

RECUPERAÇÃO DE CONHECIMENTO EM SISTEMAS DE INFORMAÇÃO LEGADOS

Me. IRAPURU HARUO FLÓRIDO

Instituto Federal do Paraná

Curitiba – PR – Brazil

irapuru@ifpr.edu.br

Dr. MARCOS AUGUSTO HOCHULI SHMEIL

Centro Universitario Dom Bosco.

Curitiba – PR – Brazil

marcos.shmeil@gmail.com

Resumo

Apresentamos uma metodologia baseada em ontologias e agentes de software, proporcionando a gestão do conhecimento dos sistemas de informação legados e auxiliar o desenvolvedor de sistemas a manter e preservar o conhecimento dos mesmos. Este estudo é direcionado principalmente para o apoio das organizações que possuem seus negócios baseados em mainframe como plataforma principal de sistemas, servidores corporativos e tecnologias de terceira geração, como a linguagem COBOL

Palavras-chave:

Sistemas de Informação Legados, Gestão do Conhecimento, Ontologias

Abstract

We present a methodology based on ontology and software agents, providing knowledge management of legacy information systems and assists the system developer to maintain and preserve the knowledge of the same. This study is aimed primarily at supporting organizations that have based their business as the main platform for mainframe systems, enterprise servers and third-generation technologies, such as COBOL.

Keywords:

Legacy System information, Knowledge Management, Ontology.

1. INTRODUÇÃO

Na maioria das vezes, o conhecimento organizacional se apresenta disperso e de forma tácita, necessitando ser explicitado de alguma forma para permitir sua aplicação e manutenção a fim de propiciar qualidade, valor agregado, serviço, inovação, agilidade, flexibilidade e velocidade (Teixeira Filho, 2000). Perante os fatos apresentados, lança-se mão das tecnologias computacionais baseadas em Inteligência Artificial (IA), sobretudo quando à representação do conhecimento com o uso de Ontologias, para viabilizar as organizações nesse cenário adverso, complexo e competitivo. Conforme Shmeil, numa dada sociedade, as organizações se apresentam como entidades independentes complexas ou distribuídas e se caracterizam pelas suas interações intra e interorganizações (Shmeil; Marcos, 1999). De um modo geral, o resultado desse processo contínuo de interações permite que as organizações envolvidas incrementem e refinem seus conhecimentos, tornando-as mais eficientes e eficazes. Como se constata, torna-se importantíssimo desenvolver, manter e reter o conhecimento, tendo em vista facilitar o

processo decisório bem como elevar o valor intrínseco das organizações. A Ontologia, tratada pela IA, vem ao encontro das questões citadas, porque tem como objetivo tornar formal e explicitar a conceituação de um determinado domínio do conhecimento, ou seja, representá-lo, (Davenport, T. H.; Prusak, 1998). No Brasil, organizações financeiras – principalmente as bancárias fundadas no século XX – foram precursoras do uso da informática em seus negócios. Herdaram as primeiras tecnologias utilizadas na área, como as linguagens de terceira geração, por exemplo, o COBOL e o uso de computadores de grande porte, ditos mainframe. A constante preocupação com a performance, em função do grande volume de dados gerados pelas características do negócio e em organizações desse porte, levou os grandes fabricantes mundiais de mainframe, IBM, Bull-Honeywell, Unisys, Digital e outros, a se concentrarem muito mais no desenvolvimento de tecnologias de hardware em detrimento do software. Consequentemente, as inovações tecnológicas de software evoluíram muito mais lentamente, para proporcionar ferramentas de auxílio ao corpo técnico de sistemas na gestão do conhecimento em Sistemas de Informação (SI). Ainda hoje, é de grande expressão o volume dos SI nas grandes organizações que utilizam as mencionadas tecnologias legadas que, por sua vez, serão nominados de Sistemas de Informação Legados (SIL). Conforme levantamento do Gartner Group, empresa internacional de consultoria em TI, no ano 2000, 75% dos dados processados nos EUA eram feitos em linguagem COBOL e havia em torno de 190 bilhões de linhas de código em COBOL e 15% das novas aplicações eram ainda são escritas em COBOL. Nesse contexto, muitas organizações tentaram – e alguns o fizeram – migrar seus SIL do ambiente de mainframe, downsizing, para tecnologias de rede, baixa plataforma, algumas com sucesso, outras não. Muitas se depararam com barreiras tecnológicas e econômicas inviabilizando estas migrações. Determinada por fatores externos, como medidas de governo e alterações nos rumos da economia, a constante mudança no ambiente dos negócios obriga as organizações financeiras a fazerem contínuas manutenções em seus SIL. Tal necessidade, não raro, sobrecarrega a quantidade de trabalho, backlog, do corpo técnico envolvido nos SIL afetados. Então, o uso da tecnologia e de novas ferramentas e métodos devem contribuir de forma a agilizar e tornar mais eficiente o trabalho desse corpo técnico, permitindo não somente facilidades para extrair conhecimentos úteis destes SIL como também viabilizar as manutenções necessárias para o cumprimento das mudanças.

Chega-se então à seguinte questão: como o conhecimento está distribuído nestas organizações? Teixeira afirma que o conhecimento disseminado nas organizações se apresenta basicamente de duas formas: tácito e explícito (Teixeira Filho, 2000). O conhecimento explícito, em geral, apresenta-se registrado de alguma forma e disponível para as pessoas, enquanto que o conhecimento tácito é o que as pessoas possuem e não está registrado de algum modo (Teixeira Filho, 2000). A sistematização e a extração do conhecimento tácito se deparam hoje com barreiras tecnologicamente não solucionadas, ou seja, ainda não é possível à extração automática do conhecimento tácito, retirar o conhecimento da mente das pessoas. Mas, por meio do uso de metodologias e ferramentas computacionais, o conhecimento explícito distribuído em diversas mídias – códigos fonte, manuais, textos, banco de dados – permite a representação e o armazenamento dos conhecimentos necessários no auxílio da manutenção e solução dos problemas dos SIL. Para Davenport e Prusak (1998) a única vantagem sustentável que uma empresa tem é aquilo que ela coletivamente sabe, a eficiência com que ela usa o que sabe e a prontidão com que ela adquire e usa novos conhecimentos. Face ao exposto, busca-se encontrar soluções, desenvolver métodos e ferramentas que auxiliem e permitam a gestão do conhecimento no ambiente organizacional, especificamente em SIL para seu desenvolvimento e manutenção.

2. METODOLOGIA PROPOSTA

Propõe-se uma metodologia para compartilhar o conhecimento nos sistemas de informação legados SIL, embasados na aplicação de recursos tecnológicos como a Ontologia e Sistemas Multiagentes (SMA). A metodologia e a ferramenta computacional, Ontolegacy, fruto deste trabalho, tem o propósito de minerar, extrair termos e conceitos em códigos fonte e realizar o processo de recuperação dos conceitos em SIL, conforme o paradigma da Ontologia. A matéria-prima, isto é, o foco da análise da aplicação do sistema Ontolegacy se concentra fundamentalmente nos códigos fonte escritos na linguagem de terceira geração, COBOL. Tem se os objetivos específicos de: i) recuperar os conhecimentos tácitos dos SIL, que o corpo técnico detém; ii) criar uma ferramenta, ou seja, um facilitador das manutenções nos SIL; iii) formalizar a representação do conhecimento por meio da Ontologia; iv) manter as Ontologias atualizadas por meio de SMA; v) compartilhar o conhecimento por meio das Ontologias criadas dos SIL; e vi) reutilizar as Ontologias para outros SIL. A tecnologia computacional e determinados conceitos da Inteligência Artificial Distribuída (IAD) e de Sistemas de Informação constituem o suporte do trabalho. A Gestão do Conhecimento e os SIL, associados à Ontologia de sistemas e Agentes de software, originaram a essência da metodologia e da ferramenta computacional construída, o Ontolegacy.

2.1 Sistemas de Informação Legados

O maior patrimônio que as organizações detêm na área de informática são suas aplicações corporativas, apropriadamente chamadas de Sistemas de Informação Legados (SIL). Desenvolvidas e aperfeiçoadas ao longo de vários anos, os SIL são o alicerce de muitas organizações e, na maioria das vezes, de missão crítica, essenciais para a sua operação diária. Nos anos 90, ao analisarem os SIL sob uma perspectiva técnica, as organizações concluíram que teriam substanciais economias nos custos do downsizing do hardware, de modo que migraram seus SI para uma arquitetura de aplicações cliente/servidor com sustentação na vantagem da GUI, em relação aos dos sistemas em mainframe. Nesse processo, embora a troca de hardware fosse simples, organizações se depararam com uma barreira mais complexa, que seria o software. Por isso, nem todas as empresas foram unânimes em migrar para plataformas cliente/servidor. Há pouco mais de uma década, as organizações realizaram nos SIL ajustes para o ano 2000, quando os sistemas não conseguiam distinguir anos entre séculos, de maneira que, conseqüentemente, algo deveria ser feito para impedir um possível descontrole no processamento dos dados e, por efeito, nas informações geradas. Outro problema técnico, não publicado extensamente, foi a troca de moeda (euro) em alguns países da Europa, a qual afetou, em muitos casos, organizações que tratavam dos clientes ou dos fornecedores europeus. A complexidade para a manutenção dos SIL aumenta a escala desse problema, quando não devidamente identificado. No Brasil, as maiores organizações detentoras de SIL são as instituições financeiras bancárias. O ambiente com relação a novas estratégias de negócios não poderia ser diferente, as várias mudanças na área econômica às obrigaram por várias vezes a tomar novos caminhos para sua sobrevivência. Algumas organizações, de fato, não sobreviveram a esses reveses.

Na visão de Holland (1999), a perspectiva atual de negócios no ambiente dos SIL tem sido alterada por três poderosos fatores: i) primeiro, a tendência crescente de globalização da economia; ii) segundo, a transformação das economias da sociedade industrial em economias lideradas por conhecimento e bases de informação, por exemplo, a competição baseada em tempo e produtos de curto ciclo de vida, e; iii) terceiro, a transformação dos negócios das empresas, pela qual a organização se afasta de uma hierarquia estruturada

centralizada, para tornar-se achatada, menos hierarquizada e descentralizada (Kelly, Gibson, Holland, & Light, 1999). Diante deste cenário, as organizações em geral, incluindo as financeiras, necessitam de uma agilidade para reagir a estas mudanças impostas pela nova economia mundial. E os SIL, mesmo consumindo grande parte dos recursos destas organizações, não conseguem responder as determinadas necessidades, requeridas em função desse novo paradigma. Um estudo efetuado em quinze grandes organizações nos Estados Unidos mostra a importância e o envolvimento dos SIL, na estrutura, no ambiente de negócios e na visão estratégica da empresa.

Em um ambiente estável de negócios, é possível responder e envolver a infraestrutura de tecnologia de informação e o modelo de negócios para taxas de mudanças relativamente pequenas. Porém, num ambiente oposto, onde as taxas de mudanças são muito mais altas, torna-se difícil adaptar a tecnologia de informação de sistemas legados a um modelo de negócio, rápida e suficientemente para suportar a visão estratégica (Kelly et al., 1999). Assim, conforme o estudo, se as organizações financeiras bancárias se situam num ambiente instável de negócios com altas taxas de mudanças, reforça-se a demanda de uma metodologia que possibilite atender rapidamente a visão estratégica da empresa. Durante vários anos, o COBOL sofreu contínuos refinamentos e padronizações; contudo, a linguagem permanece, em essência, a mesma. Atualmente, existe um padrão aprovado pelo ANSI (American National Standard Institute), instituição voltada a normas e padrões válidos para o território dos EUA, conhecido como COBOL ANS. Esta versão implementa a maioria dos computadores, em âmbito mundial. Abaixo, são listados os vários dialetos do COBOL e seus compiladores, [1]: i) 68, 74, 85. A primeira, segunda e terceira versão do COBOL. Estas linguagens foram liberadas em 1968, 1974 e 1985, respectivamente; ii) UNISYS COBOL, OS/VIS COBOL, COBOL 370, Microfocus COBOL. São produtos de software: compiladores COBOL, seus correspondentes COBOL/ dialetos, e ferramentas para suporte à programação no seu particular dialeto.

Nota-se que muitos dialetos têm incorporado, embedded, como SQL, JCL, CICS etc. Como seu próprio nome indica, o COBOL é uma linguagem voltada sobretudo para problemas comerciais. Diferentemente dos problemas científicos, os problemas comerciais não requerem fórmulas matemáticas complexas, mas necessitam manipular eficientemente arquivos extensos e de constante atualização. Por este motivo, o COBOL possui muitos recursos para a manipulação de dados de entrada e saída. Para tornar-se auto documentado, o programa pode conter comentários que o expliquem. Além disso, o COBOL foi estruturado para ter quatro grandes divisões inter-relacionadas que facilitam identificar setores do programa e a sua leitura, quais sejam: a) Identification Division: divisão de identificação; b) Environment Division: divisão de equipamento; c) Data Division: divisão de dados; d) Procedure Division: divisão de procedimentos, programa propriamente dito. Quando se pesquisa sobre SIL, pouco se encontra sobre soluções que permitam gerenciar de um modo mais simples e eficiente esse vasto conhecimento internalizado nas organizações e de grande valor intrínseco. A maioria das referências diz respeito a mudanças da tecnologia de informação, metodologias centradas em processos de migração dessa plataforma para ambientes orientados, sobretudo, a objetos. No entanto, pelo aspecto dimensional do volume e longevidade dos SIL, não se deve ignorar a importância do ambiente, que muitas vezes armazena regras de negócio no qual o autor não mais pertence ao quadro de recursos humano da organização. Eis por que o emprego das Ontologias tem se tornado tecnologicamente viável para a gestão do conhecimento dos SIL, juntamente com o uso de agentes computacionais. Contudo, a maior dificuldade nesse processo está em transformar e representar o conhecimento na forma tácita para a explícita, porque “o conhecimento tácito é altamente pessoal e difícil de formalizar, o que dificulta a transmissão e o compartilhamento com os outros”. Além disso, vale destacar que inexistente ferramenta para aquisição do conhecimento automático e será muito improvável que haja,

pelo menos em curto prazo, algo a respeito, sendo encontrado algumas que o fazem de maneira semiautomática e é neste sentido que se propõe este trabalho.

2.2 Metodologia para Recuperação do Conhecimento em Sistemas de Informação Legados (SIL): Ontolegacy

Na Informática, a transposição de um conhecimento do plano metafísico para um objeto real, é extensa fonte de pesquisa na disciplina de Inteligência Artificial Distribuída. De fato, metodologias e tecnologias, são intensamente pesquisadas para aplicação na solução de problemas. Eis por que se busca, com este trabalho, desenvolver uma metodologia capaz de melhorar a gestão do conhecimento em Sistemas de Informação Legados. Além da metodologia proposta, é objeto deste trabalho a construção de uma ferramenta para o suporte da adoção dessa metodologia. Assim, antes de se discorrer sobre ela, as características e o funcionamento da ferramenta Ontolegacy são descritos, conforme a seção seguinte.

2.2.1 Cenários da problematização

O atraso tecnológico dos SIL, em ambientes de mainframes, recai sobretudo no uso de ferramentas e linguagens de desenvolvimento ultrapassadas. Além de sua obsolescência, constatam-se outros fatores, como, por exemplo, a ausência da explicitação do conhecimento respectivo. Isto é, os SIL estão em constante transformação e, para tal, o corpo técnico responsável requer o conhecimento acerca do SIL para levar a cabo suas mudanças. Uma organização pode ser vista como uma interface entre um ambiente interior (inner) e um ambiente exterior (outer) (Shmeil; Marcos, 1999), sendo estimulada continuamente por fenômenos a ela relacionados. Conforme o dicionário da Língua Portuguesa (Priberam, 2008), fenômeno é “um fato, aspecto ou ocorrência passível de observação”. Há uma constante transformação do Inner e Outer, relativa a fenômenos ocorridos no mundo a afetar as organizações, que precisam ser flexíveis para manter sua competitividade e inovação. No ambiente dos SI, sujeito com ligações ao outer, o conceito dos sistemas e suas Ontologias são afetados por fenômenos que ocorrem constantemente. Tais fenômenos incidem nas questões dos negócios das organizações, inner. Por exemplo, a alteração da CPMF (Contribuição Provisória sobre Movimentação Financeira) do valor de 0,28% para 0,33%. Pergunta-se: onde este fenômeno gerará mudanças no ambiente organizacional e, conseqüentemente, em quais dos SIL? A resposta será compatível com determinados graus de esforços, dependentes antes do conhecimento tácito do corpo técnico e do conhecimento técnico que estiver explicitado. Quanto maior o conhecimento tácito a respeito do SIL envolvido, menor o esforço e menor o custo para a mudança. Para alcançar um nível satisfatório do conhecimento tácito, é preciso que o conteúdo do SIL esteja explícito, formalizado em documentos para a sua internalização. De acordo com os modos de conversão do conhecimento (Nonaka, I; Takeuchi, 2008) atinge-se algum grau de autossuficiência quando o corpo técnico houver realizado a socialização e a internalização do conhecimento. Mas, quando essas formas de conversão do conhecimento não forem possíveis, não há conhecimento explicitado para sua internalização nem pessoas para a socialização. A metodologia proposta tem o objetivo de resolver esta questão, possibilitar formas de representação do conhecimento para se fazer a internalização.

3 METODOLOGIA PROPOSTA: ONTOLEGACY

A metodologia Ontolegacy busca contribuir no ambiente dos SIL, sobretudo quanto às manutenções desses sistemas. Cabe lembrar novamente alguns dos quesitos a serem atingidos: i) recuperar os conhecimentos tácitos dos SIL, que o corpo técnico detém; ii) formalizar a representação do conhecimento através da Ontologia; iii) manter as Ontologias atualizadas por meio de SMA; iv) compartilhar o conhecimento por meio das Ontologias criadas dos SIL; e v) reutilizar as Ontologias para outros SIL; É importante ressaltar que há um público específico que fará uso dessa metodologia. Assim, elege-se como o maior interessado o corpo técnico envolvido com os SIL, de modo específico profissionais que de fato precisam desse conhecimento: os técnicos que em geral iniciam o aprendizado dos SIL como novos integrantes das equipes e/ou situações em que são transferidas responsabilidades pela manutenção dos SIL para outras equipes. Com o intuito de atingir os objetivos propostos, em termos formais, é preciso estabelecer algumas etapas à construção do ambiente, cuja metodologia assim se estabelece: i) construção da Ontologia dos SIL; ii) definição dos meios para extração dos termos e conceitos; iii) associação da Ontologia dos SIL com os termos e conceitos extraídos; e iv) compartilhamento do conhecimento dos SIL.

3.1 Construção da Ontologia dos SIL

Cabe ressaltar que esta etapa tem papel fundamental para a metodologia como também deve ser a inicial; isto é, as outras etapas são dependentes desta para a condução do trabalho. Há uma diversidade de métodos para construção das Ontologias, mas procurasse seguir principalmente as recomendações adotadas pelo SMI (Stanford Medical Institute) no relatório de Noy e McGuinness (2001). Para esses autores, não há um processo definido para construir Ontologias, mas alguns passos podem facilitar a sua criação. Vale lembrar que as etapas da criação da Ontologia são um processo interativo, no qual mudanças podem ser sugeridas com revisões e refinamentos. Numa primeira etapa, determina-se o escopo da Ontologia proposta, quando algumas perguntas fundamentais devem ser feitas, tais como: i) por que a Ontologia está sendo proposta? ii) O que o escopo da Ontologia proposta irá cobrir? iii) A que se destina a Ontologia? iv) qual a sua finalidade? v) Como a Ontologia será utilizada? vi) para quais tipos de questões de informações a Ontologia deverá prover respostas? vii) quem serão os usuários da Ontologia? viii) quem terá a responsabilidade de manter a Ontologia? As respostas a estas questões são de suma importância, base do escopo da Ontologia. Embora as respostas respectivas possam sofrer alterações no processo de desenvolvimento da Ontologia, busca-se que o escopo do modelo esteja o próximo possível das demandas existentes e que esteja bem definido para que as próximas etapas do processo (Noy & McGuinness, 2001).

Como segunda etapa, pode-se considerar o reuso das Ontologias existentes. Essa medida pode se tornar um requisito à Ontologia proposta, pela necessidade de interagir com outras aplicações que já tenham vocabulários controlados por ontologias particulares. Numa terceira etapa, devem-se enumerar os termos que comporão a Ontologia; ou seja, deve-se criar uma terminologia definida para o domínio de conhecimento em questão. Desse modo, pode-se criar uma lista de seus termos gerais, descrever a que se destinam e quais suas propriedades. Esses termos serão muito úteis para as etapas seguintes. Na próxima etapa, são estabelecidas as classes e a sua hierarquia, a partir da terminologia definida, ou seja, as classes serão criadas com base nos termos gerais definidos. De acordo com Uschold e Gruninger (1996), existem muitas abordagens

possíveis para o desenvolvimento da hierarquia de classes, conforme se pode verificar: i) Top Down – este processo se inicia com a definição dos conceitos mais gerais de um domínio para, posteriormente, chegar-se a um nível de especialização em torno deles; ii) Bottom-up – inicia-se com a definição das classes mais específicas do domínio para então agrupá-las aos conceitos mais gerais; iii) Combination – é a combinação dos processos de desenvolvimento top-down e bottom-up (Uschold & Gruninger, 1996). Inicia-se definindo os conceitos fundamentais para depois fazer a generalização e a especialização, quando apropriado. Nenhum desses métodos pode ser considerado melhor ou pior um em relação ao outro. O processo a ser escolhido depende muito da visão pessoal e do domínio de conhecimento a ser mapeado. Por exemplo: se um desenvolvedor tem uma visão top-down sistemática de um domínio, então pode ser fácil para ele utilizar a abordagem top-down. Por sua vez, o processo de desenvolvimento combination tem sido frequentemente utilizado por desenvolvedores de Ontologias por ser considerado o mais fácil e adaptativo dos três. Noy e McGuinness (2001) definem uma Ontologia como “uma descrição explícita formal de conceitos num domínio do discurso, classes, propriedades de cada conceito descrevendo características, atributos dos conceitos e restrições sobre as propriedades” (Noy & McGuinness, 2001). Uma única classe não provê informação suficiente para responder questões relativas ao escopo da Ontologia. Portanto, uma vez definida a hierarquia de classes, precisa-se descrever a estrutura interna de seus conceitos.

Esta é a quinta etapa, na qual se descrevem os slots que integram as classes, suas propriedades e valores permitidos. Então, os termos gerais definidos na terceira etapa serão transformados em slots das classes. Todas as subclasses de uma classe herdam seus slots e também as restrições de valores para estes slots. Na etapa seguinte, definem-se quais tipos de valores (string, integer, boolean ou outro) os slots das classes podem assumir, quais valores serão estabelecidos, qual o número de valores permitido (cardinalidade) e outras características que os valores dos slots possam ter. Para finalizar a criação da Ontologia, podem-se criar instâncias individuais das classes da hierarquia. Para isso, definir uma instância individual de uma classe requer escolhê-la, criar-lhe uma instância individual e preencher os valores dos slots. Ao se formalizar as etapas anteriores, adota-se um editor de Ontologias que permita sua implementação. Na metodologia proposta, utilizar-se-á o editor KISF (PACHECO, 2003) para a criação da camada ontológica

3.2 Definição dos meios para extração dos termos e conceitos

Com a ferramenta construída, a metodologia enfoca primeiro um aspecto específico dos SIL. Há uma concentração de esforços num meio definido, uma fonte de informação única para a extração dos termos e conceitos. O foco da análise da aplicação do sistema Ontolegacy está nos Códigos Fonte (CF), escritos na linguagem de terceira geração, COBOL. O COBOL tem como característica ser uma linguagem procedural, com divisões de estrutura (como dados e procedimentos), sendo também posicional. Isto é, existem posições (colunas) nas quais devem ser colocados os comandos para seu funcionamento.

Cabe citar que a extração dos termos nos CF ocorre de forma semiautomática, com o auxílio da ferramenta Ontolegacy e um agente específico, para este fim. Nas divisões de um programa COBOL, a DATA DIVISION concentra informações sobre os dados tratados num nível macro pelos SIL. A esta divisão dá-se especial atenção, pois aí está a estrutura de metadados das entidades e atributos, usados pelos SIL como mostrado na Figura 2.

```

000100 IDENTIFICATION DIVISION.
000200 PROGRAM-ID. ALCOMUNDO.
000300
000400 *
000500 ENVIRONMENT DIVISION.
000600 CONFIGURATION SECTION.
000700 SOURCE-COMPUTER. IBM3090.
000800 OBJECT-COMPUTER. IBM3090.
000900
001000 DATA DIVISION.
001100 FILE SECTION.
001200
100000 PROCEDURE DIVISION.
100100
100200 LOGICA-PRINCIPAL SECTION.
100300 INICIO.
100400     DISPLAY "ALO MUNDO !!!".
100500     STOP RUN.
100600 LOGICA-PRINCIPAL-SAIDA.
100700 EXIT.
    
```

Figura 1 – Exemplo de Código em COBOL

Convém ressaltar que além da identificação das entidades e atributos contidos nos CF, faz-se necessário um processo anterior que filtre os CF porque é utilizado um meio no qual estão incorporados os metadados dos arquivos sequenciais (copybook) e os layouts das tabelas do banco de dados.

```

FILE SECTION.
FD XYZS502
BLOCK                0
RECORDING            F
RECORD                200 LABEL RECORD IS STANDARD. COPY 'XYZD502B'.
01 D502B-REG.
05 D502B-TIPO-REG     PIC 9(001).
05 D502B-DT-PROCES   PIC 9(008).
05 D502B-NUM-AG      PIC 9(005)          COMP-3.
05 D502B-NUM-POSTO   PIC 9(003)          COMP-3.
05 D502B-V-INSS      PIC 9(013)V99      COMP-3.
05 D502B-TP-PGTO     PIC 9(002).
05 D502B-TP-DOC      PIC 9(001).

WORKING-STORAGE SECTION.
*****EXEC SQL
***** INCLUDE XYZBCPA
*****END-EXEC
*****
* COBOL DECLARATION FOR TABLE DBAPP.CTL_PRO_XYZ
*****
01 DCLCTL-PRO-XYZ.
10 CPA-NUM-SIST       PIC S9(3)V USAGE COMP-3.
10 CPA-DT-PROCES      PIC S9(8)V USAGE COMP-3.
10 CPA-DT-PROCES-ANT  PIC S9(8)V USAGE COMP-3.
10 CPA-NUM-REMES-GPS PIC S9(7)V USAGE COMP-3.
01 AREAS-DE-SALVAMENTO.
05 AS-RETURN-CODE-4000 PIC S9(004) COMP VALUE +4000.
05 AS-DV-AG           PIC X(001) VALUE SPACES.
05 AS-NUM-SIST        PIC 9(003) VALUE 370.
05 AS-NUM-POSTO       PIC 9(003) VALUE ZEROES.
05 AS-NUM-AG          PIC 9(005) VALUE ZEROES.

PROCEDURE DIVISION.
    
```

Figura 2 - Área de dados (Data Division COBOL)

3.3 Associações da Ontologia dos SIL aos termos e conceitos extraídos

A maior dificuldade encontrada nos processos de manutenção dos SIL se refere ao seu desconhecimento. O passo primeiro para a manutenção dos SIL é a socialização, tentativa de adquirir o conhecimento diretamente do corpo técnico responsável pelo SIL, para que seja assimilado. Outra opção seria a internalização (Nonaka, I; Takeuchi, 2008), isto é, a aquisição de conhecimento por meio de documentos e manuais dos SIL, o que implica compreender a natureza dos SIL pelos conhecimentos que estão no formato explícito.

Quando esses meios não estão disponíveis, busca-se o conhecimento diretamente nos objetos (CF) ativos que refletem a situação real dos SIL. Nesta etapa, pretende-se resolver partes dos problemas expostos com a complementação da Ontologia do SIL, com os termos e conceitos extraídos dos CF, por meio da ferramenta Ontolegacy. Os termos gerados a partir do CF, definidos conforme seção anterior, são selecionados pelo analista responsável ou engenheiro do conhecimento, de forma a complementar a Ontologia do SIL. De acordo com a Figura 3, aparece na Ontologia o conceito Número Remessa, de modo que na extração dos termos no CF exemplificados o usuário seleciona ou não o termo CPA-NUM-REMESS-GPS que fica associado ao conceito Número-Remessa.

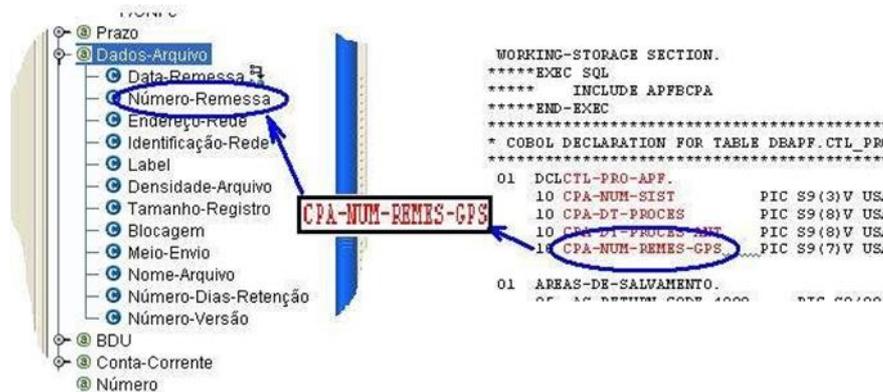


Figura 3 – Associação com a Ontologia

3.4 Compartilhamento do conhecimento dos SIL

O compartilhamento das Ontologias dos SIL se faz importante para disseminar o conhecimento entre as pessoas que trabalham num mesmo ambiente, ou outro, para que possam aproveitar e reutilizar a Ontologia criada. A Web é um meio para tal, dentro e fora das organizações. A emergente pesquisa e criação de linguagens para a Semântica Web têm sido desenvolvidas paralelamente às Ontologias (Noy, Sintek, Decker, Crubezy, & Ferguson, 2001). O entendimento, a compreensão e a troca de informações na Web devem ocorrer com base numa linguagem formal e comum em Ontologias. Por efeito, é essencial que as ferramentas para a criação das Ontologias permitam exportá-las em formatos inteligíveis na Web como o XML e o RDF e, mais recentemente o OWL como padrão de referência para Semântica Web pela World Wide Web Consortium (W3C OWL Working Group, 2012). A seguir, são abordadas as características da ferramenta Ontolegacy, essenciais no contexto desta metodologia, viabilizando, sobretudo, a etapa da associação da Ontologia dos SIL aos termos e conceitos extraídos.

4 FERRAMENTA COMPUTACIONAL: ONTOLEGACY

A ferramenta Ontolegacy é um facilitador na associação da Ontologia do SIL dos objetos reais – códigos fontes – aos objetos conceituais – Ontologia propriamente dita dos SIL. Considera-se que uma das características mais importantes do Ontolegacy seja a capacidade de extrair e gerar termos e conceitos para inclusão na Ontologia dos SIL e também o mapeamento e localização dos termos nos códigos fontes relacionados ao SIL em questão. Apesar dos artefatos computacionais em IA em geral usarem linguagens de programação como LISP ou PROLOG, optou-se como plataforma de implementação do

Ontolegacy a linguagem JAVA. Uma das razões para escolher a plataforma foi a portabilidade e interoperabilidade entre diversos sistemas operacionais, já que pelas características e aplicações do sistema, deverá estar apto para se integrar a diversos ambientes como, por exemplo, ser acessada via Web. O sistema KISF framework, já citado na seção 3.1, baseado em agentes computacionais, será usado em associação com a ferramenta Ontolegacy a fim de: i) disponibilizar o uso do editor de *Ontologias*, ii) armazenar as *Ontologias* iii) compartilhamento do ambiente de SMA, e iv) interface para o ambiente de pesquisa e consultas das *Ontologias*.

4.1 Arquitetura do sistema Ontolegacy

Na arquitetura do Sistema Ontolegacy, têm-se basicamente os seguintes elementos: bases de conhecimento e sociedade de agentes, Figura 4. O agente denominado AgMantenedor é um meta-agente responsável por coordenar um conjunto de agentes cujas competências são bem definidas, quais sejam: a seleção dos CF, a transformação dos CF para análise e a extração dos conceitos e termos respectivos, (Florida,2004).

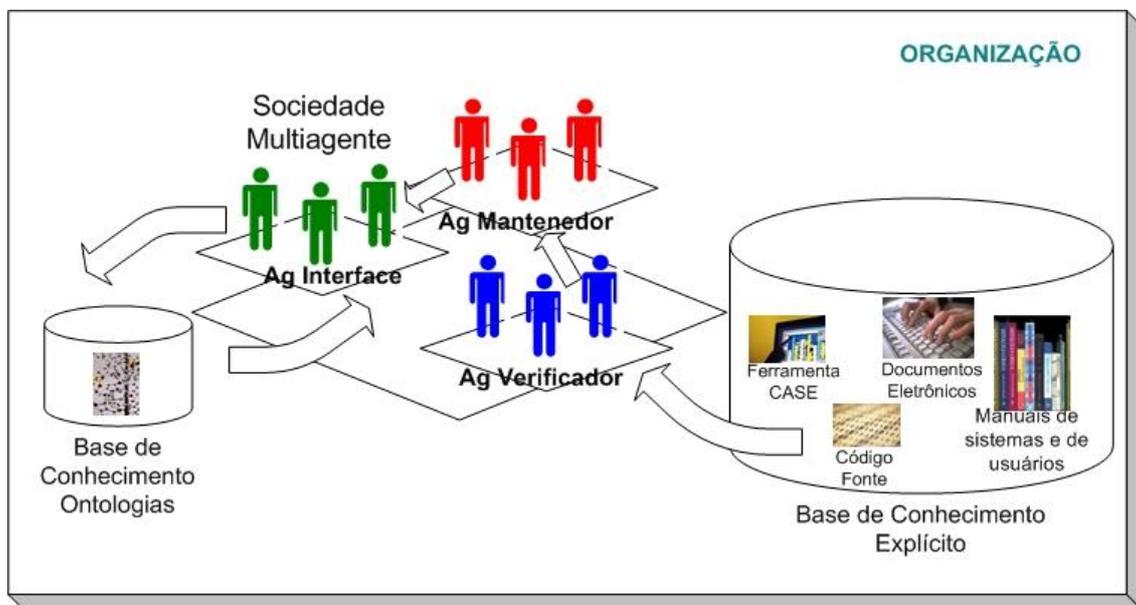


Figura 4 - Arquitetura Geral do Sistema Ontolegacy

4.2 Bases de Conhecimento

São providas de conhecimentos compartilhados pela sociedade de agentes para o alcance dos seus objetivos. Divididas em dois grupos principais são constituídas de: i) base de conhecimento ontológico – onde estão armazenadas as Ontologias dos SIL. Local onde o engenheiro do conhecimento ou o analista de sistemas do corpo técnico responsável pelo sistema armazena e mantém a Ontologia do SIL; ii) base de conhecimento explícito onde está armazenado o conhecimento explícito do SIL, submetido ao processo de externalização do conhecimento. Nas organizações, há uma diversidade de mídias nas quais está formalizado o conhecimento tácito dos SIL. No contexto deste trabalho, é considerada apenas a mídia Código Fonte COBOL, por possuir a maior parte das informações do negócio a respeito dos SIL; ainda, por deter, em termos quantitativos, a

maioria das fontes, e por refletir a situação real dos SIL, ou seja, o sistema que está ativo funciona exclusivamente em função do conteúdo do código fonte.

4.3 Sociedade de agentes do Ontolegacy

A sociedade de agentes do Ontolegacy, é composta de três grupos, formados de um ou mais subagentes, no sistema multiagente proposto, tendo uma interface de comunicação com a base de dados do sistema em que são acessados os dados gerados na investigação dos códigos fontes (Florida, 2004).

```

1PP 5668-958 IBM VS COBOL II Release 4.0 09/15/92                Date 03/07/02  Time
15:44:46  Page      1 0Invocation parameters:
NOADV, FLAG (I, I) , LIB, MAP, NONAME, , OBJECT, OFF, APOST, RENT, RES, TRUNC (BIN) , VBREF, X,
0Options in effect:
NOADV
APOST
NOAWO
BUFSIZE (4096)
NOWORD
XREF (FULL)
ZWB
1PP 5668-958 IBM VS COBOL II Release 4.0 09/15/92                XYZ510  Date 03/07/02  Time
15:44:46  Page      2
LineID  PL  SL  ----+*A-1-B-+-----2-----3-----4-----5-----6-----7-|-----8
Map and Cross Reference
000001          IDENTIFICATION DIVISION.
000003          PROGRAM-ID.                XYZ510.
000007          000700* APLICACAO..... XYZ - ARREC PREVIDENCIA          *
000009          000900* ANALISTA.....  FULANO DE TAL                    *
000013          001300* CODIFICADO EM.... MARCO/1999                      *
000024          002400* FUNCAO.....    CONSISTENCIA DO MOVIMENTO DIARIO GPS  *
000035          ENVIRONMENT DIVISION.
000037          CONFIGURATION SECTION.
000039          SPECIAL-NAMES.
000041          DECIMAL-POINT                IS COMMA.
000043          INPUT-OUTPUT SECTION.
000045          FILE-CONTROL.
  
```

Figura 5 - Código Fonte Expandido

Neste trabalho, optou-se pelo segundo meio, por ter embutidos os leiautes dos arquivos e tabelas de banco de dados do CF, Figura 5, embora este meio esteja de alguma forma “poluído”, com linhas que não dizem respeito ao código fonte em si. Após a atuação do agente AgMSeleção, o meio é transformado em um texto mais limpo, facilitando sua análise posterior pelo AgMAnalisador, Figura 6.

O agente AgMAnalisador - tem um papel importantíssimo no que se refere à preparação do meio para extração dos termos e conceitos. Atua como um parser, fazendo uma análise da sintaxe da linguagem COBOL com o propósito de selecionar as linhas do código fonte essenciais para a atuação do agente AgMConceito. O produto resultante desse agente compreende as linhas de código necessárias para a construção dos conceitos e seleção de termos. O agente Conceito atua no meio selecionado, preparado e submetido ao analisador sintático com o objetivo de gerar os termos e conceitos retirados dos CF.

A proposta da metodologia, conforme descrição na seção 3.2, baseia-se nos metadados dos arquivos e banco de dados embutidos nos CF, particularmente nos códigos fontes expandidos (Figura 5). Cabe ressaltar questões relacionadas à distinção dos termos e conceitos a serem extraídos, porque em CF COBOL é comum que grande parte dos termos, na programação nominadas de variáveis, sejam operacionais e do negócio.

Como é priorizada a Ontologia dos SIL voltada ao negócio, o agente descarta parte desses termos durante sua atuação. Embora o agente não possua inteligência específica para distinguir termos ou conceitos operacionais ou de negócio, o agente Interface possibilita ao corpo técnico do SIL a opção da seleção dos termos e conceitos relevantes à Ontologia.

```

IDENTIFICATION DIVISION.
PROGRAM-ID. XYZ510.
* APLICACAO..... XYZ - ARREC PREVIDENCIA *
* ANALISTA..... FULANO DE TAL *
* CODIFICADO EM..... MARCO/1999 *
* FUNCAO..... CONSISTENCIA DO MOVIMENTO DIARIO GPS *
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SPECIAL-NAMES.
DECIMAL-POINT IS COMMA.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
SELECT XYZE501
ASSIGN XYZE501
DATA DIVISION.
FILE SECTION.
FD XYZE501
BLOCK 0
RECORDING F
RECORD 200
LABEL RECORD IS STANDARD.
COPY 'XYZD501A'.
01 D501B-REG.
05 D501B-TIPO-REG PIC 9(001).
05 D501B-DT-PROCES PIC 9(008).
05 D501B-NUM-AG PIC 9(005) COMP-3.
05 D501B-NUM-POSTO PIC 9(003) COMP-3.
05 D501B-CD-RCT PIC 9(005) COMP-3.
05 D501B-CD-IDENT PIC 9(015) COMP-3.
05 D501B-DT-COMPET PIC 9(006).
05 D501B-V-INSS PIC 9(013)V99 COMP-3.
05 D501B-V-OUTRAS PIC 9(013)V99 COMP-3.
    
```

Figura 6 - Código Fonte Expandido (Limpo)

Os novos conceitos e termos sugeridos pelo AgMantenedor não são atualizados na Ontologia antes da aprovação pelo corpo técnico. Assim, o AgInterface recebe todas as sugestões de resultados provenientes do AgMantenedor. O AgInterface, caracterizado como um agente interface conforme seu nome já explicita, permite que o corpo técnico visualize todas as sugestões listadas, para poder selecionar e aprovar aquelas sugestões que possam auxiliar na complementação da Ontologia dos SIL.

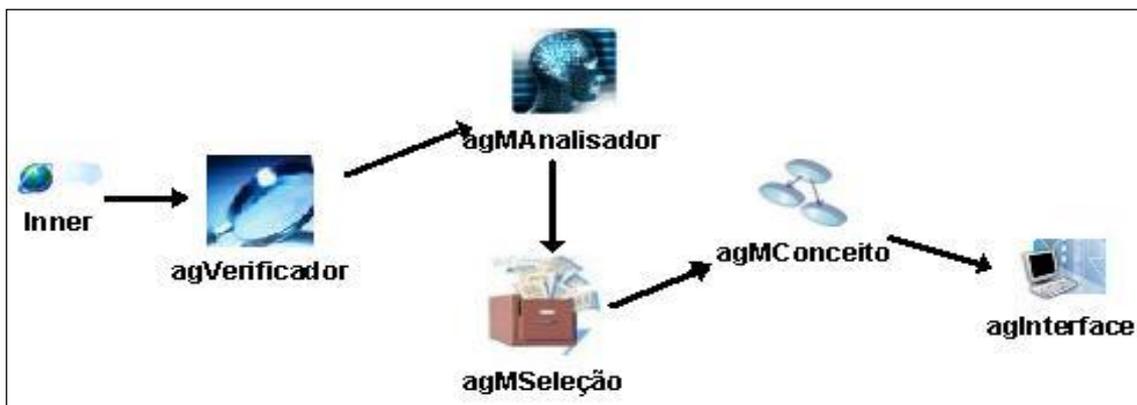


Figura 7 - Sociedade de Agentes Ontology

No momento em que o corpo técnico aprova as sugestões, elas são incorporadas automaticamente à estrutura da Ontologia do SIL. O AgInterface tem dois componentes de configuração: a expertise, que compreende a exibição dos resultados obtidos por todos os agentes mantenedores e a efetivação, na Ontologia, dos resultados aprovados pelo corpo técnico. O outro é o de comunicação, que gerencia a troca de mensagens e informações entre os agentes

5 CONSIDERAÇÕES FINAIS

A metodologia proposta e a ferramenta Ontolegacy contribuem de modo a propiciar à área de SI elementos que envolvem ampliar a obtenção do conhecimento explícito e tácito dos SIL, problema este que emerge não somente nas organizações financeiras bancárias, detentoras de uma tecnologia ultrapassada em algumas áreas específicas de TI, como em outras organizações, intensas usuárias de sistemas corporativos centralizados em mainframe. Assim, espera-se que tenha sido possível demonstrar a ferramenta proposta como facilitador dos processos de mudanças e compartilhamento do conhecimento dos SIL, tônica principal deste trabalho. A diminuição do tempo e a otimização na aquisição do conhecimento têm força na disciplina da Ontologia, a considerar que o mapeamento e a localização de pontos específicos dentro dos SIL se apoia na solução de um SMA. Portanto, o contributo da metodologia e da ferramenta proposta é dar suporte à gestão do conhecimento, sobretudo em aspectos que a norteiam, tais como: o aprendizado, o conhecimento e a inovação. Assim, vale destacar que é com o conhecimento que se faz o aprendizado e a inovação.

REFERENCIAS

BRAND, M. G. J. van den; SELLINK, A.; VERHOEF, C. (1997) “**Obtaining a COBOL grammar from legacy code for reengineering purposes**”. Electronic Workshops in Computing. Springer-Verlag.

DAVENPORT, T. H.; Prusak, L. **Conhecimento empresarial- Como as organizações gerenciam o seu capital intelectual (7ª)**. Rio de (PACHECO, 2003)Janeiro: Ed. Campus. 1998.

FLORIDO, Irapuru. H. **Gestão do Conhecimento em Sistemas de Informação Legados**. Pontificia Universidade Catolica do Paraná. 2004.

KELLY, S., GIBSON, N., HOLLAND, C., LIGHT, B. Focus Issue on Legacy Information Systems and Business Process Change. **Communications of the AIS**, 2(7), 2–27. Retrieved from <http://aisel.aisnet.org/cais/vol2/iss1/9/>. 1999.

NONAKA, Ikujiro; TAKEUCHI, Hirotaka. (2008). **Gestão do Conhecimento**. Porto Alegre: Ed. Bookman. 2008.

NOY, N. F., MCGUINNESS, D. L. (2001). Ontology Development 101: A Guide to Creating Your First Ontology. **Stanford Knowledge Systems Laboratory**, 25.

NOY, N. F., SINTEK, M., DECKER, S., CRUBEZY, M., FERGERSON, R. (2001). Creating Semantic Web Contents with Protégé-2000. **IEEE Intelligent Systems**.

PACHECO, E. J. (2003). **Kisf - knowledge integration & sharing framework**. Dissertação de mestrado em informática aplicada. Curitiba, pr, brasil.

Priberam. (2008). Dicionário Priberam da Língua Portuguesa. **Dicionário Priberam Da Língua Portuguesa**. Retrieved from <http://www.priberam.pt/dlpo/geolocalizacao>

SHMEIL;Marcos. (1999). **Sistemas multiagentes na modelação da estrutura e relações de contratação de organizações**. Tese de doutorado. Universidade do Porto.

TEIXEIRA Filho, J. (2000). **Gerenciando conhecimento: como a empresa pode usar a memória organizacional e a inteligência competitiva no desenvolvimento de negócios**. Rio de Janeiro: Editora Senac.

USCHOLD, M., GRUNINGER, M. (1996). **Ontologies: Principles, Methods and Applications**. Knowledge Engineering Review, 11(02), 63. <https://doi.org/10.1017/S0269888900007797>

W3C OWL Working Group. (2012). OWL 2 Web Ontology Language Document Overview. **OWL 2 Web Ontology Language**, (December), 1–7. Retrieved from <http://www.w3.org/TR/owl2-overview/>