enough to hold of new references? With small tweaks
 > this should work with a common scheme. We could also redefine the role
 > of _start and _end slightly to just require that the refs are removed
 > when
 > _end completes. That would allow the KVM page count ref to work as is now
 > and would avoid the individual invalidate_page() callouts.

I can already only use _end and ignore _start, only remaining problem is this won't be 100% coherent (my patch is 100% coherent thanks to PT lock implicitly serializing follow_page/get_user_pages of the KVM/GRU secondary MMU faults). My approach give a better peace of mind with

full scalability and no lock recursion when it's the KVM page fault that invokes `get_user_pages` that invokes the linux page fault routines that will try to `execute_start` taking the lock to block the page fault that is already running...

```
> > > The GRU has no page table on its own. It populates TLB entries on
demand
> > > using the linux page table. There is no way it can figure out when to
> > > drop page counts again. The invalidate calls are turned directly into
tlb
> > > flushes.
> >
> > Yes, this is why it can't serialize follow_page with only the PT lock
> > with your patch. KVM may do it once you add start,end to range_end
> > only thanks to the additional pin on the page.
>
> Right but that pin requires taking a refcount which we cannot do.
```

GRU can use my patch without the pin. XPMEM obviously can't use my patch as my `invalidate_page[s]` are under the PT lock (a feature to fit GRU/KVM in the simplest way), this is why an incremental patch adding `invalidate_range_start/end` would be required to support XPMEM too.

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Jack Steiner <steiner \(\) sgi ! com>](#)
Date: [2008-02-03 3:14:57](#)
Message-ID: [20080203031457.GA16127 \(\) sgi ! com](#)
[\[Download message RAW\]](#)

```
On Sun, Feb 03, 2008 at 03:17:04AM +0100, Andrea Arcangeli wrote:
> On Fri, Feb 01, 2008 at 11:23:57AM -0800, Christoph Lameter wrote:
> > Yes so your invalidate_range is still some sort of dysfunctional
> > optimization? Gazillions of invalidate_page's will have to be executed
> > when tearing down large memory areas.
>
> I don't know if gru can flush the external TLB references with a
> single instruction like the cpu can do by overwriting %cr3. As far as
```

The mechanism obviously is a little different, but the GRU can flush an external TLB in a single read/write/flush of a cacheline (should take a few 100 nsec). Typically, an application uses a single GRU. Threaded applications, however, could use one GRU per thread, so it is possible that multiple TLBs must be flushed. In some cases, the external flush can be avoided if the GRU is not currently in use by the thread.

Also, most (but not all) applications that use the GRU do not usually do anything that requires frequent flushing (fortunately). The GRU is intended for HPC-like applications. These don't usually do frequent map/unmap operations or anything else that requires a lot of flushes.

I expect that KVM is a lot different.

I have most of the GRU code working with the latest `mmuops` patch. I still have a list of loose ends that I'll get to next week. The most important is the exact handling of the range invalidates. The code that I currently have works (mostly) but has a few endcases that will cause problems. Once I finish, I'll be glad to send you snippets of the code (or all of it) if you would like to take a look.

```

> vmx/svm are concerned, you got to do some fixed amount of work on the
> rmap structures and each spte, for each 4k invalidated regardless of
> page/pages/range/ranges. You can only share the cost of the lock and
> of the memslot lookup, so you lookup and take the lock once and you
> drop 512 sptes instead of just 1. Similar to what we do in the main
> linux VM by taking the PT lock for every 512 sptes.
>
> So you worry about gazillions of invalidate_page without taking into
> account that even if you call invalidate_range_end(0, -1ULL) KVM will
> have to mangle over 4503599627370496 rmap entries anyway. Yes calling
> 1 invalidate_range_end instead of 4503599627370496 invalidate_page,
> will be faster, but not much faster. For KVM it remains an O(N)
> operation, where N is the number of pages. I'm not so sure we need to
> worry about my invalidate_pages being limited to 512 entries.
>
> Perhaps GRU is different, I'm just describing the KVM situation here.
>
> As far as KVM is concerned it's more sensible to be able to scale when
> there are 1024 kvm threads on 1024 cpus and each kvm-vcpu is accessing
> a different guest physical page (especially when new chunks of memory
> are allocated for the first time) that won't collide the whole time on
> naturally aligned 2M chunks of virtual addresses.
>
> > And that would not be enough to hold of new references? With small
tweaks
> > this should work with a common scheme. We could also redefine the role
> > of _start and _end slightly to just require that the refs are removed
when
> > _end completes. That would allow the KVM page count ref to work as is
now
> > and would avoid the individual invalidate_page() callouts.
>
> I can already only use _end and ignore _start, only remaining problem
> is this won't be 100% coherent (my patch is 100% coherent thanks to PT
> lock implicitly serializing follow_page/get_user_pages of the KVM/GRU
> secondary MMU faults). My approach give a better peace of mind with
> full scalability and no lock recursion when it's the KVM page fault
> that invokes get_user_pages that invokes the linux page fault routines
> that will try to execute _start taking the lock to block the page
> fault that is already running...
>
> > > The GRU has no page table on its own. It populates TLB entries on
demand
> > > using the linux page table. There is no way it can figure out when
to
> > > drop page counts again. The invalidate calls are turned directly
into tlb
> > > flushes.
> > >
> > > Yes, this is why it can't serialize follow_page with only the PT lock
> > > with your patch. KVM may do it once you add start,end to range_end
> > > only thanks to the additional pin on the page.
> >
> > Right but that pin requires taking a refcount which we cannot do.
>
> GRU can use my patch without the pin. XPMEM obviously can't use my
> patch as my invalidate_page[s] are under the PT lock (a feature to fit
> GRU/KVM in the simplest way), this is why an incremental patch adding
> invalidate_range_start/end would be required to support XPMEM too.

```

```

--- jack

```

List: [linux-mm](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Jack Steiner <steiner \(\) sgi ! com>](#)
Date: [2008-02-03 3:14:57](#)
Message-ID: [20080203031457.GA16127 \(\) sgi ! com](#)
[\[Download message RAW\]](#)

On Sun, Feb 03, 2008 at 03:17:04AM +0100, Andrea Arcangeli wrote:
 > On Fri, Feb 01, 2008 at 11:23:57AM -0800, Christoph Lameter wrote:
 > > Yes so your invalidate_range is still some sort of dysfunctional
 > > optimization? Gazillions of invalidate_page's will have to be executed
 > > when tearing down large memory areas.
 >
 > I don't know if gru can flush the external TLB references with a
 > single instruction like the cpu can do by overwriting %cr3. As far as

The mechanism obviously is a little different, but the GRU can flush an external TLB in a single read/write/flush of a cacheline (should take a few 100 nsec). Typically, an application uses a single GRU. Threaded applications, however, could use one GRU per thread, so it is possible that multiple TLBs must be flushed. In some cases, the external flush can be avoided if the GRU is not currently in use by the thread.

Also, most (but not all) applications that use the GRU do not usually do anything that requires frequent flushing (fortunately). The GRU is intended for HPC-like applications. These don't usually do frequent map/unmap operations or anything else that requires a lot of flushes.

I expect that KVM is a lot different.

I have most of the GRU code working with the latest mmuops patch. I still have a list of loose ends that I'll get to next week. The most important is the exact handling of the range invalidates. The code that I currently have works (mostly) but has a few endcases that will cause problems. Once I finish, I'll be glad to send you snippets of the code (or all of it) if you would like to take a look.

> vmx/svm are concerned, you got to do some fixed amount of work on the
 > rmap structures and each spte, for each 4k invalidated regardless of
 > page/pages/range/ranges. You can only share the cost of the lock and
 > of the memslot lookup, so you lookup and take the lock once and you
 > drop 512 sptes instead of just 1. Similar to what we do in the main
 > linux VM by taking the PT lock for every 512 sptes.
 >
 > So you worry about gazillions of invalidate_page without taking into
 > account that even if you call invalidate_range_end(0, -1ULL) KVM will
 > have to mangle over 4503599627370496 rmap entries anyway. Yes calling
 > 1 invalidate_range_end instead of 4503599627370496 invalidate_page,
 > will be faster, but not much faster. For KVM it remains an O(N)
 > operation, where N is the number of pages. I'm not so sure we need to
 > worry about my invalidate_pages being limited to 512 entries.
 >
 > Perhaps GRU is different, I'm just describing the KVM situation here.
 >
 > As far as KVM is concerned it's more sensible to be able to scale when
 > there are 1024 kvm threads on 1024 cpus and each kvm-vcpu is accessing
 > a different guest physical page (especially when new chunks of memory
 > are allocated for the first time) that won't collide the whole time on
 > naturally aligned 2M chunks of virtual addresses.


```

>
> > And that would not be enough to hold of new references? With small
tweaks
> > this should work with a common scheme. We could also redefine the role
> > of _start and _end slightly to just require that the refs are removed
when
> > _end completes. That would allow the KVM page count ref to work as is
now
> > and would avoid the individual invalidate_page() callouts.
>
> I can already only use _end and ignore _start, only remaining problem
> is this won't be 100% coherent (my patch is 100% coherent thanks to PT
> lock implicitly serializing follow_page/get_user_pages of the KVM/GRU
> secondary MMU faults). My approach give a better peace of mind with
> full scalability and no lock recursion when it's the KVM page fault
> that invokes get_user_pages that invokes the linux page fault routines
> that will try to execute _start taking the lock to block the page
> fault that is already running...
>
> > > The GRU has no page table on its own. It populates TLB entries on
demand
> > > using the linux page table. There is no way it can figure out when
to
> > > drop page counts again. The invalidate calls are turned directly
into tlb
> > > flushes.
> > >
> > > Yes, this is why it can't serialize follow_page with only the PT lock
> > > with your patch. KVM may do it once you add start,end to range_end
> > > only thanks to the additional pin on the page.
> >
> > Right but that pin requires taking a refcount which we cannot do.
>
> GRU can use my patch without the pin. XPMEM obviously can't use my
> patch as my invalidate_page[s] are under the PT lock (a feature to fit
> GRU/KVM in the simplest way), this is why an incremental patch adding
> invalidate_range_start/end would be required to support XPMEM too.

--- jack

```

List: [linux-kernel](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Jack Steiner <steiner \(\) sgi ! com>](#)
Date: [2008-02-03 3:14:57](#)
Message-ID: [20080203031457.GA16127 \(\) sgi ! com](#)
[\[Download message RAW\]](#)

```

On Sun, Feb 03, 2008 at 03:17:04AM +0100, Andrea Arcangeli wrote:
> On Fri, Feb 01, 2008 at 11:23:57AM -0800, Christoph Lameter wrote:
> > Yes so your invalidate_range is still some sort of dysfunctional
> > optimization? Gazillions of invalidate_page's will have to be executed
> > when tearing down large memory areas.
>

```

```

> I don't know if gru can flush the external TLB references with a
> single instruction like the cpu can do by overwriting %cr3. As far as

```

The mechanism obviously is a little different, but the GRU can flush an external TLB in a single read/write/flush of a cacheline (should take a few

100 nsec). Typically, an application uses a single GRU. Threaded applications, however, could use one GRU per thread, so it is possible that multiple TLBs must be flushed. In some cases, the external flush can be avoided if the GRU is not currently in use by the thread.

Also, most (but not all) applications that use the GRU do not usually do anything that requires frequent flushing (fortunately). The GRU is intended for HPC-like applications. These don't usually do frequent map/unmap operations or anything else that requires a lot of flushes.

I expect that KVM is a lot different.

I have most of the GRU code working with the latest mmuops patch. I still have a list of loose ends that I'll get to next week. The most important is the exact handling of the range invalidates. The code that I currently have works (mostly) but has a few endcases that will cause problems. Once I finish, I'll be glad to send you snippets of the code (or all of it) if you would like to take a look.

```
> vmx/svm are concerned, you got to do some fixed amount of work on the
> rmap structures and each spte, for each 4k invalidated regardless of
> page/pages/range/ranges. You can only share the cost of the lock and
> of the memslot lookup, so you lookup and take the lock once and you
> drop 512 sptes instead of just 1. Similar to what we do in the main
> linux VM by taking the PT lock for every 512 sptes.
>
> So you worry about gazillions of invalidate_page without taking into
> account that even if you call invalidate_range_end(0, -1ULL) KVM will
> have to mangle over 4503599627370496 rmap entries anyway. Yes calling
> 1 invalidate_range_end instead of 4503599627370496 invalidate_page,
> will be faster, but not much faster. For KVM it remains an O(N)
> operation, where N is the number of pages. I'm not so sure we need to
> worry about my invalidate_pages being limited to 512 entries.
>
> Perhaps GRU is different, I'm just describing the KVM situation here.
>
> As far as KVM is concerned it's more sensible to be able to scale when
> there are 1024 kvm threads on 1024 cpus and each kvm-vcpu is accessing
> a different guest physical page (especially when new chunks of memory
> are allocated for the first time) that won't collide the whole time on
> naturally aligned 2M chunks of virtual addresses.
>
> > And that would not be enough to hold of new references? With small
tweaks
> > this should work with a common scheme. We could also redefine the role
> > of _start and _end slightly to just require that the refs are removed
when
> > _end completes. That would allow the KVM page count ref to work as is
now
> > and would avoid the individual invalidate_page() callouts.
>
> I can already only use _end and ignore _start, only remaining problem
> is this won't be 100% coherent (my patch is 100% coherent thanks to PT
> lock implicitly serializing follow_page/get_user_pages of the KVM/GRU
> secondary MMU faults). My approach give a better peace of mind with
> full scalability and no lock recursion when it's the KVM page fault
> that invokes get_user_pages that invokes the linux page fault routines
> that will try to execute _start taking the lock to block the page
> fault that is already running...
>
```

> > > The GRU has no page table on its own. It populates TLB entries on demand
 > > > using the linux page table. There is no way it can figure out when to
 > > > drop page counts again. The invalidate calls are turned directly into tlb
 > > > flushes.
 > > >
 > > > Yes, this is why it can't serialize follow_page with only the PT lock
 > > > with your patch. KVM may do it once you add start,end to range_end
 > > > only thanks to the additional pin on the page.
 > >
 > > Right but that pin requires taking a refcount which we cannot do.
 >
 > GRU can use my patch without the pin. XPMEM obviously can't use my
 > patch as my invalidate_page[s] are under the PT lock (a feature to fit
 > GRU/KVM in the simplest way), this is why an incremental patch adding
 > invalidate_range_start/end would be required to support XPMEM too.

--- jack

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) qumranet ! com>](#)
Date: [2008-02-03 3:33:18](#)
Message-ID: [20080203033318.GE7185 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Sat, Feb 02, 2008 at 09:14:57PM -0600, Jack Steiner wrote:
 > Also, most (but not all) applications that use the GRU do not usually do
 > anything that requires frequent flushing (fortunately). The GRU is
 > intended
 > for HPC-like applications. These don't usually do frequent map/unmap
 > operations or anything else that requires a lot of flushes.
 >
 > I expect that KVM is a lot different.

I don't think so. invalidate_page/pages/range_start,end is a slow and
 unfrequent path for KVM (or alternatively the ranges are very small in
 which case _range_start/end won't payoff compared to _pages). Whenever
 invalidate_page[s] become a fast path, we're generally I/O
 bound. get_user_pages is always the fast path instead. I thought it
 was much more important that get_user_pages scale as well as it does
 now and that the KVM page fault isn't serialized with a mutex, than
 whatever invalidate side optimization. get_user_pages may run
 frequently from all vcpus even if there are no invalidates and no
 memory pressure and I don't mean only during startup.

> I have most of the GRU code working with the latest mmuops patch. I still
 > have a list of loose ends that I'll get to next week. The most important
 > is
 > the exact handling of the range invalidates. The code that I currently
 > have
 > works (mostly) but has a few endcases that will cause problems. Once I
 > finish, I'll be glad to send you snippets of the code (or all of it) if
 > you
 > would like to take a look.

Sure.

List: [linux-kernel](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) gumranet ! com>](#)
Date: [2008-02-03 3:33:18](#)
Message-ID: [20080203033318.GE7185 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Sat, Feb 02, 2008 at 09:14:57PM -0600, Jack Steiner wrote:
 > Also, most (but not all) applications that use the GRU do not usually do
 > anything that requires frequent flushing (fortunately). The GRU is
 intended
 > for HPC-like applications. These don't usually do frequent map/unmap
 > operations or anything else that requires a lot of flushes.
 >
 > I expect that KVM is a lot different.

I don't think so. `invalidate_page/pages/range_start,end` is a slow and
 unfrequent path for KVM (or alternatively the ranges are very small in
 which case `_range_start/end` won't payoff compared to `_pages`). Whenever
`invalidate_page[s]` become a fast path, we're generally I/O
 bound. `get_user_pages` is always the fast path instead. I thought it
 was much more important that `get_user_pages` scale as well as it does
 now and that the KVM page fault isn't serialized with a mutex, than
 whatever `invalidate` side optimization. `get_user_pages` may run
 frequently from all vcpus even if there are no invalidates and no
 memory pressure and I don't mean only during startup.

> I have most of the GRU code working with the latest mmuops patch. I still
 > have a list of loose ends that I'll get to next week. The most important
 is
 > the exact handling of the range invalidates. The code that I currently
 have
 > works (mostly) but has a few endcases that will cause problems. Once I
 > finish, I'll be glad to send you snippets of the code (or all of it) if
 you
 > would like to take a look.

Sure.

List: [linux-mm](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) gumranet ! com>](#)
Date: [2008-02-03 3:33:18](#)
Message-ID: [20080203033318.GE7185 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Sat, Feb 02, 2008 at 09:14:57PM -0600, Jack Steiner wrote:
 > Also, most (but not all) applications that use the GRU do not usually do
 > anything that requires frequent flushing (fortunately). The GRU is
 intended
 > for HPC-like applications. These don't usually do frequent map/unmap
 > operations or anything else that requires a lot of flushes.
 >
 > I expect that KVM is a lot different.

I don't think so. `invalidate_page/pages/range_start,end` is a slow and unfrequent path for KVM (or alternatively the ranges are very small in which case `_range_start/end` won't payoff compared to `_pages`). Whenever `invalidate_page[s]` become a fast path, we're generally I/O bound. `get_user_pages` is always the fast path instead. I thought it was much more important that `get_user_pages` scale as well as it does now and that the KVM page fault isn't serialized with a mutex, than whatever `invalidate` side optimization. `get_user_pages` may run frequently from all vcpus even if there are no invalidates and no memory pressure and I don't mean only during startup.

> I have most of the GRU code working with the latest mmuops patch. I still
> have a list of loose ends that I'll get to next week. The most important
is
> the exact handling of the range invalidates. The code that I currently
have
> works (mostly) but has a few endcases that will cause problems. Once I
> finish, I'll be glad to send you snippets of the code (or all of it) if
you
> would like to take a look.

Sure.

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-04 19:09:01](#)
Message-ID: [Pine.LNX.4.64.0802041106370.9656 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Sun, 3 Feb 2008, Andrea Arcangeli wrote:

> > Right but that pin requires taking a refcount which we cannot do.
>
> GRU can use my patch without the pin. XPMEM obviously can't use my
> patch as my `invalidate_page[s]` are under the PT lock (a feature to fit
> GRU/KVM in the simplest way), this is why an incremental patch adding
> `invalidate_range_start/end` would be required to support XPMEM too.

Doesnt the kernel in some situations release the page before releasing the pte lock? Then there will be an external pte pointing to a page that may now have a different use. Its really bad if that pte does allow writes.

List: [linux-mm](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-04 19:09:01](#)
Message-ID: [Pine.LNX.4.64.0802041106370.9656 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Sun, 3 Feb 2008, Andrea Arcangeli wrote:

> > Right but that pin requires taking a refcount which we cannot do.
>
> GRU can use my patch without the pin. XPMEM obviously can't use my

> patch as my invalidate_page[s] are under the PT lock (a feature to fit
> GRU/KVM in the simplest way), this is why an incremental patch adding
> invalidate_range_start/end would be required to support XPMEM too.

Doesnt the kernel in some situations release the page before releasing the
pte lock? Then there will be an external pte pointing to a page that may
now have a different use. Its really bad if that pte does allow writes.

List: [linux-kernel](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-04 19:09:01](#)
Message-ID: [Pine.LNX.4.64.0802041106370.9656 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Sun, 3 Feb 2008, Andrea Arcangeli wrote:

> > Right but that pin requires taking a refcount which we cannot do.
>
> GRU can use my patch without the pin. XPMEM obviously can't use my
> patch as my invalidate_page[s] are under the PT lock (a feature to fit
> GRU/KVM in the simplest way), this is why an incremental patch adding
> invalidate_range_start/end would be required to support XPMEM too.

Doesnt the kernel in some situations release the page before releasing the
pte lock? Then there will be an external pte pointing to a page that may
now have a different use. Its really bad if that pte does allow writes.

--

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) qumranet ! com>](#)
Date: [2008-02-05 5:25:25](#)
Message-ID: [20080205052525.GD7441 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Mon, Feb 04, 2008 at 11:09:01AM -0800, Christoph Lameter wrote:

> On Sun, 3 Feb 2008, Andrea Arcangeli wrote:
>
> > > Right but that pin requires taking a refcount which we cannot do.
> >
> > GRU can use my patch without the pin. XPMEM obviously can't use my
> > patch as my invalidate_page[s] are under the PT lock (a feature to fit
> > GRU/KVM in the simplest way), this is why an incremental patch adding
> > invalidate_range_start/end would be required to support XPMEM too.
>
> Doesnt the kernel in some situations release the page before releasing
the
> pte lock? Then there will be an external pte pointing to a page that may
> now have a different use. Its really bad if that pte does allow writes.

Sure the kernel does that most of the time, which is for example why I
had to use invalidate_page instead of invalidate_pages inside
zap_pte_range. Zero problems with that (this is also the exact reason

why I mentioned the tlb flushing code would need changes to convert some page in pages).

List: [linux-kernel](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) gumranet ! com>](#)
Date: [2008-02-05 5:25:25](#)
Message-ID: [20080205052525.GD7441 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Mon, Feb 04, 2008 at 11:09:01AM -0800, Christoph Lameter wrote:

> On Sun, 3 Feb 2008, Andrea Arcangeli wrote:

>

> > > Right but that pin requires taking a refcount which we cannot do.

> >

> > GRU can use my patch without the pin. XPMEM obviously can't use my
 > > patch as my invalidate_page[s] are under the PT lock (a feature to fit
 > > GRU/KVM in the simplest way), this is why an incremental patch adding
 > > invalidate_range_start/end would be required to support XPMEM too.

>

> Doesnt the kernel in some situations release the page before releasing
 the

> pte lock? Then there will be an external pte pointing to a page that may
 > now have a different use. Its really bad if that pte does allow writes.

Sure the kernel does that most of the time, which is for example why I had to use invalidate_page instead of invalidate_pages inside zap_pte_range. Zero problems with that (this is also the exact reason why I mentioned the tlb flushing code would need changes to convert some page in pages).

List: [linux-mm](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) gumranet ! com>](#)
Date: [2008-02-05 5:25:25](#)
Message-ID: [20080205052525.GD7441 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Mon, Feb 04, 2008 at 11:09:01AM -0800, Christoph Lameter wrote:

> On Sun, 3 Feb 2008, Andrea Arcangeli wrote:

>

> > > Right but that pin requires taking a refcount which we cannot do.

> >

> > GRU can use my patch without the pin. XPMEM obviously can't use my
 > > patch as my invalidate_page[s] are under the PT lock (a feature to fit
 > > GRU/KVM in the simplest way), this is why an incremental patch adding
 > > invalidate_range_start/end would be required to support XPMEM too.

>

> Doesnt the kernel in some situations release the page before releasing
 the

> pte lock? Then there will be an external pte pointing to a page that may
 > now have a different use. Its really bad if that pte does allow writes.

Sure the kernel does that most of the time, which is for example why I had to use invalidate_page instead of invalidate_pages inside zap_pte_range. Zero problems with that (this is also the exact reason why I mentioned the tlb flushing code would need changes to convert

some page in pages).

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-05 6:11:24](#)
Message-ID: [Pine.LNX.4.64.0802042206200.6739 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Tue, 5 Feb 2008, Andrea Arcangeli wrote:

> On Mon, Feb 04, 2008 at 11:09:01AM -0800, Christoph Lameter wrote:
 > > On Sun, 3 Feb 2008, Andrea Arcangeli wrote:
 > >
 > > > Right but that pin requires taking a refcount which we cannot do.
 > > >
 > > > GRU can use my patch without the pin. XPMEM obviously can't use my
 > > > patch as my invalidate_page[s] are under the PT lock (a feature to
 > > > fit
 > > > GRU/KVM in the simplest way), this is why an incremental patch adding
 > > > invalidate_range_start/end would be required to support XPMEM too.
 > > >
 > > Doesnt the kernel in some situations release the page before releasing
 > > the
 > > pte lock? Then there will be an external pte pointing to a page that
 > > may
 > > now have a different use. Its really bad if that pte does allow writes.
 > >
 > > Sure the kernel does that most of the time, which is for example why I
 > > had to use invalidate_page instead of invalidate_pages inside
 > > zap_pte_range. Zero problems with that (this is also the exact reason
 > > why I mentioned the tlb flushing code would need changes to convert
 > > some page in pages).

Zero problems only if you find having a single callout for every page acceptable. So the invalidate_range in your patch is only working sometimes. And even if it works then it has to be used on 2M range. Seems to be a bit fragile and needlessly complex.

"conversion of some page in pages"? A proposal to defer the freeing of the pages until after the pte_unlock?

List: [linux-kernel](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-05 6:11:24](#)
Message-ID: [Pine.LNX.4.64.0802042206200.6739 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Tue, 5 Feb 2008, Andrea Arcangeli wrote:

> On Mon, Feb 04, 2008 at 11:09:01AM -0800, Christoph Lameter wrote:
 > > On Sun, 3 Feb 2008, Andrea Arcangeli wrote:
 > >


```

> > > Right but that pin requires taking a refcount which we cannot do.
> > >
> > > GRU can use my patch without the pin. XPMEM obviously can't use my
> > > patch as my invalidate_page[s] are under the PT lock (a feature to
fit
> > > GRU/KVM in the simplest way), this is why an incremental patch adding
> > > invalidate_range_start/end would be required to support XPMEM too.
> >
> > Doesnt the kernel in some situations release the page before releasing
the
> > pte lock? Then there will be an external pte pointing to a page that
may
> > now have a different use. Its really bad if that pte does allow writes.
>
> Sure the kernel does that most of the time, which is for example why I
> had to use invalidate_page instead of invalidate_pages inside
> zap_pte_range. Zero problems with that (this is also the exact reason
> why I mentioned the tlb flushing code would need changes to convert
> some page in pages).

```

Zero problems only if you find having a single callout for every page acceptable. So the invalidate_range in your patch is only working sometimes. And even if it works then it has to be used on 2M range. Seems to be a bit fragile and needlessly complex.

"conversion of some page in pages"? A proposal to defer the freeing of the pages until after the pte_unlock?

List: [linux-mm](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-05 6:11:24](#)
Message-ID: [Pine.LNX.4.64.0802042206200.6739 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Tue, 5 Feb 2008, Andrea Arcangeli wrote:

```

> On Mon, Feb 04, 2008 at 11:09:01AM -0800, Christoph Lameter wrote:
> > On Sun, 3 Feb 2008, Andrea Arcangeli wrote:
> >
> > > Right but that pin requires taking a refcount which we cannot do.
> > >
> > > GRU can use my patch without the pin. XPMEM obviously can't use my
> > > patch as my invalidate_page[s] are under the PT lock (a feature to
fit
> > > GRU/KVM in the simplest way), this is why an incremental patch adding
> > > invalidate_range_start/end would be required to support XPMEM too.
> >
> > Doesnt the kernel in some situations release the page before releasing
the
> > pte lock? Then there will be an external pte pointing to a page that
may
> > now have a different use. Its really bad if that pte does allow writes.
>
> Sure the kernel does that most of the time, which is for example why I
> had to use invalidate_page instead of invalidate_pages inside
> zap_pte_range. Zero problems with that (this is also the exact reason
> why I mentioned the tlb flushing code would need changes to convert

```

> some page in pages).

Zero problems only if you find having a single callout for every page acceptable. So the invalidate_range in your patch is only working sometimes. And even if it works then it has to be used on 2M range. Seems to be a bit fragile and needlessly complex.

"conversion of some page in pages"? A proposal to defer the freeing of the pages until after the pte_unlock?

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) qumranet ! com>](#)
Date: [2008-02-05 18:08:02](#)
Message-ID: [20080205180802.GE7441 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Mon, Feb 04, 2008 at 10:11:24PM -0800, Christoph Lameter wrote:

> Zero problems only if you find having a single callout for every page
 > acceptable. So the invalidate_range in your patch is only working

invalidate_pages is only a further optimization that was straightforward in some places where the page isn't freed but only the pte modified.

> sometimes. And even if it works then it has to be used on 2M range. Seems
 > to be a bit fragile and needlessly complex.

The patch as a whole isn't fragile nor complex. Pretending to use invalidate_pages anywhere would be complex (and in turn more fragile than my current patch, complexity brings fragility).

Overall you can only argue against performance issues (my patch is simpler for GRU/KVM, and it sure isn't fragile, quite the opposite, given I never allow a coherency-loss between two threads that will read/write to two different physical pages for the same virtual address in remap_file_pages).

In performance terms with your patch before GRU can run follow_page it has to take a mm-wide global mutex where each thread in all cpus will have to take it. That will trash on >4-way when the tlb misses start to occur. There's nothing like that in my patch. Your approach will micro-optimize certain large pte-mangling calls, or do_exit, but those aren't interesting paths nor for GRU nor for KVM. You're optimizing for the slow path, and making the fast path slower.

> "conversion of some page in pages"? A proposal to defer the freeing of
 the
 > pages until after the pte_unlock?

There can be many tricks to optimize page in pages, but again munmap and do_exit aren't the interesting path to optimize, nor for GRU nor for KVM so it doesn't matter right now.

List: [linux-mm](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) qumranet ! com>](#)
Date: [2008-02-05 18:08:02](#)

Message-ID: [20080205180802.GE7441 \(\) v2 ! random](#)

[\[Download message RAW\]](#)

On Mon, Feb 04, 2008 at 10:11:24PM -0800, Christoph Lameter wrote:
 > Zero problems only if you find having a single callout for every page
 > acceptable. So the invalidate_range in your patch is only working

invalidate_pages is only a further optimization that was
 straightforward in some places where the page isn't freed but only the
 pte modified.

> sometimes. And even if it works then it has to be used on 2M range. Seems
 > to be a bit fragile and needlessly complex.

The patch as a whole isn't fragile nor complex. Pretending to use
 invalidate_pages anywhere would be complex (and in turn more fragile
 than my current patch, complexity brings fragility).

Overall you can only argue against performance issues (my patch is
 simpler for GRU/KVM, and it sure isn't fragile, quite the opposite,
 given I never allow a coherency-loss between two threads that will
 read/write to two different physical pages for the same virtual
 address in remap_file_pages).

In performance terms with your patch before GRU can run follow_page it
 has to take a mm-wide global mutex where each thread in all cpus will
 have to take it. That will trash on >4-way when the tlb misses start
 to occur. There's nothing like that in my patch. Your approach will
 micro-optimize certain large pte-mangling calls, or do_exit, but those
 aren't interesting paths nor for GRU nor for KVM. You're optimizing for
 the slow path, and making the fast path slower.

> "conversion of some page in pages"? A proposal to defer the freeing of
 the
 > pages until after the pte_unlock?

There can be many tricks to optimize page in pages, but again munmap
 and do_exit aren't the interesting path to optimize, nor for GRU nor
 for KVM so it doesn't matter right now.

List: [linux-kernel](#)

Subject: [Re: \[PATCH\] mmu notifiers #v5](#)

From: [Andrea Arcangeli <andrea \(\) gumranet ! com>](#)

Date: [2008-02-05 18:08:02](#)

Message-ID: [20080205180802.GE7441 \(\) v2 ! random](#)

[\[Download message RAW\]](#)

On Mon, Feb 04, 2008 at 10:11:24PM -0800, Christoph Lameter wrote:
 > Zero problems only if you find having a single callout for every page
 > acceptable. So the invalidate_range in your patch is only working

invalidate_pages is only a further optimization that was
 straightforward in some places where the page isn't freed but only the
 pte modified.

> sometimes. And even if it works then it has to be used on 2M range. Seems
 > to be a bit fragile and needlessly complex.

The patch as a whole isn't fragile nor complex. Pretending to use

invalidate_pages anywhere would be complex (and in turn more fragile than my current patch, complexity brings fragility).

Overall you can only argue against performance issues (my patch is simpler for GRU/KVM, and it sure isn't fragile, quite the opposite, given I never allow a coherency-loss between two threads that will read/write to two different physical pages for the same virtual address in remap_file_pages).

In performance terms with your patch before GRU can run follow_page it has to take a mm-wide global mutex where each thread in all cpus will have to take it. That will trash on >4-way when the tlb misses start to occur. There's nothing like that in my patch. Your approach will micro-optimize certain large pte-mangling calls, or do_exit, but those aren't interesting paths nor for GRU nor for KVM. You're optimizing for the slow path, and making the fast path slower.

> "conversion of some page in pages"? A proposal to defer the freeing of the
> pages until after the pte_unlock?

There can be many tricks to optimize page in pages, but again munmap and do_exit aren't the interesting path to optimize, nor for GRU nor for KVM so it doesn't matter right now.

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-05 18:17:41](#)
Message-ID: [Pine.LNX.4.64.0802051013440.11705 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Tue, 5 Feb 2008, Andrea Arcangeli wrote:

> given I never allow a coherency-loss between two threads that will
> read/write to two different physical pages for the same virtual
> address in remap_file_pages).

The other approach will not have any remote ptes at that point. Why would there be a coherency issue?

> In performance terms with your patch before GRU can run follow_page it
> has to take a mm-wide global mutex where each thread in all cpus will
> have to take it. That will trash on >4-way when the tlb misses start

No. It only has to lock the affected range. Remote page faults can occur while another part of the address space is being invalidated. The complexity of locking is up to the user of the mmu notifier. A simple implementation is satisfactory for the GRU right now. Should it become a problem then the lock granularity can be refined without changing the API.

> > "conversion of some page in pages"? A proposal to defer the freeing of the
> > pages until after the pte_unlock?
>
> There can be many tricks to optimize page in pages, but again munmap
> and do_exit aren't the interesting path to optimize, nor for GRU nor
> for KVM so it doesn't matter right now.

Still not sure what we are talking about here.

List: [linux-kernel](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-05 18:17:41](#)
Message-ID: [Pine.LNX.4.64.0802051013440.11705 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Tue, 5 Feb 2008, Andrea Arcangeli wrote:

> given I never allow a coherency-loss between two threads that will
 > read/write to two different physical pages for the same virtual
 > address in remap_file_pages).

The other approach will not have any remote ptes at that point. Why would there be a coherency issue?

> In performance terms with your patch before GRU can run follow_page it
 > has to take a mm-wide global mutex where each thread in all cpus will
 > have to take it. That will trash on >4-way when the tlb misses start

No. It only has to lock the affected range. Remote page faults can occur while another part of the address space is being invalidated. The complexity of locking is up to the user of the mmu notifier. A simple implementation is satisfactory for the GRU right now. Should it become a problem then the lock granularity can be refined without changing the API.

> > "conversion of some page in pages"? A proposal to defer the freeing of
 > the
 > > pages until after the pte_unlock?
 >
 > There can be many tricks to optimize page in pages, but again munmap
 > and do_exit aren't the interesting path to optimize, nor for GRU nor
 > for KVM so it doesn't matter right now.

Still not sure what we are talking about here.

List: [linux-mm](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-05 18:17:41](#)
Message-ID: [Pine.LNX.4.64.0802051013440.11705 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Tue, 5 Feb 2008, Andrea Arcangeli wrote:

> given I never allow a coherency-loss between two threads that will
 > read/write to two different physical pages for the same virtual
 > address in remap_file_pages).

The other approach will not have any remote ptes at that point. Why would there be a coherency issue?

> In performance terms with your patch before GRU can run follow_page it
 > has to take a mm-wide global mutex where each thread in all cpus will
 > have to take it. That will trash on >4-way when the tlb misses start

No. It only has to lock the affected range. Remote page faults can occur while another part of the address space is being invalidated. The complexity of locking is up to the user of the mmu notifier. A simple implementation is satisfactory for the GRU right now. Should it become a problem then the lock granularity can be refined without changing the API.

> > "conversion of some page in pages"? A proposal to defer the freeing of the
 > the
 > > pages until after the pte_unlock?
 >
 > There can be many tricks to optimize page in pages, but again munmap
 > and do_exit aren't the interesting path to optimize, nor for GRU nor
 > for KVM so it doesn't matter right now.

Still not sure what we are talking about here.

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) qumranet ! com>](#)
Date: [2008-02-05 20:55:19](#)
Message-ID: [20080205205519.GF7441 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Tue, Feb 05, 2008 at 10:17:41AM -0800, Christoph Lameter wrote:
 > The other approach will not have any remote ptes at that point. Why would
 > there be a coherency issue?

It never happens that two threads writes to two different physical pages by working on the same process virtual address. This is an issue only for KVM which is probably ok with it but certainly you can't consider the dependency on the page-pin less fragile or less complex than my PT lock approach.

> No. It only has to lock the affected range. Remote page faults can occur
 > while another part of the address space is being invalidated. The
 > complexity of locking is up to the user of the mmu notifier. A simple
 > implementation is satisfactory for the GRU right now. Should it become a
 > problem then the lock granularity can be refined without changing the
 API.

That will make the follow_page fast path even slower if it has to lookup a rbtree or a list of locked ranges. Still not comparable to the PT lock that 1) it's zero cost and 2) it'll provide an even more granular scalability.

> Still not sure what we are talking about here.

The apps using GRU/KVM never trigger large munmap/mremap/do_exit. You're optimizing for the irrelevant workload, by requiring unnecessary new locking in the GRU fast path.

List: [linux-mm](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)

From: [Andrea Arcangeli <andrea \(\) gumranet ! com>](mailto:andrea () gumranet ! com)
Date: [2008-02-05 20:55:19](#)
Message-ID: [20080205205519.GF7441 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Tue, Feb 05, 2008 at 10:17:41AM -0800, Christoph Lameter wrote:
> The other approach will not have any remote ptes at that point. Why would
> there be a coherency issue?

It never happens that two threads writes to two different physical pages by working on the same process virtual address. This is an issue only for KVM which is probably ok with it but certainly you can't consider the dependency on the page-pin less fragile or less complex than my PT lock approach.

> No. It only has to lock the affected range. Remote page faults can occur
> while another part of the address space is being invalidated. The
> complexity of locking is up to the user of the mmu notifier. A simple
> implementation is satisfactory for the GRU right now. Should it become a
> problem then the lock granularity can be refined without changing the
API.

That will make the follow_page fast path even slower if it has to lookup a rbtree or a list of locked ranges. Still not comparable to the PT lock that 1) it's zero cost and 2) it'll provide an even more granular scalability.

> Still not sure what we are talking about here.

The apps using GRU/KVM never trigger large munmap/mremap/do_exit. You're optimizing for the irrelevant workload, by requiring unnecessary new locking in the GRU fast path.

List: [linux-kernel](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) gumranet ! com>](mailto:andrea () gumranet ! com)
Date: [2008-02-05 20:55:19](#)
Message-ID: [20080205205519.GF7441 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Tue, Feb 05, 2008 at 10:17:41AM -0800, Christoph Lameter wrote:
> The other approach will not have any remote ptes at that point. Why would
> there be a coherency issue?

It never happens that two threads writes to two different physical pages by working on the same process virtual address. This is an issue only for KVM which is probably ok with it but certainly you can't consider the dependency on the page-pin less fragile or less complex than my PT lock approach.

> No. It only has to lock the affected range. Remote page faults can occur
> while another part of the address space is being invalidated. The
> complexity of locking is up to the user of the mmu notifier. A simple
> implementation is satisfactory for the GRU right now. Should it become a
> problem then the lock granularity can be refined without changing the
API.

That will make the `follow_page` fast path even slower if it has to lookup a `rbtree` or a list of locked ranges. Still not comparable to the PT lock that 1) it's zero cost and 2) it'll provide an even more granular scalability.

> Still not sure what we are talking about here.

The apps using GRU/KVM never trigger large `munmap/mremap/do_exit`. You're optimizing for the irrelevant workload, by requiring unnecessary new locking in the GRU fast path.

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-05 22:06:23](#)
Message-ID: [Pine.LNX.4.64.0802051400200.14665 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Tue, 5 Feb 2008, Andrea Arcangeli wrote:

> On Tue, Feb 05, 2008 at 10:17:41AM -0800, Christoph Lameter wrote:
 > > The other approach will not have any remote ptes at that point. Why would

> > there be a coherency issue?

>

> It never happens that two threads writes to two different physical
 > pages by working on the same process virtual address. This is an issue
 > only for KVM which is probably ok with it but certainly you can't
 > consider the dependency on the page-pin less fragile or less complex
 > than my PT lock approach.

You can avoid the page-pin and the pt lock completely by zapping the mappings at `_start` and then holding off new references until `_end`.

> > No. It only has to lock the affected range. Remote page faults can occur

> > while another part of the address space is being invalidated. The
 > > complexity of locking is up to the user of the mmu notifier. A simple
 > > implementation is satisfactory for the GRU right now. Should it become a

> > problem then the lock granularity can be refined without changing the API.

>

> That will make the `follow_page` fast path even slower if it has to
 > lookup a `rbtree` or a list of locked ranges. Still not comparable to
 > the PT lock that 1) it's zero cost and 2) it'll provide an even more
 > granular scalability.

As I said the implementation is up to the caller. Not sure what XPmem is using there but then XPmem is not using `follow_page`. The GRU would be using a lightway way of locking not `rbtrees`.

> > Still not sure what we are talking about here.

>

> The apps using GRU/KVM never trigger large
 > `munmap/mremap/do_exit`. You're optimizing for the irrelevant workload,
 > by requiring unnecessary new locking in the GRU fast path.

Maybe that is true for KVM but certainly not true for the GRU. The GRU is designed to manage several petabytes of memory that may be mapped by a series of Linux instances. If a process only maps a small chunk of 4 Gigabytes then we already have to deal with 1 mio callbacks.

List: [linux-mm](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-05 22:06:23](#)
Message-ID: [Pine.LNX.4.64.0802051400200.14665 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Tue, 5 Feb 2008, Andrea Arcangeli wrote:

> On Tue, Feb 05, 2008 at 10:17:41AM -0800, Christoph Lameter wrote:
 > > The other approach will not have any remote ptes at that point. Why
 would
 > > there be a coherency issue?
 >
 > It never happens that two threads writes to two different physical
 > pages by working on the same process virtual address. This is an issue
 > only for KVM which is probably ok with it but certainly you can't
 > consider the dependency on the page-pin less fragile or less complex
 > than my PT lock approach.

You can avoid the page-pin and the pt lock completely by zapping the mappings at `_start` and then holding off new references until `_end`.

> > No. It only has to lock the affected range. Remote page faults can
 occur
 > > while another part of the address space is being invalidated. The
 > > complexity of locking is up to the user of the mmu notifier. A simple
 > > implementation is satisfactory for the GRU right now. Should it become
 a
 > > problem then the lock granularity can be refined without changing the
 API.
 >
 > That will make the `follow_page` fast path even slower if it has to
 > lookup a rbtree or a list of locked ranges. Still not comparable to
 > the PT lock that 1) it's zero cost and 2) it'll provide an even more
 > granular scalability.

As I said the implementation is up to the caller. Not sure what XPMem is using there but then XPMem is not using `follow_page`. The GRU would be using a lightway way of locking not rbtrees.

> > Still not sure what we are talking about here.
 >
 > The apps using GRU/KVM never trigger large
 > `munmap/mremap/do_exit`. You're optimizing for the irrelevant workload,
 > by requiring unnecessary new locking in the GRU fast path.

Maybe that is true for KVM but certainly not true for the GRU. The GRU is designed to manage several petabytes of memory that may be mapped by a series of Linux instances. If a process only maps a small chunk of 4 Gigabytes then we already have to deal with 1 mio callbacks.

List: [linux-kernel](#)

Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-05 22:06:23](#)
Message-ID: [Pine.LNX.4.64.0802051400200.14665 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Tue, 5 Feb 2008, Andrea Arcangeli wrote:

> On Tue, Feb 05, 2008 at 10:17:41AM -0800, Christoph Lameter wrote:
 > > The other approach will not have any remote ptes at that point. Why would
 > > there be a coherency issue?
 >
 > It never happens that two threads writes to two different physical
 > pages by working on the same process virtual address. This is an issue
 > only for KVM which is probably ok with it but certainly you can't
 > consider the dependency on the page-pin less fragile or less complex
 > than my PT lock approach.

You can avoid the page-pin and the pt lock completely by zapping the mappings at `_start` and then holding off new references until `_end`.

> > No. It only has to lock the affected range. Remote page faults can occur
 > > while another part of the address space is being invalidated. The
 > > complexity of locking is up to the user of the mmu notifier. A simple
 > > implementation is satisfactory for the GRU right now. Should it become a
 > > problem then the lock granularity can be refined without changing the API.
 >
 > That will make the `follow_page` fast path even slower if it has to
 > lookup a rbtree or a list of locked ranges. Still not comparable to
 > the PT lock that 1) it's zero cost and 2) it'll provide an even more
 > granular scalability.

As I said the implementation is up to the caller. Not sure what XPMem is using there but then XPMem is not using `follow_page`. The GRU would be using a lightway way of locking not rbtrees.

> > Still not sure what we are talking about here.
 >
 > The apps using GRU/KVM never trigger large
 > `munmap/mremap/do_exit`. You're optimizing for the irrelevant workload,
 > by requiring unnecessary new locking in the GRU fast path.

Maybe that is true for KVM but certainly not true for the GRU. The GRU is designed to manage several petabytes of memory that may be mapped by a series of Linux instances. If a process only maps a small chunk of 4 Gigabytes then we already have to deal with 1 mio callbacks.

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Robin Holt <holt \(\) sgi ! com>](#)
Date: [2008-02-05 22:12:22](#)
Message-ID: [20080205221221.GP17211 \(\) sgi ! com](#)
[\[Download message RAW\]](#)

On Tue, Feb 05, 2008 at 02:06:23PM -0800, Christoph Lameter wrote:
 > On Tue, 5 Feb 2008, Andrea Arcangeli wrote:
 >
 > > On Tue, Feb 05, 2008 at 10:17:41AM -0800, Christoph Lameter wrote:
 > > > The other approach will not have any remote ptes at that point. Why
 would
 > > > there be a coherency issue?
 > >
 > > It never happens that two threads writes to two different physical
 > > pages by working on the same process virtual address. This is an issue
 > > only for KVM which is probably ok with it but certainly you can't
 > > consider the dependency on the page-pin less fragile or less complex
 > > than my PT lock approach.
 >
 > You can avoid the page-pin and the pt lock completely by zapping the
 > mappings at `_start` and then holding off new references until `_end`.

XPMMEM is doing this by putting our equivalent structure of the mm into a
 recalling state which will cause all future faulters to back off, it then
 marks any currently active faults in the range as invalid (we have a very
 small number of possible concurrent faulters for a different reason),
 proceeds to start remote shoot-downs, waits for those shoot-downs to
 complete, then returns from the `_begin` callout with the mm-equiv still in
 the recalling state. Additional recalls may occur, but no new faults can.
 The `_end` callout reduces the number of active recalls until there are
 none left at which point the faulters are allowed to proceed again.

Thanks,
 Robin

List: [linux-mm](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Robin Holt <holt \(\) sgi ! com>](#)
Date: [2008-02-05 22:12:22](#)
Message-ID: [20080205221221.GP17211 \(\) sgi ! com](#)
[\[Download message RAW\]](#)

On Tue, Feb 05, 2008 at 02:06:23PM -0800, Christoph Lameter wrote:
 > On Tue, 5 Feb 2008, Andrea Arcangeli wrote:
 >
 > > On Tue, Feb 05, 2008 at 10:17:41AM -0800, Christoph Lameter wrote:
 > > > The other approach will not have any remote ptes at that point. Why
 would
 > > > there be a coherency issue?
 > >
 > > It never happens that two threads writes to two different physical
 > > pages by working on the same process virtual address. This is an issue
 > > only for KVM which is probably ok with it but certainly you can't
 > > consider the dependency on the page-pin less fragile or less complex
 > > than my PT lock approach.
 >
 > You can avoid the page-pin and the pt lock completely by zapping the
 > mappings at `_start` and then holding off new references until `_end`.

XPMMEM is doing this by putting our equivalent structure of the mm into a
 recalling state which will cause all future faulters to back off, it then
 marks any currently active faults in the range as invalid (we have a very
 small number of possible concurrent faulters for a different reason),

proceeds to start remote shoot-downs, waits for those shoot-downs to complete, then returns from the `_begin` callout with the `mm-equiv` still in the recalling state. Additional recalls may occur, but no new faults can. The `_end` callout reduces the number of active recalls until there are none left at which point the faulters are allowed to proceed again.

Thanks,
Robin

List: [linux-kernel](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Robin Holt <holt \(\) sgi ! com>](#)
Date: [2008-02-05 22:12:22](#)
Message-ID: [20080205221221.GP17211 \(\) sgi ! com](#)
[\[Download message RAW\]](#)

On Tue, Feb 05, 2008 at 02:06:23PM -0800, Christoph Lameter wrote:

> On Tue, 5 Feb 2008, Andrea Arcangeli wrote:

>

> > On Tue, Feb 05, 2008 at 10:17:41AM -0800, Christoph Lameter wrote:

> > > The other approach will not have any remote ptes at that point. Why would

> > > there be a coherency issue?

> >

> > It never happens that two threads writes to two different physical
 > > pages by working on the same process virtual address. This is an issue
 > > only for KVM which is probably ok with it but certainly you can't
 > > consider the dependency on the page-pin less fragile or less complex
 > > than my PT lock approach.

>

> You can avoid the page-pin and the pt lock completely by zapping the
 > mappings at `_start` and then holding off new references until `_end`.

XPMMEM is doing this by putting our equivalent structure of the `mm` into a recalling state which will cause all future faulters to back off, it then marks any currently active faults in the range as invalid (we have a very small number of possible concurrent faulters for a different reason), proceeds to start remote shoot-downs, waits for those shoot-downs to complete, then returns from the `_begin` callout with the `mm-equiv` still in the recalling state. Additional recalls may occur, but no new faults can. The `_end` callout reduces the number of active recalls until there are none left at which point the faulters are allowed to proceed again.

Thanks,
Robin

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) qumranet ! com>](#)
Date: [2008-02-05 22:26:58](#)
Message-ID: [20080205222657.GG7441 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Tue, Feb 05, 2008 at 02:06:23PM -0800, Christoph Lameter wrote:

> On Tue, 5 Feb 2008, Andrea Arcangeli wrote:

>

> > On Tue, Feb 05, 2008 at 10:17:41AM -0800, Christoph Lameter wrote:

> > > The other approach will not have any remote ptes at that point. Why would
 > > > there be a coherency issue?
 > >
 > > It never happens that two threads writes to two different physical
 > > pages by working on the same process virtual address. This is an issue
 > > only for KVM which is probably ok with it but certainly you can't
 > > consider the dependency on the page-pin less fragile or less complex
 > > than my PT lock approach.
 >
 > You can avoid the page-pin and the pt lock completely by zapping the
 > mappings at `_start` and then holding off new references until `_end`.

Avoid the PT lock? The PT lock has to be taken anyway by the linux VM.

"holding off new references until `_end`" = per-range mutex less scalar and more expensive than the PT lock that has to be taken anyway.

> As I said the implementation is up to the caller. Not sure what
 > XPmem is using there but then XPmem is not using `follow_page`. The GRU
 > would be using a lightway way of locking not rbtrees.

"lightway way of locking" = mm-wide-mutex (not necessary at all if we take advantage of the per-pte-scalar PT lock that has to be taken anyway like in my patch)

> Maybe that is true for KVM but certainly not true for the GRU. The GRU is
 > designed to manage several petabytes of memory that may be mapped by a
 > series of Linux instances. If a process only maps a small chunk of 4
 > Gigabytes then we already have to deal with 1 mio callbacks.

KVM is also going to map a lot of stuff, but mapping involves `mmap`, `munmap`/`mremap`/`mprotect` not. The size of `mmap` is irrelevant in both approaches. optimizing `do_exit` by making the `tlb-miss` runtime slower doesn't sound great to me and that's your patch does if you force GRU to use it.

List: [linux-kernel](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) gumranet ! com>](#)
Date: [2008-02-05 22:26:58](#)
Message-ID: [20080205222657.GG7441 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Tue, Feb 05, 2008 at 02:06:23PM -0800, Christoph Lameter wrote:

> On Tue, 5 Feb 2008, Andrea Arcangeli wrote:
 >

> > On Tue, Feb 05, 2008 at 10:17:41AM -0800, Christoph Lameter wrote:
 > > > The other approach will not have any remote ptes at that point. Why would
 > > > there be a coherency issue?
 > >
 > > It never happens that two threads writes to two different physical
 > > pages by working on the same process virtual address. This is an issue
 > > only for KVM which is probably ok with it but certainly you can't
 > > consider the dependency on the page-pin less fragile or less complex
 > > than my PT lock approach.
 >

> You can avoid the page-pin and the pt lock completely by zapping the
> mappings at `_start` and then holding off new references until `_end`.

Avoid the PT lock? The PT lock has to be taken anyway by the linux VM.

"holding off new references until `_end`" = per-range mutex less scalar and more expensive than the PT lock that has to be taken anyway.

> As I said the implementation is up to the caller. Not sure what
> XPMem is using there but then XPMem is not using `follow_page`. The GRU
> would be using a lightway way of locking not rbtrees.

"lightway way of locking" = mm-wide-mutex (not necessary at all if we take advantage of the per-pte-scalar PT lock that has to be taken anyway like in my patch)

> Maybe that is true for KVM but certainly not true for the GRU. The GRU is
> designed to manage several petabytes of memory that may be mapped by a
> series of Linux instances. If a process only maps a small chunk of 4
> Gigabytes then we already have to deal with 1 mio callbacks.

KVM is also going to map a lot of stuff, but mapping involves `mmap`, `munmap`/`mremap`/`mprotect` not. The size of `mmap` is irrelevant in both approaches. optimizing `do_exit` by making the tlb-miss runtime slower doesn't sound great to me and that's your patch does if you force GRU to use it.

List: [linux-mm](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) gnumranet ! com>](#)
Date: [2008-02-05 22:26:58](#)
Message-ID: [20080205222657.GG7441 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Tue, Feb 05, 2008 at 02:06:23PM -0800, Christoph Lameter wrote:

> On Tue, 5 Feb 2008, Andrea Arcangeli wrote:

>

> > On Tue, Feb 05, 2008 at 10:17:41AM -0800, Christoph Lameter wrote:

> > > The other approach will not have any remote ptes at that point. Why would

> > > there be a coherency issue?

> >

> > It never happens that two threads writes to two different physical
> > pages by working on the same process virtual address. This is an issue
> > only for KVM which is probably ok with it but certainly you can't
> > consider the dependency on the page-pin less fragile or less complex
> > than my PT lock approach.

>

> You can avoid the page-pin and the pt lock completely by zapping the
> mappings at `_start` and then holding off new references until `_end`.

Avoid the PT lock? The PT lock has to be taken anyway by the linux VM.

"holding off new references until `_end`" = per-range mutex less scalar and more expensive than the PT lock that has to be taken anyway.

> As I said the implementation is up to the caller. Not sure what

> XPmem is using there but then XPmem is not using follow_page. The GRU
> would be using a lightway way of locking not rbtrees.

"lightway way of locking" = mm-wide-mutex (not necessary at all if we
take advantage of the per-pte-scalar PT lock that has to be taken
anyway like in my patch)

> Maybe that is true for KVM but certainly not true for the GRU. The GRU is
> designed to manage several petabytes of memory that may be mapped by a
> series of Linux instances. If a process only maps a small chunk of 4
> Gigabytes then we already have to deal with 1 mio callbacks.

KVM is also going to map a lot of stuff, but mapping involves mmap,
munmap/mremap/mprotect not. The size of mmap is irrelevant in both
approaches. optimizing do_exit by making the tlb-miss runtime slower
doesn't sound great to me and that's your patch does if you force GRU

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-05 23:10:52](#)
Message-ID: [Pine.LNX.4.64.0802051504450.16261 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Tue, 5 Feb 2008, Andrea Arcangeli wrote:

> > You can avoid the page-pin and the pt lock completely by zapping the
> > mappings at _start and then holding off new references until _end.
>
> "holding off new references until _end" = per-range mutex less scalar
> and more expensive than the PT lock that has to be taken anyway.

You can of course setup a 2M granularity lock to get the same granularity
as the pte lock. That would even work for the cases where you have to page
pin now.

> > Maybe that is true for KVM but certainly not true for the GRU. The GRU
is
> > designed to manage several petabytes of memory that may be mapped by a
> > series of Linux instances. If a process only maps a small chunk of 4
> > Gigabytes then we already have to deal with 1 mio callbacks.
>
> KVM is also going to map a lot of stuff, but mapping involves mmap,
> munmap/mremap/mprotect not. The size of mmap is irrelevant in both
> approaches. optimizing do_exit by making the tlb-miss runtime slower
> doesn't sound great to me and that's your patch does if you force GRU
> to use it.

The size of the mmap is relevant if you have to perform callbacks on
every mapped page that involved take mmu specific locks. That seems to be
the case with this approach.

Optimizing do_exit by taking a single lock to zap all external references
instead of 1 mio callbacks somehow leads to slowdown?

List: [linux-mm](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)

From: [Christoph Lameter <clameter \(\) sgi ! com>](mailto:clameter@sgi.com)
Date: [2008-02-05 23:10:52](#)
Message-ID: [Pine.LNX.4.64.0802051504450.16261 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Tue, 5 Feb 2008, Andrea Arcangeli wrote:

> > You can avoid the page-pin and the pt lock completely by zapping the
 > > mappings at _start and then holding off new references until _end.
 >
 > "holding off new references until _end" = per-range mutex less scalar
 > and more expensive than the PT lock that has to be taken anyway.

You can of course setup a 2M granularity lock to get the same granularity as the pte lock. That would even work for the cases where you have to page pin now.

> > Maybe that is true for KVM but certainly not true for the GRU. The GRU is
 > > designed to manage several petabytes of memory that may be mapped by a
 > > series of Linux instances. If a process only maps a small chunk of 4
 > > Gigabytes then we already have to deal with 1 mio callbacks.
 >
 > KVM is also going to map a lot of stuff, but mapping involves mmap,
 > munmap/mremap/mprotect not. The size of mmap is irrelevant in both
 > approaches. optimizing do_exit by making the tlb-miss runtime slower
 > doesn't sound great to me and that's your patch does if you force GRU
 > to use it.

The size of the mmap is relevant if you have to perform callbacks on every mapped page that involved take mmu specific locks. That seems to be the case with this approach.

Optimizing do_exit by taking a single lock to zap all external references instead of 1 mio callbacks somehow leads to slowdown?

List: [linux-kernel](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](mailto:clameter@sgi.com)
Date: [2008-02-05 23:10:52](#)
Message-ID: [Pine.LNX.4.64.0802051504450.16261 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Tue, 5 Feb 2008, Andrea Arcangeli wrote:

> > You can avoid the page-pin and the pt lock completely by zapping the
 > > mappings at _start and then holding off new references until _end.
 >
 > "holding off new references until _end" = per-range mutex less scalar
 > and more expensive than the PT lock that has to be taken anyway.

You can of course setup a 2M granularity lock to get the same granularity as the pte lock. That would even work for the cases where you have to page pin now.

> > Maybe that is true for KVM but certainly not true for the GRU. The GRU is designed to manage several petabytes of memory that may be mapped by a series of Linux instances. If a process only maps a small chunk of 4 Gigabytes then we already have to deal with 1 mio callbacks.

>

> KVM is also going to map a lot of stuff, but mapping involves mmap, munmap/mremap/mprotect not. The size of mmap is irrelevant in both approaches. optimizing do_exit by making the tlb-miss runtime slower doesn't sound great to me and that's your patch does if you force GRU to use it.

The size of the mmap is relevant if you have to perform callbacks on every mapped page that involved take mmu specific locks. That seems to be the case with this approach.

Optimizing do_exit by taking a single lock to zap all external references instead of 1 mio callbacks somehow leads to slowdown?

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) gumranet ! com>](#)
Date: [2008-02-05 23:47:42](#)
Message-ID: [20080205234742.GI7441 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Tue, Feb 05, 2008 at 03:10:52PM -0800, Christoph Lameter wrote:

> On Tue, 5 Feb 2008, Andrea Arcangeli wrote:

>

> > > You can avoid the page-pin and the pt lock completely by zapping the mappings at _start and then holding off new references until _end.

> >

> > "holding off new references until _end" = per-range mutex less scalar and more expensive than the PT lock that has to be taken anyway.

>

> You can of course setup a 2M granularity lock to get the same granularity as the pte lock. That would even work for the cases where you have to page pin now.

If you set a 2M granularity lock, the _start callback would need to do:

```
for_each_2m_lock()
    mutex_lock()
```

so you'd run zillion of mutex_lock in a row, you're the one with the million of operations argument.

> The size of the mmap is relevant if you have to perform callbacks on every mapped page that involved take mmu specific locks. That seems to be the case with this approach.

mmap should never trigger any range_start/_end callback unless it's overwriting an older mapping which is definitely not the interesting workload for those apps including kvm.

> Optimizing do_exit by taking a single lock to zap all external references instead of 1 mio callbacks somehow leads to slowdown?

It can if the application runs for more than a couple of seconds, i.e. not a fork flood in which you care about `do_exit` speed. Keep in mind if you had `lmio invalidate_pages` callback it means you previously called `follow_page` 1 mio of times too...

List: [linux-mm](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) qumranet ! com>](#)
Date: [2008-02-05 23:47:42](#)
Message-ID: [20080205234742.GI7441 \(\) v2 ! random](#)
[\[Download message RAW\]](#)

On Tue, Feb 05, 2008 at 03:10:52PM -0800, Christoph Lameter wrote:

> On Tue, 5 Feb 2008, Andrea Arcangeli wrote:

>

> > > You can avoid the page-pin and the pt lock completely by zapping the
 > > > mappings at `_start` and then holding off new references until `_end`.

> >

> > "holding off new references until `_end`" = per-range mutex less scalar
 > > and more expensive than the PT lock that has to be taken anyway.

>

> You can of course setup a 2M granularity lock to get the same granularity
 > as the pte lock. That would even work for the cases where you have to
 > page
 > pin now.

If you set a 2M granularity lock, the `_start` callback would need to do:

```
for_each_2m_lock()
    mutex_lock()
```

so you'd run zillion of `mutex_lock` in a row, you're the one with the million of operations argument.

> The size of the mmap is relevant if you have to perform callbacks on
 > every mapped page that involved take mmu specific locks. That seems to be
 > the case with this approach.

mmap should never trigger any `range_start/_end` callback unless it's overwriting an older mapping which is definitely not the interesting workload for those apps including kvm.

> Optimizing `do_exit` by taking a single lock to zap all external references
 > instead of 1 mio callbacks somehow leads to slowdown?

It can if the application runs for more than a couple of seconds, i.e. not a fork flood in which you care about `do_exit` speed. Keep in mind if you had `lmio invalidate_pages` callback it means you previously called `follow_page` 1 mio of times too...

List: [linux-kernel](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Andrea Arcangeli <andrea \(\) qumranet ! com>](#)
Date: [2008-02-05 23:47:42](#)
Message-ID: [20080205234742.GI7441 \(\) v2 ! random](#)

[\[Download message RAW\]](#)

On Tue, Feb 05, 2008 at 03:10:52PM -0800, Christoph Lameter wrote:

> On Tue, 5 Feb 2008, Andrea Arcangeli wrote:

>

> > > You can avoid the page-pin and the pt lock completely by zapping the
> > > mappings at `_start` and then holding off new references until `_end`.

> >

> > "holding off new references until `_end`" = per-range mutex less scalar
> > and more expensive than the PT lock that has to be taken anyway.

>

> You can of course setup a 2M granularity lock to get the same granularity
> as the pte lock. That would even work for the cases where you have to

page

> pin now.

If you set a 2M granularity lock, the `_start` callback would need to do:

```
for_each_2m_lock()
    mutex_lock()
```

so you'd run zillion of `mutex_lock` in a row, you're the one with the million of operations argument.

> The size of the `mmap` is relevant if you have to perform callbacks on
> every mapped page that involved take mmu specific locks. That seems to be
> the case with this approach.

`mmap` should never trigger any `range_start/_end` callback unless it's overwriting an older mapping which is definitely not the interesting workload for those apps including `kvm`.

> Optimizing `do_exit` by taking a single lock to zap all external references
> instead of 1 mio callbacks somehow leads to slowdown?

It can if the application runs for more than a couple of seconds, i.e. not a fork flood in which you care about `do_exit` speed. Keep in mind if you had `1mio invalidate_pages` callback it means you previously called `follow_page` 1 mio of times too...

List: [kvm](#)
Subject: [Re: \[kvm-devel\] \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-06 0:04:15](#)
Message-ID: [Pine.LNX.4.64.0802051555380.17243 \(\) schroedinger ! engr ! sgi ! com](#)

[\[Download message RAW\]](#)

On Wed, 6 Feb 2008, Andrea Arcangeli wrote:

> > You can of course setup a 2M granularity lock to get the same
granularity

> > as the pte lock. That would even work for the cases where you have to
page

> > pin now.

>

> If you set a 2M granularity lock, the `_start` callback would need to

> do:

```
>
>     for_each_2m_lock()
>         mutex_lock()
>
> so you'd run zillion of mutex_lock in a row, you're the one with the
> million of operations argument.
```

There is no requirement to do a linear search. No one in his right mind would implement a performance critical operation that way.

```
> > The size of the mmap is relevant if you have to perform callbacks on
> > every mapped page that involved take mmu specific locks. That seems to
be
> > the case with this approach.
>
> mmap should never trigger any range_start/_end callback unless it's
> overwriting an older mapping which is definitely not the interesting
> workload for those apps including kvm.
```

There is still at least the need for teardown on exit. And you need to consider the boundless creativity of user land programmers. You would not believe what I have seen....

```
> > Optimizing do_exit by taking a single lock to zap all external
references
> > instead of 1 mio callbacks somehow leads to slowdown?
>
> It can if the application runs for more than a couple of seconds,
> i.e. not a fork flood in which you care about do_exit speed. Keep in
> mind if you had 1mio invalidate_pages callback it means you previously
> called follow_page 1 mio of times too...
```

That is another problem were we are also in need of solutions. I believe we have discussed that elsewhere.

List: [linux-mm](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-06 0:04:15](#)
Message-ID: [Pine.LNX.4.64.0802051555380.17243 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Wed, 6 Feb 2008, Andrea Arcangeli wrote:

```
> > You can of course setup a 2M granularity lock to get the same
granularity
> > as the pte lock. That would even work for the cases where you have to
page
> > pin now.
>
> If you set a 2M granularity lock, the _start callback would need to
> do:
>
>     for_each_2m_lock()
>         mutex_lock()
>
> so you'd run zillion of mutex_lock in a row, you're the one with the
> million of operations argument.
```

There is no requirement to do a linear search. No one in his right mind would implement a performance critical operation that way.

> > The size of the mmap is relevant if you have to perform callbacks on
> > every mapped page that involved take mmu specific locks. That seems to
be
> > the case with this approach.
>
> mmap should never trigger any range_start/_end callback unless it's
> overwriting an older mapping which is definitely not the interesting
> workload for those apps including kvm.

There is still at least the need for teardown on exit. And you need to consider the boundless creativity of user land programmers. You would not believe what I have seen....

> > Optimizing do_exit by taking a single lock to zap all external
references
> > instead of 1 mio callbacks somehow leads to slowdown?
>
> It can if the application runs for more than a couple of seconds,
> i.e. not a fork flood in which you care about do_exit speed. Keep in
> mind if you had 1mio invalidate_pages callback it means you previously
> called follow_page 1 mio of times too...

That is another problem were we are also in need of solutions. I believe we have discussed that elsewhere.

List: [linux-kernel](#)
Subject: [Re: \[PATCH\] mmu notifiers #v5](#)
From: [Christoph Lameter <clameter \(\) sgi ! com>](#)
Date: [2008-02-06 0:04:15](#)
Message-ID: [Pine.LNX.4.64.0802051555380.17243 \(\) schroedinger ! engr ! sgi ! com](#)
[\[Download message RAW\]](#)

On Wed, 6 Feb 2008, Andrea Arcangeli wrote:

> > You can of course setup a 2M granularity lock to get the same
granularity
> > as the pte lock. That would even work for the cases where you have to
page
> > pin now.
>
> If you set a 2M granularity lock, the _start callback would need to
> do:
>
> for_each_2m_lock()
> mutex_lock()
>
> so you'd run zillion of mutex_lock in a row, you're the one with the
> million of operations argument.

There is no requirement to do a linear search. No one in his right mind would implement a performance critical operation that way.

> > The size of the mmap is relevant if you have to perform callbacks on

> > every mapped page that involved take mmu specific locks. That seems to be
> > the case with this approach.
>
> mmap should never trigger any range_start/_end callback unless it's
> overwriting an older mapping which is definitely not the interesting
> workload for those apps including kvm.

There is still at least the need for teardown on exit. And you need to consider the boundless creativity of user land programmers. You would not believe what I have seen....

> > Optimizing do_exit by taking a single lock to zap all external references
> > instead of 1 mio callbacks somehow leads to slowdown?
>
> It can if the application runs for more than a couple of seconds,
> i.e. not a fork flood in which you care about do_exit speed. Keep in
> mind if you had 1mio invalidate_pages callback it means you previously
> called follow_page 1 mio of times too...

That is another problem were we are also in need of solutions. I believe we have discussed that elsewhere.

Anexo G – Discussões da Comunidade Apache

Interação 1.7.APH

From	"Roy T. Fielding" <field...@gbiv.com>
Subject	Java Service Wrapper
Date	Wed, 02 Apr 2008 19:20:42 GMT

The Java Service Wrapper is being used (and sometimes redistributed) by many of our Java projects (ActiveMQ, ServiceMix, Geronimo, OFBiz, Tomcat).
It was formerly available under an MIT/BSD style license, as can be seen in the 3.2.3 version at

<http://sourceforge.net/projects/wrapper/>

More recent versions of the product are currently under dual GPLv2 and commercial licensing.

<http://wrapper.tanukisoftware.org/doc/english/licenseOverview.html>

And yet no mention of its license is found in our documentation.

<http://activemq.apache.org/java-service-wrapper.html>
<http://docs.ofbiz.org/display/OFBIZ/How+to+Run+OFBiz+as+Windows+Service+with+Java+Service+Wrapper>
<http://people.apache.org/~fhanik/wrapper.html>
<http://wiki.apache.org/db-derby/DerbyWindowsService>

and the only pages we have that refer to the old version are in Geronimo

<http://cwiki.apache.org/GMOxDOC11/configuring-geronimo-as-a-windows-service.html>
<http://cwiki.apache.org/GMOxDOC20/configuring-geronimo-as-a-windows-service.html>

If your project is currently distributing or telling users to make use of the Java Service Wrapper, please be sure that we only use the old licensed version, that our documentation points to the old site, and that when we do point folks to wrapper.tanukisoftware.org (which we should do because some users are not allergic to GPLv2) we should also point out that the newer versions are under different licenses (GPLv2/commercial).

....Roy

From "Henri Yandell" <bay...@apache.org>

Subject Re: Java Service Wrapper

Date Wed, 02 Apr 2008 19:34:47 GMT

Raises the question of whether anyone has needed the improvements/bugfixes/whatever in the later GPL versions, and whether we should consider forking the old version.

Hen

On Wed, Apr 2, 2008 at 12:20 PM, Roy T. Fielding <fielding@gbiv.com> wrote:

> The Java Service Wrapper is being used (and sometimes redistributed) by
> many of our Java projects (ActiveMQ, ServiceMix, Geronimo, OFBiz, Tomcat).

> It was formerly available under an MIT/BSD style license, as can be seen

> in the 3.2.3 version at

>

> <http://sourceforge.net/projects/wrapper/>

>

> More recent versions of the product are currently under dual GPLv2
> and commercial licensing.

>

> <http://wrapper.tanukisoftware.org/doc/english/licenseOverview.html>

>

> And yet no mention of its license is found in our documentation.

>

> <http://activemq.apache.org/java-service-wrapper.html>

>

>

<http://docs.ofbiz.org/display/OFBIZ/How+to+Run+OFBiz+as+Windows+Service+with+Java+Service+Wrapper>

> <http://people.apache.org/~fhanik/wrapper.html>

> <http://wiki.apache.org/db-derby/DerbyWindowsService>

>

> and the only pages we have that refer to the old version are in Geronimo

>

>

> <http://cwiki.apache.org/GMOxDOC11/configuring-geronimo-as-a-windows-service.html>

>

> <http://cwiki.apache.org/GMOxDOC20/configuring-geronimo-as-a-windows-service.html>

>

> If your project is currently distributing or telling users to make use of

> the Java Service Wrapper, please be sure that we only use the old licensed

> version, that our documentation points to the old site, and that when we do

> point folks to wrapper.tanukisoftware.org (which we should do because some

> users are not allergic to GPLv2) we should also point out that the newer

> versions are under different licenses (GPLv2/commercial).

>

>Roy

From Brett Porter <br...@apache.org>
Subject Re: Java Service Wrapper
Date Thu, 03 Apr 2008 00:02:10 GMT

Thanks for bringing this up Roy, it was on my TODO list. I found it a bit unclear on the site that this change had occurred, so I wasn't sure if others would have noticed. Note that the sourceforge site isn't necessarily the old site - the code stored in SVN there is still the latest, GPL version of what I presume is the community edition.

I actually made a fix the other day and on going to contribute it back found it was already present in the latest version, but that it was now under GPL. I took a look at the release notes and didn't see anything else that was of immediate interest to my projects. We'll continue using the 3.2.3 delta pack for the default distribution.

Cheers,
 Brett

On 03/04/2008, at 6:34 AM, Henri Yandell wrote:

> Raises the question of whether anyone has needed the
 > improvements/bugfixes/whatever in the later GPL versions, and whether
 > we should consider forking the old version.
 >
 > Hen
 >
 > On Wed, Apr 2, 2008 at 12:20 PM, Roy T. Fielding <fielding@gbiv.com>
 > wrote:
 >> The Java Service Wrapper is being used (and sometimes
 >> redistributed) by
 >> many of our Java projects (ActiveMQ, ServiceMix, Geronimo, OFBiz,
 >> Tomcat).
 >> It was formerly available under an MIT/BSD style license, as can be
 >> seen
 >> in the 3.2.3 version at
 >>
 >> <http://sourceforge.net/projects/wrapper/>
 >>
 >> More recent versions of the product are currently under dual GPLv2
 >> and commercial licensing.
 >>
 >> <http://wrapper.tanukisoftware.org/doc/english/licenseOverview.html>
 >>
 >> And yet no mention of its license is found in our documentation.
 >>
 >> <http://activemq.apache.org/java-service-wrapper.html>
 >>
 >>
 >> <http://docs.ofbiz.org/display/OFBIZ/How+to+Run+OFBiz+as+Windows+Service+with+Java+Service+Wrapper>
 >> <http://people.apache.org/~fhanik/wrapper.html>
 >> <http://wiki.apache.org/db-derby/DerbyWindowsService>
 >>
 >> and the only pages we have that refer to the old version are in
 >> Geronimo
 >>
 >>
 >> <http://cwiki.apache.org/GMOxDOC11/configuring-geronimo-as-a-windows->

```

service.html
>>
>> http://cwiki.apache.org/GMOxDOC20/configuring-geronimo-as-a-windows-
service.html
>>
>> If your project is currently distributing or telling users to make
>> use of
>> the Java Service Wrapper, please be sure that we only use the old
>> licensed
>> version, that our documentation points to the old site, and that
>> when we do
>> point folks to wrapper.tanukisoftware.org (which we should do
>> because some
>> users are not allergic to GPLv2) we should also point out that the
>> newer
>> versions are under different licenses (GPLv2/commercial).
>>
>> ....Roy
>>
>> -----
>> To unsubscribe, e-mail: community-unsubscribe@apache.org
>> For additional commands, e-mail: community-help@apache.org
>>
>>
>
> -----
> To unsubscribe, e-mail: community-unsubscribe@apache.org
> For additional commands, e-mail: community-help@apache.org
>
--
Brett Porter
brett@apache.org
http://blogs.exist.com/bporter/

```

From jean-frederic clere <jfcl...@apache.org>

Subject Re: Java Service Wrapper

Date Thu, 03 Apr 2008 07:43:33 GMT

On Wed, Apr 02, 2008 at 12:34:47PM -0700, Henri Yandell wrote:

> Raises the question of whether anyone has needed the
> improvements/bugfixes/whatever in the later GPL versions, and whether
> we should consider forking the old version.

Tomcat uses procrun from daemon (<http://commons.apache.org/daemon/>) not the Java Service Wrapper.

As we have most of the features in ASF, shouldn't we try to use daemon instead the Java Service Wrapper?

Cheers

Jean-Frederic

>

> Hen

>

> On Wed, Apr 2, 2008 at 12:20 PM, Roy T. Fielding <fielding@gbiv.com>

```
wrote:
> > The Java Service Wrapper is being used (and sometimes redistributed)
by
> > many of our Java projects (ActiveMQ, ServiceMix, Geronimo, OFBiz,
Tomcat).
> > It was formerly available under an MIT/BSD style license, as can be
seen
> > in the 3.2.3 version at
> >
> > http://sourceforge.net/projects/wrapper/
> >
> > More recent versions of the product are currently under dual GPLv2
> > and commercial licensing.
> >
> > http://wrapper.tanukisoftware.org/doc/english/licenseOverview.html
> >
> > And yet no mention of its license is found in our documentation.
> >
> > http://activemq.apache.org/java-service-wrapper.html
> >
> >
http://docs.ofbiz.org/display/OFBIZ/How+to+Run+OFBiz+as+Windows+Service+with+Java+Service+Wrapper
> > http://people.apache.org/~fhanik/wrapper.html
> > http://wiki.apache.org/db-derby/DerbyWindowsService
> >
> > and the only pages we have that refer to the old version are in
Geronimo
> >
> >
> > http://cwiki.apache.org/GMOxDOC11/configuring-geronimo-as-a-windows-
service.html
> >
> > http://cwiki.apache.org/GMOxDOC20/configuring-geronimo-as-a-windows-
service.html
> >
> > If your project is currently distributing or telling users to make
use of
> > the Java Service Wrapper, please be sure that we only use the old
licensed
> > version, that our documentation points to the old site, and that when
we do
> > point folks to wrapper.tanukisoftware.org (which we should do because
some
> > users are not allergic to GPLv2) we should also point out that the
newer
> > versions are under different licenses (GPLv2/commercial).
> >
> >
> > ....Roy
```

From	Hernan Cunico <hcun...@gmail.com>
Subject	Re: Java Service Wrapper
Date	Thu, 03 Apr 2008 15:02:33 GMT

```
Roy T. Fielding wrote:
> The Java Service Wrapper is being used (and sometimes redistributed) by
> many of our Java projects (ActiveMQ, ServiceMix, Geronimo, OFBiz,
Tomcat).
```

```

> It was formerly available under an MIT/BSD style license, as can be seen
> in the 3.2.3 version at
>
>   http://sourceforge.net/projects/wrapper/
>
> More recent versions of the product are currently under dual GPLv2
> and commercial licensing.
>
>   http://wrapper.tanukisoftware.org/doc/english/licenseOverview.html
>
> And yet no mention of its license is found in our documentation.
>
>   http://activemq.apache.org/java-service-wrapper.html
>
>
http://docs.ofbiz.org/display/OFBIZ/How+to+Run+OFBiz+as+Windows+Service+wi
th+Java+Service+Wrapper
>
>   http://people.apache.org/~fhanik/wrapper.html
>   http://wiki.apache.org/db-derby/DerbyWindowsService
>
> and the only pages we have that refer to the old version are in Geronimo
>
>
> http://cwiki.apache.org/GMOxDOC11/configuring-geronimo-as-a-windows-
service.html
>
>
> http://cwiki.apache.org/GMOxDOC20/configuring-geronimo-as-a-windows-
service.html

add another one on the Geronimo doc still using 3.2.3

http://cwiki.apache.org/GMOxDOC21/configuring-geronimo-as-a-windows-
service.html

Cheers!
Hernan

>
>
> If your project is currently distributing or telling users to make use
of
> the Java Service Wrapper, please be sure that we only use the old
licensed
> version, that our documentation points to the old site, and that when we
do
> point folks to wrapper.tanukisoftware.org (which we should do because
some
> users are not allergic to GPLv2) we should also point out that the newer
> versions are under different licenses (GPLv2/commercial).
>
>
> ....Roy
>

```

From	"Noel J. Bergman" <n...@devtech.com>
Subject	RE: Java Service Wrapper
Date	Thu, 17 Apr 2008 18:44:42 GMT

```
jean-frederic clere wrote:  
> Henri Yandell wrote:  
> > Raises the question of whether anyone has needed the  
> > improvements/bugfixes/whatever in the later GPL versions, and whether  
> > we should consider forking the old version.
```

```
> As [(http://commons.apache.org/daemon/ has] most of the features in ASF,  
> shouldn't we try to use daemon instead the Java Service Wrapper?
```

```
Yes. We should be promoting it instead, and calling for projects to drop  
the Java Service Wrapper. And there is now good reason, as Henri  
suggested,  
to consider forking the previously acceptably licensed version of Wrapper,  
or at least look at incorporating parts of it into daemon to complete the  
feature set.
```

```
--- Noel
```

From	Martin van den Bemt <m...@apache.org>
Subject	Re: Java Service Wrapper
Date	Thu, 17 Apr 2008 21:10:58 GMT

Maven is already discussing forking it..

Mvgr,
Martin

```
Noel J. Bergman wrote:  
> jean-frederic clere wrote:  
>> Henri Yandell wrote:  
>>> Raises the question of whether anyone has needed the  
>>> improvements/bugfixes/whatever in the later GPL versions, and whether  
>>> we should consider forking the old version.  
>  
>> As [(http://commons.apache.org/daemon/ has] most of the features in  
ASF,  
>> shouldn't we try to use daemon instead the Java Service Wrapper?  
>  
> Yes. We should be promoting it instead, and calling for projects to  
drop  
> the Java Service Wrapper. And there is now good reason, as Henri  
suggested,  
> to consider forking the previously acceptably licensed version of  
Wrapper,  
> or at least look at incorporating parts of it into daemon to complete  
the  
> feature set.  
>  
> --- Noel  
>
```

Interação 2.2.APH

From J Aaron Farr <fa...@apache.org>
Subject ApacheCon Evolution/Revolution
Date Tue, 15 Apr 2008 10:16:34 GMT

So, as you can imagine, there are always a lot of ideas being tossed around on how we can improve ApacheCon. Ideas ranging from evolution (have 2 day trainings) to revolution (scrap the whole thing and do barcamps!).

In an effort to better collect these ideas, I and a few other planners have started collecting thoughts on the ApacheCon wiki:

<http://wiki.apache.org/apachecon/ApacheConBrainStorm>

The idea is to improve the thoughts on this page to the point that some of them can be seriously considered and (hopefully) implemented. It is difficult to implement a proposal that is stretched along a 100+ email thread, so that's why I suggest using the wiki. I don't want to discourage discussion on the mailing lists, but I would like to see the results of the discussion presented in a clear manner on some document.

So please offer up your thoughts, criticism and suggestions. Note that this email is going out to both members@ and community@ in an effort to reach everyone.

Thanks!

--

J Aaron Farr jadetower.com [US] +1 724-964-4515
 馮傑仁 cubiclemuses.com [HK] +852 8123-7905

From "William A. Rowe, Jr." <wr...@rowe-clan.net>
Subject Re: ApacheCon Evolution/Revolution
Date Tue, 15 Apr 2008 11:06:41 GMT

David Welton wrote:

>
 > Not having gone to any of them lately, (too much \$\$\$), I don't really
 > have much to say, but if there's to be a revolution, the first thing
 > to go out the window ought to be "training" as a noun.

As much as I agree that this is a perversion of the English language, our users/their employers are paying for "a training [session]" more readily than the old "tutorials" or other ways of phrasing it. E.g. leave it and that part of the program will sell more effectively.

I'd totally agree to leave this abuse out of our non-\$\$\$ conferences :)

Interação 3.3.APH

From "Ted Husted" <hus...@apache.org>
Subject Psychopaths in the software industry
Date Tue, 08 Jan 2008 11:37:30 GMT

In a followup to Zed Shaw's recent Rant on Rails, Rickard Oberg has some salient thoughts on how good communities go bad.

* <http://www.jroller.com/rickard/date/20080106>

-Ted.

From "Bruce Snyder" <bruce.sny...@gmail.com>
Subject Re: Psychopaths in the software industry
Date Tue, 08 Jan 2008 15:24:40 GMT

On Jan 8, 2008 4:37 AM, Ted Husted <husted@apache.org> wrote:
 > In a followup to Zed Shaw's recent Rant on Rails, Rickard Oberg has
 > some salient thoughts on how good communities go bad.
 >
 > * <http://www.jroller.com/rickard/date/20080106>

+1 This is definitely a good read. As always, Rickard's assessment is very good and the book he points out looks fascinating.

Bruce
 --
 perl -e 'print unpack("u30", "D0G)U8V4\@4VYY9&5R\F)R=6-E+G-N>61E<D\!G;6%I;\YC;VT*"
);'

Apache ActiveMQ - <http://activemq.org/>
 Apache Camel - <http://activemq.org/camel/>
 Apache ServiceMix - <http://servicemix.org/>
 Apache Geronimo - <http://geronimo.apache.org/>

Blog: <http://bruceblog.org/>

From "Trustin Lee" <trus...@gmail.com>
Subject Re: Psychopaths in the software industry
Date Thu, 10 Jan 2008 01:12:42 GMT

On Jan 8, 2008 8:37 PM, Ted Husted <husted@apache.org> wrote:
 > In a followup to Zed Shaw's recent Rant on Rails, Rickard Oberg has
 > some salient thoughts on how good communities go bad.
 >
 > * <http://www.jroller.com/rickard/date/20080106>

Thanks for the nice read. And here's the follow-up post, which is easier to read IMO:

http://www.jroller.com/rickard/entry/comments_on_psychopaths_in_the

Trustin
 --

```
what we call human nature is actually human habit
--
http://gleamynode.net/
--
PGP Key ID: 0x0255ECA6
```


Anexo H – Discussões da Comunidade MySQL

Interação 1.4.SQL

Configuring MySql for Access

Posted by: [Allison Loebs](#) ()
Date: February 24, 2009 05:41AM

Hi,

I have installed MySql 5.1.31 and find I have two databasea test and information_schema. I can logon to test and there are no tables. I cannot logon to information_schema access denied for @localhost.

Reading documentation I need to set up a user table. Where do I do this as I do not have a MySql database just information_schema adn test.

Thank You

Re: Configuring MySql for Access

Posted by: [Jay Alverson](#) ()
Date: February 24, 2009 08:18AM

What client are you using to access the database ?

if it's MYSQL.exe then try:

```
mysql> select user();
+-----+
| user() |
+-----+
| Winbatch@localhost |
+-----+
1 row in set (0.00 sec)
```

mysql>

to see what account you're logged in under.

>

Thanks, Jay

Re: Configuring MySql for Access

Posted by: [Allison Loebs](#) ()
Date: February 25, 2009 02:56AM

Hi,

I am using mysql.exe and shows user of ODBC@Loclahost. I was expecting to see a physical table named user on the database and was curious when I don't see it under the test or information_schema DB. I have no default MySql Database and wondering if I should have database named MySql.

Thank You

Re: Configuring MySql for Access

Posted by: [Jay Alverson](#) ()

Date: February 25, 2009 08:05AM

Try login in as "root"

>

Thanks, Jay

Re: Configuring MySql for Access

Posted by: [Allison Loeb](#) ()

Date: March 01, 2009 07:00AM

Hi,

Thanks forgot to write back that I figured out I wasn't logged in fully. That worked.

Interação 2.4.SQL

Removing Mysql and installing it again?!?!

Posted by: [Desimira Videva](#) ()

Date: February 16, 2009 09:03AM

Ok, I installed mysql 5.1 on my windows XP, but i wanted to reinstall it, so i went to Add/Remove programs and remove it from there, then i went to c:\program files and i deleted the folder mysql then i read in one forum that i should go to "start->run->type regedit->HKEY_LOCAL_MACHINE->Services->Mysql(and delete the folder from there too)"

I did it too.

No one of these things helped me, when i start installing mysql again, it's asking me for

Current password:

New Password:

Confirm New Password:

So i guess i have to do something else, but i have no idea what , so i will be really thankful if i get any help from here.

I want to install mysql again. What should i do?

Re: Removing Mysql and installing it again?!?!

Posted by: [Jay Alverson](#) ()

Date: February 16, 2009 09:37AM

Check for:

"C:\Documents and Settings\All Users\Application Data\MySQL"

>

Thanks, Jay

Removing Mysql and tring to install again (Windows Vista)

Posted by: [ron vanderveen](#) ()

Date: February 25, 2009 09:05PM

I've had to remove Mysql and then I'm trying to reinstall. I'm having no luck.

I've tried deleting install directory - it doesn't help. When I reinstall, I get through the Install Wizard about half way. At the panel where it allows to 'configure server now', I click on check box. I then click on Finish button and that's when wizard exits. I don't get a chance to configure the server (dedicated port, set password, etc).

I'm using Window Vista and want to install MySql 5.1

Any suggestions?

Ron

Re: Removing Mysql and installing it again?!?!

Posted by: [Desimira Videva](#) ()
Date: February 17, 2009 09:07AM

Oh, great! Thank you very much!

It worked!

And I actually paid attention when I was installing it again, it said in which folders is putting mysql, i guess i had to pay attention to that before, too :)

Edited 1 time(s). Last edit at 02/17/2009 09:09AM by Desimira Videva.

Re: Removing Mysql and installing it again?!?!

Posted by: [Jay Alverson](#) ()
Date: February 17, 2009 01:53PM

Don't feel bad, often times on windows things are 100+ character-strings zipping past your eyes at 5 Ghz.

My favorites are the installs (not MySQL) that takes 5 minutes to go from 0-99% then take another 5 minutes to complete the last 1% :) :)

>

Anexo I – Discussões da Comunidade PHP

Interação 1.9.PHP

From:	Exception e	Date:	Wed Dec 24 17:45:23 2008
Subject:	atomicity/thread safeness of apc shared memory functions + mutexes		
Groups:	php.apc.dev		

Couldn't find it in the docs, so I hope you know it:

Are the userland apc cache functions (apc_add, apc_fetch, apc_delete etc) *thread-safe/atomic*?

Also: I need to ensure that concurrent access to shared memory doesn't get me in race conditions. Wouldn't it be nice if apc could offer mutex functions and the like? There is a semaphore extension for php, but it is *nix-only according to the docs. The need for synchronization mechanisms arise almost always when using shared memory.

From:	Rasmus Lerdorf	Date:	Wed Dec 24 18:26:50 2008
Subject:	Re: [APC-DEV] atomicity/thread safeness of apc shared memory functions + mutexes		
References:	1	Groups:	php.apc.dev

Exception e wrote:

```
> Couldn't find it in the docs, so I hope you know it:
>
> Are the userland apc cache functions (apc_add, apc_fetch, apc_delete
> etc) *thread-safe/atomic*?
>
>
> Also: I need to ensure that concurrent access to shared memory doesn't
> get me in race conditions. Wouldn't it be nice if apc could offer mutex
> functions and the like? There is a semaphore extension for php, but it
> is *nix-only according to the docs. The need for synchronization
> mechanisms arise almost always when using shared memory.
```

Yes, they are atomic and they are locked appropriately. You don't need any sort of mutex. This is shared memory. Generally we don't suggest running PHP in any sort of threaded environment though.

-Rasmus

From:	Exception e	Date:	Wed Dec 24 18:47:37 2008
Subject:	Re: [APC-DEV] atomicity/thread safeness of apc shared memory functions+ mutexes		

References: [1](#) [2](#)**Groups:** [php.apc.dev](#)

Rasmus Lerdorf schreef:

```
> Yes, they are atomic and they are locked appropriately. You don't need
> any sort of mutex. This is shared memory. Generally we don't suggest
> running PHP in any sort of threaded environment though.
```

To respond to your last remark: php processes can be executed in parallel by Apache at least this is my understanding. So if I need to work with multiple shared memory entries per request i still need mutexes in order to lock concurrent access to these pieces.

With a web server you don't have a guarantee that requests are all handled sequentially. If I am wrong, Im happy to be corrected!

From: [Rasmus Lerdorf](#)**Date:** Wed Dec 24 19:26:20 2008**Subject:** Re: [APC-DEV] atomicity/thread safeness of apc shared memory functions+mutexes**References:** [1](#) [2](#) [3](#)**Groups:** [php.apc.dev](#)

Exception e wrote:

```
> Rasmus Lerdorf schreef:
```

```
>
```

```
>> Yes, they are atomic and they are locked appropriately. You don't need
>> any sort of mutex. This is shared memory. Generally we don't suggest
>> running PHP in any sort of threaded environment though.
```

```
>
```

```
> To respond to your last remark: php processes can be executed in
> parallel by Apache at least this is my understanding. So if I need to
> work with multiple shared memory entries per request i still need
> mutexes in order to lock concurrent access to these pieces.
```

That's not multi-threading, that is multi-processing, and of course APC handles that correctly and locks appropriately. Multi-threading is when you have multiple threads of execution within the same process.

-Rasmus

From: [Exception e](#)**Date:** Thu Dec 25 06:35:40 2008**Subject:** Re: [APC-DEV] atomicity/thread safeness of apc shared memory functions+mutexes**References:** [1](#) [2](#) [3](#) [4](#)**Groups:** [php.apc.dev](#)

Rasmus Lerdorf schreef:

```
> Exception e wrote:
```

```
>> Rasmus Lerdorf schreef:
```

```
>>
```

```
>>> Yes, they are atomic and they are locked appropriately. You don't need
>>> any sort of mutex. This is shared memory. Generally we don't suggest
>>> running PHP in any sort of threaded environment though.
```

```
>> To respond to your last remark: php processes can be executed in
```

```
>> parallel by Apache at least this is my understanding. So if I need to
>> work with multiple shared memory entries per request i still need
>> mutexes in order to lock concurrent access to these pieces.
```

```
>
> That's not multi-threading, that is multi-processing, and of course APC
> handles that correctly and locks appropriately. Multi-threading is when
> you have multiple threads of execution within the same process.
>
```

Yes, but do you mean that all shared memory entries are blocked for the duration of a process? Let's give me an example

```
<?php
$val1 = apc_fetch('val1');

// (#1) different process does: apc_store('val1', 34);

$val2 = apc_fetch('val2');

echo $val1 + $val2;

?>
```

From what I understand is that when the same code runs in a different process, the shared memory segment 'val1' can be changed on (#1) by that other process. So in fact I need two mutexes.

But if all shared memory is locked for each process, than such a situation cannot happen. Did you really mean that? I hope you could clarify this.

From:	Rasmus Lerdorf	Date:	Thu Dec 25 08:41:59 2008
Subject:	Re: [APC-DEV] atomicity/thread safeness of apc shared memory functions+mutexes		
References:	1 2 3 4 5	Groups:	php.apc.dev

Exception e wrote:

```
> Rasmus Lerdorf schreef:
>> Exception e wrote:
>>> Rasmus Lerdorf schreef:
>>>>
>>>> Yes, they are atomic and they are locked appropriately. You don't
>>>> need
>>>> any sort of mutex. This is shared memory. Generally we don't suggest
>>>> running PHP in any sort of threaded environment though.
>>> To respond to your last remark: php processes can be executed in
>>> parallel by Apache at least this is my understanding. So if I need to
>>> work with multiple shared memory entries per request i still need
>>> mutexes in order to lock concurrent access to these pieces.
>>
>> That's not multi-threading, that is multi-processing, and of course APC
>> handles that correctly and locks appropriately. Multi-threading is when
>> you have multiple threads of execution within the same process.
>>
>
>
> Yes, but do you mean that all shared memory entries are blocked for the
> duration of a process? Let's give me an example
>
```

```

> <?php
>
> $val1 = apc_fetch('val1');
>
> // (#1) different process does: apc_store('val1', 34);
>
> $val2 = apc_fetch('val2');
>
> echo $val1 + $val2;
>
> ?>
>
>
> From what I understand is that when the same code runs in a different
> process, the shared memory segment 'val1' can be changed on (#1) by that
> other process. So in fact I need two mutexes.
>
> But if all shared memory is locked for each process, than such a
> situation cannot happen. Did you really mean that? I hope you could
> clarify this.

```

apc_fetch() grabs a copy of the value in shared memory. What you do with it locally in the process after the fetch is irrelevant. It is just a copy. You can then overwrite the value in shared memory with a new version with an apc_store() call. Locking shared memory the way you suggest would be a really bad idea. It would be a performance nightmare. You need to design your code in a way that does not require that.

APC 3.1 has functions that let you atomically increment and decrement a value in shared memory. See the apc_inc() and apc_dec() functions. That is the only way that makes sense to manipulate shared memory entries in a way resembling what you are suggesting.

-Rasmus

From:	Exception e	Date:	Thu Dec 25 10:25:24 2008
Subject:	Re: [APC-DEV] atomicity/thread safeness of apc shared memory functions+mutexes		
References:	1 2 3 4 5 6	Groups:	php.apc.dev

```

> apc_fetch() grabs a copy of the value in shared memory. What you do
> with it locally in the process after the fetch is irrelevant. It is
> just a copy. You can then overwrite the value in shared memory with a
> new version with an apc_store() call. Locking shared memory the way you
> suggest would be a really bad idea. It would be a performance
> nightmare. You need to design your code in a way that does not require
> that.
>

```

Locking all shared memory would indeed be bad, I was already wondering whether you really meant that. But I use php for a multi-user game as part of an academic investigation. To speed things up I store(=serialize) objects in shared memory. I need to ensure that certain objects in shared memory keep a coherent state. That is, when I use object1 and object2 they should not be modified during a request. So then I need to lock those two entries. Atomic reads and writes won't

help here.

But since the apc functions are themselves atomic, I could utilize them for implementing mutexes. A simple example would be

```
function lock($entry) {
    // busy wait
    while(!apc_add($entry.'.lock'))
        sleep(1);
}

function unlock($entry) {
    return apc_delete($entry.'.lock');
}

lock('obj1');
lock('obj2');
$obj1= apc_fetch('obj1');
$obj2 = apc_fetch('obj2');

//process, apc_store val1 and val2 again

unlock('obj1');
unlock('obj2');
```

Then it would only be nice if lock and unlock would be part of apc_* userland functions. I am curious about your ideas.

From:	Rasmus Lerdorf	Date:	Thu Dec 25 12:31:56 2008
Subject:	Re: [APC-DEV] atomicity/thread safeness of apc shared memory functions+mutexes		
References:	1 2 3 4 5 6 7	Groups:	php.apc.dev

Exception e wrote:

```
>
>> apc_fetch() grabs a copy of the value in shared memory. What you do
>> with it locally in the process after the fetch is irrelevant. It is
>> just a copy. You can then overwrite the value in shared memory with a
>> new version with an apc_store() call. Locking shared memory the way you
>> suggest would be a really bad idea. It would be a performance
>> nightmare. You need to design your code in a way that does not require
>> that.
>>
>
> Locking all shared memory would indeed be bad, I was already wondering
> whether you really meant that. But I use php for a multi-user game as
> part of an academic investigation. To speed things up I
> store(=serialize) objects in shared memory. I need to ensure that
> certain objects in shared memory keep a coherent state. That is, when I
> use object1 and object2 they should not be modified during a request. So
> then I need to lock those two entries. Atomic reads and writes won't
> help here.
> But since the apc functions are themselves atomic, I could utilize them
> for implementing mutexes. A simple example would be
>
> function lock($entry) {
>     // busy wait
>     while(!apc_add($entry.'.lock'))
>         sleep(1);
```



```

> }
>
> function unlock($entry)    {
>     return apc_delete($entry.'.lock');
> }
>
> lock('obj1');
> lock('obj2');
> $obj1= apc_fetch('obj1');
> $obj2 = apc_fetch('obj2');
>
> //process, apc_store val1 and val2 again
>
> unlock('obj1');
> unlock('obj2');
>
>
> Then it would only be nice if lock and unlock would be part of apc_*
> userland functions. I am curious about your ideas.

```

You could do something like that, but it would be amazingly inefficient. and can't possibly scale to multiple machines behind a load balancer. Also, storing objects in shared memory is slow and usually pointless. Only the properties of the object are stored anyway, so you may as well store an array directly. That has the benefit of not needing serialization.

-Rasmus

From:	Exception e	Date:	Thu Dec 25 12:55:41 2008
Subject:	Re: [APC-DEV] atomaticity/thread safeness of apc shared memory functions+mutexes		
References:	1 2 3 4 5 6 7 8	Groups:	php.apc.dev

Rasmus Lerdorf schreef:

```

>> function lock($entry)    {
>>     // busy wait
>>     while(!apc_add($entry.'.lock'))
>>         sleep(1);
>> }
>>
>> function unlock($entry)  {
>>     return apc_delete($entry.'.lock');
>> }
>>
>> lock('obj1');
>> lock('obj2');
>> $obj1= apc_fetch('obj1');
>> $obj2 = apc_fetch('obj2');
>>
>> //process, apc_store val1 and val2 again
>>
>> unlock('obj1');
>> unlock('obj2');
>>
>>
>> Then it would only be nice if lock and unlock would be part of apc_*
>> userland functions. I am curious about your ideas.

```

>
 > You could do something like that, but it would be amazingly inefficient.
 Because of the busy wait or is access to shared memory slow?

> and can't possibly scale to multiple machines behind a load balancer.

I also think that a load balancer will harm you. Then you need something like memcache. For this experimentation I don't need to bother about this.

> Also, storing objects in shared memory is slow and usually pointless.
 > Only the properties of the object are stored anyway, so you may as well
 > store an array directly. That has the benefit of not needing
 serialization.

Yes, but I initially thought that the only alternative (database with lockings) would be way slower. If this is not the case I would happily abandon apc. My assumption was that shared memory access would be blazingly fast.

Interação 2.12.PHP

From: shire	Date: Sat Feb 21 19:10:38 2009
Subject: [RFC] APC/PHP Lazy Loading	
Groups: php.apc.dev php.internals	

I've just checked into APC CVS preliminary support for Lazy Loading classes and functions. This means that rather than copying function entries into EG(function_table) and EG(class_table) when an include happen it will mark the functions/classes as available and only actually insert them into the tables when they are called. This is done via hooks added into the various hash table lookups in PHP. I've placed a patch for PHP_5_3 at:

http://tekrat.com/downloads/bits/apc_lazy_php53.patch

You should be able to apply the above patch to the latest PHP_5_3 branch, and recompile the latest APC CVS against it. Two ini settings enable/display lazy loading:

```
apc.lazy_functions=On/Off
apc.lazy_classes=On/OFF
```

There's still some enhancements that I need to make in both PHP and APC for cleaner code and optimizations, but I wanted to get some early feedback to track down issues. In a final version I'd prefer the above patch to abstract all function lookups into a common function. I would love to hear about success/problems with this, as well as performance results or other suggestions. This was initially implement for Facebook's codebase, and dropped CPU usage by about 30%, today it's required for the site to operate normally. Although I expect smaller gains than this in other codebases (wordpress looks to be about 3%, Joomla and Zend Framework use autoloading and appear to get no visible gains).

Alternative implementations would include replacing the function entries with an internal function that would load the opcodes when called, however I found this implementation to be problematic, still requires changes to

PHP, and would also require inserting entries into the function/class tables which itself ends up being an expensive task for extremely large codebases.

I look forward to hearing your feedback,

From:	Ronald Chmara	Date:	Sun Feb 22 00:54:30 2009
Subject:	Re: APC/PHP Lazy Loading		
References:	1	Groups:	php.apc.dev php.internals

On Feb 21, 2009, at 4:10 PM, shire wrote:

```
> I've just checked into APC CVS preliminary support for Lazy Loading
> classes and functions. This means that rather than copying
> function entries into EG(function_table) and EG(class_table) when
> an include happen it will mark the functions/classes as available
> and only actually insert them into the tables when they are called.
```

Wait... so if I understand this right, let's envision a code base where, per some random page load, 70 functions are actually called, but, oh, 7,000, or even 700,000, are being included for whatever reason?

The speed optimization is in **not** copying a massive amount of things that weren't even needed, or used, in the first place?

However, there's still the horribly massive speed hit of semi-loading, and marking, a fairly large amount of unused, un-needed, functions, as available?

While I do see, and understand, the performance benefit of such a coding direction, it's not exactly encouraging well designed code.

For example:

```
<?php
include(mega-loader.php);
//which kicks off
->include(site-mega-loader.php);
//which kicks off
-->include(sub-site-mega-loader.php);
//which kicks off
--->include(sub-site-application-loader.php);
//which kicks off
---->include(sub-site-application-feature-loader.php);
//which kicks off
----->include(sub-site-application-function-loader.php);
//which kicks off
----->include(sub-site-application-function-loader-function.php);

//and the actual code which caused the load?
my_custom_site_application_echo( "Hello World");

?>
```

This.... does not make me happy as a coder.

I do see the benefit of lazy loading, I'm just not very comfortable with enabling a philosophy of loading up a massive amount of CPU and RAM with "just in case they're wanted" features and code in the first place.

It *should* hurt to load something.

Code should not be ever loaded unless needed.

It certainly can boost an APC code set such as facebook, where many of those files and functions *will* likely be used in the next 20 minutes or so, but I also fear that it will encourage programmers to load everything they have, every time, just in case they need it.... and 2Gb apache processes (and APC space) can be.... ugly.

That being said, I think the idea is good, but the possible mis-uses are potentially tough issues.

-Bop

From: shire	Date: Sun Feb 22 01:55:18 2009
Subject: Re: APC/PHP Lazy Loading	
References: 1 2	Groups: php.apc.dev php.internals

Hi Ronald,

Ronald Chmara wrote:

```
> Wait... so if I understand this right, let's envision a code base where,
> per some random page load, 70 functions are actually called, but, oh,
> 7,000, or even 700,000, are being included for whatever reason?
>
> The speed optimization is in not copying a massive amount of things
> that weren't even needed, or used, in the first place?
```

Essentially, yes, this is probably best summed up by the 80/20 rule where we only use 20% of the code etc...

```
> However, there's still the horribly massive speed hit of semi-loading,
> and marking, a fairly large amount of unused, un-needed, functions, as
> available?
```

I don't agree with the description of describing this as a "horribly massive speed hit" at least in comparison with what was happening without lazy loading. Also, like I said there's further iterations I plan to make here, one of these being increasing the performance of this marking functions as available.

```
> I do see the benefit of lazy loading, I'm just not very comfortable with
> enabling a philosophy of loading up a massive amount of CPU and RAM with
> "just in case they're wanted" features and code in the first place.
```

Well I am assuming that this is what a large amount of code does already, except that without lazy loading the situation is significantly worse. Your point that we should be sure this does not encourage poor coding practices is well taken, but it's been my experience that code tends to take this form regardless so I'm hoping to make the best of the situation ;-).

Also keep in mind that there are cases where you may not know in advance which functions you will/will not call, but it's probably fair to say that

the 80/20 rule still holds, so including all the functions you may need is not particularly a misuse of the language, but rather a necessity of a dynamic application and language.

```
> It certainly can boost an APC code set such as facebook, where many of
> those files and functions *will* likely be used in the next 20 minutes
> or so, but I also fear that it will encourage programmers to load
> everything they have, every time, just in case they need it.... and 2Gb
> apache processes (and APC space) can be.... ugly.
```

I'm not entirely clear on where code being used in the next 20 minutes come into play, what differenc does 100 milliseconds vs. 20 minutes make in APC/lazy loading? It's actually likely that only a fraction of the code at Facebook will be used in a request, hence the need for lazy loading. But we need to make all the functions available due to the dynamic nature of the site and PHP itself as we don't know ahead of time which functions we will need to call.

Thanks for the feedback, and hopefully the above makes sense. I don't want to encourage bad progarmming form and would definitely encourage avoiding this situation if possible, however I think many applications may find this optimization a necessity.

-shire

From: Mikko Koppanen	Date: Sun Feb 22 11:36:50 2009
Subject: Re: APC/PHP Lazy Loading	
References: 1 2 3	Groups: php.apc.dev php.internals

On Sun, Feb 22, 2009 at 6:55 AM, shire <shire@tekrat.com> wrote:

```
>
> Thanks for the feedback, and hopefully the above makes sense. I don't
> want
> to encourage bad progarmming form and would definitely encourage avoiding
> this situation if possible, however I think many applications may find
> this
> optimization a necessity.
>
```

Hello,

I don't think this patch encourages bad programming any way. It just gives more flexibility for the sites that need to focus on performance. I haven't had the chance to check the patch yet but as a principle this is something I would like to see in future versions of PHP / APC.

Good work!

--
Mikko Koppanen

From: Ronald Chmara	Date: Mon Feb 23 02:47:50 2009
--	---------------------------------------

Subject: Re: APC/PHP Lazy Loading

References: [1](#) [2](#) [3](#)

Groups: [php.apc.dev](#) [php.internals](#)

On Feb 21, 2009, at 10:55 PM, shire wrote:

```
> Hi Ronald,
> Ronald Chmara wrote:
>> Wait... so if I understand this right, let's envision a code base
>> where,
>> per some random page load, 70 functions are actually called, but, oh,
>> 7,000, or even 700,000, are being included for whatever reason?
>> The speed optimization is in *not* copying a massive amount of things
>> that weren't even needed, or used, in the first place?
> Essentially, yes, this is probably best summed up by the 80/20 rule
> where we only use 20% of the code etc...
```

Well, I can see 80% actually *used* code, with 20% in there by accident.... but 80% unused code? eep! ack! Call the villagers and get the torches and pitchforks!...

...but environments vary, of course. ;)

```
>> However, there's still the horribly massive speed hit of semi-
>> loading,
>> and marking, a fairly large amount of unused, un-needed,
>> functions, as
>> available?
> I don't agree with the description of describing this as a
> "horribly massive speed hit" at least in comparison with what was
> happening without lazy loading.
```

Fair enough. Before the patch, for example, I might describe it (80% unused, 20% used code) as an "insanely awful, horribly massive speed hit", and after the patch, as being reduced to a much lesser "horribly massive speed hit", but these are just rhetorical, and qualitative, language devices that I used to characterize code issues.

In both cases, a large amount of CPU is spent on (effectively) doing nothing, but your patch (as I understand its design) reduces the amount of CPU waste... doing nothing.

```
> Also, like I said there's further iterations I plan to make here,
> one of these being increasing the performance of this marking
> functions as available.
```

One thing I see as quite a beneficial future outcome of your work is the ability to further profile code, and be able to seek out code that marks massive amounts of functions as "available".... without actually ever using them.

```
>> I do see the benefit of lazy loading, I'm just not very
>> comfortable with
>> enabling a philosophy of loading up a massive amount of CPU and
>> RAM with
>> "just in case they're wanted" features and code in the first place.
> Well I am assuming that this is what a large amount of code does
> already, except that without lazy loading the situation is
> significantly worse.
```

Different code bases and philosophies vary.

Since much of what I do (enterprise PHP tuning) involves (among many other things) finding, and eliminating, such code, I can say with great confidence that there certainly are bloat-bases out there that load metric hogs-heads of libraries to show a single web email form, but there are also code bases which do **not** rely on endless libraries, frameworks, additional template abstractions and end user libraries, or other pre-determined architectures... to complete the simple task of showing an web email form.

To frame the issue another way, you are trying to make huge, complicated, code sets less painful to use, and I am arguing that huge, complicated, code sets are a major part of the problem..... but since neither of us can wave a magic wand and reduce the problem to simple, elegant, code sets, you're reducing the magnitude of pain involved. Kudos to you.

```
> Your point that we should be sure this does not encourage poor
> coding practices is well taken, but it's been my experience that
> code tends to take this form regardless so I'm hoping to make the
> best of the situation ;-).
```

There will always be bad code, yes. ;-)

I'm trying to raise a token flag of discussion/resistance to making bad code practices less painful, as it still enables bad code practices.

```
> Also keep in mind that there are cases where you may not know in
> advance which functions you will/will not call, but it's probably
> fair to say that the 80/20 rule still holds, so including all the
> functions you may need is not particularly a misuse of the
> language, but rather a necessity of a dynamic application and
> language.
```

It all depends on the use, and environment, I suppose.

```
>> It certainly can boost an APC code set such as facebook, where
>> many of
>> those files and functions *will* likely be used in the next 20
>> minutes
>> or so, but I also fear that it will encourage programmers to load
>> everything they have, every time, just in case they need it....
>> and 2Gb
>> apache processes (and APC space) can be.... ugly.
> I'm not entirely clear on where code being used in the next 20
> minutes come into play, what differenc does 100 milliseconds vs. 20
> minutes make in APC/lazy loading?
```

FB seems to have a fair bit of traffic, with a semi-patrolled code set, so it's likely that any single APC-loaded function will be invoked **eventually**, within an hour or so.

Contrast this with 1,000 different sites hosted on a box, using a less patrolled, fairly unregulated, 1,000 different batches of PHP codesets, where "myTotallyCustomDateTime()" can have 1,000 different variants, some of which are only actually used once every 3-5 weeks or so.

Lazy loading would (as I understand it) speed up both, but lazy loading would also encourage not just one code set, but all code sets, to assume that the **language authors**, rather than the **developers**, were responsible for making sure CPU was being managed

efficiently.

```
> It's actually likely that only a fraction of the code at Facebook
> will be used in a request, hence the need for lazy loading.
```

Ouch. Seriously.

I can't tell you how to build your code, but I think you might seriously benefit from:

- a) Lazy Loading (as you've done, great idea)
- b) Using Lazy Loading to find out which apps/code are sucking up massive CPU, and taking action as need to help tune, or remove, the offending code?
- c) Breaking your Lazy Loading targets out, to where (a hypothetical) the `mytzyplk_scramble()` function/class method is only included and thus loaded as needed, rather than (as a guess) a function group (or class method group) auto-loaded which may/may not be needed for a given page load?

Does that make sense? Or did you try it already? :)

From: shire	Date: Mon Feb 23 14:19:55 2009
Subject: Re: APC/PHP Lazy Loading	
References: 1 2 3 4	Groups: php.apc.dev php.internals

Ronald Chmara wrote:

```
>
> On Feb 21, 2009, at 10:55 PM, shire wrote:
>> Hi Ronald,
>> Ronald Chmara wrote:
>>> Wait... so if I understand this right, let's envision a code base
where,
>>> per some random page load, 70 functions are actually called, but, oh,
>>> 7,000, or even 700,000, are being included for whatever reason?
>>> The speed optimization is in *not* copying a massive amount of things
>>> that weren't even needed, or used, in the first place?
>> Essentially, yes, this is probably best summed up by the 80/20 rule
>> where we only use 20% of the code etc...
>
> Well, I can see 80% actually *used* code, with 20% in there by
> accident.... but 80% unused code? eep! ack! Call the villagers and get
> the torches and pitchforks!...
```

I should probably be more specific, it's not that you *don't* use the other 80% it's that you use 20% of the code 80% of the time:

http://en.wikipedia.org/wiki/Pareto_principle

Of course I'm abusing this slightly, but this is the *basic* idea, and it's not based on actual usage statistics. In reality you probably use all 100% of your code in smaller chunks depending on the request. As an illustrative example, if you have HTTP requests for setting, retrieving, deleting items in a database which is supported by corresponding functions. In each request you'll only use one of these functions (30%), but you'll still need to load the entire file/class. The goal of lazy loading is to optimize this and similar situations so you don't have to re-organize your

code into unfeasibly small files, or in other ways that might inhibit productivity.

> One thing I see as quite a beneficial future outcome of your work is the
> ability to further profile code, and be able to seek out code that marks
> massive amounts of functions as "available".... without actually ever
> using them.

I think this would be better instrumented through tools actually designed to do this sort of profiling, specifically XDebug, or tools like included are very useful.

<http://pecl.php.net/package/xdebug>
<http://pecl.php.net/package/included>

>> It's actually likely that only a fraction of the code at Facebook will
>> be used in a request, hence the need for lazy loading.
>
> Ouch. Seriously.
>
> I can't tell you how to build your code, but I think you might seriously
> benefit from:
>....
> Does that make sense? Or did you try it already? :)

I'd rather not enter into a public discussion of Facebook's current optimization tactics outside of this patch, but suffice to say we have implemented many optimization techniques, both in PHP code and Internals, many of which we have and will continue to contribute back to the community. Lazy Loading has been a very necessary and optimal solution to this one aspect of our scalability, thus allowing our engineers to focus more of their time and energy on creating new and exciting features for Facebook.

Thanks,

-shire

From:	Rodrigo Saboya	Date:	Thu Feb 26 08:58:43 2009
Subject:	Re: APC/PHP Lazy Loading		
References:	1 2 3 4	Groups:	php.apc.dev php.internals

Ronald Chmara wrote:

>
> On Feb 21, 2009, at 10:55 PM, shire wrote:
>> Hi Ronald,
>> Ronald Chmara wrote:
>>> Wait... so if I understand this right, let's envision a code base
where,
>>> per some random page load, 70 functions are actually called, but, oh,
>>> 7,000, or even 700,000, are being included for whatever reason?
>>> The speed optimization is in *not* copying a massive amount of things
>>> that weren't even needed, or used, in the first place?
>> Essentially, yes, this is probably best summed up by the 80/20 rule
>> where we only use 20% of the code etc...

```
>
> Well, I can see 80% actually *used* code, with 20% in there by
> accident.... but 80% unused code? eep! ack! Call the villagers and get
> the torches and pitchforks!...
>
> ...but environments vary, of course. ;)
>
>>> However, there's still the horribly massive speed hit of semi-loading,
>>> and marking, a fairly large amount of unused, un-needed, functions, as
>>> available?
>> I don't agree with the description of describing this as a "horribly
>> massive speed hit" at least in comparison with what was happening
>> without lazy loading.
>
> Fair enough. Before the patch, for example, I might describe it (80%
> unused, 20% used code) as an "insanely awful, horribly massive speed
> hit", and after the patch, as being reduced to a much lesser "horribly
> massive speed hit", but these are just rhetorical, and qualitative,
> language devices that I used to characterize code issues.
>
> In both cases, a large amount of CPU is spent on (effectively) doing
> nothing, but your patch (as I understand its design) reduces the amount
> of CPU waste... doing nothing.
>
>> Also, like I said there's further iterations I plan to make here, one
>> of these being increasing the performance of this marking functions as
>> available.
>
> One thing I see as quite a beneficial future outcome of your work is the
> ability to further profile code, and be able to seek out code that marks
> massive amounts of functions as "available".... without actually ever
> using them.
>
>>> I do see the benefit of lazy loading, I'm just not very comfortable
with
>>> enabling a philosophy of loading up a massive amount of CPU and RAM
with
>>> "just in case they're wanted" features and code in the first place.
>> Well I am assuming that this is what a large amount of code does
>> already, except that without lazy loading the situation is
>> significantly worse.
>
> Different code bases and philosophies vary.
>
> Since much of what I do (enterprise PHP tuning) involves (among many
> other things) finding, and eliminating, such code, I can say with great
> confidence that there certainly are bloat-bases out there that load
> metric hogs-heads of libraries to show a single web email form, but
> there are also code bases which do *not* rely on endless libraries,
> frameworks, additional template abstractions and end user libraries, or
> other pre-determined architectures... to complete the simple task of
> showing an web email form.
>
> To frame the issue another way, you are trying to make huge,
> complicated, code sets less painful to use, and I am arguing that huge,
> complicated, code sets are a major part of the problem..... but since
> neither of us can wave a magic wand and reduce the problem to simple,
> elegant, code sets, you're reducing the magnitude of pain involved.
> Kudos to you.
>
>> Your point that we should be sure this does not encourage poor coding
```

```
>> practices is well taken, but it's been my experience that code tends
>> to take this form regardless so I'm hoping to make the best of the
>> situation ;-).
>
> There will always be bad code, yes. ;-).
>
> I'm trying to raise a token flag of discussion/resistance to making bad
> code practices less painful, as it still enables bad code practices.
>
>> Also keep in mind that there are cases where you may not know in
>> advance which functions you will/will not call, but it's probably fair
>> to say that the 80/20 rule still holds, so including all the functions
>> you may need is not particularly a misuse of the language, but rather
>> a necessity of a dynamic application and language.
>
> It all depends on the use, and environment, I suppose.
>
>>> It certainly can boost an APC code set such as facebook, where many of
>>> those files and functions *will* likely be used in the next 20 minutes
>>> or so, but I also fear that it will encourage programmers to load
>>> everything they have, every time, just in case they need it.... and 2Gb
>>> apache processes (and APC space) can be.... ugly.
>> I'm not entirely clear on where code being used in the next 20 minutes
>> come into play, what differenc does 100 milliseconds vs. 20 minutes
>> make in APC/lazy loading?
>
> FB seems to have a fair bit of traffic, with a semi-patrolled code set,
> so it's likely that any single APC-loaded function will be invoked
> *eventually*, within an hour or so.
>
> Contrast this with 1,000 different sites hosted on a box, using a less
> patrolled, fairly unregulated, 1,000 different batches of PHP codesets,
> where "myTotallyCustomDateTime()" can have 1,000 different variants,
> some of which are only actually used once every 3-5 weeks or so.
>
> Lazy loading would (as I understand it) speed up both, but lazy loading
> would also encourage not just one code set, but all code sets, to assume
> that the *language authors*, rather than the *developers*, were
> responsible for making sure CPU was being managed efficiently.
>
>> It's actually likely that only a fraction of the code at Facebook
>> will be used in a request, hence the need for lazy loading.
>
> Ouch. Seriously.
>
> I can't tell you how to build your code, but I think you might seriously
> benefit from:
> a) Lazy Loading (as you've done, great idea)
> b) Using Lazy Loading to find out which apps/code are sucking up massive
> CPU, and taking action as need to help tune, or remove, the offending
code?
> c) Breaking your Lazy Loading targets out, to where (a hypothetical) the
> mytzyplk_scramble() function/class method is only included and thus
> loaded as needed, rather than (as a guess) a function group (or class
> method group) auto-loaded which may/may not be needed for a given page
> load?
>
> Does that make sense? Or did you try it already? :)
>
> -Bop
```

Hi,

Ronald, I think you are overreacting a little bit. It may be that proper written could would get no benefit from this patch since it would not load unneeded code and this patch ends up speeding up environments where such "correct" loading isn't done. I don't think that's a reason to disqualify a feature that brings benefits with no significant drawbacks.

For the average PHP programmer, the language will simply "get faster". That can't be bad in any way. It doesn't encourage you to write bad code, it just doesn't kick you in the nuts when you do.

Regards
Rodtigo Saboya

From: Robin Burchell	Date: Thu Feb 26 09:07:09 2009
Subject: Re: APC/PHP Lazy Loading	
References: 1 2 3 4 5	Groups: php.apc.dev php.internals

On Thu, Feb 26, 2009 at 1:58 PM, Rodrigo Saboya
<rodrigo.saboya@bolsademulher.com> wrote:
<snip>

```
> For the average PHP programmer, the language will simply "get faster".
> That
> can't be bad in any way. It doesn't encourage you to write bad code, it
> just
> doesn't kick you in the nuts when you do.
```

It's probably also worth noting that in a lot of cases, you really *won't* use a lot of what you load, at least initially. If you're working in an OOP fashion, for example, you won't use every method at once on an object, but you're still going to have to pull the whole object in to use it.

If you're writing functional code, the same applies: there will often be some generic parts of stuff that you won't use for *every* request that you still won't split out into a seperate file simply because they are *often* used..

I think I'd have to guess that around 40% of the code I load gets used on a single request, but that is because of the aforementioned problem of using classes. It's not that big a worry to me, because I use APC to lessen the hit, and thankfully(? :)) I don't have thousands of r/s, just a few hundred.

Seems like a good change to me.

From: Rasmus Lerdorf	Date: Thu Feb 26 11:35:29 2009
Subject: Re: [APC-DEV] [RFC] APC/PHP Lazy Loading	
References: 1 2	Groups: php.apc.dev php.internals

Gopal V wrote:

```
> shire wrote:
>
>> http://tekrat.com/downloads/bits/apc\_lazy\_php53.patch
>>
>> You should be able to apply the above patch to the latest PHP_5_3
```

```
>> branch, and recompile the latest APC CVS against it. Two ini settings
>> enable/display lazy loading:
>>
>> apc.lazy_functions=On/Off
>> apc.lazy_classes=On/Off
>
> Awesome!
```

Yup, I am all for the lazy loading patch as well. I don't think people really realize how much code they load that is never run. Especially all the folks with massive front controllers.

```
>> Alternative implementations would include replacing the function
>> entries with an internal function that would load the opcodes when
>> called, however I found this implementation to be problematic, still
>> requires changes to PHP, and would also require inserting entries into
>> the function/class tables which itself ends up being an expensive task
>> for extremely large codebases.
>
> I still haven't given up on the executor hooks. But can't argue with
> code that works (yes, it works for most of my tests).
>
> I should finish up the RO patches in place so that we can catch stuff
> being overwritten in shm without locks - reflection, xdebug and suhosin
> looks like potential culprits here.
```

I wouldn't worry about xdebug at all. We should probably just turn off the opcode cache part when xdebug is active if it is a problem.

From: Rasmus Lerdorf	Date: Thu Feb 26 14:05:55 2009
Subject: Re: [APC-DEV] [RFC] APC/PHP Lazy Loading	
References: 1 2 3 4	Groups: php.apc.dev php.internals

shire wrote:

```
> I agree for the general case, in our development environment though this
> might cause some pains. But we could always start there and see how it
> goes. I agree that Xdebug isn't really a use case we always need to
> optimize for.
```

Is it ever a case we need to optimize for? If you are running xdebug, you aren't worried about execution speed. You certainly aren't going to be running your PHP under xdebug in any sort of production environment.

-Rasmus

From: shire	Date: Thu Feb 26 14:24:32 2009
Subject: Re: [APC-DEV] [RFC] APC/PHP Lazy Loading	
References: 1 2 3 4 5	Groups: php.apc.dev php.internals

Rasmus Lerdorf wrote:

```
> shire wrote:
>> I agree for the general case, in our development environment though this
>> might cause some pains. But we could always start there and see how it
>> goes. I agree that Xdebug isn't really a use case we always need to
>> optimize for.
>
> Is it ever a case we need to optimize for? If you are running xdebug,
```

> you aren't worried about execution speed. You certainly aren't going to
> be running your PHP under xdebug in any sort of production environment.

I agree I don't see a case for ever using XDebug in a production environment with APC (unless you're providing some sort of developer service, and even then). Obviously the user cache needs to function as you could be debugging something related. In our development environment I'm just not sure how much additional time this will cost us in addition to xdebug. I believe it takes 6 seconds (roughly there might be other crud going on there) for us to compile all the code into opcodes for execution, so tacking this on to xdebug might cause some headaches for developers.

But like I said, I think this case is pretty extraordinary, so for the sake of getting this great feature in if we need to disable this then I think I'm fine with figuring out some other solution to it should it actually be a problem at a later date. Even if I can disable this in APC then that might be enough to work around it.

-shire

From:	Ronald Chmara	Date:	Thu Feb 26 21:38:16 2009
Subject:	Re: APC/PHP Lazy Loading		
References:	1 2 3 4 5	Groups:	php.apc.dev php.internals

On Feb 26, 2009, at 5:58 AM, Rodrigo Saboya wrote:

> Ronald, I think you are overreacting a little bit. It may be that
> proper written could would get no benefit from this patch since it
> would not load unneeded code and this patch ends up speeding up
> environments where such "correct" loading isn't done. I don't think
> that's a reason to disqualify a feature that brings benefits with
> no significant drawbacks.

>

> For the average PHP programmer, the language will simply "get
> faster". That can't be bad in any way. It doesn't encourage you to
> write bad code, it just doesn't kick you in the nuts when you do.

Sold. (The best arguments are always the short ones).

As I said: "I'm trying to raise a token flag of discussion/resistance", but nobody in the discussion seems to have found any really important downsides, yet, with only me raising *any* sort of flag (though I admit it's not much of one).

:)

-Bop

Anexo J – Discussões da Comunidade SourceForge

Interação 1.4.SFG

Posted 1 year ago by [rameshch45](#) [1]

I have a few questions about using different licenses of open source software and other related topics.

What are some of the famous online discussion forum websites where people join in and discuss about open source?

Posted 1 year ago by [silverfang](#) [2]

Here (of course). You can ask me the question and I'll do my best to answer it.

Posted 1 year ago by [rameshch45](#) [3]

Sure. I wondered and asked that because I did not find many posts in the forums that I can search/read, essentially learning from earlier users... So, I thought maybe this is a site specific placeholder and there could be some other extensively exclusive forum to discuss open-source.

Thank you.

Posted 1 year ago by [doofus](#) [4]

http://developer.kde.org/documentation/licensing/licenses_summary.html

<http://www.fsf.org>

<http://www.opensource.org>

<http://www.freebsd.org>

Interação 2.2.SFG

Posted 1 year ago by [alanvw](#) [1]

How do I install Sourceforge software on Win.XP. I followed the instruction in the package using Command Prompt. Is the instruction for Linux users? as the './configure' command is not recognized by Command Prompt.I,m not a programmer.

Posted 1 year ago by [SF.net rturk](#) [2]

alanvw:

Which SourceForge project are you trying to install?

Ross

Interação 3.2.SFG

Posted 1 year ago by [tanmandb](#) [1]

Anybody heard of .fer files? How do you open them??

Posted 1 year ago by [silverfang](#) [2]

Apparently, FCR is a file/folder encryption. Looking at the source I can tell it's pretty old and outdated. You can't open the file.

Interação 4.1.SFG

Posted 1 year ago by [shivnithravi](#)

Plz any one tell me about license issues of CImg.Is it a free ware?

Interação 5.1.SFG

Posted 1 year ago by [shivnithravi](#)

How to remove the dependencies of CImg with imageMagicK

Livros Grátis

(<http://www.livrosgratis.com.br>)

Milhares de Livros para Download:

[Baixar livros de Administração](#)

[Baixar livros de Agronomia](#)

[Baixar livros de Arquitetura](#)

[Baixar livros de Artes](#)

[Baixar livros de Astronomia](#)

[Baixar livros de Biologia Geral](#)

[Baixar livros de Ciência da Computação](#)

[Baixar livros de Ciência da Informação](#)

[Baixar livros de Ciência Política](#)

[Baixar livros de Ciências da Saúde](#)

[Baixar livros de Comunicação](#)

[Baixar livros do Conselho Nacional de Educação - CNE](#)

[Baixar livros de Defesa civil](#)

[Baixar livros de Direito](#)

[Baixar livros de Direitos humanos](#)

[Baixar livros de Economia](#)

[Baixar livros de Economia Doméstica](#)

[Baixar livros de Educação](#)

[Baixar livros de Educação - Trânsito](#)

[Baixar livros de Educação Física](#)

[Baixar livros de Engenharia Aeroespacial](#)

[Baixar livros de Farmácia](#)

[Baixar livros de Filosofia](#)

[Baixar livros de Física](#)

[Baixar livros de Geociências](#)

[Baixar livros de Geografia](#)

[Baixar livros de História](#)

[Baixar livros de Línguas](#)

[Baixar livros de Literatura](#)
[Baixar livros de Literatura de Cordel](#)
[Baixar livros de Literatura Infantil](#)
[Baixar livros de Matemática](#)
[Baixar livros de Medicina](#)
[Baixar livros de Medicina Veterinária](#)
[Baixar livros de Meio Ambiente](#)
[Baixar livros de Meteorologia](#)
[Baixar Monografias e TCC](#)
[Baixar livros Multidisciplinar](#)
[Baixar livros de Música](#)
[Baixar livros de Psicologia](#)
[Baixar livros de Química](#)
[Baixar livros de Saúde Coletiva](#)
[Baixar livros de Serviço Social](#)
[Baixar livros de Sociologia](#)
[Baixar livros de Teologia](#)
[Baixar livros de Trabalho](#)
[Baixar livros de Turismo](#)