Universidade do Minho

Escola de Engenharia

Departamento de Informática

Metodologia de Integração Orientada a Serviços para a Descoberta de Conhecimento em Relatórios Médicos

Vítor da Costa Pinheiro

Dissertação de Mestrado

Novembro, 2008

Metodologia de Integração Orientada a Serviços para a Descoberta de Conhecimento em Relatórios Médicos

Vítor da Costa Pinheiro

Dissertação apresentada à Universidade do Minho para obtenção do grau de Mestre em Informática, elaborada sob orientação do Doutor Victor Alves.

Novembro, 2008

Para os meus pais, Domingos e Maria, por aquilo que sou,

Para o meu irmão, André, o meu melhor amigo,

Para ti amor, Liliana, estrela da minha vida.

Agradecimentos

Quero aqui, e desta forma, expressar a minha gratidão e o meu obrigado, em especial aos meus pais e irmão, por aquilo que sou e por estarem sempre presentes em todos os momentos. No entanto, não existem palavras para exprimir tudo aquilo que sinto, por todos os momentos, as boas e más experiências, os exemplos e a orientação que me fizeram chegar aqui e estar pronto para enfrentar, na minha caminhada, novos desafios da vida. Mas nesses desafios conto com uma estrela, que alumiará os meus caminhos e que acredita em mim desde o dia em que fixei pela primeira vez os meus olhos em ti meu amor. Liliana és a razão da minha paixão, da minha força, da minha coragem e determinação que existe em mim.

Agradeço também a todos os meus amigos que sempre me ajudaram e estiveram presentes comigo sem nunca pedirem nada em troca. Desses amigos destaco a Sandrine e o Nuno que partilharam comigo as adversidades que nos levaram ao riso, em lágrimas, e das dificuldades sentidas em conciliarmos as nossas vidas profissionais com este ciclo de estudos.

Apesar dessas dificuldades, quero agradecer ao Eng. Hugo Neto, em nome da Wipro Portugal S.A., por ter acreditado em mim e em todo o tempo que me foi dispensado para conseguir completar este mestrado. Agradeço ao CIT – Centro de Imagiologia da Trindade, pela colaboração e ajuda nos relatórios médicos anónimos disponibilizados, dos seus especialistas, técnicos e equipamento sem os quais este trabalho não teria sido possível.

Por fim, demonstro aqui o meu reconhecimento ao Prof. Victor Alves, que eu admiro pela sua forma de ensino e pela sua relação com os alunos, como orientador, assim como a forma que passa da teoria para a prática. Aprendi muito com ele e fico-lhe grato pelo tempo dispensado, conselhos e orientações que me foi dando ao longo da minha licenciatura e agora neste ciclo de estudos.

Resumo

Metodologia de Integração Orientada a Serviços para a Descoberta de Conhecimento em Relatórios Médicos

A recente introdução de normas para relatórios médicos forneceu o suporte para uma eficiente geração, distribuição e mecanismo de gestão dos mesmos, bem como da informação médica contida neles. Estes tipos de relatórios estruturados disponibilizam também uma forma intuitiva e eficaz de representação de informação, ao contrário do formato tradicional de texto livre, facilitando a mineração de dados para a descoberta de conhecimento.

Neste trabalho é efectuado um estudo das normas para a estruturação dos relatórios médicos e apresentada uma estratégia de normalização dos relatórios existentes, da possibilidade de intercâmbio entre as normas ou do uso de diferentes protocolos de comunicação ou mensagens para a transmissão dos mesmos. Com base em técnicas e tecnologias aplicada em outras áreas de conhecimento é possível apresentar uma arquitectura orientada a serviços que permite interligar diferentes sistemas, protocolos e mensagens, facilitando o processo de normalização, codificação, transformação em outros formatos, transmissão e armazenamento numa base dados relacional da informação contida nos relatórios médicos estruturados ou em formato livre. Depois de normalizada a informação, num relatório estruturado ou numa base de dados, são desenvolvidos, aplicados e avaliados modelos de mineração de dados, através de uma ferramenta de mineração de dados, com o intuito da descoberta de conhecimento.

Em suma, neste trabalho é apresentada, avaliada e demonstrada uma metodologia que facilita a descoberta de conhecimento em relatórios médicos, quer nos tradicionais relatórios em texto livre quer nos estruturados.

Palavras-chave: SOI, SOA, Open-ESB, BPEL, Mineração de Dados, YALE, DICOM, HL7.

Abstract

A Service Oriented Integration Methodology for Knowledge Discovery in Medical Reports

The recent introduction of standard structured formats provides an efficient generation, distribution, and management mechanism for the medical reports like as the contained information. These reports give an intuitive and effective manner of information representation, unlike the traditional plain text format facilitating data mining for knowledge discovery.

In this work is carried out a study for the medical reports normalisation standards and presented a normalisation strategy for legacy reports, as well as a purpose exchange definition between the standards or the use of different communication protocols or messages for their transmission. Based on techniques and technologies applied in others knowledge areas it's possible to use service oriented architecture enabling the integration of different systems, protocols and messages, facilitating the normalisation process, codification, format transformation, transmission and storage in a database of the information contained in medical reports structured or legacy. After the normalization process, in a medical report or database, are developed, implemented and evaluated data mining models, through a data mining tool, fulfilling the objective of knowledge discovery in medical reports.

Thus, in this work is presented, evaluated and demonstrated a methodology that facilitates the knowledge discovery in medical reports, whether in legacy reports or in the standard structured reports.

Keywords: SOI, SOA, Open-ESB, BPEL, Data Mining, YALE, DICOM, HL7.

Índice

Índice de Figuras

Índice de Tabelas

Notação e Terminologia

Notação Geral

Ao longo de todo o documento é utilizado itálico, para palavras apresentadas em língua estrangeira, equações e fórmulas matemáticas. O texto utilizado com cor exemplifica extractos de código utilizado. De referir ainda a existência de termos que dado a sua universalidade não foram traduzidos.

Acrónimos

ACR	American College of Radiology
ANSI	American National Standards Institute
B2B	Business to Business
CAD	Computer Aided Detection
CSS	Cascading Style Sheets
DCMR	DICOM Content Mapping Resource
DICOM	Digital Imaging and Communications in Medicine
DTD	Document Type Definitions
EAI	Enterprise Application Integration
HIMSS	Healthcare Information and Management Systems Society

HIS	Hospital Information System
HL7	Health Level 7
IHE	Integrating Healthcare Enterprise
WSDL	Web Service Description Language
IM	Informática Médica
IOD	Information Object Definition
IP	Internet Protocol
ISO	International Organization for Standardization
MIME	Multipurpose Internet Mail Extensions
LOINC	Logical Observation Identifiers Names and Codes
NEMA	National Electrical Manufacturers Association
ODBC	Open Data Base Connectivity
OFFIS	Oldenburger Forschungs- und Entwicklungsinstitut für Informatik-Werkzeuge und -Systeme
OSI	Open Systems Interconnection
PACS	Picture Archival and Communication System
RIM	Reference Information Model
RIS	Radiology Information System
RSNA	Radiological Society of North America
SNOMED	Systemized Nomenclature of Medicine
SNOMED CT	Systemized Nomenclature of Medicine Clinical Terms
SOAP	Simple Object Acess Protocol
SOP	Service Object Pair
SR	Structured Report
TC	Tomografia Computorizada
TCP	Transport Layer Security
UID	Unique Identifiers
URL	Uniform Resource Locator

XML	Extensible Markup Language
XSD	Extensible Schema Definition
XSL	eXtensible Stylesheet Language

Capítulo 1

Introdução

1.1 Normalização de relatórios

Os relatórios médicos são o suporte e ponto de contacto entre o diagnóstico e a transmissão de conhecimento de um médico especialista e o paciente ou outro médico. Na maioria dos casos, estes relatórios são em texto livre sem qualquer tipo de estrutura normalizada ou partilha de conteúdo, que pode ser conseguida através da anexação de imagens aos relatórios médicos. Um relatório médico é um caso de estudo individual sem qualquer tipo de correlação geral ou associação com outros casos de estudo ou diagnósticos. Estes casos podem ser considerados em estudos epidémicos para uma determinada população, amostra de população ou qualquer outra possibilidade.

De acordo com [Dreyer, Mehta & Thrall 2002] a definição e implementação de normas médicas abertas, como a DICOM – Digital Imaging and Communication in Medecine [Web 1] e HL7 – Health Level Seven [Web 2] possibilita a integração de sistemas heterogéneos e informação médica como proposto pela Figura 1.

A norma DICOM define normas para a transmissão, armazenamento e impressão de imagens médicas relacionadas com informação que pode ser usada por equipamentos médicos.

A norma HL7 define normas para intercomunicação e controlo de dados a nível clínico e administrativo, de todo o ciclo de vida documental de um paciente usando HL7 Clinical Document Architecture (CDA).

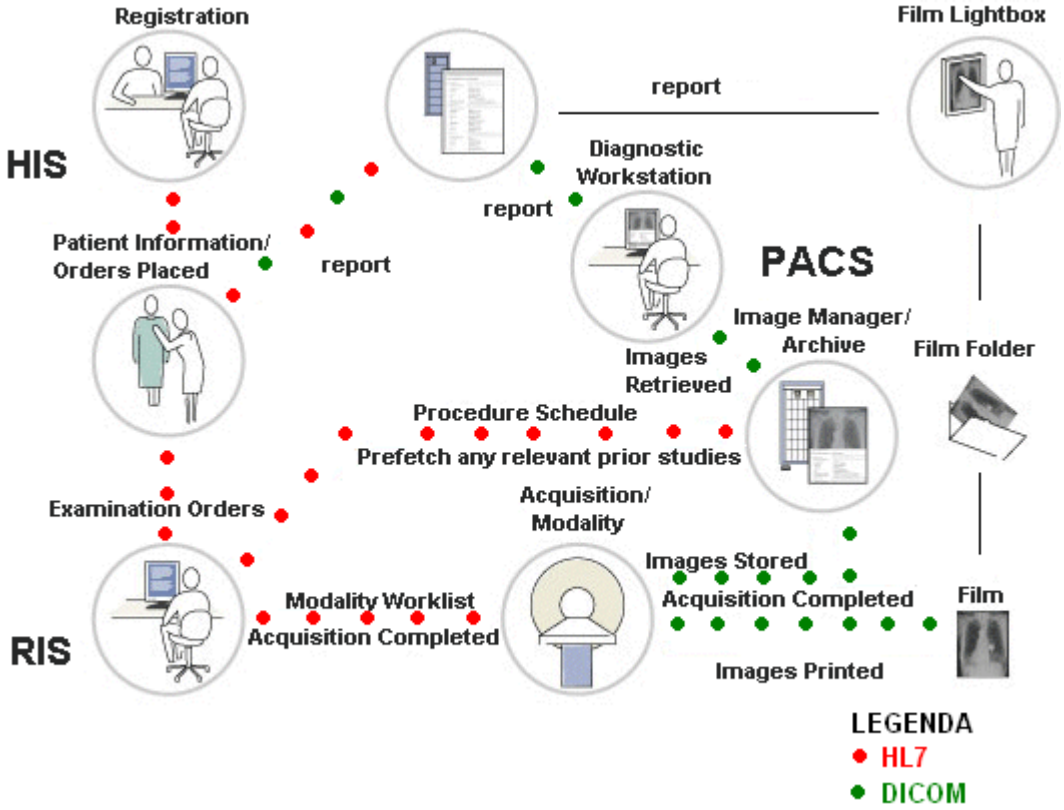


Figura 1: Interligação entre protocolos DICOM e HL7 (adaptado de [Web 23])

Em acréscimo à norma DICOM e à sua definição para a aquisição e armazenamento de formatos de onda (e.g. electrocardiograma) e imagens médicas (e.g. raio-X), a norma DICOM inclui também a definição da normalização de relatórios estruturados Structured Reporting – SR. Esta definição de relatórios estruturados permite, de uma forma expressiva e hierárquica, a representação estruturada da informação médica, como é exemplificado no Anexo A, permitindo incluir texto com referências para outros dados como imagens, formatos de onda, e coordenadas espaciais e temporais [Noumeir 2006].

Contudo, o relatório médico como conhecemos é estático, entregue em papel e algumas vezes sem qualquer tipo de formatação ou em acordo com algum standard. Em geral, este tipo de informação

médica é tratada apenas pelo médico de família que pediu o exame e o médico especialista que diagnosticou. Deste modo, não existe qualquer correlação ou outro tipo de análise de dados ou descoberta de conhecimento para esta informação específica.

Um objectivo inicial do SR passou pela definição de uma forma de codificar a informação contida nos relatórios médicos. Através desta codificação a informação seria mais legível, facilitando a leitura e extracção de informação, relativamente a um relatório em texto livre e sem qualquer tipo de estrutura [Clunie 2007], facilitando a indexação e pesquisa selectiva da informação sem ser necessário recorrer a uma análise sintáctica e semântica de língua natural (NLP) [Langlotz 2002].

De referir ainda que o formato de codificação, de um relatório estruturado e a sua própria estrutura, permite efectuar pesquisas e operações de mineração de dados como, por exemplo, procurar em todos os documentos onde uma massa maligna de uma certa dimensão foi reportada (Figura 2).

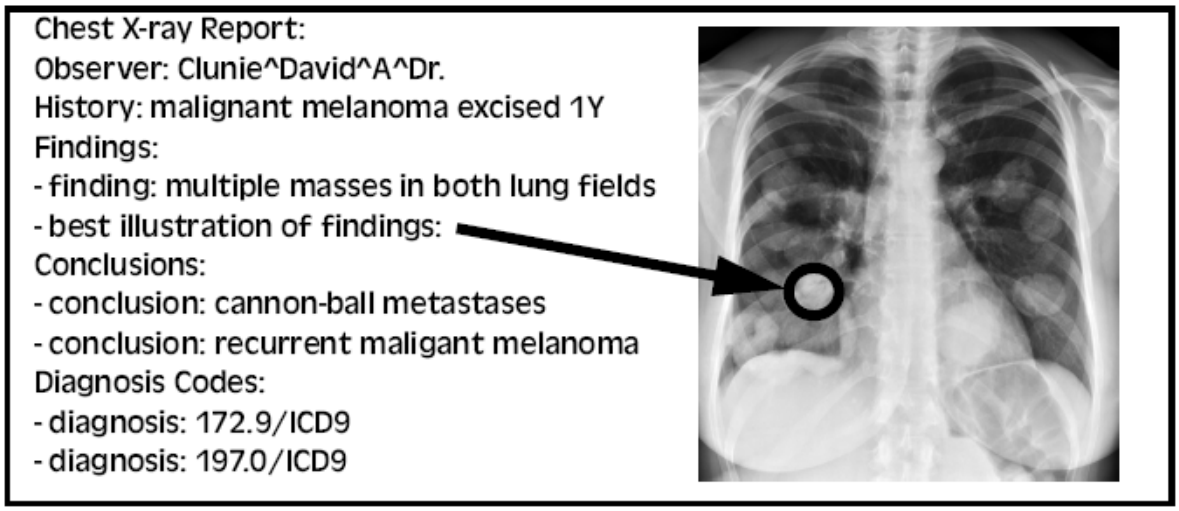


Figura 2: Exemplo simplificado de DICOM SR (retirado de [Clunie, David A 2001])

Estas operações são possíveis porque qualquer elemento de informação é descrito por um código (ver Anexo A). Um código com um determinado valor que pode ser identificado sem qualquer tipo de ambiguidade através de mineração de dados [Noumeir 2006]. As relações e padrões que estão contidas nos dados de um relatório médico podem fornecer um novo tipo de conhecimento médico através da aplicação de métodos, que já foram desenvolvidos em outros casos, para descobrir

conhecimento médico que se encontra escondido da simples observação de dados [Prather et al 1997].

Contudo, para a comunicação externa destes relatórios médicos com outros departamentos é necessário recorrer à norma HL7. De acordo com [Behlen 2007] DICOM SR é uma norma de interesse relevante para o HL7 CDA por diferentes razões. A nível de prática clínica, geralmente, o utilizador final do SR são médicos especialistas que recorrem sistemas baseados em HL7 para visualizar a informação. A usabilidade do SR implica um método de exportar a sua informação para o domínio HL7. Porém, e apesar de os grupos de trabalho que definiram a HL7 e DICOM terem reunido esforços para ajustar o CDA e o SR de forma a evitar incompatibilidades, não existe, ainda, um mecanismo bidireccional de mapeamento entre o SR e o CDA que seja fiável e definido, quer por um grupo ou pelo o outro. Este método bidireccional pode nunca chegar a ser definido bem como não existe nenhum mecanismo de mapeamento, de relatórios de texto livre para relatórios estruturados [Clunie 2007].

Todos estes problemas referidos são comuns numa aplicação de integração num ambiente empresarial (EAI). A tecnologia e métodos aplicados em EAI permitem que protocolos e formatos de mensagens incompatíveis possam ser trocadas por diferentes identidades [Web 8].

Anteriormente à abordagem por uma Arquitectura Orientada a Serviços, desenvolviam-se aplicações conforme a necessidade sem ter em conta factores de escalabilidade ou de interligação entre diferentes aplicações, protocolos ou mensagens. Com a introdução de novas abordagens à integração da informação, como uma Arquitectura Orientada a Serviços, com reutilização de serviços e criação de novos com a possibilidade de se interligarem a outros serviços, internos ou externos, com intuito de fornecerem um serviço ou de se integrarem como parte de um serviço mais abrangente, assistiu-se a um acréscimo de aplicações desenvolvidas em camadas que possibilitam quer a escalabilidade quer a interligação entre diferentes sistemas, protocolos ou mensagens. Em suma, a actual opção que a indústria tem seguido baseia-se nas definições de normas para integração de processos de negócio e dados normalizados na pilha de serviços Web, [Web 8] que estão na base de plataformas de desenvolvimento para integração da informação.

1.2 Descoberta de Conhecimento

Enquanto seres humanos, a nossa aprendizagem e aquisição de conhecimento pode compreender diversas formas, influenciadas pela própria personalidade bem como pelo meio em que vivemos. Esse processo de descoberta de conhecimento pode passar pela simples observação, pelo reconhecimento de padrões, experiências, causa-efeito, entre outras. O meio pode influenciar a forma como partilhamos a informação ou adquirimos novo conhecimento. Tomemos como exemplo, um caso extremo de meio citadino e de meio rural. No meio rural as pessoas partilham mais informação entre si, ainda que por vezes sejam maldizeres uns dos outros, relativamente a um meio citadino onde quase não existe interacção. Ora, a interacção e partilha de informação pode ser importante, por exemplo, num caso de contaminação da água. Num meio rural, ocorrerá um alerta comunitário que se transmite na informação entre pessoas e que pode levar ao conhecimento da contaminação antes desta ocorrer nas restantes pessoas da comunidade. Num meio citadino, esta transmissão da informação é retardada, uma vez que a falta da partilha directa da informação faz com que as pessoas considerem o seu caso como único e individual. No entanto, mesmo que as pessoas partilhassem mais informação entre si, essa mesma partilha de informação iria gerar uma descoberta de conhecimento tardia, visto que o número de pessoas a viver num mesmo espaço é muito superior e, como tal, quando a mesma informação alcançasse finalmente o seu destino já não iria gerar qualquer tipo de conhecimento, mas apenas reforçar o que tinha sido adquirido anteriormente.

Hoje em dia com o desenvolvimento dos meios tecnológicos é possível capturar e guardar uma enorme quantidade de dados que seriam impossíveis de serem tratados e processados por algum humano.

A procura de padrões, tendências, e anomalias nestes conjuntos de dados, e o seu resumo com simples modelos quantitativos, é um grande desafio para a actual era da informação, convertendo dados em informação e informação em conhecimento.

A quantidade de dados recolhida em base de dados, actualmente excede a nossa capacidade, enquanto humanos. Em [Frawley, Piatetsky-Shapiro & Matheus 1991] a descoberta de conhecimento é descrita como a extracção não trivial de informação implícita, desconhecida e

potencialmente útil a partir dos dados. No entanto em [Fayyad, Piatetsky-Shapiro & Smyth 1996] é feita uma clara distinção entre mineração de dados e descoberta de conhecimento que por vezes são as duas mal interpretadas. De acordo com a convenção estabelecida em [Fayyad, Piatetsky-Shapiro & Smyth 1996], o processo da descoberta de conhecimento trabalha sobre os resultados em bruto da mineração de dados. Este processo, extrai tendências ou padrões de dados e de uma forma cuidada e rigorosa transforma os resultados em informação útil e perceptível utilizando técnicas de inteligência artificial.

São necessárias grandes quantidades de dados para fornecer informação considerada suficiente e válida para adquirir conhecimento adicional e, uma vez que estamos a lidar com uma grande quantidade de dados é necessário que estes sejam processados de uma forma eficiente. A precisão é necessária para assegurar que a descoberta de conhecimento é válida e os resultados devem ser apresentados de forma a serem entendidos por humanos. As técnicas aplicadas na descoberta de conhecimento permitem-nos encontrar informação nova e com valor acrescido através dos dados existentes, em grandes conjuntos de dados que excedem a capacidade humana de análise sobre os mesmos, e a dificuldade de efectuar uma transformação precisa dos dados brutos em conhecimento ultrapassa os limites das tradicionais base de dados. Como tal, a utilização total dos dados armazenados depende das técnicas de descoberta de conhecimento aplicadas; sendo que os algoritmos computacionais de aprendizagem são parte integrante da descoberta de conhecimento em base de dados, podendo ser algoritmos supervisionados ou não. Na generalidade dos casos, os algoritmos supervisionados têm uma maior taxa de sucesso em termos de utilidade do conhecimento adquirido, apesar destes algoritmos de aprendizagem serem considerados complexos e a pior parte da aplicação de qualquer técnica de descoberta de conhecimento em base de dados [Brachman & Anand 1996].

Em suma, as diferenças entre as possibilidades humanas e de uma ferramenta especializada na mineração de dados estão, principalmente, no grande volume de dados, no poder computacional exigido, na repetição de procedimentos e na implementação de algoritmos de inteligência artificial. Uma ferramenta de mineração de dados pode adquirir e gerir uma grande quantidade de informação que está para além da capacidade humana. Estas ferramentas podem também

procurar, de uma forma automática, numa base de dados e encontrar até o mais pequeno padrão que possa ajudar numa melhor previsão [Berson & Stephen 1997].

1.3 Enquadramento

A introdução de normas médicas abertas, como HL7 e DICOM, permitem representar de forma expressiva estruturas hierárquicas de informação médica, capazes de conter texto com ligações a outras estruturas de dados como imagens, e descritas por códigos que possam ser identificadas sem ambiguidade possibilitando mineração de dados para a descoberta de conhecimento [Noumeir 2006].

Contudo, os médicos continuam a escrever os seus relatórios ignorando estas normas ou simplesmente não sabem da sua existência. Sendo assim, os seus relatórios podem sofrer de linguagem ambígua [Kong, Barnett, Mosteller, et al 1986] e por vezes não reportam sobre a questão chave [Lussier, Shagina & Friedman 2001], contêm importantes erros clínicos [Holman, Aliabadi, P. Silverman, et al 1994] mesmo que, apesar de um estudo, demonstrar que os médicos especialistas preferem relatórios radiológicos concisos e bem organizados [Naik, Hanbidge & Wilson 2001].

Uma estratégia para resolver o problema da transcrição e da criação de relatórios médicos estruturados passa pela criação de uma aplicação informática, que seja capaz, de uma forma inovadora e avançada de traduzir os relatórios já existentes de texto livre para um formato standard, um formato estruturado, ainda que seja reconhecido que o desenvolvimento de tal aplicação seja uma tarefa com um elevado grau de complexidade [Reiner, Knight & Siegel 2007].

Apesar de existirem normas para os relatórios médicos, ainda não há nenhuma definição formal ou método conhecido de um mecanismo de mapeamento bidireccional entre SR e CDA com total suporte e fiabilidade [Clunie 2007].

1.4 Objectivos

Este trabalho tem como objectivo principal o possibilitar e facilitar a descoberta de conhecimento em relatórios médicos, quer os tradicionais relatórios em texto livre quer os estruturados segundo as normas de DICOM SR e HL7 CDA. Em termos de sub-objectivos pretende-se:

- Desenvolver um serviço de codificação e descodificação de um relatório médico de texto livre para um formato normalizado, com base num modelo;
- Propor um mapeamento entre os termos médicos usados nos relatórios médicos e uma codificação não ambígua para solucionar os problemas inerentes ao uso de linguagem natural nos relatórios médicos;
- Desenvolver uma arquitectura de integração orientada a serviços que seja capaz de interligar diferentes protocolos e mensagens;
- Propor uma metodologia que permita interligar a arquitectura de integração orientada a serviços com a descoberta de conhecimento;
- Avaliar o potencial da metodologia na descoberta de conhecimento sobre a informação normalizada armazenada em base de dados;

Para a persecução deste objectivo será necessário o recurso a diferentes áreas de conhecimento e ao recurso de normas abertas à comunidade.

Em suma, este trabalho disserta sobre uma Metodologia de Integração Orientada a Serviços para a Descoberta de Conhecimento em Relatórios Médicos usando a plataforma Open-ESB [Web 9], e descoberta de conhecimento através de algoritmos de extracção de conhecimento e técnicas de mineração de dados usando a ferramenta YALE [Mierswa et al 2006] baseada na ferramenta Weka [Witten & Frank 2005].

1.5 Estrutura do documento

No Capitulo 2 apresentam-se normas e tecnologias usadas neste trabalho que são consideradas estado da arte nas suas áreas de aplicação. O Capítulo 3, abrange a problemática dos relatórios médicos, estrutura, normalização, codificação, comunicação e transformação em outros formatos. O Capítulo 4, apresenta a estratégia de integração da informação para diferentes sistemas, protocolos ou mensagens e como esta resolve problemas já encontrados em outras áreas e que pode ser aplicada neste caso específico nos relatórios médicos. O Capítulo 5, apresenta uma metodologia que abrange e desenvolve os capítulos anteriores apresentando vantagens, desvantagens e problemas encontrados na sua aplicação.

No capítulo 6, é efectuada uma análise dos resultados obtidos neste trabalho e as conclusões que dele se retiraram, bem como possíveis soluções e melhoramentos em trabalhos futuros.

Capítulo 2

Relatórios Médicos

2.1 DICOM

Com o desenvolvimento tecnológico a nível de sistemas, técnicas e metodologias houve um aumento e necessidade do uso de computadores em ambientes clínicos. Estas necessidades criaram um novo nicho de mercado e foram várias as organizações a desenvolverem sistemas, de acordo com as suas necessidades, sem terem em conta a possibilidade de interagir com outros sistemas externos ou mesmo entre diferentes departamentos dentro do mesmo hospital. Este facto levou à necessidade de estabelecer um ponto em comum que todos implementassem e seguissem como forma de normalização, aplicado a imagens e transferência de informação entre dispositivos médicos e instituições. Muitas organizações que desenvolvem standards para sistemas de informática médica estão focadas de uma forma geral com informação médica, e na troca de imagens médicas, em particular. As aplicações médicas deveriam estar em acordo com estes standards de forma a serem capazes de comunicar de forma normalizada com outras aplicações. A conectividade entre instituições médicas necessita de partilha completa de protocolos definidos por standards.

O standard DICOM [Web 3] surgiu de um esforço em conjunto do ACR – American College of Radiology [Web 4] e o NEMA – National Electrical Manufacturers Association [Web 1] que formaram um comité para desenvolverem Digital Imaging and Commnications in Medicine (DICOM) que define a comunicação e serviços entre aplicações médicas.

O standard DICOM foi desenvolvido de acordo com os procedimentos NEMA e em conjunto com outras organizações de normas como ANSI – American National Standards Institute [Web 5], HL7 – Health Level Seven [Web 2] e IEEE – Institute of Electrical and Electronics Engineers. A norma DICOM define protocolos de rede, codificação de mensagens, modelo de dados em objectos, dicionários de dados, classes de serviços e requisitos de normalização. A norma DICOM específica a comunicação em rede, utilizando o protocolo TCP/IP, os formatos dos objectos (como imagens), os serviços disponibilizados (armazenamento, impressão) e a troca de informação offline (CDs, DVDs) [Clunie, David A 2001].

Esta norma permite a integração de diversos dispositivos (impressoras, scanners, servidores, estações de trabalho entre outros) de diferentes fabricantes no sistema de comunicação e armazenamento de imagens médicas, denominado como Picture Archival and Communication System (PACS). Estes dispositivos estão em conformidade com a norma DICOM, expondo quais as classes DICOM suportadas. A partilha desta norma de uma forma aberta a qualquer pessoa ou instituição tem contribuído para a grande adopção pelos hospitais bem como em instituições de menores dimensões.

A norma DICOM pertence ao campo de informática médica na facilitação da interoperabilidade de equipamentos médicos especificando:

- Conjunto de protocolos que devem ser seguidos pelos dispositivos médicos;
- A sintaxe e semântica dos comandos e informação associada;
- Informação que deve ser fornecida com uma implementação de interoperabilidade e em conformidade com o standard;

No entanto, com o aumento da necessidade de incluir imagens com relevância clínica nos relatórios médicos, surgiu no ano 2000 um suplemento à norma DICOM, o DICOM Structured Report (SR), para normalização da marcação, troca e manipulação de informação relacionada com a informação

estruturada nos relatórios médicos e as respectivas imagens médicas [Kahn, Carrino, Flynn, Peck & Horii 2007].

2.2 HL7

A norma Health Level Seven (HL7) surgiu como uma resposta ao crescente volume de dados clínicos e administrativos, bem como da troca deste tipo de informação específica entre departamentos e instituições. Tal como a norma DICOM que estabeleceu o ponto comum para troca de informação médica, relacionado principalmente com as imagens médicas, a proliferação de sistemas informáticos em ambiente clínico e administrativo originou que toda essa informação precisasse de um ponto comum, de uma normalização para facilitar administração, gestão e troca dessa mesma informação.

A norma HL7 surgiu em 1987, depois da fundação de uma organização constituída essencialmente por prestadores de serviços da área da saúde. Não é por acaso que esta organização escolheu para denominar a sua norma Health Level Seven. A epistemologia da palavra sugere o facto de se tratar de uma norma orientada à área da medicina numa camada superior numericamente identificada como sendo a sétima. Esta estratificação por camadas tem a sua origem no requisito do qual esta norma seria responsável, principalmente, pela normalização de factores de verificação de segurança, identificação de utilizadores das aplicações, disponibilidade de serviços, mecanismos de negociação entre sistemas e estruturação da informação a ser trocada entre sistemas. Estes factores estão relacionados directamente com a última camada, neste caso a sétima camada, do Open Systems Interconnection (OSI) responsável pela aplicação, como pode observar pela Figura 3.

Figura 3: Camadas Open Systems Interconnection (OSI)

A norma HL7, tal como referido anteriormente, tem como objectivo a administração, gestão e troca de informação de dados clínicos e administrativos. Os dados clínicos compreendem informação que é recolhida em laboratórios, prestação de cuidados entre outros serviços. Esta informação clínica precisa, portanto, de ter um ponto em comum, entre as aplicações que tratem esta informação porque pode ser necessário trocas de informações entre diferentes departamentos ou diferentes sistemas. Relativamente à informação administrativa, esta engloba toda a informação relativa a registos de pacientes, quer sejam admissões, marcação de consultas, até aos movimentos financeiros que estejam relacionadas com estes processos ou outros. Resumindo, existe um controlo de todo o ciclo de documentação clínica do paciente.

Ora a informação que é trocada entre departamentos, a transmissão e a informação armazenada dos resultados laboratoriais, o ciclo de documentação necessitam de uma normalização que possa estruturar a informação, facilitando quer a sua transmissão a outros departamentos ou sistemas quer a facilidade de acesso à informação. Como resposta a esta necessidade, tal como a norma DICOM, a norma HL7 desenvolveu o Clinical Document Architecture (CDA).

2.3 Relatórios Médicos Estruturados

Um dos principais objectivos dos sistemas de informação médico é permitir que técnicos e profissionais de saúde possam reportar, e de uma forma facilitada, encontrar dados clínicos sobre os pacientes, como apresentado anteriormente, podendo este tipo de informação estar em diferentes departamentos ou instituições de saúde encontrando-se interligadas entre si por protocolos específicos como o DICOM e HL7.

Com o objectivo de definir interoperabilidade entre sistemas médicos e de normalizar protocolos e mensagens surgiu a Integrating Healthcare Enterprise (IHE) [Web 22]. A IHE é uma organização internacional criada por profissionais de saúde e de indústria com o objectivo de melhorar a forma como sistemas informáticos usados na área da saúde partilham informação através de processos de interoperabilidade, possibilitando a integração de informação médica em diferentes sistemas sem a necessidade da intervenção humana, reduzindo deste modo os custos e o risco de ocorrer erro humano. A Figura 1, exemplifica para além da interligação entre os protocolos médicos, um fluxo de trabalho que pode ocorrer num ambiente de saúde. Trata-se de um caso de estudo do processo normal de um exame radiológico. Numa primeira etapa, o paciente é atendido, registada a informação do paciente e termina com o pedido electrónico para a realização de um exame radiológico. Numa segunda etapa, depois de realizado, o exame é armazenado em formato digital, imprimido e segue para o médico especialista para diagnóstico. O estudo deste fluxo foi utilizado na definição da interligação das duas normas médicas, DICOM e HL7, num sistema de informação de saúde. Esta definição permite interoperabilidade entre o sistema de informação radiológica, Radiology Information System (RIS), o sistema de comunicação e armazenamento de imagens, Picture Archiving and Comunication Systems (PACS) e o sistema de informação hospitalar, Hospital Information Systems (HIS).

No entanto, na maior parte destes sistemas, a informação clínica relativa ao paciente e a resultados anteriores encontra-se armazenada numa base de dados em texto livre. Mas, o problema que advêm do uso de um formato em texto livre está na falta de estrutura, dado que a informação está

enredada na linguagem do relatório, dificultando a tarefa de comparar relatórios ou encontrar um detalhe específico sem ter que ler todo o texto. Por exemplo, um médico radiologista que queira seguir uma determinada lesão ao longo de uma série de relatórios vai precisar de ler muitas linhas de texto. Ao contrário, de um sistema de relatórios estruturados em que a informação encontra-se estruturada, normalizada, perceptível, com indicação clara de cada evidência clínica encontrada no exame, assim como, o tamanho e localização correspondentes.

2.3.1 DICOM SR

O Suplemento 23 da norma DICOM introduz as classes disponibilizadas pelo DICOM SR, que são usadas no armazenamento e transmissão de informação clínica sob a forma de um documento estruturado apoiado na estrutura hierárquica e dos serviços disponibilizados pela norma DICOM. Estas classes servem de suporte a texto livre assim como a informação estruturada, contribuindo para a clareza, precisão e transmissão da informação e permitindo efectuar a ponte entre sistemas de imagem e de informação médica. Esta ligação só é possível porque esta norma permite incluir imagens médicas que suportam o diagnóstico e que podem ser evidenciadas directamente no texto.
No entanto, este tipo de documento não precisa de ser necessariamente um documento complexo; apesar, de na generalidade dos casos os relatórios serem individuais e relativos a um paciente, os documentos SR não são estritamente para pessoas, pacientes, pode também ser aplicados a outros fins médicos, como por exemplo, amostras de tecido humano ou apenas incluir uma referência a uma única imagem e uma evidência clínica [Clunie, David A 2001].
Uma vez que o SR é suportado pela norma DICOM este usa como base e apenas o cabeçalho da norma DICOM, contendo informação demográfica, identificação e de uma árvore de conteúdos que na sua essência é uma estrutura recursiva de pares nome e valor correspondente. Dado que a norma DICOM é orientada a objectos, binários, de informação sequencial e a informação (valor) é identificada por um código único, inequívoco, independente da forma como a informação possa ser visualizada. O SR segue as mesmas orientações utilizando a codificação da informação em vez da

informação textual, bem como de referências a imagens, coordenadas de regiões das imagens e medições efectuadas sobre as imagens em estudo [Clunie 2007].

Os relatórios estruturados de acordo com a norma DICOM SR podem ser usados para diferente fins com diferente níveis de complexidade. Estes diferentes níveis de complexidade estão relacionados com o objectivo do relatório ou de acordo com a complexidade do diagnóstico. Com base nisto podemos considerar três diferentes classes de relatórios estruturados:

- Texto Básico – uso mínimo de códigos para título do documento, subtítulos e árvores hierárquicas de subtítulos;
- Realçado – super conjunto do Texto Básico, medidas numéricas com códigos representativos de referência para unidades e medidas, imagens e formatos de onda;
- Detalhado – super conjunto do Texto Básico e Realçado com referências entre elementos.

Estas classes estão relacionadas com a complexidade e o grau de abrangência dos relatórios. As diferenças entre estas classes encontram-se nas restrições impostas ao documento.

A informação contida num SR é agrupada em nove módulos, nos quais os itens de informação estão relacionados entre si. Existe um módulo para informação relativa ao paciente, como data de nascimento e peso; um módulo para informação genérica de acordo com o documento, como o nome da pessoa responsável por verificar o documento e indicadores para representar se um documento já se encontra completo, por exemplo, entre outros.

A informação que se encontra no módulo de conteúdo do documento é dividida em Itens de Conteúdo. Um item de conteúdo consiste num par nome e valor onde o nome é um código de um dicionário de termos como o SNOMED CT – Systematized Nomenclature of Medicine-Clinical Term [Web 10], e o valor é um tipo entre os catorze tipos de valores definidos por defeito. Os tipos principais são text (para texto), num (para numero, percentagens), image, date, e waveform. Os itens são organizados hierarquicamente, para que a informação nos níveis superiores possam conter, ou serem derivados, informação de itens num nível inferior, abaixo dos mesmos. A norma DICOM SR especifica oito diferentes tipos de relacionamento, entre eles encontramos:

- contains - o nodo pai da informação está contido no nodo filho;
- has properties - a informação para o nodo é uma propriedade da informação do filho no nodo do pai, i.e., descreve a condição das propriedades da fonte;
- has obs context – o alvo transporta especialização do Contexto de Observação necessária para a documentação da fonte;
- has concept mod - descreve ou qualifica o Nome Conceptual da fonte;
- has acq context - o alvo descreve a condição durante a aquisição de dados na fonte;
- selected from – a fonte transporta coordenadas espaciais ou temporais seleccionadas pelo alvo;
- inferred from - a fonte transporta a medição ou outra inferência feita pelo alvo.

A seguinte sentença e exemplo são adaptados de [Bortoluzzi 2003], em língua materna, e podem ser divididas em itens de informação e organizadas numa hierarquia como ilustrado na Figura 4:

Hábitos Pessoais:

Em média o paciente bebeu 7 cervejas por dia nos últimos 2 anos.

O paciente é obeso e frequentemente desmaia.

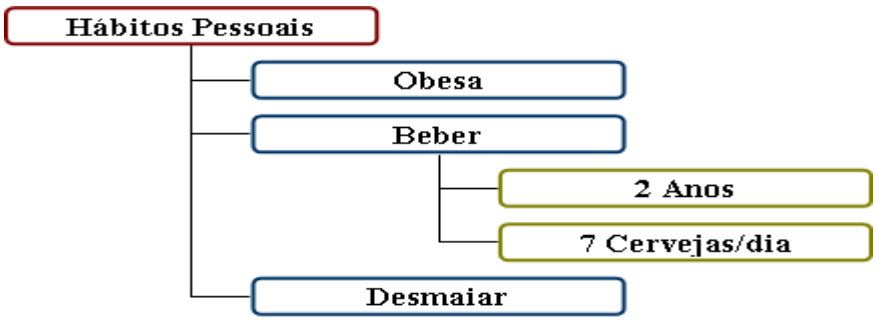


Figura 4: Exemplo de hábitos pessoais – divisão da informação de acordo com os conceitos mais importantes.

Em DICOM SR cada item deve ser um par nome e valor e cada um relacionado com o item pai e o tipo de valor indicado acima do item.

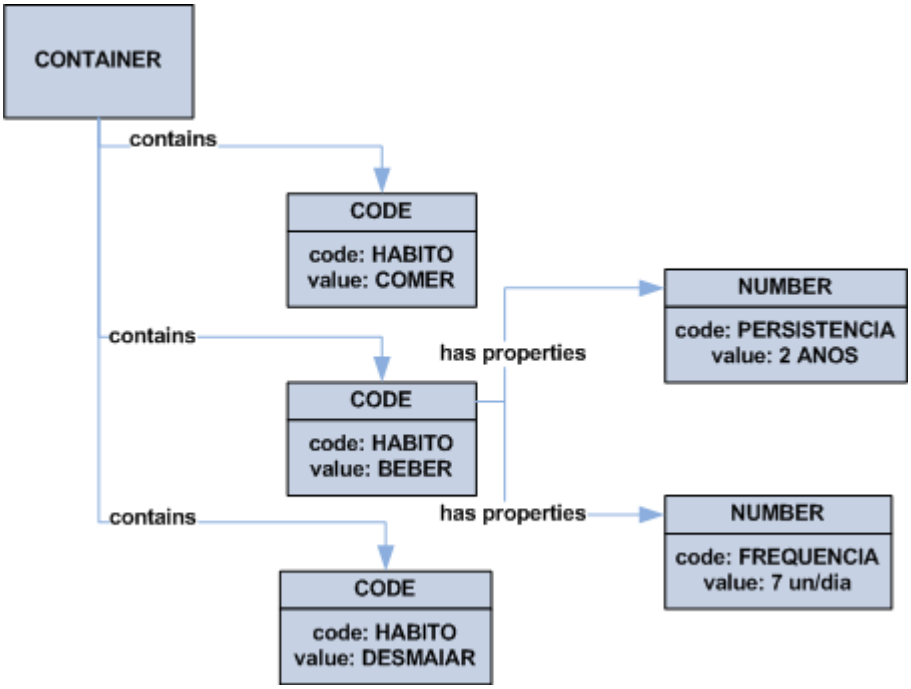


Figura 5: Hierarquia dos itens de acordo com a DICOM SR

A Figura 5 demonstra uma hierarquia de itens de acordo com a DICOM SR permitindo visualizar que cada Item de Conteúdo é representado por um par Nome Conceptual/Código facilitando a indexação e pesquisa. Na realidade o Nome Conceptual é codificado a partir de termos médicos normalizados como o SNOMED ou o mesmo usado pelo DICOM.

A principal questão sobre os relatórios estruturados está na integração e aceitação pelos médicos desta forma de representar o diagnóstico. Na sua maioria, os médicos possuem a sua própria estrutura, definida num ficheiro do tipo dot e não pretendem ser restringidos por nenhum formato ou norma.

Vantagens

A norma DICOM SR apresenta diversas vantagens significativas que podem diminuir uma possível resistência à mudança entre o formato tradicional de relatórios médicos e o que é proposto por esta norma como:

- A norma DICOM é usada e implementada em praticamente todos os equipamentos de imagem médica e encontra-se disponibilizada gratuitamente;
- As imagens contêm informações adicionais que permitem a alteração da imagem exibida;
- Permite a transferência sem perda da qualidade entre diversos sistemas;
- A norma está disponível gratuitamente e dispõe de templates que de uma forma geral facilitam a criação de um relatório;
- A informação encontra-se estruturada, normalizada, perceptível e compreensível, com indicação clara da evidência clínica encontrada no exame, assim como, tamanho e localização correspondente;
- Todos os elementos de informação são descritos por um par código e valor permitindo indexação e mineração de dados;
- Para além da informação estruturada, inclusão de imagens é também possível incluir formas de ondas e áudio (útil para armazenar a dicção do exame do medico especialista);
- De acordo com [Noumeir 2006] a sua estrutura e normalização facilita a transformação deste formato em formatos universais como o XML, PDF ou formatos específicos de informação clínica como o CDA ou através de mensagens HL7.

Desvantagens

- A norma DICOM é difícil de entender e de efectuar a sua implementação;
- Permite que várias tarefas sejam realizadas de múltiplas formas induzindo ambiguidade;
- O desenvolvimento é dificultado pela complexidade e dimensão da norma. Em termos gerais não existe uma implementação da norma DICOM por completo e existe uma acentuada curva de aprendizagem;

- Algumas das dificuldades são causadas e inesperadas entre a compatibilidade de muitos equipamentos uma vez que a implementação da norma varia com a interpretação de quem implementa.
- Em termos de software, na codificação automática de relatórios de texto livre neste formato e na utilização de um aplicativo intuitivo e fácil de utilizar que permite ser usado por diferentes técnicos e profissionais de diferentes áreas de saúde [Hussein, Engelmann, Schröter & Meinzer 2004];
- Em termos de PACS existe a dificuldade de suportar, gerir e modificar diferentes templates de diferentes organizações e de uma aplicação genérica que sirva todas as especialidades médicas [Hussein, Engelmann, Schröter & Meinzer 2004].

Desafios

As vantagens e desvantagens apresentadas aqui permitem retirar conclusões acerca da usabilidade, implementação e aplicabilidade neste trabalho.

Em termos de aplicação, o seu uso implica ter um conhecimento aprofundado desta norma, da sua especificação e possível implementação. Apesar das desvantagens apresentadas, um dos objectivos deste trabalho passa por apresentar uma solução para a codificação dos relatórios médicos, em texto livre e sem qualquer normalização em um relatório médico estruturado apoiado numa norma. Nesta secção e nas anteriores foi apresentada e estudada a DICOM SR como norma de relatórios médicos que pode ser usada neste trabalho. No entanto, um outro objectivo passa pela sua utilização para integração da informação e transformação em outros formatos quer de apresentação, em ficheiros de formatos aceites universalmente como o pdf, Word, assim como de estruturação e transporte como o CDA HL7. As vantagens e a discussão que já foram apresentadas e debatidas em secções anteriores reforçam a sua usabilidade num ambiente médico. No entanto, as desvantagens também antecipam alguns cuidados que devem ser tidos em consideração. Destes factores podemos destacar o da implementação. A implementação da especificação desta norma exigiria um grande esforço e não é um dos objectivos deste trabalho. O que se pretende desta norma passa pela sua aplicabilidade e usabilidade em termos de relatórios médicos. Sendo assim,

a preferência será pelo uso de uma aplicação que seja capaz de converter o conteúdo de um DICOM SR (formato de ficheiro ou conjunto de dados) para um formato aceite universalmente como por exemplo XML, facilitando o processo de integração da informação contida num relatório médico do tipo DICOM SR.

2.3.2 HL7 CDA

O principal objectivo da HL7 está na normalização da informação clínica e administrativa de uma unidade, departamento ou instituição médica ao invés das normas médicas já existentes que se focavam na centralidade dos sistemas médicos. No entanto, esta orientação, não à centralidade dos sistemas, mas à administração, gestão e transporte de informação clínica e administrativa específica como, por exemplo, o registo de uma admissão de um paciente ou de um procedimento clínico obriga à especificação de uma estrutura e semântica normalizada para este tipo de documentos clínicos. Como resposta a esta necessidade surgiu uma norma de Arquitectura para Documentos Clínicos (CDA – Clinical Document Architecture) acrescida ao standard HL7.

Ora esta perspectiva de normalização da informação transportada por um CDA, por exemplo, de uma admissão de um paciente, exige que exista de alguma forma um suporte que possa ser comum entre os sistemas. Este suporte comum a todos os sistemas teria que ter a vantagem de que todos eles seriam capazes de integrar a informação sem ser necessário grandes modificações. Claro que acrescida a esta vantagem ainda podíamos juntar o facto de ser informação, na sua essência, textual permitindo uma fácil leitura a olho nu sem ser necessário recorrer a alguma aplicação de visualização. Estes factores influenciaram na escolha do XML (Extensible Markup Language) como suporte para a codificação dos CDA permitindo a representação, normalização, partilha da informação de uma forma universal, quer para máquinas quer para humanos, pela flexibilização que oferece para a representação de informação, bem como a possibilidade de acrescentar conteúdos, imagens, sons e formatos aceites universalmente [Dolin et al. 2006].

O CDA está incluído na emergente família da norma HL7 da versão 3 e o seu conteúdo semântico deriva do modelo partilhado pelo HL7 RIM (Reference Information Model) [Web 25]. O principal

objectivo do HL7 RIM é de tornar a informação significativa para além do contexto local, i.e., o âmbito do HL7 RIM está na informação necessária para ser compreendida entre sistemas de informação mas não necessariamente toda a informação que se encontra num dos sistemas. Em conjunto, a versão 3 e o RIM fornecem mecanismos que possibilitam a incorporação no CDA de conceitos médicos normalizados como o Systemized Nomenclature of Medicine Clinical Terms (SNOMED CT) ou o Logical Observation Identifiers Name and Codes (LOINC). A especificação de um documento CDA é independente do transporte, ou seja, a terceira versão das mensagens do HL7 não é necessária. Este documento pode ser enviado numa mensagem HL7 ou por diferentes mecanismos de transporte como do protocolo http, serviços Web, FTP e SMTP [Web 24].
Uma vez que a informação se encontra armazenada numa base dados em XML, o CDA permite definir três níveis de arquitectura, de acordo com [Paterson, Shepherd, Wang, Watters, & Zitner 2006]:

- Nível 1 – está orientado ao conteúdo dos documentos narrados não havendo semântica a este nível mas apenas texto humanamente legível;
- Nível 2 – modela-se as observações e instruções de cada classe do RIM;
- Nível 3 – é possível um documento que seja completamente estruturado e onde a semântica de cada par entidade e informação seja especificada por um código único.

Um documento CDA fornece um suporte de partilha de dados, entre as bases dados hospitalares e uma base de dados médica, possibilitando a sua extensão para outros tipos de documentos e a realização de pesquisas através de dados clínicos que são estruturados e identificados por um código. O CDA encontra-se dividido em dois tipos: CDA R1 – Release One e CDA R2 – Release Two. A primeira versão do CDA, o CDA R1, foi formalizada no ANSI (American National Standards Institute) sendo aprovada pela norma HL7 em 2000, representado desta forma a primeira especificação derivada do HL7 e RIM. Em relação ao CDA R2, segundo [Dolin et al. 2006], esta versão foi aprovado pela ANSI através do HL7 desde a existência da primeira versão do CDA, em particular para a área de representação semântica de eventos clínicos.

O CDA como documento estruturado e normalizado para transporte de informação entre diferentes sistemas é suportado pela representação da informação em XML. Este formato proporciona o uso de identificadores (tags) que permitem identificar e estruturar a informação, facilitando o seu armazenamento e visualização através de páginas Web ou transformação em outro tipo de documento. Acrescentado a isto, o CDA possui um mecanismo próprio de validação do documento em XML e uma folha de estilos personalizada para apresentação da informação que contém.

Algumas das estratégias usadas recentemente dão uma maior importância à centralização de sistemas de informação na prestação de cuidados médicos aos pacientes. No entanto, existem diversos sistemas que são responsáveis por diferentes tipos de informação que implementam diferentes protocolos e mensagens em acordo com as necessidades específicas de cada caso. Uma vez que possuem diferentes protocolos e mensagens ficam impossibilitados de partilhar informação e existindo uma falha na utilização e reutilização de informação. No entanto o CDA pode ser útil na resolução destas questões como ilustra a Figura 6 e onde se podem ver várias das soluções propostas pelo CDA para unificar a informação distribuída por diferentes áreas e departamentos.

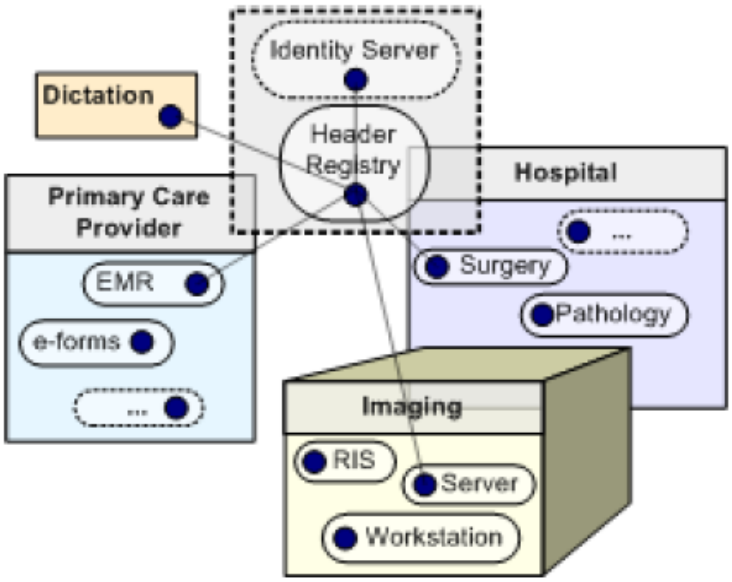


Figura 6: Modelo Distribuído proposto para uso do CDA (Retirado de [Web 24])

A Figura 6 ilustra a forma como um documento CDA pode ser composto por diversas informações de diferentes fontes e como, por exemplo alguém do departamento de cirurgia do hospital

observaria um relatório de imagiologia sendo que a comunicação entre estes diferentes sistemas ocorre por via do registo do cabeçalho [Web 24].

O modelo distribuído proposto para o uso da norma HL7 CDA pode ajudar na forma como ocorrem as transferências de informação clínica entre diferentes sistemas existentes, quer seja no departamento ou sitio em que ocorre a primeira prestação de um cuidado médico, até ao momento em que um paciente recebe alta. Desta característica advém a principal diferença para outras normas e sistemas que são mais egocêntricos. Tal como ilustrada na Figura 7, a perspectiva da norma HL7 CDA, está na informação do paciente, contrariamente ao que normalmente acontece que é a centralização dos sistemas e informação.

Figura 7: Informação do paciente centrada no cuidado médico (Adaptado de [Web 24])

Objectivos

Segundo o HL7 CDA Brief [Web 24] a utilização do CDA tem como principais objectivos os seguintes pontos:

- Assumir como prioritário o cuidado prestado ao paciente;
- Minimização das limitações técnicas na implementação de normas;
- Promoção da longevidade de documentos e registos clínicos;

- Promover a troca de informação independentemente dos mecanismos de transporte e de armazenamento;
- Garantir a independência dos sistemas de informação;
- Prevenir a falta de reutilização de dados electrónicos clínicos;
- Utilizar documentos XML com estrutura bem definida;
- Diminuir o uso de registos clínicos em papel;
- Possibilidade de ser visualizado num browser Web ou por um editor de texto;

Vantagens

De acordo com o HL7 CDA Brief [Web 24] as principais vantagens na utilização do CDA R2 são:

- Privilegia a estrutura do documento;
- Baseia-se em normas abertas à comunidade;
- Independência do mecanismo de transporte;
- Suporte para envio de meta-dados em conjunto com dados;
- Possibilita o uso de qualquer sistema de codificação e vocabulário, como o SNOMED ou LOINC;
- Abrange um conjunto de dados mais completo do que o normalmente utilizado e diminui os campos obrigatórios num documento;
- Ser suportado por XML, ter um modelo de referência bem definido como o HL7 RIM, usar um vocabulário codificado e possuir uma estrutura normalizada e simples, sendo flexível o suficiente para ser lida em qualquer plataforma e por qualquer aplicação;

Desvantagens

No entanto, apesar das vantagens identificadas em cima também se apontam algumas desvantagens identificadas em [Web 24] e em [Web 26]. As principais desvantagens identificadas são:

- Não existe um grande número de implementações;

- Implica um profundo conhecimento do RIM para a esquematização da estrutura;
- Demasiada flexibilidade;
- Em determinadas situações pode ser bastante complexo (fornecendo um suporte mais completo do que o necessário);
- A norma HL7 V2 é muitas vezes interpretada de diferentes formas dependendo da perspectiva de quem implementa ou desenvolve;
- A informação pode estar codificada em campos diferentes daqueles que seriam esperados;
- Ocorrência da mesma informação em diversos campos e/ou segmentos.

Desafios

A flexibilidade que permite que diferentes ambientes utilizem diferentes versões e aspectos do HL7 podem também servir de interface entre sistemas não relacionados tornando a sua implementação extremamente difícil.

Muitas das vezes a interpretação da norma varia de pessoa para pessoa, no entanto a sua estrutura, o facto de se apoiar num formato universalmente aceite como XML, permite que a informação contida num relatório médico no formato HL7 CDA seja facilmente tratada por qualquer sistema ou mesmo lido por humanos. Contudo, o CDA não especifica a forma como o documento pode ser transportado, mas um relatório médico no formato CDA pode ser transportado usando mensagens HL7 quer na versão 2 ou 3, assim como por qualquer outro mecanismo de transporte como o DICOM, em anexos MIME de emails, http, entre outros.

Em termos de desafios para este trabalho, como irá ser apresentado mais à frente, o facto de um CDA poder ser adquirido, transportado através de diferentes protocolos (aceites e implementados universalmente) e o seu formato em XML, que é largamente aceite como formato standard para a troca de dados e correspondente semântica servindo de base, para a troca de informação, não oferece grande resistência na sua utilização ou transformação em outro formato e facilita o acesso directo à informação.

Capítulo 3

Integração Orientada a Serviços

3.1 SOA

Uma Arquitectura Orientada a Serviços, ou Service Oriented Architecture (SOA) como é denominada em inglês, ainda é considerada como uma nova abordagem, um novo método para implementação de soluções que estejam em acordo com as necessidades do negócio. Representa uma mudança na maneira como as novas aplicações são pensadas, desenhadas, desenvolvidas e integradas com outras aplicações. Para além disso facilita no desenvolvimento de aplicações empresariais como serviços de negócio modulares que podem ser facilmente integrados em outras aplicações ou reutilizados. Disponibilizando funções como serviços que até então poderiam estar fora do alcance ou reutilizando funcionalidades já desenvolvidas em outros âmbitos. Uma das principais vantagens apontadas para esta arquitectura está na redução da distância entre aquilo que as pessoas de negócio necessitam e o que é implementado, uma vez que, parte da actividade da recolha de requisitos, mapeamento de processos de negócio e requisitos técnicos para os processos de negócios chaves de uma organização estão a par com o desenvolvimento e os requisitos do negócio.

Actualmente, muitas das empresas consideram SOA e serviços Web como modelos de processos viáveis para abrangerem as necessidades de integração para desenvolver aplicações empresariais robustas. Podemos considerar como objectivo primário, quando se está a conceber uma aplicação empresarial, que esta será o elo de ligação entre diferentes serviços de forma a responder às necessidades particulares de um negócio reduzindo o custo e complexidade da implementação. Porém, existe um grande desafio na concepção de todo o sistema uma vez que a introdução de novos serviços ou modificação de serviços já existentes não deve afectar o sistema no seu funcionamento. Isto apenas pode ser conseguido quando existe um conjunto de processos de negócio que podem orquestrar o sistema, possibilitando que os serviços comuniquem entre si quando são necessários.

Antes da abordagem por um modelo baseado numa Arquitectura Orientada a Serviços, muitas aplicações empresariais encontravam barreiras no desenvolvimento dos seus processos de negócio, quer por razões técnicas quer pela definição do próprios processos, impedindo a utilização do processo em si como um serviço passível de ser universalmente partilhado com os parceiros de negócio ou clientes, por exemplo. Com o desenvolvimento das tecnologias Web e a sua utilização universal e a combinação de protocolos de comunicação abertos, ferramentas e infra-estruturas os processos de negócio puderam assumir um novo nível de interacção entre o serviço e os seus utilizadores. Estas tecnologias estavam em plena expansão, uma vez que os serviços Web eram agora acessíveis a uma maior comunidade e apoiados em standards abertos.

O desafio era agora, para além da mera criação, partilha de serviços, utilização dos protocolos abertos de comunicação, encontrar um ponto comum de orquestração destes serviços onde fosse possível armazenar os processos de negócio e efectuar a gestão dos processos actuais, reutilizar ou desenvolver novas aplicações numa Arquitectura Orientada a Serviços.

3.1.1 Serviços Web

Enquadramento

O desenvolvimento dos sistemas informáticos e de redes de computadores deu origem a um novo paradigma de computação distribuída. Este novo paradigma serviu de suporte para a uma primeira divisão de uma aplicação informática em duas camadas: cliente e servidor (Figura 8).

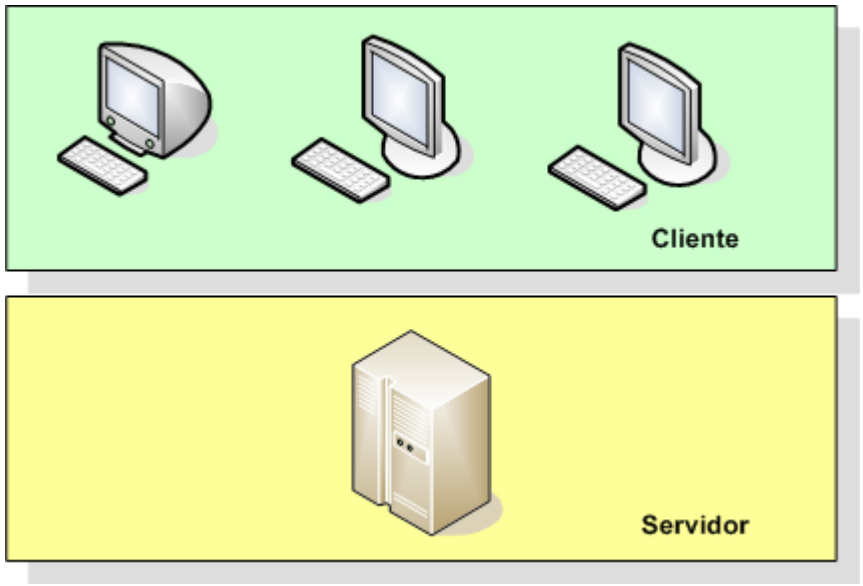


Figura 8: Arquitectura cliente-servidor (2 camadas)

Esta divisão da aplicação permite ao cliente efectuar o pedido para obtenção de informação, guardar informação, ou enviar dados para serem computados pelo servidor, por outro sistema. Esta descentralização minimiza problemas de estrangulamento de recursos distribuindo a carga computacional por vários sistemas e fornecendo flexibilização ao desenho da aplicação. No entanto, esta arquitectura de duas camadas também apresenta desvantagens, principalmente em termos de escalabilidade, heterogeneidade e de resistência a falhas, uma vez que existe um único ponto comum (Servidor) a todos os clientes que em caso de falha inibe o funcionamento da aplicação.

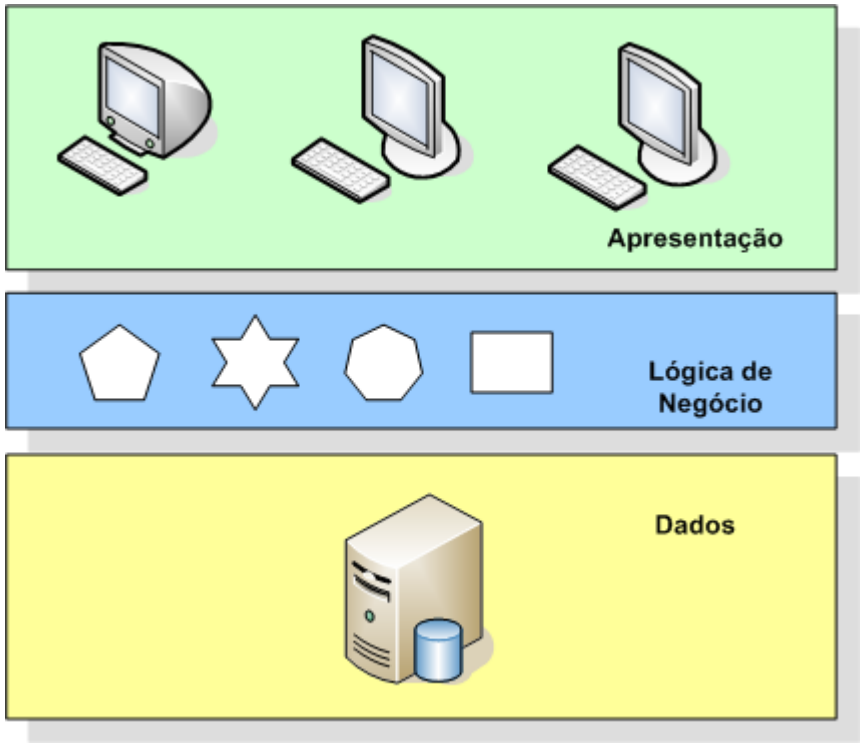


Figura 9: Arquitectura de 3 camadas

Como resposta a estas limitações foi introduzida uma arquitectura de três camadas (Figura 9), dividindo a camada de aplicação numa camada de apresentação, a intermédia contendo a lógica de negócio e a terceira responsável pelos dados. Esta nova arquitectura permite abstracção a alto nível das camadas inferiores, uma vez que entre elas existe agora uma camada intermédia que efectua a ponte entre as duas. Contribui para a escalabilidade do sistema uma vez que agora não existe uma ligação directa entre a camada superior e inferior permitindo acrescentar novos componentes, bastando actualizar a lógica de negócio. O tratamento de falhas já não é tão transparente para o utilizador uma vez que os pedidos da aplicação passam pela camada intermédia que é capaz de avaliar, através das regras de negócio, qual o tratamento do erro e se existe alguma solução, caso contrário é passado para a camada de apresentação para conhecimento do utilizador.

A necessidade de uma interacção distribuída, esconder do utilizador o modelo de dados, expor os serviços disponibilizados pela camada inferior independentemente da solução ou arquitectura implementada originou a criação de interfaces para os seus métodos (API), bem como de

chamadas de procedimento remoto (em inglês Remote Procedure Call (RPC)) que permitem executar procedimentos em outros sistemas remotos e de mediadores orientados a objectos conhecidos, como o CORBA, DCOM ou RMI, que permitem correr múltiplos processos em diferentes máquinas numa rede de computadores. Apesar das vantagens apresentadas por estes mediadores entre a camada de apresentação e de dados, e os mesmos não são os mais qualificados para serviços Web. Na World Wide Web existem sistemas heterogéneos quer do lado do servidor quer do lado do cliente, e não é conhecido de antemão que tipo de mediador é usado de cada lado para comunicação assim como a nível de infra-estrutura e de segurança. Como tal, é necessário uma nova abordagem para os serviços disponibilizados via Web.

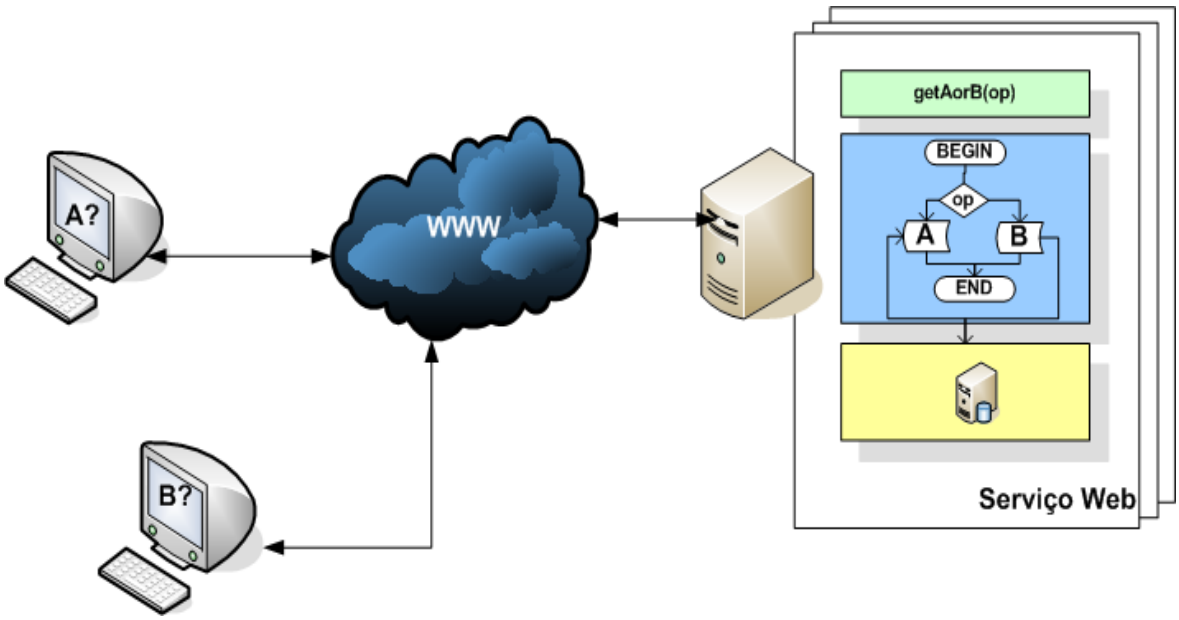


Figura 10: Serviço Web

Definição

Os serviços Web (Figura 10) podem ser considerados como aplicações Web que podem ser disponibilizadas e invocadas através da World Wide Web expondo os seus serviços de uma forma transparente e publica sem exporem necessariamente a sua lógica de negócio ou o seu modelo de dados. Existem diversas aplicações baseadas em serviços Web que informam sobre o mercado

bolsista, permitem marcação de viagens ou aluguer de viaturas bem como de hotéis. É até possível existir um serviço Web que utiliza outros serviços para poder responder aos pedidos que recebeu.
Segundo o W3C [Web 27], um serviço Web é uma aplicação informática concebida para suportar a interoperabilidade numa relação máquina para máquina numa rede. Possui uma interface descrita num formato processado por máquinas, neste caso específico, o WSDL. Outros sistemas interagem com o serviço Web de acordo com a sua descrição através de mensagens SOAP, sobre um protocolo de comunicação como http e serializadas em XML em conjunção com outras normas relacionadas com a Web. Uma outra definição comum que podemos encontrar na internet é que os serviços Web são funções invocadas encapsuladas e loosley coupled:

- Encapsulada – a implementação da função não é transparente ao utilizador;
- Loosely Coupled – mudar a implementação de uma função não implica mudar a forma como é invocada e exposta a sua funcionalidade;
- Invocada – a descrição do comportamento da função é disponibilizada publicamente, bem como a forma de ligação e parâmetros de entrada e saída;

Arquitectura

A suportar a definição de serviços Web existem tecnologias standard que são o padrão para a sua construção como a Web Services Description Language (WSDL), Simple Object Acess Protocol (SOAP) e Universal Description Discovery and Integration (UDDI). Tendo como referência estas tecnologias podemos ilustrar a arquitectura de um serviço Web através da Figura 11.

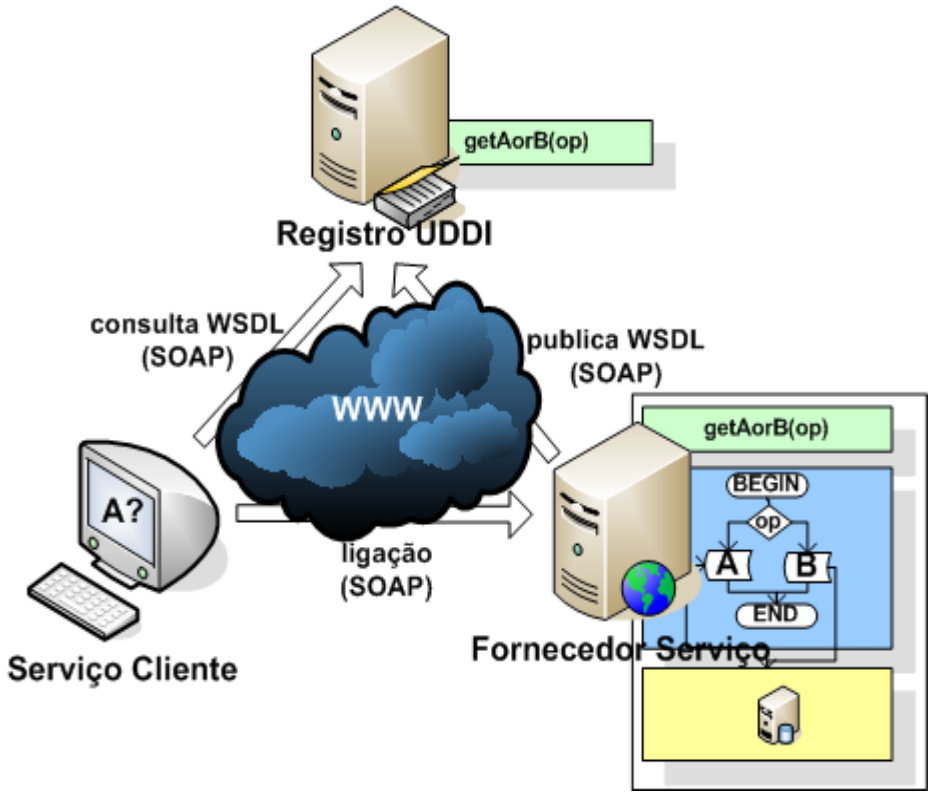


Figura 11: Visão global de serviço Web.

A arquitectura de um serviço Web pode ser descrita nos seguintes passos:

1. O Fornecedor de Serviço descreve o serviço Web num documento WSDL e publica-o para um registo através da sua API de publicação baseada em SOAP;
2. O Cliente do Serviço através da API de procura do UDDI pesquisa pelo registo de um fornecedor de serviço que se enquadre nos seus requisitos;
3. É criado um pedido SOAP de acordo com o documento WSDL;
4. O pedido SOAP será enviado ao Fornecedor do Serviço que processará o pedido;

Uma vez que esta arquitectura é orientada à internet o principal meio de transmissão de informação entre os agentes intervenientes é através de mensagens em XML sobre o protocolo HTTP, ultrapassando as limitações de comunicações dos mediadores apresentados anteriormente. O XML é largamente aceite como formato standard para a troca de dados e correspondente semântica

servindo de base, para a troca de informação, entre todas as camadas conceptuais que constituem um serviço Web (Figura 12).

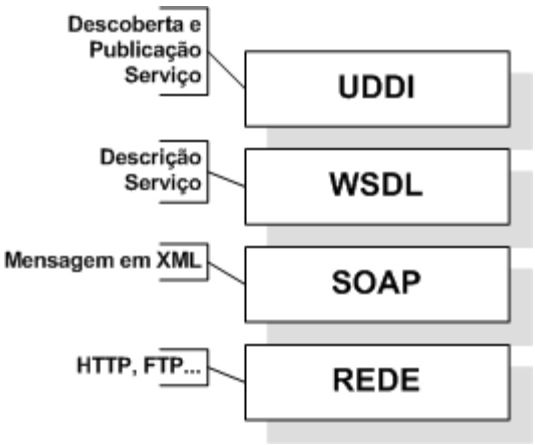


Figura 12: Camadas conceptuais dos serviços Web.

WSDL

A toma de conhecimento da existência de um serviço necessita de certo modo que este seja exposto de uma forma pública, acessível e que todos entendam qual o serviço disponibilizado. Para tal foi criada uma linguagem WSDL que descreve o serviço como um conjunto de todas as operações que podem ser invocadas, representadas por métodos que permitem descrever de uma forma clara, concisa e independente da implementação o serviço disponibilizado, possibilitando a comunicação entre os intervenientes [Hansen, Roseli Persson et al 2001].

A descrição de um serviço, segundo [Hansen, Roseli Persson et al 2001] pode ser dividida da seguinte forma:

- Implementação do Serviço
  - WSDL:service – para além da definição do serviço contêm um conjunto de elementos WSDL:port com um elemento WSDL:binding, assim como definições de extensibilidade;
  - WSDL:port – combinação de um WSDL:binding com um endereço de rede responsável pela interacção com o serviço;

- Interface do Serviço
    - WSDL:binding – descrição de protocolos, segurança, formato da datas, entre outras atributos para uma interface portType em particular;
    - WSDL:portType – interface entre o serviço, as operações disponibilizadas e os clientes;
    - WSDL:message – para além da definição dos parâmetros de entrada e saída pode também conter um documento inteiro ou argumentos a serem mapeados na invocação dos métodos;
    - WSDL:type – define tipos de dados complexos numa mensagem;

Para exemplificar a descrição de um serviço Web através de WSDL podemos aplicar à Figura 11 a seguinte descrição WSDL parcial:

```
<message name="getAorBRequest">
    <part name="op" type="xs:string"/>
</message>
<message name="getAorBResponse">
    <part name="valueAorB" type="xs:string"/>
</message>
<portType name="valueAorBLib">
    <operation name="getAorB">
        <input message="getAorBRequest"/>
        <output message="getAorBResponse"/>
    </operation>
</portType>
```

A descrição WSDL começa com um elemento message que define as partes de uma mensagem e o tipo de dados que lhe estão associados, ou por outras palavras, define os elementos de dados de uma operação. Como se pode observar neste exemplo, temos duas definições de mensagens. Uma mensagem getAorBRequest que contém um elemento denominado op, sendo do tipo string e uma

outra nomeada getAorBResponse que contém um elemento chamado valueAorB, sendo do tipo string. Estabelecendo uma comparação entre este tipo de descrição de mensagem e a abordagem tradicional de programação podemos dizer que a mensagem getAorBRequest corresponde a uma função que tem como parâmetro o op e a mensagem getAorBResponse corresponde a uma função com o parâmetro valueAorB. Como foi definido acima, o elemento portType é responsável pela definição do serviço Web, enquanto serviço, das operações que podem ser executadas e as respectivas mensagens. O elemento valueAorBLib que é definido como um elemento de operação do serviço, para a qual a operação é denominada de getAorB tem como mensagem de entrada getAorBRequest e uma mensagem resultado getAorBResponse. Usando a analogia com forma tradicional de programar, o elemento valueAorBLib pode ser considerado como a biblioteca de funções e a operação getAorB seria uma função com parâmetro de entrada getAorBRequest e como resultado getAorBReponse.

SOAP

Uma vez que os serviços Web estão disponíveis em sistema heterogéneos seria necessário que o protocolo de comunicação usado para efectuar transferências de dados ente funções fosse independente dos sistemas em que os serviços Web fossem executados. Desta forma foi criado o protocolo SOAP (Simple Object Acess Protocol) que permite a comunicação entre diversos sistemas e aplicações num ambiente distribuído e descentralizado.

O protocolo de comunicação SOAP não define, por si só, nenhuma semântica relacionada com a aplicação, isto é, com o modelo de programação ou a implementação. Define sim, um mecanismo simples de expressar a semântica da aplicação fornecendo um modelo de empacotamento modular e mecanismos de codificação para codificar a informação dentro dos módulos. Dado que o formato das mensagens é suportado por XML e sendo reconhecido por diversas plataformas pode transportar informação sem ser necessário recorrer ao uso de intermediários para codificação e descodificação da informação transmitida. Como foi apresentado e ilustrado pela Figura 11 estes protocolo é utilizado para publicar, localizar e invocar serviços Web, suportando RPC e podendo mesmo comunicar com outros protocolos de rede como o HTTP e SMTP.

De acordo com [Hansen, Roseli Persson et al 2001] um pacote SOAP é constituído por quatro partes:

- Envelope – contém o conteúdo da mensagem, destinatário, prioridade (encapsula elementos sintácticos da mensagem);
- Codificação – definição de mecanismos de serialização;
- RPC – especificação de encapsulamento chamadas remotas a métodos e respostas dentro da mensagem;
- Framework de ligação e transporte – definição abstracta para troca de envelopes SOAP entre aplicações.

De referir, que as mensagens podem conter os tipos de dados mais utilizados uma vez que possuiem a definição de tipos de dados como string, integer, float, double e date [Hansen, Roseli Persson et al 2001].

UDDI

A norma Universal Description Discovery and Integration (UDDI) foi pensada para fornecer um directório de procura para processos de negócio e os seus serviços Web de uma forma normalizada. Como tal, representa o intermediário de serviço que expõe os seus serviços de forma aberta e normalizada para que sejam possíveis de serem requisitados por alguma entidade externa ao serviço. O componente central do UDDI é denominado de UDDI Project, que manipula um registo global exposto de forma pública chamado de business registry. De alguma forma, podemos estabelecer uma analogia entre a forma como o UDDI foi concebido e uma lista telefónica. O business registry consiste em três componentes:

- Páginas amarelas (taxonomia) – a pesquisa pode ser feita para procurar negócios locais que fornece um determinado serviço ou um determinado produto ou uma empresa que se situa numa região geográfica específica;

- Páginas brancas – informação relativa ao serviço fornecido, incluindo endereço e identificadores conhecidos;
- Páginas verdes – informação técnica acerca do serviço Web que é exposto pelo negócio, por exemplo, como se estabelece contacto com o serviço Web.

Em suma, implementar a norma UDDI é criar um servidor de registos que fornece um mecanismo para publicar e localizar serviços, que armazena informações categorizadas sobre empresas, serviços disponibilizadas por elas e associação com as especificações desses serviços, feitas em WSDL através do próprio registo [Hansen, Roseli Persson et al 2001].

3.1.2 Orquestração vs Coreografia

Orquestração e Coreografia, num contexto SOA, referem-se à utilização de processos que abrangem múltiplos participantes, com mensagens a serem trocadas em todas as direcções em acordo com um conjunto complexo de regras. A coreografia e orquestração são duas abordagens diferentes para a coordenação ou controlo de toda esta actividade. Estas duas aproximações são estratégias desenvolvidas para resolver o problema da interacção e coordenação de vários serviços, colocando rigor na forma como a troca das mensagens é representada e pela organização do processo geral através da utilização do conjunto adequado de padrões de controlo de fluxo.

Por convenção, coreografia descreve o protocolo global de controlo individual dos intervenientes e da forma como estes interagem uns com os outros. Cada interveniente tem o seu próprio processo, mas a coreografia é o processo principal e age como sendo uma espécie de controlador de tráfego ainda que significativamente este não corra actualmente. Não se trata de um mediador na troca em tempo real de mensagens, mas meramente um protocolo para a troca de mensagens, analogamente a um coreografo que ensina os intervenientes de um processo a dançar em grupo.

O processo para cada participante é definido como uma processo de orquestração, cuja principal tarefa é construir um fluxo de controlo ao redor das interacções entre intervenientes. O processo de orquestração é difícil de modelar, especialmente nos quais existem complexas combinações de eventos de entrada. Se o processo está sujeito a coreografia, a sua estrutura pode ser derivada do

processo geral de coreografia, existindo ferramentas que podem gerar esqueletos de processo de orquestração a partir de definições de coreografia. O processo de coreografia conta-nos a história completa, para que o interveniente (participante) consiga determinar o seu papel isolando as partes em que este está envolvido.

Um processo de orquestração contém actividades públicas e privadas. As actividades públicas são as que requerem coreografia. As actividades privadas respondem a requisitos internos mas que não são visíveis aos intervenientes.

Com o estabelecimento de um conjunto de normas tecnológicas para os serviços Web como o SOAP, XML, WSDL, é possível fornecer os meios para descrever, localizar e invocar um serviço Web como uma entidade. Um serviço Web pode expor muitos métodos e cada WSDL descreve de uma forma atómica e a baixo nível as funções disponibilizadas por esse serviço. No entanto, as tecnologias base dos serviços Web não nos permitem ter um detalhe do comportamento que descreva o papel desempenhado por cada serviço quando interage com o todo como sistema complexo de colaboração. Quando estas colaborações entre serviços são conjunto de actividades desenhadas para atingir uma meta de negócio, são conhecidas como processos de negócio e podem alargarem-se a várias organizações.

A orquestração, para o propósito deste trabalho, pode ser definida como um mecanismos standard de definição de trabalho em conjunto dos serviços Web, incluindo a lógica de negócio, sequenciação, tratamento de erros, decomposição de processos e a reutilização de serviços e processos. De referir ainda que é possível definir processos de longa duração, paralelismo, processos fracamente ligados entre si, e é ainda possível defini-los de forma síncrona ou assíncrona. Estas características da orquestração levam à consideração que a orquestração de serviços Web é na realidade uma outra camada para além das tradicionais aproximações de aplicações de integração.

Em suma, a orquestração é necessária quando estamos a construir uma arquitectura orientada a serviços para ser aplicada dentro de uma organização ou para ligar uma organização externamente. A orquestração é a abordagem que cria soluções de negócio de uma variedade de serviços e fluxos de informação encontrados em novos sistemas ou em sistemas já existentes. Trata-se da melhor

abordagem como mecanismo para colocar uma arquitectura orientada a serviços a funcionar, bem como exercer controlo sobre a mesma. As camadas de orquestração permitem mudar a forma como o negócio funciona, conforme as necessidades, sem dessincronizar os restantes serviços ao contrário da coreografia, sendo possível definir ou redefinir qualquer processo de negócio que esteja a correr no momento.

3.1.3 BPEL

Nos últimos anos, Business Process Execution Language for Web Services (WS-BPEL ou apenas BPEL) emergiu como outra linguagem standard de composição de serviços Web e implementada por um amplo espectro de empresas fornecedoras de serviços [Jordan & Evdemon 2007].

Quando é necessário alterar ou gerir regras de negócio embebidas nos processos torna-se mais difícil, mais dispendioso, consome tempo e não pode ser eventualmente feito por um analista de negócio sem qualquer experiência em programação [Rosenberg, Florian, Dustdar & Schahram 2005].

Acrescido a isto, desenhar um sistema completo e introduzir um novo serviço ou modificar serviços existentes é um desafio que não pode afectar o sistema numa grande extensão. De acordo com [Jennings & Salter 2008] isto apenas é conseguido quando existe um conjunto de processos de negócios que podem orquestrar o sistema, fazendo com que os serviços comuniquem uns com os outros no momento certo.

BPEL é uma linguagem de execução de processos de negócio que também descreve dentro de uma Arquitectura Orientada a Serviços, a forma como uma empresa desempenha os seus processos de negócio, tornando desta forma acessível a parceiros de negócio e clientes a participação directa nos processos de negócio da empresa com o intuito de os melhorar. O BPEL está numa camada superior e estende modelo de serviço WSDL (Web Services Description Language) que define as operações específicas permitidas e como as operações WSDL são orquestradas para que possam satisfazer os processos de negócio [Web 15].

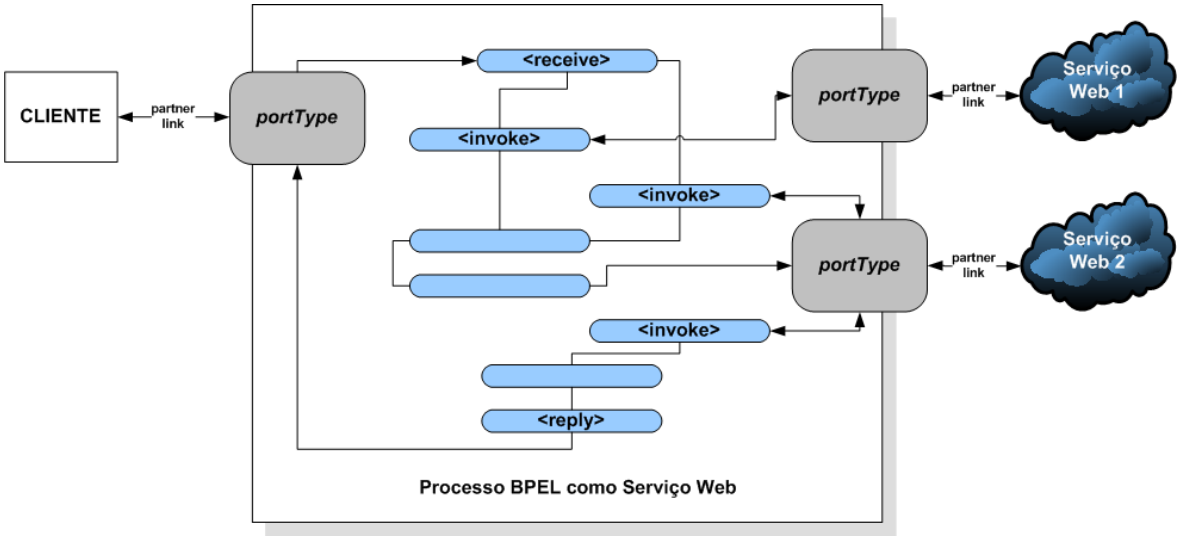


Figura 13: Abstracção de um possível processo BPEL (adaptado de [Web 16])

A Figura 13 exemplifica um possível processo BPEL que inicia com uma actividade de receive e em seguida invoca um serviço externo e termina respondendo ao cliente. Tipicamente, um processo BPEL interage com um ou mais serviços Web externos. O próprio processo BPEL, em si, também é um serviço Web.

O WS-BPEL define um modelo e uma gramática para descrever o comportamento de um processo de negócio baseado nas interacções entre processos e os seus parceiros. A interacção com cada parceiro ocorre através de interfaces de serviços Web, e a estrutura do relacionamento ao nível da interface é encapsulado no que se denomina um partnerLink. Um processo BPEL define como múltiplas interacções de serviços com estes parceiros são coordenadas para alcançar um objectivo de negócio, assim como o estado, lógica necessária para essa coordenação, mecanismos sistemáticos para lidar com excepções e processamento de faltas, entre outras [Web 14].

O BPEL utiliza várias especificações em XML como o WSDL, XML Schema, XPath e XSLT. Mensagens WSDL e tipos definidos por XML Schema fornecem o modelo de dados usado pelos processos BPEL. O XPath e o XSLT fornecem o suporte para a manipulação de dados. Todos os recursos externos e parceiros são representados por serviços WSDL [Web 14].

De acordo com [Jennings & Salter 2008] o BPEL está a emergir como um claro standard para a composição de múltiplos serviços síncronos e assíncronos em fluxos de processos colaborativos e transaccionais.

No entanto, os sistemas de integração actuais requerem mais do que a possibilidade da simples condução da interacção utilizando protocolos normalizados. A potencialidade dos serviços Web como suporte para uma plataforma de integração só será alcançada quando as aplicações e processos de negócio sejam capazes de integrar as suas interacções complexas usando um modelo de processos de integração normalizado [Web 14].

3.2 JBI

Os problemas mais comuns identificados em aplicações de integração no mundo empresarial são os protocolos e formatos de mensagens incompatíveis. Como resposta a este tipo de problemas, a actual opção que a indústria tem seguido baseia-se nas definições de normas para integração de processos de negócio e dados normalizados na pilha de serviços Web [Web 11].

Integrar identidades computacionais usando apenas interacções entre serviços numa Arquitectura Orientada a Serviços denomina-se Integração Orientada a Serviços (SOI). Soluções de Integração Orientada a Serviços lidam melhor com a integração de problemas criados por sistemas antigos e sistemas heterogéneos inflexíveis usando com maior incidência funcionalidades que não são visíveis, em diferentes aplicações, como serviços reutilizáveis. A principal vantagem, em relação à tradicional integração de aplicações empresariais (EAI) é a aplicação de normas para definição de interface normalizadas, funcionalidades opacas que estão escondidas da interface de serviço e a flexibilidade do serviço na perspectiva do consumidor e produtor que pode mudar, excepto a descrição do serviço.

A especificação Java Business Integration (JBI) 1.0, JSR-208 [Web 8], é uma ampla especificação, de iniciativa industrial, para criar uma plataforma de integração normalizada para Java e aplicações de negócio numa Arquitectura Orientada a Serviços (SOA). A especificação JBI emprega conceitos similares a J2EE para estender o empacotamento de aplicações e

disponibilização de funcionalidades com componentes JBI incluídos. Os componentes JBI são classes abertas a ligações de componentes que são baseadas em processos de negócio abstractos JBI de meta-informação [Web 8]. A JSR 208 especifica dois tipos de arquivos (ficheiros .jar) de entrega no ambiente destino denominados Unit e Service Assembly. Estes arquivos contêm código e descrições standards análogos ao empacotamento WAR e EAR do JAVA EE 5.

Uma vez instalados os componentes JBI, num ambiente JBI (Figura 14), estes interagem entre eles por via de um processo de troca de mensagens, que é descrito usando documentos WSDL, publicados pelo fornecedor do serviço. Esta descrição de serviço é a única fonte de informação necessária para os componentes consumidores interagirem com os fornecedores de serviço. JBI fornece uma infra-estrutura leve de mensagens, conhecida como encaminhador de mensagens normalizadas, que fornece o mecanismo para actual troca de mensagens num formato independente (XML), onde as interfaces de integração são desenvolvidas com assumpções mínimas entre as partes responsáveis pelo envio e recepção da mensagem, reduzindo o risco que uma mudança em uma aplicação ou módulo force uma mudança em outra aplicação ou módulo, usando sempre uma implementação JBI como intermediária.

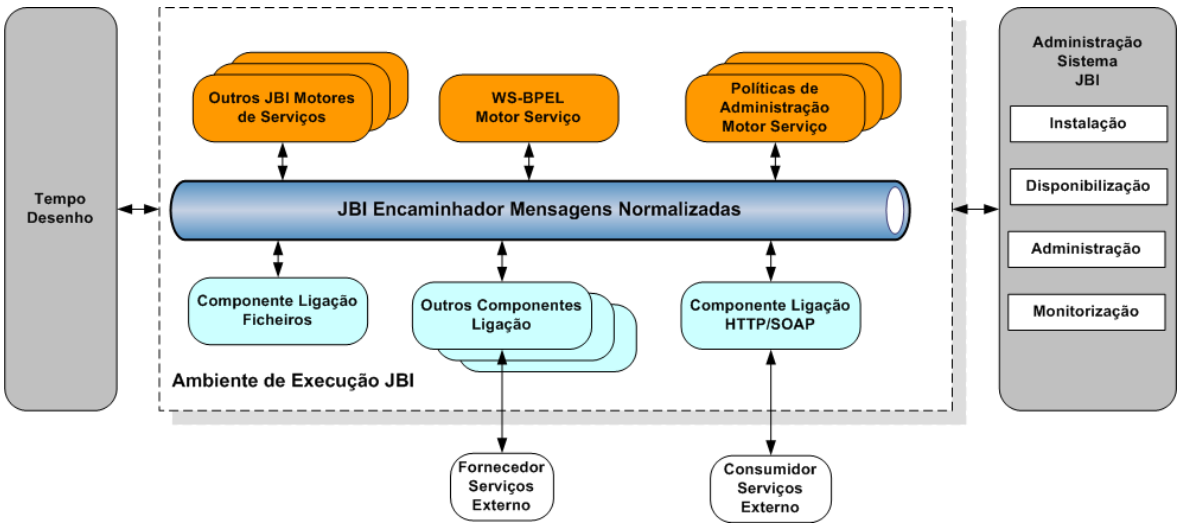


Figura 14: Arquitectura JBI (adaptado de [Web 12]).

De acordo com [Web 12] as peças fundamentais de um ambiente JBI são:

- Motores de Serviço – são componentes JBI que permitem ligações de lógica de negócio;
- Componentes de Ligação – permite conectividade com o exterior;
- Encaminhador de Mensagens Normalizadas – normaliza mensagens directamente dos componentes de fonte para destinos específicos de acordo com a política de encaminhamento;

O ambiente disponibilizado pela especificação JBI pode hospedar múltiplos Motores de Serviço e Componentes de Ligação. Quando se envia e recebe mensagens de fora do ambiente JBI, os motores de serviço comunicam usando o Encaminhador Normalizado de Mensagens e passam a mensagem para o cliente através de um Componente de Ligação. Quando a comunicação é inteiramente dentro do ambiente JBI, não é necessário nenhum protocolo de conversão, ou de serialização de mensagens, ou normalização de mensagens uma vez que todas as mensagens já se encontrão normalizadas e no formato standard WSDL 2.0.

A principal ideia desta especificação é construir um ambiente onde qualquer fornecedor de componentes de integração e infra-estrutura estejam em concordância com a especificação JBI. Portanto, esta especificação fornece uma forma de integrar serviços hospedados num ambiente gerido permitindo interacção entre motores de serviço acopláveis de diferentes fornecedores. Estes motores e os correspondentes serviços podem ser qualquer coisa que forneça integração e gestão de processos. Motores de Serviço podem ser motores de serviço XSLT, um serviço CBR ou um motor de orquestração WS-BPEL. Estes serviços podem ser fornecidos por diferentes fornecedores, e ainda poderão trabalhar juntos no mesmo ambiente [Web 8].

3.2.1 Motores de Serviço

A arquitectura JBI tal como é ilustrada pela Figura 14 consiste num meta-container, ou ambiente de execução, que contém e executa Motores de Serviços, que por sua vez hospedam Unidades de

Serviço. Os Motores de Serviço fornecem serviços, como de lógica de negócio, processamento, transformação, encaminhamento, entre outros.

Por exemplo, podemos utilizar um Motor de Serviço compatível com WS-BPEL, para orquestrar processos de negócio que usem WS-BPEL. De referir ainda que a especificação JBI não esqueceu as aplicações já existentes e muitas delas usando tecnologia disponibilizada pela versão Java EE, como os EJB ou servlets, podem interagir numa arquitectura JBI usando um o Motor de Serviço Java EE. Como exemplo, em [Web 28] podemos encontrar um exemplo do seu funcionamento, passo a passo, no caso de uma invocação por parte de um serviço Web Java EE a um serviço JBI, ilustrado pela Figura 15:

1. Inicia-se uma transacção numa aplicação Java EE;
2. Ocorre a invocação de um serviço disponibilizado no ambiente JBI;
3. O Motor de Serviço Java EE suspende a transacção associada com o pedido;
4. A transacção suspendida é colocada no conjunto de mensagens para ser expedida no Encaminhador Normalizado de Mensagens (NMR);
5. O serviço disponibilizado pelo componente JBI é invocado no mesmo contexto transaccional (se ocorrer uma falha o componente JBI retorna ao estado anterior e a transacção fica sem efeito);
6. O componente JBI retorna a resposta após definir a transacção no intercâmbio de mensagens;
7. O Motor de Serviço Java EE recebe a mensagem do Encaminhador Normalizado de Mensagens;
8. A transacção é concluída depois de retornar a transacção da mensagem recebida;
9. A resposta é devolvida à aplicação Java EE.

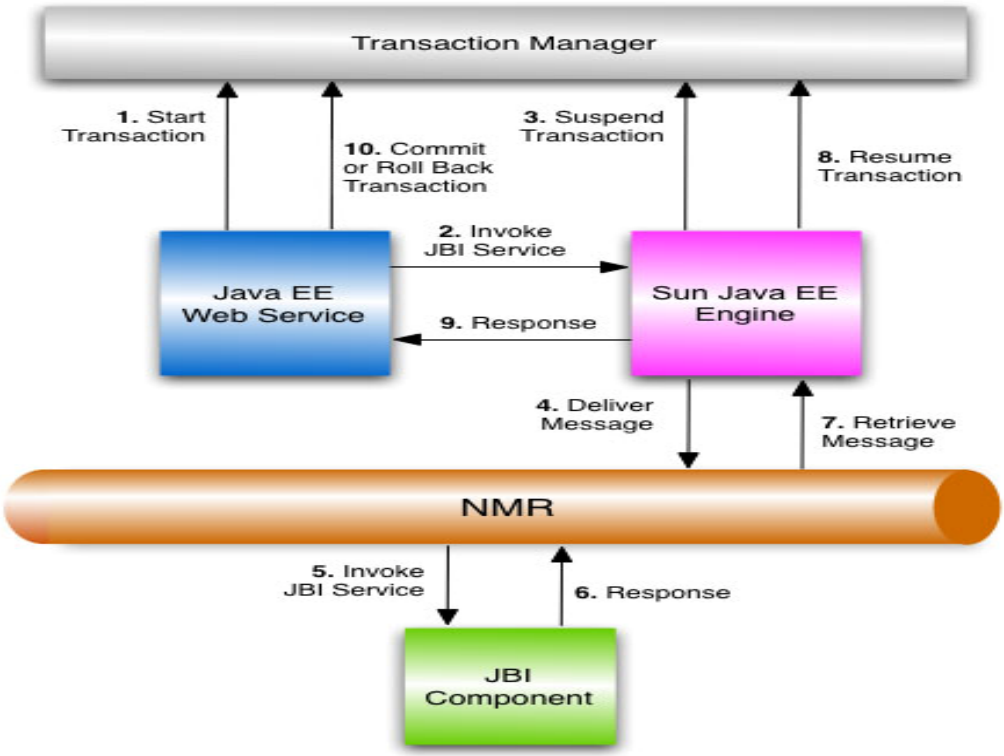


Figura 15: Exemplo de invocação de um serviço JBI por parte de um serviço Java EE (retirado de [Web 28])

3.2.2 Componentes de Ligação

Num ambiente JBI, a independência de protocolo é suportada pelos Componentes de Ligação. Estes componentes fornecem protocolos de transporte e comunicação, o acesso a serviços remotos a partir do ambiente assim como serviços dentro do próprio ambiente. Estes componentes de ligação protocolares actuam também como proxies para serviços no ambiente JBI para acesso a fornecedores de serviços que requerem um protocolo específico de comunicação.

Os componentes de ligação são construídos para cada protocolo externo e ligados ao ambiente JBI. Esta arquitectura permite que qualquer componente JBI possa comunicar sobre qualquer protocolo de comunicação ou transporte (SOAP, JMS, ...) desde que exista uma ligação que trate em particular estes protocolos dentro do ambiente JBI. Os Componentes de Ligação suportam também a interoperabilidade entre serviço usando protocolos como SOAP, Simple Mail Transfer Protocol

(SMTP), Java Message Services (JMS), entre outros. A utilização destes componentes não necessita que se implementem protocolos específicos nas regras de negócio e permite independência da implementação do serviço a partir do mecanismo de acesso.

Em suma, os Componentes de Ligação consomem mensagens de dados de protocolos específicos, convertendo as mensagens em mensagens normalizadas de acordo com a especificação JBI e direccionam as mensagens normalizadas para o Encaminhador de Mensagens Normalizadas para serem consumidas por qualquer Motor de Serviço. Da mesma forma, os Componentes de Ligação retiram as mensagens do Encaminhador de Mensagens Normalizadas e efectuam o processo inverso ao da normalização, transformando a mensagem normalizada numa mensagem de um protocolo específico e envia de volta para o consumidor final da informação.

3.2.3 Encaminhador Normalizado de Mensagens

O Encaminhador Normalizado de Mensagens é a principal razão das intercooperações entre os componentes JBI, uma vez que fornece a estrutura abstracta de mediação para a troca de mensagens. O Encaminhador Normalizado de Mensagens recebe as mensagens vindas de todos os componentes de ligação e motores de serviço e encaminha-as para o componente JBI apropriado para serem processadas. A troca de mensagens entre o encaminhador e os componentes é efectuada através de mensagens normalizadas em concordância com a especificação JBI que fornece a interoperabilidade entre componentes como ilustrado na Figura 16.

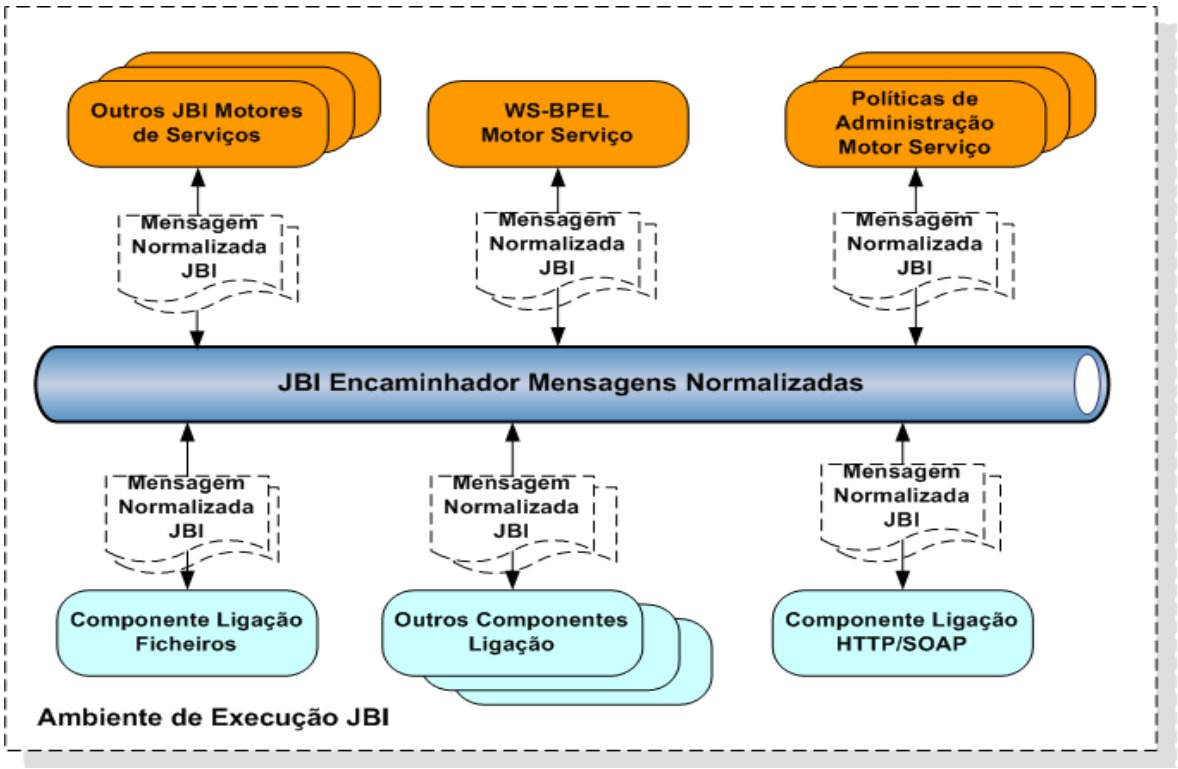


Figura 16: Troca de mensagens entre componentes JBI (adaptado de [Web 12])

As mensagens que passam pelo Encaminhador de Mensagens Normalizadas são mensagens normalizadas de acordo com a especificação JBI. Uma vez normalizadas as mensagens são encaminhadas para os seus destinatários, recebidas ou enviadas pelo Encaminhador que entende o formato e fornece o transporte, contexto transaccional e de segurança entre os Componentes de Ligação e Motores de Serviço.

Mensagens Normalizadas JBI

Uma mensagem normalizada JBI é um documento XML consistindo em duas partes:

1. Mensagem de meta dados – esta mensagem também é conhecida como contexto de dados e inclui informação de contexto como:
    a. Protocolo fornecido para o contexto de informação;
    b. Tokens de segurança;

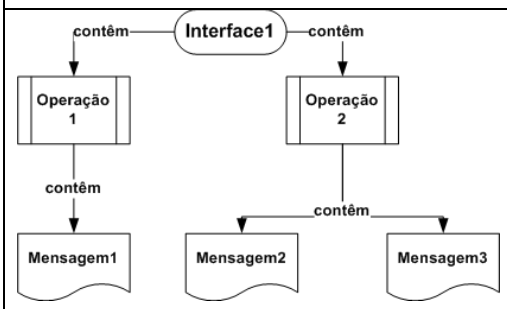
c. Contexto de informação transaccional;

d. Informação específica para outros componentes;

2. Corpo da Mensagem – o corpo da mensagem é uma fonte de abstracção que contêm todos os dados da mensagem.

O corpo da mensagem está em concordância com uma mensagem do tipo WSDL abstracto sem codificação de protocolo ou formatação como é exemplificado na Tabela 1.

Tabela 1: Modelo abstracto de mensagens WSDL 2.0 (adaptado de [Web 12])

Modelo Abstracto Mensagens WSDL	Mensagem WSDL
	<interface name="IGetAorB"> <operation name="getAorB"> <input name="op" message="tns:requestAorBMessage"> </input> <output name="valueAorB" message="tns:requestAorBResponseMessage"> </output> </operation> </interface>

Canal de Entrega

O Canal de Entrega é um canal de comunicação bidireccional usado por todos os componentes JBI para comunicação com o Encaminhador de Mensagens Normalizadas. Um serviço consumidor usa o canal de entrega para iniciar invocações de serviços assim como um fornecedor de serviço usa o canal de entrega para receber tais invocações, como é ilustrado pela Figura 17.

Qualquer componente JBI pode agir como um serviço consumidor bem como um fornecedor de serviço usando o Canal de Entrega para ambos os casos.

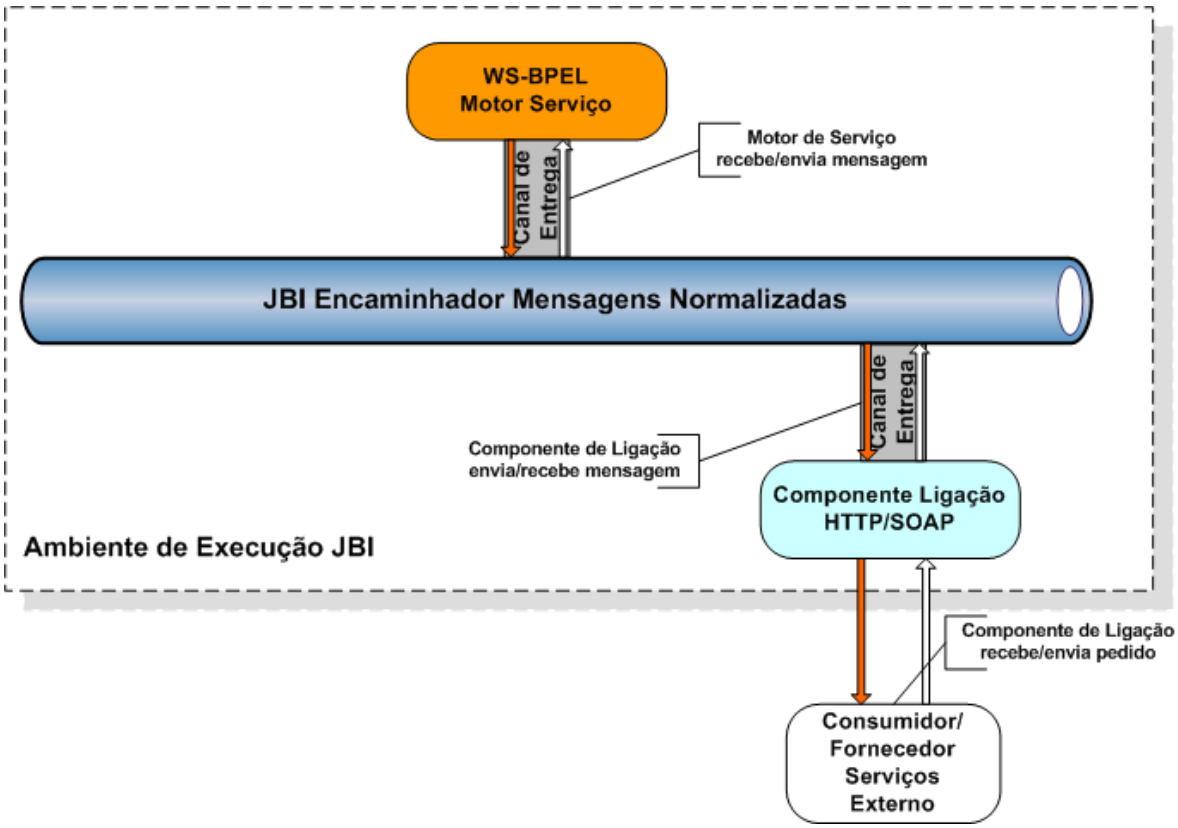


Figura 17: Exemplificação do uso do Canal de Entrega (adaptado de [Web 12])

3.3 Open-ESB

Nos últimos anos tem ocorrido um acréscimo significativo de tecnologias orientadas a Service-Oriented Architecture (SOA), Enterprise Application Integration (EAI), Business to Business (B2B) e serviços Web. Estas tecnologias tentam responder aos desafios de melhorar os resultados e gerar valor acrescido para os processos de negócios em que intervêm e reuniram sobre si a atenção não só de especialista da área informática como da área de negócio. O Enterprise Service Bus (ESB) combina o melhor destas e doutras tecnologias com o objectivo de facilitar a implementação de uma Arquitectura Orientada a Serviços [Chappel 2004].

Os princípios aplicados no ESB representam uma nova perspectiva na integração de componentes com pouca dependência entre si e altamente distribuídos que pode ir para além de uma aplicação

de mediação numa normal estratégia de integração. Um sistema de ESB é uma plataforma normalizada de integração para mensagens, serviços Web, transformação de dados, entre outros. Possibilita ainda a ligação, coordenação das interacções de um grande número de diferentes aplicações através das aplicações empresarias garantindo a integridade a nível transaccional.

O ESB permite uma nova forma de incorporar serviços e uma arquitectura orientada a serviço numa arquitectura robusta que integra serviços e aplicações num ponto central, que estabelece a ponte com as aplicações e os serviços disponibilizados internos ou externos a uma organização. Permite o uso imediato de serviços Web e outras tecnologias de integração com as tecnologias modernas actuais. Num ESB podem ser aplicadas muitas tecnologias que pertencem ao universo Java como as Java Message Service (JMS), Java Component Application (JCA), Enterprise Java Bean (EJB), Java Server Pages (JSP), JAXB, JAX-RPC e Java Message Exchange (JMX).

O ESB é um conceito de uma plataforma independente que idealmente pode ser implementada sem ser necessária uma tecnologia em particular servindo de suporte a plataformas de integração como Open-ESB.

O Open-ESB trata-se de uma infra-estrutura e plataforma de trabalho de integração normalizada e descentralizada. Fornece encaminhamento e interface de mensagens normalizadas suportado pelo uso de mensagens assíncronas em XML assim como de gestão e monitorização centralizada, permitindo compor serviços Web e aplicações empresariais usando uma arquitectura orientada a serviços. Uma arquitectura orientada a serviços pode ser implementada com o Open ESB uma vez que a composição da aplicação pode ser composta e recomposta vezes sem conta.

A versão actual do Open-ESB é composta por diversos componentes e os principais para este trabalho são:

- Plataforma de trabalho JBI – esta plataforma implementa uma instância JBI;
- Motores de Serviço
    - BPEL – A Business Process Execution Language (BPEL) é utilizada para organizar os processos numa aplicação composta;

- Java EE – permite que serviços disponibilizados a partir, por exemplo, de EJB possam interagir com servidor hospedeiro da aplicação;
- XSLT – usado para efectuar transformações em documentos XML com a ajuda de XSL;
- Processador Inteligente de Eventos – usado para notificação e despertar eventos e gerir e processar em tempo real eventos de negócio;
- SQL – permite disponibilizar motor SQL a outros componentes JBI;

- Componentes de Ligação
  - Ficheiros (FileBC) – fornece um serviço de transporte para um sistema de ficheiros;
  - File Transfer Protocol (FTP) – este componente recebe mensagens usando o protocolo FTP;
  - Hypertext Transfer Protocol (HTTP) – é responsável por ligar uma instância JBI a serviços Web externos e por ligar esses serviços externos à instância JBI;
  - Java Database Connectivity (JDBC) – através desta componente é possível configurar e ligar a bases dados que suportam a API da especificação JDBC 3.0;
  - Java Message Service (JMS) – fornece suporte a Java Message Service (JMS) para o transporte de mensagens;
  - Simple Mail Transfer Protocol (SMTP) – possibilita a configuração e ligação a servidores e clientes SMTP dentro de um ambiente JBI;
  - Health Level 7 (HL7) – fornece uma solução abrangente para a configuração e conexão aos componentes JBI bem como a sistemas externos do ambiente JBI que utilizem o protocolo HL7;

De referir ainda que a escolha desta plataforma também se baseou no facto de estar em plena expansão, suportada por uma grande comunidade de especialistas, de diferentes áreas de conhecimento possuindo diversos componentes de diferentes áreas de aplicação. Dos seus componentes destaca-se o componente de ligação para HL7 que em outras plataformas semelhantes ainda não foi abordado ou ainda se encontra numa fase inicial. Uma outra vantagem

apresentada está no componente de ligação de ficheiros, para a normalização de ficheiros de texto, que permite definir regras para extracção da informação dos relatórios, definindo uma estratégia de parsing para os ficheiros utilizando o tipo de formato (template) usado por um médico.

3.4 JBI e Open-ESB

Java Business Integration (JBI) que é o suporte para criar uma arquitectura orientada a serviços está na base da plataforma Open-ESB. Como apresentado anteriormente, esta especificação permite a qualquer um criar um componente de integração, do tipo plug-in, em concordância com a especificação JBI, de integração dinâmica numa infra-estrutura JBI. Apesar disto, a especificação JBI por si só, não possui um único ponto comum de administração de todo o sistema e cada operação no sistema necessita de conhecimento prévio da topologia do sistema.

A plataforma Open-ESB resolve estes problemas utilizando o Java Open Entreprise Service Bus (ESB) construído com tecnologia JBI possibilitando que um conjunto distribuído de instâncias JBI possam comunicar como uma única identidade lógica que pode ser centralmente gerida. Através da plataforma Open-ESB é possível integrar funções de negócio existentes como serviços e desacoplar interacções entre os fornecedores de serviço e consumidores. A plataforma Open-ESB fornece suporte directo para criar aplicações compostas através do mecanismo JBI de montagem de serviços. Este suporte permite que as aplicações sejam compostas directamente do serviço base de interfaces disponibilizadas pelo JBI como serviços unitários e usando BPEL (Business Process Execution Language) para orquestração de processos [Web 12].

Este suporte directo de composição de aplicações construídas em cima de uma arquitectura orientada a serviços e o uso de uma infra-estrutura de mensagens normalizadas tornam a plataforma Open-ESB, que implementa a especificação JBI, o suporte ideal para construir aplicações orientadas a serviços ou de serviços de sistemas existentes usando mensagens normalizadas em XML [Web 11].

Capítulo 4

Descoberta de Conhecimento em Relatórios Médicos

4.1 Descoberta de Conhecimento

A descoberta de padrões desconhecidos e a previsão de novas tendências, entre outros aspectos da mineração de dados para a descoberta de conhecimento tem atraído atenções de diversas áreas entre elas de engenharia, empresarial e científica [Han & Kamber 2001].

Por exemplo numa aplicação na área da economia, quando os mercados bolsistas começam a ficar saturados, as margens começam a ficar mais pequenas e todos começam a jogar o chamado jogo da "soma nula". Isto significa que independentemente do que uma empresa ganha à custa de outra, o objectivo está em saber como é que pode conservar os seus clientes com um custo mínimo passando a ser o atributo chave essencial para as empresas vencedoras deste jogo [Berson & Stephen 1997].

A questão neste caso é saber quando é que se corre o risco do cliente abandonar a participação num negócio da empresa uma vez que este não vai informar antecipadamente que a vai abandonar. A única solução reside na análise dos dados do cliente, através do uso de ferramentas que possam construir modelos de previsão que possam antever quais os possíveis clientes que estão em risco

de abandonar. Uma vez obtido o modelo de previsão será possível lançar, por exemplo, uma campanha de marketing para fidelização dos clientes que estão a pensar em abandonar.

A mineração de dados funciona da mesma forma que um humano analisa os dados, aprendendo com experiências passadas e aplicando o conhecimento apreendido em outras situações. As ferramentas de mineração de dados são projectadas para perceberem e aprenderem qual a informação valiosa e quais os métodos a aplicar. Portanto, as ferramentas podem usar informação já adquirida como valiosa para encontrar exemplos similares que estejam escondidos na base de dados, usando a informação aprendida no passado para obtenção de um modelo de previsão do que pode acontecer no futuro.

A mineração de dados é um processo único diferente das normais operações em base de dados. Na maior parte das normais operações de base de dados, os resultados apresentados ao utilizador são de certa forma conhecidos ou já se sabia da sua existência na base de dados. A mineração de dados, por outro lado, extrai a informação da base de dados com o objectivo de dar a conhecer ao utilizador informação que até ao momento este não sabia da sua existência.

As relações entre variáveis e os comportamentos dos clientes que não são, de certa forma, intuitivas são a informação valiosa que através da mineração de dados se tenta captar. Uma vez que o utilizador não conhece antecipadamente o que é o que o processo de mineração de dados descobriu torna-se de certa forma uma maior dificuldade pegar nos resultados da mineração e converte-los numa solução para um problema de negócio [Thearling 2003].

De acordo com [Fayyad, Piatetsky-Shapiro & Smyth 1996] a descoberta de conhecimento numa base de dados, é um processo, que não é trivial, de identificar em dados padrões que sejam válidos, novos que até então eram desconhecidos, que acrescentem valor e sejam compreensíveis. Deste modo conseguimos perceber melhor os resultados obtidos, melhorando a forma como entendemos um problema ou um procedimento para a nossa tomada de decisão sobre a resolução do mesmo. Adaptando a análise destes termos, de uma forma individual, para uma melhor compreensão tal como está enunciada em [Silva 2004] temos:

- Dados – conjunto de factos como instâncias de uma base de dados. Por exemplo, uma colecção de dados demográficos;
- Padrão – trata-se de uma expressão Exp que ocorre numa linguagem L descrevendo factos em um subconjunto S_E de F. Um padrão é considerado como padrão se é mais simples do que a enumeração de todos os factos que ocorrem em S_E . Por exemplo, um padrão que pode ocorrer numa base de dados meteorológica pode ser: "Se neva então faz frio".
- Processo – é uma sequência de vários passos que envolve a preparação de dados, pesquisa de padrões, avaliação de conhecimento, refinamento envolvendo iterações e modificações;
- Validade – os padrões descobertos devem ser válidos para novos dados com um grau de certeza associado necessário para considerar a descoberta confiável;
- Novo - conhecimento que difere do existente;
- Potencialmente útil – os padrões encontrados devem levar a alguma decisão prática, conforme medido por alguma função de utilidade;
- Compreensível – um dos objectivos principais da descoberta de conhecimento em base de dados é tornar os padrões compreensíveis para as pessoas para que estas possam entender melhor os próprios dados.

4.2 Etapas do processo

O processo de descoberta de conhecimento é interactivo, iterativo, cognitivo e exploratório, envolvendo vários passos (Figura 18) com muitas das decisões a serem tomadas pela pessoa que é responsável pela análise dos dados, conforme descrito:

1. Definição do tipo de conhecimento a descobrir, o que implica conhecer e compreender antecipadamente o domínio da aplicação bem como o tipo de decisão que esse conhecimento pode contribuir para melhorar;

2. Criação de um conjunto de dados alvo (Selecção): seleccionar um conjunto de dados, ou focar num subconjunto, onde a descoberta deve ser realizada;
3. Limpeza de dados e pré-processamento (Pré-processamento): operações básicas que actuam nos dados, por exemplo, remoção de ruído, recolha de informação necessária para modelar ou estimar ruído, escolha de estratégias para manipular campos em que os dados não existem, formatação de dados de forma a adequá-los à ferramenta de mineração de dados;
4. Redução de dados e projecção (Transformação): encontrar características úteis para representação dos dados dependendo do objectivo da tarefa, com o intuito de reduzir o número de variáveis e/ou instâncias a serem consideradas para o conjunto de dados, assim como o enriquecimento semântico das informações;
5. Mineração de Dados: seleccionar os métodos mais adequados a serem usados para encontrar padrões nos dados, seguida da efectiva busca por padrões de interesse numa forma particular de representação. Nesta etapa podemos encontrar a busca pelo melhor ajuste dos parâmetros do algoritmo para a tarefa de mineração de dados em questão;
6. Interpretação e Avaliação: nesta etapa são interpretados e avaliados os padrões encontrados, com uma possível retorno para as etapas anteriores para posteriores iterações;
7. Implantação do conhecimento descoberto: incorporação do conhecimento descoberto à performance do sistema, documentação ou reportar os resultados às partes interessadas.

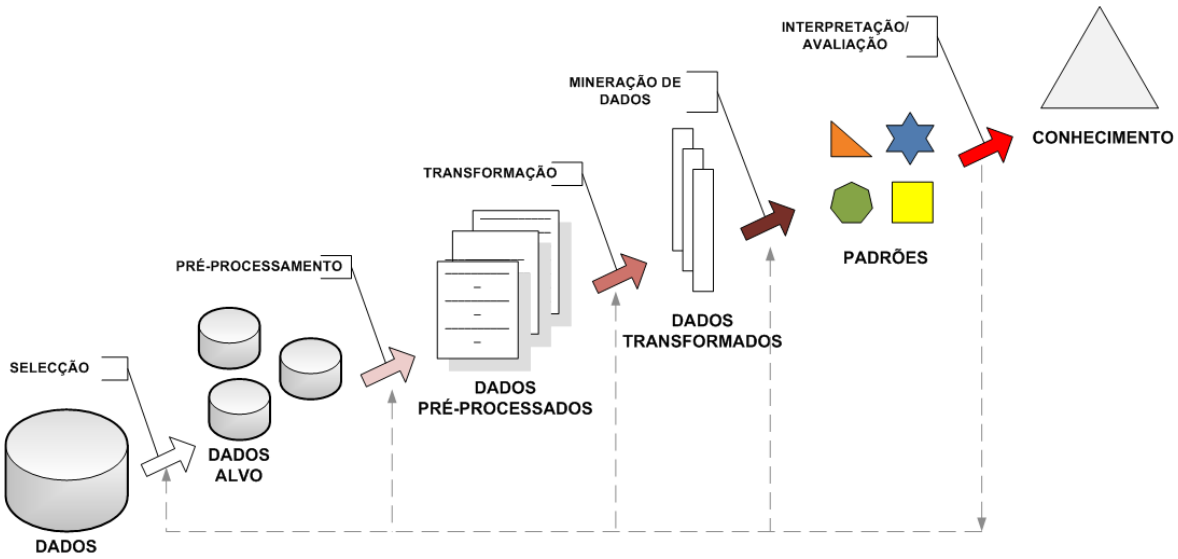


Figura 18: Etapas de descoberta de conhecimento em base de dados (Adaptado de [Fayyad, Piatetsky-Shapiro & Smyth 1996])

4.3 Ferramentas

As diferenças entre as possibilidades humanas e de uma ferramenta especializada na mineração de dados estão principalmente no grande volume de dados, no poder computacional exigido e na repetição de procedimentos. Uma ferramenta de mineração de dados pode adquirir e gerir uma grande quantidade de informação que está para além da capacidade humana. Pode também procurar, de uma forma automática, numa base de dados e encontrar até o mais pequeno padrão que possa ajudar numa melhor previsão [Berson & Stephen 1997].

Nesta secção são apresentadas ferramentas que são as mais referenciadas em artigos da área da mineração de dados e descoberta de conhecimento disponibilizadas gratuitamente e que foram analisadas para serem usadas neste trabalho.

4.3.1 Weka

A ferramenta Weka (Waikato Environment for Knowledge Analysis) [Web 30] é uma ferramenta popular de algoritmos de mineração de dados, aprendizagem computacional e pré-processamento de dados, desenvolvida em Java, na Universidade de Waikato, na Nova Zelândia. Esta ferramenta fornece um extenso suporte para todo o processo experimental de mineração de dados, incluindo a preparação dos dados para análise, avaliando os esquemas de aprendizagem estatisticamente, visualização dos dados de entrada e os resultados de aprendizagem. A Weka é suportada pelo grande conjunto de algoritmos de mineração de dados e de aprendizagem computacional disponibilizados gratuitamente. Nesses algoritmos disponibilizados podemos encontrar algoritmos de classificação, de regressão, de criação de clusters e de regras de associação. Através desta ferramenta é possível pré-processar um conjunto de dados, alimentando um esquema de aprendizagem, analisar os resultados da classificação e da sua respectiva performance sem ser necessário escrever qualquer tipo de código. No entanto as funcionalidades desta ferramenta podem ser incluídas numa aplicação através do uso de uma API disponibilizada em Java, podem ser invocadas a partir da linha de comandos ou a partir da interface gráfica tal como ilustra a Figura 19. Todos os algoritmos recebem como parâmetro de entrada na forma de simples tabela relacional no formato específico ARFF (Anexo C) que pode ser lido de um ficheiro constituído por texto ou gerado a partir de um resultado de uma pesquisa na base dados acedida por Java DataBase Connectivity (JDBC).

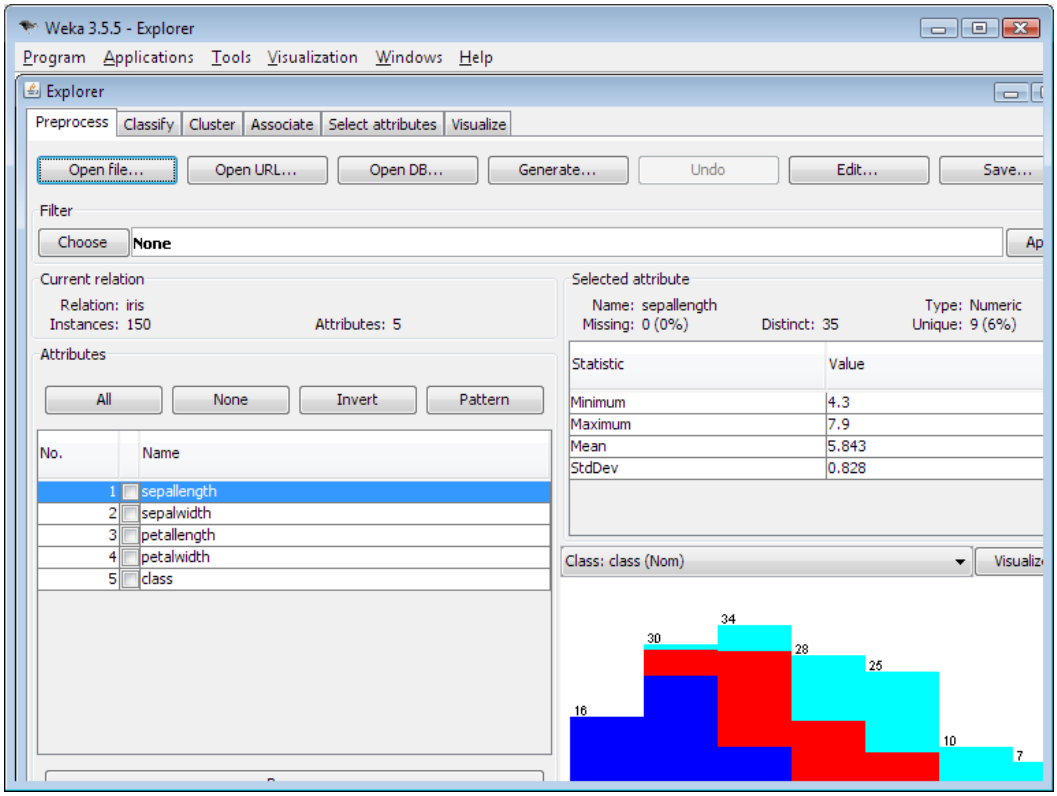


Figura 19: Janela Explorer da Weka

4.3.2 YALE

A ferramenta YALE (Yet Another Learning Environment) [Web 31] como formalmente é conhecida é uma plataforma (ambiente) que permite efectuar experiências com algoritmos de mineração de dados para descoberta de conhecimento. Permite realizar testes com um grande volume de dados e de operadores, descrevendo e armazenando o processo em XML possibilitando a reutilização. O YALE herdou muitos dos algoritmos e ferramentas disponibilizadas pela Weka. Este projecto começou a ser desenvolvido em 2001 na Universidade de Dortmund estando agora disponível numa versão comercial e aberta à comunidade denominada de RapidMiner. Segundo Arndt Faulhaber [Web 29] o nome do projecto indicava que os seus autores tinham plena consciência da já existente lista de plataformas para mineração de dados e descoberta de conhecimento em base de dados, no entanto ainda havia um lugar para uma plataforma deste tipo que se fosse aberta à

comunidade e que oferecesse processamento facilitado dos dados integrados, permitisse alterações de acordo com as preferências de cada utilizador e capaz de uma flexibilidade na criação e selecção de atributos.

De acordo com [Mierswa et al 2006] o YALE fornece mais de 400 operadores para construção de modelos, incluindo operações de input e output, pré-processamento e diversas formas de visualização dos dados. Para além dos algoritmos disponibilizados, são também disponibilizados algoritmos que se encontram no Weka, obtendo partido do melhor das duas ferramentas. Dado que foi desenvolvida em Java, pode ser instalada em múltiplas plataformas correndo portanto nos mais populares sistemas operativos. A partir da YALE podemos aceder aos dados de diversas formas, quer os dados estejam armazenados numa base de dados relacional ou num ficheiro de texto, por exemplo do tipo ARFF.

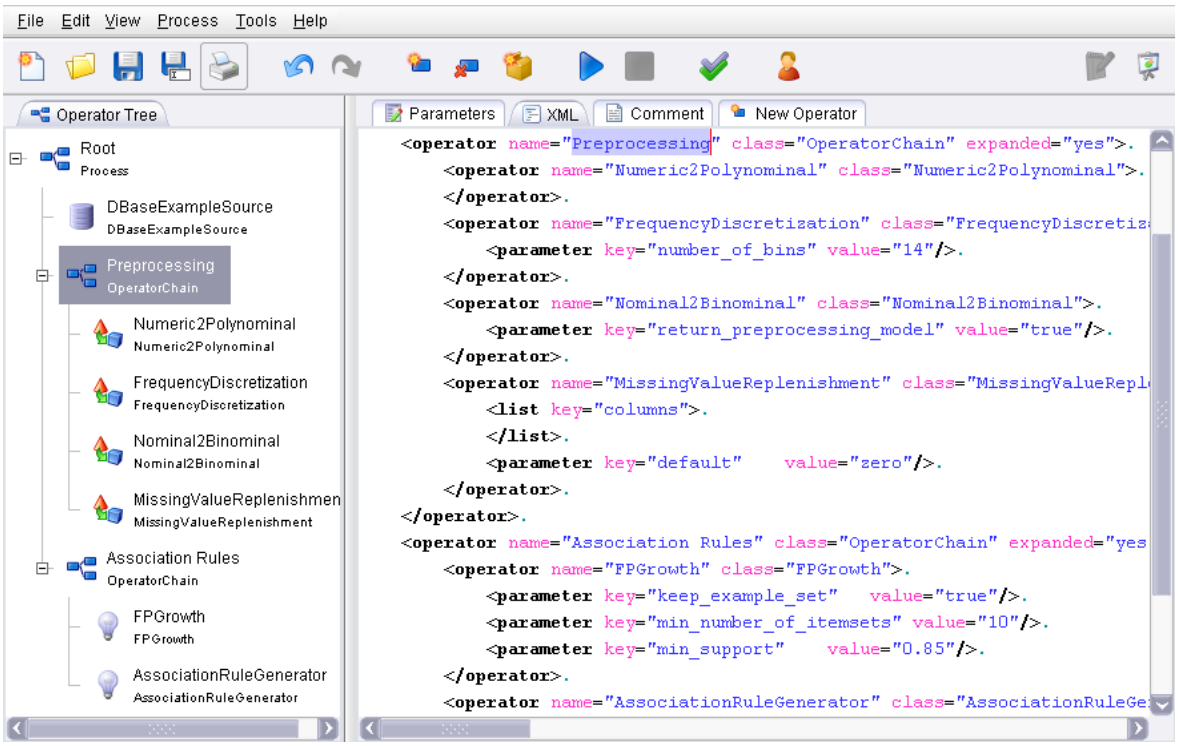


Figura 20: Representação em XML de um modelo de criação de regras de associação na YALE

4.3.3 Weka vs YALE

De acordo com Arndt Faulhaber [Web 29] um dos objectivos iniciais dos criadores da YALE foi a flexibilidade e como tal é esperado que ocorram mais melhoramentos, nomeadamente na flexibilidade na criação de experiências, reutilizando trabalho já desenvolvido anteriormente e de pré-processamento de dados que pode ser retomado posteriormente, apesar de estas propriedades já serem base do YALE. Estes critérios têm sido aplicados de acordo com Arndt Faulhaber [Web 29]:

- Na possibilidade reajustar ou criar experiências usando operadores em árvore;
- Uma interface única coerente para experimentar os dados em diferentes níveis;
- Nos imensos filtros de pré-processamento que estão integrados na ferramenta.

O uso destas duas ferramentas em código Java, de acordo com Arndt Faulhaber [Web 29], é de fácil inclusão; a documentação ou comunidade existente é satisfatória, apesar de evidenciar que a reutilização de uma experiência em Weka possa ser possível apesar da documentação não ser esclarecedora. Uma outra comparação que se pode encontrar em [Web 29] é sobre os algoritmos disponibilizados pelas duas ferramentas e que foram comprovadas pelo simples uso destas ferramentas. Os algoritmos presentes na Weka também se encontram na YALE, no entanto, a YALE é mais escalável uma vez que oferece a possibilidade de gerar novos algoritmos, mesmo que estes não sejam criados em Java, usar os existentes na Weka e os que não se encontram na Weka (apenas no YALE).

Uma vez que o YALE é uma ferramenta que herda muitas propriedades encontradas em Weka, com possibilidade de ser estendida e em grande expansão na área foi escolhida esta ferramenta para este trabalho na descoberta de conhecimento em base de dados.

4.4 Algoritmos de mineração de dados

De acordo com [Witten & Frank 2005] os algoritmos aplicados na mineração de dados podem ser agrupados em dois grupos, considerando as variáveis que são usadas na mineração, estes dois

grupos são os algoritmos de aprendizagem supervisionada e não supervisionada. Os algoritmos de aprendizagem supervisionada distinguem as variáveis entre variáveis de previsão e de resposta. O principal objectivo destes algoritmos encontra-se no estabelecimento de uma relação entre as diversas variáveis de previsão de forma a antever qual será o valor da variável de resposta.

De acordo com [Witten & Frank 2005] os algoritmos de aprendizagem não-supervisionada são algoritmos orientados para a exploração de dados sem efectuar uma distinção entre as variáveis. O seu objectivo passa por encontrar relações quando não se pretende determinar um valor de resposta em concreto.

De acordo com aquilo que foi estudado dos algoritmos de mineração de dados, no contexto deste trabalho, e dos que estão presentes na YALE foi decidido utilizar dois tipos de algoritmos que são:

- Clustering
- Regras de Associação

4.4.1 Algoritmos de Clustering

Segundo [Witten & Frank 2005] os algoritmos de clustering são algoritmos de aprendizagem não-supervisionada e que permitem criar grupos naturais de dados considerando o grau de similaridade entre eles. Ao contrário da classificação, não sabemos o que serão os clusters quando começarmos, ou por quais atributos os dados serão agrupados, pelo que será necessário conhecer de antemão os dados minerados. Por vezes, é necessário modificar o processo de clustering excluindo variáveis utilizadas no agrupamento porque podem ser consideradas irrelevantes ou sem aplicação no processo.

O algoritmo mais clássico de clustering é conhecido como k-means sendo a sua utilização relativamente simples e obriga a que cada instância (registo) pertença a um único cluster. Como restrição, este algoritmo apenas permite que as variáveis utilizadas sejam numéricas, caso contrário é necessário proceder a uma transformação prévia.

No primeiro passo do algoritmo, escolhemos antecipadamente quantos clusters vão ser criados: esta parametrização é conhecida como o parâmetro k e normalmente o seu valor recomendado corresponde à raiz quadrada de metade do número de instâncias, o que não significa que este seja o valor ideal. Em seguida são escolhidos k pontos aleatoriamente como pontos centrais dos respectivos clusters sendo que as instâncias são atribuídas ao cluster cujo centro esteja mais próximo, onde o valor de proximidade é calculado com base na distância Euclidiana. Assim que este passo finda, são repetidos os cálculos dos novos centros e atribuídas as instâncias aos clusters respectivos, até que os centros encontrados ao longo das vária iterações estabilizem e se mantenham constantes. Este método de clustering aqui descrito é considerado por [Witten & Frank 2005] como sendo simples, eficaz e sendo fácil a prova de que escolhendo o centro do cluster como ponto central minimiza o total da distância Euclidiana. No entanto, trata-se de um mínimo local não havendo garantia de que se trata de um mínimo global correspondendo a solução considerada óptima. Os clusters finais são muito sensíveis aos centros dos clusters iniciais onde arranjos completamente diferentes podem surgir a partir de pequenas alterações na escolha aleatória inicial. De acordo com [Witten & Frank 2005] muitas vezes as pessoas para aumentarem a hipótese de encontrar um mínimo global executam o algoritmo diversas vezes com escolhas iniciais diferentes e escolhem o melhor resultado final – o que tenha a menor distância Euclidiana.

O processo de avaliação dos resultados é muitas vezes uma medida subjectiva em termos da utilidade dos resultados para quem os avalia. Este processo pode ser seguido por um segundo passo de classificação em que as regras aprendidas forneçam uma descrição inteligível de como as novas instância devem ser colocadas nos clusters.

Dado que o YALE disponibiliza duas implementações baseadas no algoritmo K-Means, o KMeans e o W-SimpleKMeans, optou-se pelo uso do algoritmo KMeans, pela facilidade que permite na análise dos resultados obtidos da aplicação do algoritmo de clustering apoiado pelo maior número de opções de visualização gráfica dos resultados obtidos.

4.4.2 Algoritmos de criação de Regras de Associação

Os algoritmos de criação de Regras de Associação são algoritmos não-supervisionados e que permitem a previsão de uma qualquer variável, não sendo necessário estabelecer uma variável de resposta. Um algoritmo de criação de regras de associação é uma aproximação descritiva na exploração dos dados que pode ajudar na identificação de relacionamentos entre os dados que estão numa base de dados, sendo largamente utilizados, numa primeira fase em que não se conhece os padrões procurados. As duas aproximações mais usadas são a descoberta de associações e sequências. A descoberta de associações tenta encontrar regras acerca dos itens que aparecem em conjunto num evento como no caso de associações entre os produtos comprados por clientes de um retalhista, como o caso muito popular de extracção de conhecimento Market Basket Analysis [Witten & Frank 2005]. A descoberta de uma sequência é muito similar na medida em que uma sequência é uma associação relacionada ao longo do tempo.

As regras de associação são compostas por uma premissa e uma conclusão sendo que a premissa é uma condição que se considera necessária para a verificação de uma conclusão. Por exemplo, numa regra de associação do tipo "Se compra martelo então também compra pregos", a premissa é "compra martelo" e a conclusão "compra pregos".

Segundo [Witten & Frank 2005] diferentes regras de associação expressam diferentes regularidades que estão subentendidas no conjunto de dados e que geralmente predizem diferentes coisas. Dado que podem existir muitas e diferentes regras de associação deduzidas a partir de um pequeno conjunto de dados é necessário proceder à avaliação das regras quanto à sua qualidade. As principais métricas identificadas por [Witten & Frank 2005] na avaliação da qualidade de uma regra são os seus valores de cobertura e de confiança. A medida de cobertura de uma regra de associação indica-nos o número ou percentagem do total de instâncias em que se verifica tanto a premissa como a conclusão. A medida de confiança de uma regra de associação indica-nos o número ou percentagem do total de instâncias em que se verifica a premissa então também se verificará a conclusão. Claro que as regras obtidas durante a aplicação dos algoritmos de criação de regras de associação podem ser reduzidas bastando para tal definir parametrizações mínimas para os valores calculados de confiança e de cobertura.

Contudo, para descobrirmos regras com significado relevante devemos começar por criar conjuntos de itens e analisar a sua frequência relativa de ocorrência dos itens e as suas combinações. Portanto uma perguntar para um conjunto de itens pode ser: "dada a ocorrência do item A (a premissa) quantas vezes o item B (conclusão) ocorre?". Ou utilizando o exemplo do martelo e pregos temos: "Quantas vezes acontece quando uma pessoa compra um martelo também compra pregos?".

Por exemplo, tomemos como exemplo o seguinte conjunto hipotético de resultados (Tabela 2) para demonstrar os conceitos acima

Tabela 2: Quantidade itens vendidos em conjunto e separadamente

Itens vendidos	Total
Martelos	75
Pregos	120
Madeira	30
Martelo e Pregos	22
Pregos e Madeira	15
Martelo e Madeira	15
Martelo, Madeira e Pregos	7
Compras efectuadas na loja	1500

Com estes valores podemos efectuar alguns cálculos, como exemplo em termos de cálculo de cobertura (Tabela 3) temos que:

Tabela 3: Exemplo de cálculo de cobertura para as vendas

Itens vendidos	Cobertura
Martelos e Pregos	1,47% (22/1500)
Martelo, Madeira e Pregos	0,47% (7/1500)

Assim como, por exemplo em termos de cálculo de confiança (Tabela 4), temos que:

Tabela 4: Exemplo de cálculo de confiança para as vendas

Itens vendidos	Confiança
Se Martelo então Pregos	29,3% (22/75)
Se Pregos então Martelo	18,3% (22/120)
Se Martelo e Pregos então Madeira	31,8% (7/22)
Se Madeira então Martelo e Pregos	23,3% (7/30)

Destes cálculos podemos dizer com um maior grau de confiança que quem compra o martelo também compra pregos (29,3%), do que alguém que compre pregos compre também um martelo (18,3%). A prevalência desta associação entre martelos e pregos é suficiente para a considerar como válida.

Em termos de YALE, os algoritmos disponibilizados para criar regras de associação são o W-Apriori que é herdado da Weka e o FPGrowth que deve ser usado em conjunto com o algoritmo AssociationRuleGenerator. O algoritmo W-Apriori consegue executar todo o processo, desde a criação de conjunto de itens até à geração de regras. Apesar disto, uma vez que se trata de um algoritmo herdado da Weka, a análise de dados é relativamente limitada, dado que apenas é possível verificar as regras num formato de texto livro seguidas do valor de confiança. Ao contrário, do algoritmo FPGrowth disponibilizado pelo YALE em que é possível disponibilizar os resultados numa forma tabular, com conjunto extra de informações assim como dos valores de cobertura e confiança. Este será o algoritmo utilizado apesar de ser necessário aplicar também o algoritmo

AssociationRuleGenerator, uma vez que o FPGrowth apenas calcula os conjuntos de itens submetendo depois estes conjuntos para o AssociationRuleGenerator para criação das regras.

Capítulo 5

Metodologia de Integração para a Descoberta de Conhecimento

De acordo com o dicionário online da Porto Editora [Web 32] uma metodologia é:

1. "um conjunto de regras ou princípios empregados no ensino de uma ciência ou arte";
2. "parte da lógica que estuda os métodos das diversas ciências";
3. "arte de dirigir o espírito na investigação da verdade".

Uma vez que já foram introduzidas e analisadas as três parte principais deste trabalho:

- Relatórios médicos;
- Integração Orientada a Serviços;
- Descoberta de Conhecimento;

podemos então avançar neste capítulo para a definição de uma metodologia de Integração Orientada a Serviços para a Descoberta de Conhecimento em Relatórios Médicos, tal como foi

definida em [Pinheiro & Alves 2008], apresentando um conjunto de regras ou princípios que devem ser seguidos para cumprir os objectivos de:

1 Usar uma arquitectura de integração orientada a serviços que seja capaz de interligar os diferentes serviços, que por sua vez usam diferentes protocolos e mensagens;
2 Obter um serviço de codificação e descodificação de um relatório médico de texto livre para um formato normalizado, com base num modelo;
3 Obter um mapeamento entre os termos médicos usados nos respectivos relatórios e uma codificação desambigua para solucionar as problemáticas relativas ao uso de linguagem natural nos relatórios médicos (negação);
4 Aplicar os algoritmos para a descoberta de conhecimento sobre a informação normalizada integrada e armazenada em base de dados.

Esta metodologia pressupõe que os relatórios médicos em formato de texto livre não podem ser simplesmente abandonados e substituídos por relatórios estruturados. Tal não ocorre actualmente e não acontecerá até que os médicos se sintam confortáveis com as normas ou com ferramentas que geram ou ajudam a escrever um relatório estruturado [Bortoluzzi 2003]. Apesar disso, o próprio médico terá o seu próprio template para cada possível diagnóstico com a sua própria marca pessoal.

De referir ainda que, não nos podemos esquecer dos relatórios que já existem e se encontram guardados contendo informação médica valiosa que pode e deve ser considerada nos diagnósticos de exames futuros.

Sendo assim, e em acordo com as normas apresentadas para integração e de relatórios médicos é possível sugerir uma metodologia para a integração de informação e extracção correspondendo às necessidades dos médicos, para o novo relatório estruturado, todas as plataformas, sistemas médicos relacionados neste âmbito e descobrir conhecimento a partir da informação presente nos relatórios médicos.

Com a metodologia que vai ser apresentada podemos indicar dois caminhos possíveis de integração para a informação: relatórios médicos estruturados ou relatórios médicos herdados. Na generalidade dos casos o processo de integração de relatórios herdados é precedente do processo de integração dos relatórios estruturados.

5.1 Arquitectura de Integração Orientada a Serviços

O primeiro passo desta metodologia passa pela criação de uma Arquitectura de Integração Orientada a Serviços (Figura 21). Esta arquitectura vai ter como suporte a plataforma Open-ESB que possui uma série de componente JBI para as mais diversas funções. Como já apresentado anteriormente, os componentes JBI dividem-se em componentes de ligação e motores de serviço. Os componentes de ligação fornecem protocolos de transporte e comunicação, o acesso a serviços remotos a partir do ambiente assim como serviços dentro do próprio ambiente. Os motores de serviço permitem ligações à lógica de negócio, por exemplo, podemos utilizar um motor de serviço compatível com WS-BPEL, para orquestrar processos de negócio que sejam descritos em WS-BPEL.

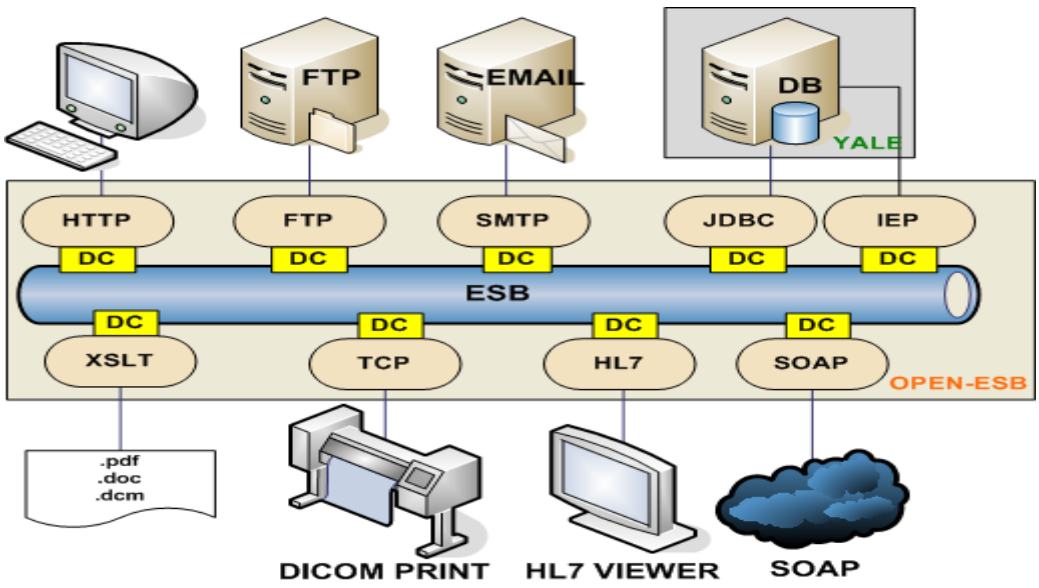


Figura 21: Visão geral da Arquitectura de Integração de Relatórios Médicos

Dado que os componentes JBI podem ser instalados na plataforma Open-ESB, os componentes considerados indispensáveis para esta metodologia são:

- Motores de Serviço
  - BPEL – A Business Process Execution Language (BPEL) é utilizada para organizar os processos numa aplicação composta;
  - Java EE – permite que serviços disponibilizados a partir, por exemplo, de EJB possam interagir com servidor hospedeiro da aplicação;
  - XSLT – usado para efectuar transformações em documentos XML com a ajuda de XSL;
- Componentes de Ligação
  - Ficheiros (FileBC) – fornece um serviço de transporte para um sistema de ficheiros;
  - File Transfer Protocol (FTP) – este componente recebe mensagens usando o protocolo FTP;
  - Hypertext Transfer Protocol (HTTP) – é responsável por ligar uma instância JBI a serviços Web externos e por ligar esses serviços externos à instância JBI;
  - Java Database Connectivity (JDBC) – através desta componente é possível configurar e ligar a bases dados que suportam a API da especificação JDBC 3.0;
  - Health Level 7 (HL7) – fornece uma solução abrangente para a configuração e conexão aos componentes JBI bem como a sistemas externos do ambiente JBI que utilizem o protocolo HL7;

Estes componentes interagem entre eles por via de um processo de troca de mensagens, que é descrito usando documentos WSDL, publicados pelo fornecedor do serviço. Esta descrição de serviço é a única fonte de informação necessária para os componentes consumidores interagirem com os fornecedores de serviço. O JBI fornece a infra-estrutura leve de mensagens, conhecida como encaminhador de mensagens normalizadas, que está na base do Java Open Entreprise Service Bus (ESB), que fornece o mecanismo para actual troca de mensagens num formato

independente (XML), onde as interfaces de integração são desenvolvidas com assumpções mínimas entre as partes responsáveis pelo envio e recepção da mensagem, reduzindo o risco que uma mudança em uma aplicação ou módulo force uma mudança em outra aplicação ou módulo, usando sempre uma implementação JBI como intermediária.

Através da plataforma Open-ESB é possível integrar funções de negócio existentes como serviços e desacoplar interacções entre os fornecedores de serviço e consumidores. A plataforma Open-ESB fornece suporte directo para criar aplicações compostas através do mecanismo JBI de montagem de serviços. Este suporte permite que as aplicações sejam compostas directamente do serviço base de interfaces disponibilizadas pelo JBI como serviços unitários e usando BPEL (Business Process Execution Language) para orquestração de processos [Web 12].

Este suporte directo de composição de aplicações construídas em cima de uma arquitectura orientada a serviços e o uso de uma infra-estrutura de mensagens normalizadas tornam a plataforma Open-ESB, que implementa a especificação JBI, o suporte ideal para construir aplicações orientadas a serviços e conseguindo integração orientada a serviços de sistemas existentes usando mensagens normalizadas em XML [Web 11] que serão necessárias para a troca de informações entre os componentes.

Considerando o caso em que os relatórios médicos são herdados e encontram-se armazenados num servidor de ficheiros FTP (File Transfer Protocol) podemos orquestrar em BPEL (Figura 22) um processo de negócio onde os relatórios são normalizados e correspondente estrutura de informação guardada numa base de dados para futuras operações de mineração de dados usando a ferramenta YALE.

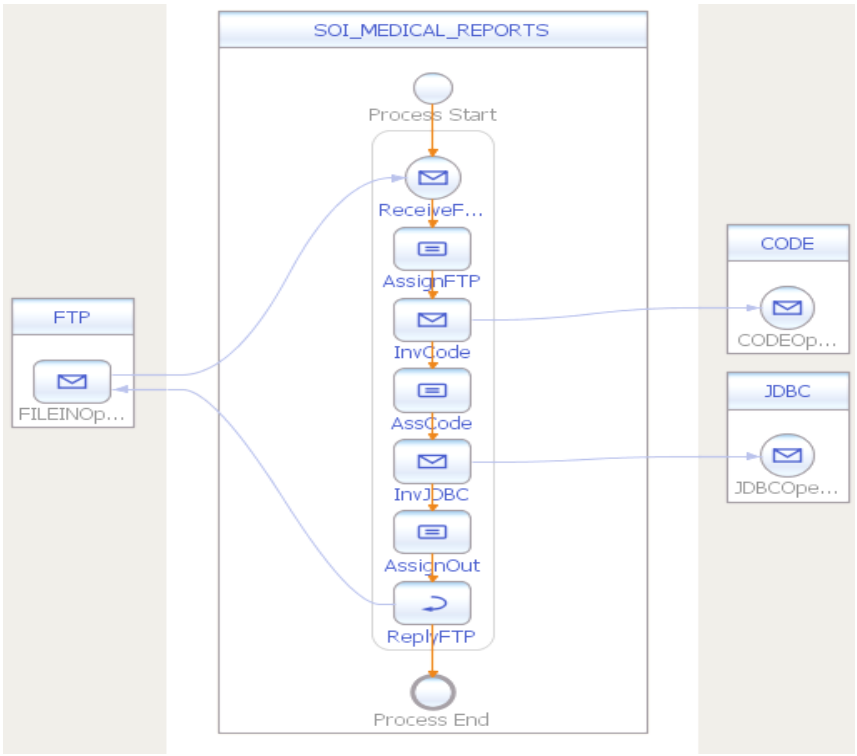


Figura 22: Exemplo BPEL de Integração de Relatórios Médicos

5.2 Serviço normalização (codificação)

O segundo passo desta metodologia reside na definição de um parser orientado pelos templates dos médicos. Uma vez que todos os relatórios médicos foram criados a partir de um template podemos efectuar uma análise sintáctica e semântica aos relatórios médicos usando o correspondente XSD para cada template de cada médico e obter toda a informação numa forma estruturada a partir do repositório dos relatórios. Este processo de codificação e de transformação ocorre muitas vezes no mundo empresarial quando se necessita efectuar uma migração de informação de sistemas antigos para os novos sistemas. Prevendo este cenário em termos de integração a plataforma Open-ESB disponibiliza uma ferramenta de análise sintáctica e semântica orientada a ficheiros. Esta análise pode ser feita recorrendo aos serviços de codificação disponibilizados pelo motor de serviços FileBC do Open-ESB, permitindo efectuar o reconhecimento dos ficheiros por posições delimitadas ou através de identificadores no texto.

Ora no caso deste trabalho essa codificação apoiou-se no reconhecimento de identificadores no texto dos relatórios médicos, uma vez que estes na sua maioria contêm referências à secção do documento como os dados do paciente, procedimentos, modalidades, observações, conclusões, entre outros. Este serviço de codificação disponibilizado pela Open-ESB fornece ferramentas para definir e testar sistemas de codificações, feitos à medida, necessitando apenas de ser definido o XML Schema correspondente aos ficheiros que se pretendem codificar. Como referido acima, cada tipo de codificação necessita do seu próprio XML Schema com as suas regras de codificação. Uma vez escolhido qual o XSD podemos construir o correspondente documento XML que vai conter a informação não estruturada do relatório médico, de acordo com as regras definidas pelos XSD.

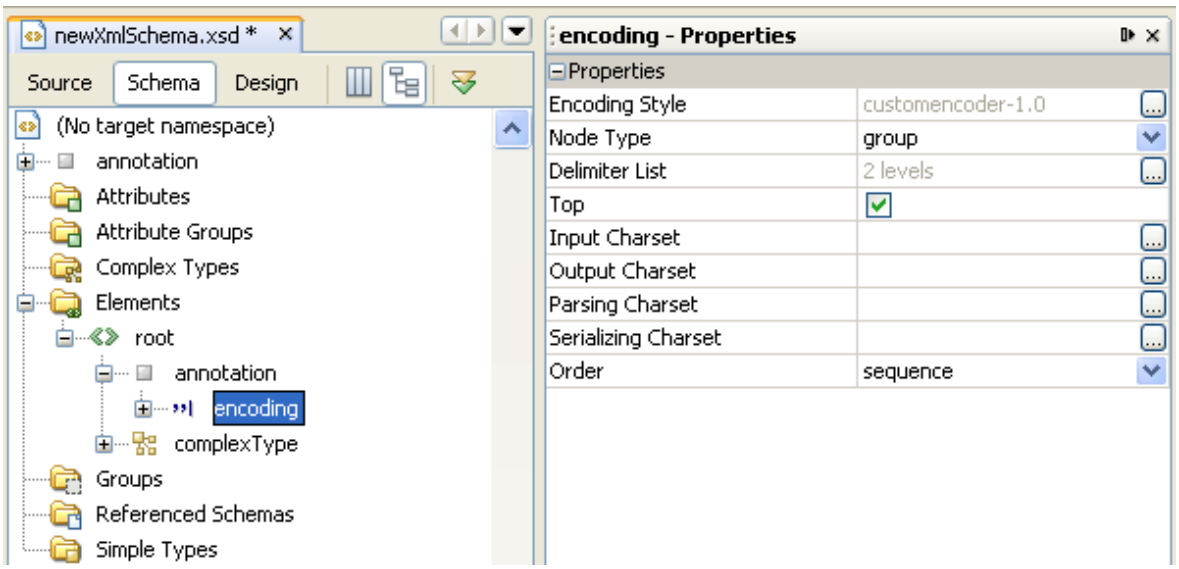


Figura 23:Sistema de normalização (codificação) disponibilizado pelo Open-ESB

Na Figura 23 podemos verificar que é dado ao utilizador a possibilidade de alterar as propriedades relativas à codificação. Quando ocorre uma mudança nas propriedades da informação da aplicação que se encontra debaixo do modelo XSD estas também irão mudar. Por exemplo, no caso deste trabalho, uma das listas de delimitadores é representada pela Figura 24.

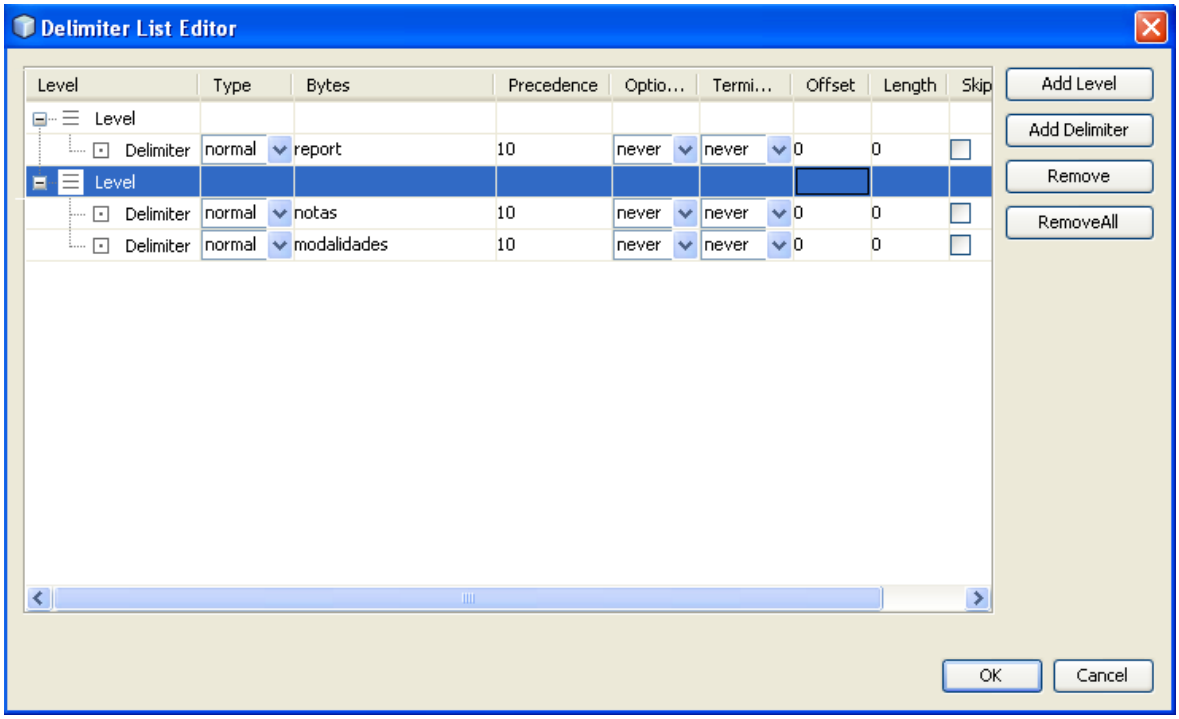


Figura 24: Exemplo de uma lista de delimitadores da normalização dos relatórios médicos

Para exemplificar a forma como este sistema de codificação é aplicado tomemos como exemplo o seguinte caso:

A notas B report C modalidades D

Pela lista de delimitadores podemos dizer que o primeiro delimitador é o termo “report” e depois o termo “notas” seguido do termo “modalidades”, Em termos de estrutura, ainda que abstracta, do XSD temos:

- Elemento: root
  - Elemento: groupNodeReport
    - Elemento: elementoNotas
      - Elemento: delimElementoA
      - Elemento: delimElementoB
    - Elemento: elementoModalidades
      - Elemento: delimElementoC
      - Elemento: delimElementoD

Aplicando as regras definidas pelo delimitadores definidas no XSD ao XML temos:

- Elemento: root (NodeType=group, DelimiterList=2 levels ')
  - Elemento: groupNodeReport (NodeType=group)
    - § Elemento: elementoNotas (NodeType=limitado, limitador herdado 'report')
      - Elemento: delimElementoA (NodeType=limitado, limitador herdado 'notas')
      - Elemento: delimElementoB (NodeType=limitado, limitador herdado 'notas')
    - § Elemento: elementoModalidades (NodeType=limitado, limitador herdado 'report')
      - Elemento: delimElementoC (NodeType=limitado, limitador herdado 'modalidades')
      - Elemento: delimElementoD (NodeType=limitado, limitador herdado 'modalidades')

A operação de normalização permite a troca de informação entre sistemas e uma possível conversão desta estrutura normalizada a partir do XSD correspondente para um relatório médico normalizado. Uma vez normalizada a informação contida nos relatórios pode ser publicada para o Java Open Enterprise Service Bus (através da ligação DC (Direct Channel) ao ESB) e vai ser subscrita por diferentes destinos através dos componentes de ligação ou utilizada em outra lógica de negócio através dos motores de serviço. Como tal, vai ser possível transformar estes relatórios normalizados em diferentes formatos, enviar por e-mail ou mostrar num ecrã de visualização HL7, entre outras coisas.

Assim que a informação esteja normalizada é necessário transformar a língua em que o relatório está escrito, neste caso português, para termos médicos identificados por um código. Um código com a descrição correspondente do termo médico, o seu valor, para que possa de uma forma não ambígua ser identificado, permitindo a aplicação de algoritmos de mineração de dados (Noumeir,

2006) facilitando o processo de indexação e obtenção selectiva, sem ser necessário recorrer a uma análise sintáctica e semântica de língua natural (NLP – Natural Language Parsing) [Langlotz 2002].

5.3 Mapeamento de termos médicos

A codificação dos termos médicos pode ser feita utilizando, por exemplo, o sistema de codificação SNOMED CT, que é um conjunto de mais de 357000 conceitos médicos com um único significado, não ambíguo e baseados em definições formais lógicas organizadas em hierarquias. No entanto, trata-se de uma biblioteca médica proprietária sendo necessário pagar pela sua utilização, e uma vez que seria incoerente o seu uso nesta metodologia, dado que o suporte deste trabalho reside na utilização de standards abertos. Portanto, apesar de ser um sistema de codificação muito utilizado na área médica e ser muito completa, a decisão foi a de recorrer a um mapeamento interno definindo um CODE serviço para esta codificação.

Este mapeamento interno, que se encontra parcialmente no anexo D, foi feito a pensar na codificação e construído de forma a reflectir uma árvore de terminologias da seguinte forma:

1. Sempre que existe um novo termo é atribuído um novo código único sequencial;
2. Se já existir um atributo pai para o termo médico é criado um atributo filho, um nodo na árvore. Isto é, tomemos como exemplo o seguinte caso:
    a. "fracturas" é o atributo pai já existente (20 – este caso significa que foi a segunda entrada nesta lista. Quando ocorrer um elemento novo que seja o vigésimo então o seu código será 200);
    b. É necessário acrescentar um atributo filho "recentes";
    c. Utiliza-se o código do pai e acrescenta-se a indicação de que se trata do primeiro filho (201), caso não exista ainda um atributo filho, e assim sucessivamente.

Trata-se portanto de um notação muito simples sendo a sua estrutura comparável à de uma árvore de decisão.

A principal dificuldade que pode ser encontrada neste tipo de codificação, da tradução de termos médicos para códigos, reside língua materna do documento, no uso da negação dos termos. De acordo com [Clunie, David A 2001] um médico especialista que apenas refira nos seus relatórios o termo "normal" é considerado que os seus relatórios são de alguma forma de qualidade inferior ou simplesmente incompletos. Portanto, os especialistas preferem listar coisas que não encontraram mesmo tratando-se de um exame que seja inteiramente "normal". Por exemplo, considerando um relatório onde o especialista não encontrou qualquer sinal de atrofia mas encontrou uma fractura. Utilizando a codificação interna definida podemos identificar de uma forma desambigua os termos médicos e codificar a negação de termos. Ora para o caso em que o serviço CODE recebe a seguinte (parte) mensagem:

```
<finding att="Not Found">atrofia</finding>
<finding att="Found">fractura</finding>
```

O mapeamento interno existente para este caso é o seguinte:

Tabela 5: Mapeamento interno de Termos Médicos (parte).

Termo Médico	Encontrado	Não Encontrado
Atrofia	110	110N
Atrofia focal	1101	1101N
Fractura	20	20N
Fractura recente	201	201N

Logo depois de receber a mensagem e consultada a Tabela 5 o serviço CODE irá responder com a seguinte (parte) mensagem:

```
<finding>110N<finding>
<finding>20</finding>
```

Portanto, a informação guardada na base de dados irá facilitar a utilização da YALE, uma vez que tal como pretendido os termos médicos, negados ou não, estão agora identificados de uma forma desambigua e única sem ser necessário usar a língua natural e codificando a negação.

5.4 Descoberta de Conhecimento

O processo de integração apresentado e ilustrado Figura 22 é um de entre muitos outros possíveis. Por exemplo, podemos ter dois caminhos paralelos para este processo de integração. Neste processo um dos caminhos pode ser responsável pela codificação das mensagens antes de serem guardadas na base de dados e outro para guardar a informação sem estar codificada, facilitando a mineração dados e permitindo a manutenção do histórico da informação processada.

Uma vez normalizada a informação contida nos relatórios médicos e codificada de forma desambigua eliminando a subjectividade da língua natural é agora possível aceder de forma directa e eficiente à informação. Uma vez que no sistema de codificação se definiram e identificaram identificadores foram transpostos para uma base dados em PostgresSQL na mesma estrutura do sistema de codificação, resultando numa estrutura genérica que pode ser observada em parte na Tabela 6.

Tabela 6: Parte dos resultados obtidos da codificação

Paciente	28304	111875	217828	111221	41669	132659	192095	163858
Sexo	1	1	2	2	1	2	1	2
Idade		3	3	4	3	3	3	3
Series	3	3	7	6	5	3	3	3
Modalidades	102	1011111	1031	1031	1031	1011111	1011111	1011111
Nota1	30	30	30	30	30		30	30
Protocolo1	20111111	203111	20311111	20111111	20111111	20111111	20111111	203111
Protocolo2		203112	20311211	20111211	20111211	20111211	20111211	203112
Report1	101113N	3011N	3011N	3012114N	3012114N	3012114N	3012114N	3011N
Report2	2013N	40111N	601111N	80111	80111	80111	80111	40111N
Report3		501	501	901N	901N	901N	901N	501
Report4		3021N	3021N	1001111N	1001111N	1001111N	1001111N	3021N
Conclusão	C10	C10	C10	C10	C10	C10	C10	C201

Integrada a informação e guardada na base de dados é possível agora aplicar técnicas de mineração de dados e algoritmos de aprendizagem computacional, que a ferramenta YALE fornece, podemos aplicar os algoritmos para a criação de regras de associação e de clustering para os relatórios médicos.

5.4.1 Modelo para a criação de Regras de Associação

Esta técnica de mineração de dados possibilita a identificação de padrões em grandes base de dados. Esta identificação de padrões ajuda na recolha e interpretação dos resultados obtidos para reunir e interpretar os resultados obtidos para adquirir o conhecimento específico para uma conclusão ou assumpção para o caso de estudo.

O processo para a descoberta de regras de associação em YALE usa um conector a uma base de dados, um operador de pré-processamento em cadeia, um operador de aprendizagem não supervisionada de um conjunto de itens e um algoritmo gerador de regras de associação (Figura 25). A maior parte do pré-processamento encontra-se no operador FrequencyDiscretization, o filtro Nominal2Binomial e MissingValueReplenishment. O operador "FrequencyDiscretization" efectua transformações discretas em atributos numéricos e agrupando os valores, isto é, trata-se de uma actividade de preparação de dados que converte dados contínuos para dados discretos pela substituição de um valor, que se encontra dentro de um intervalo contínuo, por um identificador do intervalo correspondente. Por exemplo, a idade pode ser convertida em intervalos representativos bebé [0,20], criança [4,12], jovem [12,26], adulto [27,65] e idoso acima de 65 anos.

A operação de filtragem "Nominal2Binomial" converte um possível valor nominal (YALE considera que os termos negados são nominais por serem alfanuméricos) de um atributo polinomial em binomial (binário) que é verdadeiro se o valor em questão tem o valor nominal procurado.

O operador "MissingValueReplenishment" substitui valores que não existem num determinado atributo. Se um valor está em falta, é substituído por um valor resultante da aplicação de umas das funções disponibilizadas por este operador: "minimum", "maximum", "average", "zero", "value" e

"none", no entanto, se a função utilizada for "none" não se procede à substituição. No caso deste modelo uma vez que lidava com dados numéricos utilizou-se a função "zero" que substitui pelo valor numérico zero.

Os operadores de pré-processamento são necessários uma vez que alguns esquemas particulares de aprendizagem não conseguem lidar com atributos com valores de determinado tipo. O próximo operador de mineração de dados para conjunto de dados frequentes chama-se FPGrowth [Han, Pei & Yin 2000]. Este algoritmo calcula de uma forma eficiente o conjunto de valores dos atributos que ocorrem mais vezes juntos. A partir deste conjunto, denominado de conjunto de itens frequentes, as regras com maior confiança são calculadas com o gerador de regras de associação.

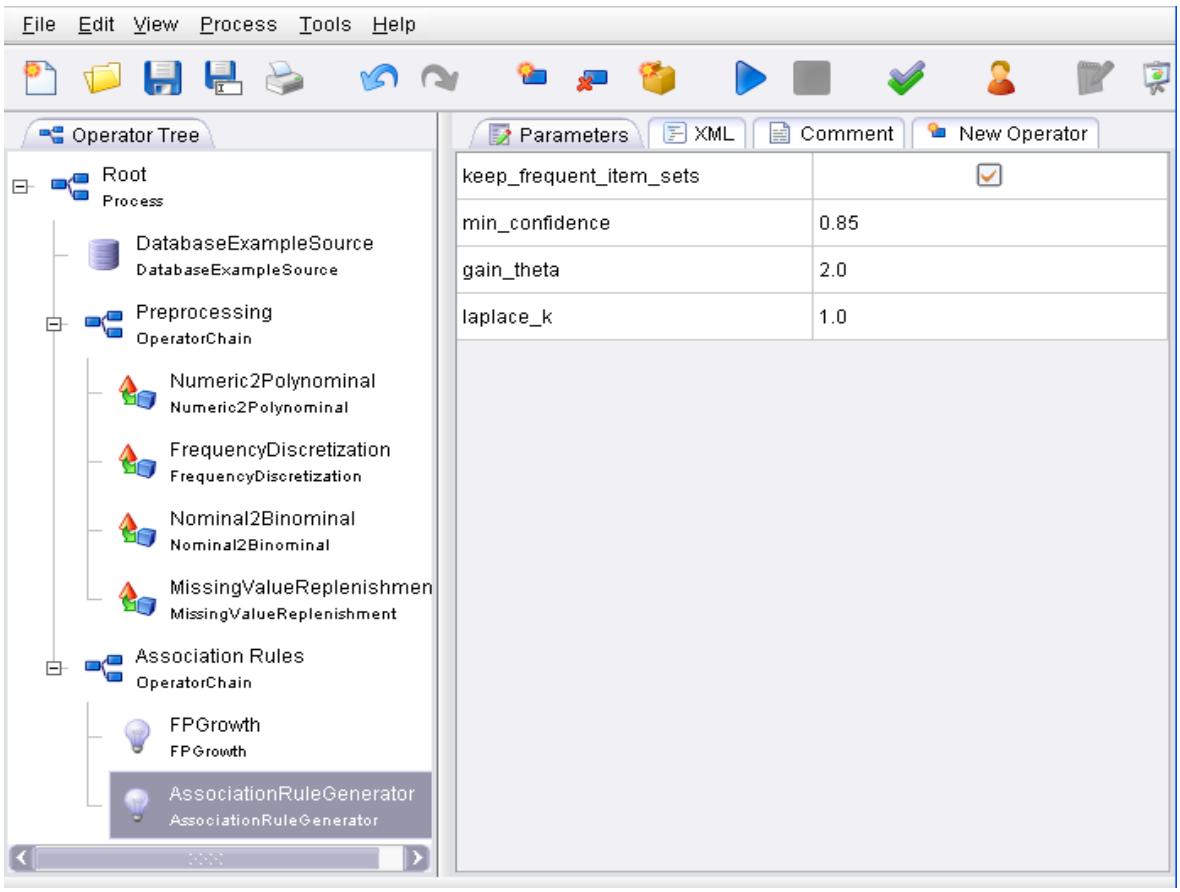


Figura 25: Modelo para a criação de regras de associação em YALE

5.4.2 Modelo para a criação de Clusters

O processo da criação de clusters pode ser considerado como o mais importante algoritmo de aprendizagem não-supervisionada. Como qualquer outro problema em que se aplique este tipo de algoritmos o objectivo é de encontrar uma estrutura numa colecção de dados, isto é, o objectivo é organizar objectos em grupos cujos membros sejam similares entre eles de alguma forma. Como tal um cluster pode ser considerado uma colecção de objectos similares entre si e que são diferentes dos objectos pertencentes a outros clusters.

O processo para a criação de cluster em YALE usa um conector a uma base de dados, um operador de pré-processamento em cadeia, um operador para a criação de clusters (Figura 26). A maior parte do pré-processamento encontra-se no operador FrequencyDiscretization, o filtro Nominal2Numeric e MissingValueReplenishment. O operador "Nominal2Numeric" mapeia todos os atributos não numéricos em valores reais. No entanto não altera os valores de atributos numéricos e os atributos binários são mapeados para 0 e1. No caso de atributos nominais é feito um dos seguintes cálculos:

- Dicotomia – um novo atributo por cada valor do atributo nominal. O novo atributo que corresponde ao actual valor nominal fica com o valor 1 e todos os outros atributos ficam o valor 0;
- Alternativamente os valores dos atributos nominais podem ser visto como estando na mesma posição, e como tal os atributos nominais simplesmente irão ser transformados em valores reais.

Os operadores de pré-processamento são necessários uma vez que alguns esquemas particulares de aprendizagem não conseguem lidar com atributos com valores de determinado tipo (os restantes já foram abordados no modelo de criação de regras de associação). O próximo operador de mineração de dados chama-se KMeans representando uma simples implementação do algoritmo k-means. Finalmente, o operador "SVDReduction" é um método de redução de dimensões baseado no Singular Value Decomposition.

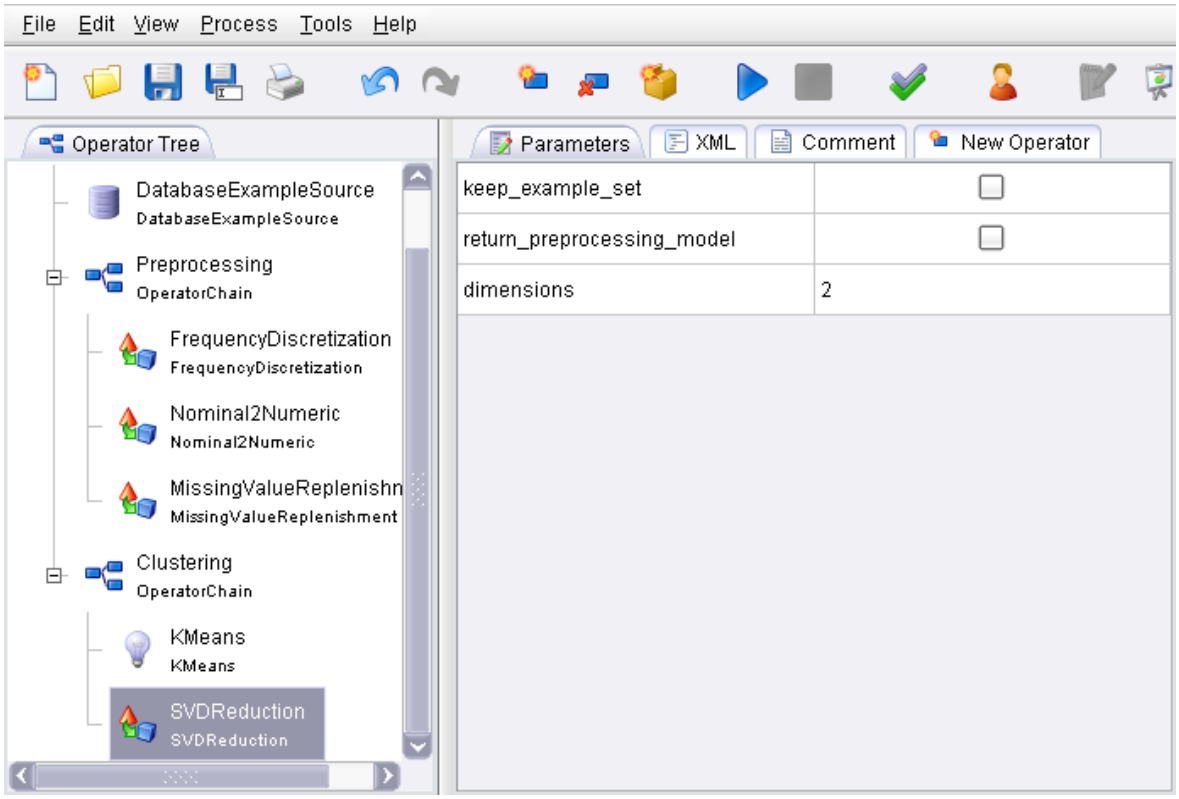


Figura 26: Modelo para a criação de clusters em YALE

5.5 Avaliação da Metodologia

Este estudo utilizado para avaliação da metodologia é constituído por três diferentes tipos de relatórios de TC (Tomografia Computorizada) de três diferentes médicos, disponibilizados em formato html, perfazendo um número total de 100 relatórios médicos TC, de diferente pacientes sem repetição. A informação destes relatórios foi previamente extraída e normalizada no primeiro passo do processo de integração de forma a facilitar o processo de codificação. As idades dos pacientes foram agrupadas nos intervalos de idades seguintes:

- Bebé [0,4]
- Criança [4, 12]
- Jovem [12,26]
- Adulto [27,65]

- Idoso [65,]

Uma vez que estamos a lidar com linguagem natural, em Português, existem muitas possibilidades para representar a mesma palavra, com ou sem acento e em minúsculas ou maiúsculas transformando todas as letras em minúsculas e converte-se os acentos ortográficos para a letra correspondente na codificação ASCII (referencia), por exemplo, a letra "Ç" é convertida em "c".

Na generalidade dos casos, os relatórios médicos contêm um módulo dedicado ao paciente: identificação do paciente, sexo e idade. Este módulo possui o número de séries, modalidades e notas. A seguir a este módulo temos o protocolo ou protocolos e a observação ou observações seguidas de conclusão ou não. Portanto cada conjunto de itens consiste em um ou mais módulos de códigos de paciente, séries, modalidades, notas, protocolos, observações e conclusões. Através desta informação podemos tentar encontrar algumas regras de associação entre os procedimentos seguidos e o diagnóstico.

Apesar do número de relatórios médicos, cada um dos relatórios pode ter muitos tipos, mais de que uma nota, protocolos e observações que aumentam a complexidade do problema para encontrar regras de associação.

5.5.1 Resultados do modelo de criação de regras de associação

Usando o processo para a descoberta de regras de associação em YALE (Figura 25) para o caso de estudo apresentado foram obtidos, por exemplo, alguns conjuntos itens frequentes interessantes na análise do caso de estudo:

- 75% dos pacientes diagnosticados eram mulheres;
- 86% dos relatórios foram considerados "normais";
- 69,4% são considerados "normais" quando não se encontrou desvio médio (901N) e foi encontrado sistema cisterna ventricular permeável (80111).

Apesar de a partir destes relatórios terem sido encontradas muitas regras de associação, para os procedimentos médicos e respectivos diagnósticos, que podiam ser consideradas previsíveis emergiram duas que se destacaram particularmente mais de todas as outras:

Tabela 7: Regras de Associação dos Relatórios Médicos (parte)

Regra	25	59
Permissas	Notas = 30	Protocolo = 20111211
Conclusão	Protocolo = 20111211	Conclusão = C10
Cobertura	61%	75%
Confiança	81,5%	96,4%

As regras da Tabela 7 evidenciam um facto curioso, de que as notas dos relatórios médicos podem desempenhar um papel importante na definição do protocolo e indirectamente influenciar o diagnóstico. Como demonstra a Regra 25 (notas = 30 => protocolo = 20111211 [61%; 81,5%]) temos que 61% da nossa amostra demonstra lombalgias e o protocolo seguido foram cortes axiais contíguos paralelos de 10mm ao plano orbito-meatal com uma confiança de 81,5%. Para a Regra 59 (protocolo = 20111211 => conclusão = C10 [75%, 96,4%]) temos se o protocolo seguido foi cortes axiais contíguos paralelos de 10mm ao plano orbito-meatal em 75% dos casos obtivemos exames normais com uma confiança de 96,4% (um quadro mais completo de regras de associação criadas a partir do modelo pode ser observado no Anexo E).

5.5.2 Resultados do modelo de criação clusters

O principal objectivo da aplicação do modelo de criação de clusters sobre os dados dos relatórios médico era de saber se existia algum grau de similaridade entre eles. Desta forma observaríamos grupos cujos membros são similares entre eles de alguma forma para tentar perceber como se agrupavam. Desta forma garantimos que há uma colecção de relatórios médicos similares entre si e que são diferentes dos relatórios médicos pertencentes a outros clusters.

Na Figura 27, podemos encontrar o resultado do cálculo dos pontos centrais para a nossa amostra de relatórios considerando a seguinte informação:

- Eixo X: Variáveis de cada um dos pontos;
- Eixo Y: Valor das variáveis;

- Cor: Cluster.

Comparando os resultados obtidos, com a Tabela 6, ainda que esta contenha parte dos dados, podemos dizer que o atributo "report4" é um grande centro gravítico para a maioria dos clusters, i.e., o dado que aparece neste atributo é quase sempre igual na maioria dos casos (1001111N – sem conflito de espaço na transição bulbo medular) e é muitas vezes repetido reforçando a sua posição como atributo central, indicando que um dos termos mais utilizados num relatório médico de imagiologia TC é garantir que não ocorre nenhum conflito de espaço na transição bulbo medular. O mesmo tipo de raciocino pode ser aplicado para encontrar outros pontos centrais, ainda que o atributo "report4" seja o que evidencia uma maior importância. Aplicando o mesmo tipo de raciocínio anterior podemos considerar como pontos centrais os atributos "report1" e "modalidades".

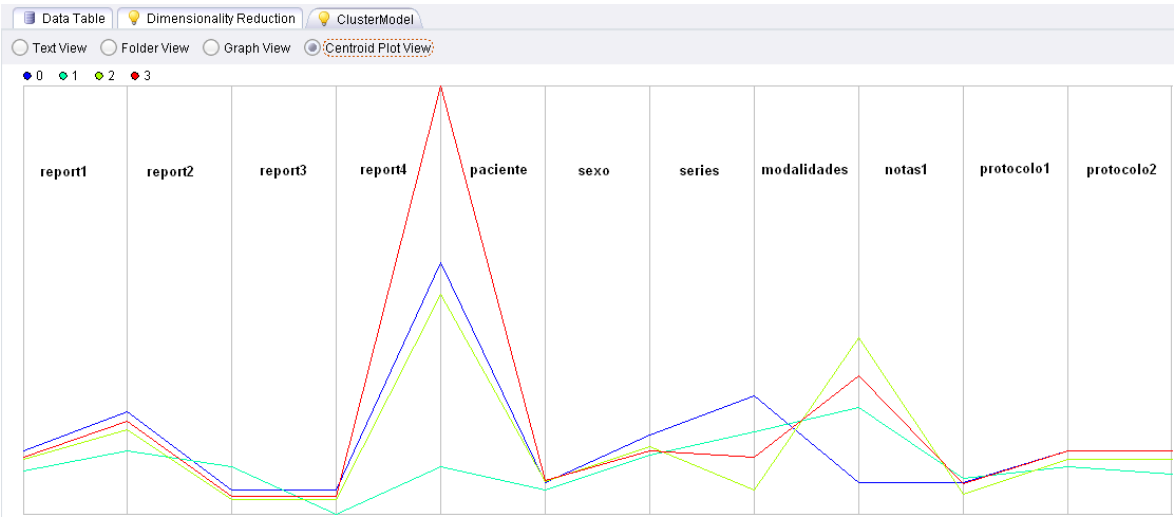


Figura 27: Pontos centrais dos relatórios médicos

Com base nestes pontos centrais podemos construir uma representação gráfica dos clusters (Figura 28). A partir da Figura 28 é evidenciado que o maior número de resultado "normais" encontra-se no cluster 0 e o que apresenta um maior número de resultados "anormais" é o cluster 1. Ora uma vez já se conclui que o atributo "report4" é o que evidência uma maior importância, pela positiva, a

falta de peso no cluster 1 poderá indicar que muitos dos seus exames não são considerados "normais". Em relação ao cluster 0, o facto de todos os seus relatórios serem normais pode indicar que se trata de um cluster que obteve um agrupamento de relatório normais "optimizado" a partir da informação disponibilizada pelos mesmos.

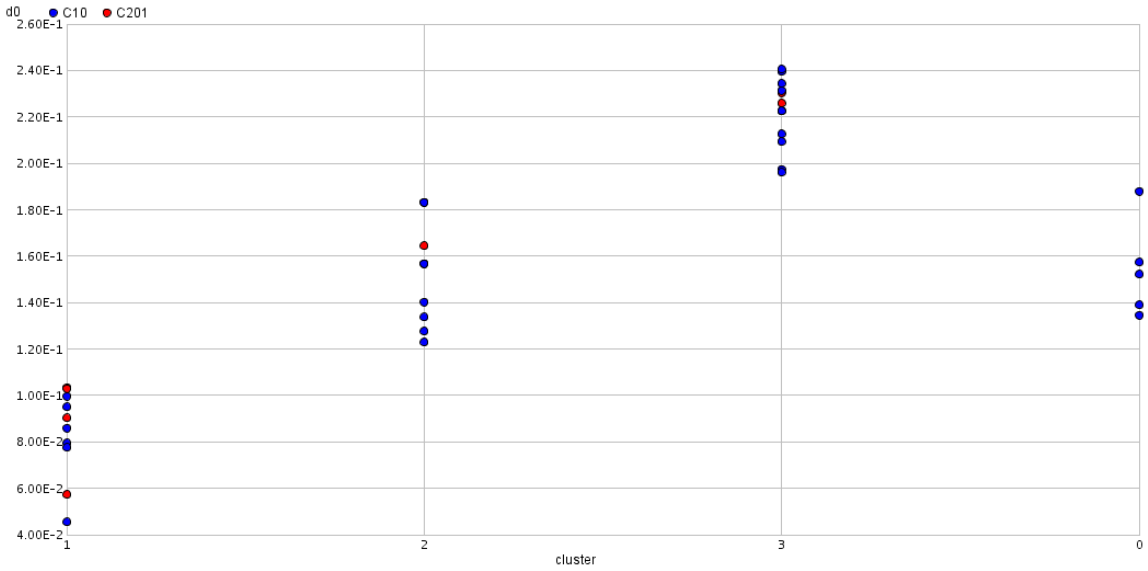


Figura 28: Resultado da aplicação algoritmos de clustering

Capítulo 6

Conclusões e Trabalho Futuro

6.1 Avaliação do trabalho

Nesta secção são avaliadas as etapas, opções e suas percussões, que permitem retirar diversas conclusões detalhadas sobre este trabalho.

Para a realização deste trabalho, foi efectuado um estudo, sobre as normas de criação e estruturação de um relatório médico bem como da possibilidade de descobrir conhecimento a partir da informação contida nestes.

Ao longo do processo tornou-se imperativo conhecer e entender melhor as normas DICOM SR e HL7 CDA, o seu conteúdo, a sua aplicabilidade e usabilidade no âmbito deste trabalho. As vantagens e desvantagens apresentadas resultaram do conhecimento reunido a partir da interacção com as mesmas, assim como de casos de outros trabalhos. Depois deste estudo, da definição do seu âmbito em termos de trabalho, foi decidido utilizar bibliotecas, como a DCMTK, que são universalmente aceites e com provas dadas na comunidade cientifica, permitindo assim aceder à informação contida nos relatórios normalizados sem precisar de um conhecimento profundo da norma. Contudo, as bibliotecas não resolviam alguns dos problemas que foram detectados no estudo das normas. Um dos mais importantes era o problema da transmissão entre diferentes

sistemas e conversão dos relatórios médicos em outros formatos. Esta diferença de protocolos e mensagens existentes nos sistemas e nos formatos levou à procura de soluções que respondessem a estas necessidades.

Esse estudo foi realizado na perspectiva "out of the box", i.e., procurar soluções em outras áreas de conhecimento, onde foi verificado que os problemas mais comuns identificados em aplicações de integração no mundo empresarial são os protocolos e formatos de mensagens incompatíveis. Como resposta a este tipo de problemas, a actual opção que a indústria tem seguido baseia-se nas definições de normas para integração de processos de negócio e dados normalizados na pilha de serviços Web. Muitas das vezes as empresas utilizam diferentes sistemas e protocolos de transmissão da informação para disponibilizarem os seus serviços aos seus clientes, ou internamente possuindo diferentes aplicações. Como resposta a esta problemática foi desenvolvida uma arquitectura orientada a serviços, normalizando a sua concepção, desenvolvimento e implementação através da norma JBI, permitindo que cada empresa desenvolvesse os seus próprios componentes, em concordância com a norma JBI, para transmissão e conversão de diferentes formatos assim como do uso de diferentes protocolos e mensagens entre diferentes sistemas de diferentes empresas através da JBI.

Foi necessário um estudo aprofundado da especificação JBI para compreender cada componente, utilidade, aplicabilidade e escalabilidade no âmbito deste trabalho. No entanto, o estudo da especificação JBI permitiu concluir que por si só, a mesma não define um único ponto comum de administração de todo o sistema. Cada operação no sistema necessita de conhecimento prévio da topologia do sistema e não pode ser considerada uma plataforma de trabalho para desenvolver um serviço de integração orientada a serviços.

O passo seguinte consistiu em procurar para estes problemas uma solução que foi encontrada na plataforma Open-ESB. O Open-ESB resolve estes problemas utilizando o Java Open Entreprise Service Bus (ESB) construído com tecnologia JBI, possibilitando que um conjunto distribuído de instâncias JBI possam comunicar como uma única identidade lógica e que pode ser centralmente gerida. Através da plataforma Open-ESB é possível integrar funções de negócio existentes como serviços e desacoplar interacções entre os fornecedores de serviço e consumidores. A plataforma

Open-ESB fornece suporte directo para criar aplicações compostas através do mecanismo JBI de montagem de serviços. Este suporte permite que as aplicações sejam compostas directamente do serviço base de interfaces disponibilizadas pelo JBI como serviços unitários, e usando BPEL (Business Process Execution Language) para orquestração de processos. Para além disso, a escolha desta plataforma também se baseou no facto de estar em plena expansão, suportada por uma grande comunidade de especialistas, de diferentes áreas de conhecimento possuindo diversos componentes de diferentes áreas de aplicação. Dos seus componentes destaca-se o componente de ligação para HL7 que em outras plataformas semelhantes ainda não foi abordado, ou ainda se encontra numa fase inicial de desenvolvimento. Uma outra vantagem apresentada está no componente de ligação de ficheiros, para a normalização de ficheiros de texto, que permite definir regras para extracção da informação dos relatórios, definindo uma estratégia de parsing para os ficheiros utilizando o tipo de formato (template) usado por um médico.

Resolvido o problema da normalização, o próximo passo foi a codificação dos termos médicos. A codificação desenvolvida para os termos médicos torna a informação mais legível, facilita a leitura, a aplicação de algoritmos de mineração de dados e representação da negação de um termo. No entanto, esta codificação não se compara ao nível de um sistema de codificação como o SNOMED-CT, organizada de forma sistemática, já com provas dadas na comunidade cientifica e que abrange a maior parte das áreas de informação clínica como doenças, evidências clínicas, procedimentos, entre outros.

Com a aplicação dos algoritmos de mineração de dados, como os da criação de regras de associação ou de clusters, obtiveram-se resultados quer considerados expectáveis, bem como resultados que só a partir de uma ferramenta de mineração de dados podiam ser descobertos. Ainda que a amostra de relatórios cedidos pelo CIT não sejam considerados um grande volume de dados, as variáveis a ser consideradas e que existem num relatório são inúmeras. Neste caso, o uso da identificação e a divisão genérica em dados demográficos do paciente, series, modalidades, notas, protocolos, evidências (reports) e conclusão permitiram comprovar o uso da metodologia com sucesso.

6.2 Considerações Finais

A metodologia de integração orientada a serviços para a descoberta de conhecimentos em relatórios médicos aqui proposta constitui-se como uma resposta a problemas encontrados no processamento dos relatórios herdados, possibilitando a descoberta de conhecimento usando standards abertos de indústria, negócio e médicos.

Foi uma metodologia concebida para solucionar os problemas típicos apresentados pelos relatórios médicos (e.g., termos de língua natural ambíguos, falta de estrutura, difícil intercomunicação inter-aplicações relacionadas com lógica de negócio, protocolos e formatos de mensagens incompatíveis).

A normalização dos relatórios baseados nos formatos habituais dos médicos especialistas provou ser uma excelente abordagem na análise sintáctica e semântica dos relatórios, uma vez que cada médico tem geralmente o seu próprio formato facilitando deste modo a conversão da informação para um formato estruturado. Numa primeira abordagem foi aplicada a transformação XSLT ao XML Schema dsr2xml.xsd [Web 18] com a estrutura normalizada a partir dos templates dos médicos especialistas em XSD. No entanto, verificou-se que nos relatórios herdados (i.e., relatórios já existentes em texto livre) faltavam informações chave necessárias na estrutura da OFFIS.

A decisão de codificar os termos médicos com um mapeamento interno, definindo um serviço para esta codificação, sem usar codificações existentes de bibliotecas proprietárias (e.g., SNOMED-CT, ACR, LOINC) resolve o problema dos termos ambíguos e negação, utilizando um código específico do conceito, nome ou valor. Esta noção de negação é crucial porque poderão aparecer uns textos em que se afirma "não foi encontrada atrofia" e outros em que se afirma "foi encontrada atrofia".

Por fim, os resultados obtidos a partir da informação integrada demonstram situações óbvias (e.g., "as fracturas são mais comuns em pessoas idosas", "os exames considerados normais são mais comuns em pessoas jovens"). Contudo, através das regras encontradas também foi possível concluir que as notas assentes sobre o paciente, o seu estado físico, por exemplo, antes de efectuar o exame influenciaram o protocolo médico seguido, bem como indirectamente o diagnóstico. A partir do modelo de criação de cluster foi possível concluir, para os exames que construíram a

nossa amostra, que o atributo mais importante a ser referido num exame de imagiologia médica TC trata-se da evidência clínica da existência ou não de conflito de espaço na transição bulbo medular.

6.3 Trabalho Futuro

Actualmente este trabalho encontra-se ainda em avaliação, alguns dos componentes encontram-se em fase de melhoramento ou são trabalhos a serem atribuídos a futuros investigadores, sendo de salientar alguns dos aspectos relativos nos quais se está a trabalhar:

- OFFIS dsr2xml.xsd – mapear toda a informação dos relatórios médicos herdados para esta estrutura. Para futuras conversões do conteúdo de um documento DICOM Structures Reporting (SR) (em ficheiro ou em dados brutos) para XML;
- Desenvolvimento de um componente ligação DICOM Open-ESB – actualmente a comunicação em DICOM é apenas possível por TCP/IP;
- Melhoramento do sistema de codificação, de termos médicos, com a inclusão de novos termos médicos que não se encontraram nos exames que serviram de estudo a este trabalho;
- Melhorar os processos BPEL a nível de tratamento de erros;
- Processador Inteligente de Eventos Open-ESB – actualmente o pré-processamento é efectuado recorrendo ao YALE. Utilização do IEP reduziria a carga do YALE;
- Mineração de Dados Geograficamente – permitir o estudo epidémico sobre os relatórios médicos.

Referências

[Behlen 2007] Behlen, Fred M. (2007). "DICOM Structured Reporting and the CDA Tutorial", HL7 International CDA Conference, Berlim.

[Berson & Stephen 1997] Berson, Alex, and Stephen J. Smith (1997). "Introduction to Data Mining". In Data Warehousing, Data Mining, & OLAP. Nova Yorque: McGraw-Hill, 1997.

[Bortoluzzi 2003] Bortoluzzi, Mariana Kessler (2003). "Desenvolvimento e Implementação de um Editor de Documentos Estruturados no padrão DICOM Structured Report", Universidade Federal de Santa Catarina, Florianópolis.

[Brachman & Anand 1996] Brachman, R.J., and Anand, T. (1996). "The Process Of Knowledge Discovery In Databases: A Human-Centered Approach". In Advances In Knowledge Discovery And Data Mining, eds. U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, AAAI Press/The MIT Press, Menlo Park, CA., 1996, pp. 37-57.

[Chappel 2004] Chappel, D. (2004). "Enterprise Service Bus: Theory in Practice", O' Reilly Media, 2004.

[Clunie 2007] Clunie, David A (2007). "DICOM Structured Reporting and Cancer Clinical Trials Results", Cancer Informatics 2007:4 33-56.

[Clunie, David A 2001] Clunie, David A (2001). “DICOM Structured Reporting”, PixelMed Publishing, Bangor, Pensilvânia. ISBN 0-9701369-0-0.

[Dolin et al. 2006] Dolin et al. (2006). “HL7 Clinical Document Architecture, Release 2 “. In Journal of the American Medical Informatics Association, volume 13.

[Dreyer, Mehta & Thrall 2002] Dreyer, Keith J., Mehta, Amit, Thrall, James H. (2002). “PACS: a guide to the digital revolution”, Springer. Nova York.

[Fayyad, Piatetsky-Shapiro & Smyth 1996] Fayyad, U.M., Piatetsky-Shapiro, G., and Smyth, P. (1996). “From Data Mining To Knowledge Discovery: An Overview”. In Advances In Knowledge Discovery And Data Mining, eds. U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, AAAI Press/The MIT Press, Menlo Park, CA., 1996, pp. 1-34.

[Frawley, Piatetsky-Shapiro & Matheus 1991] Frawley, W.J., Piatetsky-Shapiro, G., and Matheus, C. (1991). “Knowledge Discovery In Databases: An Overview. In Knowledge Discovery In Databases”, eds. G. Piatetsky-Shapiro, and W. J. Frawley, AAAI Press/MIT Press, Cambridge, MA., 1991, pp. 1-30.

[Han & Kamber 2001] Han, Jiawei, and Micheline Kamber (2001). “Applications and Trends in Data Mining”. In Data Mining: Concepts and techniques. São Francisco: Morgan Kaufmann Publishers.

[Han, Pei & Yin 2000] Han, J., Pei, J., Yin, Y. (2008). “Mining frequent patterns without candidate generation”. In Proc. 2000 ACM-SIGMOD Int. Conf. Management of Data (SIGMOD'00), pages 1-12, Dallas, TX.

[Hansen, Roseli Persson et al 2001] Hansen, Roseli Persson et al (2001). “Web services: an architectural overview”. International Conference on Formal Ontology in Information Systems, 2001. Ogunquit. New York: ACM, 2001. p. 297-308.

[Holman, Aliabadi, P. Silverman, et al 1994] Holman, B., Aliabadi, P. Silverman, S., et al. (1994). “Medical impact of unedited preliminary radiology reports”. Radiology 191:519–521.

[Hussein, Engelmann, Schröter & Meinzer 2004] Hussein R, Engelmann U, Schröter A, Meinzer HP. (2004). “DICOM Structured Reporting: Part 2. Problems and Challenges in Implementation for PACS Workstations”. Radiographics. 2004 Maio-Junho;24(3):897-909.

[Jennings & Salter 2008] Jennings, Frank and Salter, David (2008). “Netbeans enterprise pack: building SOA applications”. Packt Pub Ltd. ISBN 1847192629.

[Jordan & Evdemon 2007] D. Jordan, J. Evdemon (2007). “Web Services Business Process Execution Language Version 2.0”. OASIS Standard, Abril, 2007.

[Kahn, Carrino, Flynn, Peck & Horii 2007] Kahn, C. E., Carrino, J. A., Flynn, M. J., Peck, D. J., & Horii, S. C. (2007). “DICOM and Radiology: Past, Present and Future”. Journal of the American College of Radiology, Setembro 2007 (Vol.4, Issue 9, Pag. 652-657).

[Kong, Barnett, Mosteller, et al 1986] Kong, A., Barnett, G., Mosteller, F., et al (1986). “How medical professionals evaluate expressions of probability”. N Engl J Med; 315:740–744.

[Langlotz 2002] Langlotz, C.P. (2002). “Automatic Structuring of Radiology Reports: Harbinger of a Second Information Revolution in Radiology”, Radiology 224:5-7.

[Lussier, Shagina & Friedman 2001] Lussier, Y. A., Shagina, L., Friedman, C. (2001). "Automating SNOMED coding using medical language understanding: a feasibility study". Proc AMIA Symp, 418–422.

[Mierswa et al 2006] Mierswa et al (2006). "YALE: Rapid Prototyping for Complex Data Mining Tasks", in Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-06).

[Naik, Hanbidge & Wilson 2001] Naik, S., Hanbidge, A., Wilson, S. (2001). "Radiology reports: examining radiologist and clinician references regarding style and content". AJR (American Journal of Roentgenology) 176:591-598.

[Noumeir 2006] Noumeir, Rita (2006). "Benefits of the DICOM Structured Report", Journal of Digital Imaging, 295-306, Springer.

[Paterson, Shepherd, Wang, Watters, & Zitner 2006] Paterson, G. I., Shepherd, M., Wang, X., Watters, C., & Zitner, D. (2002). "Using the XML-based Clinical Document Architecture for Exchange of Structured Discharge Summaries". In 35th Hawaii International Conference on System Sciences, Pag. 1200 – 1209, Halifax, Canada.

[Pinheiro & Alves 2008] Pinheiro, Vitor & Alves, Victor (2008). "A Service Oriented Integration Methodology for Knowledge Discovery in Medical Reports". Aprovado para apresentação e publicação na International Conference on Agents and Artificial Intelligence, Porto, 2009.

[Prather et al 1997] Prather, J. C., Lobach, D. F., Goodwin, L. K., Hales, J.W., ML Hage, Hammond, W. E. (1997). "Medical Data Mining: Knowledge Discovery in a Clinical Data Warehouse", Proceedings of the AMIA Annual Fall Symposyum, Vol. 101, No. 5.

[Reiner, Knight & Siegel 2007] Reiner, B. I., Knight, N., & Siegel, E. L. (2007). “Radiology Reporting, Past, Present, and Future: The Radiologist’s Perspective”. Journal of the American College of Radiology

[Rosenberg, Florian, Dustdar & Schahram 2005] Rosenberg, Florian and Dustdar, Schahram (2005). “Business Rule Integration in BPEL – A Service-Oriented Approach”. Proceedings of the 7th International IEEE Conference on E-Commerce Technology.

[Silva 2004] Marcelino Pereira dos Santos Silva (2004). "Mineração de Dados – Conceitos, Aplicações e Experimentos com Weka". Universidade do Estado do Rio Grande do Norte.

[Thearling 2003] Thearling, Kurt (2003). "An Introduction to Data Mining: Discovering Hidden Value in Your Data Warehouse". In Data Mining and Analytic Technologies.

[Witten & Eibe 2005] Witten, Ian H. and Frank, Eibe (2005). “Data Mining: Practical machine learning tools and techniques”, 2nd Edition, Morgan Kaufmann, São Francisco.

[Witten & Frank 2005] Witten, I & Frank, E. (2005). “Data Mining: Practical Machine Learning Tools and Techniques”. Morgan Kaufmann Series in Data Management Systems. Amesterdão: Morgan Kaufmann, 2ª edição, pp. 61-82, 96-105, 112-119, 136-139, 144-157.

Referências WWW

[Web 1] http://www.nema.org/

NEMA é uma associação americana de indústrias eléctricas criada em 1 de Setembro de 1926 e define diversas normas para aparelhos electrónicos utilizados em aplicações industriais (acedido em 2008/09/01).

[Web 2] http://www.hl7.org

Comunidade internacional sem fins lucrativos constituída por profissionais e técnicos de saúde e de outras áreas do conhecimento que colaboram no desenvolvimento de normas, como a própria norma HL7, para a troca, gestão e integração da informação clínica na sua forma electrónica (acedido em 2008/09/01).

[Web 3] http://medical.nema.org/

DICOM é uma norma para manuseamento, armazenamento, impressão e transmissão da informação de imagens médicas, incluindo um formato de ficheiro e um protocolo de comunicação em rede (acedido em 2008/09/01).

[Web 4] http://www.acr.org/

Colégio Americano de Radiologistas, fundado 1923, é uma organização sem fins lucrativos de profissionais na área da medicina de diversas especialidades como de radiologia, oncologia, entre outras. Tem a sua sede em Reston, no estado da Virginia, nos Estados Unidos da

América. Este colégio é responsável por duas publicações The Journal of the American College of Radiology (JACR) e The ACR Bulletin (acedido em 2008/09/01).

[Web 5] http://www.ansi.org/

ANSI é uma organização sem fins lucrativos que tem como objectivo padronizar as normas usadas em produtos, serviços, processos, entre outros aspectos nos Estados Unidos da América (acedido em 2008/09/01).

[Web 6] http://www.ieee.org

O IEEE é uma associação profissional sem fins lucrativos, que é líder mundial para o desenvolvimento tecnológico (acedido em 2008/09/09).

[Web 7] http://en.wikipedia.org/wiki/Picture_archiving_and_communication_system

Sistema ou redes de computadores dedicados para armazenamento, retorno, distribuição ou visualização de imagens médicas (acedido em 2008/09/09).

[Web 8] https://open-esb.dev.java.net/public/whitepapers/JBIforSOI.pdf

Neste documento Ron Ten-Hove (Sun Microsystems) introduz a especificação JBI como fundação de SOA e como esta pode ser orientada para um caso ou arquitectura específica (SOI) através de alguns exemplos de aplicação (acedido em 2008/09/09).

[Web 9] https://open-esb.dev.java.net/

Site oficial da plataforma OpenESB (100% Código Livre) para integração de processos de negócio, aplicações empresariais e arquitecturas orientadas a serviços (acedido em 2008/09/15).

[Web 10] http://www.ihtsdo.org/snomed-ct/

O Systematized Nomenclature of Medicine Clinical Terms (SNOMED CT) é uma colecção informática, organizada de forma sistemática, de terminologia médica, que abrange a maior parte das áreas de informação clínica como doenças, evidências clínicas, procedimentos, entre outros. Permitindo de forma consistente indexar, armazenar, pesquisar e agregar informação clínica de várias especialidades e áreas de cuidados médicos (acedido em 2008/09/15).

[Web 11] http://jcp.org/en/jsr/detail?id=208

Site oficial da especificação Java Business Integration (JBI) que é construída de acordo com o modelo de serviços Web e fornece uma arquitectura de ligação para um container de serviços e componentes. Desenvolvida em acordo com a Java Comnuity Process (JCP) como abordagem para a implementação de uma Arquitectura Orientada a Serviços (AOS) (acedido em 2008/10/04).

[Web 12] http://java.sun.com/developer/technicalArticles/WebServices/soa3/ImplementingSOA.pdf

Um documento que aborda a implementação de uma Arquitectura Orientada a Serviços com a Java EE 5 SDK. Neste documento são apresentadas especificações Java como a Java Business Integration (JBI) – JSR 208, os componentes de ligação e motores de serviço disponibilizados em termos de plataforma para integração, neste caso, Open-ESB, proposta pela Sun Microsystems, interligação entre a plataforma e a especificação bem como algumas conclusões acerca das suas aplicações (acedido em 2008/10/06).

[Web 13] http://www.oasis-open.org/home/index.php

A Organization for the Advancement of Structured Information Standards – OASIS é um consórcio global que tem como objectivo principal orientar o desenvolvimento, convergência e adopção normas para serviços Web e negócios electrónicos (acedido em 2008/10/06).

[Web 14] http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.pdf

Documento de especificação da Web Services Business Process Execution Language, versão 2, por parte da Organization for the Advancement of Structured Information Standards (acedido em 2008/10/06).

[Web 15] http://www.oasis-open.org/committees/download.php/23070/The%20Business%20Value%20of%20WS-BPEL%20for%20Business%20Analysts%20and%20Managers%20(Frank%20Leymann)%20-%20Part%201.pdf

A Organization for the Advancement of Structured Information Standards – OASIS é um consórcio global que orienta o desenvolvimento, convergência e adopção normas para serviços Web e negócios electrónicos. Através desta apresentação de um webinar, do Prof. Dr. Frank Leyman, é apresentado, em resumo, o valor acrescido da utilização de BPEL em processos de negócio, qual o seu papel e utilização numa Arquitectura Orientada a Serviços (acedido em 2008/10/06).

[Web 16] http://dret.net/lectures/services-fall06/img/bpel-process.gif

Esquematização de um possível processo BPEL (acedido em 2008/10/20).

[Web 17] http://downloads.sourceforge.net/yale/rapidminer-4.2-tutorial.pdf

Tutorial do RapiMiner formalmente conhecido como Yet Another Learning Environment – YALE de aprendizagem computacional e mineração de dados (acedido em 2008/10/20).

[Web 18] http://www.offis.de/

Site oficial do Oldenburger Forschungs- und Entwicklungsinstitut für Informatik-Werkzeuge und –Systeme – OFFIS. Trata-se de um instituto orientado à investigação e desenvolvimento científico, como centro de excelência, para tecnologias de informação e suas aplicações, localizado na cidade de Oldenburg, na Alemanha, fundado em 1991 e que tem uma estreita ligação com a universidade local. Actualmente as principais competências do centro em termos de

investigação e desenvolvimento tecnológico são a energia, saúde e transportes (acedido em 2008/10/20).

[Web 19] http://dicom.offis.de/dcmtk

O DCMTK é uma colecção de bibliotecas e aplicações que implementam a maior parte da norma DICOM. Esta colecção de bibliotecas e inclui aplicações, escritas em uma mescla de ANSI C e C++, para examinar, construir e converter ficheiros de imagem médica DICOM, manuseamento de dados, envio e recepção de imagens através de uma rede, entre outras (acedido em 2008/10/20).

[Web 20] http://support.dcmtk.org/docs/dsr2xml.html

Trata-se de uma aplicação que converte o conteúdo de um DICOM SR (formato de ficheiro ou conjunto de dados) para XML apesar do XML Schema que esta aplicação possui, dsr2xml.xsd, não seguir ainda nenhuma norma (acedido em 2008/10/20).

[Web 21] http://www.snpnet.com/customer_pub/sun/SOA_OpenESB/

Neste webinar podemos encontrar uma visão geral, de Amol Khire, Lei Liu e Ariane Naderzad, da Sun Microsystems, sobre o desenvolvimento das estratégias de integração da informação num ambiente empresarial, a evolução para uma Arquitectura Orientada a Serviços, especificação JBI, a plataforma de trabalho para integração apresentada pelo projecto Open-ESB que se baseia em JBI e a orquestração de serviços Web em BPEL.

[Web 22] http://www.ihe-uk.org

O Integrating the Healthcare Enterprise – IHE é uma organização internacional criada por profissionais de saúde e de indústria com o objectivo de melhorar a forma como sistemas informáticos usados na área da saúde partilham informação através de processos de interoperabilidade possibilitando a integração de informação médica em diferentes sistemas,

sem a necessidade da intervenção humana, reduzindo deste modo os custos e o risco de ocorrer um erro humano (acedido em 2008/10/20).

[Web 23] http://www.ihe-uk.org/cgi-bin/forum/show.cgi?3450/3452

Ilustração da arquitectura apresentada pela IHE (acedido em 2008/10/20).

[Web 24] http://hl7book.net/index.php?title=CDA

Site com informação de referência sobre a norma HL7 CDA (Electronic Medical Summary (e-MS) Project: Phase 2) (acedido em 2008/09/05).

[Web 25] http://www.miforum.net/distillate/rim/RIM0112_body.htm

O principal objectivo do HL7 Reference Information Model é de tornar a informação significativa para além do contexto local, i.e., o âmbito do HL7 RIM está na informação necessária para ser compreendida entre sistemas de informação mas não necessariamente toda a informação que se encontra num dos sistemas. Neste site podemos encontrar uma introdução ao modelo bem como uma visão geral, áreas abrangidas, as classes e associações de informação que formam o modelo (acedido em 2008/09/05).

[Web 26] http://www.gillogley.com/hl7_pmi.shtml

Neste site, de uma empresa especialista em integração de aplicações, na área da saúde, podemos encontrar algumas das dificuldades que já se depararam durante a sua actividade profissional na implementação da norma HL7 (acedido em 2008/09/05).

[Web 27] http://www.w3.org/TR/ws-arch/

Neste documento, podemos encontrar a definição, por parte do W3C, da arquitectura dos serviços Web, identificação dos componentes funcionais e definição do relacionamento entre esses componentes (acedido em 2008/10/06).

[Web 28] http://java.sun.com/developer/technicalArticles/J2EE/sunjavaee_engine/

Este artigo descreve a sinergia entre o Motor de Serviço Sun Java EE e a Java EE no ambiente JBI, abordando conceitos e características do Motor Serviço Sun Java EE e através de exemplos demonstra a forma de utilização num ambiente JBI (acedido em 2008/10/20).

[Web 29] http://www.dfki.de/~kipp/seminar_ws0607/reports/ArndtFaulhaber.pdf

Este artigo de Arndt Faulhaber, da Universidade de Saarland, aborda uma introdução da ferramenta de aprendizagem computacional denominada YALE (Yet Another Learning Environment) bem como alguns aspectos sobre a ferramenta, usabilidade e aplicação (acedido em 2008/10/06).

[Web 30] http://www.cs.waikato.ac.nz/ml/weka

A Weka é uma colecção de algoritmos de extracção de conhecimento para mineração de dados. Os algoritmos podem ser aplicados directamente no conjunto de dados ou invocados a partir de código Java. A Weka também possui ferramentas para pré-processamento, regressão, clustering, regras de associação e visualização de dados (acedido em 2008/10/06).

[Web 31] http://rapid-i.com

A ferramenta YALE (Yet Another Learning Environment) como formalmente é conhecida é uma plataforma (ambiente) para efectuar experiencias com algoritmos mineração de dados e descoberta de conhecimento. Permite realizar testes com um grande volume de dados e de operadores, descrevendo e armazenando o processo em XML possibilitando a reutilização. O YALE herdou muitos dos algoritmos e ferramentas disponibilizadas pela Weka. Actualmente, esta ferramenta possui uma versão comercial e outra aberta à comunidade denominada RapidMiner (acedido em 2008/10/06).

[Web 32] http://www.infopedia.pt

Dicionário online de Português da Porto Editora (acedido em 2008/10/06).

Anexos

A. DICOM SR

Os exemplos que se seguem foram retirados de [Clunie, David A 2001] e servem para exemplificar a forma como a informação de um possível relatório médico estruturado DICOM SR é representado pela norma DICOM.

Exemplo de uma representação DICOM SR proposta por Clunie

<CONTAINER:(,,“Chest X-Ray Report”)>

<contains TEXT:(,,“Procedure”)=“PA and lateral”>

<contains CONTAINER:(,,“Findings”)>

<contains TEXT:(,,“Finding”)=“Multiple masses”>

<contains CONTAINER:(,,“Conclusions”)>

<contains CONTAINER:(,,“Impressions”)>

<contains TEXT:(,,“Impression”)=“Metastases”>

<contains CONTAINER:(,,“Recommendations”)>

<contains TEXT:(,,“Recommendation”)=“Biopsy”>

<contains CONTAINER:>

<contains CODE:(,,“Diagnosis”)=(197.0,I9C,“Secondary malignant neoplasm of lung”)>

Representação completa da informação em DICOM

(0040,a040) Value Type “CONTAINER”

(0040,a043) Concept Name Code Sequence

(fffe,e000) Item

(0008,0100) Code Value “209076”

(0008,0102) Coding Scheme Designator “99PMP”

(0008,0104) Code Meaning “Chest X-ray Report”

(fffe,e00d) Item Delimitation Item

(fffe,e0dd) Sequence Delimitation Item

(0040,a050) Continuity of Content “SEPARATE”

(0040,a730) Content Sequence

(fffe,e000) Item

(0040,a010) Relationship Type “CONTAINS”

(0040,a040) Value Type “TEXT”

(0040,a043) Concept Name Code Sequence

(fffe,e000) Item

(0008,0100) Code Value “209007”

(0008,0102) Coding Scheme Designator “99PMP”

(0008,0104) Code Meaning “Procedure”

(fffe,e00d) Item Delimitation Item

(fffe,e0dd) Sequence Delimitation Item

(0040,a160) Text Value “PA and Lateral”

(fffe,e00d) Item Delimitation Item

(fffe,e000) Item

(0040,a010) Relationship Type “CONTAINS”

(0040,a040) Value Type “CONTAINER”

(0040,a043) Concept Name Code Sequence

(fffe,e000) Item

(0008,0100) Code Value “209002”

(0008,0102) Coding Scheme Designator “99PMP”

(0008,0104) Code Meaning “Findings”

(fffe,e00d) Item Delimitation Item

(fffe,e0dd) Sequence Delimitation Item

(0040,a050) Continuity of Content “SEPARATE”

(0040,a730) Content Sequence

(fffe,e000) Item

(0040,a010) Relationship Type “CONTAINS”

(0040,a040) Value Type “TEXT”

(0040,a043) Concept Name Code Sequence

(fffe,e000) Item

(0008,0100) Code Value “209001”

(0008,0102) Coding Scheme Designator “99PMP”

(0008,0104) Code Meaning “Finding”

(fffe,e00d) Item Delimitation Item

(fffe,e0dd) Sequence Delimitation Item

(0040,a160) Text Value “Multiple masses”

(fffe,e00d) Item Delimitation Item

(fffe,e0dd) Sequence Delimitation Item

(fffe,e00d) Item Delimitation Item

(fffe,e000) Item

(0040,a010) Relationship Type “CONTAINS”

(0040,a040) Value Type “CONTAINER”

(0040,a043) Concept Name Code Sequence

(fffe,e000) Item

(0008,0100) Code Value “209006”

(0008,0102) Coding Scheme Designator “99PMP”

(0008,0104) Code Meaning “Conclusions”

(fffe,e00d) Item Delimitation Item

(fffe,e0dd) Sequence Delimitation Item

(0040,a050) Continuity of Content “SEPARATE”

(0040,a730) Content Sequence

(fffe,e000) Item

(0040,a010) Relationship Type “CONTAINS”

(0040,a040) Value Type “CONTAINER”

(0040,a043) Concept Name Code Sequence

(fffe,e000) Item

(0008,0100) Code Value “209013”

(0008,0102) Coding Scheme Designator “99PMP”

(0008,0104) Code Meaning “Impressions”

(fffe,e00d) Item Delimitation Item

(fffe,e0dd) Sequence Delimitation Item

(0040,a050) Continuity of Content “SEPARATE”

(0040,a730) Content Sequence

(fffe,e000) Item

(0040,a010) Relationship Type “CONTAINS”

(0040,a040) Value Type “TEXT”

(0040,a043) Concept Name Code Sequence

(fffe,e000) Item

(0008,0100) Code Value “209014”

(0008,0102) Coding Scheme Designator “99PMP”

(0008,0104) Code Meaning “Impression”

(fffe,e00d) Item Delimitation Item

(fffe,e0dd) Sequence Delimitation Item

(0040,a160) Text Value “Metastases”

(fffe,e00d) Item Delimitation Item

(fffe,e0dd) Sequence Delimitation Item

(fffe,e00d) Item Delimitation Item

(fffe,e000) Item

(0040,a010) Relationship Type “CONTAINS”

(0040,a040) Value Type “CONTAINER”

(0040,a043) Concept Name Code Sequence

(fffe,e000) Item

(0008,0100) Code Value “209015”

(0008,0102) Coding Scheme Designator “99PMP”

(0008,0104) Code Meaning “Recommendations”

(fffe,e00d) Item Delimitation Item

(fffe,e0dd) Sequence Delimitation Item

(0040,a050) Continuity of Content “SEPARATE”

(0040,a730) Content Sequence

(fffe,e000) Item

(0040,a010) Relationship Type “CONTAINS”

(0040,a040) Value Type “TEXT”

(0040,a043) Concept Name Code Sequence

(fffe,e000) Item

(0008,0100) Code Value “209016”

(0008,0102) Coding Scheme Designator “99PMP”

(0008,0104) Code Meaning “Recommendation”

(fffe,e00d) Item Delimitation Item

(fffe,e0dd) Sequence Delimitation Item

(0040,a160) Text Value “Biopsy”

(fffe,e00d) Item Delimitation Item

(fffe,e0dd) Sequence Delimitation Item

(fffe,e00d) Item Delimitation Item

(fffe,e0dd) Sequence Delimitation Item

(fffe,e00d) Item Delimitation Item

(fffe,e000) Item

(0040,a010) Relationship Type “CONTAINS”

(0040,a040) Value Type “CONTAINER”

(0040,a050) Continuity of Content “SEPARATE”

(0040,a730) Content Sequence

(fffe,e000) Item

(0040,a010) Relationship Type “CONTAINS”

(0040,a040) Value Type “CODE”

(0040,a043) Concept Name Code Sequence

(fffe,e000) Item

(0008,0100) Code Value “209017”

(0008,0102) Coding Scheme Designator “99PMP”

(0008,0104) Code Meaning “Diagnosis”

(fffe,e00d) Item Delimitation Item

(fffe,e0dd) Sequence Delimitation Item

(0040,a168) Concept Code Sequence

(fffe,e000) Item

(0008,0100) Code Value “197.0”

(0008,0102) Coding Scheme Designator “I9C”

(0008,0104) Code Meaning “Secondary malignant neoplasm of lung”

(fffe,e00d) Item Delimitation Item

(fffe,e0dd) Sequence Delimitation Item

(fffe,e00d) Item Delimitation Item

(fffe,e0dd) Sequence Delimitation Item

(fffe,e00d) Item Delimitation Item

(fffe,e0dd) Sequence Delimitation Item

Representação gráfica do exemplo

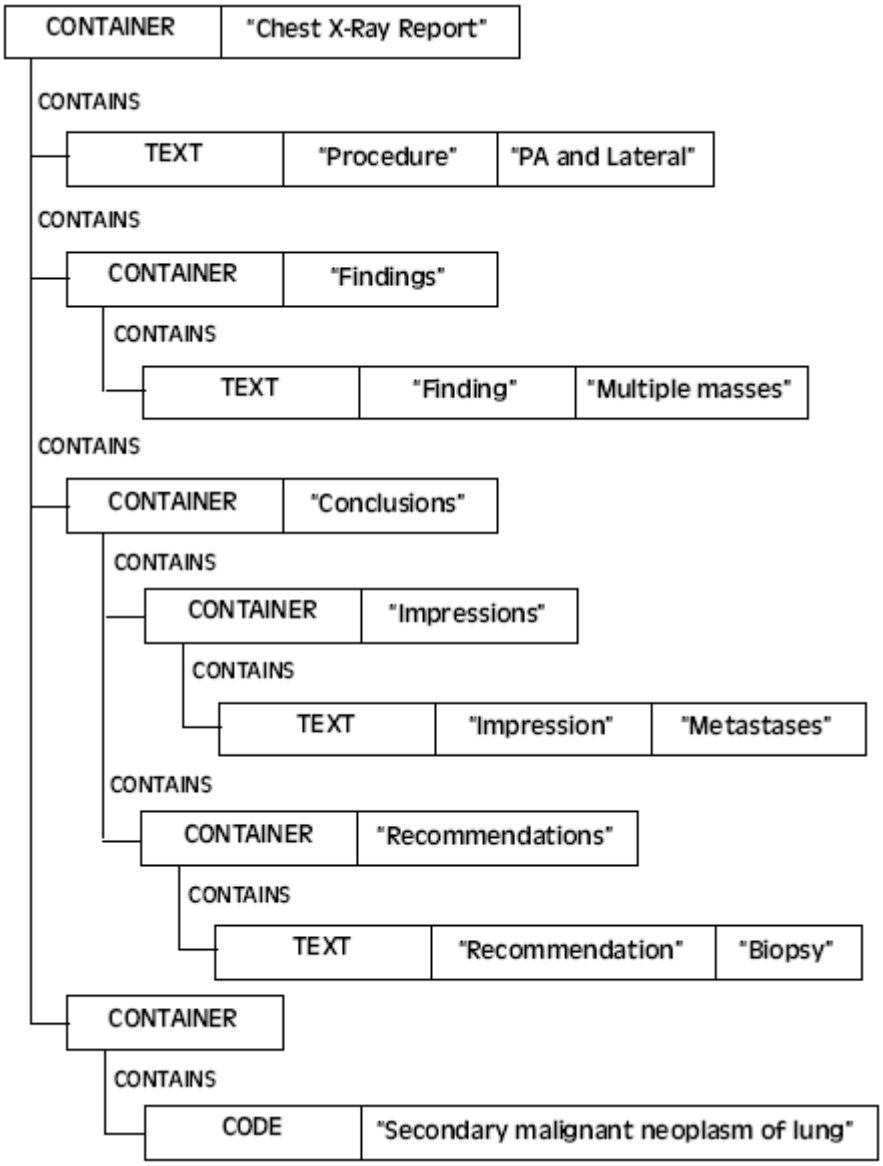


Figura 29: Representação gráfica de um DICOM SR (retirado de [Clunie, David A 2001])

B. HL7 CDA

```
1  <?xml version="1.0"?>
2  <!DOCTYPE levelone PUBLIC "-//HL7//DTD CDA Level One 1.0//EN" "levelone_1.0.dtd">
3  <levelone>
4    <clinical_document_header>
5      <id EX="a123" RT="2.16.840.1.113883.3.933"/>
6      <set_id EX="B" RT="2.16.840.1.113883.3.933"/>
7      <version_nbr V="2"/>
8      <document_type_cd V="11488-4" S="LOINC"
9        DN="Consultation note"/>
10     <origination_dttm V="2000-04-07"/>
11     <confidentiality_cd ID="CONF1" V="N" S="HL7_Confidentiality"/>
12     <confidentiality_cd ID="CONF2" V="R" S="HL7_Confidentiality"/>
13     <document_relationship>
14       <document_relationship.type_cd V="RPLC"/>
15       <related_document>
16         <id EX="a234" RT="2.16.840.1.113883.3.933"/>
17         <set_id EX="B" RT="2.16.840.1.113883.3.933"/>
18         <version_nbr V="1"/>
19       </related_document>
20     </document_relationship>
21     <fulfills_order>
22       <fulfills_order.type_cd V="FLFS"/>
23       <order><id EX="x23ABC" RT="2.16.840.1.113883.3.933"/></order>
24       <order><id EX="x42CDE" RT="2.16.840.1.113883.3.933"/></order>
25     </fulfills_order>
26     <patient_encounter>
27       <id EX="KPENC1332" RT="2.16.840.1.113883.3.933"/>
28       <practice_setting_cd V="GIM"
29         S="HL7_PracticeSetting" DN="General internal medicine clinic"/>
30       <encounter_tmr V="2000-04-07"/>
31       <service_location>
32         <id EX="KXLPa123" RT="2.16.840.1.113883.3.933"/>
33         <addr>
34           <HNR V="970"/><STR V="Post St"/><DIR V="NE"/>
35           <CTY V="Alameda"/><STA V="CA"/><ZIP V="94501"/>
36         </addr>
37       </service_location>
38     </patient_encounter>
39     <legal_authenticator>
40       <legal_authenticator.type_cd V="SPV"/>
41       <participation_tmr V="2000-04-08"/>
42       <signature_cd V="S"/>
43       <person>
44         <id EX="KP00017" RT="2.16.840.1.113883.3.933"/>
45         <person_name>
46           <nm>
47             <GIV V="Robert"/><FAM V="Dolin"/><SFX V="MD" QUAL="PT"/>
48           </nm>
```

Figura 30: Parte de um relatório médico no formato HL7 CDA

C. Exemplo de um ficheiro ARFF

```
@relation weather.symbolic

@attribute outlook {sunny, overcast, rainy}
@attribute temperature {hot, mild, cool}
@attribute humidity {high, normal}
@attribute windy {TRUE, FALSE}
@attribute play {yes, no}

@data
sunny,hot,high,FALSE,no
sunny,hot,high,TRUE,no
overcast,hot,high,FALSE,yes
rainy,mild,high,FALSE,yes
rainy,cool,normal,FALSE,yes
rainy,cool,normal,TRUE,no
overcast,cool,normal,TRUE,yes
sunny,mild,high,FALSE,no
sunny,cool,normal,FALSE,yes
rainy,mild,normal,FALSE,yes
sunny,mild,normal,TRUE,yes
overcast,mild,high,TRUE,yes
overcast,hot,normal,FALSE,yes
rainy,mild,high,TRUE,no
```

D. Mapeamento interno de termos médicos

Tabela 8: Mapeamento interno de termos médicos (parte)

código	*atributo1*	*atributo2*	*atributo3*	*atributo4*	*atributo5*	*atributo6*
10	anomalias					
101	anomalias	hematicas				
1011	anomalias	hematicas	grosseiras			
10111	anomalias	hematicas	grosseiras	agudas		
101111	anomalias	hematicas	grosseiras	agudas	intra-axiais	
101112	anomalias	hematicas	grosseiras	agudas	peri-encefalicas	
101113	anomalias	hematicas	grosseiras	agudas	intra-axiais/ peri-encefalicas	
20	fracturas					
201	fracturas	recente				
2011	fracturas	recente	desalinhadas			
2012	fracturas	recente	afundadas			
2013	fracturas	recente	desalinhadas/ afundadas			
30	alteracoes					
301	alteracoes	morfologicas				
3011	alteracoes	morfologicas	hipofise			
3012	alteracoes	morfologicas	densitometricas			
30121	alteracoes	morfologicas	densitometricas	focais		
301211	alteracoes	morfologicas	densitometricas	focais	parenquima	
3012111	alteracoes	morfologicas	densitometricas	focais	parenquima	cerebral
3012112	alteracoes	morfologicas	densitometricas	focais	parenquima	cerebeloso
3012113	alteracoes	morfologicas	densitometricas	focais	parenquima	tronco cerebral
3012114	alteracoes	morfologicas	densitometricas	focais	parenquima	cerebral\ cerebeloso\ tronco cerebral

E. Parte dos resultados das regras de associação

Tabela 9: As cem primeiras regras de associação descobertas nos relatórios médicos

Regra	Premissas	Conclusão	Cobertura	Confiança
1	conclusao = C10	protocol2 = 20111211	72,22%	89,66%
2	conclusao = C10	protocol1 = 20111111, protocol2 = 20111211	72,22%	89,66%
3	protocol1 = 20111111	report4 = 1001111N	72,22%	92,86%
4	protocol1 = 20111111	report3 = 901N	72,22%	92,86%
5	protocol1 = 20111111	report2 = 80111	72,22%	92,86%
6	protocol1 = 20111111	report1 = 3012114N	72,22%	92,86%
7	protocol1 = 20111111	conclusao = C10, protocol2 = 20111211	72,22%	92,86%
8	protocol1 = 20111111	protocol2 = 20111211, report4 = 1001111N	72,22%	92,86%
9	protocol1 = 20111111	protocol2 = 20111211, report3 = 901N	72,22%	92,86%
10	protocol1 = 20111111	protocol2 = 20111211, report2 = 80111	72,22%	92,86%
11	protocol1 = 20111111	protocol2 = 20111211, report1 = 3012114N	72,22%	92,86%
12	protocol1 = 20111111	report4 = 1001111N, report3 = 901N	72,22%	92,86%
13	protocol1 = 20111111	report4 = 1001111N, report2 = 80111	72,22%	92,86%
14	protocol1 = 20111111	report4 = 1001111N, report1 = 3012114N	72,22%	92,86%
15	protocol1 = 20111111	report3 = 901N, report2 = 80111	72,22%	92,86%
16	protocol1 = 20111111	report3 = 901N, report1 = 3012114N	72,22%	92,86%
17	protocol1 = 20111111	report2 = 80111, report1 = 3012114N	72,22%	92,86%
18	protocol1 = 20111111	protocol2 = 20111211, report4 = 1001111N, report3 = 901N	72,22%	92,86%
19	protocol1 = 20111111	protocol2 = 20111211, report4 = 1001111N, report2 = 80111	72,22%	92,86%
20	protocol1 = 20111111	protocol2 = 20111211, report4 = 1001111N, report1 = 3012114N	72,22%	92,86%
21	protocol1 = 20111111	protocol2 = 20111211, report3 = 901N, report2 = 80111	72,22%	92,86%
22	protocol1 = 20111111	protocol2 = 20111211, report3 = 901N, report1 = 3012114N	72,22%	92,86%
23	protocol1 = 20111111	protocol2 = 20111211, report2 = 80111, report1 = 3012114N	72,22%	92,86%
24	protocol1 = 20111111	report4 = 1001111N, report3 = 901N, report2 = 80111	72,22%	92,86%
25	notas1 = 30	protocolo1 = 20111211	61,00%	81,50%
26	protocol1 = 20111111	report4 = 1001111N, report2 = 80111, report1 = 3012114N	72,22%	92,86%
27	protocol1 = 20111111	report3 = 901N, report2 = 80111, report1 = 3012114N	72,22%	92,86%
28	protocol1 = 20111111	protocol2 = 20111211, report4 = 1001111N, report3 = 901N, report2 = 80111	72,22%	92,86%

29	protocol1 = 20111111	protocol2 = 20111211, report4 = 1001111N, report3 = 901N, report1 = 3012114N	72,22%	92,86%
30	protocol1 = 20111111	protocol2 = 20111211, report4 = 1001111N, report2 = 80111, report1 = 3012114N	72,22%	92,86%
31	protocol1 = 20111111	protocol2 = 20111211, report3 = 901N, report2 = 80111, report1 = 3012114N	72,22%	92,86%
32	protocol1 = 20111111	report4 = 1001111N, report3 = 901N, report2 = 80111, report1 = 3012114N	72,22%	92,86%
33	protocol1 = 20111111	protocol2 = 20111211, report4 = 1001111N, report3 = 901N, report2 = 80111, report1 = 3012114N	72,22%	92,86%
34	conclusao = C10	protocol1 = 20111111	75,00%	93,10%
35	protocol2 = 20111211	conclusao = C10	72,22%	96,30%
36	protocol2 = 20111211	report4 = 1001111N	72,22%	96,30%
37	protocol2 = 20111211	report3 = 901N	72,22%	96,30%
38	protocol2 = 20111211	report2 = 80111	72,22%	96,30%
39	notas1 = 30	report1 = 3012114N	72,22%	96,30%
40	conclusao = C10, protocol1 = 20111111	protocol2 = 20111211	72,22%	96,30%
41	protocol2 = 20111211	conclusao = C10, protocol1 = 20111111	72,22%	96,30%
42	protocol1 = 20111111, protocol2 = 20111211	conclusao = C10	72,22%	96,30%
43	protocol2 = 20111211	protocol1 = 20111111, report4 = 1001111N	72,22%	96,30%
44	protocol1 = 20111111, protocol2 = 20111211	report4 = 1001111N	72,22%	96,30%
45	protocol2 = 20111211	protocol1 = 20111111, report3 = 901N	72,22%	96,30%
46	protocol1 = 20111111, protocol2 = 20111211	report3 = 901N	72,22%	96,30%
47	protocol2 = 20111211	protocol1 = 20111111, report2 = 80111	72,22%	96,30%
48	protocol1 = 20111111, protocol2 = 20111211	report2 = 80111	72,22%	96,30%
49	protocol2 = 20111211	protocol1 = 20111111, report1 = 3012114N	72,22%	96,30%
50	protocol1 = 20111111, protocol2 = 20111211	report1 = 3012114N	72,22%	96,30%
51	protocol2 = 20111211	report4 = 1001111N, report3 = 901N	72,22%	96,30%
52	protocol2 = 20111211	report4 = 1001111N, report2 = 80111	72,22%	96,30%
53	protocol2 = 20111211	report4 = 1001111N, report1 = 3012114N	72,22%	96,30%
54	protocol2 = 20111211	report3 = 901N, report2 = 80111	72,22%	96,30%
55	protocol2 = 20111211	report3 = 901N, report1 = 3012114N	72,22%	96,30%
56	protocol2 = 20111211	report2 = 80111, report1 = 3012114N	72,22%	96,30%
57	protocol2 = 20111211	protocol1 = 20111111, report4 = 1001111N, report3 = 901N	72,22%	96,30%
58	protocol1 = 20111111, protocol2 = 20111211	report4 = 1001111N, report3 = 901N	72,22%	96,30%
59	protocol1 = 20111211	conclusao = C10	75,00%	96,40%

60	protocol1 = 20111111, protocol2 = 20111211	report4 = 1001111N, report2 = 80111	72,22%	96,30%
61	protocol2 = 20111211	protocol1 = 20111111, report4 = 1001111N, report1 = 3012114N	72,22%	96,30%
62	protocol1 = 20111111, protocol2 = 20111211	report4 = 1001111N, report1 = 3012114N	72,22%	96,30%
63	protocol2 = 20111211	protocol1 = 20111111, report3 = 901N, report2 = 80111	72,22%	96,30%
64	protocol1 = 20111111, protocol2 = 20111211	report3 = 901N, report2 = 80111	72,22%	96,30%
65	protocol2 = 20111211	protocol1 = 20111111, report3 = 901N, report1 = 3012114N	72,22%	96,30%
66	protocol1 = 20111111, protocol2 = 20111211	report3 = 901N, report1 = 3012114N	72,22%	96,30%
67	protocol2 = 20111211	protocol1 = 20111111, report2 = 80111, report1 = 3012114N	72,22%	96,30%
68	protocol1 = 20111111, protocol2 = 20111211	report2 = 80111, report1 = 3012114N	72,22%	96,30%
69	protocol2 = 20111211	report4 = 1001111N, report3 = 901N, report2 = 80111	72,22%	96,30%
70	protocol2 = 20111211	report4 = 1001111N, report3 = 901N, report1 = 3012114N	72,22%	96,30%
71	protocol2 = 20111211	report4 = 1001111N, report2 = 80111, report1 = 3012114N	72,22%	96,30%
72	protocol2 = 20111211	report3 = 901N, report2 = 80111, report1 = 3012114N	72,22%	96,30%
73	protocol2 = 20111211	protocol1 = 20111111, report4 = 1001111N, report3 = 901N, report2 = 80111	72,22%	96,30%
74	protocol1 = 20111111, protocol2 = 20111211	report4 = 1001111N, report3 = 901N, report2 = 80111	72,22%	96,30%
75	protocol2 = 20111211	protocol1 = 20111111, report4 = 1001111N, report3 = 901N, report1 = 3012114N	72,22%	96,30%
76	protocol1 = 20111111, protocol2 = 20111211	report4 = 1001111N, report3 = 901N, report1 = 3012114N	72,22%	96,30%
77	protocol2 = 20111211	protocol1 = 20111111, report4 = 1001111N, report2 = 80111, report1 = 3012114N	72,22%	96,30%
78	protocol1 = 20111111, protocol2 = 20111211	report4 = 1001111N, report2 = 80111, report1 = 3012114N	72,22%	96,30%
79	protocol2 = 20111211	protocol1 = 20111111, report3 = 901N, report2 = 80111, report1 = 3012114N	72,22%	96,30%
80	protocol1 = 20111111, protocol2 = 20111211	report3 = 901N, report2 = 80111, report1 = 3012114N	72,22%	96,30%
81	protocol2 = 20111211	report4 = 1001111N, report3 = 901N, report2 = 80111, report1 = 3012114N	72,22%	96,30%

82	protocol2 = 20111211	protocol1 = 20111111, report4 = 1001111N, report3 = 901N, report2 = 80111, report1 = 3012114N	72,22%	96,30%
83	protocol1 = 20111111, protocol2 = 20111211	report4 = 1001111N, report3 = 901N, report2 = 80111, report1 = 3012114N	72,22%	96,30%
84	protocol1 = 20111111	conclusao = C10	75,00%	96,43%
85	protocol1 = 20111111	protocol2 = 20111211	75,00%	96,43%
86	protocol2 = 20111211	protocol1 = 20111111	75,00%	100,00%
87	report4 = 1001111N	protocol1 = 20111111	72,22%	100,00%
88	report3 = 901N	protocol1 = 20111111	72,22%	100,00%
89	report2 = 80111	protocol1 = 20111111	72,22%	100,00%
90	report1 = 3012114N	protocol1 = 20111111	72,22%	100,00%
91	report4 = 1001111N	protocol2 = 20111211	72,22%	100,00%
92	report3 = 901N	protocol2 = 20111211	72,22%	100,00%
93	report2 = 80111	protocol2 = 20111211	72,22%	100,00%
94	report1 = 3012114N	protocol2 = 20111211	72,22%	100,00%
95	report4 = 1001111N	report3 = 901N	72,22%	100,00%
96	report3 = 901N	report4 = 1001111N	72,22%	100,00%
97	report4 = 1001111N	report2 = 80111	72,22%	100,00%
98	report2 = 80111	report4 = 1001111N	72,22%	100,00%
99	report4 = 1001111N	report1 = 3012114N	72,22%	100,00%
100	report1 = 3012114N	report4 = 1001111N	72,22%	100,00%