



Ingeniare. Revista Chilena de Ingeniería

ISSN: 0718-3291

facing@uta.cl

Universidad de Tarapacá

Chile

Uriona Maldonado, Mauricio; Coser, Adriano
ENGENHARIA DO CONHECIMENTO APLICADA AO SERVIÇO DE ATENDIMENTO AO CLIENTE
NA INDÚSTRIA DO SOFTWARE

Ingeniare. Revista Chilena de Ingeniería, vol. 18, núm. 1, 2010, pp. 53-63
Universidad de Tarapacá
Arica, Chile

Disponível em: <http://www.redalyc.org/articulo.oa?id=77218811007>

- ▶ Como citar este artigo
- ▶ Número completo
- ▶ Mais artigos
- ▶ Home da revista no Redalyc

redalyc.org

Sistema de Informação Científica

Rede de Revistas Científicas da América Latina, Caribe, Espanha e Portugal
Projeto acadêmico sem fins lucrativos desenvolvido no âmbito da iniciativa Acesso Aberto

ENGENHARIA DO CONHECIMENTO APLICADA AO SERVIÇO DE ATENDIMENTO AO CLIENTE NA INDÚSTRIA DO SOFTWARE

KNOWLEDGE ENGINEERING APPLIED TO COSTUMER SERVICE IN THE SOFTWARE INDUSTRY

Mauricio Uriona Maldonado¹ Adriano Coser¹

Recibido 6 de abril de 2009, aceptado 19 de enero de 2010

Received: April 6, 2009 Accepted: January 19, 2010

RESUMO

A representação do conhecimento apoia as técnicas e ferramentas de gestão e deve possuir uma compreensão substancial do negócio em que está inserido. A identificação e análise do negócio onde estão embutidos os aspectos de conhecimento são fundamentais para o sucesso da gestão do conhecimento nas organizações. Este artigo descreve a construção da primeira fase de um sistema de conhecimento para o serviço de atendimento ao usuário de uma software-house no Brasil. A metodologia empregada é a engenharia do conhecimento e mais especificamente os modelos da camada contextual do CommonKADS. Inicialmente é apresentada uma revisão sobre o framework metodológico CommonKADS, com foco nos modelos de organização, tarefa e agente. Em seguida são descritas as etapas de construção do sistema de conhecimento e apresentado o local de aplicação. Conclui com as implicações do sistema de conhecimento para a software-house analizada bem como as recomendações dos próximos passos de aplicação do CommonKADS.

Palavras clave: Atendimento ao usuário de software, engenharia do conhecimento, modelagem CESM, CommonKADS.

ABSTRACT

Knowledge representation supports management tools and techniques and must possess a substantial comprehension of the business in which is inserted. This paper describes the first phase construction of a knowledge system for a Brazilian software-house's customer service. The method applied is knowledge engineering and specifically the context-layer models of CommonKADS. First, a literature review is presented regarding the CommonKADS Methodology, focusing in the Organization, Tasks and Agent Models. Next, the development phases for the knowledge system are described as well as the company where the study was made. It concludes with the business implications of the knowledge system for the software-house as well as the next steps of application for CommonKADS.

Keywords: Software customer service, knowledge engineering, CESM modelling, CommonKADS.

INTRODUÇÃO

Na utilização de sistemas de software é bastante comum o usuário se deparar com problemas que afetam negativamente a sua produtividade. Alguns fatores ajudam a explicar essa situação:

- As interações entre usuário e software são complexas, pois exigem que o usuário possua conhecimentos sobre informática, sobre a operação do próprio software e sobre as atividades automatizadas pelo mesmo, ou seja, sobre o domínio de aplicação.

- A indústria de software não atingiu um grau de maturidade elevado ao ponto de gerar produtos sem falhas. Mesmo quando um software atende à sua especificação por completo e sem falhas, é possível que os requisitos não tenham sido bem elaborados e que, portanto, o software não atenda adequadamente às necessidades do usuário.

Como parte dos seus pacotes de serviço, as software-houses implantam serviços de atendimento ao usuário, com o objetivo de garantir a plena utilização das suas soluções de software.

¹ Programa de Pós Graduação em Engenharia e Gestão do Conhecimento. Universidade Federal de Santa Catarina. Campus UFSC, Bairro Trindade. CEP 88040-970. Florianópolis, Brasil. E-mail: uriona@ieee.org; coser@egc.ufsc.br

A estrutura para suportar esse serviço pode representar um peso importante no fluxo de caixa da organização, porque emprega normalmente pessoal técnico especializado não só em informática como no campo de aplicação dos programas [1]. Otimizar essa estrutura é uma questão relevante para o equilíbrio financeiro da empresa.

Por outro lado, o serviço de suporte se encontra “na ponta” da cadeia de valor das organizações de software, constituindo-se no maior canal de comunicação com os seus usuários. Extrair conhecimento das interações entre os usuários e o serviço de suporte pode ser determinante para se garantir competitividade e longevidade para as soluções produzidas.

Seja pelo lado da otimização de recursos, seja pela necessidade de se extrair conhecimento útil do serviço, os métodos e técnicas da engenharia do conhecimento apresentam alternativas para identificar e, em algum grau, automatizar as tarefas intensivas em conhecimento desempenhadas nos processos de atendimento.

Este artigo lança uma visão sistêmica sobre o serviço de atendimento ao usuário, com objetivo de identificar oportunidades para automatização de tarefas intensivas em conhecimento. O sistema em questão é analisado segundo duas abordagens: o modelo Composição-Ambiente-Estrutura-Mecanismo (CESM) de Bunge [2] e os modelos da camada conceitual do CommonKADS [3].

A abordagem CESM [2] fornece uma visão de alto nível de qualquer tipo de sistema, identificando os componentes, os elementos externos com os quais interagem os componentes (ambiente), a estrutura de ligações entre os componentes e destes com o ambiente, e os mecanismos que determinam o comportamento do sistema.

O framework metodológico CommonKADS nasceu da necessidade de se conceber sistemas de conhecimento com alto grau de qualidade e em grande escala, o que implica em um processo de desenvolvimento estruturado, controlável e repetível [3]. Derivada da metodologia KADS, destinada à aquisição de conhecimento, o CommonKADS traz uma série de invocações que vão de encontro às necessidades da nova engenharia do conhecimento, caracterizada pela visão ampla do domínio de aplicação e intimamente ligada às atividades de gestão [4]. Seus autores seguiram a mudança de paradigma que transformou a engenharia do conhecimento em uma atividade de construção de modelos para diferentes aspectos do conhecimento humano. O framework consiste em seis modelos agrupados nas camadas contextual, conceitual e de artefatos [3].

O serviço de atendimento ao usuário aqui modelado é uma leitura do serviço oferecido pela empresa AltoQi

Tecnologia², mas os modelos concebidos e as oportunidades de automatização identificadas são bastante genéricas, podendo ser adaptadas a outras organizações que desenvolvem e comercializam soluções de software.

A próxima seção descreve brevemente a metodologia CommonKADS, empregada na modelagem dos aspectos organizacionais do sistema de atendimento a usuários de software. A seção III fornece uma visão do sistema orientada pela metodologia CESM. Na seção IV são desenvolvidos os modelos prescritos na camada contextual do CommonKADS. A seção V trás as considerações finais do trabalho.

MODELAGEM DE CONHECIMENTO: O COMMONKADS

A presente seção descreve a metodologia CommonKADS [3], utilizada para guiar a aquisição e a organização do conhecimento sobre o sistema de atendimento a usuários de software. A escolha desta metodologia deve-se principalmente à sua organização em camadas, partindo da análise dos aspectos organizacionais e evoluindo até as especificidades de um modelo implementável. O interesse deste trabalho é, justamente, o conhecimento sobre o sistema no contexto organizacional, identificando oportunidades para a introdução de sistemas baseados em conhecimento que apóiem as tarefas mais especializadas. A camada organizacional do CommonKADS fornece modelos que vão de encontro a esses interesses.

O CommonKADS nasceu da necessidade de se conceber sistemas de conhecimento com alto grau de qualidade e em grande escala, o que implica em um processo de desenvolvimento estruturado, controlável e repetível [3]. Derivada da metodologia KADS, destinada à aquisição de conhecimento, o CommonKADS traz uma série de invocações que vão de encontro às necessidades da nova engenharia do conhecimento, caracterizada pela visão ampla do domínio de aplicação e intimamente ligada às atividades de gestão [4]. Seus autores seguiram a mudança de paradigma que transformou a engenharia do conhecimento em uma atividade de construção de modelos para diferentes aspectos do conhecimento humano.

A proposta básica da metodologia CommonKADS é não atacar de forma direta a ferramenta para solucionar o

² A AltoQi Tecnologia em Informática desenvolve e comercializa softwares para o projeto de edificações prediais. Está sediada em Florianópolis e conta com cerca de 16.000 usuários espalhados pelo Brasil e, em menor proporção, por outros países.

problema, mas sim, estudar as características e objetivos da organização, de forma a moldar um sistema que realmente atente para as necessidades fundamentais a plena gestão do conhecimento.

A Metodologia *CommonKADS* possui um conjunto predefinido de modelos (Figura 1), cada um focando em um aspecto específico, mas com o objetivo de fornecer no conjunto, as respostas para as principais questões envolvidas na construção de sistemas conhecimento (Por que? O que? Como?).

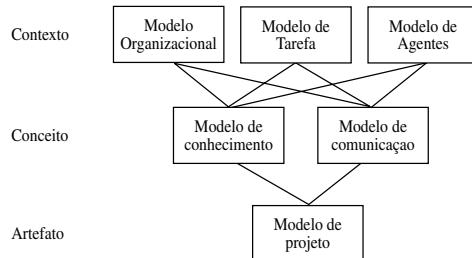


Figura 1. Modelos do CommonKADS [3].

Na Figura 1 apresentam-se os cinco modelos que constituem o *CommonKADS*. O nível contextual é o mais amplo e é onde se definem características da organização, das tarefas e atividades que são executadas na organização bem como os agentes humanos e computacionais que executam essas tarefas.

O seguinte nível é o conceitual, onde se identificam os modelos de conhecimento e comunicação para finalmente entrar no nível do artefato, onde se realiza o modelo de projeto, ou seja do artefacto de conhecimento que será desenvolvido.

A seguir, detalham-se cada um dos modelos do *CommonKADS* [3]:

Modelo da Organização: Suporta a análise das maiores características da organização, com o objetivo de descobrir problemas e oportunidades para sistemas de conhecimento, estabelecer sua viabilidade e medir o impacto das ações de conhecimento pretendidas, para a organização.

Modelo da Tarefa: Tarefas são subdivisões relevantes de um processo de negócio. Este modelo analisa a composição da tarefa global, suas entradas, saídas, pré-condições e critérios de performance, bem como recursos e competências necessárias na sua execução.

Modelo de Agente: Agentes são executores de uma tarefa (humano, sistema de informação ou qualquer entidade

capaz de realizar uma tarefa). Este modelo descreve as características dos agentes (competências, autoridades, restrições de ações, etc.). Além disso, relaciona as ligações de comunicação entre agentes necessárias na execução de uma tarefa. Os modelos da organização, tarefa e agente auxiliam na análise do ambiente organizacional e os fatores críticos ao sucesso de um sistema de conhecimento.

Modelo de Conhecimento: Objetiva explicar em detalhes os tipos e estruturas de conhecimento utilizadas na realização de uma tarefa. Permite uma descrição, independente de implementação, do perfil dos diferentes componentes de conhecimento necessários na resolução de problemas, de forma que sejam compreensíveis por seres humanos. Isto torna o modelo de conhecimento uma importante ferramenta para comunicação com especialistas e usuários sobre os aspectos da resolução do problema de um sistema de conhecimento, tanto durante o desenvolvimento como na execução.

Modelo de Comunicação: Dado que muitos agentes podem estar envolvidos em uma tarefa, é importante modelar a transação de comunicação entre os agentes envolvidos, de forma independente da implementação ou do conceito. Os modelos do conhecimento e de comunicação produzem uma descrição conceitual das funções de resolução de problema e dos dados que são tratados e gerados por um sistema de conhecimento.

Modelo de Projeto. Os modelos anteriores podem ser vistos como constituintes dos requisitos de especificação de um sistema de conhecimento, dividido em diferentes aspectos. Com base nesses requisitos, o modelo de projeto fornece a especificação técnica do sistema em termos de arquitetura, plataforma de implementação, módulos de software, representações e mecanismos computacionais necessários para implementar as funções descritas nos modelos de comunicação e conhecimento.

Para utilizar o *CommonKADS* não é necessário que todos os modelos sejam construídos, tudo depende dos objetivos do projeto e das experiências adquiridas na sua execução.

O CommonKads tem um foco específico na representação de sistemas de conhecimento, para isso, ele trabalha várias fases da modelagem que buscam estabelecer o contexto que o sistema se coloca. Esta metodologia se caracteriza por ser autófágica, uma vez que o estudo do contexto vai permitir a organização definir a real necessidade de um sistema de conhecimento.

Apesar de sua grande quantidade de modelos, e seu detalhamento dos processos da organização, o CommonKads

não é uma ferramenta específica para a gestão do conhecimento. Aspectos que estruturam a gestão são considerados unicamente para possibilitar a construção do sistema de conhecimento, gerando então algumas lacunas que podem ser trabalhadas pela gestão.

Três pontos básicos caracterizam estas demandas; o detalhamento das competências envolvidas na execução dos processos, a representação dos processos através de artefatos semânticos e a análise de processo que busca estruturar as oportunidades de melhoria no que tange o processo e o uso do conhecimento.

O SERVIÇO DE ATENDIMENTO AO USUÁRIO SEGUNDO O MODELO CESM

Precedendo a análise contextual do serviço a partir da metodologia CommonKADS, esta seção fornece uma visão de mais alto nível do sistema, guiada pelo modelo Composição-Ambiente-Estrutura-Mecanismo (CESM) de Bunge [2]. Considera-se esta análise importante, principalmente, pela necessidade de se definir claramente a fronteira que delimita o sistema em análise no contexto da organização. Como este serviço interage intimamente com diversos setores da empresa, sua delimitação não deve ser considerada trivial, e o modelo CESM serve como guia.

A função essencial de um serviço de atendimento ao usuário é receber dúvidas, queixas e sugestões dos clientes e reponde-las de modo a satisfazer as necessidades imediatas dos mesmos. Embora essa descrição sugira um sistema simples e com entradas e saídas bem definidas, uma análise no contexto das organizações de software deve levar em conta alguns fatores importantes:

- O serviço de atendimento a usuários de software deve lidar com um universo potencialmente infinito de situações, devido à complexidade das interações do usuário com o software e deste com o seu ambiente computacional. Tem-se então um ambiente que precisa aprender continuamente. Sem uma estratégia de gestão voltada à aprendizagem organizacional, o conhecimento empregado nos atendimentos sairia da empresa todos os dias (na cabeça dos funcionários).
- Outro ponto importante é a complexidade dos problemas apresentados, que tem implicação direta sobre o nível técnico da equipe de suporte. Ocupar pessoas muito especializadas com problemas simples é muito dispendioso. A situação contrária também é indesejável, pois tende a diminuir a eficiência do sistema.

- Deve-se considerar também a integração o serviço de atendimento com o restante da cadeia produtiva da organização. Nesse aspecto é importante notar que o serviço de suporte caracteriza um grande canal de ligação entre a empresa e seus usuários. Extrair conhecimento desse ambiente aumenta as chances da empresa colocar seus produtos no patamar esperado pelo mercado. Softwares são concebidos para evoluírem e o contato com os usuários gera conhecimentos essenciais para a definição dos rumos a serem seguidos.

Tais fatores dirigem as análises sobre o sistema, tanto na modelagem CESM quanto nos modelos do CommonKADS.

A modelagem CESM, mostrada no Quadro 1, fornece uma visão de alto nível do sistema, que constitui um subsistema da organização de software.

Quadro 1. Aplicação do Modelo CESM para o caso de atendimento ao usuário de software.

Composição	Usuários, equipe de atendimento, ferramentas de apoio ao atendimento.
Ambiente	Os softwares produzidos e comercializados pela software-house. O processo de manutenção e evolução dos softwares. Nível de conhecimento dos usuários (informática, uso dos softwares, domínio de aplicação). Ambiente computacional em que os softwares operam (rede, computador, sistema operacional, etc.).
Estrutura	Interações dos usuários com os softwares, dos usuários com a equipe de atendimento, da equipe de atendimento com as ferramentas de apoio, da equipe de atendimento com o desenvolvimento.
Mecanismo	O mecanismo central é a exposição de problemas por parte dos usuários e a busca da solução pelos atendentes. Memória organizacional onde os casos apresentados são relacionados às soluções adotadas. Essa estrutura fomenta a aprendizagem e reutilização do conhecimento para agilizar o processo de atendimento. Utilização do conhecimento adquirido para alimentar a manutenção e evolução do software.

ASPECTOS CONTEXTUAIS DO SISTEMA SEGUNDO O FRAMEWORK METODOLÓGICO COMMONKADS

Nesta seção o serviço de atendimento ao usuário é analisado segundo os modelos da camada contextual do CommonKADS [3]. Diferente da modelagem CESM, os modelos contextuais do CommonKADS fornecem

uma leitura bem mais detalhada do sistema analisado. O principal resultado é a identificação e detalhamento das tarefas intensivas em conhecimento, levando em conta os agentes e ativos de conhecimento envolvidos.

Neste artigo o modelo da organização contempla características de todo sistema analisado e da organização (super sistema) em que está inserido. Os modelos de tarefa e de agente tratam apenas de uma das tarefas intensivas em conhecimento identificadas no modelo da organização.

O Quadro 2 trás a primeira planilha do modelo da organização. Em “problemas e oportunidades” são identificados claramente os aspectos citados na seção 2 como oportunidades para aplicação de sistemas de conhecimento. No “contexto organizacional” são descritas as relações do sistema analisado com o seu super sistema (software-house).

Quadro 2. Modelo da organização, planilha OM-1: Problemas e Oportunidades.

Problemas e oportunidades	Contexto organizacional	Soluções
<ul style="list-style-type: none"> * Uma organização que produz e comercializa software (software-house) deve procurar garantir que seus produtos sejam utilizados da melhor forma pelos usuários. * Mesmo primando pela qualidade dos produtos e da sua documentação, dependendo do domínio de aplicação, a organização deve lidar com as dificuldades enfrentadas pelos usuários na utilização dos softwares. Essas dificuldades podem advir de projetos de interface mal concebidos, de erros de execução do software, do não atendimento da documentação, do baixo nível de conhecimento do usuário, entre outros. * Um serviço de atendimento ao usuário é uma fórmula comumente adotada pelas organizações visando garantir a boa utilização dos seus produtos. <p>Problemas</p> <ul style="list-style-type: none"> * O serviço de atendimento a usuários de software deve lidar com um universo potencialmente infinito de situações. Tem-se então um ambiente que precisa aprender continuamente. Sem o apoio de sistemas computacionais, o conhecimento empregado nos atendimentos sairia da empresa todos os dias (na cabeça dos funcionários). A aprendizagem contínua e a reutilização do conhecimento são, portanto, essenciais para a efetividade do serviço. => Oportunidade: implementação de memória organizacional visando melhoraria contínua do processo de atendimento. * A complexidade dos problemas apresentados tem implicação direta sobre o nível técnico da equipe de suporte. => Oportunidade: apoio à triagem e alocação de problemas visando otimizar a utilização da equipe de atendimento. * Do serviço de atendimento ao usuário deve emergir conhecimentos importantes para a evolução dos produtos (problemas recorrentes, deficiências, excessos e omissões relatados pelos usuários). => Oportunidade: priorização de problemas a serem encaminhados para a área de desenvolvimento (manutenção). Identificação de requisitos importantes não atendidos pelo software (evolução). 	<p>Missão:</p> <ul style="list-style-type: none"> * De uma software-house: Desenvolver software que atenda, na maior extensão e com a maior qualidade possíveis, os requisitos do domínio de aplicação. * De um serviço de atendimento ao usuário: Promover a plena utilização dos softwares por parte dos clientes. Responder prontamente às dúvidas de utilização dos clientes e encaminhar a resolução dos problemas (erros, deficiências, omissões, etc.) apresentados pelos softwares. <p>Fatores externos à organização:</p> <ul style="list-style-type: none"> * Ambiente computacional (rede, computador, sistema operacional, etc.) onde operam os softwares. * Políticas de tecnologia da informação a que estão sujeitos os clientes (restrições de acesso a sistemas de arquivos, autorização para instalar software, restrições de uso da Internet, por exemplo). * Nível de conhecimento dos usuários (informática, operação dos softwares, domínio da aplicação, etc.). * Pressões de tempo, relações de poder e outros tipos de pressões a que estão sujeitos os usuários e que podem gerar problemas na utilização dos softwares. * Mudanças produzidas na legislação, regras e modelos de negócio, etc. do domínio para o qual o software foi desenvolvido. <p>Estratégia da organização:</p> <ul style="list-style-type: none"> * Garantir que as suas soluções de software sejam plenamente utilizadas e agreguem valor aos negócios que apóiam. * Direcionar os processos de manutenção e evolução de software para as necessidades reais dos clientes, visando estender ao máximo a longevidade dos produtos. 	<ul style="list-style-type: none"> * Implementar um ambiente computacional que suporte a memória organizacional, visando adquirir, armazenar, recuperar e reutilizar o conhecimento sobre os casos tratados no serviço de atendimento. * O sistema poderia apoiar também a triagem e alocação dos problemas apresentados, visando otimizar a utilização dos recursos humanos do serviço de atendimento. Pensar também em fornecer respostas automáticas para casos mais estruturados. * Outra oportunidade seria a identificação, classificação e priorização de requisitos para a manutenção e evolução dos softwares. Das interações com o usuário emerge conhecimento sobre problemas recorrentes, situações de uso não previstas no projeto do software, recursos desejáveis ainda não contemplados, etc. Suprir as deficiências identificadas, com uma priorização adequada, deve garantir maior longevidade aos softwares.

Como complemento, a Figura 2 mostra a cadeia de valor de uma organização de software, com base na nomenclatura de Porter [5], onde o sistema analisado é representado por “atendimento ao usuário”. Em organizações de software, a pesquisa e desenvolvimento (P&D) é essencialmente atividade de apoio, porém, com uma forte influência nas atividades-fim, visualizada na Figura 2, pelo tamanho relativo da atividade de P&D.

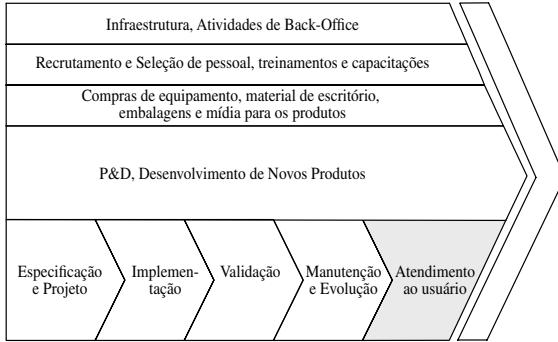


Figura 2. Cadeia de Valor de uma software-house.

As atividades-fim da cadeia de valor do processo de software da Figura 2 foram levantadas a partir das classificações dadas por Schwartz [6], Pressman [7] e Sommerville [8].

A atividade de *Especificação e Projeto* envolve o estabelecimento da solução geral para o problema, o levantamento das necessidades do software a ser implementado, a descrição funcional do sistema, e os projetos arquitetural (modelo conceitual), de interface (comunicação) e detalhado (definição dos módulos).

A atividade de *Implementação* envolve a codificação do sistema *per se* em uma linguagem de computador.

A *Validação* envolve a realização de testes para verificar a presença de erros e comportamento e a integração dos módulos em um produto de software e a verificação da interação entre estes quando operados em conjunto.

A *Manutenção e Evolução* envolvem as atividades iterativas que abrangem todas as fases anteriores.

O *Atendimento ao Usuário*, que se encontra “na ponta” das atividades-fim, envolve as interações com os usuários relacionadas com o esclarecimento de dúvidas e questionamentos.

Seguindo no modelo da organização, as Figura 3 e 4 mostram, respectivamente, a estrutura organizacional da empresa analisada e a dinâmica do seu serviço de atendimento ao usuário por meio do uso de uma ferramenta de representação de processos chamada de Diagrama Quem-Qué ou *Cross-Functional Diagram* [9].

Nas duas figuras, vê-se que o departamento de suporte técnico, que provê o serviço em questão, está intimamente ligado ao desenvolvimento dos softwares. Como já destacado, o serviço de suporte deve fornecer insumos para orientar tanto o desenvolvimento de novos produtos quanto as atividades de manutenção e evolução dos produtos existentes.

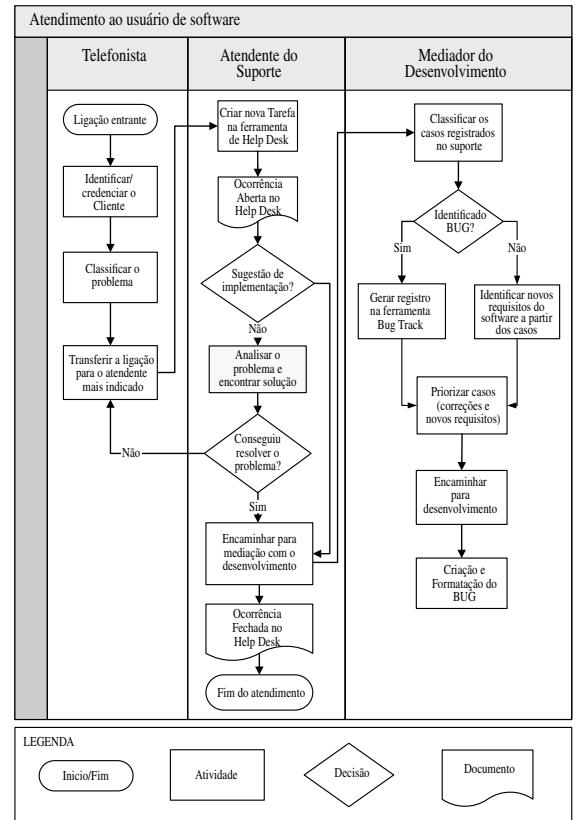


Figura 3. Diagrama Quém-Qué descrevendo o processo de atendimento ao usuário.

O Quadro 3 destaca os aspectos de mudança criados pela solução proposta. Essa planilha destaca um ponto importante: a disposição das pessoas envolvidas no processo de atendimento em compartilhar informações e conhecimentos é fator essencial para o sucesso das soluções de gestão do conhecimento que se deseja implantar. O conhecimento é gerado, efetivamente, na mente das pessoas e por elas o conhecimento deve ser explicitado para que possa incorporada por outras pessoas.

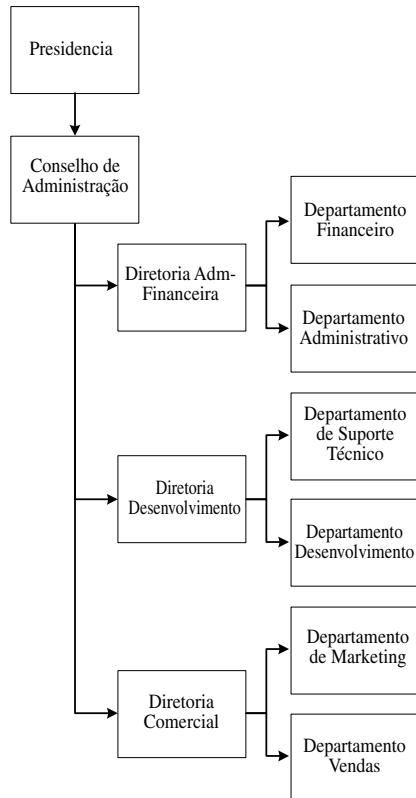


Figura 4. Estrutura organizacional da empresa analisada.

Quadro 3. Modelo da organização, planilha OM-2: Aspectos de mudança criados pela solução proposta.

Recursos	*Estrutura computacional para suportar as ferramentas de gestão do conhecimento propostas (alocação de casos, memória organizacional, identificação de requisitos). *Tempo e disposição dos atendentes para alimentar as ferramentas de gestão (casos na memória organizacional, por exemplo).
Conhecimento	*Conhecimento sobre o perfil dos atendentes para melhor alocação dos casos. *Capacidade de síntese e estruturação dos “casos” para a memória organizacional. *Capacidade de filtrar requisitos importantes para a manutenção e evolução dos softwares produzidos pela organização.
Cultura & poder	*Disposição dos atendentes de compartilhar o conhecimento gerado durante os atendimentos. *Valorização do trabalho de estruturação da memória organizacional por parte das chefias (no início, o ônus tende a ser bem maior que o bônus). *O conhecimento gerado no serviço de atendimento deve ser um guia importante para os processos de manutenção e evolução dos softwares (priorização de problemas, identificação de novos requisitos, etc.). Mas deve existir um alinhamento de vontades na organização para que isso aconteça. Um gerente de projetos que não se relaciona bem com o gerente de suporte, por exemplo, pode ser um grande obstáculo.

Quadro 4. Planilha OM-3 Principais tarefas do processo de negócio.

Tarefa	Por	Ativos de conhecimento	Intensiva?
Classificar o problema	Telefonista	Conhecimento sobre os tipos de problemas	Sim
Transferir a ligação para o atendente mais adequado	Telefonista	Conhecimento sobre os tipos de problemas e competências dos atendentes	Sim
Criar nova tarefa no Help Desk	Atendente do Suporte		Não
Analizar o problema e encontrar a solução	Atendente do Suporte	Conhecimentos gerais de informática, sobre o uso dos softwares da empresa, sobre o domínio de aplicação	Sim
Encaminhar para mediação com o desenv.	Atendente do Suporte		Não
Classificar os casos registrados no suporte	Mediador do desenvolv.	Conhecimentos sobre os softwares e domínio de aplicação	Sim
Gerar registro no Bug Track	Mediador do desenvolv.		Não
Identificar novos requisitos	Mediador do desenvolv.		Não
Priorizar casos (correções e novos requisitos)	Mediador do desenvolv.	Conhecimento sobre os softwares, domínio de aplicação, estado das versões, capacidade de desenvolvimento, diretrivas da organização (missão, visão, planejamento estratégico, etc.)	Sim
Encaminhar para o Desenv.	Atendente de Mediação		Não

No Quadro 4 são enumeradas as tarefas que compõem o processo de atendimento. O objetivo principal é identificar tarefas intensivas em conhecimento e os ativos de conhecimento envolvidos. Também são identificados os executores e os locais onde ocorrem as tarefas.

Embora não exista um mecanismo quantitativo específico na metodologia CommonKADS para realizar a classificação e ponderação de intensidade de conhecimento nas atividades

supra-citadas, a identificação das atividades intensivas em conhecimento foi realizada considerando o conhecimento e a experiência dos autores sobre o processo em questão: Atendimento ao usuário de software.

Uma vez identificadas as tarefas que são intensivas em conhecimento, Devem-se listar os ativos de conhecimento que são empregados pelos agentes para a execução das tarefas. Os ativos são listados no Quadro 5.

Ainda no modelo da organização, o Quadro 6 trata das ações propostas para atender às atividades intensivas em conhecimento e a viabilidade de implementação dessas ações.

A partir do contexto organizacional, a análise parte para a construção dos modelos de tarefas e de agentes. Para este trabalho foi eleita uma das tarefas intensivas em conhecimento para detalhamento nessas planilhas: #5 – *Analisar o problema e encontrar a solução*. O Quadro 7 mostra a primeira planilha do modelo de tarefas, identificando objetivos, objetos manipulados, entradas e saídas e medidas de desempenho para a tarefa escolhida.

Quadro 5. Planilha OM-4-Principais ativos de conhecimento.

Ativo de conhecimento	Possuído por	Usado em
Conhecimento sobre os tipos de problemas	Telefonista	2 Classificar o problema
Conhecimento sobre os tipos de problemas e competências dos atendentes	Telefonista	3 Transferir a ligação para o atendente mais adequado
Conhecimentos gerais de informática, sobre o uso dos softwares da empresa, sobre o domínio de aplicação	Atendente do Suporte	5 Analizar o problema e encontrar a solução
Conhecimentos sobre os softwares e domínio de aplicação	Mediador do desenvolv.	7 Classificar os casos registrados no suporte
Conhecimento sobre os softwares, domínio de aplicação, estado das versões, capacidade de desenvolvimento, diretrivas da organização (missão, visão, planejamento estratégico, etc.)	Mediador do desenvolv.	10 Priorizar casos (correções e novos requisitos)

Quadro 6. Planilha OM5-Checklist da decisão de viabilidade.

Ações propostas	<p>Com base na análise feita no sistema em questão, recomenda-se a implementação de um sistema de gestão de conhecimento para o serviço de atendimento ao usuário de software-house. Recomenda-se também, que o foco do sistema seja concebido com foco especial sobre as atividades #3 (“Transferir a ligação para o atendente mais adequado”), #5 (“Analizar o problema e encontrar a solução”) e #10 (“Priorizar casos”). Essas atividades possuem valor estratégico para a organização de software, e são apoiadas pelas outras atividades enumeradas. A tarefa #5 concentra o ponto principal do atendimento ao usuário e emprega o volume mais crítico de conhecimento. Um sistema de gestão do conhecimento (Raciocínio Baseado em Casos, por exemplo), deve aumentar a qualidade do atendimento e diminuir os impactos negativos da rotatividade de pessoal no setor. A tarefa #10 também é importantíssima por tratar dos rumos da manutenção e evolução dos produtos da organização. Processos de manutenção e evolução devem estar alinhados aos interesses dos usuários, para garantir a longevidade dos produtos no mercado.</p> <p>A tarefa #3 impacta diretamente a otimização da equipe de atendimento e objetiva evitar, por exemplo, que atendentes muito especializados gastem tempo com tarefas mais simples.</p>
Viabilidade do negócio	<p>A complexidade do sistema de gestão do conhecimento proposto pode ser considerada alta, mas sua concepção parece viável a partir das técnicas de engenharia do conhecimento. A tarefa #5 constitui um núcleo importante do sistema proposto e poderia ser apoiada por uma estrutura de Raciocínio Baseado em Casos (RBC). As tarefas #2 (classificar o problema) e #3 (transferir ligação para o atendente mais adequado), que conduzem o problema até a tarefa #5, podem ser apoiadas por um sistema de inferência sobre ontologias (essas ontologias tratam de classificar os tipos de problemas e as competências dos atendentes). Outro núcleo importante é a tarefa #10 (priorizar casos para o desenvolvimento), cuja complexidade é bastante grande porque, devendo equilibrar os interesses e dificuldades dos usuários com a capacidade produtiva e planejamento estratégico da organização. Nesta versão do trabalho deixaremos de fora a análise deste núcleo do sistema e nos concentraremos na tarefa #5, mas acreditamos que ela seja automatizável com as técnicas de engenharia do conhecimento já desenvolvidas. Falando do sistema de gestão como um todo, acredita-se que existam riscos tecnológicos (como a dificuldade de se estruturar os casos de suporte), mas estes não inviabilizam o projeto.</p>

No Quadro 8 são detalhados os itens de conhecimento para a tarefa #5. Vários itens representam gargalos, tanto para o processo atual quanto para a solução de gestão do conhecimento. Os itens de conhecimento são agrupados pela sua natureza, forma e disponibilidade.

Quadro 7. Planilha TM-1 Análise da tarefa #5.

Tarefa	5. Analisar o problema e encontrar solução
Organização	Tarefa desenvolvida no processo de atendimento ao usuário de software, dentro da cadeia de valor da software-house
Objetivo e valor	A tarefa deve garantir que o atendente encontre as causas que originaram o problema bem como a proposição de ações ou recomendação que possibilitem a solução do mesmo.
Dependência e fluxo	Veja Figura 2: a tarefa #5 depende das atividades de classificação do problema e alocação do atendente mais adequado.
Objetos manipulados	Problema reportado pelo usuário; informações sobre a versão do software em uso; informações sobre o ambiente computacional em que o software está rodando; o sistema de helpdesk que controla o fluxo dos atendimentos; o sistema de gestão do conhecimento proposto (mais especificamente a estrutura de RBC para recuperação de casos passados e registro de novos casos).
Controle de tempo, pré e pós-condições	O controle de tempo é realizado pelo sistema de help-desk, mas pode ser um artifício importante para avaliação do sistema de gestão proposto - espera-se que, com a aplicação sucessiva do sistema, os tempos de atendimento sejam menores, graças à reutilização do conhecimento. A pré-condição para a tarefa é o relato adequado do problema pelo usuário, que deve fornecer também as informações sobre a versão do software utilizada e sobre o ambiente computacional em que o software opera. Pós-condições: (i) problema do cliente resolvido, ou, quando não for possível, (ii) encaminhamento do problema a um atendente mais adequado. (iii) Caso registrado no sistema de gestão do conhecimento. (iv) Caso reportado para a equipe de desenvolvimento, que pode aproveitar esse conhecimento para planejar suas ações de manutenção e evolução.
Agentes	Atendente de suporte.
Conhecimento e competência	Conhecimentos gerais de informática, sobre o uso dos softwares da empresa, sobre o domínio de aplicação.
Recursos	Sistema de help-desk; sistema de gestão do conhecimento proposto.
Qualidade e desempenho	Podem ser medidos pela satisfação do cliente com o atendimento. O próprio sistema de help-desk pode registrar a opinião do cliente sobre o atendimento, assim como as informações de tempo consumido.

Quadro 8. Planilha TM-2 Itens de conhecimento para a tarefa #5.

Natureza	Gargalo	A ser melhorado
Empírico, quantitativo	x	O conhecimento empregado nos atendimentos tem componentes heurísticos, que, se externados num sistema de gestão do conhecimento, podem ser mais facilmente disseminado entre os atendentes.
Baseado em experiência	x	Atendentes mais experientes têm conhecimento sobre um número maior de casos que atendentes iniciantes. O sistema de gestão visa a reduzir o tempo de aprendizado dos atendentes mais novos.
Incompleto	x	Dada a natureza complexa das interações com o software, nem sempre é possível estruturar todos os aspectos que caracterizam um caso de suporte.
Incerto, pode ser incorreto	x	O sucesso do atendimento depende muito das informações reportadas pelo usuário (características do problema, versão do software utilizada, ambiente computacional, ações realizadas), e deve-se considerar um nível de incerteza nesses componentes.
Muda rapidamente	x	Softwares possuem uma interface muito complexa se comparados à outros dispositivos. A variedade dos casos de suporte é potencialmente infinita, impondo a necessidade de aprendizagem contínua.
Forma do conhecimento		
Na mente	x	Heurísticas já assimiladas pelos atendentes precisam ser explicitadas para o sistema de gestão do conhecimento.
Em meio eletrônico	x	O conhecimento explicitado deve ser arزانado pelo sistema de gestão proposto.
Disponibilidade do conhecimento		
Limitações de tempo	x	O tempo necessário para alimentar a memória do sistema proposto pode ser um gargalo importante no serviço de atendimento.
Limitações de qualidade	x	A qualidade do conhecimento que alimenta o sistema de gestão depende muito da habilidade de síntese e estruturação dos atendentes.
Limitações de forma	x	A complexidade dos casos pode dificultar seu armazenamento de uma forma bem estruturada. Formatos menos estruturados tendem a diminuir a eficiência e eficácia da reutilização do conhecimento.

Quadro 9. Planilha AM-1 Agentes para a tarefa #5.

Nome	Atendente de suporte
Organização	Software-house, setor de suporte
Envolvido em	5 (Analisar o problema e encontrar a solução) e 7 (Classificar os casos registrados no suporte)
Comunica-se com	Telefonista, mediador do desenvolvimento, usuário
Conhecimento	Conhecimentos gerais de informática, sobre o uso dos softwares da empresa, sobre o domínio de aplicação
Outras competências	Habilidade para obter do usuário a melhor descrição sobre o problema reportado e sobre as condições em que o problema foi gerado (ambiente computacional, sequência de operações, etc.). Capacidade de síntese e estruturação para compartilhar os casos no sistema de gestão do conhecimento proposto.
Responsabilidades e restrições	O atendente de suporte interage diretamente com o cliente e, normalmente, essa interação é motivada por um problema, ou seja, por uma situação incômoda. Ele tem a responsabilidade de procurar manter ou reestabelecer a confiança do cliente na organização.

Quadro 10. Planilha OTA-1 Impactos e melhorias-Documento para decisão.

Impactos e mudanças na organização	O sistema de gestão do conhecimento proposto deve impactar positivamente o desempenho da equipe de suporte. A distribuição dos problemas pelos atendentes deverá ser mais coerente com o nível técnico dos mesmos. Com as técnicas de engenharia do conhecimento aplicadas à priorização de casos para o desenvolvimento, espera-se que os processos de manutenção e evolução dos softwares fiquem mais alinhados às necessidades reais dos usuários.
Impactos e mudanças relativas a tarefas e agentes específicos	Tratando especificamente da tarefa #5, à medida que o sistema de gestão (especificamente o módulo de RBC) for alimentado, espera-se que os problemas sejam mais rapidamente solucionados, graças à recuperação de casos passados. Pela mesma razão, espera-se uma maior padronização nas ações propostas para solucionar problemas que se repetem. Outro impacto importante deverá ser a redução dos efeitos negativos da rotatividade de pessoal, já que parte do conhecimento empregado na solução dos casos será explicitado no sistema de gestão.
Reações e comprometimento	(i) Estabelecer um processo formal de utilização do sistema de gestão do conhecimento durante os atendimentos de suporte. (ii) Premiar os atendentes pelo volume e qualidade do conhecimento produzido no sistema.s
Ações propostas	(i) Estabelecer um processo formal de utilização do sistema de gestão do conhecimento durante os atendimentos de suporte. (ii) Premiar os atendentes pelo volume e qualidade do conhecimento produzido no sistema.

O Quadro 9 identifica o agente envolvido na tarefa analisada. Além dos conhecimentos previamente listados, são consideradas também outras competências desejáveis e responsabilidades e restrições para o agente. Verifica-se também as relações com os outros agentes e tarefas do contexto organizacional.

A última planilha da camada contextual do CommonKADS, apresentada no Quadro 10, trata dos impactos e melhorias das ações propostas como resultado da análise.

CONCLUSÕES

O artigo apresentou a análise de um serviço de atendimento ao usuário com base em duas abordagens: a modelagem CESM [2] e o framework metodológico CommonKADS [3]. A visão sistêmica trazida pela modelagem CESM permitiu uma melhor compreensão do serviço de atendimento, ajudando a identificar os elementos críticos necessários para realizar a análise posterior com base no CommonKADS.

As planilhas do Modelo de Organização do CommonKADS ajudaram a identificar o contexto no qual se insere o serviço de atendimento, desde a análise macro da Cadeia de Valor das organizações de software até o mapeamento dos processos por meio da ferramenta *Cross-Functional Diagram*. A partir dessas análises se conclui que o serviço de atendimento é fundamental para garantir a satisfação dos usuários e que a problemática de se gerenciar vários tipos de conhecimento gera riscos para a qualidade de serviço da software-house.

As planilhas dos modelos contextuais do CommonKADS permitiram a identificação de várias tarefas intensivas em conhecimento, cuja automatização (parcial ou total) deve trazer benefícios à dinâmica do serviço analisado.

Dentre as tarefas identificadas, aquela denominada de “*Analisa o problema e encontrar a solução*” foi escolhida para ser aprofundada. Conclui-se que essa atividade precisa de vários ativos de conhecimento para ser executada e que um sistema de conhecimento (englobando uma estrutura de memória organizacional) traria, potencialmente, vários benefícios, como maior agilidade no atendimento e atenuação dos efeitos negativos da rotatividade de pessoal.

Conclui-se também que aspectos de cultura e poder são fundamentais para essa atividade e em geral para os serviços de atendimento ao usuário. A presença de sistemas de conhecimento pode gerar receios associados à perda de poder e de controle, já que o conhecimento,

que era “propriedade” dos atendentes, tende a ser socializado.

As recomendações para continuação deste trabalho seriam: (i) aprofundar a especificação da tarefa “Analisa o problema e encontrar a solução” com as camadas conceitual e de projeto do CommonKADS, que conduzem à implementação final; (ii) preencher os modelos de tarefa e de agente para as demais tarefas intensivas em conhecimento identificadas, aprofundando as que foram consideradas viáveis para implementação.

AGRADECIMENTOS

Os autores agradecem o apoio financeiro da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior-CAPES e do Conselho Nacional de Desenvolvimento Científico e Tecnológico ao autor Mauricio Uriona Maldonado como Bolsista da CAPES/CNPq-IEL Nacional-Brasil.

REFERÊNCIAS

- [1] M. Uriona-Maldonado, A. Coser and G. Varvakis. “Modeling knowledge reuse in technical support operations”. 27th International System Dynamics Conference. Albuquerque, Estados Unidos. 2009.
- [2] M. Bunge. “Emergence and convergence: Qualitative novelty and the unity of knowledge”. University of Toronto Press. Toronto, Canada. 2003.
- [3] G. Schreiber, H. Akkermans, A. Anjewierden, R. de Hoog, N. Shadbolt, W. Van de Velde and B. Wielinga. “Knowledge engineering and management: the CommonKADS methodology”. MIT Press. Cambridge, Massachusetts, USA. 2000.
- [4] E. Motta. “The Knowledge Modelling Paradigm on Knowledge Engineering”. Handbook of Software Engineering and Knowledge Engineering. 2000.
- [5] M.E. Porter. “Competitive Advantage: Creating and Sustaining Superior Performance”. Free Press. 1985.
- [6] J.I. Schwartz. “Construction of software, Problems and Practicalities”. In Practical Strategies for Developing Large Software Systems. E. Horowitz, ed., Addison-Wesley. Reading, Massachusetts, USA. 1975.
- [7] R.S. Pressman. “Software engineering: A Practitioner’s approach”. 4th ed. McGraw-Hill. 1997.
- [8] P. Sommerville. “Software Engineering”. 5th ed. Addison-Wesley. 1995.
- [9] R. Damelio. “The Basics of Process Mapping”. Productivity Press. New York, NY, USA. 1996.