



UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

ABORDAGEM BASEADA EM SERVIÇOS PARA O AUMENTO DO SUPORTE SEMÂNTICO À
COLABORAÇÃO EM PROCESSOS INTENSIVOS EM CONHECIMENTO

EDNILSON VELOSO MOURA

Orientadores

Prof^a. Dr^a. Fernanda Araujo Baião Amorim

Prof^a. Dr^a. Flávia Maria Santoro

RIO DE JANEIRO, RJ – BRASIL

SETEMBRO DE 2013

ABORDAGEM BASEADA EM SERVIÇOS PARA O AUMENTO DO SUPORTE SEMÂNTICO À
COLABORAÇÃO EM PROCESSOS INTENSIVOS EM CONHECIMENTO

EDNILSON VELOSO MOURA

DISSERTAÇÃO APRESENTADA COMO REQUISITO PARCIAL PARA
OBTENÇÃO DO TÍTULO DE MESTRE PELO PROGRAMA DE PÓS-GRADUAÇÃO
EM INFORMÁTICA DA UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE
JANEIRO (UNIRIO). APROVADA PELA COMISSÃO EXAMINADORA ABAIXO
ASSINADA.

PROF^a FERNANDA ARAUJO BAIÃO AMORIM, D.Sc. – UNIRIO

PROF^a FLÁVIA MARIA SANTORO, D.Sc. - UNIRIO

**PROF. LEONARDO GUERREIRO AZEVEDO, D.Sc. –PPGI-UNIRIO E IBM RESEARCH -
BRASIL**

PROF. JOSE MARIA NAZAR DAVID, D.Sc. - UFJF

RIO DE JANEIRO

2013

Moura, Ednilson Veloso.

M929 Abordagem baseada em serviços para o aumento do suporte semântico à colaboração em processos intensivos em conhecimento / Ednilson Veloso Moura, 2013.
141 f. ; 30 cm + CD-ROM

Orientadora: Fernanda Araujo Baião Amorim.

Coorientadora: Flávia Maria Santoro.

Dissertação (Mestrado em Informática) - Universidade Federal do Estado do Rio de Janeiro, Rio de Janeiro, 2013.

1. Processos intensivos em conhecimento. 2. Arquitetura orientada a serviços (Computador). 3. Ontologia (Informática). 4. Colaboração. I. Amorim, Fernanda Araujo Baião. II. Santoro, Flávia Maria. III. Universidade Federal do Estado do Rio de Janeiro. Centro de Ciências Exatas e Tecnológicas. Curso de Mestrado em Informática. IV. Título.

CDD - 005.5

Dedico esta dissertação a minha doce e amada esposa Liliane.

Não há diálogo se não há uma intensa fé nos homens. Fé no seu poder de fazer e de refazer. De criar e recriar. Fé na sua vocação de ser mais. A fé nos homens é um dado a priori de diálogo. Por isso, existe antes mesmo de que se instale. O homem dialógico tem fé nos homens antes de encontrar-se frente a frente com eles. Esta, contudo, não é uma fé ingênua. O homem dialógico, que é crítico, sabe que, o poder de fazer, criar, de transformar é um poder dos homens. Sabe também que podem eles, em situação concreta, alienados, ter esse poder prejudicado. Esta possibilidade, porém em lugar de cortar no homem dialógico a sua fé nos homens, aparece a ele, pelo contrário – como um desafio ao qual tem de responder. Está convencido de que este poder de fazer e transformar, mesmo que negado em situações concretas, tende a renascer. Pode renascer, tende a renascer. Não gratuitamente, mas na e pela luta por sua libertação.

(Paulo Freire)

AGRADECIMENTOS

Agradeço as minhas orientadoras Fernanda e Flávia que guiaram meus passos para conseguir realizar este trabalho. A Fernanda pela paciência, sabedoria, simpatia e amizade demonstrada durante todo este período. A Flávia pela dose certa de firmeza e sensibilidade. Sem vocês não teria conseguido.

Também ao professor Leonardo Azevedo pela sua participação crítico construtivo de extrema importância. Ao meu amigo Thiago Procacci por sua valiosa ajuda na revisão do meu texto.

Agradeço a todos os professores da UNIRIO, juntamente com os colegas da turma do Mestrado 2011 com os quais pude fazer reflexões à construção deste trabalho. Muito obrigado.

Moura, Ednilson Veloso. **Abordagem Baseada em Serviços para o Aumento do Suporte Semântico à Colaboração em Processos Intensivos em Conhecimento**. UNIRIO, 2013. 106 páginas. Dissertação de Mestrado. Departamento de Informática Aplicada, UNIRIO.

RESUMO

Processos que envolvem a criação e o uso do conhecimento de maneira intensiva, denominados Processos Intensivos em Conhecimento (PIC), tem sido prioridade para a gestão do conhecimento orientada por processos de negócio. A colaboração entre as pessoas envolvidas neste tipo de processo é fundamental para a efetividade da criação, uso e compartilhamento do conhecimento. Nesses cenários, a colaboração entre os participantes é fundamental para que o conhecimento possa ser efetivamente criado, usado e compartilhado. Contudo, o suporte computacional para esta colaboração não é adequado, devido à falta de metodologias e ferramentas para apoiar a perspectiva colaborativa de um PIC, essencialmente, sua execução em um ambiente distribuído e heterogêneo. Este trabalho propõe XCuteKIP, uma abordagem orientada a serviços para suporte à colaboração em PIC que engloba um método e uma ferramenta para a recomendação de atividades colaborativas. O método é composto por duas fases: em tempo de desenvolvimento, os serviços de colaboração são automaticamente recomendados com base nas características que se encontram presentes no modelo do PIC, que deve seguir uma ontologia de colaboração; em tempo de execução, os serviços recomendados são compostos e instanciados de modo a fornecer um suporte de *groupware* para a execução das tarefas de conhecimento intensivo do processo. A arquitetura da ferramenta XCuteKIP é especificada e implementada sobre a arquitetura de *groupware* denominada WGWSOA, a qual disponibiliza uma infraestrutura baseada em serviços de apoio ao desenvolvimento e execução de aplicações colaborativas. O trabalho desta dissertação incluiu a definição e a implementação da arquitetura e do método, focando na recomendação dos serviços colaborativos que mais adequadamente suportam a colaboração em um PIC. A aplicação de XCuteKIP em um estudo de caso mostrou que seu uso facilitou a escolha de serviços colaborativos que mais adequadamente apoiem as atividades colaborativas.

Palavras-chave: Processos Intensivos em Conhecimento, Ontologia de Colaboração, Arquitetura Orientada a Serviços.

ABSTRACT

Business Process-Oriented Knowledge Management approaches in Organizations are beginning to focus on processes that involve intense generation, use and dynamic conversion of knowledge during their execution, so called Knowledge-intensive processes (KIP). In those scenarios, the collaboration among participants is fundamental so that knowledge can be effectively created, used and shared. However, the computational support for this collaboration is not adequate, due to the lack of methodologies and tools to model the collaborative perspective of a KIP, and to support its execution in a distributed and heterogeneous environment. This work proposes XCuteKIP, a service-oriented approach for collaboration support in KIPs that encompasses a method and a supporting tool. The method is composed by two main phases: at design time, collaboration services are automatically recommended based on the characteristics that are present in the KIP model, which should follow a Collaboration Ontology; at runtime, the recommended services are composed and instantiated so as to provide a groupware support for the execution of the knowledge-intensive tasks of the process. The architecture of the XCuteKIP tool is specified and implemented on top of WGWSOA, a service-based infrastructure to support the development and execution of collaborative applications. The application of XCuteKIP on a case study showed that its use helps the participants in KIP, by automatically recommending groupware services that adequately support collaborative activities within the KIP.

Key-words: Knowledge-intensive Process, Collaboration Ontology, Service Oriented Architecture.

Índice

| | |
|--|-----------|
| Capítulo 1 – Introdução | 1 |
| 1.1. Problema..... | 3 |
| 1.2. Hipótese..... | 4 |
| 1.3. Objetivos..... | 4 |
| 1.4. Metodologia da Pesquisa..... | 5 |
| 1.5. Estrutura da Dissertação | 6 |
| Capítulo 2 – Fundamentação Teórica | 7 |
| 2.1. Processos Intensivos em Conhecimento..... | 7 |
| 2.1.1. Ontologia de Processo Intensivo em Conhecimento (KIPO)..... | 10 |
| 2.2. Modelo 3C de Colaboração | 14 |
| 2.3. CONTO Ontology | 16 |
| 2.3.1. Ontologia de Cooperação | 16 |
| 2.3.2. Ontologia de Comunicação | 18 |
| 2.3.3. Ontologia de Coordenação | 19 |
| 2.4. A Web-based Groupware Service-Oriented Architecture (WGWSOA)..... | 21 |
| Capítulo 3 – Colaboração em Processos Intensivos em Conhecimento..... | 25 |
| 3.1. Taxonomias para <i>Groupware</i> | 25 |
| 3.2. Suporte Computacional para Atividades Colaborativas..... | 28 |
| Capítulo 4 – Proposta de Solução: XCuteKIP..... | 31 |
| 4.1. Classificação dos serviços colaborativos da WGWSOA | 33 |
| 4.2. Especificação formal de Atividades Colaborativas em um PIC..... | 33 |
| 4.3. Mapeamento KIPO X WGWSOA..... | 42 |
| 4.4. Método para Recomendação de Serviços colaborativos | 46 |
| 4.4.1. Recomendação de Serviços Colaborativos..... | 46 |
| 4.4.2. Execução de Atividades Colaborativas do Processo..... | 51 |

| | |
|---|-----------|
| 4.5. Arquitetura XCuteKIP | 52 |
| 4.5.1. Protótipo do XCuteKIP | 54 |
| 4.5.1. Tradução da KIPO para Esquema Relacional | 56 |
| Capítulo 5 – Estudo de Caso e Análise de Resultados..... | 57 |
| 5.1. Cenário do Estudo de Caso..... | 57 |
| 5.2. Descrição Geral do Estudo de Caso | 60 |
| 5.3. Questionários de avaliação aplicados no estudo de caso..... | 60 |
| 5.4. Análise dos Resultados..... | 63 |
| Capítulo 6 – Conclusão e Trabalhos Futuros | 70 |
| 6.1. Resumo da pesquisa | 70 |
| 6.2. Contribuições..... | 70 |
| 6.3. Limitações da Pesquisa..... | 71 |
| 6.4. Trabalhos Futuros | 71 |
| Referências..... | 73 |
| Anexo I – Glossário | 78 |
| Anexo II – Respostas ao Questionário de Recomendação de Groupware..... | 79 |
| Anexo III – Avaliação do XCuteKIP | 92 |
| Anexo III – Tradução da KIPO para Esquema Relacional | 98 |
| Anexo IV – Esquema Relacional | 99 |

Índice de Figuras

| | |
|---|-----------|
| Figura 1: Modelo de processo de geração de ideias Fonte: MIGUEZ e ABREU (2011). | 9 |
| Figura 2: Estrutura da KIPO Fonte: FRANÇA (2012). | 11 |
| Figura 3: KIPO – Reuso da BPO Fonte: FRANÇA (2012). | 12 |
| Figura 4: Business Rules Ontology Fonte: LOPES (2011)..... | 13 |
| Figura 5: Interação em PIC Fonte: FRANÇA (2012)..... | 14 |
| Figura 6: Modelo 3C Fonte: FUKS (2005). | 15 |
| Figura 7: Ontologia de Cooperação Fonte: OLIVEIRA (2009). | 17 |
| Figura 8: Ontologia de Comunicação Fonte: OLIVEIRA (2009)..... | 18 |
| Figura 9: Ontologia de Coordenação – Dependência de fluxo Fonte: OLIVEIRA (2009). | 20 |
| Figura 10: Ontologia de Coordenação – Dependência de usabilidade Fonte: OLIVEIRA (2009). | 20 |
| Figura 11: Ontologia de Coordenação – Dependência de transferência Fonte: OLIVEIRA (2009)..... | 21 |
| Figura 12: Arquitetura da WGWSOA Fonte: DAVID e MACIEL (2009)..... | 22 |
| Figura 13: Árvore de classificação da taxonomia dos meios de comunicação Fonte: CALVÃO et al. (2012)..... | 27 |
| Figura 14: Visão Geral da XCuteKIP..... | 31 |
| Figura 14: Socialização Fonte: FRANÇA (2012). | 34 |
| Figura 15: Ontologia de colaboração Fonte: OLIVEIRA (2009). | 35 |
| Figura 16: Ontologia de Decisão Fonte: FRANÇA (2012)..... | 36 |
| Figura 17: Sincronia de eventos Fonte: FRANÇA (2012). | 36 |
| Figura 18: Interação comunicativa Fonte: FRANÇA (2012). | 37 |
| Figura 19: Agentes Sociais Fonte: FRANÇA (2012). | 38 |
| Figura 20: Ato comunicativo Fonte: FRANÇA (2012). | 40 |
| Figura 21: Mapeamento Fórum WGWSOA com Classes KIPO..... | 44 |
| Figura 22: Serviços de percepção WGWSOA Fonte: DAVID e MACIEL (2010). | 44 |

| | |
|---|-----------|
| Figura 23: Mapeamento entre a taxonomia de serviços colaborativos e os serviços da WGWSOA..... | 45 |
| Figura 24: Algoritmo Baseado no Axioma A1 | 47 |
| Figura 25: Algoritmo Baseado no Axioma A2 | 47 |
| Figura 26: Algoritmo Baseado no Axioma A3..... | 48 |
| Figura 27: Algoritmo Baseado no Axioma A4..... | 48 |
| Figura 28: Algoritmo Baseado nos Axiomas A5 ao A9 | 49 |
| Figura 29: Algoritmo Baseado nos Axiomas A10 ao A12 | 49 |
| Figura 30: Fase de Recomendação de Serviços Colaborativos | 50 |
| Figura 31: Fase de Execução de Atividades Colaborativas do Processo..... | 52 |
| Figura 32: Arquitetura Conceitual da XCuteKIP..... | 53 |
| Figura 33: Tela Lista de Atividades do módulo <i>Collaboration Designer</i> | 55 |
| Figura 34: Tela Recomendação do módulo <i>Collaboration Designer</i> | 55 |
| Figura 35: Tela Configuração de Serviço Colaborativo para a WGWSOA | 56 |
| Figura 36: Processo de Levantamento de Requisitos Fonte: SOMMERVILLE (2007). | 58 |
| Figura 37: Experiência com atividades de levantamento de requisitos para desenvolvimento de sistemas..... | 63 |
| Figura 38: Experiência com atividades com base no uso de <i>groupware</i> (interações colaborativas)..... | 64 |
| Figura 39: Percentual de recomendações de <i>groupware</i> para atividades colaborativas por cada analista e pela XCuteKIP | 64 |
| Figura 40: Percentual de participantes que classificaram as atividades como colaborativas | 65 |
| Figura 41: Importância da estrutura do discurso para escolha do <i>groupware</i> de apoio à atividade | 66 |
| Figura 42: Importância de quantidade de interlocutores para escolha da atividade | 67 |
| Figura 43: Importância de sincronia na escolha do <i>groupware</i> para atividade | 67 |
| Figura 44: Quantidade de Atividades Recomendadas para o Configurador (Anexo III) | 68 |

Índice de Tabelas

| | |
|--|-----------|
| Tabela 1: Tarefas de Processo de Levantamento de Requisitos. | 59 |
| Tabela 2: Bloco de perguntas para avaliar a experiência dos participantes | 61 |
| Tabela 3: Bloco de Recomendação de Serviços Colaborativos | 61 |

Lista de Abreviaturas Utilizadas

| | |
|------------|---|
| BPO | Business Process Ontology |
| BPO-KM | Business Process-Oriented Knowledge Management |
| BRO | Business Rules Ontology |
| CO / CONTO | Collaborative Ontology |
| DO | Decision Ontology |
| GC | Gestão do Conhecimento |
| KIPCO | Knowledge Intensive Process Core Ontology |
| KIPO | Knowledge Intensive Process Ontology |
| ORN | Ontologia de Regras de Negócio |
| PIC | Processo Intensivo em Conhecimento |
| SOA | Service Oriented Architecture |
| UML | Unified Modeling Language |
| URML | Unified Rule Modeling Language |
| WGWSOA | Web-based Groupware Service Oriented Architecture |

Capítulo 1 – Introdução

As formas de trabalho têm se modificado e se transformado com o tempo e continuam em constante evolução, as organizações de hoje vêm mudando rapidamente. Atualmente, pessoas trabalham em atividades menos mecanicistas e mais intelectuais e colaborativas. Essa transição de atividades mecânicas e individuais para intelectuais e em grupo tem chamado à atenção da comunidade acadêmica (ANDERSSON, BIDER e PERJONS, 2004; STRIJBOSCH, READER e VAN BOMMEL, 2011).

As atividades ligadas ao conhecimento e executadas em grupo (colaborativamente) têm sido investigadas por pesquisadores de áreas multidisciplinares incluindo a área de Gestão do Conhecimento. Segundo Santos (2001), a Gestão do Conhecimento pode ser definida como a disciplina que tem como objetivo facilitar o acesso ao conhecimento e prover sua integração.

A Gestão do Conhecimento tem ganhado muita importância, tanto na academia quanto no mundo dos negócios. Diversos autores consideram o conhecimento e sua gestão como recursos fundamentais dentro das organizações, uma vez que podem prover maior competitividade e produtividade (NONAKA e TAKEUCHI, 1997) (DAVENPORT e PRUSAK, 1998) e (DRUCKER, 1999). Entretanto, para que projetos de Gestão do Conhecimento tenham sucesso, estes devem refletir a rotina dos funcionários da organização, de forma a vincular suas atividades diárias e experiências aos processos de negócio. Nesta linha, a Gestão de Conhecimento Orientada a Processos de Negócio (BPO-KM, ou *Business Process-Oriented Knowledge Management*) se concentra no conhecimento compartilhado entre os participantes de um processo de negócio, conectando o conhecimento gerado aos elementos do processo (STROHMAIER, 2004). Neste cenário, a Gestão do Conhecimento tem se mostrado relevante e a abordagem BPO-KM (*Business Process Oriented Knowledge*

Management) tem tido um impacto positivo sobre o desempenho dos processos de negócio das organizações (DAVENPORT, 1993; HAMMER e CHAMPY, 2009; HARMON, 2010).

Conhecimento é o tema central de um tipo de processo de negócio chamado de Processo Intensivo em Conhecimento (PIC), que tem sido o foco das abordagens de BPO-KM. Nesse processo, os participantes do processo usam de suas experiências para a tomada de decisões e para a resolução de problemas durante o fluxo do processo (PROBST *et al.*, 1997; REMUS, 2002; FRANÇA, 2012). Os cenários típicos de PIC ocorrem, por exemplo, em agências governamentais, grandes escritórios de advocacia e empresas de seguros, onde o conhecimento tácito sobre como a organização do trabalho e *know-how* desenvolvidos em casos anteriores são reconhecidos por atores do processo como recursos valiosos. Exemplos de PIC em organizações incluem a tomada de decisão da alta gerência, a criação de novos produtos, melhoria de produtos existentes, etapas de especificação e modelagem de sistemas de informação, entre outros. Todavia, devido à imprevisibilidade desses cenários, as atividades mais importantes do PIC normalmente são realizadas com apoio computacional insuficiente ou inadequado. Isto se deve ao fato que o PIC não segue a estrutura tradicional de processos de negócio, dificultando, assim, seu apoio computacional efetivo (FRANÇA, 2012).

FRANÇA (2012) propôs uma ontologia para tornar explícito um PIC, e assim promover o entendimento destes processos dentro das organizações. A esta ontologia chamou KIPO (*Knowledge Intensive Process Ontology*), que tem como propósito definir de maneira clara os conceitos característicos de PIC, apresentando-os em uma ontologia com suas hierarquias, cardinalidade e relacionamentos, tendo por base uma ontologia de fundamentação. Segundo Baião *et al.* (2008) e Guizzardi *et al.* (2010), a representação de um modelo conceitual de forma mais precisa é buscada utilizando uma linguagem baseada em ontologias de fundamentação.

Em função de sua complexidade, muitos Processos Intensivos em Conhecimento são de natureza colaborativa (SINGULARITY, 2009). Nos cenários em que os processos são colaborativos, as atividades são realizadas coletivamente, obtendo maior produtividade. O apoio computacional voltado para o trabalho colaborativo, neste sentido, pode contribuir para o alcance de resultados mais significativos para as organizações (ANDERSSON, BIDER e PERJONS, 2004).

Dentre as ferramentas e técnicas colaborativas utilizadas nesse tipo de processo, pode-se citar: *brainstorming*, edição colaborativa de documentos, workshops, debates, grupo de trabalhos. Dependendo das atividades executadas, algumas ferramentas colaborativas são

mais adequadas do que outras. No entanto, segundo FUCKS, RAPOSO e GEROSA (2003), se tem apenas um conhecimento superficial de como as pessoas colaboram e, por essa razão, a escolha das ferramentas que mais bem atendem às necessidades da colaboração desejadas em uma determinada atividade não é uma tarefa trivial. Além disso, a vasta quantidade de opções de ferramentas colaborativas reforça a complexidade desta escolha no suporte às atividades em processos intensivos em conhecimento. A escolha de tais ferramentas colaborativas está na seara das habilidades de especialistas em colaboração. Contudo, nem sempre especialistas estão disponíveis ou os participantes não conhecem as características dos serviços de colaboração para apoiar este importante tipo de processo. Por isso, uma ferramenta que auxilie na escolha dos serviços colaborativos se torna uma importante contribuição para apoiar as atividades de PIC.

1.1. Problema

Para manter o conhecimento mais assertivo, organizações têm se preocupado com o aspecto colaborativo em seus processos de negócio. Em um ambiente colaborativo, busca-se explorar ao máximo o potencial de seus colaboradores e os possíveis relacionamentos entre eles. O auxílio de computadores para automatizar as atividades do processo tem sido a estratégia adotada por diversas organizações a fim de ampliar a colaboração nas atividades destes processos (ANDERSSON, BIDER e PERJONS, 2004).

Processos organizacionais tradicionais são planejados primeiro e depois executados por seus participantes. A coordenação de suas atividades é frequentemente apoiada por ferramentas de suporte a estes processos (MOLZ, THOM e SCHEIDT, 1999).

Devido à sua natureza dinâmica e complexa, PIC tem pouco ou nenhum apoio de ferramentas computacionais (STOITSEV, 2009). Contudo, PAPAVALIOU *et al.* (2002) afirmaram que o suporte de TI para os trabalhadores que realizam atividade em PIC no seu trabalho diário pode ser através de sistemas de *groupware*, que é a tecnologia projetada para suportar especificamente o trabalho em grupo com facilidades para colaboração, e ferramentas de gestão de conhecimento para suporte à recuperação e acesso à informação.

Assim, esta dissertação apresenta uma abordagem de provimento de serviços colaborativos para apoiar as atividades colaborativas de processos intensivos em conhecimento. A estratégia utilizada nesta pesquisa utiliza uma arquitetura de *groupware* para se integrar a uma plataforma de suporte a atividades em PIC através de serviços.

A abordagem de serviços para integração entre sistemas está evidenciada em diversos trabalhos na literatura (PAPAZOGLU, *et al.* 2006) (TSAI, *et al.* 2004). Na Arquitetura Orientada a Serviços (SOA – *Service Oriented Architecture*), serviço é uma função independente, sem estado (*stateless*) que aceita uma ou mais requisições e devolve uma ou mais respostas através de uma interface padronizada e bem definida. Serviços podem também realizar partes discretas de um processo tal como editar ou processar uma transação. Serviços não devem depender do estado de outras funções ou processos. A tecnologia utilizada para prover o serviço, tal como uma linguagem de programação, não faz parte da definição do serviço (ERL, 2005).

Utilizando a abordagem de serviço, esta pesquisa se beneficiará das vantagens oriundas da arquitetura SOA, especificamente a redução da dependência tecnológica e a simplificação do processo de desenvolvimento por conta do aumento da flexibilidade e do reuso da infraestrutura de negócios pelas corporações (KRAFZIG; BANKE; SLAMA, 2007).

Dado o cenário da colaboração nos Processos Intensivos em Conhecimento e a dificuldade de escolher ferramentas de colaboração adequadas para apoiar um conjunto de atividades, esse trabalho tem por objetivo investigar uma forma automática para recomendar serviços de colaboração mais adequados, considerando as atividades envolvidas no processo.

O problema dessa pesquisa pode ser enunciado da seguinte forma: ***Como recomendar automaticamente serviços colaborativos adequados para apoiar atividades colaborativas em um Processo Intensivo em Conhecimento?***

1.2. Hipótese

No presente trabalho, assume-se que o processo intensivo em conhecimento sendo investigado seja conhecido e modelado, segundo o meta-modelo da KIPO (FRANÇA, 2012). A partir daí, a seguinte hipótese norteia o desenvolvimento desta pesquisa: ***Se for aplicado um mapeamento entre atividades colaborativas de um PIC e um conjunto de serviços colaborativos, então será possível recomendar serviços colaborativos adequados para apoiar atividades realizadas em grupo de um processo intensivo em conhecimento.***

1.3. Objetivos

O objetivo geral da presente dissertação é criar uma arquitetura, baseada em serviços, para apoiar a execução das tarefas colaborativas de um processo intensivo em conhecimento, realizando a descoberta automática de ferramentas de colaboração a partir de uma ontologia de colaboração.

Para alcançar o objetivo geral, foram estabelecidos os seguintes objetivos específicos:

- Especificar formalmente uma taxonomia para classificação de atividades colaborativas e descoberta automática de requisitos colaborativos nestas atividades;
- Desenvolver algoritmos para descobrir as características colaborativas das atividades de um PIC ;
- Especificar formalmente o mapeamento de atividades colaborativas para serviços colaborativos;
- Elaborar um método para recomendação automática de serviços colaborativos para apoio a atividades colaborativas em processos intensivos em conhecimento;
- Construir um protótipo computacional para execução do método proposto

1.4. Metodologia da Pesquisa

Quanto à natureza, esta pesquisa é considerada aplicada, pois objetiva gerar conhecimentos para aplicação prática e dirigida ao uso de ferramentas de grupo para apoiar atividades de processos dinâmicos que tem como particularidade depender do conhecimento de seus participantes para execução de suas atividades (GIL, 2008).

Do ponto de vista da forma de abordagem do problema esta pesquisa considera que há uma relação dinâmica entre o mundo real e o sujeito, isto é, um vínculo indissociável entre o mundo objetivo e a subjetividade do sujeito que não pode ser traduzido em números, o que define uma pesquisa qualitativa. A coleta de dados se faz diretamente em um ambiente produtivo. O cenário da coleta de dados ocorre em um ambiente de desenvolvimento de software, onde os participantes dividem-se nos papéis de analistas de sistemas, arquitetos de software, desenvolvedores e responsáveis técnicos.

Quanto aos objetivos, o delineamento da pesquisa está conduzido para o levantamento bibliográfico. Por fim é realizado um Estudo de Caso que envolve trabalhadores do conhecimento que atuam no setor de construção de software em uma empresa do ramo petrolífero, para validar a hipótese desta pesquisa.

Do ponto de vista dos procedimentos técnicos esta é uma pesquisa de campo. GIL (2008) define a pesquisa de campo como “aquela utilizada com o objetivo de conseguir informações e/ou conhecimentos acerca de um problema para o qual procuramos uma resposta, ou de uma hipótese que queiramos comprovar, ou, ainda, descobrir novos fenômenos ou as relações entre eles.”.

Com base nas pesquisas realizadas, esta pesquisa propõe um método e uma arquitetura fundamentada na Ontologia KIPO e em uma arquitetura de serviços colaborativos para comprovar a hipótese aqui levantada.

1.5. Estrutura da Dissertação

O restante desta dissertação está estruturado como se segue:

O segundo capítulo descreve os conceitos de Processos Intensivos em Conhecimento, as ontologias que embasam a fundamentação destes processos e a arquitetura utilizada para facilitar o desenvolvimento de *groupware*.

Já o terceiro capítulo apresenta os trabalhos relacionados ao apoio computacional das tarefas colaborativas de processos intensivos em conhecimento.

O objetivo do capítulo quatro é apresentar e descrever os passos do método e arquitetura propostos que integram serviços colaborativos que funcionam como instrumento de solução para os problemas propostos nesta dissertação.

O capítulo cinco delinea o estudo de caso conduzido, onde o método proposto foi aplicado. Aqui apresentaremos e discutiremos os resultados de tal aplicação metodológica.

Finalmente, o capítulo seis apresenta as considerações acerca da pesquisa, contribuições, e uma lista de trabalhos futuros.

Capítulo 2 – Fundamentação Teórica

Este capítulo tem como objetivo apresentar os conceitos necessários para a compreensão das ideias propostas nesta pesquisa, demonstrando o embasamento teórico necessário. Os Processos Intensivos em Conhecimento e seus fundamentos são evidenciados através de ontologias propostas para este fim. É descrita em detalhes a Ontologia de Colaboração utilizada nesta pesquisa, que estabelece os conceitos e relações que fundamentam conceitualmente a proposta de solução. Além disso, é descrita a arquitetura WGWSOA, que é a base da solução em termos de sua implementação.

2.1. Processos Intensivos em Conhecimento

Processo de negócio, no ambiente organizacional, é constituído por diversas atividades executadas por agentes, pelas especificações de fluxo de dados e o controle entre eles. Através de seus processos, as empresas realizam seus negócios (WESKE, 2007), (WFMC, 1999). GONÇALVES (2000) afirma que não existem produtos ou serviços oferecidos pelas organizações sem que sejam realizadas por um processo empresarial. DUMAS *et al.* (2013) concordam com tais declarações quando afirmam que processo de negócio é o que as organizações fazem quando entregam algum serviço ou produto para seus clientes, sejam internos ou externos. Tradicionalmente, a literatura classifica os processos empresariais em processos de negócio (ou de cliente), organizacionais (apoio ao processo de negócio) e gerenciais (GONÇALVES, 2000).

Processos de negócio visam à produção de bens e serviços por meio da transformação de insumos, gerando valor primário para as organizações. Fabricação de bicicletas e atendimento de pedidos de clientes são exemplos de processo de negócio.

Processos organizacionais, por sua vez, dão apoio aos processos produtivos. São essencialmente burocráticos, comportamentais ou de mudança. Insumos lógicos são usados

nestes processos. Incluem-se, como desta natureza, processos tais como contas a pagar, integração gerencial, dentre outros.

Por seu turno, os processos gerenciais dão direcionamento ao negócio, realizam negociações e fazem monitoração. Seus insumos são as informações e têm a capacidade de geração de grande valor agregado ao suporte para as organizações. Como ilustrações desta classe de processo citam-se: definição de metas da empresa, definição de preços com o fornecedor e acompanhamento do planejamento e orçamento.

Contudo, alguns processos se diferenciam dos processos tradicionais por contar com o conhecimento individual de seus participantes como peça central para a condução de suas atividades. Estes processos constituem uma parcela significativa dos processos núcleo das empresas e são conhecidos como Processos Intensivos em Conhecimento (PIC) (SINGULARITY, 2009).

A intensidade de conhecimento demandada em um processo de negócio é reconhecida por meio da complexidade das atividades que são realizadas, e do quanto essas atividades dependem do conhecimento de seus participantes. Processos que envolvem a criatividade (EPPLER et al.; 1999), inovação, tomada de decisão e constantes intervenções de especialistas demandam muito conhecimento (HAGEN et al., 2005); estas são as principais características que definem os PIC.

EPPLER, SEIFRIED e RÖPNACK (1999) apontaram seis características que definem a intensidade do conhecimento em um processo: contingência (evento externo e imprevisível que influencia a execução do processo), escopo de decisão (identifica informações relacionadas à decisão como um todo), agente de inovação (responsável por resolver questões durante o processo intensivo em conhecimento com inovação e criatividade), meia-vida do conhecimento (o quão rápido um conhecimento está desatualizado), agente de impacto (responsável por executar um processo intensivo em conhecimento e identificar questões durante a execução de processos intensivos em conhecimento) e tempo de aprendizado. PIC tem um tempo de aprendizado muito longo, o escopo de tomada de decisão pode variar muito, o fluxo de atividades é imprevisível devido ao alto número de eventualidades que podem ocorrer durante as atividades, o conhecimento gerado dentro das atividades se desatualiza rapidamente e o executor resolve os problemas com criatividade e inovação. MALDONADO (2008) citou que a complexidade é típica em PIC e as decisões tomadas são muitas vezes baseadas no julgamento pessoal e na experiência dos participantes.

MALDONADO (2008) afirma que, no que se refere à estrutura do fluxo de operação destes processos, eles são classificados como não estruturados ou semiestruturados. Quando as atividades realizadas nestes processos não obedecem a uma sequência cronológica definida antecipadamente, são classificados como processos não estruturados, e quando algumas atividades têm atividades subsequentes definidas a priori e outras não, estes processos são classificados como semiestruturados.

Ainda como propriedades de Processos Intensivos em Conhecimento, os participantes têm diferentes experiências e possuem conhecimento em áreas distintas, há uma grande diversidade nas fontes de informação e de tipos de mídia como entrada para as atividades (HAGEN et al., 2005). Há, em consequência, geração de conhecimento, que fica rapidamente obsoleta após a execução das atividades, sendo utilizada para fornecer entrada para a atividade seguinte (EPPLER et al.; 1999). PIC são baseados em exceção e têm um fluxo contínuo de repetição e realimentação, onde as atividades são altamente interdependentes.

ABECKER *et al.* (2001) citaram como características adicionais encontradas em processos intensivos em conhecimento a mudança dinâmica de objetivos e a exigência de ambiente de informação e restrições. Por outro lado, Gronau *et al.* (HOFFMAN, 2002 apud GRONAU *et al.*; 2005) argumentam que PIC se apoiam fortemente na socialização e na troca de conhecimento informal, dificultando o suporte de ferramentas de TI.

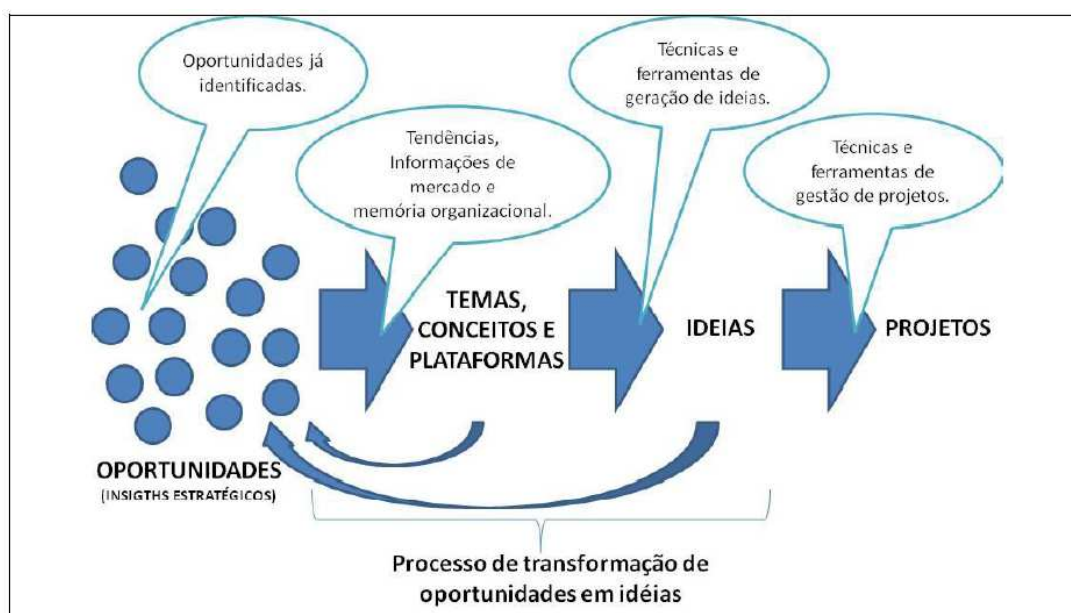


Figura 1: Modelo de processo de geração de ideias
 Fonte: MIGUEZ e ABREU (2011).

Neste sentido, DONADEL (2007) compartilha que PIC são difíceis de serem modelados como workflow de negócio tradicional, por sua complexidade, falta de estrutura fixa e pelo fato de que os objetivos nesses processos mudam dinamicamente e demandam constantes informações do contexto das atividades que estão sendo executadas.

A Figura 1 apresenta como exemplo o processo intensivo em conhecimento de transformação de oportunidades em ideias, onde as entradas para o processo são as oportunidades. Diversas atividades podem ser executadas para a coleta de informações que apoiará o executor nas tarefas, tais como verificação de tendências e de informações de mercado, além da própria memória organizacional e utilização de diversas técnicas e ferramentas para a geração das ideias (MIGUEZ e ABREU, 2011). Cada instância deste processo terá suas características peculiares dependentes do executor do processo e do contexto do processo no momento da execução.

2.1.1. Ontologia de Processo Intensivo em Conhecimento (KIPO)

Observando a falta de abordagens para a definição e organização de processos intensivos em conhecimento que buscasse organizar e externalizar o conhecimento envolvido em PIC, FRANÇA (2012) propôs uma ontologia para organizar os conceitos envolvidos em Processos Intensivos em Conhecimento, denominada *Knowledge-intensive Process Ontology* (KIPO). O objetivo geral do trabalho de FRANÇA (2012) é construir um modelo conceitual que, ao ser instanciado, é capaz de tornar explícito um processo intensivo em conhecimento, promovendo o seu entendimento na organização.

O emprego de ontologias para criar um vocabulário comum de termos organizados em uma taxonomia vem sendo amplamente adotado na literatura devido ao potencial de formalizar uma representação do conhecimento e de sua capacidade de formar o cerne de qualquer sistema de conhecimento que representa um dado domínio (CHANDRASEKARAN *et al.*, 1999). GUARINO (2001) definiu ontologia como: “*An ontology is a logical theory accounting for the intended meaning of a formal vocabulary*”. Para a disciplina de Sistemas de Informação, como visto em (GRUBER, 2009), ontologia é uma estrutura composta por quatro conjuntos (conceitos, relações, atributos e tipos de dados), uma hierarquia de conceitos (taxonomia de conceitos) e uma hierarquia de relações (taxonomia de relações) para representar um universo do discurso (GUARINO, 1997). Portanto, ontologia é uma descrição formal de um conceito, suas características e seus relacionamentos com outros conceitos. Tem

em vista definir, desambiguar ou identificar similaridade entre conceitos em um determinado universo de discurso. A abrangência da ontologia limita-se ao quanto se faz necessário para haver a distinção ou similaridade buscada.

A KIPO visa organizar e externalizar o conhecimento relacionado a diversos fatores em um PIC, dentre os quais: fatores motivacionais, interações sociais, sentimentos, crenças, imagens mentais, socializações, trocas informais, inovação e decisões envolvidas. Segundo seus autores, a KIPO é uma composição de quatro ontologias que possuíam, na semântica de suas classes, características que definem um processo intensivo em conhecimento, além de uma ontologia núcleo que foi construída especificamente para o propósito de conceituar um PIC, integrando diversos conceitos das demais ontologias (Figura 2). As quatro ontologias advindas da literatura são: Ontologia de Colaboração (*Collaborative Ontology* – CO ou CONTO), Ontologia de Decisão (*Decision Ontology* – DO), Ontologia de Regras de Negócio (*Business Rules Ontology* – BRO) e a Ontologia de Processos de Negócio (*Business Process Ontology* – BPO). Dentre elas, a Ontologia de Regras de Negócio e a Ontologia de Colaboração são ontologias bem fundamentadas em uma ontologia de alto nível, ou de fundamentação (FRANÇA, 2012).

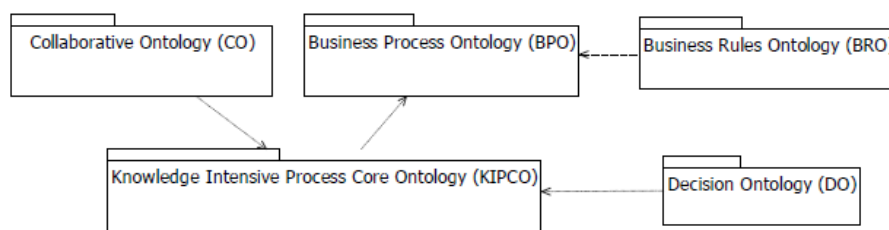


Figura 2: Estrutura da KIPO

Fonte: FRANÇA (2012).

Processos intensivos em conhecimento são essencialmente colaborativos devido à frequente troca informal de conhecimento realizada pelos participantes dos PIC. A Ontologia de Colaboração (*Collaborative Ontology* – CO ou CONTO) que faz parte da KIPO foi definida em (Oliveira, 2009) para apoiar a caracterização de PIC no que diz respeito aos conceitos envolvidos com colaboração. Como este é o aspecto de PIC mais explorado no escopo desta dissertação, a CONTO será apresentada na Seção 2.3.

A Ontologia de Processos de Negócio (*Business Process Ontology* – BPO) considera a BPMN (OMG, 2011) na definição de seu metamodelo. Em alguns casos é possível, dependendo da abstração empregada, prever um fluxo básico de atividades em PIC,

principalmente se estes processos forem apresentados em alto nível e com pouco detalhamento. Na Figura 3, duas atividades são conectadas por um fluxo de mensagem. As atividades podem estar associadas a objetos de conhecimento. Estes conceitos pertencem ao BPO e foram usados na KIPO. Os processos intensivos em conhecimento são compostos por atividades, que na BPO tem o conceito de Activity.

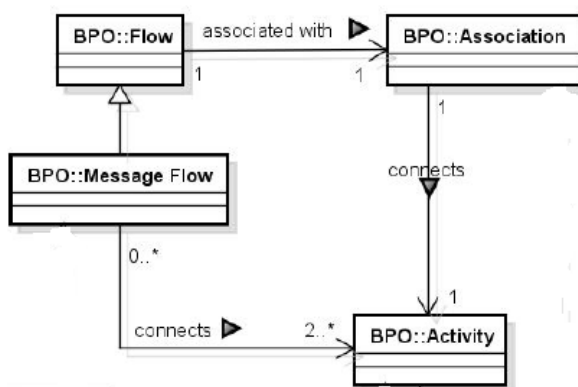


Figura 3: KIPO – Reuso da BPO
 Fonte: FRANÇA (2012).

O processo cognitivo de tomada de decisão está detalhado na ontologia proposta por Pereira (2010), a Ontologia de Decisão (Decision Ontology – DO). Esta ontologia propõe capturar e representar, através de seu metamodelo, o racional utilizado pelo profissional em seu ofício de tomar decisões. O uso da DO na KIPO se deve ao desejo de não perder o conhecimento que conduz um profissional a tomar suas decisões. Conceitos como Decision, Alternative, Disadvantage, Advantage, Criterion, Discarded Alternative, Chosen Alternative e Question fazem parte desta ontologia. As atividades intensivas em conhecimento podem envolver decisões, ou seja, as questões que estão associadas ao que precisa ser decidido. Tais decisões consideram alternativas, tais alternativas podem apresentar vantagens e desvantagens, que exigem alguns critérios para decidir entre as possíveis alternativas.

A Ontologia de Regras de Negócio (Business Rule Ontology – BRO) estabelece um metamodelo para representação de regras de negócio sem ambiguidade, mais completo e correto (LOPES, 2011). Um pequeno trecho da taxonomia de classes da BRO é ilustrado na Figura 4. A KIPO a reusa por conter conceitos e relacionamentos importantes que envolvem regras organizacionais, as quais influenciam a execução de um PIC. O modelo da BRO, representado na figura 4, reúne conceitos definidos nos metamodelos das duas linguagens de representação de regras de negócio mais expressivas na literatura, a UML-Based Rule

Modeling Language (URML) e a REVERSE Rule Markup Language (R2ML) (WAGNER *et al.*, 2006; GIURCA *et al.*, 2006).

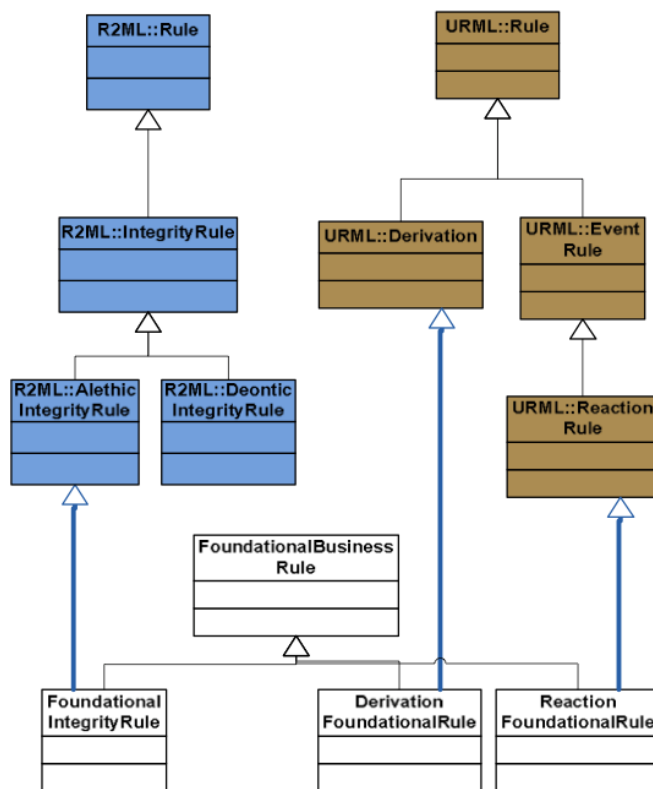


Figura 4: Business Rules Ontology
Fonte: LOPES (2011).

Na BRO são tratados três tipos de regras: de integridade; de derivação; e de reação. Regra de Integridade restringe alguma característica estrutural do domínio, como cardinalidade ou restrição aplicada a um relacionamento. A Regra de Derivação representa a derivação de novos conceitos no domínio a partir de conhecimento já existente no domínio trabalhado. Regra de Reação representa uma restrição no comportamento do domínio.

Na KIPO, os conceitos relativos aos domínios de Colaboração, Decisão, Regras de Negócio e Processo de Negócio foram reusados das respectivas ontologias para representar estes aspectos de um PIC. Já os conceitos que respondem diretamente a questões relacionadas exclusivamente a PIC foram definidos em uma quinta ontologia, a ontologia central de PIC (Knowledge Intensive Process Core Ontology – KIPCO), representada parcialmente na Figura 5. A KIPCO representa conceitos específicos necessários para completamente representar os PIC, inter-relacionando os conceitos das demais ontologias componentes da KIPO.

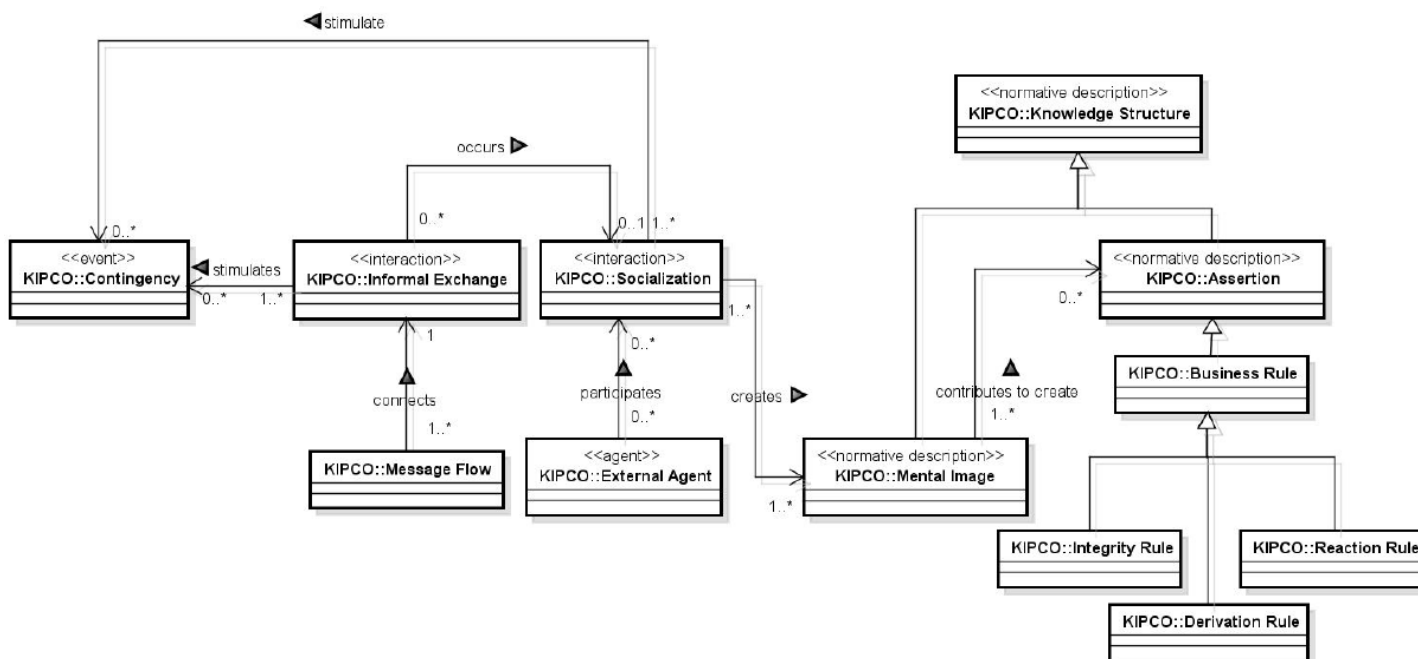


Figura 5: Interação em PIC
 Fonte: FRANÇA (2012).

Em um PIC, podem existir dois tipos distintos de agentes internos ao processo: Agente de Inovação (*Innovation Agent*) e Agente de Impacto (*Impact Agent*). O agente de inovação possui uma especialidade e é responsável por resolver questões durante o PIC com inovação e criatividade, enquanto que o agente de impacto é responsável por executar um PIC e identificar questões durante a execução de um PIC. No entanto, como PIC requer interações entre agentes a fim de estabelecer seu dinamismo, Agentes Externos (*External Agent*) podem contribuir com o processo, propondo ideias e contribuindo com o conhecimento usual em tomadas de decisão.

A próxima seção introduz o modelo que fundamenta a construção da Ontologia de Colaboração, que é o tema central desta dissertação.

2.2. Modelo 3C de Colaboração

O modelo 3C é frequentemente utilizado para classificar os sistemas colaborativos (FUKS *et al.*, 2005). O modelo se baseia no fato de que indivíduos trocam informações (comunicação),

organizam-se (coordenação) e operam em conjunto num espaço compartilhado (cooperação) com a finalidade de colaborar em atividades que não tenham objetivos conflitantes. A comunicação no processo gera compromissos que são gerenciados pela coordenação, que, por sua vez, organiza e dispõe as tarefas que são executadas cooperativamente. A perspectiva central deste modelo é a percepção, que leva os indivíduos se informarem sobre o que está acontecendo, e fornece informações para permitir aos colaboradores conduzirem seu trabalho.



Figura 6: Modelo 3C
Fonte: FUKS (2005).

A Figura 6 apresenta o diagrama do modelo 3C. A Comunicação é a ação de tornar comum e gerar compromissos através da troca de mensagens entre os participantes. Esta comunicação pode eventualmente gerar conflitos que prejudicam a Cooperação.

A Cooperação, que é a ação de trabalhar em conjunto em um espaço compartilhado (FUKS *et al.*, 2005) para tratar estes conflitos, estimula e intermedia a Percepção. A Percepção é o processo de adquirir informação por meio dos sentidos. A Percepção é estimulada e intermediada pela Comunicação e pela Cooperação.

Coordenação designa a ação de dispor, segundo uma ordem ou método, harmonizar (FERREIRA, 1986). Por meio dela, as pessoas, as atividades e os recursos são gerenciados para lidar com conflitos. A perspectiva central do Modelo 3C é a Percepção. Este conceito está relacionado ao contexto que ocorre quando os membros de um grupo trabalham. A Percepção fornece o conhecimento do conteúdo das contribuições individuais, o significado destas contribuições e seus objetivos (PINHEIRO, LIMA e BORGES, 2001).

Desenvolvedores e pesquisadores de *groupware* vêm desenvolvendo ao longo do tempo várias ferramentas para auxiliar o trabalho colaborativo. Essas ferramentas têm sido

classificadas segundo o Modelo 3C, exemplos destes tipos de ferramentas e sua classificação podem ser citados como a seguir:

- **Comunicação:** *e-mail*, lista de discussão, fórum, ferramentas de CSCA (*Computer Supported Collaborative Argumentation*), ferramentas de votação, mensagem instantânea, *chat*, videoconferência, teleconferência, telefone, etc.
- **Coordenação:** a agenda eletrônica e o *workflow*, que permitem o gerenciamento de fluxos de ações visando atingir um determinado objetivo.
- **Cooperação:** editor colaborativo de documentos, editor colaborativo de planilhas, editor colaborativo de gráficos, etc.

2.3. CONTO Ontology

O modelo 3C de colaboração, apresentado por FUCKS (2005), descreve uma estrutura que serve de base para formalizar o conhecimento sobre o domínio colaborativo presente em uma ontologia de colaboração, a *Collaboration Ontology* – CONTO (OLIVEIRA, 2009). Os conceitos que envolvem este modelo serviram de base para formar esta ontologia. A CONTO tem como objetivo representar conceitos que são relacionados ao trabalho que é desempenhado em conjunto por indivíduos.

Portanto, tendo por base o Modelo 3C, a Ontologia de Colaboração foi decomposta em três ontologias: Cooperação, Comunicação e Coordenação. A Ontologia de Comunicação apoia a realização das atividades de cooperação, na qual a comunicação gera compromissos fechados entre agentes. Estes compromissos são gerenciados pela coordenação durante o processo de comunicação e cooperação e, a coordenação é a responsável por promover a harmonia dos processos de cooperação.

2.3.1. Ontologia de Cooperação

A Ontologia de Cooperação apresenta conceitos que estão relacionados ao trabalho realizado em conjunto entre indivíduos em um espaço compartilhado. Nesta ontologia estão presentes, dentre outros, os conceitos de agentes envolvidos na cooperação, os espaços virtual e real onde os agentes interagem e as sessões colaborativas onde é realizado o trabalho (Figura 7).

Uma sessão colaborativa (*Collaborative Session*) é composta por ações de contribuição e é deliberada por acordos colaborativos. Pelo menos dois agentes devem

participar de uma sessão colaborativa. Durante a sessão colaborativa, pode haver a participação de recursos (*Resource Participation*). A coordenação organiza e harmoniza as participações (*Participation*) de uma sessão colaborativa através do conceito Protocolo (*Protocol*).

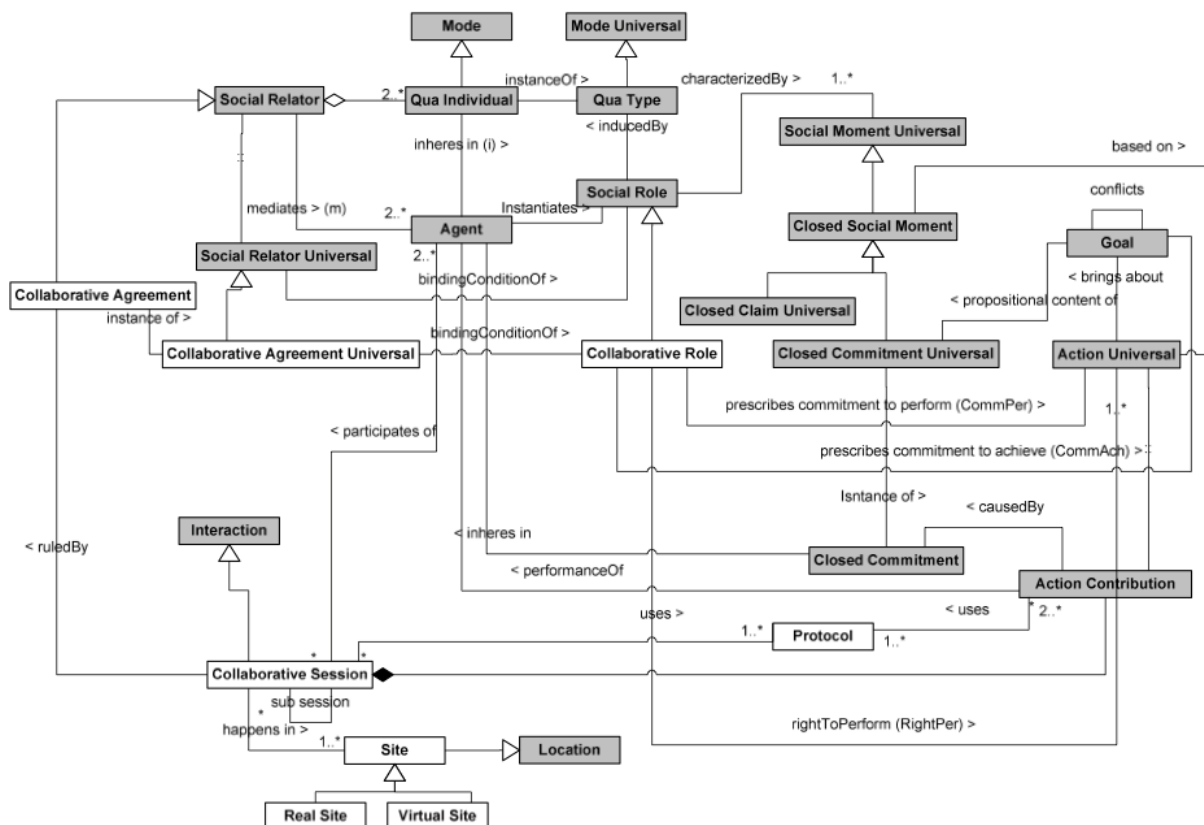


Figura 7: Ontologia de Cooperação

Fonte: OLIVEIRA (2009).

Os agentes assumem papéis nas sessões colaborativas. Estes papéis colaborativos levam os agentes a realizar ações que são causadas pelo Compromisso Fechado (*Closed Commitment*) de instanciar esse papel. Os acordos colaborativos podem ser firmados entre agentes de tal forma que sejam refletidos nos Papéis Colaborativos que esses agentes se comprometem a desempenhar em uma sessão colaborativa.

O trabalho colaborativo acontece nas sessões em um ambiente (*Site*), virtual (*Virtual Site*) ou real (*Real Site*). Estes são os conceitos que representam a localização espacial das sessões colaborativas. Um Ambiente Virtual é uma simulação do mundo real por uma ferramenta de software. Ambiente Real é o conceito que representa um espaço que existe no mundo físico, concreto. Exemplos de Ambientes Virtuais são: Fóruns, *Blogs*, Gerenciadores

de Documentos, etc. Já os Ambientes Reais são: a sala onde acontece a reunião, o estádio onde acontece a partida de futebol, etc.

2.3.2 Ontologia de Comunicação

Nesta ontologia são abordados elementos que representam a comunicação e as demais entidades envolvidas durante esse processo. Nesta ontologia estão presentes, dentre outros, os conceitos de Contribuições Materiais (*Material Contribution*), Atos Comunicativos (*Communicative Acts*), Percepção (*Perception*), Mensagens (*Messages*), Linguagem Idiomática (*Idiomatic Language*), Emissor (*Sender*) e Receptor (*Receiver*), como ilustrado na Figura 8.

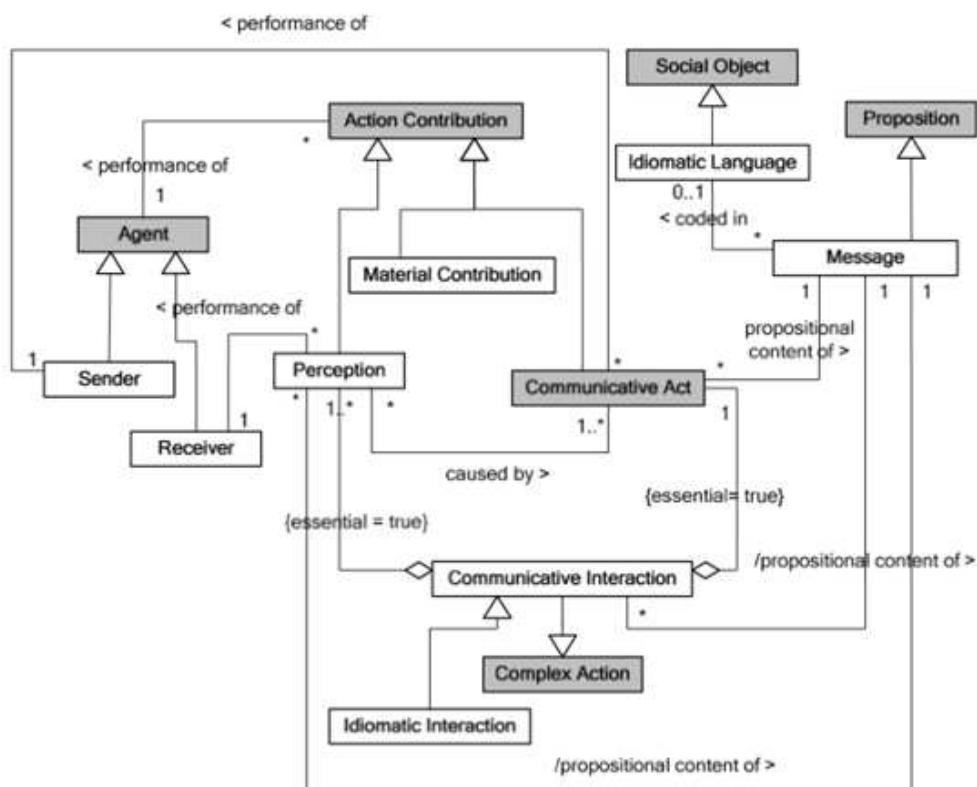


Figura 8: Ontologia de Comunicação
Fonte: OLIVEIRA (2009).

Uma Interação Comunicativa (*Communicative Interaction*) é uma ação complexa (*Complex Action*) composta por, pelo menos, um Ato Comunicativo (*Communicative Act*) e uma Percepção (*Perception*). Como ilustração de uma interação comunicativa, pode ser citado o caso de um escritor, assumindo o papel de Sender, que publica um livro. Esta é uma ação de

contribuição que está realizando um Ato Comunicativo (publicação de informação). Quando o livro é lido por outro agente que assume o papel de Receptor ou Receiver, observa-se a realização de uma Percepção (*Perception*) dessa informação. Desta forma, fecha-se a cadeia do processo de comunicação e, portanto, é gerada uma Interação Comunicativa. O conteúdo proposicional do ato comunicativo é a Mensagem (*Message*). A mensagem pode ser codificada em uma linguagem idiomática.

Agentes assumem o papel de emissor quando executam (*performance of*) um ato comunicativo em uma interação comunicativa. Já quando executam (*performance of*) a Percepção (*Perception*) de uma mensagem, assumem o papel de receptor.

As Interações Idiomáticas (*Idiomatic Interaction*) são subtipos de uma interação comunicativa e representam comunicação através de diálogos, jornais, revistas, televisão, livros, etc. Outro tipo de Interação Comunicativa que não seja idiomática é a utilização de gestos como um aperto de mão, um sinal de ok, entre outros.

2.3.3. Ontologia de Coordenação

A Ontologia de Coordenação apresenta conceitos que estão relacionados a promover organização e harmonia para os demais conceitos presentes nas duas ontologias previamente apresentadas. Esta ontologia captura as relações de dependência entre ações, potenciais de colaboração e possíveis conflitos, e é ilustrada pelas Figuras 9, 10 e 11.

A base para esta ontologia é a Teoria de coordenação de CROWSTON (1994). Esta teoria versa a respeito da dependência entre ações de agentes. Na ontologia de coordenação estão representadas algumas dessas dependências.

A dependência de Fluxo (*Flow*) é um tipo de dependência entre Ações na qual uma delas produz um Recurso que é utilizado (requerido) por outra Ação. Um exemplo desta dependência é a linha de montagem automotiva, onde a atividade de pintura de um automóvel só poderá acontecer se o carro já tiver sido produzido.

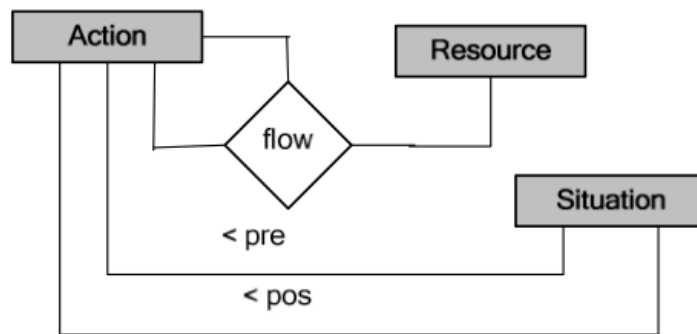


Figura 9: Ontologia de Coordenação – Dependência de fluxo
 Fonte: OLIVEIRA (2009).

Dependência de Usabilidade (*Usability*) pressupõe que um recurso deve atender as características específicas para as necessidades de uma determinada ação (Figura 10). No mesmo exemplo de produção automotiva anterior, a dependência de usabilidade garante que as peças encaminhadas para a montagem satisfaçam critérios, como por exemplo, de tamanho e formato necessários, para que o carro possa ser montado.

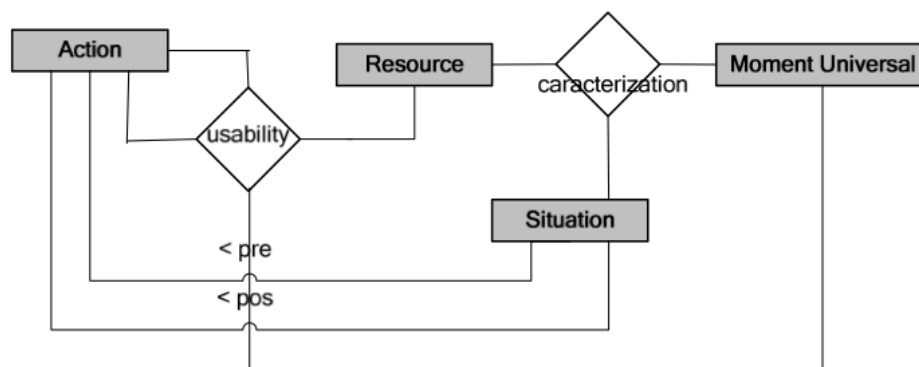


Figura 10: Ontologia de Coordenação – Dependência de usabilidade
 Fonte: OLIVEIRA (2009).

A dependência de Transferência (*Transfer*) define um recurso que foi criado em um local A será disponibilizado para uma ação que dele necessita em um local B (Figura 11). Novamente no exemplo da produção automotiva, essa dependência, quando atendida, garante

que os produtos produzidos na linha de produção estejam no lugar correto da linha de montagem para a utilização.

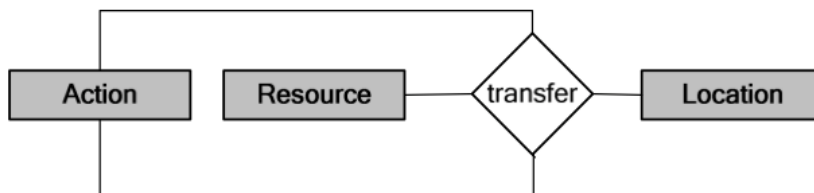


Figura 11: Ontologia de Coordenação – Dependência de transferência
Fonte: OLIVEIRA (2009).

Como visto anteriormente, o conceito Protocolo delibera as ações de colaboração durante as sessões colaborativas. Neste contexto, as regras representadas pelo conceito Protocolo regem as atividades dos agentes, que são suas ações de contribuição e as dependências de fluxo dos recursos. O Protocolo é um tipo de descrição normativa (*Normative Description*) que tem a descrição da definição dos Papéis Colaborativos e seus objetivos adotados através de Compromissos Fechados.

Tendo apresentado o modelo conceitual que provê a base teórica da pesquisa, na próxima seção é descrito o ambiente físico utilizado na sua implementação.

2.4. A Web-based Groupware Service-Oriented Architecture (WGWSOA)

O desenvolvimento de aplicações de *groupware* distribuído é complexo. Os dados e procedimentos da interação não são os únicos aspectos considerados no desenvolvimento deste tipo de aplicação. Requisitos de suporte à comunicação entre aplicações distintas, o apoio à interação entre os participantes, as políticas e as regras dos processos colaborativos devem ser contemplados para atender de maneira satisfatória (DAVID e MACIEL, 2009).

Em ambientes colaborativos, o contexto em que ocorrem as interações está em constante mudança, coisas como o objeto de interesse dos participantes, a necessidade das atividades e o ambiente em que ocorre a interação precisam ser representados por elementos dinâmicos e flexíveis. Um suporte adequado a este tipo de requisito é exigido em um *middleware* que apoie o desenvolvimento de aplicações distribuídas.

Essa foi a principal questão abordada pela WGWSOA (DAVID e MACIEL, 2009): propor uma solução que permita a comunicação entre aplicações de grupo distribuídas e que seja suficientemente flexível para se adaptar ao contexto dinâmico que tais ferramentas estão inseridas. O objetivo da WGWSOA é apoiar a construção de *groupwares* distribuídos e oferecê-los como serviços colaborativos. A Figura 12 ilustra os módulos desta arquitetura.

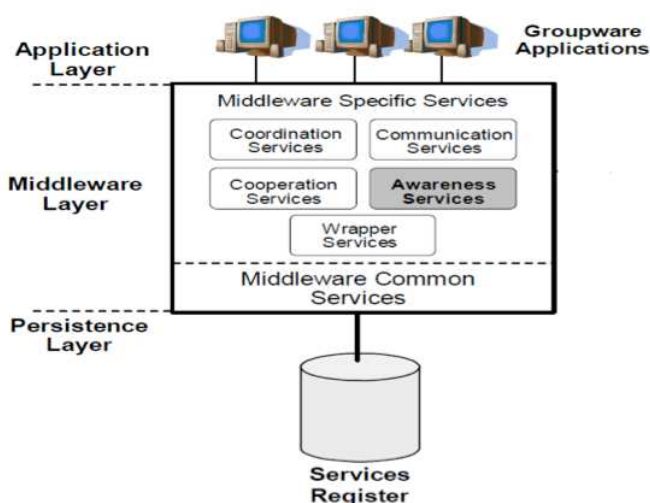


Figura 12: Arquitetura da WGWSOA
 Fonte: DAVID e MACIEL (2009).

A WGWSOA é centrada no desenvolvimento de um arcabouço de *middleware* dividido em três camadas, utilizando a abordagem de Arquitetura Orientada a Serviço (SOA – Service Oriented Architecture). SOA é uma abordagem de tecnologia da informação ou de uma estratégia em que as aplicações fazem uso de (talvez mais precisamente, dependem de) serviços disponíveis em uma rede, como a World Wide Web. A implementação de uma arquitetura orientada a serviços pode envolver o desenvolvimento de aplicações que utilizam os serviços ou fazer aplicações disponíveis como serviços para que outros aplicativos possam usar esses serviços (ERL, 2005). Os princípios de SOA, tais como a especificação de um contrato, o acoplamento fraco, a reutilização, a composição, a autonomia e a ausência de estado, foram empregados para obter maior interoperabilidade entre as aplicações de *groupware*. Em seu modelo de referência, a OASIS (2006) defende que serviço é um mecanismo para habilitar o acesso a um conjunto de uma ou mais funcionalidades, onde o acesso é provido usando uma interface que é regida por restrições e políticas especificadas na descrição de serviço.

As três camadas conceituais do framework são: aplicação, *middleware* e persistência. A primeira oferece as funcionalidades de aplicações de *groupware* como chat e fórum, por exemplo. A segunda oferece serviços específicos e comuns contidos no domínio de colaboração que aplicam os conceitos de comunicação, coordenação, cooperação e percepção, por exemplo: grupo de usuários, papéis, atividades, ações e lista de usuários. A terceira camada atua especificamente na gravação dos tipos de serviços oferecidos pela camada de *middleware*, sua eventual descoberta para reuso, assim como o registro dos dados das interações entre participantes nas sessões colaborativas.

No domínio de colaboração, a WGWSOA disponibiliza diversos serviços de suporte para cada uma das perspectivas do Modelo 3C: Comunicação (serviços de Chat, Fórum, envio de SMS, Voip, envio de e-mail), Coordenação (serviços de Gerência de grupos e atividades), Cooperação (serviços de Edição compartilhada de documentos, Gerência de Repositório para Persistência, e Gerência de Workspace compartilhado, para controle de edição de artefatos), e Percepção (serviços de Notificação de Evento, e Visualização de dados como relatórios de participação).

Os serviços de gerenciamento de grupos e de projetos foram desenvolvidos para permitir a manutenção destes. Com estes serviços é possível criar um novo grupo ou projeto e atualizar suas informações. Os serviços de Gerência de Repositório para Persistência é composto por funcionalidades que permitem armazenar informações relativas aos próprios serviços e também os dados que os serviços manipulam. Os serviços de Gerência de Workspace utilizam os serviços de Percepção para apoiar a contextualização do ambiente em que os utilizadores do *groupware* estão trabalhando.

A percepção, definida no Modelo 3C, é tratada na WGWSOA como um serviço específico de *middleware*. Seu objetivo é apoiar a percepção sobre ações ocorridas, tanto nas aplicações quanto nos serviços da WGWSOA. Os autores chamaram este serviço de AW2SOA (*Awareness to Service-Oriented Architecture*). O serviço é capaz de coletar informações contextuais de sua execução de maneira não intrusiva, aplicando técnicas do paradigma de orientação a aspectos. Para colher tais informações, a AW2SOA tem os serviços de Eventos (monitora o acionamento dos demais serviços da WGWSOA), Notificação (informa aos cadastrados sobre as ocorrências dos eventos), Consulta (permite busca de eventos específicos), Persistência (armazena eventos), Gerenciador de Perfil (manipula as configurações dos usuários) e o Sensor (faz medições para apoiar a coordenação das atividades). Eventos ocorridos na WGWSOA são capturados pelo Sensor, enquanto eventos

ocorridos nas aplicações que usam a WGWSOA devem ser explicitamente informados para o serviço de Eventos. Este serviço funciona como forma de entrada dos eventos no ambiente de *groupware* no qual a WGWSOA está atuando.

Capítulo 3 – Colaboração em Processos Intensivos em Conhecimento

Este capítulo tem como objetivo apresentar os trabalhos relacionados a apoio computacional das tarefas colaborativas de processos intensivos em conhecimento. Os trabalhos relacionados são agrupados em duas categorias de acordo com as partes que constituem esta pesquisa. A primeira discute taxonomias para *groupware*. A segunda descreve o suporte computacional para as atividades que se inserem os processos intensivos em conhecimento que são realizadas colaborativamente.

3.1. Taxonomias para *Groupware*

Tarefas em processos são orientadas a atingir um objetivo no negócio, executadas de maneira colaborativa e ocorrem nas interações entre os participantes (STROHMAIER e TOCHTERMANN, 2005). Nos processos intensivos em conhecimento há uma dependência ainda maior por colaboração devido ao alto nível de conhecimento tácito e explícito compartilhado dentro das atividades nestes processos, porquanto eles evoluem na medida em que são executados (FRANÇA, 2012).

Atividades intensivas em conhecimento são realizadas quando uma instância de um processo desta natureza é invocado. Dentre estas atividades, algumas são colaborativas. Conseqüentemente, pelo menos dois agentes estarão envolvidos no processo. Tais agentes trocam informações, operam de maneira conjunta e organizam suas atividades de acordo com o papel-protocolo-social demandado pela execução das tarefas.

Com a intenção de responder quais as ferramentas encontradas na WGWSOA são mais adequadas para apoiar computacionalmente cada atividade executada em particular, uma classificação dos objetivos de suas ferramentas de grupo se faz necessária.

A partir desta classificação é possível definir um mapeamento entre as classes da KIPO e os serviços colaborativos oferecidos pelo *middleware* WGWSOA. Instâncias de classes da KIPO permitem a modelagem de cada elemento descrito em um processo intensivo em conhecimento. Elementos de decisão são modelados para representar as atividades tomadas por agentes que ora atuam com agentes de inovação resolvendo questões que surgem durante a execução do processo, e em outro momento atuam como agentes de impacto responsável pelo processo. Os artefatos utilizados como recursos em uma sessão colaborativa prescrevem os itens de trabalho que auxiliam a cooperação entre agentes. Todos estes conceitos classificam as atividades executadas em um processo.

O elemento de ligação entre estas atividades e as ferramentas de grupo motivou uma pesquisa por classificar as características de ferramentas que apoiem as atividades em grupo. Uma formalização destas características é proposta em algumas taxonomias encontradas na literatura de classificação de ferramentas de *groupware*.

Contudo, não existe um consenso entre pesquisadores sobre uma taxonomia única que defina tais ferramentas. CORREA (2011) investigou o estado da arte de ferramentas de *groupware* e apontou as taxonomias encontradas. As buscas sobre o assunto resultaram em 1840 artigos. Foram selecionados artigos dos anos de 2003 ao ano de 2010. Foram elencados pelo autor quatro tipos distintos de taxonomias. A Matriz Espaço-Tempo (ELLIS *et al.*, 1991), Domínios de Aplicação (POLTROCK & GRUDIN, 1998), Modelo 3C (FUCKS, 2005) e Taxonomias híbridas (BOLSTAD & ENDSLEY, 2003).

Analogamente, em seu artigo, CALVÃO *et al.* (2012) apresenta uma taxonomia para classificar sistemas quanto à perspectiva de comunicação. Mais especificamente, os sistemas de conversação mediados por computador são o alvo da pesquisa dos autores. A taxonomia desenvolvida procura organizar os diversos tipos de ferramentas de *groupware* em classes de maneira a, inequivocamente, apontar semelhanças e diferenças entre elas. Observou-se que esta taxonomia poderia servir de base para um sistema de decisão computacional recomendar, com bastante precisão, ferramentas de *groupware*. Para esta pesquisa, buscam-se elementos que atendam os critérios de coincidência entre objetivos de uma ferramenta de grupo e características da atividade colaborativa intensiva em conhecimento.

A taxonomia prescrita por CALVÃO *et al.* (2011) monta uma árvore decisória que como elemento folha direciona para uma ou mais ferramentas de grupo capaz de atender ao requisito de comunicação. Os critérios-características da taxonomia são nominalmente: sincronia dos interlocutores, quantidade de interlocutores, relação entre os interlocutores,

linguagem de comunicação, estruturação do discurso e tamanho da mensagem. Estes critérios apontam diretamente para um ou mais sistemas de conversação em grupo. Cada um dos critérios da taxonomia formulando uma árvore de decisão pode ser vistos na figura 13.

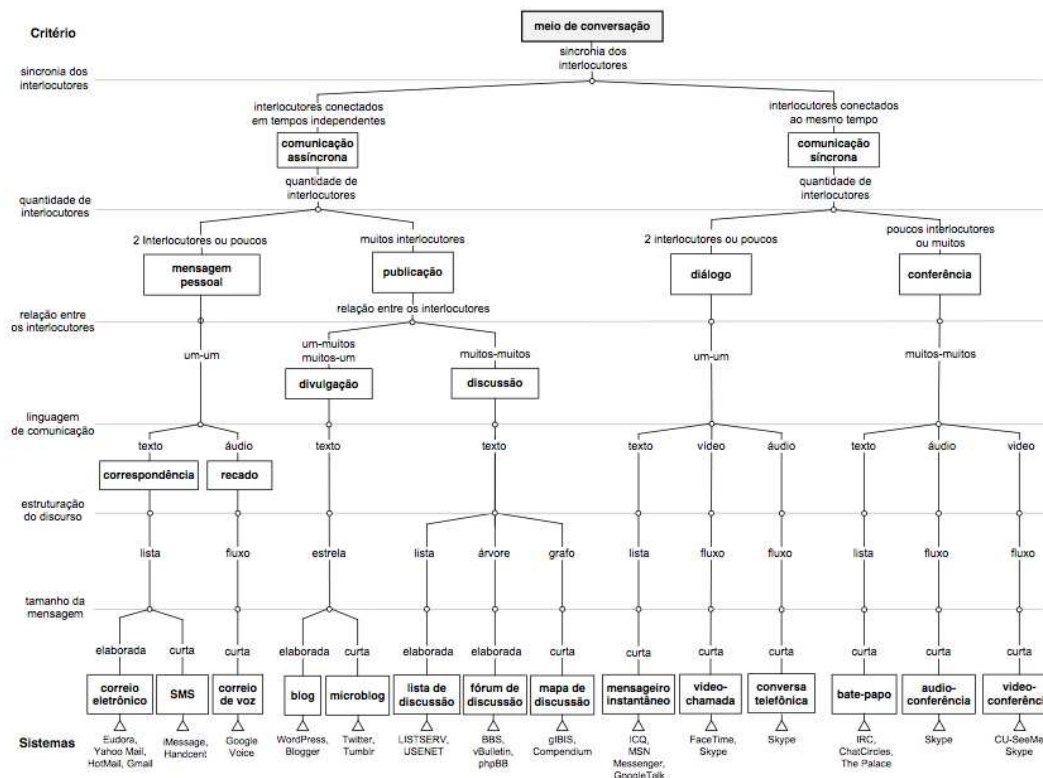


Figura 13: Árvore de classificação da taxonomia dos meios de comunicação
 Fonte: CALVÃO et al. (2012).

A proposta desta taxonomia pode ser aplicada em diferentes cenários. Tanto para a especificação dos meios de conversação a ser implementados em sistemas colaborativos quanto para apoiar a caracterização e a compreensão dos diferentes meios de conversação. Outra utilidade apontada por CALVÃO *et al.* (2012) em seu trabalho foi a de sugerir meios de conversação ainda não estabelecidos culturalmente, aplicando sua Árvore de Classificação.

Os critérios utilizados nos sistemas colaborativos que foram propostos por CALVÃO *et al.* (2012) são: sincronia entre interlocutores, quantidade de interlocutores e relação entre interlocutores; critérios relacionados à organização e à forma das mensagens - estruturação do discurso, tamanho da mensagem e linguagem de comunicação.

- **Classificação espaço-tempo** – classifica *groupware* em função do espaço (local ou distribuído) e do tempo (interação síncrona ou assíncrona). A comunicação em função do espaço estabelece que os interlocutores possam estar em locais diferentes ou localizados para se comunicar. A classificação relativa ao tempo indica que uma mensagem enviada por um interlocutor é recebida praticamente no mesmo instante por outro interlocutor em uma

comunicação síncrona. Por outro lado, a mensagem é armazenada para ser recuperada posteriormente por outro interlocutor na comunicação assíncrona.

- **Quantidade de interlocutores** – a comunicação pode ocorrer com poucos interlocutores ou com muitos interlocutores. Uma comunicação síncrona com poucos (dois) interlocutores é classificada como um diálogo, enquanto uma comunicação síncrona com muitos interlocutores é classificada como uma conferência. Já uma comunicação assíncrona com poucos (dois) interlocutores é classificada como uma mensagem pessoal, enquanto uma comunicação assíncrona com muitos interlocutores é classificada como uma publicação.

- **Relação entre interlocutores** – *groupware* pode relacionar interlocutores um-um (ou pessoa-pessoa), um-muitos e muitos-muitos. Seguindo a hierarquia vista na Figura 5, mensagem pessoal é classificada como relacionamento um-um e pode ainda ser classificada como correspondência ou recado. As publicações relacionam muitos interlocutores e pode também receber a classificação de divulgação (um-muitos/muitos-um) ou discussão (muitos-muitos). Os diálogos relacionam os interlocutores como um-um e por fim a conferência relaciona muitos interlocutores.

- **Linguagem de comunicação** – a linguagem do sistema de comunicação pode ser do tipo texto, áudio ou vídeo. Divulgação, discussão e correspondência são tipos de *groupware* que utilizam linguagem de texto. Recado, nesta classificação utiliza áudio. Diálogo e conferência podem utilizar os três tipos de linguagem.

- **Estruturação do discurso** – *groupware* pode ser classificado relativo à sua estrutura de discurso como funções contínuas, lista, árvore, estrela e grafo. Mensagem transmitida por meio de áudio e vídeo é organizada em fluxo contínuo, já mensagem enviada após ter sido formulada, como no correio-eletrônico e bate-papo, pode ser organizada tanto de forma linear (lista), como centralizada (estrela), como hierárquica (árvore) ou em rede (grafo), o que possibilita a convergência de uma discussão.

- **Tamanho da mensagem** – *groupware* pode ter mensagens mais curtas ou mais longas. Alguns podem exigir a limitação do tamanho da mensagem, como é o caso do *Twitter* que impõe uma limitação em 140 caracteres a suas mensagens.

3.2. Suporte Computacional para Atividades Colaborativas

ANDERSON, BIDER e PERJONS (2004) descrevem os desafios da comunicação interna entre colegas e gestores durante a execução de processos de negócio dentro das organizações.

Em seu artigo os autores fundamentam-se na suposição de que os envolvidos na comunicação estão compartilhando informações pertencentes a uma instância de um processo mesmo quando eles não estejam cientes disso. Eles apresentam uma solução prática com uma abordagem orientada a estado, derivada da teoria matemática de sistemas que lida com processos de negócio (KHOMYAKOV, BIDER, 2000), para apoiar a automação da comunicação daqueles processos, incluindo os fortemente e os fracamente estruturados. A abordagem de solução dos autores foca na criação de um sistema cliente/servidor cuja arquitetura exige que os participantes criem tarefas de comunicação manualmente para colaborar em alguma atividade dentro do processo. O que diferencia a pesquisa atual da proposta acima referenciada é que na primeira, a identificação das atividades colaborativas é feita automaticamente a partir de heurísticas sobre os relacionamentos no modelo do processo e, são providos amplos serviços colaborativos que compreendem não somente a comunicação, mas a cooperação, coordenação e percepção.

O objetivo de STOITSEV *et al.* (2009) é o suporte computacional para executar atividades colaborativas de fim-a-fim em processos de negócio. Como motivação para a proposta, os autores afirmam que em processos de negócio não existem meios estruturados de armazenamento do conhecimento, além disso, acrescentam que a recuperação e as melhores práticas de troca deste conhecimento não são adotadas. Em sua proposta, STOITSEV *et al.* (2009) propõem o desenvolvimento de uma ferramenta de gestão de tarefas integrada ao email e associada à lista de tarefas (*todo-list*) com suporte a definição, adaptação e reuso de um modelo de processo fracamente estruturado como os que ocorrem em PIC. Embora emails e lista de tarefas sejam muito comuns entre usuários finais oferecendo baixa resistência na sua aceitação como solução para suporte de processo fracamente estruturado, esta solução limita a abrangência da colaboração em processos intensivos em conhecimento por não envolver outras ferramentas que fazem parte do dia a dia de colaboradores que realizam seu trabalho em grupo.

PAPAVASSILIOU *et al.* (2002) utiliza uma ontologia para propiciar a especificação do contexto de criação e uso do conhecimento em processos intensivos em conhecimento e tangencia a proposta deste artigo. Em sua abordagem, Papavassiliou *et al.* (2002) apresentam uma ferramenta para modelagem de processos fracamente estruturados (PICs) e focam na gestão do conhecimento envolvida nestes processos. Embora a abordagem forneça um método para a modelagem de processos dirigidos por uma ontologia e uma *engine* para automatização

de atividades de processos intensivos em conhecimento, ele não foca na colaboração existente entre os participantes dos processos.

GEROSA *et al.* (2005) afirmam que sistemas colaborativos são complexos de se desenvolver devido à sua natureza evolutiva. Ainda que o software seja construído de uma maneira ótima, novos problemas e situações o tornam inadequado com o passar do tempo. O desenvolvimento baseado em componentes se torna uma estratégia viável para solucionar esse problema. Uma arquitetura componentizada possibilita conectar e desconectar componentes de forma a montar uma ferramenta adequada aos propósitos específicos de cada grupo e tarefa. Nesta abordagem a colaboração em processos de negócio não é levada em consideração.

É possível notar que os sistemas colaborativos estão presentes em diversas propostas de solução que visam apoiar as atividades realizadas em PIC. Contudo, uma abordagem que leve em consideração as características deste tipo de processo (complexidade, variabilidade, tempo de vida volátil do conhecimento criado, frequente mudança de objetivos, diversidade nas fontes de informação) e que facilite a escolha dinâmica de ferramentas colaborativas para apoiar a execução das atividades do PIC é um desafio apontado pela literatura. Uma abordagem que leve em consideração tais características reduz a complexidade na escolha de sistemas colaborativos em cenários em que especialistas nestes sistemas não estão disponíveis.

Capítulo 4 – Proposta de Solução: XCuteKIP

Este capítulo propõe a XCuteKIP, que tem como objetivo responder a questão delineada nesta pesquisa. A XCuteKIP é composta por um método e por uma arquitetura para automatizar a escolha entre as alternativas de serviços de colaboração a partir da definição de um PIC em particular. Desta maneira, será possível recomendar automaticamente serviços adequados para apoiar atividades colaborativas de um processo intensivo em conhecimento. Uma versão preliminar da XCuteKIP foi proposta em MOURA (2013).

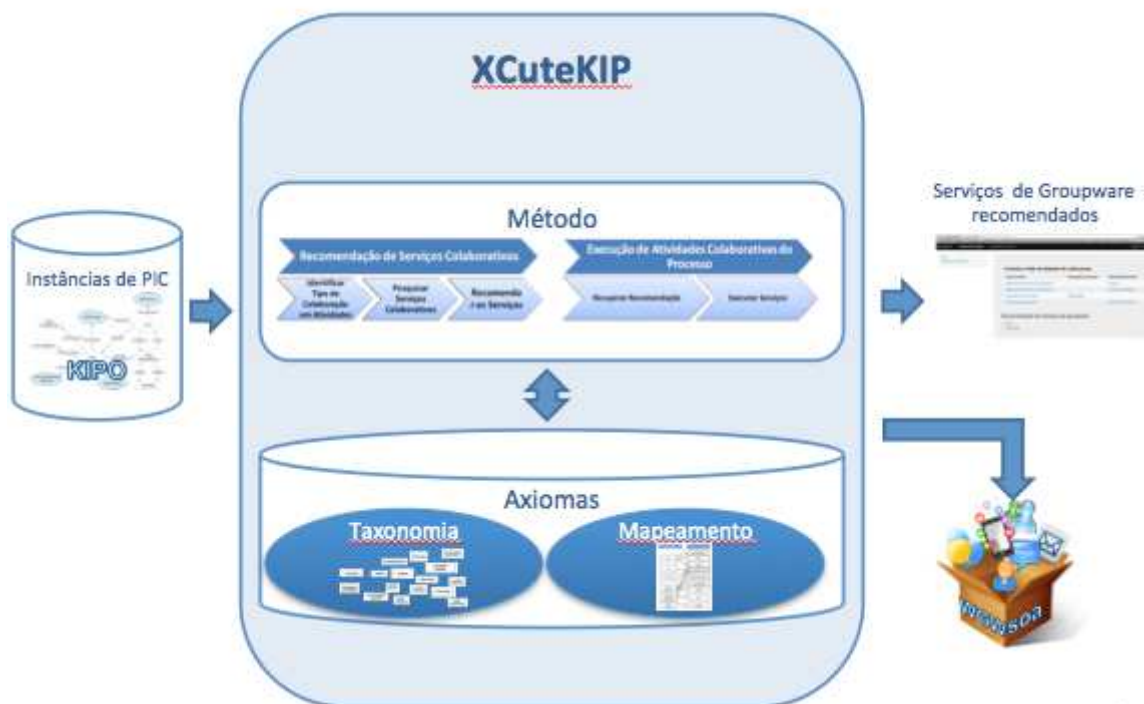


Figura 14: Visão Geral da XCuteKIP

Para adequadamente conectar as atividades colaborativas aos serviços colaborativos, é proposto um arcabouço com uma arquitetura orientada a serviço e um método para descoberta e invocação de serviços de maneira semi-automática (Figura 14).

Para ilustrar o método, tome-se como exemplo o processo de levantamento de requisitos. O processo é composto por diversas atividades. Dentre estas atividades, algumas são colaborativas. O XcuteKIP aplicará três axiomas definidos em seu processo, e apresentados na sessão 4.2, para identificar quais são as atividades colaborativas. A atividade de “Levantar lista de desejos” será incluída junto a outras nesta lista.

Em seguida, mais quatorze algoritmos serão aplicados sobre cada atividade colaborativa. Na atividade de “Levantar lista de desejos”, o quarto dos quatorze algoritmos aplicado retornará como resposta que esta atividade é assíncrona. A quantidade de interlocutores, segundo o método XcuteKIP, é “Muitos” e o relacionamento dos interlocutores é “Um para Muitos”. Após executados todos os algoritmos, o serviço recomendado para esta atividade, “Levantar lista de desejos”, será “Correio Eletrônico”.

Assume-se que as atividades são representadas como instâncias das classes da KIPO e o conjunto de serviços de apoio à colaboração virão da WGWSOA. O relacionamento entre as atividades instanciadas com as classes da KIPO com os serviços colaborativos serão definidos através de um mapeamento, que utilizará uma taxonomia para categorizar os serviços da WGWSOA. Foi realizada uma extensão do banco de dados do middleware WGWSOA, incluindo tabelas para persistir os tipos de serviços colaborativos, e uma coluna na tabela de serviço, referenciando esta tabela.

No exemplo anterior, onde o tipo de serviço “Correio Eletrônico” foi recomendado para a atividade “Levantar lista de desejos”, a XcuteKIP utilizando a api implementada com serviços recupera o serviço de “Email” da WGWSOA.

A escolha da WGWSOA deve-se ao fato de que construir sistemas colaborativos a partir do zero é muito complexo por diversos fatores, dentre eles sua natureza evolutiva (GEROSA, 2006). A WGWSOA prevê a evolução e disponibilização de novos serviços colaborativos de maneira isolada da execução dos processos intensivos em conhecimento. À medida que novos serviços são criados, a XcuteKIP os identifica e os disponibiliza para que as atividades colaborativas dos PIC possam fazer uso deles.

A taxonomia usada na classificação dos serviços disponibilizados pela arquitetura é detalhada na seção 4.1. Os axiomas propostos que formalizam o aspecto de colaboração em um PIC são apresentados na seção 4.2. A seção 4.3 apresenta o mapeamento proposto, que

estabelece a correspondência entre atividades colaborativas e serviços colaborativos. Finalmente, a seção 4.4 apresenta o método proposto para recomendação de serviços de colaboração, seguido da arquitetura que prevê a configuração e execução das atividades colaborativas de um PIC.

4.1. Classificação dos serviços colaborativos da WGWSOA

A taxonomia de CALVÃO *et al.* (2012), apresentada na seção três, permite a categorização de sistemas de comunicação. Esta taxonomia será utilizada nesta pesquisa devido a sua facilidade de uso para classificação dos serviços colaborativos disponibilizados pela arquitetura WGWSOA. Os parâmetros da taxonomia agem como filtro para categorizar cada serviço da arquitetura. Esta categorização é útil principalmente para serviços que não sejam do senso comum, como no caso de um mapa de discussão. Por exemplo, utilizando a taxonomia como filtro é possível observar que um mapa de discussão é um serviço assíncrono, que tem muitos interlocutores e sua estrutura de mensagens é realizada em forma de grafo.

Neste sentido, esta taxonomia é utilizada por desenvolvedores de serviços colaborativos que, a cada novo serviço desenvolvido, utiliza os parâmetros como filtro classificatório para registrar este serviço com uma das categorias definidas na taxonomia. A XCuteKIP provê uma funcionalidade (seção 4.4.2) que facilita ao desenvolvedor de serviços colaborativos classificar novas funcionalidades desenvolvidas de acordo com esta taxonomia, consequentemente, adicionando à lista de serviços oferecidos da ferramenta XCuteKIP.

4.2. Especificação formal de Atividades Colaborativas em um PIC

Para que as atividades colaborativas que fazem parte de um PIC possam ser automaticamente identificadas pela XCuteKIP, elas devem ser precisamente especificadas. No presente trabalho, assume-se que um PIC esteja modelado segundo o meta-modelo da KIPO e, portanto, o conceito de a atividade colaborativa faz uso das classes e relacionamentos previstos na KIPO.

Segundo a KIPO, uma Socialização (*Socialization*) é um tipo de Interação Comunicativa (*Communicative Interaction*) composta pela comunicação e percepção, desenvolvida por um agente. Uma comunicação (*Communication*) é um tipo de ato comunicativo que tem por conteúdo proposicional uma mensagem (*Message*), que compõe o

fluxo de mensagem do PIC. Socialização também é um tipo de sessão colaborativa (*Collaborative Session*).

A partir desta conceitualização, é possível derivar a seguinte definição para atividade colaborativa, formalizada através do axioma A1:

Axioma A1: Todos os atos comunicativos realizados em uma *Socialization* são atividades colaborativas.

Formalmente:

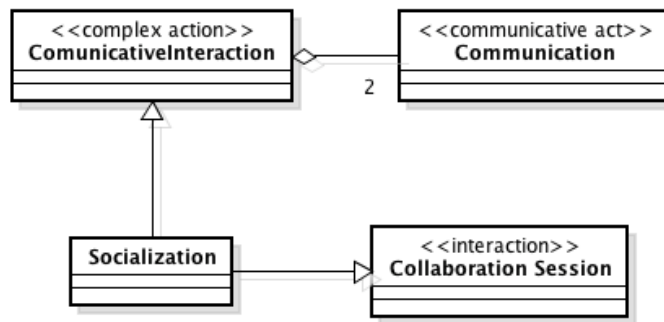
$$\text{CollaborativeActivity}(ca) \leftarrow \text{CommunicativeAct}(ca) \wedge \text{CommunicativeInteraction}(ci) \wedge \text{Socialization}(ci) \wedge \text{composed_by}(ci, ca) \wedge \text{happens_in}(ca, ci)$$


Figura 14: Socialização
Fonte: FRANÇA (2012).

Uma Sessão Colaborativa (*Collaboration Session*) (Figuras 14 e 15) é uma Interação (*Interaction*) entre agentes, deliberada por (*ruled by*) Acordos Colaborativos (*Collaborative Agreements*). Dentro das sessões colaborativas, agentes instanciam papéis colaborativos (*Collaborative Roles*) que são caracterizados por um Tipo de Compromisso Fechado (*Closed Commitment Universal*) que firmam entre si para desempenhar suas atividades colaborativas. Cada atividade realizada pelos agentes na Sessão Colaborativa é denominada Ação de Contribuição (*Action Contribution*) (Figura 15) e tem pelo menos dois participantes. Por esta conceitualização, define-se que:

Axioma A2: Todas as atividades realizadas em uma *Collaboration Session* são colaborativas.

Formalmente:

$$\text{CollaborativeActivity}(cs) \leftarrow \text{CollaborationSession}(cs) \wedge \text{ActionContribution}(ac) \wedge \text{composed_by}(cs, ac)$$

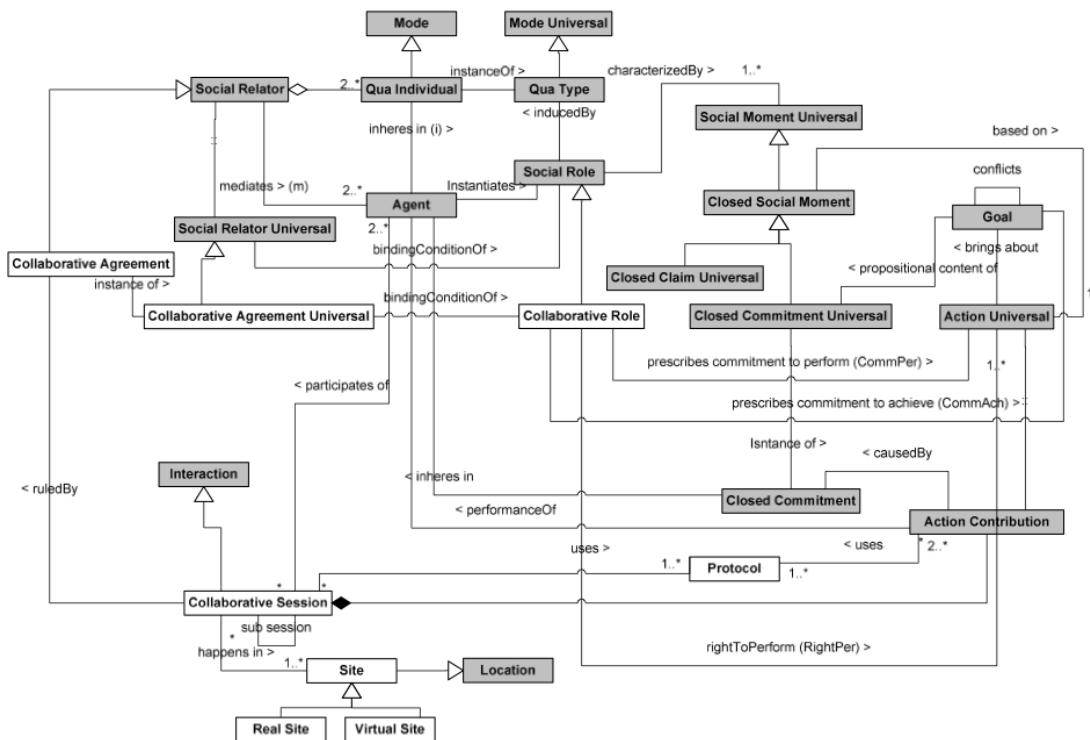


Figura 15: Ontologia de colaboração
 Fonte: OLIVEIRA (2009).

Toda Atividade Intensiva em Conhecimento (*Knowledge Intensive Activity*) pode envolver uma Decisão (*Decision*) e tem como responsável um Agente de Inovação (*Innovation Agent*) (Figura 16). A tomada de uma Decisão considera (*considers*) várias Alternativas (*Alternative*). As alternativas discutem Especialidades (*Speciality*) dos Agentes de Inovação (*InnovationAgent*). O que se conclui que um Agente de Inovação é especialista em uma área de conhecimento quando tem alguma Especialidade (*Speciality*) nesta área. Nesta conceitualização, define-se que:

Axioma A3: Toda atividade intensiva em conhecimento é uma atividade colaborativa quando envolver decisão e o agente de inovação responsável pela atividade não for especialista nas áreas de conhecimento das alternativas associadas à decisão.

Formalmente:

$$\begin{aligned}
 \text{CollaborativeActivity}(kia) \leftarrow & \text{KnowledgeIntensiveActivity}(kia) \wedge \text{Decision}(d) \wedge \\
 & \text{Alternative}(a) \wedge \text{Speciality}(s) \wedge \text{InnovationAgent}(ia) \wedge \text{InnovationAgent}(ia') \wedge \\
 & \text{involves}(kia, d) \wedge \text{considers}(d, a) \wedge \text{associated_with}(a, s) \wedge \text{has}(ia', s) \wedge ia \neq ia' \\
 & \text{is_discussed_in}(a, s)
 \end{aligned}$$

Uma vez definidos os axiomas que permitem identificar as atividades colaborativas em um PIC modelado com base na KIPO, nesta seção são definidos os critérios para identificação dos requisitos de colaboração relacionados a estas atividades. Tais requisitos também são formalizados através de axiomas, para uma definição precisa e livre de ambiguidades. Os axiomas aqui especificados seguem a ordem hierárquica definida pelo trabalho de CALVÃO *et al.* (2012) e apresentada na sessão 3.1.

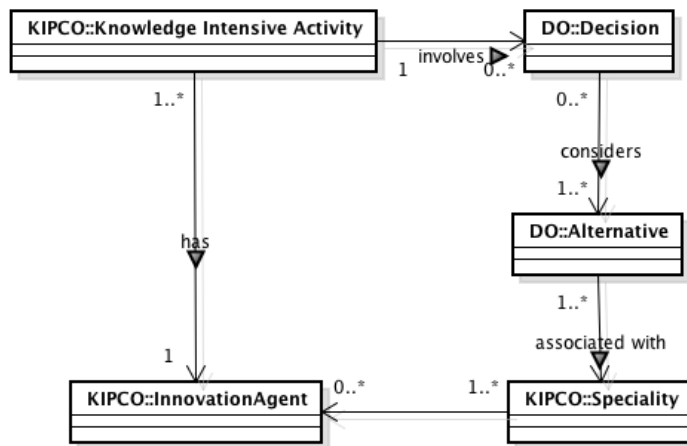


Figura 16: Ontologia de Decisão
 Fonte: FRANÇA (2012).

Sincronia: Segundo a KIPO, uma Interação Comunicativa (*Communicative Interaction*) é composta por pelo menos um ato comunicativo (*Communicative Act*) e uma percepção (*Perception*). Tanto os atos comunicativos quanto a percepção são tratados como eventos. Os eventos contêm a propriedade *synchronized with*, e define um autorrelacionamento com outro evento. Quando esta propriedade tem valor, significa que os dois eventos relacionados ocorrem ao mesmo tempo. Com isso, é possível definir que:

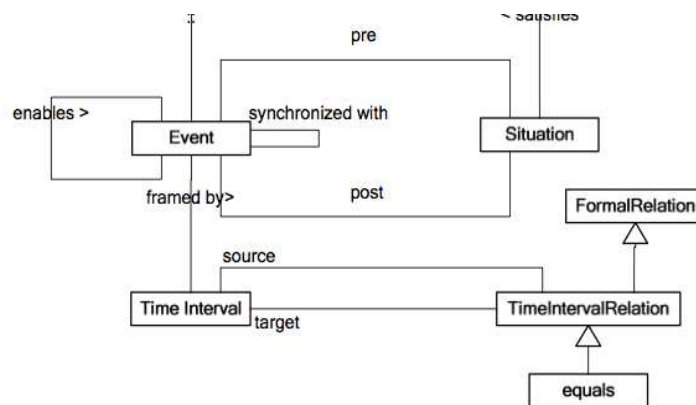


Figura 17: Sincronia de eventos
 Fonte: FRANÇA (2012).

Axioma A4: Sejam duas atividades a e a' que compõem a mesma instância de um PIC. Sabendo que uma atividade é uma especialização de um evento, a e a' são consideradas atividades síncronas se a tiver a propriedade *synchronized with* configurada com uma referência para a' .

Formalmente:

$$\text{synchronous}(e, e') \leftarrow \text{Event}(e) \wedge \text{Event}(e') \wedge \text{KnowledgeIntensiveProcess}(\text{kip}) \wedge \text{composed_by}(\text{kip}, e) \wedge \text{composed_by}(\text{kip}, e') \wedge \text{synchronized_with}(e, e')$$

Com este axioma é possível inferir tanto a sincronia quanto a assincronia, assumindo-se que duas atividades que não sejam síncronas são necessariamente assíncronas. Quando o

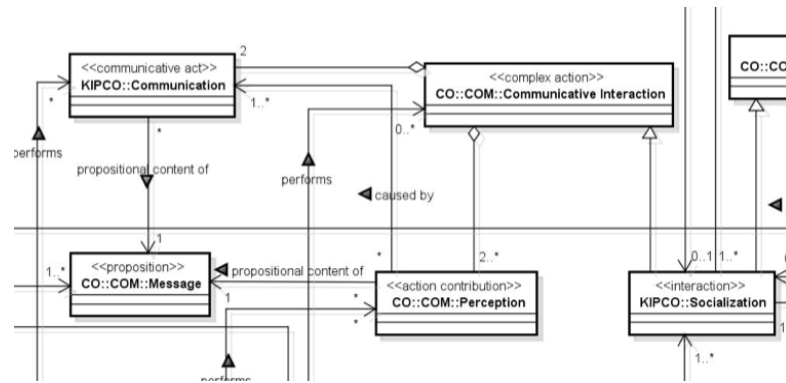


Figura 18: Interação comunicativa
 Fonte: FRANÇA (2012).

Quantidade de interlocutores: A cardinalidade dos interlocutores relaciona-se com quantos agentes realizam atos comunicativos dentro de uma Interação Comunicativa (Figura 19). Um ato comunicativo (*Communicative Act*) é realizado por um agente (*Agent*) na interação comunicativa (*Communicative Interaction*) assumindo o papel de emissor. Outro agente participante da mesma interação comunicativa assume o papel de Receptor (*Receiver*) quando realiza a atividade de percepção (*Perception*) da mensagem enviada pelo emissor. Os agentes podem ser individuais ou sociais. Um agente individual pode ser, por exemplo, um jogador de futebol, um contador, um analista de requisitos. Os agentes sociais referem-se a um grupo, uma organização ou uma sociedade. Exemplos de agentes sociais são o Clube de Futebol Flamengo e as Nações Unidas.

Nesta pesquisa, definimos cardinalidade “um” para agentes individuais e cardinalidade “muitos” para agentes sociais.

Com estas premissas é possível especificar as seguintes definições:

Axioma A5: Uma interação comunicativa tem cardinalidade “um” de interlocutores se um agente individual (que não seja um agente social) estiver participando nesta interação comunicativa.

Formalmente:

$$\text{interlocutorCardinality}(ci, \text{“um”}) \leftarrow \text{CommunicativeInteraction}(ci) \wedge \text{part_of}(ci, ca) \wedge \text{communicativeAct}(ca) \wedge \text{performanceOf}(ca, a) \wedge \text{agent}(a) \wedge \text{not socialAgent}(a)$$

Axioma A6: Uma interação comunicativa tem cardinalidade “muitos” de interlocutores se um agente social estiver participando desta interação comunicativa.

Formalmente:

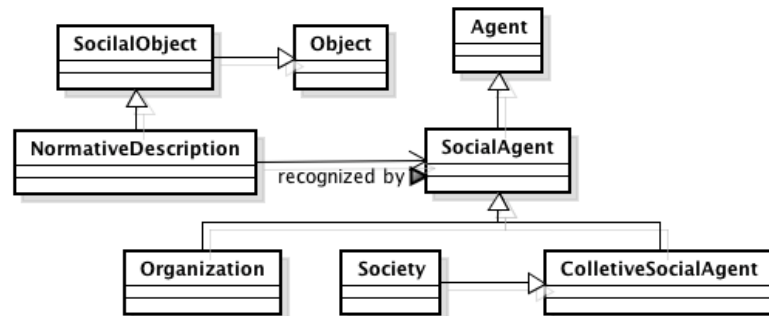
$$\text{interlocutorCardinality}(ci, \text{“muitos”}) \leftarrow \text{CommunicativeInteraction}(ci) \wedge \text{part_of}(ci, ca) \wedge \text{communicativeAct}(ca) \wedge \text{performanceOf}(ca, a) \wedge \text{agent}(a) \wedge \text{socialAgent}(a)$$


Figura 19: Agentes Sociais
Fonte: FRANÇA (2012).

Relação entre interlocutores: A taxonomia de CALVÃO *et al.* (2012) delimita que interlocutores podem se relacionar em uma comunicação observando as seguintes relações entre si: um-um (ou pessoa-pessoa), um- muitos e muitos-muitos. Baseando-se na definição já referida nos axiomas A5 e A6 é possível compor as relações entre interlocutores analisando a comunicação ocorrida entre os participantes das interações comunicativas (Figura 19 e 20).

Axioma A7: Uma interação comunicativa tem uma relação “um-um” entre interlocutores se um agente individual, que está presente em uma interação comunicativa, tem como interlocutor outro agente que também seja um agente individual.

Formalmente:

$$\begin{aligned} \text{interlocutorRelator}(\text{ci}, \text{“um-um”}) \leftarrow & \text{CommunicativeInteraction}(\text{ci}) \wedge \text{part_of}(\text{ci}, \text{ca}) \wedge \\ & \text{communicativeAct}(\text{ca}) \wedge \text{performanceOf}(\text{ca}, \text{a}) \wedge \text{agent}(\text{a}) \wedge \text{not socialAgent}(\text{a}) \\ \wedge \text{part_of}(\text{ci}, \text{p}) \wedge \text{Perception}(\text{p}) \wedge & \text{performanceOf}(\text{p}, \text{a}') \wedge \text{agent}(\text{a}') \wedge \text{not} \\ & \text{socialAgent}(\text{a}') \end{aligned}$$

Axioma A8: Uma interação comunicativa tem uma relação “um-muitos” entre interlocutores se um agente individual que está presente em uma interação comunicativa tem como interlocutor agente que seja do tipo social.

Formalmente:

$$\begin{aligned} \text{interlocutorRelator}(\text{ci}, \text{“um-muitos”}) \leftarrow & \text{CommunicativeInteraction}(\text{ci}) \wedge \text{part_of}(\text{ci}, \text{ca}) \\ \wedge \text{communicativeAct}(\text{ca}) \wedge \text{performanceOf}(\text{ca}, \text{a}) \wedge & \text{agent}(\text{a}) \wedge \text{not socialAgent}(\text{a}) \wedge \\ \text{part_of}(\text{ci}, \text{p}) \wedge \text{Perception}(\text{p}) \wedge \text{performanceOf}(\text{p}, \text{a}') \wedge & \text{agent}(\text{a}') \wedge \text{socialAgent}(\text{a}') \end{aligned}$$

Axioma A9: Uma interação comunicativa tem uma relação “muitos-muitos” entre interlocutores se um agente social está presente em uma interação comunicativa e tem como interlocutor outro agente do tipo social.

Formalmente:

$$\begin{aligned} \text{interlocutorRelator}(\text{ci}, \text{“muitos-muitos”}) \leftarrow & \text{CommunicativeInteraction}(\text{ci}) \wedge \text{part_of}(\text{ci}, \\ \text{ca}) \wedge \text{communicativeAct}(\text{ca}) \wedge \text{performanceOf}(\text{ca}, \text{a}) \wedge & \text{agent}(\text{a}) \wedge \text{socialAgent}(\text{a}) \wedge \\ \text{part_of}(\text{ci}, \text{p}) \wedge \text{Perception}(\text{p}) \wedge \text{performanceOf}(\text{p}, \text{a}') \wedge & \text{agent}(\text{a}') \wedge \text{socialAgent}(\text{a}') \end{aligned}$$

Linguagem de comunicação:

A taxonomia de CALVÃO *et al.* (2012) define como tipos de linguagem de comunicação os seguintes itens: vídeo, áudio e texto. As mensagens (*Message*) são associadas aos atos comunicativos (*Communicative Act*), que são seus conteúdos proposicionais. Mensagens são codificadas em uma linguagem idiomática (*Idiomatic Language*). Nesta pesquisa assumimos a linguagem de comunicação como uma propriedade de uma mensagem,

e definimos três constantes (“VIDEO”, “AUDIO” e “TEXTO”) para compor o domínio de valores possíveis desta propriedade, conforme definição formalizada nos axiomas A10, A11 e A12.

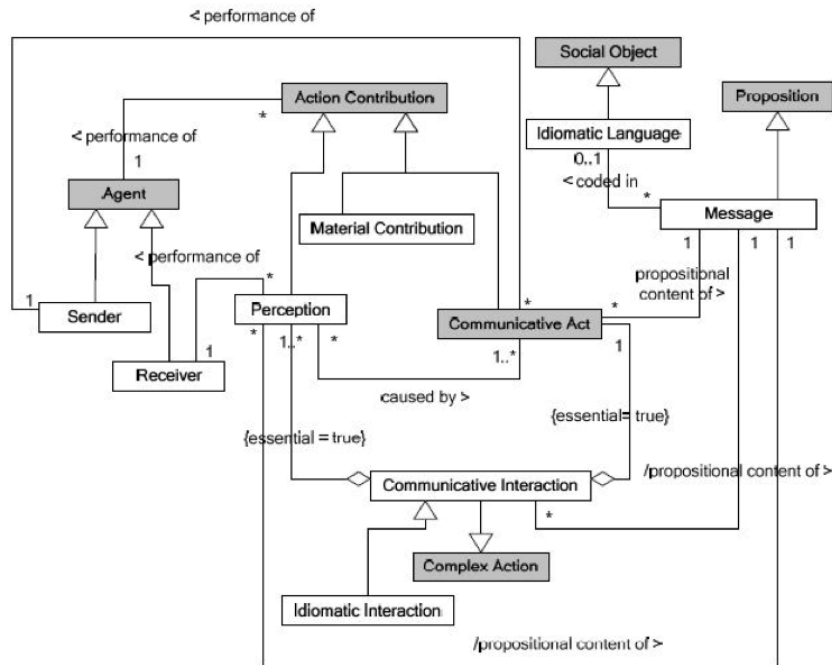


Figura 20: Ato comunicativo
 Fonte: FRANÇA (2012).

Axioma A10: Uma mensagem está codificada em texto, quando o conteúdo de sua linguagem idiomática assim informar.

Formalmente:

$$\text{communicationLanguage}(m, \text{“TEXTO”}) \leftarrow \text{Communicative Act}(ca) \wedge \text{Message}(m) \wedge \text{IdiomaticLanguage}(il) \wedge \text{propositional_content_of}(m, ca) \wedge \text{coded_in}(m, il) \wedge \text{content_of}(il, \text{“TEXTO”})$$

Axioma A11: Uma mensagem está codificada em áudio, quando o conteúdo de sua linguagem idiomática assim informar.

Formalmente:

$$\text{communicationLanguage}(m, \text{“AUDIO”}) \leftarrow \text{Communicative Act}(ca) \wedge \text{Message}(m) \wedge \text{IdiomaticLanguage}(il) \wedge \text{propositional_content_of}(m, ca) \wedge \text{coded_in}(m, il) \wedge \text{content_of}(il, \text{“AUDIO”})$$

Axioma A12: Uma mensagem está codificada em vídeo, quando o conteúdo de sua linguagem idiomática assim informar.

Formalmente:

$$\text{communicationLanguage}(m, \text{"VIDEO"}) \leftarrow \text{Communicative Act}(ca) \wedge \text{Message}(m) \wedge \text{IdiomaticLanguage}(il) \wedge \text{propositional_content_of}(m, ca) \wedge \text{coded_in}(m, il) \wedge \text{content_of}(il, \text{"VIDEO"})$$

Estruturação do discurso:

A taxonomia de CALVÃO *et al.* (2012) define que *groupware* pode ser classificado relativo à sua estrutura de discurso como funções contínuas, lista, árvore, estrela e grafo. A estrutura do discurso está relacionada à estrutura do sequenciamento das mensagens.

Nesta pesquisa foi definido um atributo para informar a estrutura do discurso (*messageStructure*) em uma Interação Comunicativa (*Communicative Interacion*), cujo domínio de valores possíveis é (LISTA, ARVORE, ESTRELA e GRAFO), atribuídos conforme os axiomas A13, A14 e A15.

Axioma A12: A estrutura do discurso de uma interação comunicativa está no formato de lista quando o conteúdo de seu atributo *messageStructure* assim informar.

Formalmente:

$$\text{communicativeInteraction}(ci, \text{"LISTA"}) \leftarrow \text{messageStructure}(ci, \text{"LISTA"})$$

Axioma A13: A estrutura do discurso de uma interação comunicativa está no formato de árvore quando o conteúdo de seu atributo *messageStructure* assim informar.

Formalmente:

$$\text{communicativeInteraction}(ci, \text{"ARVORE"}) \leftarrow \text{messageStructure}(ci, \text{"ARVORE"})$$

Axioma A14: A estrutura do discurso de uma interação comunicativa está no formato de estrela quando o conteúdo de seu atributo *messageStructure* assim informar.

Formalmente:

$$\text{communicativeInteraction}(ci, \text{"ESTRELA"}) \leftarrow \text{messageStructure}(ci, \text{"ESTRELA"})$$

Axioma A15: A estrutura do discurso de uma interação comunicativa está no formato de grafo quando o conteúdo de seu atributo *messageStructure* assim informar.

Formalmente:

```
communicativeInteraction (ci, "GRAFO") ← messageStructure(ci, "GRAFO")
```

Tamanho da mensagem: Esta propriedade poderia ser inferida automaticamente se o processo já tivesse sido executado, calculando-se o tamanho das mensagens trocadas. Mas como a recomendação de serviços na XCuteKIP é feita em tempo de projeto, foi necessário definir um atributo para a classe *Message(messageSize)*.

Os valores possíveis para atribuição neste campo são definidos nas variáveis contidas nos axiomas A16 e A17. Seus valores possíveis são CURTA e ELABORADA.

Axioma A16: Uma mensagem será curta quando o conteúdo de seu atributo *messageSize* assim informar.

Formalmente:

```
message(m, "CURTA") <- messageSize(m, "CURTA")
```

Axioma A17: Uma será longa quando o conteúdo de seu atributo *messageSize* assim informar.

Formalmente:

```
message (m, "ELABORADA") <- messageSize(m, "ELABORADA")
```

4.3. Mapeamento KIPO X WGWSOA

Conforme já discutido, as atividades colaborativas em um PIC instanciam classes da KIPO. Para encontrar serviços colaborativos adequados a suportar estas atividades, tais serviços precisam ser relacionados às características particulares destas atividades, tais como se uma atividade possui troca informal de conhecimento, se o tipo de comunicação é síncrono ou assíncrono, se uma tarefa envolve decisão. Para isso, esta seção descreve um mapeamento entre as atividades e os serviços colaborativos.

A WGWSOA foi definida nesta pesquisa como provedor de serviços colaborativos. Portanto, o mapeamento entre as atividades colaborativas e os serviços colaborativos levará em conta os serviços providos pela WGWSOA.

A maneira pela qual as classes da WGWSOA está estruturada permite um relacionamento direto para diversos conceitos da KIPO, ou seja, um mapeamento da KIPO

com a WGWSOA. Por exemplo, na WGWSOA, o serviço de troca instantânea de mensagens (*ExchangeMessage*) é utilizado para enviar, receber e listar mensagens entre usuários. Este mesmo conceito na KIPO está relacionado à interação comunicativa (*CommunicativeInteraction*). A KIPO separa os conceitos de percepção (*Perception*), que é o ato de receber uma mensagem (*Message*) e uma comunicação (*Communication*) que é o ato de enviar uma mensagem.

A classe de categorização (*Categorization*) na KIPO é uma dependência para os recursos (*Resource*) e tipo de momento (*MomentUniversal*). A categorização pode ser relacionada ao serviço de classificação da WGWSOA. Este serviço da arquitetura está relacionado ao conceito da metodologia IBIS (GEROSA, 2002), que classifica as mensagens como Questão, Posição e Argumentação.

Os usuários online dos sistemas colaborativos (*UserList*) da WGWSOA são mapeados à classe *CollaborativeGroup*. Este último tem como característica ser regido por um protocolo, que pode ser o protocolo social para se articularem (coordenação) e trabalharem juntos (cooperação). Os agentes envolvidos no grupo colaborativo assumem no contexto de comunicação os papéis de emissor (*Sender*) e receptor (*Receiver*) de mensagens durante o ato comunicativo (*CommunicativeAct*), todos estes conceitos são oriundos da KIPO. A classe corresponde ao agente na WGWSOA é *User*.

Para ilustrar um exemplo deste mapeamento para um serviço de cooperação, foi desenvolvido um Fórum de discussões no projeto WGWSOA (Figura 21), onde as classes com fundo cinza compõem a funcionalidade do Fórum na WGWSOA, as com fundo amarelo fazem parte da ontologia de comunicação, as com fundo vermelho fazem parte da ontologia de cooperação e o mapeamento entre classes é mostrado como linhas que relacionam as classes com o fundo cinza com as de fundo vermelho e as de fundo cinza com as de fundo amarelo. O local onde ocorrem as sessões colaborativas (*CollaborativeSession* da KIPO) corresponde ao Fórum na WGWSOA. Os participantes do Fórum são agentes (*Agent*) que contribuem com diversas mensagens (*Message*), conceito do domínio de comunicação. Para a KIPO, o espaço compartilhado para troca de informações é definido como *VirtualSite*. O serviço de atividade (*Activity*) da WGWSOA pode ser percebido como uma ação de contribuição (*ActionContribution*) da KIPO. Cada atividade pode ser percebida por outros colaboradores do grupo de trabalho. Na KIPO, ações que criam recursos podem ter tipos de dependências conhecidas como dependências de fluxo, que basicamente estão relacionadas à criação, localização, usabilidade. Além dessas, ainda é importante mencionar a dependência

de *fit*, na qual um recurso é produzido de forma coletiva através da execução de mais de uma ação ao mesmo tempo. O reconhecimento desses diversos tipos de dependência pela coordenação é essencial para uma análise das possíveis estratégias de coordenação.

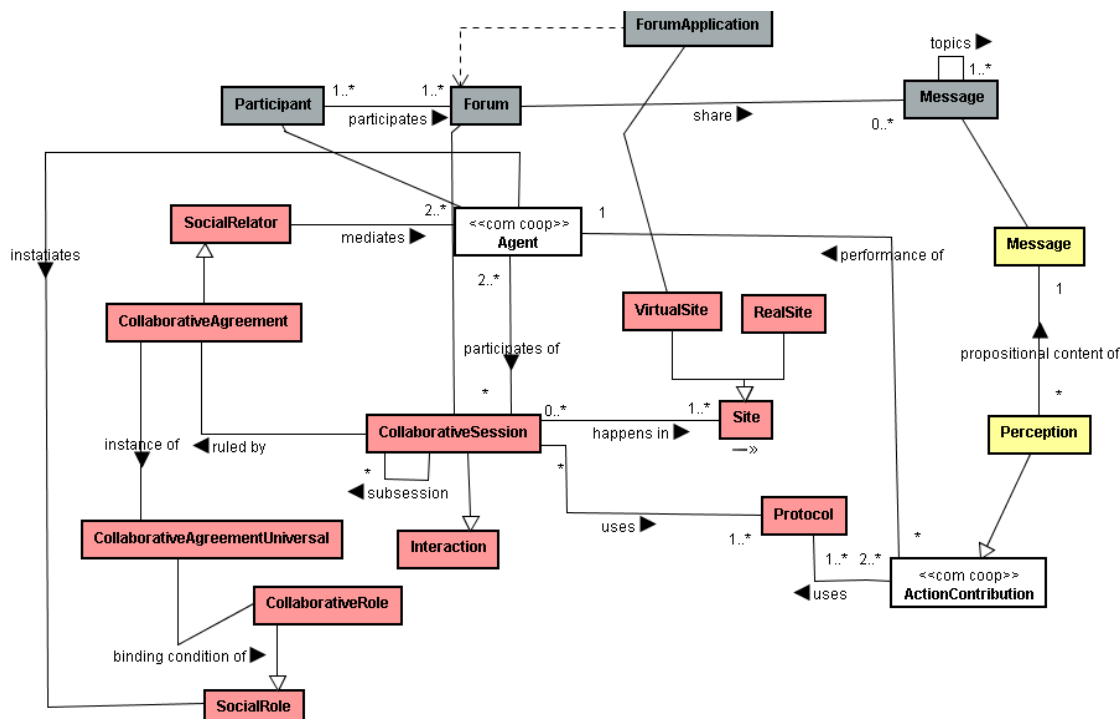


Figura 21: Mapeamento Fórum WGWSOA com Classes KIPO

O serviço de percepção, ilustrado na figura 22, faz parte do serviço AW2SOA da WGWSOA e seu correspondente na KIPO é a classe *Perception*. Dentre os serviços oferecidos pela AW2SOA, podem ser citados os serviços de: notificação, pesquisa, perfil, gerência de banco de dados e um sensor transversal (um serviço que faz monitoração das atividades e eventos ocorridos na arquitetura) para aplicar o conceito de percepção utilizando o paradigma de desenvolvimento orientado a aspectos. O serviço de percepção fornece o contexto necessário para que os participantes das atividades saibam o que o outro está fazendo e qual a situação deste trabalho, uma importante função em sistemas colaborativos (BORGES *et al.* 1995).

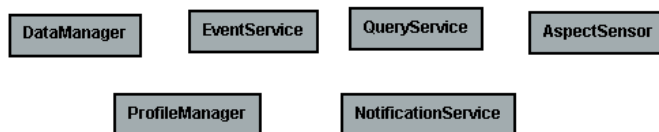


Figura 22: Serviços de percepção WGWSOA
Fonte: DAVID e MACIEL (2010).

Nesta camada é fornecido o conhecimento do que está acontecendo no momento da execução das atividades, informações tais como o nome do evento e do ator responsável, uma lista de nomes que identificam os parâmetros do evento e outra lista com os seus valores são obtidos através destes serviços (DAVID e MACIEL, 2010). A preocupação com a manutenção em banco de dados, manipulação de XML e gerenciamento de perfis são delegados ao *DataManager* e *ProfileManager*. Os dois últimos apoiam os demais serviços de percepção dentro da AW2SOA.

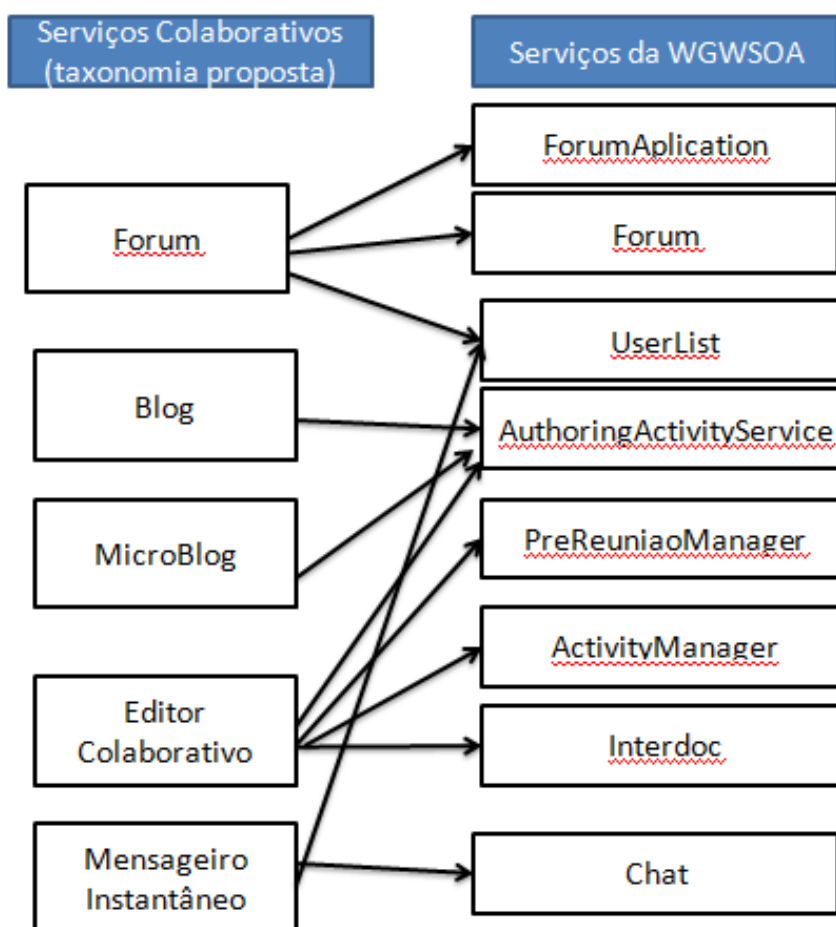


Figura 23: Mapeamento entre a taxonomia de serviços colaborativos e os serviços da WGWSOA

Embora seja possível um mapeamento das classes da KIPO para as classes da WGWSOA, nem todas as classes da KIPO tem um correspondente na arquitetura de serviços colaborativos, como é possível observar na figura 23. Por esta razão esta pesquisa propõe a utilização de uma taxonomia que categoriza os serviços colaborativos (MOURA *et al.*,2013).

Na figura 23 é possível observar que diversos serviços da WGWSOA são mapeados para um ou mais de um serviço colaborativo. Como é o caso do serviço de Mensageiro

Instantâneo que é composto por dois serviços da WGWSOA, *Chat* e *UserList*, que tem seu uso compartilhado pelo serviço de *Forum*.

Desta forma, todos os serviços da WGWSOA foram mapeados conforme a taxonomia empregada nesta dissertação, viabilizando o uso da arquitetura WGWSOA para prover os serviços colaborativos necessários para apoiar as atividades colaborativas de PIC.

4.4. Método para Recomendação de Serviços colaborativos

O método proposto consiste em duas fases: **Recomendação de Serviços Colaborativos** (figura 30) e **Execução de Atividades Colaborativas do Processo** (figura 31). O resultado da fase de recomendação de serviços colaborativos é um conjunto de serviços colaborativos adequados para apoiar a execução das atividades do PIC. Esta fase executa algoritmos, baseados nos axiomas apresentados na seção 4.1.1, para identificar cada tipo de atividade em que colaboração é necessária e identificar os serviços colaborativos que atendem às necessidades de colaboração nessas atividades. Na fase de execução do processo, as atividades colaborativas são instanciadas com apoio dos serviços recomendados na fase anterior. Neste trabalho, o papel atribuído ao participante do processo que define as atividades do PIC é chamado de Configurador e o papel atribuído a participante que executa as atividades do PIC é chamado de Executor.

4.4.1. Recomendação de Serviços Colaborativos

A primeira fase do método (Figura 2) é constituída por três passos: i - Identificar Tipo de Colaboração em Atividades; ii - Pesquisar Serviços Colaborativos e; iii - Recomendar os Serviços. Os tipos de atividades previstos neste trabalho são os que foram definidos na taxonomia de CALVÃO *et al.* (2012), a saber: Correio Eletrônico, SMS, Correio de Voz, Blog, Microblog, Lista de Discussão, Forum de Discussão, Mapa de Discussão, Mensageiro Instantâneo, Vídeo Conferência, Bate Papo, Audio-Conferência e Vídeo-Conferência. O método será aplicado para, a partir dos axiomas criados, relacionar as atividades colaborativas do PIC aos serviços da WGWSOA que foram classificados usando os filtros, apresentados na seção 4.1.

4.4.1.1 Identificar Tipo de Colaboração em Atividades

O primeiro passo, Identificar o Tipo de Colaboração, analisa as classes da KIPO que foram instanciadas pela atividade do processo e aplica os dois grupos de algoritmos baseados nos axiomas. O primeiro grupo de axiomas, A1, A2 e A3, distingue se a atividade é de fato colaborativa ou não. O segundo grupo, axiomas A4 a A17, filtram as atividades colaborativas aplicando a taxonomia de CALVÃO *et al.* (2012) e recomendam o serviço colaborativo mais adequado à atividade.

```
82= <select id="isSocialization" parameterType="int" resultType="boolean">
83   select case when count(*) > 0 then 1 else 0 end
84   from socialization soc
85   inner join activity ac on ac.id = soc.communicativeinteraction_id
86   where ac.id = #{id};
87 </select>
```

Figura 24: Algoritmo Baseado no Axioma A1

Algumas atividades de um PIC realizam troca informal de conhecimento, que ocorre durante uma socialização entre seus participantes. No ato de instanciar um PIC utilizando a KIPO, as classes *Informal Exchange* e *Socialization* serão instanciadas para registrar o conhecimento das atividades. Toda Socialização também é uma Interação Comunicativa (*Communicative Interaction*). O algoritmo ilustrado na figura categoriza instâncias de atividades Socialização como atividades colaborativas.

```
89= <select id="isCollaborativeSession" parameterType="int" resultType="boolean">
90   select case when count(*) > 0 then 1 else 0 end
91   from collaborativesession cs
92   inner join COLLABORATIVESESSION_HAS_PROTOCOL CHP on cs.id =
93   chp.collaborativesession_id
94   inner join protocol p on p.id = chp.protocol_id
95   inner join ACTIONCONTRIBUTION_HAS_PROTOCOL AHP on ahp.protocol_id = p.id
96   inner join ACTIONCONTRIBUTION AC on ac.id = ahp.actioncontribution_id
97   where ac.id = #{id};
98 </select>
--
```

Figura 25: Algoritmo Baseado no Axioma A2

Em um PIC podem também ocorrer atividades colaborativas onde o trabalho em conjunto acontece sem necessariamente haver alguma comunicação. Para estes casos em particular a classe *Collaborative Session* será instanciada. O algoritmo baseado no Axioma A2 é apresentado na Figura 25.

```

55= <select id="isCollaborativeActivity" parameterType="int" resultType="boolean" >
56     SELECT CASE WHEN count(*) > 0 THEN 1 ELSE 0 END
57     from alternative_has_speciality alsp
58     inner join alternative al on al.id = alsp.alternative_id
59     inner join speciality sp on sp.id = alsp.speciality_id
60     inner join inovationagent ia on ia.id = sp.inovationagent_id
61     inner join agent ag on ag.id = ia.id
62     inner join (
63         select kia.agent_id ag, al.id al_id
64         from knowledgeintensiveactivity kia
65         inner join decision dc on dc.kia_activity_id = kia.activity_id
66         inner join alternative al on al.id = dc.id
67         where kia.activity_id = #{id}
68         ) kia on kia.al_id = al.id
69     where kia.ag <![CDATA[ <> ]]> ia.id;
70 </select>

```

Figura 26: Algoritmo Baseado no Axioma A3

Igualmente, quando decisões precisarem ser tomadas por um agente de inovação, algumas alternativas podem estar envolvidas. Serão instanciadas as classes *Decision*, *Alternative*, *Speciality* e *Innovation Agent* nestes casos. A figura 26 apresenta o algoritmo que baseado no axioma A3 identifica as atividades que participam de decisão e que são colaborativas. Cada agente de inovação (*Innovation Agent*) possui uma ou mais especialidades (*Speciality*) e a complexidade dos PIC pode demandar diversas especialidades, significando diversos agentes.

O atributo sincronizado com (*synchronized with*) presente na classe *Event*, que é especializada por uma atividade do PIC, define a sincronia de duas atividades, segundo o axioma 4.

Uma vez que uma atividade do PIC tenha sido classificada como colaborativa, os demais axiomas são aplicados. A ferramenta realiza um filtro baseado nos axiomas A4 ao A17. Para cada atividade identificada como colaborativa, aplica-se o algoritmo derivado.

A figura 27 apresenta o algoritmo (derivado do axioma A4) que retorna se uma atividade é síncrona ou não.

```

79= <select id="getSinchronization" parameterType="int" resultMap="socializationInfoResult">
80     select synchronous
81     from socialization sc
82     inner join activity ac on ac.id = sc.communicativeinteraction_id
83     inner join agent ag on ag.id = agentid
84     where sc.collaborativesession_id = #{id};
85 </select>

```

Figura 27: Algoritmo Baseado no Axioma A4

Após a definição da sincronia (isto é, se uma atividade é síncrona ou assíncrona), é realizada para todas as atividades a inferência de quantos interlocutores participam da interação comunicativa aplicando o algoritmo visto na figura 27, e derivado dos axiomas A5 ao A9.

```

129= <select id="getInterlocutorRelation" parameterType="int" resultMap="comunicationResult" >
130     select ci.complexaction_id comunicativeinteraction_id,
131           ac.name comunicativeinteraction_name,
132           kia.agent_id kia_agent,
133           ag.id agent_comunication_id,
134           ag.name agent_comunication_name,
135           il.languagetype language_type,
136           case
137             when (sa.agent_id is not null)
138             then 'Muitos'
139             else 'Um'
140           end cadinality
141     from comunicativeinteraction ci
142     inner join activity ac on ac.id = ci.complexaction_id
143     left outer join knowledgeintensiveactivity kia on kia.activity_id = ac.id
144     inner join communication cm on ci.complexaction_id = cm.comunicativeinteraction_id
145     inner join message m on m.id = cm.message_id
146     left outer join idiomaticlanguage il on il.id = m.idiomaticlanguage_id
147     inner join agent ag on ag.id = cm.sender_agent_id
148     left outer join socialagent sa on ag.id = sa.agent_id
149     where ci.complexaction_id = #{id};
150 </select>

```

Figura 28: Algoritmo Baseado nos Axiomas A5 ao A9

Os algoritmos derivados dos axiomas A10 ao A12 referem-se à linguagem utilizada na interação comunicativa e podem ser observados na figura 29. As algoritmos derivados dos axiomas A13 ao A15 recuperam as informações diretamente da interação comunicativa e os algoritmos derivados dos axiomas A16 e A17 recuperam informações da propriedade *messageSize* da tabela de Mensagem.

```

153= <select id="getServiceByCategory" parameterType="HashMap"
154           resultMap="recomendationResult" >
155     select s.id service_id, c.id cat_id, c.description category,
156           s.servicename service_name
157     from wgwsa_category c
158     inner join wgwsa_service_has_category sc on c.id = sc.category_id
159     inner join wgwsa_service s on s.id = sc.service_id
160     left outer join wgwsa_language l on l.id = c.wgwsa_language_id
161     where c.description = #{category}
162= <if test="language != null and language != ''">
163         and l.description = #{language};
164 </if>

```

Figura 29: Algoritmo Baseado nos Axiomas A10 ao A12

4.4.1.2. Pesquisar Serviços Colaborativos

Pesquisar Serviços Colaborativos (passo dois do método) é executado somente após a etapa anterior ter descoberto as atividades colaborativas do processo. O objetivo deste passo, Pesquisar Serviços Colaborativos, é encontrar no repositório de serviços colaborativos os que correspondam a estes requisitos colaborativos das atividades que foram listados.

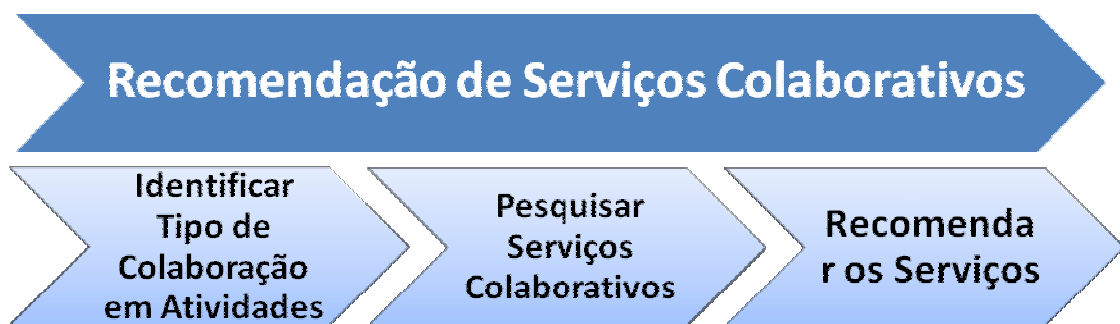


Figura 30: Fase de Recomendação de Serviços Colaborativos

Como premissa, um analista deve avaliar cada novo *groupware* desenvolvido e classificá-lo utilizando a taxonomia de CALVÃO *et al.* (2012). Como exemplo das informações recuperadas na fase anterior tem-se, por exemplo, que uma atividade é colaborativa, síncrona, que tem muitos interlocutores e que sua linguagem de comunicação definida é áudio. No passo de “Pesquisar Serviços Colaborativos”, todos os serviços no repositório de *groupware* que atendam parcial ou totalmente as características colaborativas serão identificados.

O artefato resultante do segundo passo é uma lista de serviços encontrados para cada atividades colaborativas. Esta lista de recomendações é construída aplicando-se o mapeamento de conceitos da KIPO com os serviços colaborativos do repositório. Para ilustrar o cenário, cada vez que um agente faz uma Ação Contribuição usando de sua criatividade e inovação, um serviço de Percepção deve ser recomendado para que os demais participantes estejam cientes da contribuição.

Na ocorrência de uma Decisão (um subtipo de Evento de Contingência), que é realizada por diversos agentes para resolver uma questão, um serviço virtual (como por

exemplo um Fórum) deve ser recomendado para que os agentes possam discutir alternativas (*Alternative*).

Finalmente, no caso da atividade colaborativa (como uma Socialização), uma vez que um grupo de trabalho ocorre em um local real ou virtual, e se um recurso precisar ser compartilhado para edição, um serviço de Edição Compartilhada será recomendado para tornar o recurso de conhecimento disponível para a colaboração.

Para o caso deste último exemplo, se o recurso de conhecimento é um arquivo de texto, um Serviço de Edição de Documento Colaborativo será recomendado. Caso o recurso de conhecimento seja um diagrama, um Serviço de Editor Visual Colaborativo será recomendado, e assim por diante.

No caso em que os algoritmos não encontrarem nenhum serviço adequado aos requisitos colaborativos para alguma atividade, o algoritmo tentará propor a melhor alternativa possível. Isto é, entende-se como o serviço mais adequado aquele que for aplicado à maior quantidade de filtros da taxonomia de CALVÃO *et al.* (2012) e que retornar algum resultado. Os algoritmos serão executados eliminando um critério de filtro, ou seja, ampliando a abrangência do retorno de resultado a cada execução sem resposta de pelo menos um serviço que satisfaça a necessidade colaborativa da atividade.

4.4.1.3. Recomendar Serviços

Quando a recomendação é finalizada, o Configurador decide se ele / ela quer utilizar a totalidade ou alguns dos serviços colaborativos recomendados.

O último passo, Recomendar os Serviços, é responsável pela construção de uma lista de serviços recomendados para o Configurador. Esta atividade é interativa, em que o configurador pode escolher quais serviços da lista oferecida ele/ela quer utilizar. Esta etapa armazena, para recuperação posterior, a configuração realizada pelo Configurador.

4.4.2. Execução de Atividades Colaborativas do Processo

A segunda fase (Figura 3) apoia a execução das atividades colaborativas do PIC. Esta fase consiste em dois passos: i - Recuperar Recomendação e; ii - Executar Serviços.

4.4.2.1. Recuperar Recomendação

O passo de “Recuperar Recomendação” lista os serviços colaborativos que foram recomendados no passo anterior, e configurados pelo configurador. Esta etapa começa quando as atividades colaborativas precisam ser instanciadas, em tempo de execução do PIC. Como as atividades de um PIC podem não seguir um fluxo definido a priori, o executor pode realizar qualquer atividade a qualquer momento. Quando uma atividade colaborativa for executada, este passo do método recupera os serviços recomendados para esta atividade.

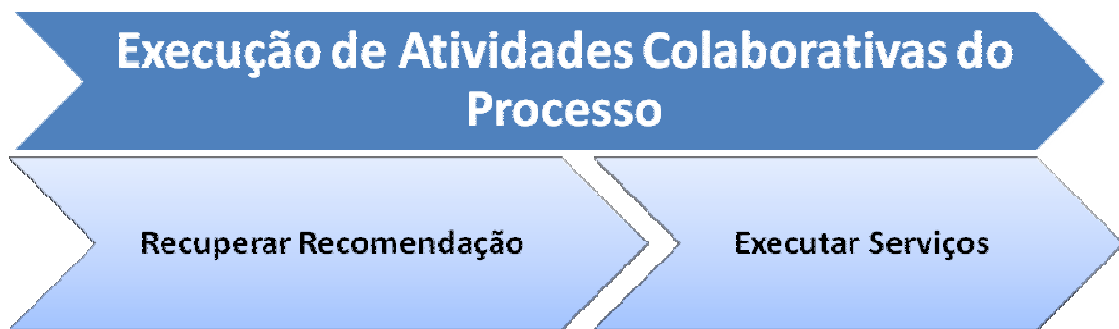


Figura 31: Fase de Execução de Atividades Colaborativas do Processo

4.4.2.2. Executar Serviços

Finalmente, os serviços serão executados para apoiar a execução das atividades colaborativas. Existem diversas abordagens na literatura para invocação de serviços de forma automática (LEITNER, ROSENBERG e DUSTDAR; 2009). A maioria destas abordagens utiliza a descrição de serviços como webservices. Embora os serviços da WGWSOA não sejam necessariamente disponibilizados como webservices, essa tecnologia pode ser utilizada para que se possa tirar proveito das vantagens oferecidas pelos seus protocolos. Por exemplo, para organizar a sequência de atividades das quais esses serviços participam, mecanismos de orquestração podem ser utilizados (DAVID e MACIEL, 2010).

Portanto, para a execução dos serviços colaborativos esta pesquisa propõe o uso de invocação automática, com as citadas em (PAPAZOULOU *et al.*, 2007).

4.5. Arquitetura XCuteKIP

A XCuteKIP foi implementada seguindo uma arquitetura em três camadas (Figura 32):

- i) KIP Client Environment: interface gráfica para interação com o usuário;

- ii) KIP Server Enviroment: camada da aplicação que se integra à arquitetura WGWSOA via serviço e que é responsável por realizar a descoberta das necessidades de colaboração nas atividades e a descoberta dos serviços adequados na camada de *groupware*; e
- iii) KIP Data: camada de armazenamento tanto dos serviços como das instâncias de PIC.

A camada **KIP Client Environment** fornece uma interface de usuário através da qual os participantes do processo podem configurar e executar atividades colaborativas do KIP. A tela de configuração é responsável pela configuração das atividades colaborativas. Ela mostra uma lista de todas as atividades colaborativas. Para cada atividade, exibe ao Configurador uma lista dos serviços existentes. A camada cliente é ainda responsável por executar os serviços de colaboração escolhidos quando o executor iniciar uma instância de uma atividade colaborativa.

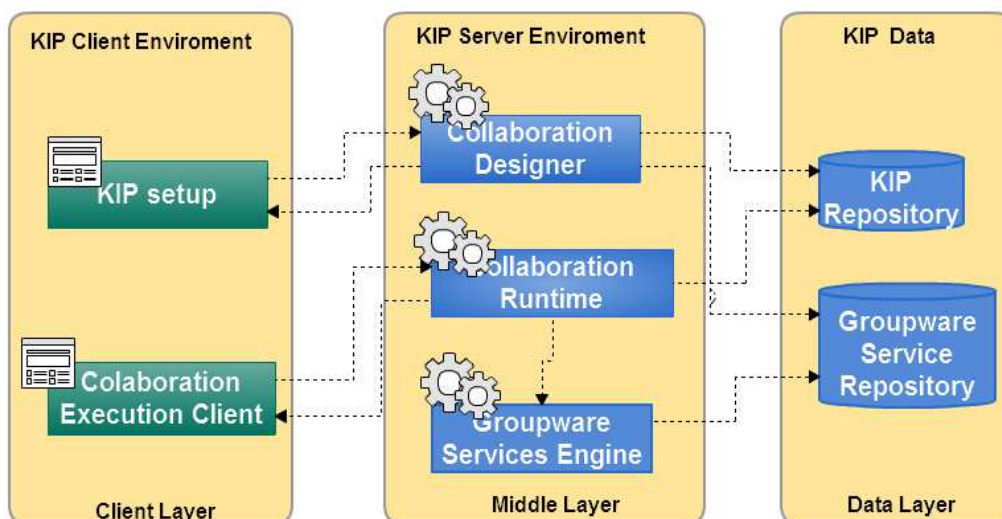


Figura 32: Arquitetura Conceitual da XCuteKIP

A camada **KIP Server Environment** é composta por três módulos. O módulo *Designer Collaboration* recupera o processo a partir do repositório KIP e identifica qual o tipo de colaboração é necessário para cada atividade do processo. Então recupera os serviços candidatos que atendam às necessidades de cada atividade colaborativa, a partir do repositório de serviços de *groupware*. O módulo *Collaboration Runtime* é invocado pelo *Collaboration Execution Client*, e é responsável por instanciar os serviços colaborativos, e enviar os resultados de atendimento ao *Collaboration Execution Client*, de modo que eles podem ser utilizados pelo participante da atividade. O módulo *Groupware Services Engine* representa o conjunto de serviços colaborativos que pode ser invocado na execução. Este módulo é

representado pela WGWSOA. O conjunto de serviços colaborativos da WGWSOA é invocado em tempo de execução.

A camada **KIP Data** compreende as informações exigidas pela XCuteKIP, e é composta por dois componentes. O *KIP Repository* armazena as instâncias dos processos mapeados com a KIPO. O *Groupware Service Repository* armazena todos os serviços colaborativos, bem como os dados manipulados pelos serviços.

4.5.1. Protótipo do XCuteKIP

Foi desenvolvido um sistema web para implementar a arquitetura XCuteKIP proposta nesta dissertação. A arquitetura interna do sistema foi implementada seguindo as boas práticas de separação de camadas do desenvolvimento de software, de acordo com o padrão de desenvolvimento MVC (*Model-View-Controller*) (REENSKAUG, 1978).

O framework SpringMVC foi utilizado para apoiar a construção do sistema na camada de controle e o MyBatis como framework objeto-relacional para interagir com o SGBD.

A integração com o middleware WGWSOA foi realizado através de serviços usando a tecnologia RMI (Remote Method Invocation). A interface de comunicação da WGWSOA forneceu as operações para consultar os serviços existentes e interagir com eles.

A interface web, como pode ser observada na figura 33, mostra os dois módulos (Collaboration Designer e Collaboration Execution) como link na parte superior da tela. Na lateral esquerda da página do Collaboration Designer ficam os processos intensivos em conhecimento a serem configurados. Já na parte central são listadas todas as atividades do PIC. Quando o usuário do sistema clica no PIC que deseja configurar, o sistema identifica as atividades e os serviços colaborativos para cada atividade. Ao lado de cada atividade colaborativa há um ícone que identifica se aquela atividade é colaborativa.

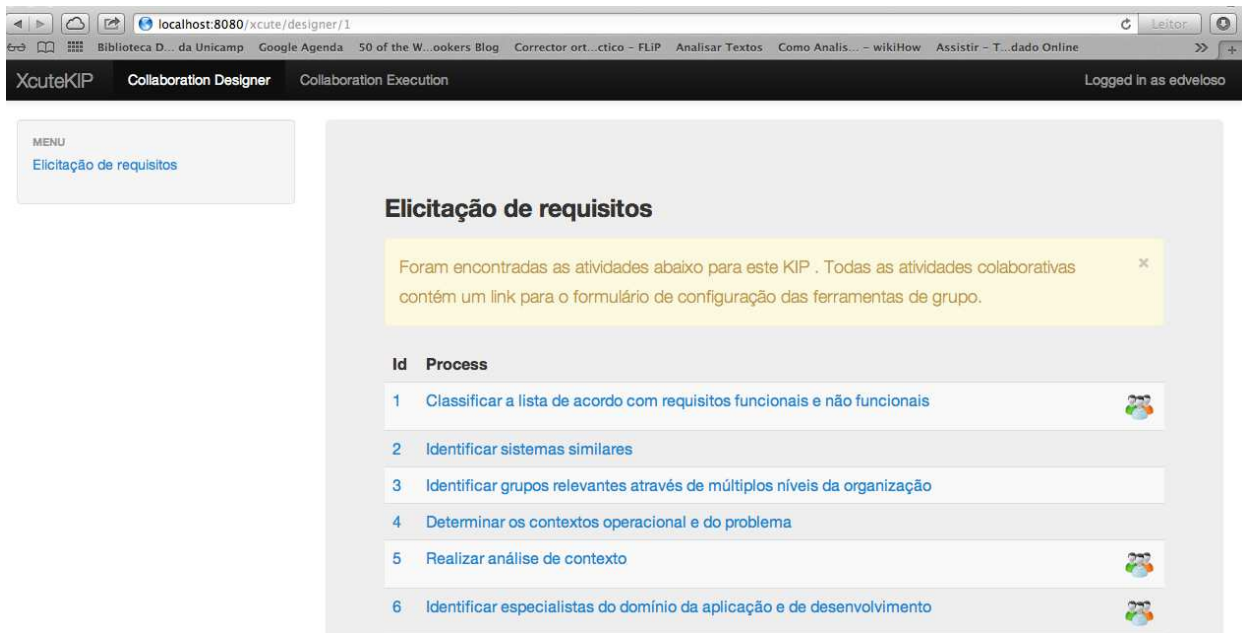


Figura 33: Tela Lista de Atividades do módulo *Collaboration Designer*

Ao clicar no ícone da atividade colaborativa, o sistema apresenta uma tela (figura 34) com as informações de contexto da atividade e o serviço recomendado pelo sistema.

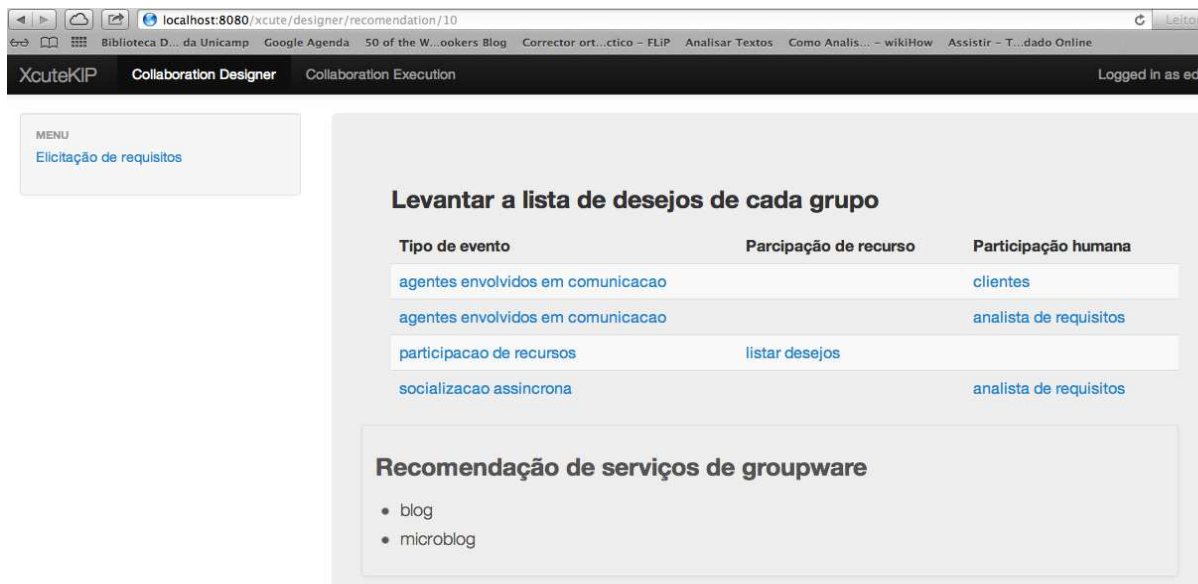


Figura 34: Tela Recomendação do módulo *Collaboration Designer*

Os serviços colaborativos recomendados pela ferramenta são filtrados usando a tela observada na figura 35, que tem como finalidade utilizar os filtros da taxonomia de CALVÃO

et al. (2012), facilitando a classificação destes serviços por parte do desenvolvedor de novos serviços colaborativos.

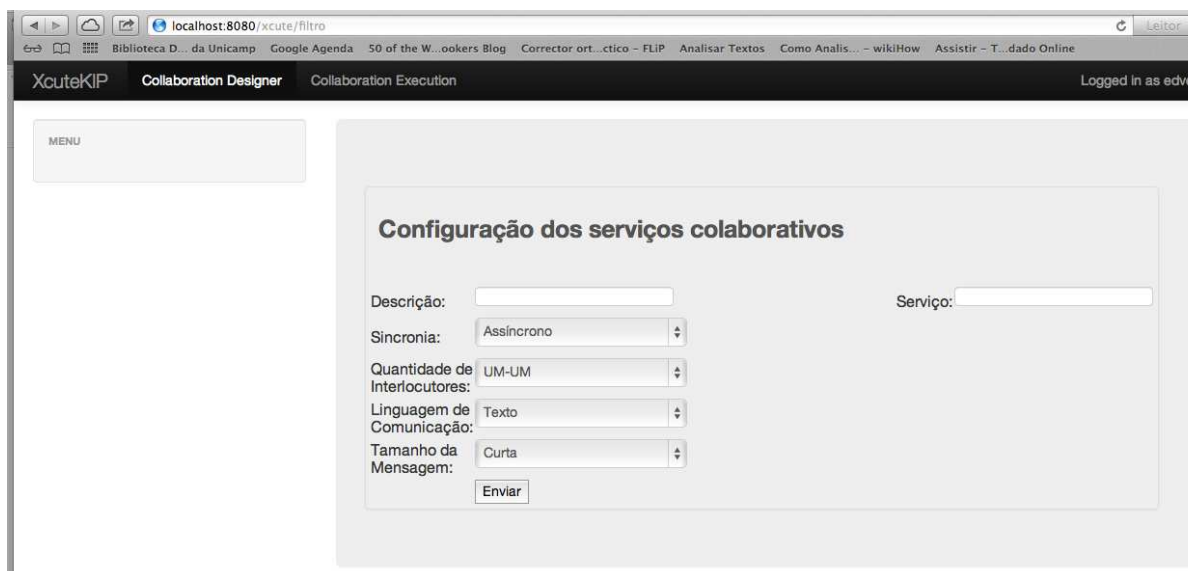


Figura 35: Tela Configuração de Serviço Colaborativo para a WGWSOA

4.5.1. Tradução da KIPO para Esquema Relacional

Foi realizada nesta pesquisa uma tradução das classes da ontologia KIPO para um esquema relacional de um SGBD. Esta abordagem foi adotada devido às dificuldades impostas sobre a abordagem de manipulação de arquivos (OWL e RDF) sobre grandes ontologias e diversas instâncias. Neste cenário se enquadram a KIPO e as ocorrências dos processos intensivos em conhecimento instanciadas. A abordagem de armazenamento de ontologias em SGBDs tem a vantagens de usufruírem de todos os recursos desenvolvidos e amplamente testados por essa tecnologia, tais como processamento, segurança e otimização de consultas. O anexo III apresenta esta tradução.

Para cada classe da KIPO, foi implementada uma tabela com seus atributos necessários e relacionamento.

Capítulo 5 – Estudo de Caso e Análise de Resultados

O objetivo deste capítulo é apresentar um estudo de caso, onde foi aplicado o método proposto em uma organização real e são apresentados e discutidos os resultados desta aplicação.

5.1. Cenário do Estudo de Caso

Um cenário onde um processo de conhecimento intensivo tem uma forte necessidade de colaboração entre os agentes é o processo de levantamento de requisitos de software, como descrito na Figura 36 (SOMMERVILLE, 2007).

Os analistas e engenheiros de software trabalham com diversos colaboradores envolvidos no processo, dentre eles os clientes e utilizadores finais para obter sucesso no projeto. O intuito é encontrar um conjunto de funções para o sistema, levantar as características de desempenho, restrições de hardware e outras informações relativas a este projeto. As técnicas e métodos para apoiar tais atividades de levantamento de requisitos são diversos.

Contudo, existem alguns desafios na condução deste processo, dentre eles a falta de confiança sobre os objetivos do sistema e o conhecimento implícito e tácito de pessoas envolvidas. Além disso, fatores externos (política e econômica) interferem diretamente sobre o processo e sobre o seu resultado (SOMMERVILLE, 2007), aumentando sua complexidade. A Figura 36 ilustra os principais passos deste processo, de acordo com Sommerville (2007). Este cenário ilustra bem os desafios de execução do processo de levantamento de requisitos, caracterizando-o como um processo intensivo em conhecimento.

Com o objetivo de avaliar o método proposto, o estudo de caso foi conduzido sobre este processo.

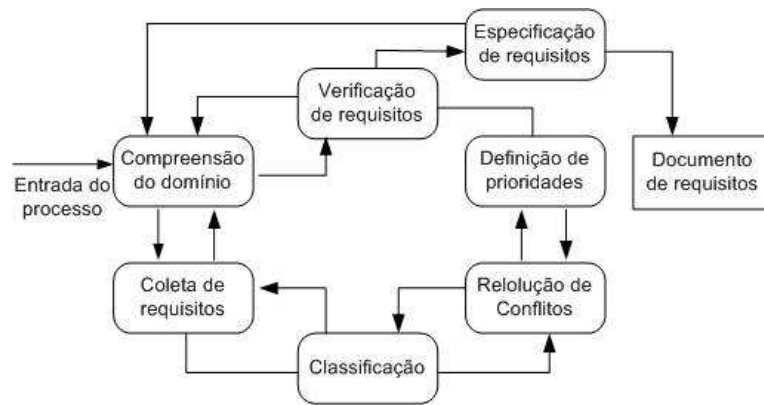


Figura 36: Processo de Levantamento de Requisitos
 Fonte: SOMMERVILLE (2007).

O estudo de caso foi realizado em uma empresa de ramo de exploração de petróleo. As atividades de desenvolvimento de software dentro da empresa são geridas por processos e metodologias de gestão e desenvolvimento de software. O grupo que participou do estudo de caso utiliza abordagens ágeis para o desenvolvimento de software. Basicamente as demandas por aplicações oriundas dos clientes surgem por um de seus representantes, no caso específico desta empresa, chamado de Agilidade. São realizadas diversas reuniões para definir o escopo macro da aplicação, riscos e a viabilidade técnica da solução. Frequentemente, estas atividades reúnem um grande grupo de colaboradores para decidir a melhor solução técnica. São aplicadas diversas técnicas da engenharia de requisitos para listar os requisitos para a aplicação.

As atividades de um processo genérico de levantamento de requisitos são mostradas dentro da tabela 1 e foram apresentadas no relatório técnico de CHRISTEL e KANG (2012).

Tabela 1: Tarefas de Processo de Levantamento de Requisitos.
 Fonte: CHRISTEL e KANG (2012)

| Tarefas da Elicitação de Requisitos | | |
|--|--|--|
| Atividade | Tarefas orientadas ao usuário | Tarefas orientadas ao desenvolvedor |
| Busca de Fatos | Identificar grupos relevantes através de múltiplos níveis da organização. Determinar os contextos operacional e do problema, incluindo a definição dos modos operacionais, objetivos e cenários de missão como apropriados. Identificar sistemas similares. Realizar análise de contexto. | Identificar especialistas do domínio da aplicação e de desenvolvimento. Identificar modelo de domínio e modelo de arquitetura. Conduzir pesquisas tecnológicas, para mais tarde fazer estudo de viabilidade e análise de risco. Identificar custos e restrições à implementação impostas pelo patrocinador. |
| Coleta e Classificação dos Requisitos | Levantar a lista de desejos de cada grupo. | Classificar a lista de acordo com funcionais, não funcionais, restrições de ambiente, restrições de projeto e ainda de acordo com as partições definidas pelo modelo de domínio e pelo paradigma de desenvolvimento. |
| Racionalização e Avaliação | Responder questões da forma "Por que você precisa de X?", a partir de raciocínio abstrato. Isso auxilia a transformar o raciocínio das questões sobre "como?" para as questões sobre "o quê?". | Realizar uma análise de riscos, investigando técnicas, custos, prazos e incluindo análise de custos e benefícios e viabilidade baseado na disponibilidade da tecnologia. |
| Priorização | Determinar funcionalidades críticas para a missão. | Priorizar requisitos baseados em custo e dependência. Estudar como o sistema pode ser implementado de forma incremental, investigando modelos arquiteturais apropriados. |
| Integração e Validação | Resolver a maior quantidade possível de pontos em aberto. Validar que os requisitos estão concordando com os objetivos iniciais. Obter autorização e verificação para passar ao próximo passo de desenvolvimento (e.g. a demonstração e a validação). | Resolver conflitos e verificar consistência. |

Os principais objetivos do estudo de caso foram: a) Identificar se o método e a ferramenta propostos auxiliam a colaboração no desenvolvimento da atividade de levantamento de requisitos; b) Identificar se os serviços colaborativos que o método e a arquitetura propostos recomendam são adequados para realizar as atividades de levantamento de requisitos.

5.2. Descrição Geral do Estudo de Caso

Para a realização do estudo de caso, participou um grupo constituído de profissionais que atuam em diferentes áreas, sendo distribuídos da seguinte forma em relação a sua principal ocupação: seis desenvolvedores (analistas e programadores); um profissional da área de qualidade (testes, padrões, etc.); três arquitetos de solução; dois gerentes de projeto. Ao grupo foi entregue a lista de tarefas do processo de levantamento de requisitos (ver Tabela 1) e uma lista de serviços de colaboração presentes na WGWSOA, bem como a descrição das atividades e das ferramentas de colaboração para equalizar o conhecimento.

Os participantes realizaram as seguintes tarefas, respondendo um questionário apresentado a seguir:

- (1) Avaliar e responder quais atividades das listadas na tabela 1 são colaborativas;
- (2) Recomendar um ou mais serviços colaborativos para cada atividade que foi classificada como colaborativa; e
- (3) Avaliar a recomendação obtida pelo XCuteKIP.

Antes de realizar as atividades de avaliação, os participantes foram treinados para conhecerem a taxonomia proposta no estudo de caso. As telas do XcuteKIP foram apresentadas, e foi realizado um alinhamento de expectativas de resultados esperados dos participantes. A expectativa era que cada participante pudesse, usando os critérios contidos na taxonomia, listar um ou mais serviços de groupware para cada atividade que o participante julgasse colaborativa. A lista de atividades e a recomendação dos serviços de groupware para as atividades colaborativas foram coletadas no questionário do estudo de caso.

5.3. Questionários de avaliação aplicados no estudo de caso

O questionário utilizado na coleta de dados foi organizado em três blocos de questões. O objetivo da divisão foi separar as questões por assuntos, a saber: a experiência do participante do estudo de caso com os aspectos abordados, a recomendação de serviços colaborativos e a avaliação do método e arquitetura propostos.

As perguntas do primeiro bloco, apresentadas na Tabela 2, procuraram identificar a experiência dos profissionais com os dois enfoques principais do estudo de caso: a experiência anterior com a atividade de levantamento de requisitos e com *groupware*. No

primeiro bloco, para cada questão apresentada, o participante indica o seu grau de concordância através das seguintes opções: Discordo totalmente, Discordo parcialmente, Opinião Neutra, Concordo parcialmente, Concordo fortemente, e Não tenho como opinar. O objetivo deste bloco para a pesquisa foi caracterizar os participantes no cenário do estudo de caso segundo sua experiência.

Tabela 2: Bloco de perguntas para avaliar a experiência dos participantes

| # | Atividade | Experiência | | | | | |
|---|--|-----------------|-------------------|----------------------|-------------------|--------------|-----------------------|
| | | Sem experiência | Pouca experiência | Experiência razoável | Muita Experiência | Especialista | Não tenho como Opinar |
| 1 | Atividades de levantamento de requisitos para desenvolvimento de sistemas | | | | | | |
| 2 | Atividades com base no uso de groupware (interações colaborativas apoiadas por computadores) | | | | | | |

O objetivo do segundo bloco (Tabela 3) é obter dos especialistas recomendações de serviços para as mesmas atividades que a ferramenta irá recomendar. Estes são os dados essenciais para esta pesquisa. Uma avaliação quantitativa e qualitativa pode ser realizada a partir deles.

Tabela 3: Bloco de Recomendação de Serviços Colaborativos

| # | Atividade | É colaborativa? (sim/não) | Serviço colaborativo recomendado |
|---|---|---------------------------|----------------------------------|
| 1 | Identificar grupos relevantes através de múltiplos níveis da organização. | | |
| 2 | Determinar os contextos operacional e do problema, incluindo a definição dos modos operacionais, objetivos e cenários de missão como apropriados. | | |
| 3 | Identificar sistemas similares | | |
| 4 | Realizar análise de contexto | | |
| 5 | Identificar especialistas do domínio da aplicação e de desenvolvimento. | | |

| | | | |
|----|--|--|--|
| 6 | Identificar modelo de domínio e modelo de arquitetura. | | |
| 7 | Conduzir pesquisas tecnológicas, para mais tarde fazer estudo de viabilidade e análise de risco. Identificar custos e restrições à implementação impostas pelo patrocinador. | | |
| 8 | Classificar a lista de acordo com funcionais, não funcionais, restrições de ambiente, restrições de projeto e ainda de acordo com as partições definidas pelo modelo de domínio e pelo paradigma de desenvolvimento. | | |
| 9 | Levantar a lista de desejos de cada grupo. | | |
| 10 | Realizar uma análise de riscos, investigando técnicas, custos, prazos e incluindo análise de custos e benefícios e viabilidade baseado na disponibilidade da tecnologia. | | |
| 11 | Responder questões da forma “Por que você precisa de X?”, a partir de raciocínio abstrato. Isso auxilia a transformar o raciocínio das questões sobre “como?” para as questões sobre “o quê?”. | | |
| 12 | Estudar como o sistema pode ser implementado de forma incremental, investigando modelos arquiteturais apropriados. | | |
| 13 | Priorizar requisitos baseados em custo e dependência. | | |
| 14 | Determinar funcionalidades críticas para a missão. | | |
| 15 | Obter autorização e verificação para passar ao próximo passo de desenvolvimento (e.g. a demonstração e a validação). | | |
| 16 | Validar que os requisitos estão concordando com os objetivos iniciais. | | |
| 17 | Resolver a maior quantidade possível de pontos em aberto. | | |
| 18 | Resolver conflitos e verificar consistência. | | |

As perguntas discursivas e não obrigatórias, apresentadas no terceiro e último bloco, tiveram o objetivo de buscar a opinião dos usuários sobre a ferramenta e obter sugestões de melhoria. Os participantes do estudo de caso navegaram na ferramenta responderam as seguintes questões:

1. Você considera que a ferramenta XCuteKIP auxilia de fato a atividade de escolhas de *groupware* nos processos de levantamento de requisitos?
2. O que você considera que poderia ser melhorado ou incluído na ferramenta para um maior suporte das escolhas de *groupware* para processos de levantamento de requisitos?
3. De uma forma geral, você avaliaria positivamente a ferramenta e a adotaria o uso para escolha de *groupware*?

5.4. Análise dos Resultados

Todos os participantes responderam ao questionário. No primeiro bloco, observa-se que a maioria possui experiência em levantamento de requisitos, contando inclusive com 3 especialistas e 4 profissionais com larga experiência na atividade. Porém, a maioria do grupo apresenta razoável a pouca experiência em *groupware*. As figuras 37 e 38 apresentam o resultado deste bloco.

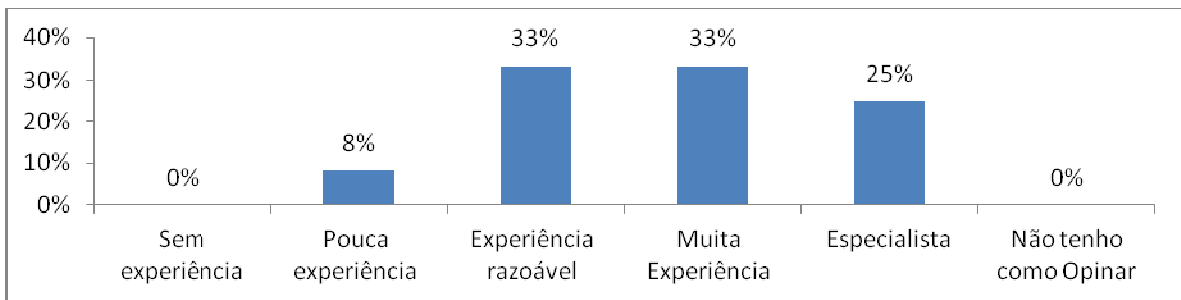


Figura 37: Experiência com atividades de levantamento de requisitos para desenvolvimento de sistemas

Quanto à recomendação que o grupo realizou sobre as dezoito atividades do processo de levantamento de requisitos, os resultados foram analisados e apresentados nas figuras 39 a 43.

Devido à diversidade das experiências vivenciadas e o nível de conhecimento das atividades e de ferramentas de *groupware*, as respostas resultantes do estudo de caso foram muito diversas. A figura 39 mostra que dois participantes (os analistas 9 e 11) consideram todas as atividades do processo como atividades colaborativas, conseqüentemente recomendaram ferramentas de *groupware* para todas as atividades. Por outro lado, os analistas 1 e 3 foram os que menos recomendaram ferramentas de *groupware* para apoiar suas atividades. Os participantes com a menor experiência com o processo e com o menor conhecimento sobre as ferramentas de *groupware* realizam as atividades de maneira mais individual.

Analisando as respostas dos analista participantes do estudo de caso, e comparando-as com as oferecidas pelo XcuteKIP, é possível notar que a recomendação automática fornecida pela ferramenta pode encontrar todas as atividades que a maioria dos analistas consideraram colaborativas. Retirando os analistas que se posicionaram nos extremos, em suas recomendações, isto é, quatro analistas, todas as atividades que a ferramenta apontou como

colaborativa, pelo menos dois outros analistas concordaram com a sugestão do XcuteKIP. O que fortalece a hipótese de que o mapeamento das atividades colaborativas de um PIC e os serviços colaborativos aumenta as probabilidade de recomendação de serviços adequados para as atividades do PIC.

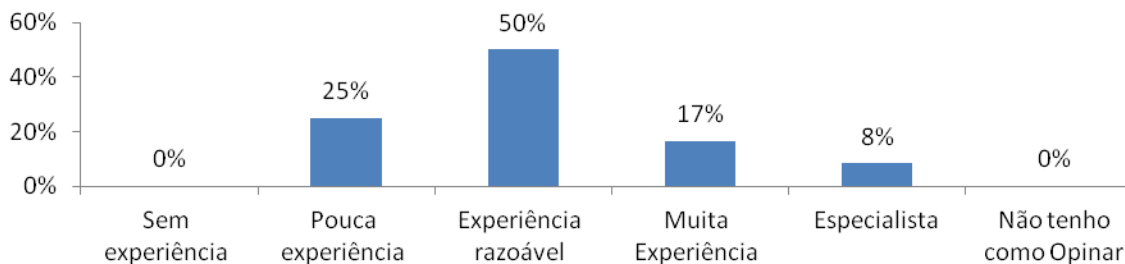


Figura 38: Experiência com atividades com base no uso de *groupware* (interações colaborativas)

O conjunto dos dados coletados neste estudo de caso mostrou uma grande dispersão nas respostas. Os analistas divergiram em relação a quantas atividades eram colaborativas e na recomendação dos serviços colaborativos para estas atividades. O que de certa forma já era esperado. Por definição um processo intensivo em conhecimento é realizado de acordo com o conhecimento e experiência do participante do processo. Diversos participantes podem executar o mesmo processo de maneira totalmente diferente.

No estudo de caso conduzido nesta pesquisa, a menor quantidade de atividades colaborativas identificadas foi 5 (ou 28%). Quatro analistas identificaram um número semelhante de atividades colaborativas em relação à ferramenta XCutKIP. Dois analistas (analistas 6 e 8) identificaram setenta e dois por cento das atividades como sendo colaborativas, e dois outros analistas (analista 5 e 10) apontaram que eram oitenta e três por cento delas colaborativas (Figura 39).

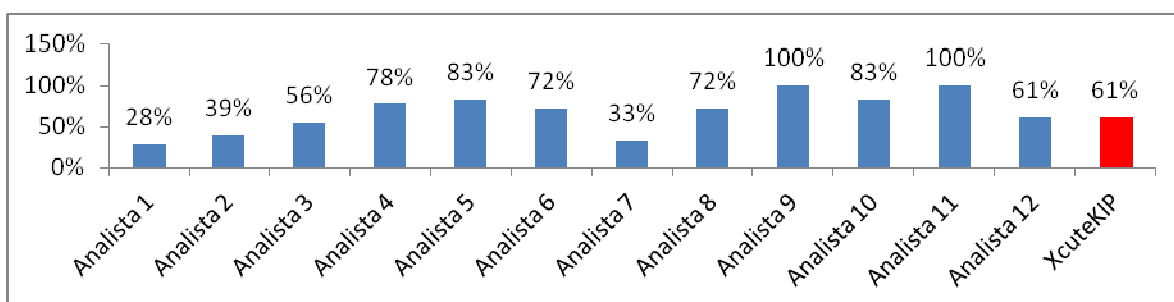


Figura 39: Percentual de recomendações de groupware para atividades colaborativas por cada analista e pela XCutKIP

A média da série de quantitativo de atividades colaborativas encontradas foi de sessenta e sete por cento. Embora tenha havido uma variação muito grande, retirando-se os extremos, que podem ser justificadas por excesso de individualismo ou de colaboração de algum participante, a maioria concorda que pouco mais da metade das atividades listadas são colaborativas. O XCuteKIP se desviou pouco desta média, uma vez que classificou sessenta e um por cento das atividades como colaborativas.

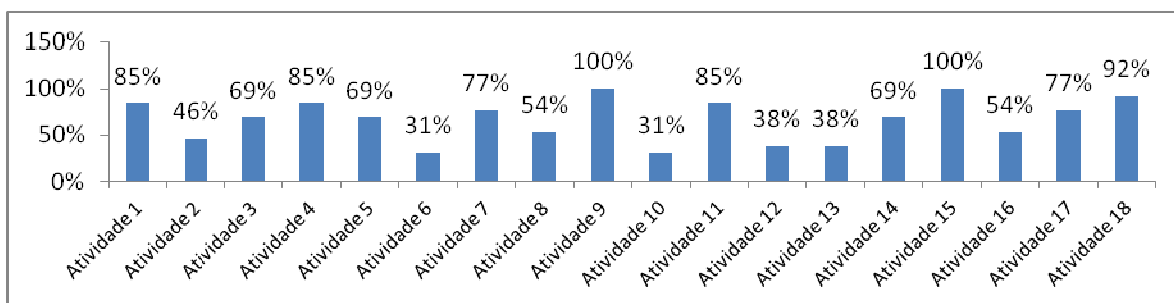


Figura 40: Percentual de participantes que classificaram as atividades como colaborativas

A figura 40 retrata outra divergência entre os participantes do estudo de caso. O gráfico mostra o grau de concordância dos participantes em classificar uma atividade como atividade colaborativa. Só houve unanimidade entre os participantes, incluindo a classificação feita pelo XCuteKIP, nas atividades 9 e 15. Não houve nenhuma dúvida para o grupo que “Levantar a lista de desejos de cada grupo” e “Obter autorização e verificação para passar ao próximo passo de desenvolvimento (e.g. a demonstração e a validação)” são atividades colaborativas. Por outro lado, “Identificar modelo de domínio e modelo de arquitetura” para o XCuteKIP é uma atividade realizada de forma individual. Apenas quatro analistas (trinta e um por cento do grupo) discordaram desta avaliação.

Estes dados podem indicar que prevaleceu as experiências anteriores de cada participante ao classificar as atividades como colaborativas ou não colaborativas.

Os resultados do estudo de caso relativos à importância dada à estrutura do discurso, quantidade de interlocutores na atividade e sincronia da atividade podem ser observados nas figuras 41, 42 e 43. Uma possível interpretação da análise destes resultados favorece a importância do método proposto neste trabalho, uma vez que no momento de análise das atividades para fins de recomendação de serviços de colaboração os participantes ficaram bastante confusos em relação às quais grupos de atividades deveriam recomendar. Tome-se o exemplo de participantes que sugeriram serviços de colaboração que apoiam a relação entre interlocutores de um para um e muitos para muitos para uma mesma atividade, sem levar esta

característica em consideração. Ou o serviço de um para um não será eficiente ou o serviço muitos para muitos será demais. Este mesmo problema não foi verificado na recomendação do XCuteKIP, que levou em consideração estas características das atividades.

A hipótese desta pesquisa prevê uma recomendação automática de serviços colaborativos para as atividades de um PIC. A ferramenta analisa a modelagem do PIC e é apoiada pelos algoritmos de recomendação para decidir os pontos confusos.

O resultado apresentado na figura 41 verifica que apenas quarenta e dois por cento daqueles que fizeram mais de uma recomendação para uma determinada atividade fizeram questão de não recomendar *groupware* que apresentavam estruturas de discurso distintas.

A estrutura de discurso, como analisado na sessão 3.1, está relacionada com a formulação do discurso e a maneira como as respostas são estruturadas. Esta característica é importante quando se deseja saber se uma mensagem pode ser respondida para uma ou mais questões, por exemplo. Nem todos os participantes, envolvidos no estudo de caso, atentaram em não escolher atividades que eram antagônicas quanto a estrutura de discurso. Ressaltando a dificuldade da escolha de ferramentas adequadas quando estas questões são cruciais para a escolha correta do serviço colaborativo.

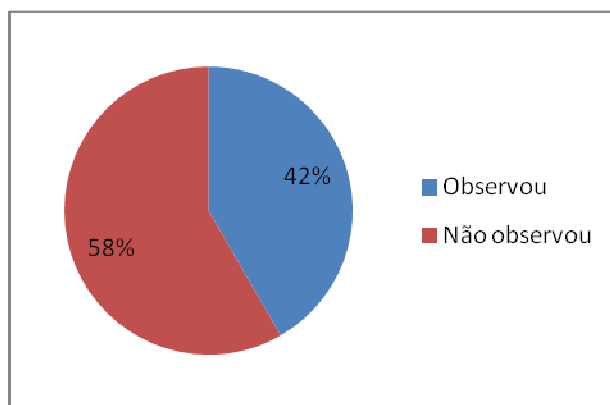


Figura 41: Importância da estrutura do discurso para escolha do *groupware* de apoio à atividade

Em relação à quantidade de interlocutores, a metade do grupo não fez a menor distinção entre *groupware* voltado para comunicação para poucos ou muitos interlocutores. Isto mostrou que esta característica não é consensualmente fundamental para a recomendação de *groupware* para atividades colaborativas, embora tenha sido um pouco mais relevante que a propriedade de estrutura do discurso.

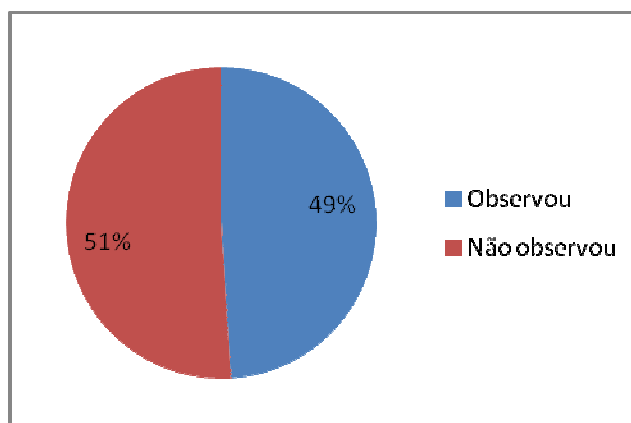


Figura 42: Importância de quantidade de interlocutores para escolha da atividade

A propriedade mais reconhecida como importante para a recomendação de *groupware* foi sem dúvida a sincronia, possivelmente porque é uma característica fácil de reconhecer entre os participantes. Dois terços do grupo realizaram suas recomendações de *groupware* para as atividades reconhecendo esta propriedade como importante. Para estes dois terços, um conjunto de recomendações pertenciam ao grupo de ferramentas síncrona ou assíncrona, e não ambos (Figura 43).

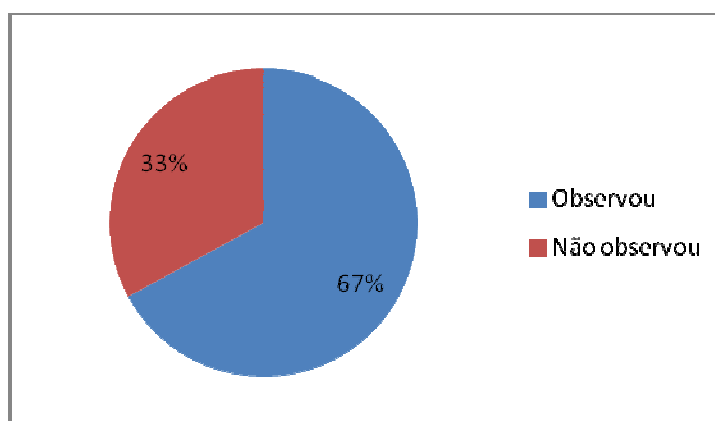


Figura 43: Importância de sincronia na escolha do *groupware* para atividade

Com relação ao objetivo do estudo de caso de identificar se os serviços colaborativos recomendados automaticamente pelo XCuteKIP são adequados para apoiar as atividades colaborativas do PIC, observou-se (figura 44) que o XCuteKIP indicou onze atividades como colaborativa, a mesma quantidade de atividades foi encontrada manualmente pelo analista 3. O analista 12 por sua vez indicou que dez atividades eram colaborativas. Depois de indicar

que uma atividade é colaborativa precisa recomendar o serviço colaborativo. A quantidade de recomendação coincidente foi alta, porém em algumas atividades tais como “Identificar especialistas do domínio da aplicação e de desenvolvimento” a recomendação do XCuteKIP foi completamente diferente que os outros dois analistas. Os dois analistas concordaram em recomendar os serviços de Conversa Telefônica, Vídeo Conferencia, Áudio Conferencia, enquanto que o XCuteKIP recomendou o serviço de Correio Eletrônico. Todas as recomendações dos analistas e do XCuteKIP estão no anexo II. A taxonomia teve grande importância para as recomendações do XCuteKIP.

As questões discursivas deram insumos para realizar a análise qualitativa da proposta desta dissertação.

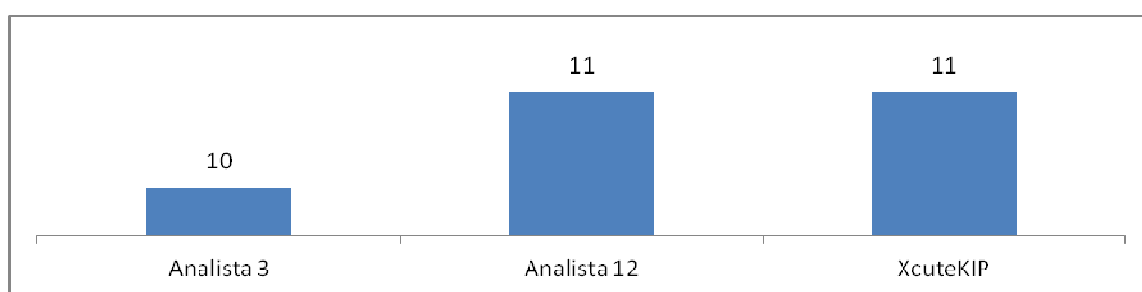


Figura 44: Quantidade de Atividades Recomendadas para o Configurator (Anexo III)

Nove das onze respostas em relação à primeira questão do bloco, que perguntou se o XCuteKIP poderia auxiliar nas atividades de levantamento de requisitos, foram positivas. Seguidos comentários como [*“Sim. Acredito que a ferramenta permite filtrar a melhor opção de groupware, evitando impasses ao longo do processo de desenvolvimento das atividades”*] e [*“Sim, a ferramenta auxilia bastante o analista de requisitos no processo de escolha que atende ao processo”*] sugerem que a ferramenta é útil para o processo de levantamento de requisitos. Entretanto, os comentários que seguem a linha como [*“Não porque vai depender da Justificativa e também do ambiente de trabalho se o ambiente é muito setorizado ou não”*] apontam a complexidade do processo intensivo em conhecimento.

Em geral, o XCuteKIP recebeu forte aprovação dos participantes, cem por cento dos que responderam a terceira pergunta do terceiro bloco foram favoráveis ao uso da ferramenta, mesmo aqueles que declararam que a opinião do especialista é suprema.

Resgatando-se a questão que norteou esta pesquisa (Como recomendar automaticamente serviços colaborativos adequados para apoiar atividades colaborativas em um Processo Intensivo em Conhecimento?), os resultados apontam indícios de que o método

utilizado foi capaz de classificar adequadamente as atividades colaborativas. Entretanto, o aspecto da experiência do participante teve impacto nesta decisão. Como as atividades no XCuteKIP não foram instanciadas por cada um dos participantes, mas sim por um analista sênior, o nível de concordância entre a ferramenta e os participantes foi apenas na média. Além disso, o alto percentual de recomendações coincidentes da XCuteKIP em relação às recomendações manuais apontam para a adequabilidade dos serviços recomendados no cenário investigado.

A utilização de uma taxonomia para fazer o mapeamento entre atividades de um PIC e a KIPO confere à XCuteKIP flexibilidade para cobrir diversos cenários que não foram endereçados pelo estudo de caso realizado nesta dissertação. Tipos de serviços diferentes dos contemplados na proposta (como os de Coordenação e de Cooperação) poderiam ser incluídos, acrescentando estes tipos na taxonomia e adicionando os novos axiomas capazes de recomendar os serviços com esta classificação. Estes passos seriam suficientes para que novos serviços sejam classificados com os novos tipos e habilitados para disponibilização de recomendação pela XCuteKIP.

Capítulo 6 – Conclusão e Trabalhos Futuros

Este capítulo apresenta as conclusões da pesquisa realizada, mostrando suas contribuições, pontos positivos e negativos, além das dificuldades encontradas no seu desenvolvimento. Descreve, ainda, possíveis trabalhos futuros que podem surgir baseados neste.

6.1. Resumo da pesquisa

Este trabalho pesquisou os aspectos de processos intensivos em conhecimento que tem por natureza serem dinâmicos, complexos, com constantes mudanças de objetivos e em grande parte dos casos, são conduzidos de forma colaborativa. Esta dissertação focou no aspecto colaborativo, com o intuito de apoiar a recomendação de suas atividades. Mais especificamente, esta recomendação se concentrou no suporte às atividades colaborativas.

A proposta deste trabalho contemplou um método que utilizou como base uma ontologia de processos intensivos em conhecimento para instanciar processos desta natureza. Foi utilizada uma arquitetura orientada a serviços para fornecer serviços colaborativos e uma arquitetura para, através da execução dos passos contidos no método, prover uma recomendação dos serviços da WGWSOA.

Os resultados alcançados através da utilização do método e da implementação da arquitetura XCuteKIP nos permitem afirmar que a instanciação de processos intensivos em conhecimento com o uso da ontologia KIPO, e o mapeamento com os serviços colaborativos, foram fundamentais para recomendar serviços colaborativos de maneira a facilitar a recomendação de serviços para dar apoio às atividades colaborativas em tais processos.

6.2. Contribuições

As principais contribuições deste trabalho incluem:

- Definição de um mapeamento da ontologia KIPO para um esquema relacional de um SGBD: Devido às dificuldades impostas sobre a abordagem de manipulação de arquivos (OWL e RDF) sobre grandes ontologias e diversas instâncias, a abordagem de armazenamento de ontologias em SGBDs ainda apresentam como vantagens o fato de usufruírem de todos os recursos desenvolvidos e amplamente testados por essa tecnologia, tais como processamento, segurança e otimização de consultas. Problemas relacionados a processamento de arquivos, como baixo desempenho no tempo de resposta de consultas (por conta da verificação de consistência da ontologia e o processamento de inferências) podem ser evitados;
- Especificação de um método para identificar o tipo de colaboração para cada atividade de um PIC, baseando-se em uma conceituação do domínio de colaboração: A abordagem proposta torna explícitas as características de um processo intensivo em conhecimento e seus aspectos colaborativos, recomendando serviços colaborativos que mais adequadamente auxiliem as atividades colaborativas em um PIC;
- Especificação formal de conceitos de colaboração em PIC: a presente pesquisa também contribui com a literatura de PICs ao avaliar como os elementos da KIPO apoiam a colaboração em PIC e definir axiomas que descrevem formalmente a colaboração em atividades de PIC.

Por fim, a pesquisa contribui com a literatura de colaboração ao aplicar a arquitetura WGWSOA a cenários inéditos de PIC.

6.3. Limitações da Pesquisa

Como premissa de aplicação do método proposto, é mandatário que as atividades do processo intensivo em conhecimento tenham sido instanciadas com as classes da KIPO e exista no repositório de serviços colaborativos funcionalidades que satisfaçam o mapeamento.

6.4. Trabalhos Futuros

Nos trabalhos futuros, pretende-se estender a arquitetura para, além da recomendação, também apoiar a execução das atividades que foram recomendadas, e conseqüentemente apoiar mais efetivamente as atividades desempenhadas pelos participantes do processo. Embora a execução dos serviços colaborativos não tenha sido o foco desta dissertação, foram

realizados testes da WGWSOA invocando os serviços tanto com RMI (*Remote Method Invocation*) quanto com webservices para avaliar a facilidade do uso dos serviços colaborativos e dos serviços de percepção. Verificou-se ser mais vantajoso utilizar webservices, por permitir que um maior número de plataformas pudesse interoperar com os serviços colaborativos. Em um trabalho futuro, pode-se expor os serviços de maneira mais interoperável entre plataformas, que neste caso podem incluir REST (*REpresentational State Transfer*) e webservices.

Pretende-se ainda realizar novos estudos de caso em diferentes cenários de processos intensivos em conhecimento distintos para tornar possível generalizar os resultados. Os participantes do estudo de caso poderiam ser especialistas no domínio do processo intensivo em conhecimento e em *groupware*. Com o estudo de caso realizado nesta pesquisa consegue-se apenas fazer afirmações sobre cenários semelhantes ao investigado e nas mesmas condições em que foi realizado o estudo.

Além disso, se propõe a ampliação da arquitetura para utilizar o estilo arquitetural REST na camada de serviços colaborativos, para promover sobre estes maior flexibilização. Esta arquitetura visa integrar qualquer *middleware* ou repositório de serviço que utilize esta arquitetura.

A extensão da XCuteKIP levando-se em conta a composição de serviços colaborativos também é uma rica futura direção de pesquisa, acompanhada pela proposta do estilo arquitetural REST.

Finalmente, pretende-se otimizar o método de recomendação para que possa guardar as informações de execuções anteriores dos processos e utilizá-las para melhorar a recomendação baseada nestas informações.

Referências

- ABECKER, A.; BERNARDI, A.; ESLT. L. V.; HERTERICH, R.; HOUY, C.; MÜLLER, S.; DIOUDS, S.; MENTZAS, G.; LEGAL, M.: Workflow-Embedded Organizational Memory Access: The DECOR Project. KnowTech, 2001.
- ALLEN, J. F. Maintaining Knowledge About Temporal Intervals. *Communications of the ACM*, Vol.26, no. 11, 1983.
- ANDERSSON, Birger; BIDER, Ilia; PERJONS, Erik. Business Process Support System as a Tool for Communication/Collaboration. IbisSoft AB – Internal Report. 2004.
- BAIÃO, F.; SANTORO, F. M.; IENDRIKE, H.; CAPPELLI, C.; LOPES, M.; NUNES, V. T.: Towards a Data Integration Approach based on Business Process Models and Domain Ontologies. In: International Conference on Enterprise Information Systems (ICEIS), Barcelona, 2008.
- BOLSTAD, Cheryl A.; ENDSLEY, Mica R. Tools for supporting team sa and collaboration in army operations. In: Advanced Decision Architecture Conference. 2003.
- BORGES, M. R. S.; CAVALCANTI, M. C. R.; CAMPOS, M. L. M. Suporte por computador ao trabalho cooperativo. In: XV Congresso da Sociedade Brasileira de Computação. XV Jornada de Atualização em Informática. JAI'95. Anais. Canela, RS. Julho, 1995.
- BROMBY, M.; MCMILLAN, M; MCKELLAR, P.: A CommonKADS Representation for a Knowledge-based System to Evaluate Eyewitness Identification. in *International Review of Law, Computers and Technology*, Vol. 17, No. 1, pp 99 – 108. 2003.
- BURKE, Robin. Hybrid Recommender Systems: Survey and Experiments. Department of Information Systems and Decision Sciences. California State University. Fullerton, California, United States. Kluwer Academic Publishers. (2002).
- CALVÃO, L.; PIMENTEL, M.; FUKS, H. E.; GEROSA, M. A. Uma taxonomia para os Meios de Conversação por Computador. *Anais do SBSC, IX Simpósio Brasileiro de Sistemas Colaborativos*, pp. 29-34. São Paulo, SP. 2012.
- CAMPOS, M. L. A. Modelização de domínios de conhecimento: uma investigação de princípios fundamentais. *Ci. Inf., Brasília*, v.33, n.1, p. 22-32, jan./abr. 2004.

CAZELLA, Sílvio César; NUNES, M. A. S. N. REATEGUI, Eliseo. A Ciência da Opinião: Estado da arte em Sistemas de Recomendação. XXX Congresso da SBC Jornada de Atualização da Informática, Porto Alegre, Brasil, 52p. 2010.

CHRISTEL, Michael G.; KYO, C. Kang. Issues in requirements elicitation. No. CMU/SEI-92-TR 12. Carnegie-Mellon Univ Pittsburgh pa Software Engineering Inst, 1992.

COLEMAN, D. Levels of collaboration. Collaborative Strategies, March 2002 Editorial, URL: <http://www.collaborate.com>. Latest visit February 2005.

CORREIA, A. J. G. Caracterização do estado da arte de CSCW. Dissertação de Mestrado. Escola de Ciências e Tecnologia. Trás-os-Montes, Portugal. 2011.

DAVENPORT, T. H.; PRUSAK, L. Working Knowledge. Harvard Business School Press, 2000.

DAVENPORT, Thomas H. Process innovation: reengineering work through information technology. Harvard Business Press, 1993.

DAVID, José Maria N.; MACIEL, Rita Suzana P. WGWSOA – Implementando Serviços de Colaboração em uma Infra-estrutura de Middleware. 2010.

DUMAS, Marlo; LA ROSA, Marcello; MENDLING, Jan; REIJERS, Hajo A. Fundamentals of Business Process Management. Springer, 2013.

ELLIS, C. A. An evaluation framework for collaborative systems. Technical Report, CU-CS-901-00, Department of Computer Science, University of Colorado, Boulder, USA. 2000.

ELLIS, C. A.; GIBBS, S. J.; REIN, G. L. Groupware: Some issues and experiences. Communications of the ACM, Vol. 34, No. 1. pp. 38-58. 1991.

EPPLER, Martin J.; SEIFRIED, Patrick M.; RÖPNACK, Axel. Improving knowledge intensive processes through an enterprise knowledge medium, Proceedings of the 1999 ACM SIGCPR Conference on Computer personnel research, p. 222-230, April 08-10, 1999, New Orleans, Louisiana, United States

ERL, Thomas. Service-Oriented Architecture (SOA): concepts, technology, and design. 2005.

FERREIRA, A. B. H. Novo Dicionário da língua portuguesa. 2. ed. Rio de Janeiro: Nova Fronteira, 1986.

FRANÇA, J. An Ontology for Defining Knowledge Intensive Processes. (In Portuguese), MsC Dissertation, UNIRIO, Brazil, 2012.

FRANÇA, J. B. Dos S. Uma Ontologia para Definição de Processos Intensivos em Conhecimento. Dissertação de Mestrado. Unirio. 2012.

FUCKS, Hugo; RAPOSO, Alberto B.; GEROSA, Marcos. A. “Do Modelo de Colaboração 3C à Engenharia de Groupware”, Simpósio Brasileiro de Sistemas Multimídia e Web – Webmedia 2003.

GEROSA, Marco Aurélio. Categorização e estruturação de mensagens textuais em ambientes virtuais de colaboração. 2002.

GEROSA, M. A.; PIMENTEL, M.; FILIPPO, D.; BARRETO, C.G.; RAPOSO, A.; FUKS, H.; LUCENA, C. J. P. Componentes Baseados no Modelo 3C para o Desenvolvimento de Ferramentas Colaborativas. Anais do 5º Workshop de Desenvolvimento Baseado em Componentes - WDBC 2005, 7-9 de novembro de 2005, Juiz de Fora, MG, pp. 109-112.

GIL, Antonio Carlos. Como elaborar projetos de pesquisa. São Paulo, v. 5, 2002.

GONÇALVES, José Ernesto Lima. As empresas são grandes coleções de processos. Revista de administração de empresas. 2000.

GUIZZARDI, G.; LOPES, M.; BAIÃO, F.; FALBO, R., 2010.: On the importance of truly ontological representation languages. In: International Journal of Information Systems Modeling and Design (IJISMD), Information Resources Management Association (IRMA), Editor-in-Chief: Remigijus Gustas, IGI Publishing, Hershey-New York, USA, Volume 1, Issue 2, April-June.

HAGEN, C. R.; RATZ, D.; POVALEJ, R.: Towards self-organizing knowledge intensive processes. Journal of universal knowledge management, v.0, n. 2, p.148-169, 2005.

HAMMER, Michael; CHAMPY, James. Reengineering the Corporation: Manifesto for Business Revolution, A. HarperCollins, 2009.

HARMON, Paul. Business process change: a guide for business managers and BPM and six sigma professionals. Morgan Kaufmann, 2010.

KHOMYAKOV, M.; BIDER, I. Achieving Workflow Flexibility through Taming the Chaos. OOIS 2000 - 6th international conference on object oriented information systems, pp.85-92, Springer, 2000.

LEITNER, Philipp; ROSENBERG, Florian; DUSTDAR, Schahram. Daios: Efficient dynamic web service invocation. Internet Computing, IEEE, v. 13, n. 3, p. 72-80, 2009.

LOPES, M. G.: Modelagem Conceitual de Regras de Negócio Baseada em Ontologia de Fundamentação. Dissertação de Mestrado. 223f. Programa de Pós- Graduação em Informática, Universidade Federal do Estado do Rio de Janeiro, Rio de Janeiro, 2011.

MOLZ, Kurt Werner; THOM, Lucinéia Heloisa; SCHEIDT, Neiva ESTUDO SOBRE MODELAGEM E APLICAÇÃO DE SISTEMAS DE WORKFLOW. Universidade de Santa Cruz do Sul – UNISC. 1999.

MOURA, Ednilson Veloso; SANTORO, Flavia Maria; BAIÃO, Fernanda Araujo. Collaboration support for knowledge-intensive processes through a service based approach. In: Computer Supported Cooperative Work in Design (CSCWD), 2013 IEEE 17th International Conference on. IEEE, 2013. p. 319-324

OASIS, SOA. OASIS reference model for service oriented architecture 1.0. www.oasis-open.org/committees/download.php/19679/soa-rm-cs.pdf, 2006.

OLIVEIRA, F. F. Uma ontologia de colaboração e suas aplicações. Dissertação de Mestrado. Unirio. 2009

PAPAVASSILIOU, Giorgos; NTIOUDIS, Spyridon Adreas Abecker; MENTZAS, Gregoris. Managing Knowledge in Weakly Structured Administrative Process. The Third European Conference on Organization Knowledge, Learning and Capabilities. 2002.

PAPAZOGLU, Michael P.; TRAVERSO, Paolo; DUSTDAR, Schahram; LEYMAN, Frank. Service-oriented computing: State of the art and research challenges. *Computer*, v. 40, n. 11, p. 38-45, 2007.

PENDERGAST, M., HAYNE, S. Groupware and social networks: will life ever be the same again? *Information and Software Technology*, 1999, 41, 311-318.

PENICHER, Victor M. Ruiz *et al.* A classification method for CSCW systems. *Electronic Notes in Theoretical Computer Science*, v. 168, p. 237-247, 2007.

PEREIRA, A. C. T. D.: Modelagem do Processo Cognitivo de Tomada de Decisão como Informação de Contexto para Apoio à Aprendizagem Organizacional. 2010. Dissertação de Mestrado. 226f. Programa de Pós-Graduação em Informática, Universidade Federal do Estado do Rio de Janeiro, Rio de Janeiro, 2010.

PINHEIRO, Manuele Kirsch; LIMA, José Valdeni de; BORGES, Marcos R. S. Awareness em Sistemas de Groupware. 2001.

POLTROCK, Steven; GRUDIN, Jonathan. Computer Supported Cooperative Work and Groupware, Tutorial Notes, CHI92, Monterey, California, 1992.

PROBST, G.; RAUB, S.; ROMHARDT, K. Wissen managen. Wie Unternehmen ihre wertvollste Ressource sinnvoll nutzen. 1997.

REMUS, U. Process oriented knowledge management, Concepts and modeling. 2002. Tese de Doutorado. PhD thesis, University of Regensburg, Germany, Regensburg.

SANTOS, A. R. dos (Org.). Knowledge Management: An Experience for the Enterprise Success. (In Portuguese). Curitiba: Champagnat, 2001.

SINGULARITY, W. P. Case Management: Combining Knowledge with Process. Singularity, Ltda. 2009.

SOMMERVILLE, I. Software Engineering, 8th Ed., Addison-Wesley, 2007.

STOITSEV, Todor; SCHEIDL, Stefan; FLENTGE, Felix; MÜHLHÄUSER, Max. Enabling End Users to Proactively Tailor Underspecified, Human Centric Business Processes: "Programming by Example" of Weakly-Structured Process Models. In: *Enterprise Information Systems*. Springer Berlin Heidelberg, 2009..

STRIJBOSCH, Koen; READER, R. U.; VAN BOMMEL, P. Master Thesis Adaptive Case Management. 2011.

STROHMAIER, Markus B. B-KIDE: A Framework and a Tool for Business Process Oriented Knowledge Infrastructure Development. PhD Thesis. Institute for Knowledge Management and Knowledge Visualization Graz University of Technology, Austria. 2004.

TSAI, W. T.; CHEN, Yinong Gary Bitter; MIRON, Dorina. Introduction to Service-Oriented Computing. In IFIP International Federation for Information Processing 2004, pages 294-310, <http://www.public.asu.edu/%7Eychen10/activities/SOAWorkshop/Background.pdf>, 2004.

URIONA, Maldonado, M. Análise do Impacto das Políticas de Criação e Transferência de Conhecimento em Processos Intensivos em Conhecimento: Um modelo de Dinâmica de Sistemas, 2008.

WAGNER, G., GIURCA, A., Lukichev, S., 2006, “A Usable Interchange Format for Rich Syntax Rules. Integrating OCL, RuleML and SWRL”. In: Reasoning on the Web Workshop at WWW2006

WESKE, M. Business Process Management – Concepts, Languages, Architectures. Verlag; Berlin; Heidelberg: Springer, 2007.

WFMC. Workflow Management Coalition, Interface 1: Process Definition Interchange Process Model., Version 1.1, 1999.

Anexo I – Glossário

| | |
|--|---|
| Communicative Interaction | Representa a comunicação entre dois ou mais agentes através de troca de informações: envio e percepção. |
| Message | A mensagem é o conteúdo proposicional trocado de um ato de comunicação e, conseqüentemente, de uma interação comunicative entre agentes |
| Communicative Act | Representa a ato de enviar uma mensagem |
| Perception | Representa a ação de perceber uma mensagem |
| Idiomatic Language | Linguagem que usa um idioma para a sua representação |
| Idiomatic Interaction | Interação Comunicativa que envolve uma mensagem codificada em uma linguagem idiomática |
| Collaborative Session | Eventos complexos nos quais participantes interagem através de participações/contribuições com o propósito de colaborar |
| Collaborative Agreement | Representa um tipo de Social role que delibera uma sessão colaborativa. |
| Collaborative Agreement Universal | Tipo de Social Relator Universal que pode ser instanciado por um Collaborative Agreement e fornece a condição de ligação de um Papel colaborativo |
| Collaborative Role | Papel Social que agentes se comprometem a desempenhar devido aos acordos por eles assumidos |

Anexo II – Respostas ao Questionário de Recomendação de Groupware

| Atividade | Analista 1 | Analista 2 | Analista 3 | Analista 4 | Analista 5 |
|--|--|---|---|--|-----------------------------------|
| Identificar grupos relevantes através de múltiplos níveis da organização. | Lista de discussão, Forum de discussão | Lista de discussão, Forum de discussão, blog | Correio Eletrônico, Lista de Discussão, Bate papo | Mapa de Discussão | |
| Determinar os contextos operacional e do problema, incluindo a definição dos modos operacionais, objetivos e cenários de missão como apropriados. | | | | Microblog, blog | |
| Identificar sistemas similares | Blog, Microblog | Blog, Microblog, Forum de discussão, Lista de discussão, mapa de discussão. | | Correio Eletrônico e Fórum de discussão. | Lista de discussão, Mapa ou Fórum |

| | | | | | |
|---|--|--|--|--------------------|--------------------------------------|
| Realizar análise de contexto | | Correio Eletrônico, SMS, Correio de Voz, Mensageiro Eletrônico, Vídeo Chamado, Conversa Telefônica, Bate-papo, Áudio Conferência, Vídeo Conferência. | Correio Eletrônico, | Fórum de discussão | |
| Identificar especialistas do domínio da aplicação e de desenvolvimento. | | | Áudio-conferencia, Vídeo conferencia, Conversa telefônica, Bate-papo | Fórum de discussão | Áudio Conferência, Vídeo Conferência |
| Identificar modelo de domínio e modelo de arquitetura. | | | | | Lista de Discussão |
| Conduzir pesquisas tecnológicas, para mais tarde fazer estudo de viabilidade e análise de risco. Identificar custos e restrições à implementação impostas pelo patrocinador. | | | Correio de Voz, Bate-papo, Fórum de discussão. | Blog, microblog | Lista de Discussão |

| | | | | | |
|--|---|---|---------------------------|---|--|
| <p>Classificar a lista de acordo com funcionais, não funcionais, restrições de ambiente, restrições de projeto e ainda de acordo com as partições definidas pelo modelo de domínio e pelo paradigma de desenvolvimento.</p> | | | | | <p>Lista de Discussão, Mapa ou Fórum</p> |
| <p>Levantar a lista de desejos de cada grupo.</p> | <p>Correio Eletrônico, SMS, Correio de Voz, Mensageiro Eletrônico, Vídeo Chamado, Conversa Telefônica, Bate-papo, Áudio Conferência, Vídeo Conferência.</p> | <p>Correio Eletrônico, SMS, Correio de Voz, Mensageiro Eletrônico, Vídeo Chamado, Conversa Telefônica, Bate-papo, Áudio Conferência, Vídeo Conferência.</p> | <p>Correio Eletrônico</p> | <p>Lista de discussão, fórum de discussão</p> | <p>Áudio Conferência, Vídeo Conferência, conversa telefônica</p> |
| <p>Realizar uma análise de riscos, investigando técnicas, custos, prazos e incluindo análise de custos e benefícios e viabilidade baseado na disponibilidade da tecnologia.</p> | | | | | <p>Lista de Discussão</p> |

| | | | | | |
|---|--------------------|--|---|-----------------------|---|
| <p>Responder questões da forma “Por que você precisa de X?”, a partir de raciocínio abstrato. Isso auxilia a transformar o raciocínio das questões sobre “como?” para as questões sobre “o quê?”.</p> | | Correio Eletrônico | Correio Eletrônico, e Conversa telefônica. | Mensageiro Eletronico | Áudio Conferência, Vídeo Conferência, conversa telefônica |
| <p>Estudar como o sistema pode ser implementado de forma incremental, investigando modelos arquiteturais apropriados.</p> | | | | | Lista de Discussão, Fórum ou Mapa |
| <p>Priorizar requisitos baseados em custo e dependência.</p> | | | | Correio Eletrônico | Correio Eletrônico |
| <p>Determinar funcionalidades críticas para a missão.</p> | | | | Microblog | Lista de Discussão, Fórum ou Mapa |
| <p>Obter autorização e verificação para passar ao próximo passo de desenvolvimento (e.g. a demonstração e a validação).</p> | Correio Eletrônico | Correio Eletrônico, SMS, Correio de Voz, Mensageiro Eletrônico, Vídeo Chamado, Conversa Telefônica, Bate-papo, Áudio Conferência, Vídeo Conferência. | Correio Eletrônico, Bate-papo, Conversa telefônica. | Correio Eletronico | Vídeo conferência |

| | | | | | |
|---|---|--|---|---|---|
| Validar que os requisitos estão concordando com os objetivos iniciais. | | | Correio Eletrônico, Bate-papo, Conversa telefônica, Áudio-conferencia, Vídeo conferencia. | Correio Eletrônico, Conversa telefônica | Vídeo Conferência |
| Resolver a maior quantidade possível de pontos em aberto. | | | Lista de discussão, Conversa telefônica | Blog e Microblog | : Vídeo, áudio Conferência ou conversa telefônica |
| Resolver conflitos e verificar consistência. | Correio Eletrônico, Correio de Voz, Vídeo Chamado, Conversa Telefônica, Áudio Conferência, Vídeo Conferência. | Correio Eletrônico, SMS, Correio de Voz, Mensageiro Eletrônico, Vídeo Chamado, Conversa Telefônica, Bate-papo, Áudio Conferência, Vídeo Conferência. | Conversa telefônica, Bate-papo, Correio Eletrônico. | Videoconferência e Áudio - conferencia. | Vídeo Conferência |

| Atividade | Analista 6 | Analista 7 | Analista 8 | Analista 9 | Analista 10 |
|--|-------------------|--|-------------------|--|--------------------|
| Identificar grupos relevantes através de múltiplos níveis da organização. | Blog | Lista de discussão, Forum de discussão, blog | | Correio Eletrônico, Lista de Discussão, Bate papo. | Mapa de Discussão |

| | | | | | |
|--|-----------------------|--|---|---|--|
| Determinar os contextos operacional e do problema, incluindo a definição dos modos operacionais, objetivos e cenários de missão como apropriados. | Lista de discussão | | bate-papo, audio conferência, video conferencia | Correio Eletrônico, Conversa telefônica, Bate-papo, Áudio-conferencia, Vídeo conferência. | Microblog, blog |
| Identificar sistemas similares | Lista de discussão | Blog, Microblog, Fórum de discussão, Lista de discussão, mapa de discussão. | | : Correio, Blog, Microblog, Fórum de Discussão. | Correio Eletrônico e Fórum de discussão. |
| Realizar análise de contexto | Mensageiro eletrônico | Correio Eletrônico, SMS, Correio de Voz, Mensageiro Eletrônico, Vídeo Chamado, Conversa Telefônica, Bate-papo, Áudio | bate-papo, audio conferência, video conferencia | Correio Eletrônico, | Fórum de discussão |
| Identificar especialistas do domínio da aplicação e de desenvolvimento. | | | bate-papo, audio conferência, video conferencia | Correio Eletrônico, SMS, Correio de Voz, Conversa telefônica, Bate-papo | Fórum de discussão |
| Identificar modelo de domínio e modelo de arquitetura. | | | bate-papo, audio conferência, video conferencia | Correio de Voz, Blog, Microblog, Bate-papo. | |

| | | | | | |
|--|---------------------------|---|--|---|------------------------|
| <p>Conduzir pesquisas tecnológicas, para mais tarde fazer estudo de viabilidade e análise de risco. Identificar custos e restrições à implementação impostas pelo patrocinador.</p> | <p>Lista de discussão</p> | | <p>Blog, Microblog</p> | <p>Correio de Voz, Bate-papo, Fórum de discussão.</p> | <p>Blog, microblog</p> |
| <p>Classificar a lista de acordo com funcionais, não funcionais, restrições de ambiente, restrições de projeto e ainda de acordo com as partições definidas pelo modelo de domínio e pelo paradigma de desenvolvimento.</p> | <p>Lista de discussão</p> | | <p>bate-papo, audio conferência, video conferencia</p> | <p>Correio Eletrônico, Bate-papo.</p> | |
| <p>Levantar a lista de desejos de cada grupo.</p> | <p>Lista de discussão</p> | <p>Correio Eletrônico, SMS, Correio de Voz, Mensageiro Eletrônico, Vídeo Chamado, Conversa Telefônica, Bate-papo, Áudio Conferência, Vídeo Conferência.</p> | <p>Email, Lista de discussão, forum</p> | <p>Correio Eletrônico, Correio de Voz, Bate-papo.</p> | |

| | | | | | |
|---|-----------------------|--|---|---|-----------------------|
| Realizar uma análise de riscos, investigando técnicas, custos, prazos e incluindo análise de custos e benefícios e viabilidade baseado na disponibilidade da tecnologia. | Correio Eletrônico | | | Correio Eletrônico, Blog, Microblog, Bate-papo, Lista de discussão. | |
| Responder questões da forma “Por que você precisa de X?”, a partir de raciocínio abstrato. Isso auxilia a transformar o raciocínio das questões sobre “como?” para as questões sobre “o quê?”. | Mensageiro eletrônico | | bate-papo, audio conferência, video conferencia | Correio Eletrônico, Correio de Voz, Bate-papo, Conversa telefônica. | Mensageiro Eletrônico |
| Estudar como o sistema pode ser implementado de forma incremental, investigando modelos arquiteturais apropriados. | | | bate-papo, audio conferência, video conferencia | Correio Eletrônico, Blog, Microblog, Bate-papo, Lista de discussão, Forum de discussão. Conversa telefônica | Videoconferência |
| Priorizar requisitos baseados em custo e dependência. | | | | Correio Eletrônico, Bate-papo, Conversa telefônica. | Correio Eletrônico |
| Determinar funcionalidades críticas para a missão. | Correio Eletrônico | | bate-papo, audio conferência, video conferencia | Conversa telefônica, Bate-papo, Áudio-conferencia, Vídeo conferencia. | Microblog |

| | | | | | |
|---|--------------------|--|--------------------|---|---|
| Obter autorização e verificação para passar ao próximo passo de desenvolvimento (e.g. a demonstração e a validação). | Correio Eletrônico | Correio Eletrônico, SMS, Correio de Voz, Mensageiro Eletrônico, Vídeo Chamado, Conversa Telefônica, Bate-papo, Áudio Conferência, Vídeo Conferência. | Correio Eletrônico | Correio Eletrônico, Bate-papo, Conversa telefônica. | Correio Eletronico |
| Validar que os requisitos estão concordando com os objetivos iniciais. | Correio Eletrônico | | | Correio Eletrônico, Bate-papo, Conversa telefônica, Áudio-conferencia, Vídeo conferencia. | Correio Eletrônico, Conversa telefônica |
| Resolver a maior quantidade possível de pontos em aberto. | Correio Eletrônico | | Correio Eletrônico | Correio Eletrônico, Lista de discussão, Bate-papo, Conversa telefônica. | Blog e Microblog |

| | | | | | |
|---|--|--|---|--|---|
| Resolver conflitos e verificar consistência. | | Correio Eletrônico, SMS, Correio de Voz, Mensageiro Eletrônico, Vídeo Chamado, Conversa Telefônica, Bate-papo, Áudio Conferência, Vídeo Conferência. | bate-papo, audio conferência, video conferencia | Conversa telefônica, Bate-papo, Correio Eletrônico | Videoconferência e Áudio - conferência. |
|---|--|--|---|--|---|

| Atividade | Analista 11 | Analista 12 | XCuteKIP |
|--|-----------------------|--------------------|---|
| Identificar grupos relevantes através de múltiplos níveis da organização. | Forum de discussão | Fórum de Discussão | Lista de discussão , Forum de discussão , Mapa de discussão |
| Determinar os contextos operacional e do problema, incluindo a definição dos modos operacionais, objetivos e cenários de missão como apropriados. | Tudo que for síncrono | | |
| Identificar sistemas similares | Tudo que for síncrono | | |
| Realizar análise de contexto | Tudo que for síncrono | Blog, Microblog | Blog, Microblog |

| Atividade | Analista 11 | Analista 12 | XCuteKIP |
|---|---------------------------|---|---|
| Identificar especialistas do domínio da aplicação e de desenvolvimento. | Lista e Mapa de discussão | Conversa Telefônica, Video Conferencia, Audio Conferencia | Correio Eletronico |
| Identificar modelo de domínio e modelo de arquitetura. | Mapa de discussão | | |
| Conduzir pesquisas tecnológicas, para mais tarde fazer estudo de viabilidade e análise de risco. Identificar custos e restrições à implementação impostas pelo patrocinador. | Forum de discussão | Fórum, Microblog, Mapa de Discussão | Bate-papo, Audio-conferência e Vídeo Conferência |
| Classificar a lista de acordo com funcionais, não funcionais, restrições de ambiente, restrições de projeto e ainda de acordo com as partições definidas pelo modelo de domínio e pelo paradigma de desenvolvimento. | Tudo que for síncrono | Audio Conferencia, Video Conferencia | Mensageiro Instantâneo, Vídeo-chamada e Áudio-chamada |
| Levantar a lista de desejos de cada grupo. | Mapa de discussão | Fórum, Blog, Microblog, Lista de Discussão | Blog, Microblog |

| Atividade | Analista 11 | Analista 12 | XCuteKIP |
|---|-----------------------|--|---|
| Realizar uma análise de riscos, investigando técnicas, custos, prazos e incluindo análise de custos e benefícios e viabilidade baseado na disponibilidade da tecnologia. | Tudo que for síncrono | | |
| Responder questões da forma “Por que você precisa de X?”, a partir de raciocínio abstrato. Isso auxilia a transformar o raciocínio das questões sobre “como?” para as questões sobre “o quê?”. | Forum de discussão | Blog, Microblog, E-mail | Mensageiro Eletrônico, Videochamada e Conversa telefônica |
| Estudar como o sistema pode ser implementado de forma incremental, investigando modelos arquiteturais apropriados. | Fórum | | |
| Priorizar requisitos baseados em custo e dependência. | Tudo que for síncrono | | |
| Determinar funcionalidades críticas para a missão. | Tudo que for síncrono | Blog, Microblog, Fórum, Lista de Discussão | Correio Eletrônico |
| Obter autorização e verificação para passar ao próximo passo de desenvolvimento (e.g. a demonstração e a validação). | Tudo que for síncrono | Video Conferencia, Audio Conferencia, E-mail | Correio Eletrônico |

| Atividade | Analista 11 | Analista 12 | XCuteKIP |
|---|-----------------------|--|---|
| Validar que os requisitos estão concordando com os objetivos iniciais. | Tudo que for síncrono | | |
| Resolver a maior quantidade possível de pontos em aberto. | Fórum de discussão | Video Conferencia, Audio Conferencia, E-mail | Fórum de discussão , Lista de discussão e Mapa de discussão , |
| Resolver conflitos e verificar consistência. | Tudo que for síncrono | Blog, Microblog, Lista de Discussão | Mensageiro Eletrônico, Vídeo Chamada e Conversa Telefônica |

Anexo III – Avaliação do XCuteKIP

| Pergunta aberta | Analista 1 | Analista 2 | Analista 3 | Analista 4 | Analista 5 |
|---|------------|--|------------|---|------------|
| <ul style="list-style-type: none">• Você considera que o sistema de recomendação implementado auxilia de fato a atividade de escolhas de groupware nos processos de levantamento de requisitos? | Sim. | Não porque vai depender da Justificativa e também do ambiente de trabalho se o ambiente é muito setorizado ou não. | Sim | Sim, ao relacionar algumas características do levantamento de requisito com algumas sugestões, a ferramenta faz com que o analista de sistema pense em como aquela escolha atende o processo. | Sim |

| Pergunta aberta | Analista 1 | Analista 2 | Analista 3 | Analista 4 | Analista 5 |
|--|--------------|--|------------|---|--|
| <ul style="list-style-type: none"> O que você considera que poderia ser melhorado ou incluído na ferramenta para um maior suporte das escolhas de groupware para processos de levantamento de requisitos? | | Antes de realizar o resultado das recomendações ter uma maneira de escolher o ambiente de trabalho, por causa da minha resposta de número 4. | | A possibilidade e de afinar a matriz de características do levantamento de requisito com as possíveis sugestões, enriquecendo cada vez mais a ferramenta e evitando o apontamento para sugestões indevidas. | Ser possível determinar a quantidade e de pessoas envolvidas na atividade, porque existem atividades que possui como recomendação mais de uma ferramenta. A quantidade e de pessoas facilitaria no filtro. |
| <ul style="list-style-type: none"> De uma forma geral, você avaliaria positivamente a ferramenta e a adotaria o uso para escolha de groupware? | Sim, usaria. | Sim, claro. | Adotaria | Sim, usaria a ferramenta, mas avaliaria a escolha da ferramenta, para o cenário desejado. | Sim |

| Pergunta aberta | Analista 6 | Analista 7 | Analista 8 | Analista 9 | Analista 10 |
|---|--|--|------------|--|--|
| <ul style="list-style-type: none"> Você considera que o sistema de recomendação implementado auxilia de fato a atividade de escolhas de groupware nos processos de levantamento de requisitos? | Sim, pode ajudar na tomada de decisão da melhor ferramenta colaborativa para fazer o trabalho. | Não estas atividades dependem do analista de requisitos. | | Sem dúvida que o sistema de recomendação o auxilia o processo de levantamento de requisitos, pois tem um papel importante na organização e escolha de ferramentas que agilizam as etapas de desenvolvimento de um projeto, orientando as equipes na direção do objetivo determinado. | Sim, a ferramenta auxilia bastante o analista de requisitos no processo de escolha que atende ao processo. |

| | | | | | |
|--|---|-------------|--|---|---|
| <ul style="list-style-type: none"> O que você considera que poderia ser melhorado ou incluído na ferramenta para um maior suporte das escolhas de groupware para processos de levantamento de requisitos? | <p>Bom, o sistema poderia levar em consideração o ambiente do usuário por exemplo se é parte de uma grande empresa. Também poderia aceitar sugestões dos usuários quanto ao tipo de ferramenta escolhida.</p> | <p>Não.</p> | | | <p>Possibilitar o analista modelar as atividades ajuda no hora de definir quais as ferrament as de groupware o sistema recomend aria.</p> |
| <ul style="list-style-type: none"> De uma forma geral, você avaliaria positivamente a ferramenta e a adotaria o uso para escolha de groupware? | <p>Avalio positivamente e adotaria pois poderia ajudar bastante na escolha correta da ferramenta na análise de requisitos.</p> | | | <p>Adotaria com certeza nas minhas metodologias de trabalho</p> | <p>Sim, mas avaliaria a escolha da ferrament a, para o cenário desejado.</p> |

| Pergunta aberta | Analista 11 | Analista 12 |
|--|-------------|---|
| <ul style="list-style-type: none"> • Você considera que o sistema de recomendação implementado auxilia de fato a atividade de escolhas de groupware nos processos de levantamento de requisitos? | Sim | Sim. Acredito que a ferramenta permite filtrar a melhor opção de groupware, evitando impasses ao longo do processo de desenvolvimento das atividades. |
| <ul style="list-style-type: none"> • O que você considera que poderia ser melhorado ou incluído na ferramenta para um maior suporte das escolhas de groupware para processos de levantamento de requisitos? | Enquete | Acho que um adicional seriam instruções sobre a utilização da ferramenta de groupware. |

| | | |
|---|--|--|
| <ul style="list-style-type: none"> De uma forma geral, você avaliaria positivamente a ferramenta e a adotaria o uso para escolha de groupware? | <p>Sim, desde que ela já direcionass e para qual o tipo mais adequado, que acredito ser o produto final deste estudo. Vários pontos acredito que devam usar as ferramentas síncronas porque é em uma conversa que conseguimos identificar pontos que estão implícitos.</p> | <p>Sim. Com certeza adotaria a ferramenta.</p> |
|---|--|--|

Anexo IV – Esquema Relacional

```
CREATE TABLE `Advantage` (  
  `id` int(11) NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `AtomicAction` (  
  `id` int(11) NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `Belief` (  
  `id` int(11) NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `BusinessRule` (  
  `id` int(11) NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `ChosenAlternative` (  
  `id` int(11) NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `CollaborativeRole` (  
  `id` int(11) NOT NULL,
```

```
PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `CollaborativeSession` (  
  `id` int(11) NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `ColletiveSocialAgent` (  
  `SocialAgent_id` int(11) NOT NULL,  
  PRIMARY KEY (`SocialAgent_id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `ComplexAction` (  
  `id` int(11) NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `Criterion` (  
  `id` int(11) NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `Desire` (  
  `id` int(11) NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `Disadvantage` (  
  `id` int(11) NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `DiscardedAlternative` (  
  `id` int(11) NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `Evidence` (  
  `id` int(11) NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `ExternalAgent` (  
  `id` int(11) NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `Fact` (  
  `id` int(11) NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `Feeling` (  
  `id` int(11) NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `IdiomaticLanguage` (  
  `id` int(11) NOT NULL,  
  `languageType` varchar(45) DEFAULT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `InformalExchangeContingency` (  
  `Contingency_id` int(11) DEFAULT NULL,
```

```
    `InformalExchange_id` int(11) NOT NULL,  
    PRIMARY KEY (`InformalExchange_id`),  
    CONSTRAINT `fk_InformalExchangeContigency_InformalExchange1` FOREIGN  
KEY (`InformalExchange_id`) REFERENCES `InformalExchange` (`id`) ON DELETE NO  
ACTION ON UPDATE NO ACTION
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `Intention` (  
    `id` int(11) NOT NULL,  
    PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
    `id` int(11) NOT NULL,  
    PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
    `id` int(11) NOT NULL,  
    PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `Interaction` (  
    `id` int(11) NOT NULL,  
    PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
    `id` int(11) NOT NULL,  
    PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
    `id` int(11) NOT NULL,  
    PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `KnowldgeStructure` (  
    `id` int(11) NOT NULL,  
    PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
    `id` int(11) NOT NULL,  
    PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
    `id` int(11) NOT NULL,  
    PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `KnowledgeStructure` (  
    `id` int(11) NOT NULL,  
    PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
    `id` int(11) NOT NULL,  
    PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
    `id` int(11) NOT NULL,  
    PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `MaterialContribution` (  
    `ActionContribution_id` int(11) NOT NULL,  
    PRIMARY KEY (`ActionContribution_id`),
```

```
    `id` int(11) NOT NULL,  
    PRIMARY KEY (`id`),
```

```
    `id` int(11) NOT NULL,  
    PRIMARY KEY (`id`),
```

```
CONSTRAINT `fk_MaterialContribution_ActionContribution1` FOREIGN KEY  
(`ActionContribution_id`) REFERENCES `ActionContribution` (`id`) ON DELETE NO  
ACTION ON UPDATE NO ACTION
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `Organization` (  
  `SocialAgent_id` int(11) NOT NULL,  
  PRIMARY KEY (`SocialAgent_id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `ProcessGoal` (  
  `id` int(11) NOT NULL,  
  `name` varchar(45) DEFAULT NULL,  
  `description` varchar(255) DEFAULT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `Protocol` (  
  `id` int(11) NOT NULL,  
  `description` varchar(45) DEFAULT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `QuestionSocialization` (  
  `Question_id` int(11) NOT NULL,  
  `Socialization_id` int(11) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `ResourceParticipation` (  
  `Participation_id` int(11) NOT NULL,  
  `participationType` varchar(200) DEFAULT NULL,
```

```

`Resource_id` int(11) DEFAULT NULL,
PRIMARY KEY (`Participation_id`),
KEY `fk_ResourceParticipation_Resource1` (`Resource_id`),
CONSTRAINT `fk_ResourceParticipation_Participation1` FOREIGN KEY
(`Participation_id`) REFERENCES `Participation` (`id`) ON DELETE NO ACTION ON
UPDATE NO ACTION,
CONSTRAINT `fk_ResourceParticipation_Resource1` FOREIGN KEY
(`Resource_id`) REFERENCES `Resource` (`id`) ON DELETE NO ACTION ON UPDATE
NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `Restriction` (
  `id` int(11) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `Risk` (
  `id` int(11) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `Site` (
  `id` int(11) NOT NULL,
  `realSite` tinyint(1) DEFAULT NULL,
  `kindOfSite` varchar(45) DEFAULT NULL,
  `Socialization_id` int(11) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `Situation` (
  `id` int(11) NOT NULL,

```

```
PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `SocialAgent` (  
  `Agent_id` int(11) NOT NULL,  
  PRIMARY KEY (`Agent_id`),  
  CONSTRAINT `fk_SocialAgent_Agent1` FOREIGN KEY (`Agent_id`)  
REFERENCES `Agent` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `SocialRole` (  
  `id` int(11) NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `TimerIntervalRelation` (  
  `id` int(11) NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `action` (  
  `id` int(11) NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `actioncontribution` (  
  `id` int(11) NOT NULL,  
  `Agent_id` int(11) DEFAULT NULL,  
  `CollaborativeRole_id` int(11) DEFAULT NULL,  
  PRIMARY KEY (`id`),
```



```

KEY `fk_ActionContribution_Agent1_idx` (`Agent_id`),
KEY `fk_ActionContribution_CollaborativeRole1_idx` (`CollaborativeRole_id`),
CONSTRAINT `fk_ActionContribution_Agent1` FOREIGN KEY (`Agent_id`)
REFERENCES `Agent` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION,
CONSTRAINT `fk_ActionContribution_CollaborativeRole1` FOREIGN KEY
(`CollaborativeRole_id`) REFERENCES `CollaborativeRole` (`id`) ON DELETE NO
ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `actioncontribution_has_protocol` (
`ActionContribution_id` int(11) NOT NULL,
`Protocol_id` int(11) NOT NULL,
PRIMARY KEY (`ActionContribution_id`,`Protocol_id`),
KEY `fk_ActionContribution_has_Protocol_Protocol1_idx` (`Protocol_id`),
KEY
`fk_ActionContribution_has_Protocol_ActionContribution1_idx`
(`ActionContribution_id`),
CONSTRAINT `fk_ActionContribution_has_Protocol_ActionContribution1`
FOREIGN KEY (`ActionContribution_id`) REFERENCES `ActionContribution` (`id`) ON
DELETE NO ACTION ON UPDATE NO ACTION,
CONSTRAINT `fk_ActionContribution_has_Protocol_Protocol1` FOREIGN KEY
(`Protocol_id`) REFERENCES `Protocol` (`id`) ON DELETE NO ACTION ON UPDATE
NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `activity` (
`id` int(11) NOT NULL,
`name` varchar(255) DEFAULT NULL,
`KnowledgeIntensiveProcessId` int(11) NOT NULL,
`AgentId` int(11) NOT NULL,
`KnowledgeIntensive` tinyint(1) DEFAULT NULL,
`MessageFlow_id` int(11) DEFAULT NULL,
PRIMARY KEY (`id`),

```

```

        KEY                `fk_Activity_KnowledgeIntensiveProcess1_idx`
(KnowledgeIntensiveProcessId`),
        KEY `fk_Activity_Agent1_idx` (`AgentId`),
        KEY `fk_Activity_MessageFlow1_idx` (`MessageFlow_id`),
        CONSTRAINT `fk_Activity_Agent1` FOREIGN KEY (`AgentId`) REFERENCES
`Agent` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION,
        CONSTRAINT `fk_Activity_KnowledgeIntensiveProcess1` FOREIGN KEY
(KnowledgeIntensiveProcessId`) REFERENCES `KnowledgeIntensiveProcess` (`id`) ON
DELETE NO ACTION ON UPDATE NO ACTION,
        CONSTRAINT `fk_Activity_MessageFlow1` FOREIGN KEY (`MessageFlow_id`)
REFERENCES `MessageFlow` (`Flow_id`) ON DELETE NO ACTION ON UPDATE NO
ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `activitygoal` (
    `id` int(11) NOT NULL,
    `KnowledgeIntensiveActivity_id` int(11) NOT NULL,
    `KnowledgeIntensiveActivity_Agent_id` int(11) NOT NULL,
    PRIMARY KEY (`id`,`KnowledgeIntensiveActivity_id`,`KnowledgeIntensiveActivity_Agent_id`),
    KEY `fk_ActivityGoal_KnowledgeIntensiveActivity1`
(KnowledgeIntensiveActivity_Agent_id`),
    KEY `fk_ActivityGoal_KnowledgeIntensiveActivity1_idx`
(KnowledgeIntensiveActivity_id`,`KnowledgeIntensiveActivity_Agent_id`),
    CONSTRAINT `fk_ActivityGoal_KnowledgeIntensiveActivity1` FOREIGN KEY
(KnowledgeIntensiveActivity_Agent_id`) REFERENCES `KnowledgeIntensiveActivity`
(Agent_id`) ON DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `agent` (
    `id` int(11) NOT NULL,
    `Resource_id` int(11) DEFAULT NULL,
    `name` varchar(255) DEFAULT NULL,

```

```

`SocialRole_id` int(11) DEFAULT NULL,
PRIMARY KEY (`id`),
KEY `fk_Agent_Resource1_idx` (`Resource_id`),
KEY `fk_Agent_SocialRole1_idx` (`SocialRole_id`),
CONSTRAINT `fk_Agent_Resource1` FOREIGN KEY (`Resource_id`)
REFERENCES `Resource` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION,
CONSTRAINT `fk_Agent_SocialRole1` FOREIGN KEY (`SocialRole_id`)
REFERENCES `SocialRole` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `alternative` (
`id` int(11) NOT NULL,
`description` varchar(45) DEFAULT NULL,
`decision_id` int(11) NOT NULL,
PRIMARY KEY (`id`),
KEY `fk_alternative_decision1` (`decision_id`),
CONSTRAINT `fk_alternative_decision1` FOREIGN KEY (`decision_id`)
REFERENCES `decision` (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `alternative_has_speciality` (
`Alternative_id` int(11) NOT NULL,
`Speciality_id` int(11) NOT NULL,
PRIMARY KEY (`Alternative_id`,`Speciality_id`),
KEY `fk_Alternative_has_Speciality_Speciality1_idx` (`Speciality_id`),
KEY `fk_Alternative_has_Speciality_Alternative1_idx` (`Alternative_id`),
CONSTRAINT `fk_Alternative_has_Speciality_Alternative1` FOREIGN KEY
(`Alternative_id`) REFERENCES `Alternative` (`id`) ON DELETE NO ACTION ON
UPDATE NO ACTION,
CONSTRAINT `fk_Alternative_has_Speciality_Speciality1` FOREIGN KEY
(`Speciality_id`) REFERENCES `Speciality` (`id`) ON DELETE NO ACTION ON UPDATE
NO ACTION

```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `assertion` (  
  `id` int(11) NOT NULL,  
  `KnowldgeStructure_id` int(11) DEFAULT NULL,  
  PRIMARY KEY (`id`),  
  KEY `fk_Assertion_KnowldgeStructure1_idx` (`KnowldgeStructure_id`),  
  CONSTRAINT `fk_Assertion_KnowldgeStructure1` FOREIGN KEY  
  (`KnowldgeStructure_id`) REFERENCES `KnowldgeStructure` (`id`) ON DELETE NO  
  ACTION ON UPDATE NO ACTION
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `association` (  
  `id` int(11) NOT NULL,  
  `Activity_id` int(11) NOT NULL,  
  `DataObject_id` int(11) NOT NULL,  
  PRIMARY KEY (`id`),  
  KEY `fk_Association_Activity1_idx` (`Activity_id`),  
  KEY `fk_Association_DataObject1_idx` (`DataObject_id`),  
  CONSTRAINT `fk_Association_Activity1` FOREIGN KEY (`Activity_id`)  
  REFERENCES `Activity` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION,  
  CONSTRAINT `fk_Association_DataObject1` FOREIGN KEY (`DataObject_id`)  
  REFERENCES `DataObject` (`Resource_id`) ON DELETE NO ACTION ON UPDATE NO  
  ACTION
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `comunication` (  
  `id` int(11) NOT NULL,  
  `Message_id` int(11) DEFAULT NULL,  
  `Sender_Agent_id` int(11) NOT NULL,  
  `ComunicativeInteraction_id` int(11) NOT NULL,  
  PRIMARY KEY (`id`),  
  KEY `fk_Comunication_Message1_idx` (`Message_id`),
```

```

        KEY `fk_Communication_Sender1_idx` (`Sender_Agent_id`),
        KEY
            `fk_Communication_CommunicativeInteraction1_idx`
(`CommunicativeInteraction_id`),
        CONSTRAINT `fk_Communication_CommunicativeInteraction1` FOREIGN KEY
(`CommunicativeInteraction_id`) REFERENCES `CommunicativeInteraction`
(`ComplexAction_id`) ON DELETE NO ACTION ON UPDATE NO ACTION,
        CONSTRAINT `fk_Communication_Message1` FOREIGN KEY (`Message_id`)
REFERENCES `Message` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION,
        CONSTRAINT `fk_Communication_Sender1` FOREIGN KEY (`Sender_Agent_id`)
REFERENCES `Sender` (`Agent_id`) ON DELETE NO ACTION ON UPDATE NO
ACTION
    ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `communicativeinteraction` (
    `ComplexAction_id` int(11) NOT NULL,
    PRIMARY KEY (`ComplexAction_id`),
    KEY `fk_CommunicativeInteraction_ComplexAction1_idx` (`ComplexAction_id`),
    CONSTRAINT `fk_CommunicativeInteraction_ComplexAction1` FOREIGN KEY
(`ComplexAction_id`) REFERENCES `ComplexAction` (`id`) ON DELETE NO ACTION
ON UPDATE NO ACTION
    ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `contingency` (
    `Event_id` int(11) NOT NULL,
    PRIMARY KEY (`Event_id`),
    KEY `fk_Contingency_Event1_idx` (`Event_id`),
    CONSTRAINT `fk_Contingency_Event1` FOREIGN KEY (`Event_id`)
REFERENCES `Event` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION
    ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `dataobject` (
    `Resource_id` int(11) NOT NULL,

```

```

`KnowledgeStructure_id` int(11) NOT NULL,
PRIMARY KEY (`Resource_id`,`KnowledgeStructure_id`),
KEY `fk_DataObject_Resource1_idx` (`Resource_id`),
KEY `fk_DataObject_KnowledgeStructure1_idx` (`KnowledgeStructure_id`),
CONSTRAINT `fk_DataObject_KnowledgeStructure1` FOREIGN KEY
(`KnowledgeStructure_id`) REFERENCES `KnowledgeStructure` (`id`) ON DELETE NO
ACTION ON UPDATE NO ACTION,
CONSTRAINT `fk_DataObject_Resource1` FOREIGN KEY (`Resource_id`)
REFERENCES `Resource` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `decision` (
`id` int(11) NOT NULL,
`decision` text,
`ActivityId` int(11) DEFAULT NULL,
`Experience_id` int(11) DEFAULT NULL,
`KIA_Activity_id` int(11) NOT NULL,
`KIA_AtomicAction_id` int(11) NOT NULL,
PRIMARY KEY (`id`),
KEY `fk_Decision_Activity1_idx` (`ActivityId`),
KEY `fk_Decision_Experience1_idx` (`Experience_id`),
KEY `fk_Decision_KnowledgeIntensiveActivity1_idx`
(`KIA_Activity_id`,`KIA_AtomicAction_id`),
CONSTRAINT `fk_Decision_Activity1` FOREIGN KEY (`ActivityId`)
REFERENCES `Activity` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION,
CONSTRAINT `fk_Decision_Experience1` FOREIGN KEY (`Experience_id`)
REFERENCES `Experience` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION,
CONSTRAINT `fk_Decision_KnowledgeIntensiveActivity1` FOREIGN KEY
(`KIA_Activity_id`,`KIA_AtomicAction_id`) REFERENCES
`KnowledgeIntensiveActivity` (`Activity_id`,`AtomicAction_id`) ON DELETE NO
ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `event` (
  `id` int(11) NOT NULL,
  `TimeInterval_id` int(11) DEFAULT NULL,
  `pre-state_Situation` int(11) DEFAULT NULL,
  `pos-state_Situation` int(11) DEFAULT NULL,
  `synchronized_Event` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `fk_Event_TimeInterval1_idx` (`TimeInterval_id`),
  KEY `fk_Event_Situation1_idx` (`pre-state_Situation` ____),
  KEY `fk_Event_Situation2_idx` (`pos-state_Situation`),
  KEY `fk_Event_Event1_idx` (`synchronized_Event`),
  CONSTRAINT `fk_Event_Event1` FOREIGN KEY (`synchronized_Event`)
REFERENCES `Event` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION,
  CONSTRAINT `fk_Event_Situation1` FOREIGN KEY (`pre-state_Situation`)
REFERENCES `Situation` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION,
  CONSTRAINT `fk_Event_Situation2` FOREIGN KEY (`pos-state_Situation`)
REFERENCES `Situation` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION,
  CONSTRAINT `fk_Event_TimeInterval1` FOREIGN KEY (`TimeInterval_id`)
REFERENCES `TimeInterval` (`id`) ON DELETE NO ACTION ON UPDATE NO
ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `experience` (
  `id` int(11) NOT NULL,
  `Agent_id` int(11) NOT NULL,
  PRIMARY KEY (`id`,`Agent_id`),
  KEY `fk_Experience_Agent1_idx` (`Agent_id`),
  CONSTRAINT `fk_Experience_Agent1` FOREIGN KEY (`Agent_id`)
REFERENCES `Agent` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `externalagent_particip_socialization` (
  `ExternalAgent_id` int(11) NOT NULL,

```

```

        `Socialization_CommunicativeInteraction_id` int(11) NOT NULL,
        `Socialization_CollaborativeSession_id` int(11) NOT NULL,
        PRIMARY KEY KEY
        (`ExternalAgent_id`,`Socialization_CommunicativeInteraction_id`,`Socialization_Collaborative
        Session_id`),
        KEY `fk_ExternalAgent_has_Socialization_Socialization1_idx`
        (`Socialization_CommunicativeInteraction_id`,`Socialization_CollaborativeSession_id`),
        KEY `fk_ExternalAgent_has_Socialization_ExternalAgent1_idx`
        (`ExternalAgent_id`),
        CONSTRAINT `fk_ExternalAgent_has_Socialization_ExternalAgent1` FOREIGN
        KEY (`ExternalAgent_id`) REFERENCES `ExternalAgent` (`id`) ON DELETE NO
        ACTION ON UPDATE NO ACTION,
        CONSTRAINT `fk_ExternalAgent_has_Socialization_Socialization1` FOREIGN
        KEY (`Socialization_CommunicativeInteraction_id`,`Socialization_CollaborativeSession_id`)
        REFERENCES `Socialization` (`CommunicativeInteraction_id`,`CollaborativeSession_id`)
        ON DELETE NO ACTION ON UPDATE NO ACTION
    ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `flow` (
    `id` int(11) NOT NULL,
    `Association_id` int(11) NOT NULL,
    PRIMARY KEY (`id`),
    KEY `fk_Flow_Association1_idx` (`Association_id`),
    CONSTRAINT `fk_Flow_Association1` FOREIGN KEY (`Association_id`)
    REFERENCES `Association` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `impactagent` (
    `id` int(11) NOT NULL,
    `Agent_id` int(11) NOT NULL,
    `Ag_Resource_id` int(11) DEFAULT NULL,
    PRIMARY KEY (`id`,`Agent_id`),
    KEY `fk_ImpactAgent_Agent1_idx` (`Agent_id`),

```



```

        CONSTRAINT `fk_ImpactAgent_Agent1` FOREIGN KEY (`Agent_id`)
REFERENCES `Agent` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION
    ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `informalexchange` (
    `id` int(11) NOT NULL DEFAULT '0',
    `Socialization_id` int(11) NOT NULL DEFAULT '0',
    `Interaction_id` int(11) NOT NULL,
    `MessageFlow_id` int(11) NOT NULL,
    PRIMARY KEY (`id`,`Socialization_id`,`MessageFlow_id`),
    KEY `fk_InformalExchange_Interaction1_idx` (`Interaction_id`),
    CONSTRAINT `fk_InformalExchange_Interaction1` FOREIGN KEY
(`Interaction_id`) REFERENCES `Interaction` (`id`) ON DELETE NO ACTION ON
UPDATE NO ACTION
    ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `innovationagent` (
    `id` int(11) NOT NULL,
    `Agent_id` int(11) NOT NULL,
    PRIMARY KEY (`id`),
    KEY `fk_InnovationAgent_Agent1_idx` (`Agent_id`),
    CONSTRAINT `fk_InnovationAgent_Agent1` FOREIGN KEY (`Agent_id`)
REFERENCES `Agent` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION
    ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `inovation` (
    `id` int(11) NOT NULL,
    `KnowledgeIntensiveProcess_id` int(11) NOT NULL,
    `InovationAgent_id` int(11) NOT NULL,
    PRIMARY KEY (`id`),
    KEY `fk_Inovation_KnowledgeIntensiveProcess1_idx`
(`KnowledgeIntensiveProcess_id`),
    KEY `fk_Inovation_InovationAgent1_idx` (`InovationAgent_id`),

```

```
CONSTRAINT `fk_Inovation_InovationAgent1` FOREIGN KEY
(`InovationAgent_id`) REFERENCES `InovationAgent` (`id`) ON DELETE NO ACTION
ON UPDATE NO ACTION,
```

```
CONSTRAINT `fk_Inovation_KnowledgeIntensiveProcess1` FOREIGN KEY
(`KnowledgeIntensiveProcess_id`) REFERENCES `KnowledgeIntensiveProcess` (`id`) ON
DELETE NO ACTION ON UPDATE NO ACTION
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `inovationagent` (
  `id` int(11) NOT NULL,
  `Agent_id` int(11) NOT NULL,
  `Ag_Resource_id` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`,`Agent_id`),
  KEY `fk_InovationAgent_Agent1_idx` (`Agent_id`),
  CONSTRAINT `fk_InovationAgent_Agent1` FOREIGN KEY (`Agent_id`)
REFERENCES `Agent` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `kipgoal` (
  `KnowledgeIntensiveProcess_id` int(11) NOT NULL,
  `KipProcessGoal_ProcessGoal_id` int(11) NOT NULL,
  KEY `fk_KIPGoal_KnowledgeIntensiveProcess1_idx`
(`KnowledgeIntensiveProcess_id`),
  KEY `fk_KIPGoal_KipProcessGoal1_idx` (`KipProcessGoal_ProcessGoal_id`),
  CONSTRAINT `fk_KIPGoal_KipProcessGoal1` FOREIGN KEY
(`KipProcessGoal_ProcessGoal_id`) REFERENCES `KipProcessGoal` (`ProcessGoal_id`)
ON DELETE NO ACTION ON UPDATE NO ACTION,
  CONSTRAINT `fk_KIPGoal_KnowledgeIntensiveProcess1` FOREIGN KEY
(`KnowledgeIntensiveProcess_id`) REFERENCES `KnowledgeIntensiveProcess` (`id`) ON
DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```

CREATE TABLE `kipprocessgoal` (
  `ProcessGoal_id` int(11) NOT NULL,
  PRIMARY KEY (`ProcessGoal_id`),
  KEY `fk_KipProcessGoal_ProcessGoal1_idx` (`ProcessGoal_id`),
  CONSTRAINT `fk_KipProcessGoal_ProcessGoal1` FOREIGN KEY
(`ProcessGoal_id`) REFERENCES `ProcessGoal` (`id`) ON DELETE NO ACTION ON
UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `knowledgeintensiveactivity` (
  `Activity_id` int(11) NOT NULL,
  `AtomicaAction_id` int(11) NOT NULL,
  `Agent_id` int(11) NOT NULL,
  `InovationAgent_id` int(11) DEFAULT NULL,
  PRIMARY KEY (`Activity_id`,`AtomicaAction_id`),
  KEY `fk_KnowledgeIntensiveActivity_Agent1_idx` (`Agent_id`),
  KEY `fk_KnowledgeIntensiveActivity_InovationAgent1_idx` (`InovationAgent_id`),
  CONSTRAINT `fk_KnowledgeIntensiveActivity_Activity1` FOREIGN KEY
(`Activity_id`) REFERENCES `Activity` (`id`) ON DELETE NO ACTION ON UPDATE
NO ACTION,
  CONSTRAINT `fk_KnowledgeIntensiveActivity_Agent1` FOREIGN KEY
(`Agent_id`) REFERENCES `Agent` (`id`) ON DELETE NO ACTION ON UPDATE NO
ACTION,
  CONSTRAINT `fk_KnowledgeIntensiveActivity_InovationAgent1` FOREIGN KEY
(`InovationAgent_id`) REFERENCES `InovationAgent` (`id`) ON DELETE NO ACTION
ON UPDATE NO ACTION

```

```

CREATE TABLE `knowledgeintensiveprocess` (
  `id` int(11) NOT NULL,
  `ComplexAction_id` int(11) NOT NULL,
  `process` varchar(45) DEFAULT NULL,
  PRIMARY KEY (`id`,`ComplexAction_id`),

```

```

        KEY `fk_KnowledgeIntensiveProcess_ComplexAction1_idx` (`ComplexAction_id`),
        CONSTRAINT `fk_KnowledgeIntensiveProcess_ComplexAction1` FOREIGN KEY
(`ComplexAction_id`) REFERENCES `ComplexAction` (`id`) ON DELETE NO ACTION
ON UPDATE NO ACTION
    ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `mentalimage` (
    `id` int(11) NOT NULL,
    `information` varchar(1000) DEFAULT NULL,
    `topic` varchar(45) DEFAULT NULL,
    `KnowldgeStructure_id` int(11) DEFAULT NULL,
    PRIMARY KEY (`id`),
    KEY `fk_MentalImage_KnowldgeStructure1_idx` (`KnowldgeStructure_id`),
    CONSTRAINT `fk_MentalImage_KnowldgeStructure1` FOREIGN KEY
(`KnowldgeStructure_id`) REFERENCES `KnowldgeStructure` (`id`) ON DELETE NO
ACTION ON UPDATE NO ACTION
    ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `mentalimage_has_assertion` (
    `MentalImage_id` int(11) NOT NULL,
    `Assertion_id` int(11) NOT NULL DEFAULT '0',
    PRIMARY KEY (`MentalImage_id`,`Assertion_id`),
    KEY `fk_MentalImage_has_Assertion_Assertion1_idx` (`Assertion_id`),
    KEY `fk_MentalImage_has_Assertion_MentalImage1_idx` (`MentalImage_id`),
    CONSTRAINT `fk_MentalImage_has_Assertion_Assertion1` FOREIGN KEY
(`Assertion_id`) REFERENCES `Assertion` (`id`) ON DELETE NO ACTION ON UPDATE
NO ACTION,
    CONSTRAINT `fk_MentalImage_has_Assertion_MentalImage1` FOREIGN KEY
(`MentalImage_id`) REFERENCES `MentalImage` (`id`) ON DELETE NO ACTION ON
UPDATE NO ACTION
    ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `message` (

```

```

`id` int(11) NOT NULL,
`content` varchar(4000) DEFAULT NULL,
`IdiomaticLanguage_id` int(11) DEFAULT NULL,
PRIMARY KEY (`id`),
KEY `fk_Message_IdiomaticLanguage1_idx` (`IdiomaticLanguage_id`),
CONSTRAINT `fk_Message_IdiomaticLanguage1` FOREIGN KEY
(`IdiomaticLanguage_id`) REFERENCES `IdiomaticLanguage` (`id`) ON DELETE NO
ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `messageflow` (
`Flow_id` int(11) NOT NULL,
PRIMARY KEY (`Flow_id`),
KEY `fk_MessageFlow_Flow1_idx` (`Flow_id`),
CONSTRAINT `fk_MessageFlow_Flow1` FOREIGN KEY (`Flow_id`)
REFERENCES `Flow` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `participation` (
`id` int(11) NOT NULL,
`ComplexAction_id` int(11) DEFAULT NULL,
`description` varchar(80) DEFAULT NULL,
PRIMARY KEY (`id`),
KEY `fk_Participation_ComplexAction1` (`ComplexAction_id`),
CONSTRAINT `fk_Participation_ComplexAction1` FOREIGN KEY
(`ComplexAction_id`) REFERENCES `ComplexAction` (`id`) ON DELETE NO ACTION
ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `perception` (
`ActionContribution_id` int(11) NOT NULL,

```

```

`id` int(11) DEFAULT NULL,
`Message_id` int(11) DEFAULT NULL,
`Message_Socialization_id` int(11) DEFAULT NULL,
`Receiver_id` int(11) DEFAULT NULL,
`CommunicativeInteraction_id` int(11) NOT NULL,
PRIMARY KEY (`ActionContribution_id`),
KEY `fk_Perception_Message1_idx` (`Message_id`,`Message_Socialization_id`),
KEY `fk_Perception_CommunicativeInteraction1_idx`
(`CommunicativeInteraction_id`),
CONSTRAINT `fk_Perception_ActionContribution1` FOREIGN KEY
(`ActionContribution_id`) REFERENCES `ActionContribution` (`id`) ON DELETE NO
ACTION ON UPDATE NO ACTION,
CONSTRAINT `fk_Perception_CommunicativeInteraction1` FOREIGN KEY
(`CommunicativeInteraction_id`) REFERENCES `CommunicativeInteraction`
(`ComplexAction_id`) ON DELETE NO ACTION ON UPDATE NO ACTION,
CONSTRAINT `fk_Perception_Message1` FOREIGN KEY (`Message_id`)
REFERENCES `Message` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `question` (
`Contingency_id` int(11) NOT NULL,
`description` varchar(4000) DEFAULT NULL,
`type` varchar(45) DEFAULT NULL,
`ImpactAgent_id` int(11) DEFAULT NULL,
PRIMARY KEY (`Contingency_id`),
KEY `fk_Question_ImpactAgent1_idx` (`ImpactAgent_id`),
KEY `fk_Question_Contingency1_idx` (`Contingency_id`),
CONSTRAINT `fk_Question_Contingency1` FOREIGN KEY (`Contingency_id`)
REFERENCES `Contingency` (`Event_id`) ON DELETE NO ACTION ON UPDATE NO
ACTION,

```

```
CONSTRAINT `fk_Question_ImpactAgent1` FOREIGN KEY (`ImpactAgent_id`)
REFERENCES `ImpactAgent` (`id`) ON DELETE NO ACTION ON UPDATE NO
ACTION
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `receiver` (
  `Agent_id` int(11) NOT NULL,
  PRIMARY KEY (`Agent_id`),
  KEY `fk_Receiver_Agent1_idx` (`Agent_id`),
  CONSTRAINT `fk_Receiver_Agent1` FOREIGN KEY (`Agent_id`) REFERENCES
`Agent` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `resource` (
  `id` int(11) NOT NULL,
  `resourceType` varchar(45) DEFAULT NULL,
  `type` varchar(80) DEFAULT NULL,
  PRIMARY KEY (`id`)
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `ruleimage` (
  `MentalImage_id` int(11) NOT NULL,
  `BusinessRule_id` int(11) NOT NULL,
  KEY `fk_RuleImage_MentalImage1_idx` (`MentalImage_id`),
  KEY `fk_RuleImage_BusinessRule1_idx` (`BusinessRule_id`),
  CONSTRAINT `fk_RuleImage_BusinessRule1` FOREIGN KEY
(`BusinessRule_id`) REFERENCES `BusinessRule` (`id`) ON DELETE NO ACTION ON
UPDATE NO ACTION,
```

```
CONSTRAINT `fk_RuleImage_MentalImage1` FOREIGN KEY
(`MentalImage_id`) REFERENCES `MentalImage` (`id`) ON DELETE NO ACTION ON
UPDATE NO ACTION
```

```

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `sender` (
  `Agent_id` int(11) NOT NULL,
  PRIMARY KEY (`Agent_id`),
  KEY `fk_Sender_Agent1_idx` (`Agent_id`),
  CONSTRAINT `fk_Sender_Agent1` FOREIGN KEY (`Agent_id`) REFERENCES
`Agent` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `socialization` (
  `ComunicativeInteraction_id` int(11) NOT NULL,
  `CollaborativeSession_id` int(11) NOT NULL,
  `description` varchar(45) DEFAULT NULL,
  `formalExchange` tinyint(1) DEFAULT NULL,
  `knowledgeintensiveprocess_id` int(11) DEFAULT NULL,
  `synchronous` int(1) DEFAULT NULL,
  PRIMARY KEY (`ComunicativeInteraction_id`,`CollaborativeSession_id`),
  KEY `fk_Socialization_CollaborativeSession1_idx` (`CollaborativeSession_id`),
  KEY `fk_Socialization_ComunicativeInteraction1_idx`
(`ComunicativeInteraction_id`),
  KEY `knowledgeintensiveprocess_id` (`knowledgeintensiveprocess_id`),
  CONSTRAINT `fk_Socialization_CollaborativeSession1` FOREIGN KEY
(`CollaborativeSession_id`) REFERENCES `CollaborativeSession` (`id`) ON DELETE NO
ACTION ON UPDATE NO ACTION,
  CONSTRAINT `fk_Socialization_ComunicativeInteraction1` FOREIGN KEY
(`ComunicativeInteraction_id`) REFERENCES `ComunicativeInteraction`
(`ComplexAction_id`) ON DELETE NO ACTION ON UPDATE NO ACTION,
  CONSTRAINT `socialization_ibfk_1` FOREIGN KEY
(`knowledgeintensiveprocess_id`) REFERENCES `knowledgeintensiveprocess` (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `socialization_has_externalagent` (

```



```

`Socialization_id` int(11) NOT NULL,
`ExternalAgent_id` int(11) NOT NULL,
PRIMARY KEY (`Socialization_id`,`ExternalAgent_id`),
KEY `fk_Socialization_has_ExternalAgent_ExternalAgent1_idx`
(`ExternalAgent_id`),
CONSTRAINT `fk_Socialization_has_ExternalAgent_ExternalAgent1` FOREIGN
KEY (`ExternalAgent_id`) REFERENCES `ExternalAgent` (`id`) ON DELETE NO
ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `socialization_has_mentalimage` (
`Socialization_id` int(11) NOT NULL,
`MentalImage_id` int(11) NOT NULL,
PRIMARY KEY (`Socialization_id`,`MentalImage_id`),
KEY `fk_Socialization_has_MentalImage_MentalImage1_idx` (`MentalImage_id`),
CONSTRAINT `fk_Socialization_has_MentalImage_MentalImage1` FOREIGN
KEY (`MentalImage_id`) REFERENCES `MentalImage` (`id`) ON DELETE NO ACTION
ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `socialization_has_protocol` (
`Socialization_id` int(11) NOT NULL,
`Protocol_id` int(11) NOT NULL,
PRIMARY KEY (`Socialization_id`,`Protocol_id`),
KEY `fk_Socialization_has_Protocol_Protocol1_idx` (`Protocol_id`),
CONSTRAINT `fk_Socialization_has_Protocol_Protocol1` FOREIGN KEY
(`Protocol_id`) REFERENCES `Protocol` (`id`) ON DELETE NO ACTION ON UPDATE
NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `socializationatervative` (
`Socialization_id` int(11) NOT NULL,
`Alternative_id` int(11) NOT NULL,

```

```

        KEY `fk_SocializationAternative_Alternative1_idx` (`Alternative_id`),
        CONSTRAINT `fk_SocializationAternative_Alternative1` FOREIGN KEY
(`Alternative_id`) REFERENCES `Alternative` (`id`) ON DELETE NO ACTION ON
UPDATE NO ACTION
    ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `speciality` (
    `id` int(11) NOT NULL,
    `name` varchar(45) DEFAULT NULL,
    `InnovationAgent_id` int(11) DEFAULT NULL,
    `inovationagent_id` int(11) DEFAULT NULL,
    PRIMARY KEY (`id`),
    KEY `fk_Speciality_InnovationAgent1_idx` (`InnovationAgent_id`),
    KEY `inovationagent_id` (`inovationagent_id`),
    CONSTRAINT `fk_Speciality_InnovationAgent1` FOREIGN KEY
(`InnovationAgent_id`) REFERENCES `InnovationAgent` (`id`) ON DELETE NO ACTION
ON UPDATE NO ACTION,
    CONSTRAINT `speciality_ibfk_1` FOREIGN KEY (`inovationagent_id`)
REFERENCES `inovationagent` (`id`)
    ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `timeinterval` (
    `id` int(11) NOT NULL,
    `TimerIntervalRelation_id` int(11) DEFAULT NULL,
    PRIMARY KEY (`id`),
    KEY `fk_TimeInterval_TimerIntervalRelation1_idx` (`TimerIntervalRelation_id`),
    CONSTRAINT `fk_TimeInterval_TimerIntervalRelation1` FOREIGN KEY
(`TimerIntervalRelation_id`) REFERENCES `TimerIntervalRelation` (`id`) ON DELETE NO
ACTION ON UPDATE NO ACTION
    ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `wgwsoa_Service_has_Category` (
    `Service_id` int(11) NOT NULL,

```

```

`Category_id` int(11) NOT NULL,
PRIMARY KEY (`Service_id`,`Category_id`),
KEY `fk_Service_has_Category_Category1` (`Category_id`),
CONSTRAINT `fk_Service_has_Category_Category1` FOREIGN KEY
(`Category_id`) REFERENCES `wgwsoa_Category` (`id`) ON DELETE NO ACTION ON
UPDATE NO ACTION,
CONSTRAINT `fk_Service_has_Category_Service1` FOREIGN KEY (`Service_id`)
REFERENCES `wgwsoa_Service` (`id`) ON DELETE NO ACTION ON UPDATE NO
ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `wgwsoa_category` (
`id` int(11) NOT NULL,
`description` varchar(45) DEFAULT NULL,
`wgwsoa_language_id` int(11) DEFAULT NULL,
`parent` int(11) DEFAULT NULL,
PRIMARY KEY (`id`),
KEY `fk_wgwsoa_Category_wgwsoa_language1` (`wgwsoa_language_id`),
KEY `parent` (`parent`),
CONSTRAINT `fk_wgwsoa_Category_wgwsoa_language1` FOREIGN KEY
(`wgwsoa_language_id`) REFERENCES `wgwsoa_language` (`id`) ON DELETE NO
ACTION ON UPDATE NO ACTION,
CONSTRAINT `wgwsoa_category_ibfk_1` FOREIGN KEY (`parent`)
REFERENCES `wgwsoa_category` (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `wgwsoa_language` (
`id` int(11) NOT NULL,
`description` varchar(45) DEFAULT NULL,
PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `wgwsoa_service` (

```

```
`id` int(11) NOT NULL AUTO_INCREMENT,  
`serviceName` varchar(80) DEFAULT NULL,  
`remiObjectName` int(11) DEFAULT NULL,  
`location` varchar(80) DEFAULT NULL,  
`ip` varchar(45) DEFAULT NULL,  
`port` int(11) DEFAULT NULL,  
PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO_INCREMENT=39 DEFAULT CHARSET=utf8;
```

