

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
CURSO DE GRADUAÇÃO EM CIÊNCIA DA
COMPUTAÇÃO**

Tiago Silva Proença

**Uma ferramenta para conversão de exames de
eletrocardiografia para o formato DICOM**

Trabalho de Conclusão de Curso

Prof. Euclides de Moraes Barros Júnior, Msc.

Orientador

João Bosco Manguiera Sobral, Dr.

Coorientador

Florianópolis, Maio de 2004

Uma ferramenta para conversão de exames de eletrocardiografia para o formato DICOM

Tiago Silva Proença

Este Trabalho de Conclusão de Curso foi aprovado em sua forma final pelo Curso de em Ciência da Computação da Universidade Federal de Santa Catarina.

Prof. Euclides de Moraes Barros Júnior, Msc.

Orientador

Prof. José Mazzucco Jr., Dr.

Coordenador do Curso

Banca Examinadora

Prof. Michel Sehn, Msc.

Prof. Lidiane Pereira dos Reis, Msc.

À minha família.

Agradecimentos

Aos meus pais e familiares, pelo apoio e incentivo.

Aos colegas e amigos de trabalho que, ao seu modo, contribuíram para a realização deste trabalho, em especial à amiga Lidiane, pela grande ajuda.

Aos professores da Universidade Federal de Santa Catarina, especialmente o professor João Bosco Manguiera Sobral, por ter me dado a oportunidade de realizar este trabalho sobre sua coorientação, e me dado apoio quando necessário.

À Deus.

Sumário

Lista de Figuras	ix
Lista de Tabelas	xi
Lista de Siglas e Abreviaturas	xii
Resumo	xiv
Abstract	xv
1 Introdução	1
1.1 Objetivos	2
1.1.1 Objetivo Geral	2
1.1.2 Objetivos Específicos	2
1.2 Motivação e Justificativa	2
1.3 Organização do Texto	3
2 A linguagem XML	5
2.1 Um breve histórico	5
2.2 O que é o XML?	6
2.3 Estrutura de um documento XML	6
2.3.1 Elementos	6
2.3.2 Atributos	7
2.3.3 Comentários	8
2.3.4 Instruções de processamento	8

2.3.5	O prólogo XML	9
2.4	XML bem formado	10
2.5	XML Válido	10
3	Eletrocardiograma	11
3.1	Um breve histórico	11
3.2	O eletrocardiograma	12
3.3	O sistema circulatório	13
3.3.1	O coração	14
3.3.2	Os vasos	17
3.4	O ciclo cardíaco	19
3.5	A ativação elétrica do coração	21
3.6	Derivações eletrocardiográficas	23
3.6.1	As derivações do plano frontal	24
3.6.2	As derivações do plano horizontal	26
3.7	Ondas, intervalos e segmentos do eletrocardiograma	27
3.7.1	A onda P	29
3.7.2	O intervalo PR	29
3.7.3	O complexo QRS	29
3.7.4	A onda R	30
3.7.5	A onda S	30
3.7.6	O Segmento ST	31
3.7.7	A onda T	31
3.7.8	O intervalo QT	31
4	DICOM	32
4.1	Evolução histórica	33
4.2	Organização do padrão	34
4.3	Conceitos gerais sobre o padrão	37
4.3.1	Classes de serviço e classes SOP	37

4.3.2	Definição de objeto de informação - IOD	38
4.3.3	Atributos	39
4.3.4	Elementos de serviço	40
4.3.5	Instâncias SOP	40
4.3.6	Identificação	41
4.3.7	Relações	42
4.3.8	Representações de Valor - VR	42
4.3.9	Elementos de Dados	43
4.3.10	Sintaxe de Transferência	43
5	Codificação de um arquivo DICOM	44
5.1	Cabeçalho - File Meta Information	45
5.2	Conjunto de dados - Data Set	45
5.2.1	Campos de um Elementos de Dados	46
5.2.2	Estrutura de um Elemento de Dados com VR explícito	47
5.2.3	Estrutura de um Elemento de Dados com VR implícito	49
5.2.4	Tamanho de Grupo - Group Length	49
5.2.5	Ordenação dos <i>bytes</i>	49
5.2.6	Elemento de Dados como seqüência	50
6	A entidade de informação DICOM <i>Waveform</i>	54
6.1	Objetos de informação <i>Waveform</i>	56
6.1.1	<i>Basic Voice Audio Information Object Definition</i>	56
6.1.2	<i>12-Lead Electrocardiogram Information Object Definition</i>	56
6.1.3	<i>General Electrocardiogram Information Object Definition</i>	56
6.1.4	<i>Ambulatory Electrocardiogram Information Object Definition</i>	56
6.1.5	<i>Hemodynamic Information Object Definition</i>	57
6.1.6	<i>Basic Cardiac Electrophysiology Information Object Definition</i>	57
6.2	Principais características entre as modalidades	57

7	Estado da arte	59
7.1	Dicom Waveform Viewer	60
7.2	Cyclops Waveform Viewer	61
7.3	Quinton Q-TRACK II Holter System	62
8	Tecnologias utilizadas	63
8.1	<i>Extensible Markup Language</i> - XML	63
8.2	Java	63
8.2.1	<i>Java Document Object Model</i> - JDOM	64
8.3	<i>Digital Imaging and Communications in Medicine</i> - DICOM	64
8.4	Eclipse	64
8.5	Concurrent Versions System - CVS	65
9	Sistema proposto	66
9.1	O arquivo XML	67
9.2	Modelagem OO do IOD Waveform	67
9.3	Implementação do Sistema	67
9.3.1	Identificação	68
9.3.2	Dump	68
9.3.3	Validação	69
10	Considerações finais	71
11	Trabalhos futuros	73
	Referências Bibliográficas	75
	Anexo 1	78
	Anexo 2	79

Lista de Figuras

3.1	Aparelho circulatório	14
3.2	Conformação interna do coração	15
3.3	Superfície externa do coração	17
3.4	A pequena e a grande circulação	19
3.5	As quatro fases da contração cardíaca	20
3.6	Conjunto formado pelos nó SA, nó NAV e pelo feixe de His	22
3.7	As linhas de derivação do plano frontal: D1, D2, D3, VR, VL e VF. Triângulo de Eithoven	24
3.8	Pontos de aplicação dos eletrodos para o registro das derivações precordiais: V1 a V6	26
3.9	Corte horizontal do tórax. As linhas de derivações precordiais partem do 4º e do 5º espaços intercostais e se dirigem para o centro elétrico do coração.	27
3.10	As derivações precordiais V1 a V6 e suas relações com um corte horizontal do coração.	27
3.11	As deflexões, segmentos e intervalos do eletrocardiograma normal.	28
4.1	Principais estruturas do modelo de informação DICOM	37
4.2	Definições de Objetos de Informação e suas relações	39
5.1	Formato do arquivo DICOM	44
5.2	Conjunto de dados DICOM e as estruturas de um elemento de dados	46
6.1	Modelo de informação DICOM <i>Waveform IOD</i>	55

7.1	<i>Dicom Waveform Viewer</i>	60
7.2	<i>Cyclops Waveform Viewer</i>	61
7.3	<i>Quinton Q-TRACK II Holter System</i>	62
9.1	O sistema proposto	66
9.2	Janela da aplicação	68
9.3	Arquivo DICOM em representação textual	69
9.4	<i>DICOM Toolkit - dcmdump</i>	70
9.5	<i>PixelMed Java DICOM Toolkit - ECGViewer</i>	70

Lista de Tabelas

5.1	Elementos de dados com VR explícito do tipo OB, OW, OF, SQ, UT ou UN.	48
5.2	Elementos de dados com VR explícito de tipo diferentes dos da tabela 5.1	48
5.3	Elementos de dados com VR implícito	49
5.4	Seqüência com três itens de tamanho explícito	52
5.5	Seqüência de tamanho indefinido, com dois itens de tamanho explícito . .	52
5.6	Seqüência com tamanho indefinido, com um item de tamanho explícito e um de tamanho indefinido	53
6.1	Tabela de módulos de um <i>Waveform</i> IOD	55
6.2	Modalidades <i>Waveform</i>	57
6.3	Quantidade de seqüências de <i>Waveform</i> por modalidades	58
6.4	Quantidade de canais por modalidade	58
6.5	Freqüência de amostragem por modalidade	58

Lista de Siglas e Abreviaturas

AAPM	<i>American Association of Physicists in Medicine</i>
ACR	<i>American College of Radiology</i>
DCMR	Eletrocardiograma
DICOM	<i>Digital Imaging and Communications in Medicine</i>
DOM	<i>Document Object Model</i>
DTD	<i>Document Type Definition</i>
ECG	Eletrocardiograma
HIS	<i>Hospital Information System</i>
HTML	<i>Hypertext Markup Language</i>
IE	<i>Information Entity</i>
IOD	<i>Information Object Definition</i>
IOM	<i>Information Object Modules</i>
ISO	<i>International Standards Organization</i>
JAXP	<i>Java API for XML Processing</i>
JDOM	<i>Java Document Object Model</i>
NEMA	<i>National Electrical Manufacturers Association</i>
OSI	<i>Open Systems Interconnection</i>
PACS	<i>Picture Archiving and Communications System</i>
PS	<i>Part of Standard</i>
SAX	<i>Simple API for XML</i>
SCP	<i>Service Class Provider</i>

SCU	<i>Service Class User</i>
SGML	<i>Standard Generalized Markup Language</i>
SOP	<i>Service Object Pair Class</i>
SQ	<i>Sequence</i>
TCP/IP	<i>Transmission Control Protocol/Internet Protocol</i>
UID	<i>Unique Identifier</i>
VR	<i>Value Representation</i>
W3C	<i>World Wide Web Consortium</i>
XML	<i>Extensible Markup Language</i>

Resumo

A grande maioria dos aparelhos médicos utilizados no âmbito de clínicas e hospitais modernos, codificam e trocam dados seguindo o protocolo padrão de comunicação DICOM, permitindo um intercâmbio de exames entre equipamentos e profissionais da área médica.

O projeto em questão foi elaborado para suprir carências no que diz respeito ao armazenamento de sinais vitais provenientes de exames de eletrocardiografia, onde seria possível utilizar os recursos computacionais na análise e diagnóstico médico mesmo em ambientes que não disponibilizam equipamentos modernos.

Para tanto, foi implementada uma metodologia capaz de adequar um arquivo XML para uma futura exportação no formato DICOM, através de uma aplicação desenvolvida para este propósito.

Palavras Chave: Eletrocardiograma (ECG), DICOM, *Waveform*.

Abstract

Most medical equipment that is used in modern clinics and hospitals, codifies and process data following the DICOM communication protocol, allowing medical exams exchange between equipment and professionals in the medical area.

This project was developed to resolve the problem of storing vital signals from electrocardiographic exams, so that it would be possible to use the computational resources for analysis and diagnosis, even in places where modern equipments is not available.

To do this, a methodology was implemented capable of adjusting an XML file for a future exportation into the DICOM format, through an application developed for this purpose.

Key-Words: Electrocardiographic (ECG), DICOM, Waveform.

Capítulo 1

Introdução

Desde o início do século XX, a inovação e utilização de eletrocardiógrafos vieram influenciar de forma significativa à medicina, trazendo inúmeros benefícios na detecção de patologias do coração.

A grande maioria dos aparelhos médicos utilizados em modernas clínicas de diagnóstico por imagem atualmente codificam e trocam dados seguindo o protocolo padrão de comunicação DICOM (*Digital Imaging and Communications in Medicine*).

O projeto proposto foi elaborado com o propósito de suprir uma necessidade da empresa DIXTAL, a qual foi utilizada neste trabalho como estudo de caso. Esta empresa atua na área da saúde, desenvolvendo tecnologia de informação para a captação de sinais vitais, e apresentava uma carência no que diz respeito ao armazenamento de informações em conformidade com o padrão DICOM.

Diante disto, foi desenvolvido um software utilizando iniciativas de software livre, para resolver tal problema.

Neste capítulo serão abordados os objetivos e a relevância de se desenvolver este projeto.

1.1 Objetivos

1.1.1 Objetivo Geral

O objeto geral deste projeto é a implementação de uma aplicação na linguagem Java, que efetua a conversão de um exame de eletrocardiograma (ECG) armazenado no formato XML (*Extensible Markup Language*), previamente modelado, para o formato DICOM.

1.1.2 Objetivos Específicos

- Estudo sobre eletrocardiografia;
- Estudo do padrão DICOM utilizado para a comunicação entre equipamentos médicos;
- Análise de arquivos XML proveniente do sistema DX-25 da empresa DIXTAL;
- Modelagem de um arquivo no formato XML, seguindo o modelo do objeto de informação: *General ECG IOD*;
- Implementação de uma ferramenta computacional capaz de ler arquivos XML e exportá-los no padrão DICOM;

1.2 Motivação e Justificativa

A grande relevância ao desenvolver este projeto é o fato de trazer contribuições em vários aspectos. Do ponto de vista tecnológico é importante ressaltar que a codificação e transferência de sinais e imagens médicas são imprescindíveis no diagnóstico por imagem, auxílio em cirurgias e monitoramento de pacientes, visto que a medicina tem se mostrado cada vez mais interessada em utilizar recursos tecnológicos para obter mais informações em estados clínicos.

Se analisarmos a realidade brasileira, é comum encontrarmos no âmbito de clínicas e hospitais, equipamentos de eletrocardiografia e radiológicos obsoletos. Neste contexto

fica evidente a dificuldade no armazenamento de informações dos exames médicos, pelo fato da dependência de sistemas externos. Sendo assim, a solução mais adequada para tal problema seria o aperfeiçoamento dos sistemas atuais ou desenvolvimento de novas aplicações que possam utilizar os equipamentos existentes, sem haver a necessidade de novas aquisições, já que isso seria mais um obstáculo para a melhoria no sistema de saúde pública.

Desta forma, um fator crucial para as inovações tecnológicas na medicina, seria fazer com que as informações de exames médicos, sejam armazenados no protocolo padrão de comunicação entre os equipamentos radiológicos (DICOM), permitindo que exista um intercâmbio dessas informações entre os profissionais e outros equipamentos.

A natureza multidisciplinar do projeto que engloba várias áreas da computação, revela a importância da formação científica adquirida e os benefícios que os profissionais podem trazer a sociedade.

1.3 Organização do Texto

O presente capítulo mostrou os objetivos gerais e específicos do trabalho e nos dá a localização dos conceitos explorados em cada capítulo posterior.

O Capítulo 2 apresenta uma descrição da tecnologia XML e alguns conceitos necessários para se estruturar um documento XML.

O Capítulo 3 apresenta um estudo sobre a eletrocardiografia e conceitos relacionados.

O Capítulo 4 apresenta vários conceitos sobre o padrão DICOM e seu histórico de evolução.

O Capítulo 5 descreve a codificação de um arquivo DICOM.

O Capítulo 6 apresenta a entidade de informação Waveform e seus objetos de informação.

O Capítulo 7 apresenta uma breve descrição sobre outros sistemas DICOM existentes.

O Capítulo 8 apresenta uma descrição de todas as tecnologias utilizadas no desenvolvimento do sistema.

O Capítulo 9 apresenta o sistema proposto, bem como as etapas para a sua construção.

O Capítulo 10 apresenta as considerações finais do trabalho.

O Capítulo 11 apresenta sugestões para trabalhos futuros.

Capítulo 2

A linguagem XML

A aplicação do XML está sendo estudada atualmente por diversas empresas e instituições em todo o mundo, com o objetivo de se obterem abordagens mais eficientes para a manipulação de dados. A linguagem está sendo amplamente utilizada em aplicações para a área da saúde, pela facilidade com que se pode construir software para o processamento de documentos XML, e por ser facilmente combinada com folhas de estilo, além de prover mecanismos de validação como os DTD (*Document Type Definition*) e/ou XML Schemas (BORTOLUZZI, 2003).

Neste capítulo será apresentada uma descrição da tecnologia XML, juntamente com alguns conceitos pertinentes para a estruturação de um documento XML.

2.1 Um breve histórico

O HTML (*Hypertext Markup Language*) possibilitou que os dados fossem apresentados em uma estrutura simples e de fácil leitura. Entretanto, o HTML apresenta limitações fundamentadas em sua própria concepção, baseada em marcações fixas.

A emergência do XML como um padrão para a representação de dados na Internet, facilitou a publicação em meios eletrônicos, por prover uma sintaxe simples, clara e legível para computadores e seres humanos.

O XML é uma linguagem derivada da SGML (*Standard Generalized Markup Lan-*

guage) e foi idealizada por Jon Bosak, engenheiro da Sun Microsystems. O autor era conhecedor e usuário da SGML e apresentou ao W3C (*World Wide Web Consortium*) sua idéia de explorar o SGML em aplicações voltadas para Internet.

Em 1996, foi criado o XML, inicialmente como uma versão simplificada do SGML, e, em dezembro de 1997 foi publicada a primeira versão do XML. Em fevereiro de 1998, o XML tornou-se uma especificação formal, reconhecida pelo W3C (ALMEIDA, 2002).

2.2 O que é o XML?

XML é uma linguagem de marcação baseada em texto a qual é usada para intercâmbio de dados, principalmente na Internet. Podemos pensar no XML como uma linguagem para descrição de dados.

Como o HTML, os dados são identificados usando etiquetas, também conhecida como *tags*. Coletivamente, as etiquetas são conhecidas como marcação. O XML não é uma linguagem de marcação predefinida como o HTML e possibilita ao autor do documento projetar sua própria marcação. Um exemplo de marcação é apresentado a seguir:

```
<mensagem>isto é uma mensagem...</mensagem>
```

2.3 Estrutura de um documento XML

Elementos são os itens de informação fundamentais para um documento XML, mas não são os únicos que estão presente em um documento. Nas próximas seções, será descrito o conceito de elemento além de mais alguns conceitos para a melhor compreensão de um documento XML.

2.3.1 Elementos

Um elemento XML costuma ser formado por um texto representando um dado, delimitado por etiquetas de início e fim. A etiqueta de fim, é precedida pelo caracter /. A sintaxe básica para um elemento XML é apresentado a seguir:

```
<nome_do_elemento>texto</nome_do_elemento>
```

Um documento XML é formado por um conjunto de elementos. Pode existir elementos dentro de elementos de forma a representar informação relacionada, formando o que se conhece como árvore. O exemplo a seguir mostra a representação de uma mensagem de email.

```
<mensagem>
  <de>eu@meuEndereco.com</de>
  <para>voce@seuEndereco.com</para>
  <assunto>Tudo bem?</assunto>
  <texto>
    blá, blá, blá...
  </texto>
</mensagem>
```

Os elementos que estão dentro de outro elementos são chamados de elementos filhos. No exemplo acima, o Elemento `<assunto>` é filho de `<mensagem>`.

Existe ainda elementos vazios, que são aqueles que não possuem texto nem elementos filhos. Uma forma de representá-lo seria:

```
<elemento_vazio></elemento_vazio>
```

Ou de maneira simplificada:

```
<elemento_vazio/>
```

2.3.2 Atributos

Os atributos são utilizados para fornecer informações adicionais sobre um elemento. Os valores dos atributos devem ser sempre cercados por aspas. A seguir é apresentado um exemplo utilizando atributos:

```
<mensagem de="eu@meuEndereco.com" para="voce@seuEndereco.com"
  assunto="Tudo bem?">
  <texto>
    blá, blá, blá...
  </texto>
</mensagem>
```

De fato, não existe uma regra que diga quando utilizar atributos e quando utilizar elementos. Isto cabe ao desenvolvedor do documento XML decidir qual é melhor abordagem para representar o propósito do documento.

2.3.3 Comentários

Os comentários podem ser colocados em qualquer lugar dentro de um documento XML, exceto dentro de uma etiqueta. Comentários são cercados por `<!--` e `-->` e podem abranger várias linhas. Existem algumas restrições, como a utilização de comentários aninhados, e, a colocação de `'-`', pois poderia confundir o *parser* XML. Um exemplo de comentário é apresentado abaixo:

```
<!-- Isto é um comentário -->
<nome>João da Silva</nome>
```

2.3.4 Instruções de processamento

Um documento XML pode conter instruções de processamento as quais fornecem informações adicionais ao *parser* XML para ajudá-lo a interpretar melhor o documento. Um exemplo disso é quando usamos uma instrução de processamento para aplicar uma folha de estilo para que transforme a aparência do documento XML quando exibido em um navegador Web. Um exemplo de instrução de processamento é apresentado a seguir:

```
<?xml-stylesheet type="text/xsl" href="stylesheet.xsl"?>
```

Como podemos observar, as instruções de processamento começam com um sinal de menor e uma interrogação (<?) e terminam com uma interrogação seguida de um sinal de maior (? >) de forma semelhante à declaração XML.

2.3.5 O prólogo XML

O prólogo contém metadados a respeito do resto do documento e é composto de uma declaração XML, das instruções de processamento e das definições de DTD ou XML *Schemas*.

A declaração no começo do documento XML, serve para dizer a um processador de textos que o documento se trata definitivamente de um documento XML. A sintaxe mínima dessa declaração é apresentada a seguir:

```
<?xml version="1.0"?>
```

Pode se usar atributos adicionais como é mostrado a seguir:

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes"?>
```

O atributo *standalone* especifica se outros arquivos são necessários para renderizar o documento, tais como DTD ou XML *Schemas*.

O atributo *encoding* especifica o conjunto de caracteres em uso no documento. Alguns dos valores mais comuns para este atributo são:

- **ISO-8859-1:** Alfabeto latino 1. Usado para a maioria das linguagens do oeste europeu. Similar ao ASCII de 8 bits. Permite o uso de acentuação.
- **UTF-8:** Unicode de 8 bits.
- **UTF-16:** Unicode de 16 bits usado para conjuntos de caracteres internacionais.

2.4 XML bem formado

A especificação do XML disponibilizada pelo W3C define que um documento XML bem formado é aquele que é formado por um prólogo e um elemento raiz. O elemento raiz é aquele que contém todos os outros elementos do documento.

Para ser carregado corretamente por um *parser* XML, o documento XML deve ser bem formado. Abaixo, um exemplo:

```
<?xml version="1.0"?>
<automovel>
  <ano>1996</ano>
  <marca>Corsa</marca>
  <fabricante>Chevrolet</fabricante>
</automovel>
```

2.5 XML Válido

Um documento XML válido é aquele cujo código e conteúdo estejam corretos, e as regras do XML como as do DTD e XML *Schemas* sejam seguidas:

- Documentos válidos seguem as regras de especificação do XML, inclusive aninhamento apropriado, presença de atributos exigidos e valores corretos para os atributos (MOULTIS; KIRK, 2000).
- O documento deve ter um DTD ou XML Schema correspondente, e atende as regras associadas.

Documentos bem-formados obedecem as regras da especificação XML, ao passo que documentos XML válidos requerem adequação tanto à especificação XML como as regras de validação associadas. Todos os documentos válidos são bem formados, mas nem todos os documentos bem formados são válidos.

Capítulo 3

Eletrocardiograma

Neste capítulo será abordado o estudo da eletrocardiografia, onde será visto um breve histórico, noções gerais sobre a anatomia, funcionalidade e atividade elétrica do coração. Isto nos ajudará a compreender como surge o sinal de ECG.

3.1 Um breve histórico

Harvey, em 1616, foi quem primeiro descreveu a relação entre a circulação do sangue e o batimento cardíaco. Kolliker e Müller, em 1856, constataram que o batimento cardíaco gerava um impulso elétrico. A partir de então, iniciou-se tentativas de registrar tal fenômeno. Augustus Waller, em 1887, registrou as variações do potencial de ação geradas pelo coração na superfície corpórea, usando um eletrômetro capilar de Lippmann (TILLEY, 1992). Adler, desenvolveu um galvanômetro para cabos de transatlântico e, Eithoven, percebendo o potencial deste aparelho, o modificou, melhorando sua sensibilidade, e o aplicou para registrar atividades elétricas do coração. Também foi Eithoven, em 1903, o responsável pela nomenclatura das ondas do ECG em P, Q, R, S e T, e pela criação do sistema de derivações bipolares (D_1, D_2, D_3).

Com o aprofundamento de estudos nesta área, Wilson, em 1943, propõe as derivações unipolares V_R ("R"= right, membro anterior direito), V_L ("L"= left, membro anterior esquerdo) e V_F ("F"= foot, pés, membros posteriores). Goldberger, em 1942 aperfeiçoou as

derivações de Wilson, para a nomenclatura atualmente utilizada: aV_R , aV_L e aV_F (onde "a" significa aumentada) (EDWARDS, 1987; TILLEY, 1992)

A partir de então, inúmeras contribuições vieram enriquecer os conhecimentos eletrocardiográficos, tanto teórica como praticamente.

3.2 O eletrocardiograma

Segundo a Sociedade Brasileira de Cardiologia (1996) o eletrocardiograma (ECG) consiste em um exame complementar usado em cardiologia, o qual avalia a atividade elétrica do coração. Neste exame, são identificados o ritmo, a frequência cardíaca, possíveis sobrecargas, alterações na nutrição do coração (isquemia) e outras várias informações complementares que o método pode nos fornecer. O ECG se destaca por não oferecer riscos para o paciente, é indolor e de baixo custo quando comparado a outros procedimentos.

A metodologia consiste em ligar sobre alguns centros de excitação denominados nós, eletrodos que recolhem e traduzem os estímulos elétricos em uma série de ondas, e fazer o registro em papel desta atividade elétrica do coração. O aparelho usado para registrar a atividade cardíaca é chamado eletrocardiógrafo.

Existe uma série de características eletrocardiográficas consideradas, em conjunto, como um "padrão de normalidade", e que apresentam variações individuais como as dependentes de idade, sexo e tipo constitucional (TRANCHESI, 1972).

Devemos ressaltar o fato de existir anomalias cardíacas que não alteram o ECG. Assim um traçado que estaria dentro dos limites de normalidade, não exclui a possibilidade de cardiopatia (TRANCHESI, 1972).

O ECG é formado por ondas de despolarização e repolarização. São elas as ondas P, Q, R, S e T. A onda P e o complexo QRS correspondem respectivamente às despolarizações atrial e ventricular que ocorrem antes do fenômeno de contração do coração. A onda T, por sua vez, corresponde à repolarização ventricular. O fenômeno de despolarização deve sempre ocorrer para que ocorra o fenômeno de contração. Isto implica que as ondas P ocorrem imediatamente antes do início da contração atrial e o com-

plexo QRS imediatamente antes da contração ventricular. A repolarização atrial ocorre quase que simultaneamente à despolarização ventricular, e como consequência não é observada no ECG (TILLEY, 1992).

Para a correta interpretação do traçado do ECG, é necessário que estes estejam padronizados. Diante disto, todos os traçados devem ser realizados em papel apropriado, do tipo quadriculado, com 1 mm^2 de área. O papel desloca-se sob a agulha do aparelho em duas (25 mm/s e 50 mm/s) ou mais velocidades. De acordo com a velocidade usada, obtém-se, por lei da física ($v_m = \Delta s / \Delta t$, onde v_m = velocidade média, Δs = espaço e Δt = tempo), que na velocidade de 25 mm/s a distância de 1 mm equivale a 0,04 s e que na velocidade de 50 mm/s a distância de 1 mm equivale a 0,02 s. A amplitude das ondas está diretamente relacionada a diferença de potencial registrada. Foi estabelecido que 1 mm equivale a 0,1 mV. Logo, tem-se que a padronização freqüentemente utilizada nos ECG, que 1 mV equivale a 10 mm. Portanto, o papel do ECG nada mais é que um sistema cartesiano onde no eixo de abscissas marca-se o tempo (em segundos) e no eixo das ordenadas as diferenças de potencial (em milivolts) (TILLEY, 1992).

3.3 O sistema circulatório

Todas as células do nosso corpo necessitam de oxigênio e nutrientes para desempenhar suas funções e para manter-se em vida. A tarefa de transportar estes elementos cabe ao sangue, que por sua vez recebe resíduos descartados pelas células. Para realizar esta tarefa o sangue necessita de circular por todo o corpo. Os canais por onde o sangue circula são chamados de vasos sanguíneos, enquanto o coração desempenha o papel de bomba propulsora que dá ao sangue impulso para a circulação. Coração e vasos constituem, no seu conjunto, o aparelho circulatório.

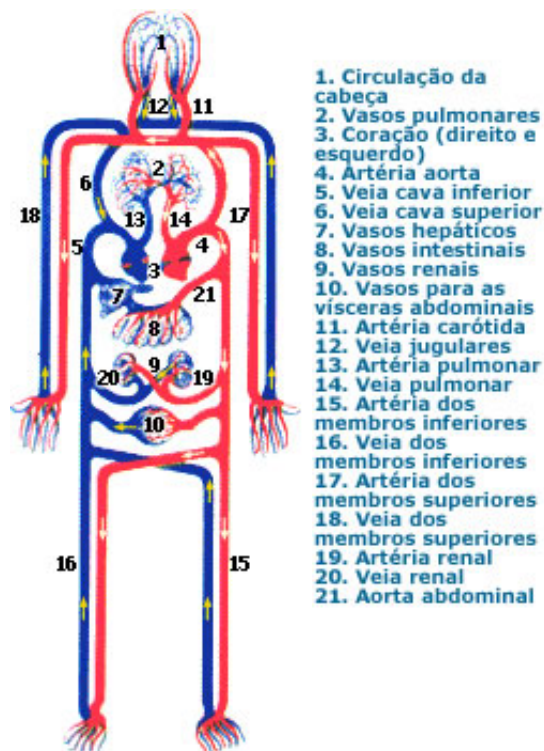


Figura 3.1: Aparelho circulatório

3.3.1 O coração

O coração é um órgão muscular localizado na cavidade torácica, abaixo do osso anterior do tórax o qual é chamado de esterno, deslocado em direção ao lado esquerdo. Tem aproximadamente o tamanho de um punho de um adulto, fechado. Este possui cor vermelho escuro, porém a uniformidade dessa cor é interrompida por estrias amareladas, que são formações de tecido adiposo.

O coração é o órgão central da circulação. É um músculo cuja função é recolher o sangue provenientes das veias e lançá-lo nas artérias. O coração está envolvido por um tecido conjuntivo denominado pericárdio. Suas cavidades internas estão forradas por uma membrana delgada: o endocárdio. A parte muscular do coração se chama miocárdio. Tanto o coração como todos os vasos do corpo humano, estão revestidos por uma capa de células planas, denominadas endotélio, a qual evita que o sangue se coagule.

A cavidade do coração está dividida em duas partes: uma direita e outra es-

querda. Estas estão separadas por um septo muscular. Seja do lado direito ou do lado esquerdo, encontramos duas cavidades: uma superior, a aurícula (ou átrio) e outra inferior, o ventrículo.

Aurícula e ventrículo estão em comunicação por meio de um orifício, o orifício aurículo-ventricular. Ao todo, o coração está, dividido em quatro cavidades: a aurícula e o ventrículo da direita, e a aurícula e o ventrículo da esquerda. A aurícula e o ventrículo que estão do mesmo lado se comunicam entre si. Isto não acontece com as cavidades do outro lado, pois não existe nenhuma comunicação. Logo, o sangue da metade esquerda não se mistura com o sangue da metade direita.

As aurículas recebem o sangue do interior do organismo e o impulsiona aos ventrículos. Estes, por sua vez, propulsam o sangue que recebem a todo o corpo. Não existe comunicação interauricular nem interventricular.

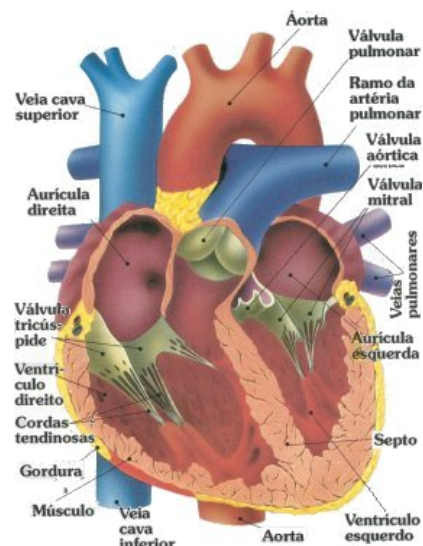


Figura 3.2: Conformação interna do coração

Na Figura 3.2, pode se observar de forma clara as quatro cavidades do coração. A seguir será feita uma descrição distinta dessas cavidades.

- **Aurícula direita:** Recebe a veia cava superior e a veia cava inferior, que trazem ao coração o sangue proveniente da cabeça e dos braços (cava superior) e da parte inferior do corpo (cava inferior). A aurícula direita está em comunicação com o ventrículo direito pelo orifício aurículo-ventricular, ao nível do qual existe uma válvula: a válvula aurículo-ventricular direita, dita também tricúspide porque é formada de três peças valvulares. A função desta válvula é permitir a passagem do sangue da aurícula para o ventrículo e de impedir o refluxo do ventrículo para a aurícula.
- **Ventrículo direito:** Recebe o sangue da aurícula direita pelo orifício aurículo-ventricular e a válvula tricúspide. No ventrículo direito tem início a artéria pulmonar que leva o sangue aos pulmões. O orifício de saída da artéria pulmonar vem a ser fechado pelas válvulas sigmóides, as quais impedem que o sangue volte da artéria pulmonar ao ventrículo.
- **Aurícula esquerda:** Recebe o sangue das quatro veias pulmonares e comunica com o ventrículo subjacente pelo orifício aurículo - ventricular esquerdo, ao nível do qual existe uma válvula: a válvula bicúspide. Na verdade, contrariamente à válvula análoga da direita, é ela formada somente por duas peças. É chamada também mitral porque tem a forma de uma mitra de bispo invertida.
- **Ventrículo esquerdo:** Recebe o sangue da aurícula esquerda através da sua válvula mitral, e nele se inicia a maior artéria do organismo: a artéria aorta. Também a artéria aorta, no seu início no ventrículo, é dotada de válvulas sigmóides, as quais têm a mesma função das válvulas sigmóides da pulmonar, isto é, impedir a volta do sangue da artéria para o ventrículo.

O coração, como qualquer outro músculo do corpo, necessita de receber oxigênio para que funcione adequadamente. A musculatura do coração é nutrida através de um

sistema de artérias, denominadas artérias coronárias. Estas se originam da aorta. As duas artérias coronárias mais importantes são a coronária direita e a coronária esquerda - esta última se divide (mais freqüentemente) em artéria coronária descendente anterior e artéria circunflexa. Isto pode ser observado na Figura 3.3.



Figura 3.3: Superfície externa do coração

O volume do coração varia nos diversos indivíduos. As suas dimensões médias, em um homem adulto, são as seguintes:

- **comprimento:** 98 mm.
- **largura:** 105 mm.
- **circunferência:** 230 mm.
- **peso:** 275 g.

O coração da mulher tem dimensões inferiores de 5 a 10 milímetros, e pesa 5 a 10 gramas a menos.

3.3.2 Os vasos

Como dito anteriormente, O sangue circula em canais chamados de vasos sanguíneos. Existem três tipos de vasos sanguíneos: artérias, veias e capilares.

Os vasos que partem do coração e vão a periferia são chamados de artérias. Aqueles que seguem o percurso inverso, isto é, da periferia e se dirigem ao coração, são chamados de veias.

Em linhas gerais, nas artérias corre um sangue rico em oxigênio e em substâncias nutritivas. Elas o levam aos vários tecidos do organismo. Nas veias, ocorre o contrário, elas trazem o sangue da periferia para o coração. Este sangue é rico em anidrido carbônico e substâncias de rejeição, e é conhecido como sangue venoso. As substâncias de rejeição serão depois eliminadas pelos rins, que têm justamente a tarefa de filtrar o sangue.

As trocas gasosas, ou seja, a eliminação de anidrido carbônico e a absorção de oxigênio, têm lugar nos pulmões, por efeito da respiração.

As artérias, chegando à periferia do corpo humano, isto é, nos músculos, na pele, em todos os órgãos, se dividem em artérias sempre menores (arteríolas) até que o seu calibre se torna microscópico. É a este nível que têm lugar as trocas entre sangue e células. Os capilares são estes vasos microscópicos e formam nos tecidos e nos órgãos uma vasta rede. Os capilares confluem para pequenas veias (vênulas) que aos poucos se vão unindo umas com outras, tornam-se veias verdadeiras e trazem de volta o sangue ao coração.

Do coração partem duas grandes artérias: a artéria pulmonar e a artéria aorta. A artéria pulmonar tem a tarefa de levar o sangue aos pulmões. Depois de ter cedido o anidrido carbônico e de se ter carregado de oxigênio, o sangue volta ao coração pelas veias pulmonares. Todo esse conjunto constitui a pequena circulação. A artéria aorta leva o sangue ao resto do organismo e os seus numerosos ramos acabam formando a rede capilar de todos os órgãos. O sangue é trazido de volta ao coração pelas veias, que se reúnem em dois grossos troncos. Todo esse conjunto constitui a grande circulação.

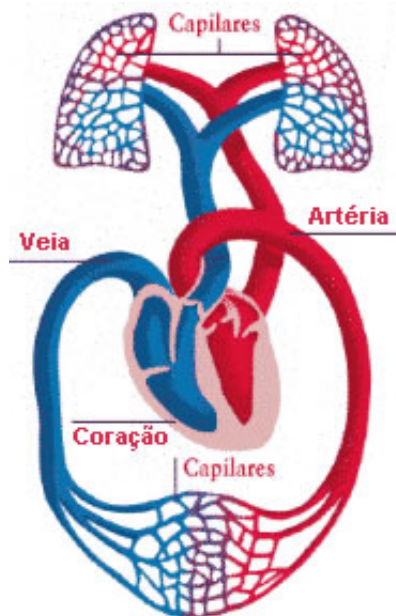


Figura 3.4: A pequena e a grande circulação

3.4 O ciclo cardíaco

O coração, para desempenhar a sua função de bomba, deve dilatar as suas cavidades, de modo que se encham de sangue, e, em seguida, comprimi-las, de modo que o sangue seja lançado nas artérias. Esta alternância de dilatações e de contrações se chama revolução cardíaca. A contração chama-se sístole e a dilatação diástole.

O coração se contrai cerca de 70 vezes por minuto, reproduzindo de forma periódica o que chamamos de ciclo cardíaco (ARAÚJO, 2002). É sabido, contudo, que certas pessoas têm o pulso menos freqüente (as batidas que sentimos no pulso não são outra coisa do que a expressão das contrações cardíacas), enquanto em certas moléstias, nos estados febris, nas crianças, a freqüência do pulso é maior.

A passagem do sangue, das aurículas para os ventrículos é regulado por uma válvula, situada entre as cavidades. Deste mesmo modo, a passagem do sangue dos ventrículos para as artérias é regulado por uma outra válvula.

Durante o ciclo cardíaco são repetidos uma série de fases. Vejamos como se sucedem as diversas fases.

A aurícula direita recebe o sangue das veias cavas, e a aurícula esquerda recebe o sangue das veias pulmonares. As aurículas se contraem no que chamamos de sístole auricular e o sangue é assim lançado nos ventrículos. A duração da sístole auricular é breve, apenas 1/10 de segundo. O sangue lançado pela contração auricular ocasiona, na passagem, a abertura das válvulas aurículo ventriculares, as quais se fecham quando a contração auricular terminou. Neste momento se contraem os ventrículos. O sangue é assim lançado nas artérias (pulmonar do ventrículo direito, aorta do ventrículo Esquerdo) determinando, na passagem, a abertura das válvulas sigmóides. Condição indispensável desta fase é a oclusão das válvulas aurículo-ventriculares, de outro modo o sangue refluiria para as aurículas. A duração da contração (sístole ventricular) é de 3/10 de segundo. Terminada a sístole ventricular, as válvulas sigmóides se fecham para impedir ao sangue de refluir nos ventrículos. A próxima fase é aquela de pausa, que é uma fase de recuperação, durante a qual o coração está em repouso. A sua duração é de 4/10 de segundo.

Durante a sístole auricular, os ventrículos estão em diástole, e vice-versa. Em outras palavras, o coração se contrai na metade superior (aurículas) e se dilata naquela inferior (ventrículos). Isto acontece quando o sangue passa das aurículas para os ventrículos. Sucessivamente a parte inferior se contrai, isto é, os ventrículos, e se dilata a parte superior, isto é, as aurículas.

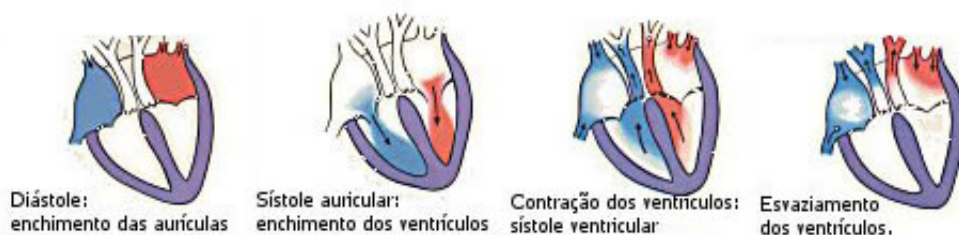


Figura 3.5: As quatro fases da contração cardíaca

3.5 A ativação elétrica do coração

O coração possui uma função elétrica que compreende a formação e a condução de estímulo.

O fenômeno elétrico que representa a atividade do miocárdio é consequência do automatismo do coração. Entendemos por automatismo a propriedade que o coração possui de gerar seus próprios estímulos (TRANCHESI, 1972).

As células do coração são altamente especializadas. Elas produzem e transmitem estímulos elétricos, formando um sistema com uma anatomia não apresentada por outros órgãos. Este sistema é constituído por alguns centros de excitação, o qual chamamos de nós. Estes nós geram os impulsos elétricos por algumas ramificações que os distribuem pelo coração, para que este se contraia (ARAÚJO, 2002).

O coração possui dois tipos fundamentais de células: as ditas de contração e as células chamadas específicas.

A célula miocárdia é dita de contração. Esta é formada por três componentes:

- um sistema para ativar e relaxar a célula, formado pela membrana celular, o sistema T e o retículo sarcoplasmático;
- um sistema para gerar força e provocar contração, o sarcômetro;
- um sistema para proporcionar energia para que a célula funcione, o mitocondrial.

As células específicas são as responsáveis pela formação e condução do estímulo. Estas correspondem a três tipos: células P, células transicionais e células de Purkinje.

Normalmente, os impulsos originam-se nas células P do nó sino-atrial (SA) que também é conhecido por nó sinusal ou de Keith-Flack. Estas também se encontram no nó átrio ventricular (NAV) e no feixe de His, sobretudo na união destas duas estruturas (zona N-H).

O conjunto formado pelos nós sino-atrial e átrio ventricular, pelas fibras do feixe de His e suas ramificações constitui o tecido de condução (Figura 3.6).

No nó sino-atrial encontram-se dois tipos principais de células específicas: as células P e as células transicionais. Estas possuem estrutura intermediária entre as células

P e as contráteis e conduzem mal o estímulo elétrico. Do ponto de vista funcional, as células transacionais são as únicas vias de ligação entre as células P e o restante da musculatura cardíaca.

Pelo seu tipo peculiar de potencial de transmembrana de ação, pode-se dizer que as células P do nó sino-atrial são as responsáveis pelo automatismo fundamental do coração, ao passo que as transacionais tem a função de levar o estímulo elétrico ao exterior do nó.

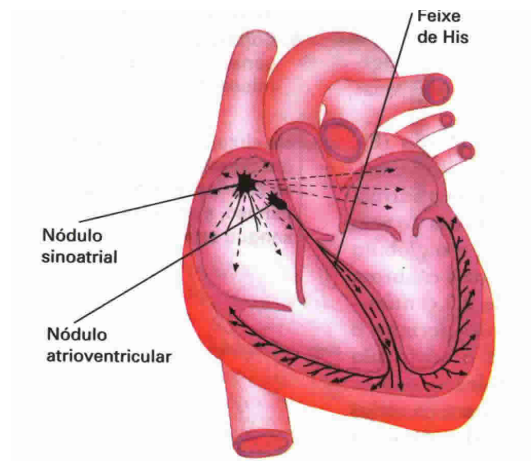


Figura 3.6: Conjunto formado pelos nó SA, nó NAV e pelo feixe de His

O processo tem início no nó sino-atrial. O estímulo então se propaga pelas aurículas e chega no nó átrio ventricular a partir de onde o estímulo se propaga por todo o miocárdio dos ventrículos.

A atividade elétrica do coração é estudada mediante a colocação de uma série de eletrodos no tórax. Estes recolhem e traduzem os estímulos elétricos em uma série de ondas. Na plotagem das ondas, se distingue a onda P que corresponde à contração das aurículas, e um consecutivo complexo QRS determinado pela contração dos ventrículos. Conclui o ciclo uma onda T. Muitas alterações cardíacas determinam uma modificação da onda eletrocardiográfica normal, de modo que o eletrocardiograma representa um preciso meio de diagnóstico.

Vale lembrar que, embora o coração se contraia de maneira automática, este também recebe estímulos do sistema nervoso central, o qual aumenta ou diminui o seu ritmo conforme as necessidades do organismo.

A seguir descreveremos distintamente algumas das estruturas que compõem o sistema de condução do coração.

- **Nó sino-atrial (SA):** localizado na união entre a veia cava superior e a aurícula direita. É considerado marca-passo cardíaco, por gerar regularmente potenciais de ação espontâneos, cuja frequência de despolarização determina a frequência dos batimentos cardíacos;
- **Nó átrio ventricular (NAV):** localizado na parte posterior direita do septo interatrial;
- **Fibras de condução internodais:** têm como função a transmissão rápida dos potenciais de ação do nó sino-atrial ao nó átrio ventricular.
- **Feixe de His:** origina-se no nó átrio ventricular e divide-se em um ramo direto e dois ramos esquerdos, estendendo-se pelo septo interventricular para baixo, até o ápice dos ventrículos.
- **Sistema de Purkinje:** encontra-se distribuído no miocárdio ventricular, projetando-se para cima, em direção à base. Ramifica-se extensamente, formando uma densa rede de fibras sob o endocárdio.

3.6 Derivações eletrocardiográficas

Na superfície do corpo existem diferenças de potencial, conseqüentes dos fenômenos elétricos gerados durante a excitação cardíaca. Para medir e registrar estas ondas são utilizados galvanômetros de tipo particular, os quais constituem as unidades fundamentais dos eletrocardiógrafos. Os pontos do corpo a serem explorados são ligados ao aparelho de registro por meio de fios condutores. Dessa forma, obtém-se as chamadas *derivações* que podem ser definidas de acordo com a posição dos eletrodos.

Por convenção, registram-se curvas ditas positivas (para cima da linha considerada como isoeétrica), quando um dos eletrodos, admitido como *explorador*, está orientado

para as áreas que se comportam como positivas em relação às que se encontram voltadas para o outro eletrodo (eletrodo indiferente).

A seguir descreveremos as derivações do plano frontal e as derivações do plano horizontal.

3.6.1 As derivações do plano frontal

As derivações na superfície do corpo podem ser infinitas. Diante disto, foi necessário estabelecer uma convenção, para que os registros obtidos pudessem ser comparados. Eithoven, baseados em estudos de ordem prática e teórica, estabeleceu três derivações, dispostas de modo a formar os lados de um triângulo equilátero (TRANCHESI, 1972). Este triângulo recebeu o nome de "Triângulo de Eithoven" e é ilustrado na Figura 3.7.

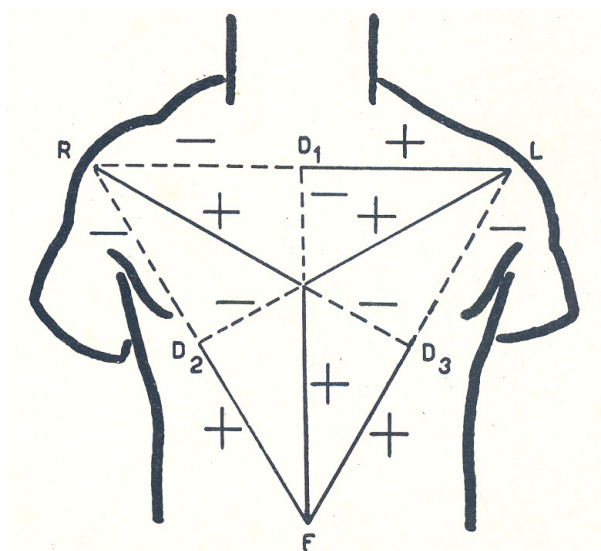


Figura 3.7: As linhas de derivação do plano frontal: D_1 , D_2 , D_3 , V_R , V_L e V_F . Triângulo de Eithoven

Assim a primeira derivação bipolar, também chamada de derivação D_1 , estuda a diferença de potencial (V) entre o braço esquerdo (L = "left") e o braço direito (R = "right"): $D_1 = V_L - V_R$.

A segunda derivação, também chamada de derivação D_2 , mede a diferença de potencial entre a perna esquerda (F = "foot") e o braço direito: $D_2 = V_F - V_R$.

A terceira derivação, também chamada de derivação D_3 , mede a diferença de potencial entre a perna esquerda e o braço esquerdo: $D_3 = V_F - V_L$.

As derivações bipolares dos membros são também chamadas de "Derivações Clássicas" ou "Standards".

O centro do "Triângulo de Eithoven" corresponde ao *centro elétrico do coração* ou centro aparente de origem dos vetores.

Se traçarmos retas perpendiculares ao "Triângulo de Eithoven" e que passem pelo centro do mesmo, teremos dividido as linhas de derivação em duas metades: uma positiva e outra negativa (Figura 3.7).

As derivações bipolares apenas medem a diferença de potencial entre dois pontos. Diante disto, quando fazemos $D_1 = V_L - V_R$, nada conhecemos sobre o valor absoluto de V_L ou V_R . Entretanto, se pudermos fazer $V_R = 0$, teremos: $D_1 = V_L - 0$. Isto nos ajudaria a medir o potencial absoluto do ponto L. Logo, devemos encontrar um ponto de potencial igual a zero e fazer uma derivação a partir daquele ponto até o local da superfície corporal do qual desejamos conhecer o potencial.

Supôs-se, a princípio, que um ponto bem afastado do coração teria um potencial bem próximo de zero. Logo, verificou-se que isso não era satisfatório.

Wilson propôs um sistema unipolar, baseando-se na segunda lei de Kirchoff. Ele demonstrou que o ponto resultante da união dos três eletrodos das derivações clássicas teria um potencial teoricamente igual a zero. De fato, dadas as condições de derivação, verifica-se que este terminal não se mantém como zero durante toda a revolução cardíaca, e que, varia com esta. Entretanto, dado as suas muito pequenas variações em relação a magnitude das correntes do coração, pode ser na prática considerado como um ponto de potencial nulo.

As correntes captadas com o sistema de Wilson inscreveram-se com relativa pequena voltagem, e por esta razão, Goldberger ideou as derivações unipolares aumentadas. Goldberger verificou que ao desligar o membro explorado da central terminal, a amplitude das deflexões aumentava em 50%. Ele chamou essas novas derivações de aV_R , aV_L e aV_F .

Denomina-se "linha de derivação unipolar" à linha que une o ponto explorado ao

centro elétrico do coração. Na Figura 3.7, a porção proximal voltada para o eletrodo explorador da linha de derivação unipolar é considerado como positiva. Diante disto, todo o vetor que se projetar sobre ela, dará origem a uma deflexão positiva no ECG (TRANCHESI, 1972).

3.6.2 As derivações do plano horizontal

O coração é um órgão tridimensional e por isto os vetores cardíacos têm uma orientação espacial. Como sabemos, bastam dois planos perpendiculares para orientar um vetor no espaço. Logo, se conhecermos as projeções de um vetor, nos planos frontal e horizontal, podemos localizar este vetor no espaço.

Além das seis derivações descritas em 3.6.1, são registradas outras seis derivações unipolares. São elas V_1, V_2, V_3, V_4, V_5 e V_6 . Elas exploram o fenômeno elétrico a partir da face anterior do tórax e são chamadas de "precordiais". Os seus pontos de exploração são ilustrados na Figura 3.8.

As derivações precordiais são consideradas como derivações do plano horizontal. Assim num segmento torácico que compreendesse a 4^a e a 5^a costelas e o 4^o e o 5^o espaços intercostais, teríamos as linhas de derivações precordiais como na Figura 3.9. Observando este segmento de cima para baixo, veríamos estas linhas projetadas como na Figura 3.10.

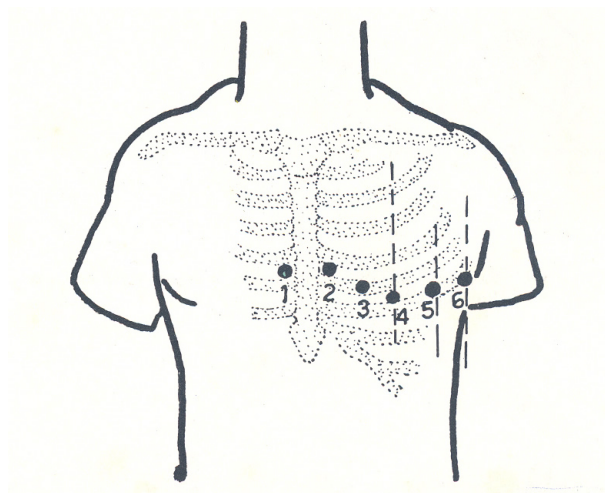


Figura 3.8: Pontos de aplicação dos eletrodos para o registro das derivações precordiais: V1 a V6

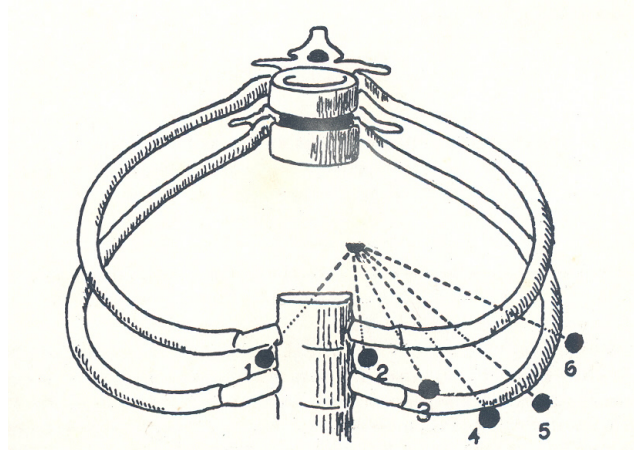


Figura 3.9: Corte horizontal do tórax. As linhas de derivações precordiais partem do 4º e do 5º espaços intercostais e se dirigem para o centro elétrico do coração.

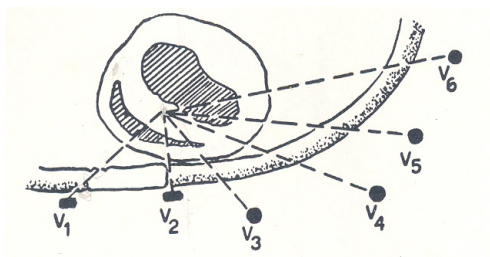


Figura 3.10: As derivações precordiais V1 a V6 e suas relações com um corte horizontal do coração.

3.7 Ondas, intervalos e segmentos do eletrocardiograma

O ECG é o registro da soma algébrica das variações de potencial produzidas pelas células cardíacas. Este nos fornece informações sobre a seqüência temporal dos eventos elétricos no coração. A atividade elétrica do coração produz potenciais detectados na superfície do corpo com amplitudes variando de 0,5 mV (milivolts) até 4 mV (TRAN-CHESI, 1972). A Figura 3.11 ilustra os nomes das várias ondas, intervalos e segmentos que ocorrem no ECG humano.

Na Figura 3.11 podemos distinguir diferentes configurações, as quais são definidas como:

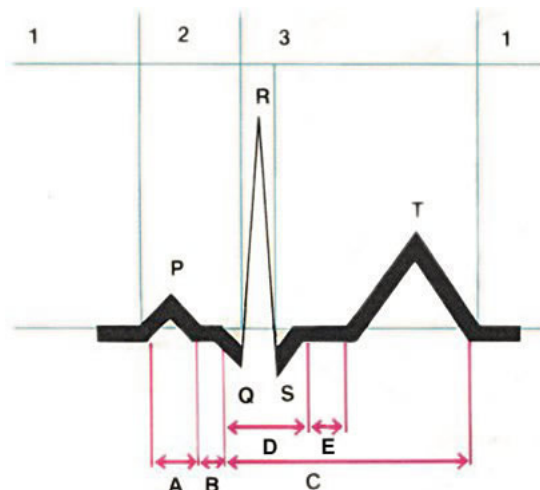


Figura 3.11: As deflexões, segmentos e intervalos do eletrocardiograma normal.

- **Ondas:** são as deflexões P, QRS (complexo), T e U.
- **Intervalos:** intervalo P-R - vai do início da onda P até o início do complexo QRS. Compreende, portanto, a onda P e o segmento PR. É ilustrado na figura como sendo $A + B$.
Intervalo Q-T - vai do início do complexo QRS até o fim da onda T. Compreende todo o fenômeno da sístole elétrica ventricular (despolarização e repolarização), isto é, o complexo QRS, o segmento ST e a onda T. É ilustrado na figura como sendo C .
- **Segmentos:** segmento PR ou PQ - vai do fim da onda P até o início do complexo QRS. É ilustrado na figura como sendo B .
Segmento ST - vai do fim do complexo QRS, até o início da onda T. É ilustrado na figura como sendo E .

A onda de despolarização que se espalha pelos átrios após a despolarização do módulo sino-atrial, produz a onda P. Quando todas as fibras atriais estão na fase de platô de seus potenciais de ação, o traçado do ECG retorna à sua linha de base. O próximo evento evidente do ECG é o complexo de QRS, resultante da despolarização dos ventrículos. Enquanto os ventrículos estão despolarizados, os átrios estão se repolarizando. No entanto

o sinal gerado pelos átrios é mascarado pelo sinal de maior intensidade proveniente dos ventrículos. Quando todas as fibras dos ventrículos atingem a fase platô, o traçado do ECG novamente retorna a sua linha de base. Em seguida ocorre a repolarização ventricular, que dá origem a onda T (TRANCHESI, 1972).

A seguir descreveremos algumas características das ondas, intervalos e seguimentos.

3.7.1 A onda P

A onda P é analisada observando um conjunto de características, as quais é descrito abaixo:

- **Eixo:** no plano frontal o eixo de P fica entre 0° e 90° (onda P positiva em D_1 , D_2 e aV_F e negativa em aV_R), considerado o vetor normal dirigido para baixo e para a esquerda.

No plano horizontal, o vetor se dirige para frente (onda P positiva em V_1). Em V_1 , a onda P pode ser difásica, tipo plus-minus. Quando isso ocorre a fase positiva deve ser maior do que a negativa.

- **Amplitude:** a maior amplitude não deve exceder 2,5 mm (0,25 mV).
- **Morfologia:** arredondada e monofásica, podendo ser difásica em V_1 .
- **Duração:** duração máxima é de 0,10s.

3.7.2 O intervalo PR

É medido do início da onda P até o início do QRS. Varia de 0,12s a 0,20s e representa o tempo que o impulso gerado pelo nó SA, levou para atingir as Fibras de Purkinje.

3.7.3 O complexo QRS

O complexo QRS é analisado observando um conjunto de características, as quais são descritas abaixo:

- **Eixo:** a faixa de variação do eixo do QRS no plano frontal é de -30° a $+120^\circ$. No plano horizontal, o vetor médio do QRS é orientado para trás.
- **Amplitude:** diz-se que existe baixa voltagem, quando não se registra qualquer deflexão maior do que 5mm em derivação bipolar ou se a maior deflexão no plano horizontal não ultrapassa 8mm.

Alta voltagem é definida quando se registra ondas R ou S > 20 mm nas derivações frontais ou, no plano horizontal, ondas S (V_1/V_2) ou ondas R (V_5/V_6) > 30 mm.

- **Morfologia:** varia de acordo com a derivação e a posição elétrica do coração. A onda Q é a primeira deflexão negativa do QRS e representa a ativação septal. A onda Q patológica é definida quando exceder 25% do tamanho da R e duração $> 0,04$ s. Em algumas derivações, estes limites podem ser ultrapassados (aV_R , aV_L e D_3). A presença da onda Q em V_1 , V_2 e V_3 deve ser sempre considerada anormal. A ausência de onda Q em V_5 e V_6 também é anormal.

3.7.4 A onda R

É a primeira deflexão positiva do QRS e representa fundamentalmente a ativação das paredes livres. Normalmente deve progredir de amplitude de V_1 para V_6 .

3.7.5 A onda S

É a segunda deflexão negativa do complexo QRS e representa a ativação das porções basais dos ventrículos. Normalmente deve diminuir de amplitude de V_1 para V_6 .

- **Duração:** o complexo QRS deve ter duração máxima de 0,12s. Deflexão intrinsecóide é o tempo de ativação ventricular. Medido do início do QRS até o vértice da onda R, deve ser no máximo de 0,045s.

O aumento da deflexão intrinsecóide pode ocorrer por: hipertrofia ventricular, bloqueio de ramo, bloqueio divisional ou infarto agudo do miocárdio.

3.7.6 O Segmento ST

Começa no ponto J (término do QRS) e termina na porção ascendente da onda T. Normalmente a primeira porção do segmento ST é isoelétrica. Desníveis do segmento ST podem ocorrer por múltiplas causas, sejam elas primárias (corrente de lesão do IAM) ou secundárias (hipertrofias, bloqueios de ramo entre outros).

3.7.7 A onda T

Sua orientação segue o vetor médio do QRS. Tem morfologia tipicamente assimétrica, com a porção inicial mais lenta. Não deve exceder 5mm nas derivações frontais ou 10 mm nas precordiais. Sua polaridade pode ser muito variável, sendo obrigatoriamente positiva em V_5 e V_6 e obrigatoriamente negativa em aV_R .

3.7.8 O intervalo QT

É medido do início do QRS até o final da onda T e representa o tempo de ativação e recuperação do miocárdio ventricular. O intervalo QT, varia com a idade, sexo e muito com a frequência cardíaca.

Portanto, deve ser corrigido através da fórmula de Bazzet: (O limite superior para homens fica em torno de 0,425s e para mulheres em torno de 0,440s).

Capítulo 4

DICOM

O DICOM é uma padrão internacional que engloba diversos aspectos de aquisição de uma imagem ou onda em formato digital. Nele além dos dados codificados, são inclusos parâmetros para visualização destas imagens, a forma de arquivamento em disco, informações sobre transferência em rede de computadores entre outros.

O DICOM utiliza modelos explícitos e detalhados de como entidades reais (pacientes, imagens, ondas, relatórios) envolvidos no contexto de aquisição de um exame, são descritos e como são relatados. São estes modelos chamados de entidade-relacionamento (E-R) que irão garantir que fabricantes e usuários de equipamentos compreendam e utilizem o padrão de forma adequada.

No modelo de dados DICOM, um paciente tem um ou mais estudos. Um estudo corresponde a uma visita do paciente a uma instituição de saúde. Cada estudo contém uma ou mais séries, que são seqüências de exames da mesma modalidade. Uma série contém imagens, laudos, curvas ou outros objetos (BORTOLUZZI, 2003).

Apesar de sua importância na área de informática médica, radiologia e sistemas de informação hospitalar (HIS), o padrão DICOM não é largamente conhecido (SAMPAIO, 1999). Neste capítulo iremos apresentar vários conceitos definidos pelo padrão e seu histórico de evolução.

4.1 Evolução histórica

O comitê ACR-NEMA foi criado em 1983 com o objetivo de desenvolver meios padronizados para interconexão de equipamentos de imagens médicas digitais a dispositivos de visualização e armazenamento. Este comitê é formado pelo *American College of Radiology* (ACR) e a *National Electrical Manufacturers Association* (NEMA). Além de especificações de conexão de *hardware*, o padrão deveria incluir um dicionário de dados que possibilitasse a interpretação e a visualização correta das imagens (BORTOLUZZI, 2003; SAMPAIO, 1999; CARVALHO, 1998).

O comitê estudou diversos padrões propostos, mas nenhum demonstrou ser inteiramente satisfatório. No entanto, alguns desses padrões sugeriram idéias bastante úteis. Por exemplo, o *American Association of Physicists in Medicine* (AAPM), tinha desenvolvido um ano antes, um formato para gravar imagens em dispositivos magnéticos. Este formato era constituído por um cabeçalho e por elementos de dados. Estes conceitos foram adotados pelo comitê por parecer particularmente interessante (SAMPALIO, 1999; CARVALHO, 1998).

A primeira versão do padrão foi publicada em 1985. Em 1988 foi publicada a segunda versão do padrão. Esta usou substancialmente a mesma especificação de *hardware* da primeira versão, mas adicionou novos elementos de dados e corrigiu vários erros e inconsistências da versão anterior. Embora bastante utilizada, esta nova versão ainda não possuía uma comunicação de rede robusta para conectar diretamente os equipamentos. Para resolver este problema o comitê decidiu que desenvolver uma interface para suportar redes requeria muito mais que adicionar partes ao padrão. Diante disso, o projeto inteirou passou por um processo de reengenharia e foi adotado um novo método de concepção: a orientação a objetos (BORTOLUZZI, 2003; SAMPAIO, 1999; CARVALHO, 1998).

A terceira versão do padrão foi publicada em 1992, e é um padrão maior do que as versões anteriores e possibilita o uso de um número maior de recursos. A concepção básica consistia que uma aplicação médica pudesse comunicar com outros dispositivos através de uma série de protocolos de rede. A pilha existente na segunda versão, que definia uma ligação ponto-a-ponto é um dos protocolos. Os outros dois são o TCP/IP

(*Transmission Control Protocol/Internet Protocol*) e o ISO/OSI(*International Standards Organization/Open Systems Interconnection*). (BORTOLUZZI, 2003).

4.2 Organização do padrão

Diferente das versões anteriores, a terceira versão do padrão DICOM encontra-se dividida em várias partes. Tal divisão permite que cada parte possa expandir-se individualmente, sem haver a necessidade de republicar todo o padrão. Dentro das partes, as seções sujeitas a adições ou modificações encontram-se em suplementos, reduzindo assim o esforço de edição necessário quando da atualização das mesmas. As partes também são conhecidas como PS (*Part of Standard*).

Atualmente a terceira versão compreende 16 partes distintas e 98 suplementos (CLUNIE, 2004). De forma a dar uma idéia geral do conteúdo do padrão, descreve-se resumidamente cada uma das partes constituintes:

- **PS 3.1: Introdução e Visão Geral (*Introduction and Overview*)**

É o documento que proporciona uma visão geral sobre o padrão. Justifica a existência do padrão, define muitos dos termos utilizados e faz uma breve descrição das outras partes que o constituem.

- **PS 3.2: Conformidade (*Conformance*)**

Define os princípios para todas as implementações que desejam estar em conformidade com o padrão.

- **PS 3.3: Definição de Objetos de Informação (*Information Object Definitions*)**

Este documento descreve como são definidos os objetos de informação (IOD) e apresenta todas as classes de IODs definidas no padrão.

- **PS 3.4: Especificações das Classes de Serviço (*Service Class Specifications*)**

Contém as especificações das classes de serviço. As regras de *Service Class User* e *Service Class Provider* também são definidos nesta parte, apresentando o comportamento esperado de cada uma delas.

- **PS 3.5: Estruturas de Dados e Codificação (*Data Structures and Encoding*)**

Este documento descreve a estrutura de dados e o método de codificação dos dados de um arquivo DICOM.

- **PS 3.6: Dicionário de Dados (*Data Dictionary*)**

Este documento apresenta uma listagem completa de todos os elementos de dados, detalhando suas partes constituintes e quais tipos de valores são permitidos para cada tipo.

- **PS 3.7: Intercâmbio de Mensagens (*Message Exchange*)**

Este documento descreve as operações e os protocolos usados para trocas de mensagens. Estas operações são usadas para executar os serviços definidos pelo PS 3.4.

- **PS 3.8: Suporte para Comunicação em Rede para Intercâmbio de Mensagens (*Network Communication Support for Message Exchange*)**

Este documento define os serviços e protocolos usados para trocar mensagens diretamente em redes OSI e TCP/IP. É considerada a parte mais complexa para implementadores DICOM (SAMPAIO, 1999).

- *** PS 3.9: Suporte para Comunicação Ponto-a-Ponto para Intercâmbio de Mensagens (*Point to Point Communication Support for Message Exchange*)**

Este documento define os serviços e protocolos usados para trocar mensagens em redes ponto-a-ponto.

- **PS 3.10: Mídia de Armazenamento e Formato do Arquivo para a Troca de dados (*Media Storage and File Format for Data Interchange*)**

Este documento define os formatos lógicos para armazenar informação DICOM em vários meios de armazenamento.

- **PS 3.11: Perfis de Aplicação de Mídias de Armazenamento (*Media Storage Application Profiles*)**

Este documento define uma forma de usuários e fabricantes especificarem a seleção de meio de armazenamento (mídia) entre os existentes no PS 3.12 e os objetos de informação definidos no PS 3.3.

- **PS 3.12: Formatos de Mídia e Mídia Física para Intercâmbio de Dados (*Media Formats and Physical Media for Data Interchange*)**

Este documento referencia as especificações da indústria para os meios físicos de armazenamento e seus correspondentes formatos.

- * **PS 3.13: Suporte para Comunicação e Gerenciamento de Impressão Ponto-a-Ponto (*Print Management Point-to-Point Communication Support*)**

Este documento descreve os serviços e protocolos necessários para suportar a comunicação de entidades de impressão DICOM em redes ponto-a-ponto.

- **PS 3.14: Função de Exibição Padrão para Escalas de Cinza (*Grayscale Standard Display Function*)**

Este documento define um padrão para visualização de imagens em tons de cinza.

- **PS 3.15: Perfis de Segurança (*Security Profiles*)**

Este documento define técnicas de criptografia para ser usado com um arquivo DICOM seguro em mídias de armazenamento seguras.

- **PS 3.16: Recurso de Mapeamento de Conteúdo (*Content Mapping Resource*)**

Este documento descreve o DCMR (*DICOM Content Mapping Resource*), o qual define os modelos (*templates*) e os grupos de contexto usados em alguma parte do padrão.

Vale lembrar que os documentos marcados com * foram retirados do padrão por estarem obsoletos.

4.3 Conceitos gerais sobre o padrão

O padrão DICOM utiliza uma terminologia própria para descrever contexto, relacionamento e outros. A Figura 4.1 apresenta as principais estruturas do modelo de informação DICOM.

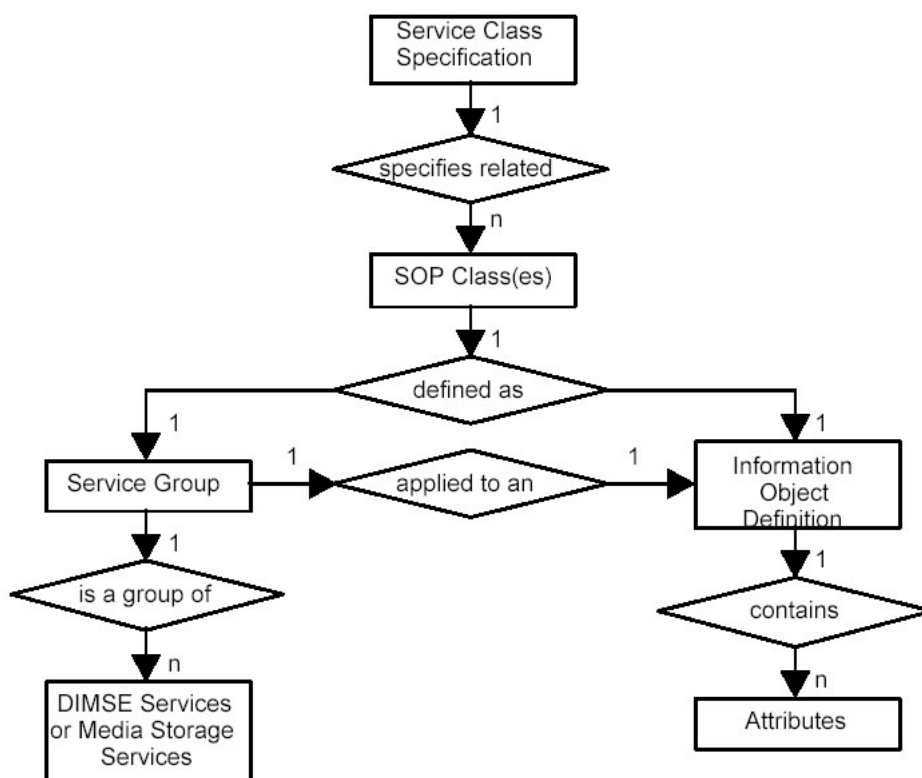


Figura 4.1: Principais estruturas do modelo de informação DICOM

Nas próximas subseções iremos apresentar uma descrição das estruturas da Figura 4.1, visando um maior entendimento do modelo de informação DICOM.

4.3.1 Classes de serviço e classes SOP

O relacionamento entre ambos os parceiros é definido pela descrição de uma *Service Class* (Classe de Serviço). A classe de serviço descreve explicitamente os papéis que os parceiros desempenham. O contexto dos serviços é definido de acordo com a classe de serviço em questão. No padrão DICOM, os papéis que cada aplicação pode assumir são

nomeados como *Service Class User* (SCU), ou cliente e *Service Class Provider* (SCP), ou servidor (DELLANI, 2001).

Parte da classe de serviço é a descrição da informação e das operações. Estas operações são combinadas com a definição da classe, denominadas por *Service Object Pair Class* (SOP). Em cada definição de uma classe SOP, um único *Information Object Definition* (IOD), é combinado com um ou mais grupos de serviços. Para cada um destes grupos de serviços, são fixados os detalhes dos papéis que ambos os parceiros têm de desempenhar. Pode existir mais do que uma classe SOP em uma classe de serviço em que estejam envolvidos mais do que um IOD. Uma classe de serviço denota o relacionamento de informação definido em diferente IODs (CARVALHO, 1998; DELLANI, 2001).

As classes SOP identificam as capacidades de processamento distribuído específicas para uma classe de serviço. Quando os parceiros concordam em usar determinada classe SOP, ambos devem assegurar que desempenharão seu papel como descrito, usando o contexto da classe de serviço em questão. Antes que a troca de informação tenha início, a identificação da classe SOP é um ponto importante que deve ser negociado entre os parceiros. O mecanismo usado depende do tipo de troca: via rede ou meios de armazenamento (CARVALHO, 1998).

Usando a classe de serviço e outras definições, os parceiros em um ambiente distribuído funcionam juntos, através dos serviços fornecidos pelo domínio de transferência (*exchange domain*) (DELLANI, 2001).

4.3.2 Definição de objeto de informação - IOD

Um IOD é um modelo abstrato de dados orientado a objeto usado para especificar informações de objetos do mundo real (DELLANI, 2001). Ele é composto por uma coleção de pedaços relacionados, agrupados em Entidades de Informação (*Information Entities*). Dependendo do contexto definido pela classe de serviço, um IOD consiste numa única entidade de informação e designa-se por IOD normalizado (*normalized IOD*), ou numa combinação de entidades de informação designando-se por IOD composto (*composite IOD*). As classes de serviço que implementam funções de gestão usam IODs normali-

zados, enquanto as que manipulam o fluxo de dados de imagem usam IODs compostos (CARVALHO, 1998).

As entidades de informação consistem de um conjunto de atributos que descrevem um elemento básico do mundo real, como um paciente, um estudo, uma imagem, entre outros. Os atributos relacionados podem ser agrupados em *Information Object Modules* (IOM). Este módulos possuem a vantagem de possuir atributos com a mesma descrição semântica. Além disso, estes módulos são definidos de forma a poderem ser usados em mais do que um IOD.

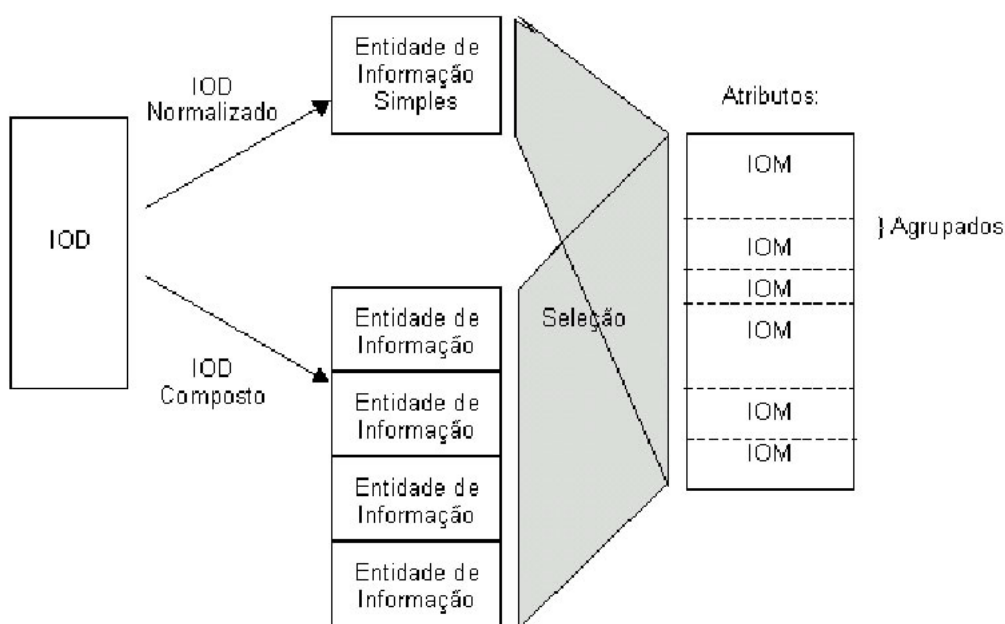


Figura 4.2: Definições de Objetos de Informação e suas relações

4.3.3 Atributos

Os atributos são as entidades de informação básicas do padrão, e suas características são apresentadas em detalhe:

- **Nome (*Attribute Name*):** Contém o nome do atributo, o qual é único e humanamente legível.

- **Identificador (*Attribute Tag*):** Contém o identificador do atributo, o qual é único e geralmente representado por um par de números em representação hexadecimal.
- **Descrição (*Attribute Description*):** Descrição do atributo.
- **Representação de Valor (*Value Representation*):** Funciona como uma idéia de tipo de dados para o atributo. É especificada no PS 3.5 do padrão.
- **Multiplicidade do Valor (*Value Multiplicity*):** Especifica o número de valores que podem ser codificados para um atributo. É especificada no PS 3.6 do padrão.
- **Classificação do Tipo (*Type Classification*):** Especifica o uso do atributo relacionado a classe SOP e com o papel de SCU ou SCP. Dependendo da situação, o atributo ou é obrigatório e tem que ter um valor (Tipo 1), ou é obrigatório com ou sem valor (Tipo 2), ou é opcional (Tipo 3). Ainda existem os tipos 1C, 2C, onde C corresponde a condicional. Estes seguem a mesma idéia dos outros, só que é apresentada uma condição na descrição do atributo para que estes sejam obrigatórios.

4.3.4 Elementos de serviço

Os elementos de serviço são as operações permitidas em IODs para uma certa classe SOP. O grupo de elementos de serviço pertencentes a uma classe SOP é chamado de Grupo de Serviço (*Service Group*). Basicamente, temos os serviços de associação e apresentação de dados da Camada Superior OSI de Serviços (*OSI Upper Layer Service*) definidos no PS 3.8 e os serviços do protocolo DIMSE (*DICOM Message Service Element*).

4.3.5 Instâncias SOP

Após o acordo de quais classes SOP suportadas (e implicitamente também as classes de serviço) e de como são divididos os papéis de SCU e SCP, instâncias da classe SOP podem ser transferidas entre as duas partes. Para isto, têm de ser fornecidos atributos com

os valores corretos (em termos de semântica) e armazenados dentro de uma instância SOP (*SOP Instance*) como especificado pelas definições de atributo (CARVALHO, 1998).

A informação coletada vai ser codificada nos formatos definidos pelo padrão DICOM, usando a etiqueta (*Tag*) e a Representação de Valor (*Value Representation*), para criar um Conjunto de Dados DICOM (*DICOM Data Set*), no qual cada atributo é codificado em um Elemento de Dados (*Data Element*). Este conjunto de dados é passado ao Provedor de Serviço de Transferência (*Exchange Service Provider*) o qual garante que o outro lado da associação receba o mesmo conjunto de dados. As diferenças nas representações específicas de cada sistema são tomadas em conta durante a troca, assegurando deste modo que os valores semânticos permaneçam intactos. O receptor do conjunto de dados irá decodificá-lo para extrair a informação necessária e agir de acordo pela semântica da classe SOP (DELLANI, 2001).

4.3.6 Identificação

Durante o processo de criação de uma instância SOP é gerada uma identificação como atributo dessa mesma instância SOP. A identificação é para ser utilizada pelos sistemas de informação mais do que pelo uso humano. Esta identificação tem dois objetivos: a identificação da classe e a identificação da instância.

Essa identificação tem de ser utilizada em um ambiente de múltiplos fornecedores em diferentes partes do mundo. Para assegurar que cada identificação seja globalmente singular, é usado um mecanismo para gerar um conjunto de caracteres, chamado de Identificador Único (*Unique Identifier*) ou UID, cujo formato é o seguinte:

<raiz>.<sufixo>

A raiz é fornecida por uma autoridade que garante que ninguém mais irá utilizá-la. Esse número é alocado por organizações de padronização para companhias tais como fabricantes, hospitais e clínicas. O sufixo é criado dinamicamente por cada sistema quando da criação da instância. Vale lembrar que existem ainda UIDs definidos no PS 3.6 do padrão, que são utilizados para representar sintaxes de transferências, classes SOP, entre

outros. Maiores informações sobre o registro e o processo de codificação de um UID são encontrados nos anexos B e C do PS 3.5.

4.3.7 Relações

Os UIDs além de identificarem a classe SOP e a instância SOP, são também utilizados para identificar a relação entre as instâncias. Em uma instância composta, a qual contém uma única imagem pertencente a uma série de imagens, a entidade de informação (IE) que contém a informação da série será comum para todas essas instâncias. A relação é identificada pela utilização do mesmo UID para o atributo "*Series Instance UID*" em todas as instâncias compostas pertencentes àquela série. Neste caso só é necessário um UID de instância, pois o atributo por si identifica qual tipo de entidade de informação é utilizada.

No caso de instâncias normalizadas, só são possíveis referências a instâncias fora de si próprias, sendo necessária a combinação da identificação da classe com a da instância.

4.3.8 Representações de Valor - VR

Uma representação de valor descreve como um atributo é codificado em um elemento de dados.

O conhecimento da representação de valor é compartilhado pelos parceiros na troca de informação. O processo de codificação e decodificação tem de ter o cuidado de utilizar a VR correta para um atributo (identificada por sua etiqueta).

Há duas maneiras de compartilhar essa informação: compartilhando um dicionário de dados que contém todos os possíveis atributos, ou a inclusão da representação de valor como uma parte do elemento de dados. A última opção, apesar de aumentar a quantidade de dados a ser transferida, é muito mais flexível comparado ao uso de um dicionário de dados compartilhado. Quando a representação de valor está incluída, diz-se que a mensagem é codificada com VR explícito (*Explicit VR*), caso contrário diz-se que a codificação é feita com VR implícito (*Implicit VR*).

4.3.9 Elementos de Dados

É um atributo de um IOD codificado, que é composto no mínimo por três campos:

- Um campo contendo uma etiqueta (*Tag*);
- Um campo com o Tamanho do Valor (*Value Length*);
- Um campo contendo o Valor (*Value*).

Para algumas sintaxes de transferências específicas, um elemento de dados também contém um campo de VR, onde a Representação do Valor é explicitamente especificada. Isto é visto com mais detalhes na seção 5.2.1 do capítulo 5.

4.3.10 Sintaxe de Transferência

Quando um conjunto de dados de uma instância SOP é transferido via rede de computadores ou armazenada em algum meio de armazenamento, a forma de como este está codificado em um *byte stream* precisa ser conhecido. A codificação é especificada por uma Sintaxe de Transferência (*Transfer Syntax*). Existem três aspectos definidos pela sintaxe de transferência:

- como é especificada a VR;
- como é feita a ordenação dos *bytes* (*little endian* ou *big endian*);
- no caso de compressão, qual é o formato de compressão.

De forma análoga à identificação da classe SOP, a sintaxe de transferência é identificada por um UID.

Capítulo 5

Codificação de um arquivo DICOM

Segundo o PS 3.10 do padrão, um arquivo DICOM é formado por duas partes distintas: um cabeçalho e um conjunto de dados. Estas partes são ilustradas a seguir:

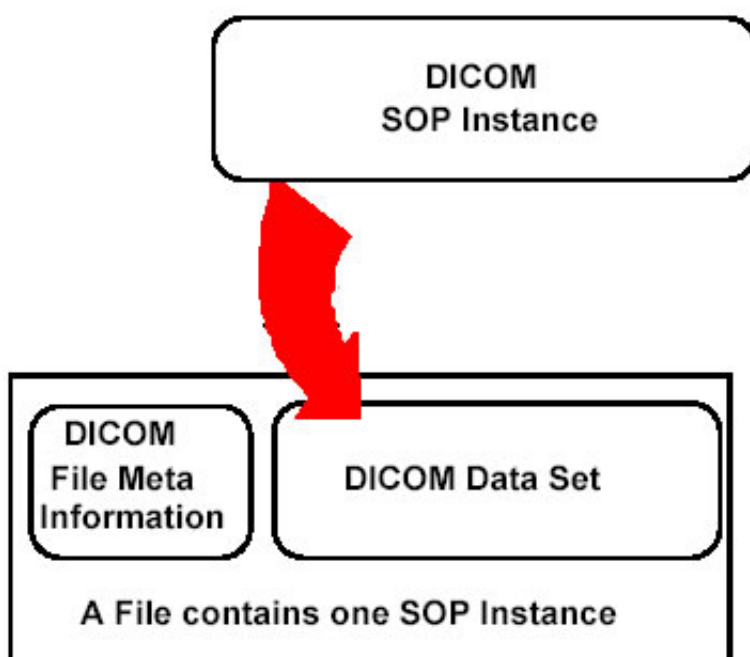


Figura 5.1: Formato do arquivo DICOM

5.1 Cabeçalho - File Meta Information

O cabeçalho (*File Meta Information*) inclui informação sobre o conjunto de dados encapsulado e deve estar presente em todo arquivo DICOM. Este consiste em um preâmbulo de 128 *bytes*, seguidos de um prefixo DICOM de 4 *bytes*, seguidos pelos elementos de cabeçalho.

O preâmbulo do arquivo é usado em implementações específicas. Por exemplo, o preâmbulo pode conter informações para ativar o acesso randômico de imagens contidas em um conjunto de dados de uma aplicação multimídia. Logo, este arquivo poderia ser acessado de duas maneiras: por esta aplicação multimídia que usa o preâmbulo e por uma aplicação DICOM que ignora o preâmbulo. Se o preâmbulo não for utilizado, todos os 128 *bytes* deverão ter o valor 0x00.

Os quatro *bytes* do prefixo DICOM deverão ser codificados em letras maiúsculas contendo o *string* de caracteres "DICM". Este caracteres devem estar de acordo com a norma ISO 8859 G0.

Exceto pelo preâmbulo e pelo prefixo DICOM, o cabeçalho deve sempre ser codificado usando ordenação de *bytes little endian* e VR explícito.

5.2 Conjunto de dados - Data Set

Um conjunto de dados representa uma instância do mundo real, que é representada no padrão como um IOD. Um conjunto de dados é composto por vários elementos de dados, os quais contém o valor de determinados atributos. O conteúdo específico e semânticos destes atributos são descritos no seu correspondente IOD. Uma lista com todos os IODs existentes no padrão pode ser encontrado no PS 3.3 do padrão.

Cada arquivo DICOM deve conter um único conjunto de dados representado uma única instância SOP relacionada com sua classe SOP e correspondente IOD (Figura 5.1).

5.2.1 Campos de um Elementos de Dados

No capítulo 4 seção 4.3.9, é apresentada uma breve descrição de elementos de dados. Nesta seção iremos apresentar mais alguns conceitos e a maneira correta de como codificá-los.

Um elemento de dados é unicamente identificado por uma etiqueta (*Tag*). Os elementos de dados contidos em conjunto de dados devem ser ordenados em ordem crescente do valor da etiqueta e devem ocorrer uma única vez. Os elementos de dados possuem ainda três campos obrigatórios e um campo opcional. Isto é ilustrado na figura

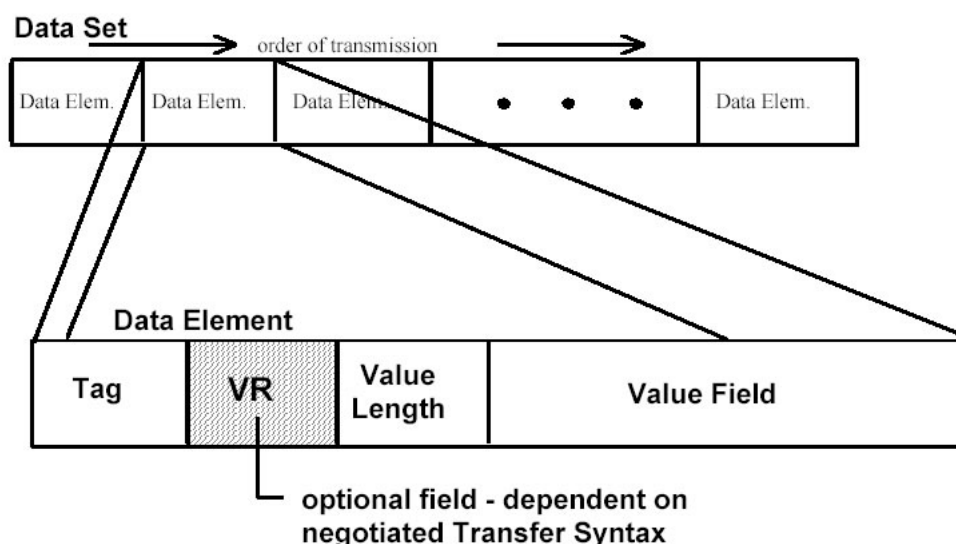


Figura 5.2: Conjunto de dados DICOM e as estruturas de um elemento de dados

As definições para estes campos são descritas a seguir:

- **Etiqueta (*Tag*):** É formado por um par de inteiros não sinalizados de 16 *bits* representando o Número do Grupo (*Group Number*) seguido do Número do Elemento (*Element Number*).
- **Representação de Valor (VR):** É formado por um *string* de caracteres de 2 *bytes* contendo o VR do elemento de dados. O VR para uma determinada etiqueta é definido pelo dicionário de dados como especificado no PS 3.6 do padrão. Os

dois caracteres VR devem ser codificados usando caracteres a partir do conjunto de caracteres padrão definidos pelo DICOM.

- **Tamanho do Valor (*Value Length*):** Este campo pode conter dois tipo de valores:
 1. um inteiro não sinalizado de 16 ou 32 *bits* (dependendo se o VR está implícito ou explícito) contendo o tamanho explícito do campo de valor (*Value Field*) definido em *bytes*. Este campo não inclui o tamanho da etiqueta, do VR e do campo tamanho de valor.
 2. um inteiro de 32 *bits* setados para 0xffffffff para representar que o tamanho do campo de valor é indefinido. Isto é usado para elementos de dados cujo VR representa uma Seqüência de Items (SQ) e para um elemento desconhecido (Unknown - UN). Para elementos de dados com VR OW ou OB, o tamanho indefinido também pode ser usado, o que vai depender da sintaxe de transferência.

- **Campo de Valor (*Value Field*):** Um número par de *bytes* contendo o(s) valor(es) do elemento de dados. O tipo de dado armazenado neste campo é especificado pelo VR. O Valor de Multiplicidade (VM) especifica quantos valores podem ser colocados dentro do campo de valor. se o VM é maior que um, múltiplos valores devem ser delimitados dentro do campo de valor, segundo a definição da seção 6.4 do PS 3.5 do padrão.

5.2.2 Estrutura de um Elemento de Dados com VR explícito

Quando utilizamos estruturas com VR explícito, os elementos de dados devem ser codificados com quatro campos consecutivos: Etiqueta, VR, Tamanho de Valor e Valor. Dependendo do VR do elemento de dados, este poderá ser codificado de duas formas:

1. para VRs com valor OB, OW, OF, SQ, UN e UT, os 16 *bits* que seguem os dois caracteres do campo VR, são reservados para uso futuro do padrão DICOM. Estes

bytes reservados devem ser setados para 0x0000 e não devem ser usados ou decodificados (Tabela 5.1). O campo tamanho de valor é um inteiro não sinalizado de 32 *bits*. Se o campo de valor tem um tamanho explícito, então o campo tamanho de valor deve conter o número de *bytes* do campo de valor. Entretanto, se o campo de valor possuir um tamanho indefinido, deverá ser utilizado um Delimitador de Sequência (Sequence Delimitation Item), para marcar o fim do campo de valor. Isto será visto com mais detalhes na seção 5.2.6 deste capítulo. Vale lembrar que elementos de dados com VR igual a UT nunca deverão ter tamanho indefinido, isto é, o campo tamanho de valor igual a 0xffffffff.

- para todos os outros VRs, o campo tamanho de valor deve ser um inteiro não sinalizado de 16 *bits*, seguindo os dois caracteres do campo VR (Tabela cap:cod:tab:VRExp2). Ainda, o campo tamanho de valor, deve conter o número de *bytes* do campo de valor.

Tag		VR		Value Length	Value
Group Number (16-bit unsigned integer)	Element Number (16-bit unsigned integer)	VR (2 byte character string) of "OB", "OW", "OF", "SQ", "UT" or "UN"	Reserved (2 bytes) set to a value of 0000H	32-bit unsigned integer	Even number of bytes containing the Data Element Value(s) encoded according to the VR and negotiated Transfer Syntax. Delimited with Sequence Delimitation Item if of Undefined Length.
2 bytes	2 bytes	2 bytes	2 bytes	4 bytes	'Value Length' bytes if of Explicit Length

Tabela 5.1: Elementos de dados com VR explícito do tipo OB, OW, OF, SQ, UT ou UN.

Tag		VR	Value Length	Value
Group Number (16-bit unsigned integer)	Element Number (16-bit unsigned integer)	VR (2 byte character string)	(16-bit unsigned integer)	Even number of bytes containing the Data Element Value(s) encoded according to the VR and negotiated Transfer Syntax.
2 bytes	2 bytes	2 bytes	2 bytes	'Value Length' bytes

Tabela 5.2: Elementos de dados com VR explícito de tipo diferentes dos da tabela 5.1

5.2.3 Estrutura de um Elemento de Dados com VR implícito

Quando utilizamos estruturas com VR implícito, os elementos de dados devem ser codificados com três campos consecutivos: Etiqueta, Tamanho de Valor e Valor. Se o campo de valor tem um tamanho explícito, então o campo tamanho de valor deve conter o número de *bytes* do campo de valor. Caso o campo de valor contenha um valor indefinido, O campo tamanho de valor deverá ser igual a 0xffffffff e um Delimitador de Seqüência (*Sequence Delimitation Item*) deverá ser incluído, para marcar o fim do campo de valor.

Tag		Value Length	Value
Group Number (16-bit unsigned integer)	Element Number (16-bit unsigned integer)	32-bit unsigned integer	Even number of bytes containing the Data Elements Value encoded according to the VR specified in PS 3.6 and the negotiated Transfer Syntax. Delimited with Sequence Delimitation Item if of Undefined Length.
2 bytes	2 bytes	4 bytes	'Value Length' bytes or Undefined Length

Tabela 5.3: Elementos de dados com VR implícito

5.2.4 Tamanho de Grupo - Group Length

O tamanho de grupo (gggg,0000) é um elemento de dados opcional (Tipo 3) e representa o tamanho de cada grupo de elemento de dados (em *bytes*). Ele fornece um esquema de otimização para as implementações, permitindo "pular" grupos sem interesse. Ele deve ser codificado usando VR igual a UL e VM igual a 1. Implementações pode ou não incluí-lo em seu conjunto de dados. Entretanto, todas as implementações devem ser capaz de aceitá-lo, mesmo que ignorando-o em seguida.

5.2.5 Ordenação dos *bytes*

Um outro ponto importante a considerar na codificação de um arquivo DICOM é a ordenação dos bytes. Esta ordenação pode ser feita de duas formas: utilizando ordenação

little endian ou *big endian*.

Na forma *little endian* o *byte* menos significativo deve ser codificado primeiro, com os *bytes* restantes codificados em ordem crescente de significância. Na forma *big endian*, o *byte* mais significativo deve ser codificado primeiro, com os *bytes* restantes codificados em ordem decrescente de significância.

Em um *string* de caracteres, onde um caracterer corresponde a um *byte*, o tipo de ordenação não afeta a codificação. Os caracteres devem ser codificados na ordem de ocorrência no *string* (da esquerda para a direita).

O ordenamento dos bytes é um componente que é definido pela sintaxe de transferência. A sintaxe de transferência que deve ser suportada por todas as entidades de aplicação, usa *little endian* como padrão e é especificada no anexo A do PS 3.5.

5.2.6 Elemento de Dados como seqüência

O VR identificado como *Sequence* (SQ) deve ser utilizado em elementos de dados que possuam valor consistindo em zero ou mais itens, onde cada item possui seu próprio conjunto de dados. Seqüências fornecem um esquema de codificação flexível, o qual deve ser utilizado para estruturas simples onde há repetição de conjunto de dados, ou IODs mais complexos chamados *folders*. Elementos de dados como seqüências ainda podem ser usados recursivamente para estruturas em multi-nível.

Os itens presentes em um elemento de dados como seqüência devem ser um conjunto ordenado, onde cada item deve ser referenciado pela sua posição ordinal. Cada item implicitamente deve ter sua posição, começando de 1 para o primeiro item da seqüência, e incrementando 1 para cada item subsequente. Logo, o último item na seqüência deve ter sua posição igual ao número de itens da seqüência.

Os itens ainda, podem ou não ter um conjunto de elementos de dados. Os elementos de dados como seqüência podem conter múltiplos itens, mas sempre devem ter um VM igual 1.

Existem três elementos de dados especiais relacionados às seqüências, que não estão sujeitos as regras de codificação impostas pela sintaxe de transferência. Estes ele-

mentos sempre devem ser codificados usando VR implícito. São eles *Item* (ffe,e000), *Item Delimitation* (ffe,e00d), e *Sequence Delimitation Item* (ffe,e0dd). Entretanto, cada conjunto de dados contido no campo de valor de um item, deve ser codificado normalmente, seguindo as regras de codificação impostas pela sintaxe de transferência.

5.2.6.1 Regras para a codificação de um item

Cada item de uma seqüência deve ser codificado normalmente, como um elemento de dados, usando a etiqueta específica (ffe,e000). A etiqueta é seguida por 4 bytes do campo tamanho de valor, o qual pode ser codificado de duas maneiras:

- **Tamanho explícito:** O número de *bytes* contidos no campo de valor do item deve ser codificados como um inteiro não sinalizado de 32 *bits*. Este tamanho deve incluir o tamanho total do conjunto de dados. Se o item não tiver dados, seu tamanho deverá conter o valor 0x00000000.
- **Tamanho implícito:** O campo tamanho do valor deve conter o valor 0xffffffff para indicar tamanho indefinido. Isto deve ser utilizado em conjunto com o elemento de dado de delimitação de item (*Item Delimitation*). Este deve ser incluído após o último elemento de dados do conjunto de dados contido dentro do item. Nenhum valor deve ser colocado em seu campo de valor, e seu o campo tamanho de valor deve ser codificado como 0x00000000.

Dentro do contexto de cada item, os elementos de dados contidos no conjunto de dados devem ser ordenados na ordem crescente de etiqueta e aparecer somente uma vez. Um ou mais elementos de dados em um item podem ter VR igual a SQ, gerando assim o que é chamado de recursão.

Data Element Tag	Data Element Length	Data Element Value								
(gggg, eeee) with VR of SQ	0000F00H	First Item			Second Item			Third Item		
		Item Tag (FFFE, E000)	Item Length 0000 04F8H	Item Value Data Set	Item Tag (FFFE, E000)	Item Length 0000 04F8H	Item Value Data Set	Item Tag (FFFE, E000)	Item Length 0000 04F8H	Item Value Data Set
4 bytes	4 bytes	4 bytes	4 bytes	04F8H bytes	4 bytes	4 bytes	04F8H bytes	4 bytes	4 bytes	04F8H bytes

Tabela 5.4: Sequência com três itens de tamanho explícito

Data Element Tag	Value Representation		Data Element Length	Data Element Value							
(gggg, eeee) with VR of SQ	SQ	0000H Reserved	FFFF FFFFH un-defined length	First Item			Second Item			Sequence Delimitation Item	
				Item Tag (FFFE, E000)	Item Length 98A5 2C68H	Item Value Data Set	Item Tag (FFFE, E000)	Item Length B321 762CH	Item Value Data Set	Seq. Delim. Tag (FFFE, E0DD)	Item Length 0000 0000H
4 bytes	2 bytes	2 bytes	4 bytes	4 bytes	4 bytes	98A5 2C68H bytes	4 bytes	4 bytes	B321 762CH bytes	4 bytes	4 bytes

Tabela 5.5: Sequência de tamanho indefinido, com dois itens de tamanho explícito

Data Element Tag	Data Element Length	Data Element Value									
(gggg, eeee) with VR of SQ	FFFF FFFFH un-defined length	First Item			Second Item					Sequence Delimitation Item	
		Item Tag	Item Length	Item Value	Item Tag	Item Length	Item Value	Item Delim. Tag	Length	Seq. Delim. Tag	Item Length
		(FFFE, E000	0000 17B6H	Data Set	(FFFE, E000)	FFFF FFFFH un-defined length	Data Set	(FFFE, E00D)	0000 0000H	(FFFE, E0DD)	0000 0000H
4 bytes	4 bytes	4 bytes	4 bytes	17B6H bytes	4 bytes	4 bytes	un-defined length	4 bytes	4 bytes	4 bytes	4 bytes

Tabela 5.6: Sequência com tamanho indefinido, com um item de tamanho explícito e um de tamanho indefinido

Capítulo 6

A entidade de informação DICOM

Waveform

A entidade de informação DICOM *Waveform* é a entidade DICOM para representação de exames no formato de sinais, e é composta por várias classes de objetos de informação, os quais representam ondas específicas como eletrocardiograma, sinais de equipamentos de hemodinâmica e áudio.

O modelo Entidade-Relacionamento (E-R) da entidade de informação *Waveform* é aplicado a todos os IODs *Waveform*, e é apresentado na Figura 6.1.

De acordo com o PS 3.3 do padrão, pode-se observar que todos os IODs possuem uma tabela com módulos (IOM) que representam sua constituição. Todos os IODs *Waveform* possuem a mesma tabela de módulos, mudando somente como são preenchidos alguns destes módulos. Esta tabela é ilustrada na Tabela 6.1.

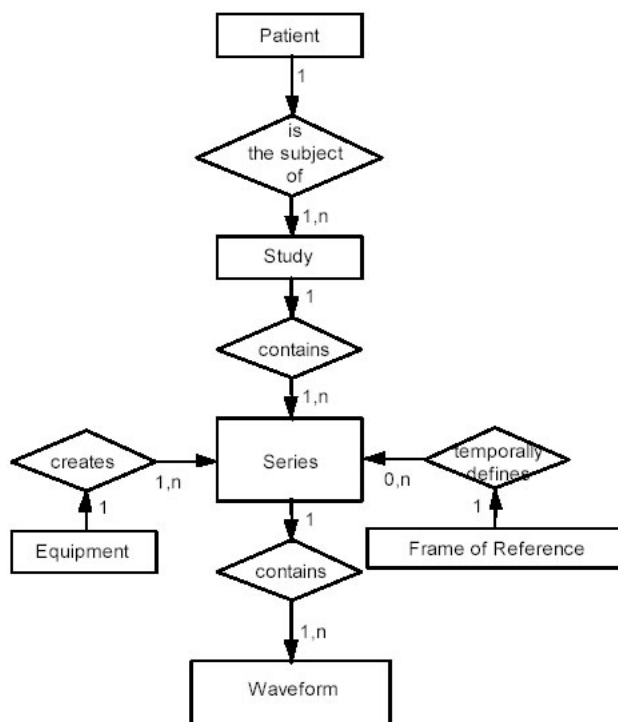


Figura 6.1: Modelo de informação DICOM *Waveform* IOD

IE	Module	Reference	Usage
Patient	Patient	C.7.1.1	M
	Clinical Trial Subject	C.7.1.3	U
Study	General Study	C.7.2.1	M
	Patient Study	C.7.2.2	U
	Clinical Trial Study	C.7.2.3	U
Series	General Series	C.7.3.1	M
	Clinical Trial Series	C.7.3.2	U
Frame of Reference	Synchronization	C.7.4.2	U
Equipment	General Equipment	C.7.5.1	M
Waveform	Waveform Identification	C.10.8	M
	Waveform	C.10.9	M
	Acquisition Context	C.7.6.14	M
	Waveform Annotation	C.10.10	C – required if annotation is present
	SOP Common	C.12.1	M

Tabela 6.1: Tabela de módulos de um *Waveform* IOD

6.1 Objetos de informação *Waveform*

A seguir será apresentado uma breve descrição de todos os IODs que constituem a entidade de informação *Waveform*. Informações mais detalhadas, incluindo parâmetros para codificação, podem ser encontrados no anexo A seção 34 do PS 3.3 do padrão.

6.1.1 *Basic Voice Audio Information Object Definition*

Este IOD é representado pela modalidade AU, e é a especificação de som digitalizado, que foi adquirido ou criado por uma modalidade de áudio ou por uma função de aquisição de áudio, contida dentro de uma modalidade de imagem. O uso típico é a aquisição de áudio do ultrassom.

6.1.2 *12-Lead Electrocardiogram Information Object Definition*

Este IOD é representado pela modalidade ECG, e é a especificação de sinais elétricos digitalizados do sistema de condução cardíaco do paciente, coletado na superfície do corpo, adquirido por uma modalidade ECG ou por uma função de aquisição ECG, contida dentro de uma modalidade de imagem. Este IOD deve ter um número de canais *waveform* entre 1 e 13.

6.1.3 *General Electrocardiogram Information Object Definition*

Apresenta as mesmas características do *12-Lead Electrocardiogram Information Object Definition*, a não ser pelo fato de ser mais abrangente, e poder ter números de canais *waveform* entre 1 e 24.

6.1.4 *Ambulatory Electrocardiogram Information Object Definition*

Este IOD é representado pela modalidade ECG, e é a especificação de sinais elétricos digitalizados do sistema de condução cardíaco do paciente, coletado na superfície do corpo, adquirido por um dispositivo de eletrocardiografia ambulatorial (*Holter*).

6.1.5 *Hemodynamic Information Object Definition*

Este IOD é representado pela modalidade HD, e é a especificação de medidas de pressão digitalizados e outros sinais elétricos do sistema circulatório do paciente, adquirido por uma modalidade de hemodinâmica.

6.1.6 *Basic Cardiac Electrophysiology Information Object Definition*

Este IOD é representado pela modalidade EPS, e é a especificação de sinais elétricos digitalizados do sistema de condução cardíaco do paciente, coletado no coração, adquirido por um modalidade de eletrofisiologia.

6.2 Principais características entre as modalidades

Esta seção apresenta uma série de tabelas, onde é possível identificar as principais diferenças e características de cada modalidade que compõem os IODs que constituem a entidade de informação *Waveform*.

Descrição da modalidade	Modalidade
<i>Basic Voice Audio IOD</i>	AU
<i>12-Lead Electrocardiogram IOD</i>	ECG
<i>General Electrocardiogram IOD</i>	ECG
<i>Ambulatory Electrocardiogram IOD</i>	ECG
<i>Hemodynamic IOD</i>	HD
<i>Basic Cardiac Electrophysiology IOD</i>	EPS

Tabela 6.2: Modalidades *Waveform*

Descrição da modalidade	Seqüência de Waveform
<i>Basic Voice Audio IOD</i>	1
<i>12-Lead Electrocardiogram IOD</i>	1-5
<i>General Electrocardiogram IOD</i>	1-4
<i>Ambulatory Electrocardiogram IOD</i>	1
<i>Hemodynamic IOD</i>	1-4
<i>Basic Cardiac Electrophysiology IOD</i>	1-4

Tabela 6.3: Quantidade de seqüências de *Waveform* por modalidades

Descrição da modalidade	Número de canais
<i>Basic Voice Audio IOD</i>	1 ou 2
<i>12-Lead Electrocardiogram IOD</i>	1-13
<i>General Electrocardiogram IOD</i>	1-24
<i>Ambulatory Electrocardiogram IOD</i>	1-12
<i>Hemodynamic IOD</i>	1-8
<i>Basic Cardiac Electrophysiology IOD</i>	1

Tabela 6.4: Quantidade de canais por modalidade

Descrição da modalidade	Frequência de amostragem
<i>Basic Voice Audio IOD</i>	8000 Hz
<i>12-Lead Electrocardiogram IOD</i>	200-1000 Hz
<i>General Electrocardiogram IOD</i>	200-1000 Hz
<i>Ambulatory Electrocardiogram IOD</i>	50-1000 Hz
<i>Hemodynamic IOD</i>	\dot{z} = 400 Hz
<i>Basic Cardiac Electrophysiology IOD</i>	\dot{z} = 2000 Hz

Tabela 6.5: Frequência de amostragem por modalidade

Capítulo 7

Estado da arte

Pelo caráter particular deste projeto, não se sabe da existência de nenhum outro software que se proponha a fazer algo de forma semelhante. Além disto, mesmo na área de DICOM Waveform, existe muito poucos softwares para visualização e análise de sinais. A grande maioria deles vem em soluções que são vendidos junto com os equipamentos do fabricante. O software não é vendido separadamente, e geralmente esta solução apresenta um alto custo, dificultando o acesso por parte de pequenas clínicas e hospitais públicos. A seguir, apresentaremos algumas destas aplicações, as quais abrangem o assunto DICOM Waveform.

7.1 Dicom Waveform Viewer

O Dicom Waveform Viewer é um visualizador para exames de eletrocardiograma e hemodinâmica. Ele foi desenvolvido pelo Instituto do Coração do Hospital das Clínicas em São Paulo (Incor) em parceria com a empresa americana Excel Medical System.

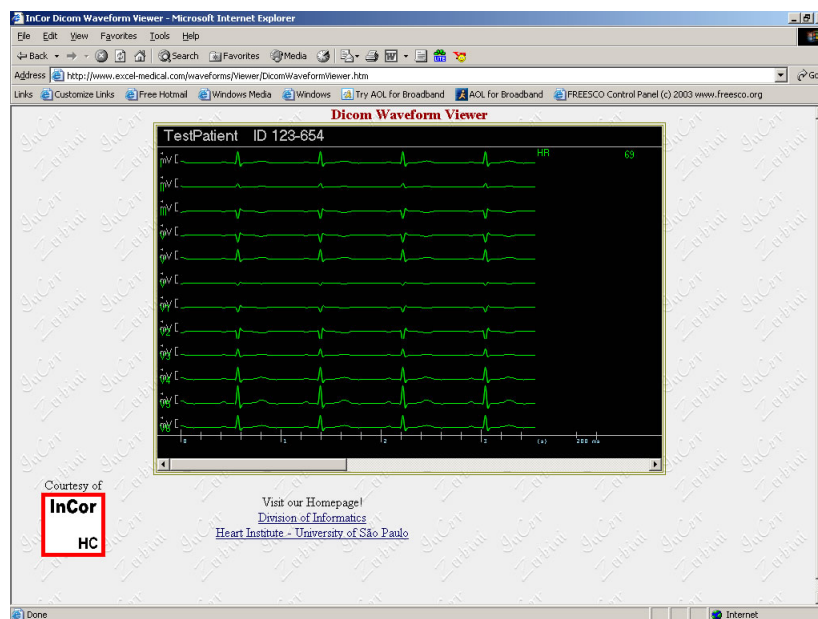


Figura 7.1: Dicom Waveform Viewer

7.2 Cyclops Waveform Viewer

Esta aplicação foi desenvolvida na Universidade Federal de Santa Catarina, no contexto do projeto Cyclops, para suprir a necessidade deste na área de sinais biológicos. O Cyclops Waveform Viewer é um software que permite a visualização, impressão e análise de sinais de eletrocardiograma.

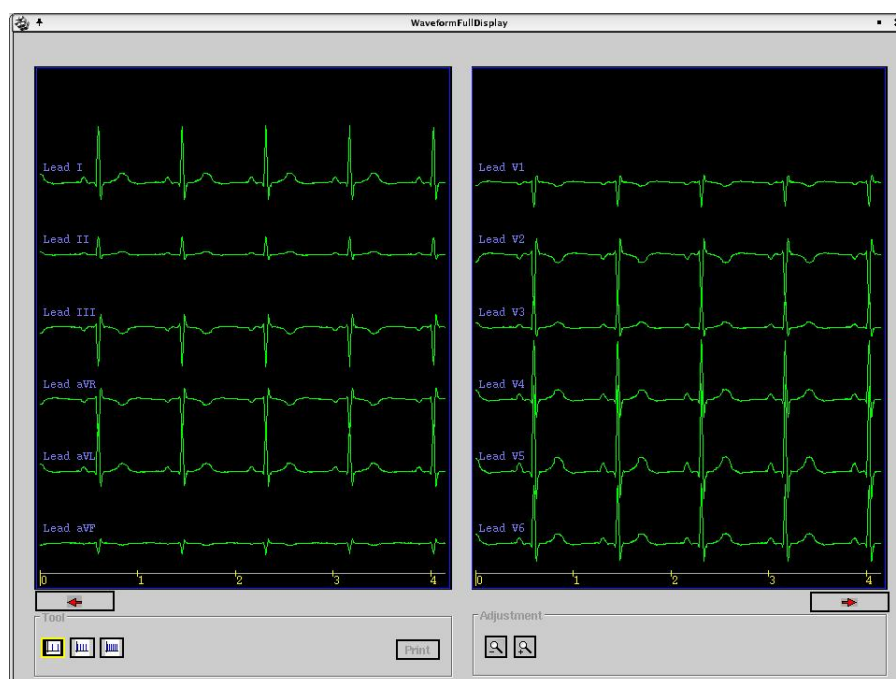


Figura 7.2: Cyclops *Waveform Viewer*

7.3 Quinton Q-TRACK II Holter System

Esta aplicação utiliza tecnologia desenvolvida pela Mortara Instruments, Inc. O software é vendido junto com o equipamento do fabricante. Dentre as características do software, estão análise e detecção de frequência cardíaca, análise e detecção de fibrilação arterial, impressão, exportação de dados no formato Adobe Acrobat PDF, e um avançado processamento de sinal.

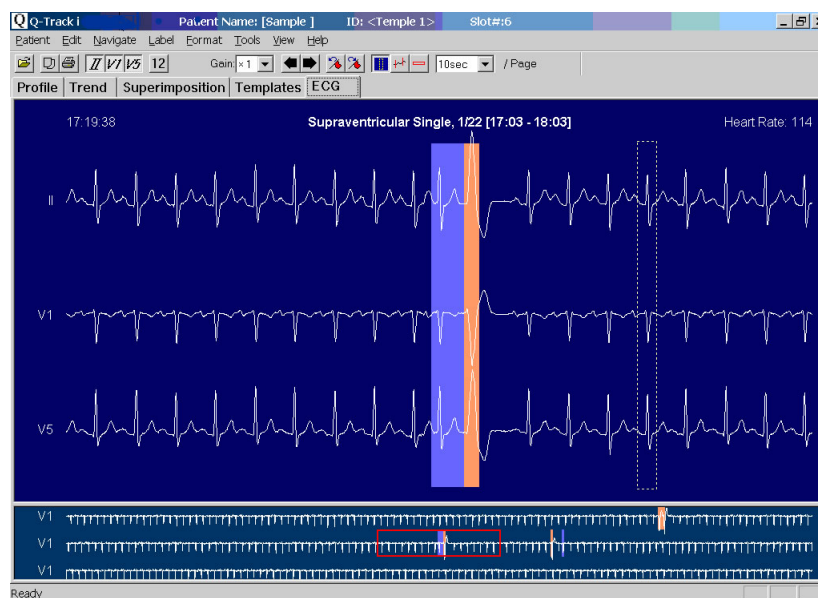


Figura 7.3: Quinton Q-TRACK II *Holter System*

Capítulo 8

Tecnologias utilizadas

Este capítulo destina-se a descrever as ferramentas e tecnologias utilizadas para o desenvolvimento do projeto. Foi feito um estudo sobre cada uma das tecnologias e ferramentas empregadas, procurando encontrar no histórico, vantagens e desvantagens, comparativos com outras tecnologias da mesma linha de funcionamento e o estado da arte, o qual representa o atual estado em que se encontra cada tecnologia.

8.1 *Extensible Markup Language - XML*

O XML é uma linguagem de marcação baseada em texto, a qual vem se tornando um padrão para intercâmbio de dados. Foi desenvolvido para suportar uma grande variedade de aplicações, mantendo compatibilidade com SGML. Uma descrição mais detalhada sobre esta tecnologia é apresentado no capítulo 2.

8.2 **Java**

Java é a linguagem que mais cresce em termos de utilização por desenvolvedores em aplicações hoje em dia. Sua sintaxe parece muito com o tradicional C/C++, possibilitando uma fácil aprendizagem por quem já conhece C. Java é orientado objetos, possui um gigantesco *framework* e componentes desenvolvidos nas mais diversas áreas da computação. Java tem a vantagem de ser uma linguagem de programação sem custo para

o desenvolvedor. Para sua utilização, basta efetuar o download do SDK (*Software Development Kit*) no site da Sun Microsystems.

Abaixo são descritos os principais pacotes java utilizados neste trabalho.

8.2.1 *Java Document Object Model - JDOM*

Se você já utilizou o DOM (*Document Object Model*) ou o SAX (*Simple API for XML*) ambos do pacote JAXP (*Java API for XML Processing*), com certeza já se deparou com a complexidade deles. Diante disto, Jason Hunter e Brett McLaughlin criaram uma API para ser uma versão *Java-friendly* do DOM/SAX a qual foi chamada de JDOM.

O JDOM não é um parser, como o Xerces ou o JAXP. Ele é uma API que facilita a criação e atualização de documentos XML. Por definição ele é configurado para utilizar o JAXP mais pode ser configurado para usar a maioria dos parsers existentes.

8.3 *Digital Imaging and Communications in Medicine - DICOM*

O DICOM é um protocolo para a troca de informações médicas, tanto entre equipamentos como por profissionais de saúde. Uma descrição mais detalhada sobre esta tecnologia é apresentado no capítulo 4.

8.4 Eclipse

O Eclipse é uma plataforma para edições de código Java. Ele possui uma interface amigável, além de ser leve, rápido e sem custo para o desenvolvedor. O Eclipse é todo customizável, possui uma enorme quantidade de *plugins* para adicionar inúmeras funcionalidades ao mesmo. Isso torna mais fácil a criação, integração e utilização das ferramentas, economizando tempo e dinheiro. A plataforma Eclipse tem sido desdobrada em uma grande escala de desenvolvimento de estações de trabalho como HP-UX, Solaris, AIX e Linux (ECLIPSE RIO, 2004).

Segundo o projeto Eclipse Rio (2004), a plataforma Eclipse é uma proposta de consórcio de empresas que apóiam o uso de uma arquitetura aberta para a criação de ambientes integrados de desenvolvimento (IDEs), onde a indústria de software possa desenvolver diversos programas, aplicativos e ferramentas, de forma otimizada e padronizada, baseando-se nas iniciativas de software livre. Uma das empresas envolvidas no projeto da plataforma Eclipse é a conhecida IBM.

8.5 Concurrent Versions System - CVS

O CVS (*Concurrent Versions System*) permite que se organizem grupos de trabalho para desenvolvimento de projetos colaborativos. Um projeto pode ser um programa em qualquer linguagem de programação, um documento, enfim, algo que se precise compartilhar e ter um controle de versão.

O uso do CVS é recomendado para qualquer desenvolvimento de projeto, mesmo aqueles em que existe vários envolvidos trabalhando ao mesmo tempo, até aqueles em que existe apenas um desenvolvedor.

Imagine uma situação onde você está desenvolvendo um programa de computador e após a última modificação ele pare de funcionar. Com o CVS é possível ver o que foi modificado e voltar até a versão que estava funcionando para consertar o problema. No desenvolvimento de uma documentação, o CVS é igualmente interessante, pois possibilita visualizar o que foi modificado entre uma versão anterior do documento e a versão mais recente, para posterior análise e ajustes.

Foi justamente por isto, que no desenvolvimento deste trabalho, optou-se por usar o CVS.

Capítulo 9

Sistema proposto

O sistema proposto foi desenvolvido com o objetivo de suprir uma necessidade da empresa DIXTAL, a qual apresentava uma carência em armazenamento de sinais em conformidade com o padrão DICOM. Para a solucionar o problema, foi feita a implementação de uma aplicação em linguagem JAVA, que converte um exame de ECG armazenado em formato XML para o formato DICOM. Diante deste contexto, o projeto desenvolveu-se em três partes principais: modelagem do arquivo XML, modelagem orientada a objetos (OO) do IOD *Waveform* e implementação do sistema.

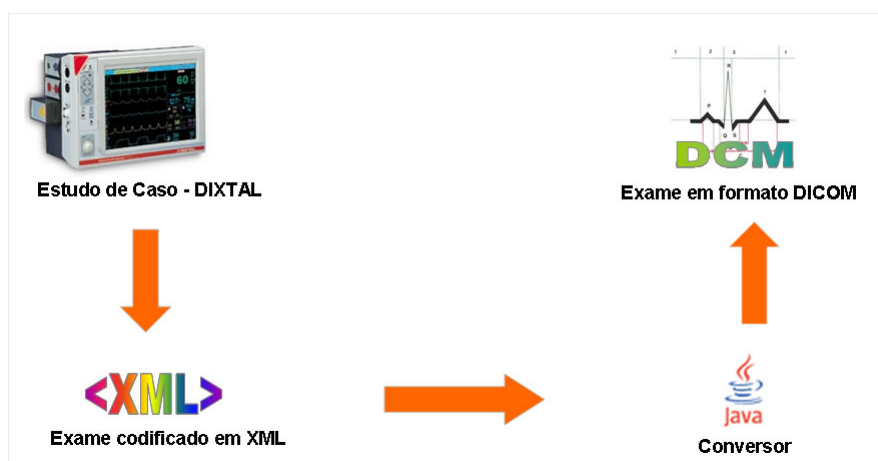


Figura 9.1: O sistema proposto

9.1 O arquivo XML

Um dos fatores mais importantes para o funcionamento do sistema proposto é o arquivo XML, por ser a interface de comunicação entre o sistema e o sistema DX-25 da DIXTAL.

O arquivo XML armazenado pelo sistema DX-25, deverá conter informações sobre o paciente, parâmetros de captura do eletrocardiograma, além dos traçados fornecidos pelo exame.

A estratégia utilizada foi modelar o arquivo XML seguindo o modelo de um *Waveform IOD*. O objeto de informação que utilizamos foi o *General ECG IOD* (Capítulo 6), por possuir uma abrangência maior em relação ao número de canais.

Para validação da estrutura e dos elementos do documento XML foi criado um *XML Schema*. O modelo do arquivo XML poder ser encontrado no Anexo 1 e o *XML Schema* no Anexo 2.

9.2 Modelagem OO do IOD Waveform

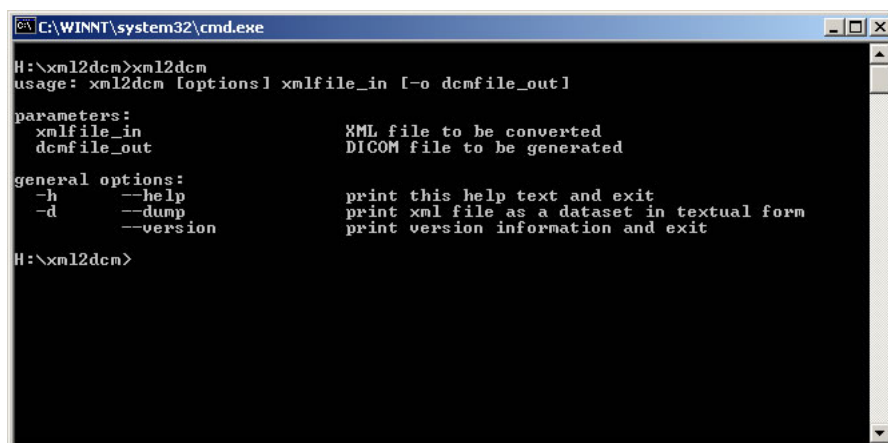
Conforme abordado no Capítulo 6, um *IOD Waveform* é formado por uma tabela de módulos que representam a constituição do seu objeto de informação. Dentre estes módulos, existem aqueles que são tipo M (*Mandatory Modules*), que são aqueles que são obrigatórios para a implementação do IOD. Por questões óbvias, na modelagem das classes só foi utilizado módulos tipo M.

Durante o processo de modelagem, tentou-se seguir de maneira mais fiel possível o modelo de informação DICOM. O resultado disto é um conjunto formado por 52 classes, contidas em 10 pacotes.

9.3 Implementação do Sistema

O sistema se trata de uma simples aplicação modo texto, que é o resultado da implementação do modelo OO gerado na segunda fase do projeto. Para executar a aplicação,

basta entrar com seu nome na linha de comando, seguido do nome do arquivo XML. O *parser* XML irá ler o arquivo XML e efetuar a validação do documento XML. Feito isto, será instanciado um objeto da classe *GeneralECGXMLReader* que irá ler os valores dos elementos do documento XML e preencher o objeto *GeneralECGIOD*. Após isto, uma instância da classe *DicomOutputStream* irá gerar um arquivo com extensão DCM, que é a representação do exame no padrão DICOM.



```
C:\WINNT\system32\cmd.exe
H:\xml2dcm>xml2dcm
usage: xml2dcm [options] xmlfile_in [-o dcmfile_out]

parameters:
  xmlfile_in      XML file to be converted
  dcmfile_out     DICOM file to be generated

general options:
  -h             --help      print this help text and exit
  -d             --dump     print xml file as a dataset in textual form
  --version     print version information and exit

H:\xml2dcm>
```

Figura 9.2: Janela da aplicação

9.3.1 Identificação

Um aspecto importante no desenvolvimento da aplicação foi conseguir uma raiz perante uma autoridade de padronização. Isto foi visto com mais detalhes no item 4.3.6. Esta raiz é utilizada para representar a empresa, e assim poder gerar os identificadores únicos (UID). A entidade de padronização escolhida foi a *Internet Assigned Numbers Authority* (IANA) e raiz fornecida foi 1.3.6.1.4.1.19491. Dessa forma, foi possível gerar o identificador para a instância SOP sem problemas, além de outros identificadores.

9.3.2 Dump

Um outro aspecto implementado na aplicação foi a capacidade de representação de um possível arquivo DICOM de forma textual. Isto é feito de forma muito similar ao conhecido aplicativo *dcmdump* do instituto alemão OFFIS. Visualizando um arquivo

DICOM de forma textual, fica muito mais fácil para identificar possíveis falhas no preenchimento adequado do arquivo XML.

```

(0008,0090) PN [FORD MDAGREGORY] # 16, 1 ReferringPhysiciansName
(0008,1010) SH [INVASIVE CARDIOL] # 16, 1 StationName
(0008,1050) PN [CANSLER MDALISA] # 16, 1 PerformingPhysiciansName
(0008,1070) PN [A] # 2, 1 operatorsName
(0008,1090) LO [MACLB] # 6, 1 ManufacturerModelName
(0010,0010) PN [Patient^Test] # 12, 1 PatientName
(0010,0020) LO [123-654] # 8, 1 PatientID
(0010,0030) DA [19440102] # 8, 1 PatientBirthDate
(0010,0040) CS [M] # 2, 1 PatientSex
(0010,1010) AS [055Y] # 4, 1 PatientAge
(0010,1020) DS [1.77] # 4, 1 PatientSize
(0010,1030) DS [90] # 2, 1 PatientWeight
(0018,1020) LO [17A] # 6, 1 Softwareversion
(0018,106a) CS [NO TRIGGER] # 10, 1 SynchronizationTrigger
(0018,1800) CS [N] # 2, 1 AcquisitionTimeSynchroniz
(0020,000d) UI [1.3.6.1.4.1.6018.4.999] # 22, 1 StudyInstanceUID
(0020,000e) UI [1.3.6.1.4.1.6018.5.999] # 22, 1 SeriesInstanceUID
(0020,0010) SH [43288] # 6, 1 StudyID
(0020,0011) IS [0001] # 4, 1 SeriesNumber
(0020,0013) IS [0001] # 4, 1 ImageNumber
(0020,0200) UI [1.3.6.1.4.1.6018.6.999] # 22, 1 SynchronizationFrameofRefer
(0040,0555) SQ (Sequence undefined length #=1) # u/1, 1 AcquisitionContextSequence
(fffe,e000) na (Item with undefined length #=2) # u/1, 1 Item
(0040,a043) SQ (Sequence undefined length #=2) # u/1, 1 ConceptNameCodeSequence
(fffe,e000) na (Item with undefined length #=4) # u/1, 1 Item
(0008,0100) SH [109057] # 6, 1 Codevalue
(0008,0102) SH [DCM] # 4, 1 CodingschemeDesigna
(0008,0103) SH [01] # 2, 1 Codingschemeversion
(0008,0104) LO [Catheterization Procedure Phase] # 32, 1 CodeMeaning
(fffe,e00d) na (ItemDelimitationItem) # 0, 0 ItemDelimitationItem
(fffe,e0dd) na (SequenceDelimitationItem) # 0, 0 SequenceDelimitationItem
(0040,a168) SQ (Sequence undefined length #=1) # u/1, 1 ConceptCodeSequence
(fffe,e000) na (Item with undefined length #=4) # u/1, 1 Item
(0008,0100) SH [G-7293] # 6, 1 Codevalue
(0008,0102) SH [SRT] # 4, 1 CodingschemeDesigna

```

Figura 9.3: Arquivo DICOM em representação textual

9.3.3 Validação

Para verificar se o arquivo gerado atendia corretamente à especificação do padrão DICOM, optou-se por fazer testes com duas aplicações bem conhecidas e respeitadas: o DCMTK - DICOM *Toolkit* do instituto alemão OFFIS e o PixelMed Java DICOM *Toolkit* criado pelo respeitado David Clunie, uma das pessoas que ajuda a desenvolver o padrão.

- **DCMTK - DICOM *Toolkit*:** Dentro desse conjunto de ferramentas, foi utilizado para realizar o teste de conformidade da estrutura do documento DICOM a aplicação *dcmdump*. Se algum elemento de dados estiver sido gravado incorretamente, é possível identificar através deste aplicativo. A Figura 9.4 ilustra a saída do software, mostrando a árvore dos elementos de dados codificados sem nenhum erro.

```

C:\WINNT\system32\cmd.exe
<003a,021a> US 16 # 2, 1 Waveform
mBitsStored
<003a,0220> DS [ .05 ] # 4, 1 FilterL
owFrequency
<003a,0221> DS [100] # 4, 1 FilterH
ighFrequency
<fffe,e00d> na <ItemDelimitationItem> # 0, 0 ItemDelimitationItem
<fffe,e0dd> na <SequenceDelimitationItem> # 0, 0 SequenceDelimitationItem
<5400,1004> US 16 # 2, 1 WaveformBitsAllocated
<5400,1006> CS [SS] # 2, 1 WaveformSampleInterpretation
<5400,100a> OW 8000 # 2, 1 WaveformPaddingValue
<5400,1010> OW 00ba0030ff76ff8b00a2ffd3ffaeff50006200c4011e00c200ba... # 57600, 1 WaveformData
<fffe,e00d> na <ItemDelimitationItem> # 0, 0 ItemDelimitationItem
<fffe,e0dd> na <SequenceDelimitationItem> # 0, 0 SequenceDelimitationItem
H:\dcm>pause
Press any key to continue . . .

```

Figura 9.4: DICOM Toolkit - dcmdump

- PixelMed Java DICOM Toolkit:** Dentro desse conjunto de ferramentas, foi utilizado o *ECGViewer*, uma aplicação que permite a visualização de exames de ECG. Através deste aplicativo é possível verificar se as coordenadas dos pontos do plano cartesiano tinham sido armazenadas corretamente, através da plotagem dos sinais do ECG. A Figura 9.5 ilustra um exame de ECG de 12 canais.

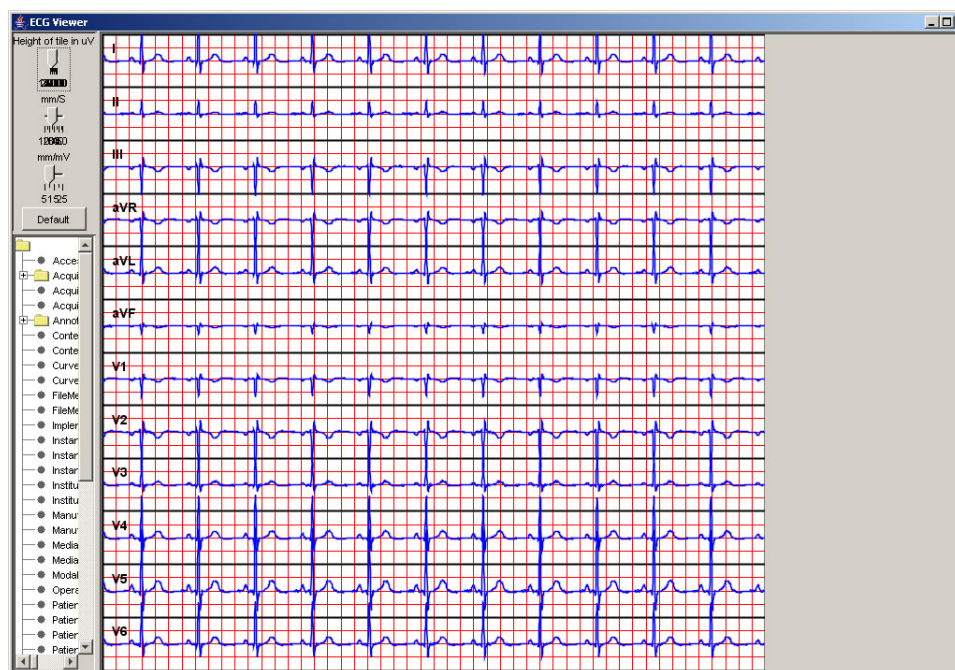


Figura 9.5: PixelMed Java DICOM Toolkit - ECGViewer

Capítulo 10

Considerações finais

Através da implantação do software, é possível mensurar que o objetivo do trabalho tenha sido alcançado, pois o software apresentou os resultados almejados, conseguindo gerar de forma eficiente os arquivos em conformidade com o padrão DICOM.

A validação obtida foi realizada através de testes com outras aplicações compatíveis com o padrão e com a utilização do software em ambientes como clínicas e hospitais, possibilitando a conclusão do projeto de acordo com os objetivos propostos.

Apesar do foco do trabalho ter sido a conversão de exames da empresa DIXTAL, seu campo de aplicações é vasto e pode ser utilizado para conversão de sinais de qualquer empresa, desde que esta armazene os exames no formato XML, seguindo o modelo de documento utilizado no trabalho.

Além disso, a flexibilidade da ferramenta desenvolvida permite que pequenas adaptações e adição de novos módulos ampliem mais seu campo de atuação, podendo efetuar conversões para outras modalidades DICOM.

Com relação às contribuições à DIXTAL, o sistema se mostra uma ótima solução, pois possibilita a conversão de exames em equipamentos já existentes. Utilizando DICOM ao invés do padrão proprietário que era utilizado, a empresa consegue atingir um número maior de clientes e se coloca no mesmo patamar de grandes empresas como SIEMENS, GE entre outras.

Existem grandes potenciais em utilizar o padrão DICOM no contexto de clínicas e

hospitais. Isto é verificado claramente em um ambiente PACS (*Picture Archiving and Communications System*). De qualquer forma, estes potenciais devem ser explorados, mesclando-se outras tecnologias, para se obter ferramentas e soluções interessantes e criativas, que venham a contribuir com a área de informática médica.

Capítulo 11

Trabalhos futuros

O sistema desenvolvido apresenta um grande número de classes que podem ser utilizados para desenvolver facilmente uma série de novas aplicações. Podemos citar algumas idéias:

- **Desenvolvimento deste sistema em hardware:** Seria bem mais interessante desenvolver esta aplicação no próprio equipamento, onde já fizesse o armazenamento dos sinais no padrão DICOM. Assim, não precisaríamos ter uma aplicação externa nem um arquivo XML usado como interface de comunicação.
- **Conversão de imagens de ultrassom antigos para o padrão DICOM:** Este sistema faria a captura das imagens de ultrassom através de uma placa de captura RGB, e realizaria a conversão destas imagens para o padrão DICOM. Muitas clínicas ainda possuem equipamentos de ultrassom antigos, e um software como este poderia ajudar estas clínicas a utilizar este equipamento em um ambiente PACS sem precisar de adquirir um equipamento mais novo.
- **Sistema de plotagem e análise de sinais:** Este sistema efetuaria a leitura de sinais armazenados no padrão DICOM e faria plotagem desses sinais em uma interface de janela. Ele poderia efetuar uma análise dos traçados em buscas de patologias que alteram os mesmos. Podemos citar como exemplo a Hipertrofia Ventricular Esquerda (HVE), a qual apresenta aumento dos potenciais do ventrículo esquerdo.

- **Visualizador de imagens DICOM:** Uma grande parte dos visualizadores de imagens DICOM atuais, não implementa todos os tipos de modalidades e sintaxe de transferência existentes no padrão. Uma idéia seria desenvolver um que fosse mais completo.

Referências Bibliográficas

ALMEIDA, M. B. *Uma introdução ao XML, sua utilização na Internet e alguns conceitos complementares*. Dissertação (Mestrado em Ciências da Informação) — Universidade Federal de Minas Gerais, Minas Gerais, 2002.

ARAÚJO, G. A. de. *Desenvolvimento de um cliente PACS DICOM 3.0 - compatível para consulta análise e laudos de eletrocardiografia digital*. Dissertação (Mestrado em Ciências da Computação) — Universidade Federal de Santa Catarina, Florianópolis, 2002.

BIBLIOMED. *O coração: sua anatomia e funções*. Disponível em:
<<http://boasaude.uol.com.br/Lib/ShowDoc.cfm?LibDocID=4165&ReturnCatID=357>>.
Acesso em: Maio, 2004.

BORTOLUZZI, M. K. *Desenvolvimento e implementação de um editor de documentos estruturados no padrão DICOM Structured Report*. Dissertação (Mestrado em Ciências da Computação) — Universidade Federal de Santa Catarina, Florianópolis, 2003.

CARDOSO, S. H. *O sistema circulatório*. Disponível em:
<<http://www.nib.unicamp.br/svol/corac.htm>>. Acesso em: Maio, 2004.

CARVALHO, J. H. *DICOM*. Coimbra, 1998.

CLUNIE, D. A. *DICOM Standard Status*. Maio 2004. Disponível em:
<<http://www.dclunie.com/dicom-status/status.html>>. Acesso em: Maio, 2004.

DELLANI, P. R. *Desenvolvimento de um servidor de imagens médicas digitais no padrão DICOM*. Dissertação (Mestrado em Ciências da Computação) — Universidade Federal de Santa Catarina, Florianópolis, 2001.

DUBIN, D. *Interpretação rápida do ECG*. 3. ed. Rio de Janeiro: Editora de Publicações Científicas, 1996.

ECLIPSE RIO. *O que é o Eclipse?* Maio 2004. Disponível em:
<<http://web.teccomm.les.inf.puc-rio.br/eclipse>>. Acesso em: Maio, 2004.

EDWARDS, N. J. *Bolton's handbook of canine and feline electrocardiography*. 2. ed. Philadelphia: W.B. Saunders, 1987.

FUNDÃO. *Introdução ao XML*. Disponível em:
<<http://www2.fundao.pro.br/articles.asp?cod=165>>. Acesso em: Maio, 2004.

MOULTIS, N. P.; KIRK, C. *XML Black Book*. São Paulo: Makron Books, 2000.

NATIONAL ELECTRICAL MANUFACTURERS ASSOCIATION. *Part 3: Information object definitions*. Rosslyn, 2003.

NATIONAL ELECTRICAL MANUFACTURERS ASSOCIATION. *Part 3: Data structures and encoding*. Rosslyn, 2003.

NATIONAL ELECTRICAL MANUFACTURERS ASSOCIATION. *Part 3: Data dictionary*. Rosslyn, 2003.

NATIONAL ELECTRICAL MANUFACTURERS ASSOCIATION. *Part 3: Media storage and file format for data interchange*. Rosslyn, 2003.

O CORPO HUMANO. *Sistema circulatório*. Disponível em:
<<http://www.corpohumano.hpg.ig.com.br/circulacao/ciculacao.html>>. Acesso em: Maio, 2004.

SAMPAIO, S. C. *Modelagem e implementação orientada a objetos de um cliente de rede para banco de dados de imagens médicas digitais utilizando o padrão DICOM 3.0*.

Dissertação (Mestrado em Ciências da Computação) — Universidade Federal de Santa Catarina, Florianópolis, 1999.

SOCIEDADE BRASILEIRA DE CARDIOLOGIA. *O que é eletrocardiograma e para que serve?* Maio 1996. Disponível em: <<http://prevencao.cardiol.br/sbc-funcor/cuide/coracaonline/Responline.asp?P1=40&CodSes=2>>. Acesso em: Maio, 2004.

SUN MICROSYSTEMS. *Understanding XML*. Disponível em: <<http://java.sun.com/j2ee/1.4/docs/tutorial/doc/IntroXML.html>>. Acesso em: Maio, 2004.

TILLEY, L. P. *Essentials of canine and feline electrocardiography: interpretation and treatment*. 3. ed. Philadelphia: Lea & Febiger, 1992.

TOCCHIO, H. *Interpretação clínica do eletrocardiograma*. 1. ed. São Paulo: Atheneu, 1986.

TRANCHESI, J. *Eletrocardiograma normal e patológico*. São Paulo: Atheneu, 1972.

TÁRRAGA, K. M. *Eletrocardiografia - Breve histórico e descrição do método*. Disponível em: <<http://www.provet.com.br/informetecnico.html>>. Acesso em: Maio, 2004.

Anexo 1

modelo.xml

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <Modules xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3 xsi:noNamespaceSchemaLocation="dcm_template.xsd">
4   <Patient>
5     <FirstName/>
6     <LastName/>
7     <ID/>
8     <BirthDate/>
9     <Sex/>
10  </Patient>
11  <Study>
12    <GeneralStudy>
13      <Date/>
14      <Time/>
15      <ReferringPhysiciansName>
16        <FirstName/>
17        <LastName/>
18      </ReferringPhysiciansName>
19      <ID/>
20      <AccessionNumber/>
21    </GeneralStudy>
22    <PatientStudy>
23      <Age/>
24      <Size/>
25      <Weight/>
26    </PatientStudy>
27  </Study>
28  <Series>
29    <Modality/>
30    <SeriesNumber/>
31  </Series>
```

```
32 <Equipment>
33   <Manufacturer/>
34   <InstitutionName/>
35   <ManufacturersModelName/>
36 </Equipment>
37 <WaveformModules>
38   <WaveformIdentification>
39     <Date/>
40     <Number/>
41     <Time/>
42     <AcquisitionDatetime/>
43   </WaveformIdentification>
44   <AcquisitionContext>
45     <AcquisitionContextSequence>
46       <item>
47         <ConceptNameCodeSequence/>
48         <ConceptCodeSequence/>
49       </item>
50     </AcquisitionContextSequence>
51   </AcquisitionContext>
52   <WaveformAnnotation>
53     <WaveformAnnotationSequence>
54       <item>
55         <ConceptNameCodeSequence/>
56         <ReferencedWaveformChannels/>
57         <NumericValue/>
58       </item>
59     </WaveformAnnotationSequence>
60   </WaveformAnnotation>
61   <Waveform>
62     <WaveformSequence>
63       <item>
64         <ChannelDefinitionSequence/>
65         <WaveformOriginality/>
66         <NumberOfWaveformChannels/>
67         <SamplingFrequency/>
68         <WaveformPaddingValue/>
69         <WaveformData/>
70       </item>
71     </WaveformSequence>
72   </Waveform>
73 </WaveformModules>
74 </Modules>
75
```

Anexo 2

dcmtemplate.xsd

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <xs:schema elementFormDefault="qualified" attributeFormDefault="unqualified"
3 xmlns:xs="http://www.w3.org/2001/XMLSchema">
4   <xs:element name="Modules">
5     <xs:complexType>
6       <xs:sequence>
7         <xs:element name="Patient">
8           <xs:complexType>
9             <xs:complexContent>
10              <xs:extension base="dicom_name">
11                <xs:sequence>
12                  <xs:element name="ID" type="xs:token"/>
13                  <xs:element name="BirthDate" type="dicom_date"/>
14                  <xs:element name="Sex" type="dicom_sex"/>
15                </xs:sequence>
16              </xs:extension>
17            </xs:complexContent>
18          </xs:complexType>
19        </xs:element>
20        <xs:element name="Study">
21          <xs:complexType>
22            <xs:sequence>
23              <xs:element name="GeneralStudy">
24                <xs:complexType>
25                  <xs:sequence>
26                    <xs:element name="Date" type="dicom_date"/>
27                    <xs:element name="Time" type="dicom_time"/>
28                    <xs:element name="ReferringPhysiciansName" type="dicom_name"/>
29                    <xs:element name="ID" type="xs:token"/>
30                    <xs:element name="AccessionNumber" type="xs:token"/>
31                  </xs:sequence>
```

```
32     </xs:complexType>
33 </xs:element>
34 <xs:element name="PatientStudy" minOccurs="0">
35   <xs:complexType>
36     <xs:sequence>
37       <xs:element name="Age" type="xs:NMTOKEN" minOccurs="0"/>
38       <xs:element name="Size" type="xs:float" minOccurs="0"/>
39       <xs:element name="Weight" type="xs:float" minOccurs="0"/>
40     </xs:sequence>
41   </xs:complexType>
42 </xs:element>
43 </xs:sequence>
44 </xs:complexType>
45 </xs:element>
46 <xs:element name="Series">
47   <xs:complexType>
48     <xs:sequence>
49       <xs:element name="Modality" type="xs:NMTOKEN"/>
50       <xs:element name="SeriesNumber" type="xs:integer"/>
51     </xs:sequence>
52   </xs:complexType>
53 </xs:element>
54 <xs:element name="Equipment">
55   <xs:complexType>
56     <xs:sequence>
57       <xs:element name="Manufacturer" type="xs:token"/>
58       <xs:element name="InstitutionName" type="xs:token" minOccurs="0"/>
59       <xs:element name="ManufacturersModelName" type="xs:token" minOccurs="0"/>
60     </xs:sequence>
61   </xs:complexType>
62 </xs:element>
63 <xs:element name="WaveformModules">
64   <xs:complexType>
65     <xs:sequence>
66       <xs:element name="WaveformIdentification">
67         <xs:complexType>
68           <xs:sequence>
69             <xs:element name="Date" type="dicom_date"/>
70             <xs:element name="Number" type="xs:integer"/>
71             <xs:element name="Time" type="dicom_time"/>
72             <xs:element name="AcquisitionDatetime" type="dicom_datetime"/>
73           </xs:sequence>
74         </xs:complexType>
75       </xs:element>
76     <xs:element name="AcquisitionContext">
```



```
77     <xs:complexType>
78     <xs:sequence>
79     <xs:element name="AcquisitionContextSequence">
80     <xs:complexType>
81     <xs:sequence minOccurs="0" maxOccurs="unbounded">
82     <xs:element name="item">
83     <xs:complexType>
84     <xs:sequence>
85     <xs:element name="ConceptNameCodeSequence" type="dicom_code_seq_macro"/>
86     <xs:element name="ConceptCodeSequence" type="dicom_code_seq_macro"/>
87     </xs:sequence>
88     </xs:complexType>
89     </xs:element>
90     </xs:sequence>
91     </xs:complexType>
92     </xs:element>
93     </xs:sequence>
94     </xs:complexType>
95     </xs:element>
96     <xs:element name="WaveformAnnotation">
97     <xs:complexType>
98     <xs:sequence>
99     <xs:element name="WaveformAnnotationSequence">
100    <xs:complexType>
101    <xs:sequence maxOccurs="unbounded">
102    <xs:element name="item">
103    <xs:complexType>
104    <xs:sequence>
105    <xs:element name="ConceptNameCodeSequence" type="dicom_code_seq_macro"/>
106    <xs:element name="ReferencedWaveformChannels" type="xs:token"/>
107    <xs:element name="NumericValue" type="xs:float" minOccurs="0"/>
108    </xs:sequence>
109    </xs:complexType>
110    </xs:element>
111    </xs:sequence>
112    </xs:complexType>
113    </xs:element>
114    </xs:sequence>
115    </xs:complexType>
116    </xs:element>
117    <xs:element name="Waveform">
118    <xs:complexType>
119    <xs:sequence>
120    <xs:element name="WaveformSequence">
121    <xs:complexType>
```

```
122     <xs:sequence maxOccurs="unbounded">
123         <xs:element name="item">
124             <xs:complexType>
125                 <xs:sequence>
126                     <xs:element name="ChannelDefinitionSequence">
127                         <xs:complexType>
128                             <xs:sequence maxOccurs="unbounded">
129                                 <xs:element name="item">
130                                     <xs:complexType>
131                                         <xs:sequence>
132                                             <xs:element name="ChannelSourceSequence" type="dicom_code_seq_macro"/>
133                                             <xs:element name="ChannelSensitivity" type="xs:float"/>
134                                             <xs:element name="ChannelSensitivityUnitsSequence" type="dicom_code_seq_macro"/>
135                                             <xs:element name="ChannelSensitivityCorrectionFactor" type="xs:float"/>
136                                             <xs:element name="ChannelBaseline" type="xs:float"/>
137                                             <xs:element name="ChannelTimeSkew" type="xs:float"/>
138                                             <xs:element name="FilterLowFrequency" type="xs:float" minOccurs="0"/>
139                                             <xs:element name="FilterHighFrequency" type="xs:float" minOccurs="0"/>
140                                             <xs:element name="NotchFilterFrequency" type="xs:float" minOccurs="0"/>
141                                             <xs:element name="ChannelLabel" type="xs:token" minOccurs="0"/>
142                                             <xs:element name="ChannelStatus" type="xs:token" minOccurs="0"/>
143                                         </xs:sequence>
144                                     </xs:complexType>
145                                 </xs:element>
146                             </xs:sequence>
147                         </xs:complexType>
148                     </xs:element>
149                     <xs:element name="WaveformOriginality" type="xs:NMTOKEN"/>
150                     <xs:element name="NumberofWaveformChannels" type="xs:integer"/>
151                     <xs:element name="SamplingFrequency" type="xs:float"/>
152                     <xs:element name="WaveformPaddingValue" type="xs:short"/>
153                     <xs:element name="WaveformData" type="xs:string"/>
154                 </xs:sequence>
155             </xs:complexType>
156         </xs:element>
157     </xs:sequence>
158 </xs:complexType>
159 </xs:element>
160 </xs:sequence>
161 </xs:complexType>
162 </xs:element>
163 </xs:sequence>
164 </xs:complexType>
165 </xs:element>
166 </xs:sequence>
```

```
167     </xs:complexType>
168 </xs:element>
169 <xs:complexType name="dicom_name">
170   <xs:sequence>
171     <xs:element name="FirstName" type="xs:NMTOKEN"/>
172     <xs:element name="LastName" type="xs:NMTOKEN"/>
173   </xs:sequence>
174 </xs:complexType>
175 <xs:complexType name="dicom_code_seq_macro">
176   <xs:sequence>
177     <xs:element name="item">
178       <xs:complexType>
179         <xs:sequence>
180           <xs:element name="CodeValue" type="xs:token"/>
181           <xs:element name="CodingSchemeDesignator" type="xs:token"/>
182           <xs:element name="CodingSchemeVersion" type="xs:token"/>
183           <xs:element name="CodeMeaning" type="xs:token"/>
184         </xs:sequence>
185       </xs:complexType>
186     </xs:element>
187   </xs:sequence>
188 </xs:complexType>
189 <xs:simpleType name="dicom_datetime">
190   <xs:restriction base="xs:NMTOKEN">
191     <xs:length value="14"/>
192   </xs:restriction>
193 </xs:simpleType>
194 <xs:simpleType name="dicom_time">
195   <xs:restriction base="xs:NMTOKEN">
196     <xs:length value="6"/>
197   </xs:restriction>
198 </xs:simpleType>
199 <xs:simpleType name="dicom_date">
200   <xs:restriction base="xs:NMTOKEN">
201     <xs:length value="8"/>
202   </xs:restriction>
203 </xs:simpleType>
204 <xs:simpleType name="dicom_sex" final="restriction">
205   <xs:restriction base="xs:NMTOKEN">
206     <xs:enumeration value="M"/>
207     <xs:enumeration value="F"/>
208     <xs:enumeration value="O"/>
209   </xs:restriction>
210 </xs:simpleType>
211 </xs:schema>
```

Uma ferramenta para conversão de exames de eletrocardiografia para o formato DICOM

Tiago Silva Proença¹, João Bosco Manguiera Sobral², Euclides de Moraes Barros Júnior³

12 de julho de 2004

Departamento de Informática e Estatística - INE
Universidade Federal de Santa Catarina (UFSC), Brasil, 88040-900
Fone: (48)331-9498, Fax: (48)331-9770
tsp@inf.ufsc.br, bosco@inf.ufsc.br, euclides@sc.estacio.br

Resumo

A grande maioria dos aparelhos médicos utilizados no âmbito de clínicas e hospitais modernos, codificam e trocam dados seguindo o protocolo padrão de comunicação DICOM, permitindo um intercâmbio de exames entre equipamentos e profissionais da área médica.

O projeto em questão foi elaborado para suprir carências no que diz respeito ao armazenamento de sinais vitais provenientes de exames de eletrocardiografia, onde seria possível utilizar os recursos computacionais na análise e diagnóstico médico mesmo em ambientes que não disponibilizam equipamentos modernos.

Para tanto, foi implementada uma metodologia capaz de adequar um arquivo XML para uma futura exportação no formato DICOM, através de uma aplicação desenvolvida para este propósito.

Palavras Chave: Eletrocardiograma (ECG), DICOM, *Waveform*.

Abstract

Most medical equipment that is used in modern clinics and hospitals, codifies and process data following the DICOM communication protocol, allowing medical exams exchange between equipment and professionals in the medical area.

This project was developed to resolve the problem of storing vital signals from electrocardiographic exams, so that it would be possible to use the computational resources for analysis and diagnosis, even in places where modern equipments is not available.

To do this, a methodology was implemented capable of adjusting an XML file for a future exportation into the DICOM format, through an application developed for this purpose.

Key-Words: Electrocardiographic (ECG), DICOM, *Waveform*.

1 Introdução

Desde o início do século XX, a inovação e utilização de eletrocardiógrafos vieram influenciar de forma significativa à medicina, trazendo inúmeros benefícios na detecção de patologias do coração.

A grande maioria dos aparelhos médicos utilizados em modernas clínicas de diagnóstico por imagem atualmente codificam e trocam dados seguindo o protocolo padrão de comunicação DICOM (*Digital Imaging and Communications in Medicine*).

O projeto proposto foi elaborado com o propósito de suprir uma necessidade da empresa DIXTAL, a qual foi utilizada neste trabalho como estudo de caso. Esta empresa atua na área da saúde, desenvolvendo tecnologia de informação para a captação de sinais vitais, e apresentava uma carência no que diz respeito ao armazenamento de informações em conformidade com o padrão DICOM.

Diante disto, foi desenvolvido um software utilizando iniciativas de software livre, para resolver tal problema. Em contrapartida, o produto final resultante deste trabalho pode ser utilizado para conversão de sinais de qualquer empresa, desde que esta armazene os exames no formato XML, seguindo o modelo de documento utilizado no trabalho.

Então, este documento apresenta os principais aspectos técnicos, metodológicos e práticos envolvidos na experiência de efetuar a conversão de exames de Eletrocardiograma (ECG) provenientes de um equipamento médico. Tal experiência consistiu no desenvolvimento de uma aplicação em linguagem JAVA que efetua a conversão de um exame de ECG armazenado no formato XML, previamente modelado, para o formato DICOM.

2 Eletrocardiograma

Segundo a Sociedade Brasileira de Cardiologia (1996) o eletrocardiograma (ECG) consiste em um exame complementar usado em cardiologia, o qual avalia a atividade elétrica do coração. Neste exame, são identificados o ritmo, a frequência cardíaca, possíveis sobrecargas, alterações na nutri-

ção do coração (isquemia) e outras várias informações complementares que o método pode nos fornecer. O ECG se destaca por não oferecer riscos para o paciente, é indolor e de baixo custo quando comparado a outros procedimentos.

A metodologia consiste em ligar sobre alguns centros de excitação denominados nós, eletrodos que recolhem e traduzem os estímulos elétricos em uma série de ondas, e fazer o registro em papel desta atividade elétrica do coração. O aparelho usado para registrar a atividade cardíaca é chamado eletrocardiógrafo.

Existe uma série de características eletrocardiográficas consideradas, em conjunto, como um "padrão de normalidade", e que apresentam variações individuais como as dependentes de idade, sexo e tipo constitucional (TRANCHESI, 1972).

Devemos ressaltar o fato de existir anomalias cardíacas que não alteram o ECG. Assim um traçado que estaria dentro dos limites de normalidade, não exclui a possibilidade de cardiopatia (TRANCHESI, 1972).

3 DICOM

O DICOM é uma padrão internacional que engloba diversos aspectos de aquisição de uma imagem ou onda em formato digital. Nele além dos dados codificados, são inclusos parâmetros para visualização destas imagens, a forma de arquivamento em disco, informações sobre transferência em rede de computadores entre outros.

O DICOM utiliza modelos explícitos e detalhados de como entidades reais (pacientes, imagens, ondas, relatórios) envolvidos no contexto de aquisição de um exame, são descritos e como são relacionados. São estes modelos chamados de entidade-relacionamento (E-R) que irão garantir que fabricantes e usuários de equipamentos compreendam e utilizem o padrão de forma adequada.

No modelo de dados DICOM, um paciente tem um ou mais estudos. Um estudo corresponde a uma visita do paciente a uma instituição de saúde. Cada estudo contém uma ou mais séries, que são seqüências de exames da mesma modalidade. Uma série contém imagens, laudos, curvas ou outros objetos (BORTOLUZZI, 2003).

Apesar de sua importância na área de informática médica, radiologia e sistemas de informação hospitalar (HIS), o padrão DICOM não é largamente conhecido (SAMPAIO, 1999).

3.1 *Waveform*

A entidade de informação DICOM *Waveform* é a entidade DICOM para representação de exames no formato de sinais, e é composta por várias classes de objetos de informação (IOD), os quais representam ondas específicas como eletrocardiograma, sinais de equipamentos de hemodinâmica e áudio.

O modelo Entidade-Relacionamento (E-R) da entidade de informação *Waveform* é aplicado a todos os IODs *Waveform*, e é apresentado na Figura 1.

De acordo com a PS (*Part of Standard*) 3.3 do padrão, pode-se observar que todos os IODs possuem uma tabela com módulos (IOM) que representam sua constituição. Todos os IODs *Waveform* possuem a mesma tabela de módulos, mudando somente como são preenchidos alguns destes módulos. Esta tabela é ilustrada na Tabela ??.

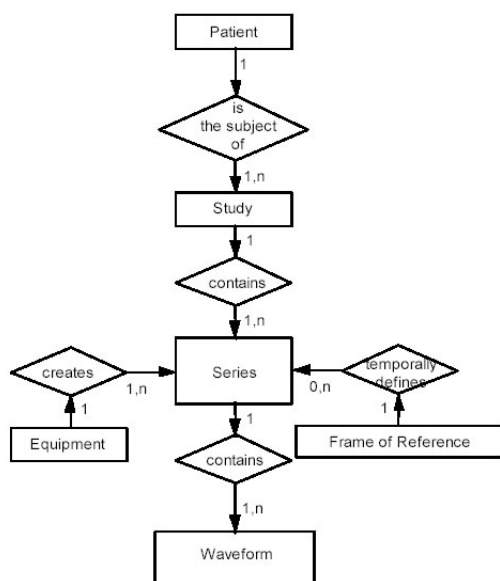


Figura 1: Modelo de informação DICOM *Waveform* IOD

3.2 *General Electrocardiogram Information Object Definition*

Este IOD é representado pela modalidade ECG, e é a especificação de sinais elétricos digitalizados do sistema de condução cardíaco do paciente, coletado na superfície do corpo, adquirido por uma modalidade ECG ou por uma função de aquisição ECG, contida dentro de uma modalidade de imagem. Este IOD deve ter um número de canais *waveform* entre 1 e 13 e frequência de amostragem de 200 à 1000 Hz.

4 O Sistema

Para a solucionar o problema, foi feita a implementação de uma aplicação em linguagem JAVA, que converte um exame de ECG armazenado em formato XML para o formato DICOM. Diante deste contexto, o projeto desenvolveu-se em três partes principais: modelagem do arquivo XML, modelagem orientada a objetos (OO) do IOD *Waveform* e implementação do sistema.

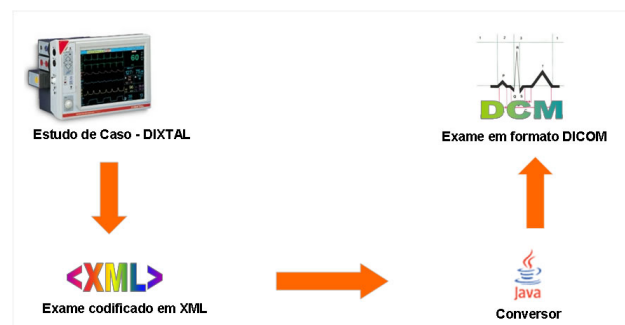


Figura 2: O sistema proposto

4.1 O Arquivo XML

Um dos fatores mais importantes para o funcionamento do sistema proposto é o arquivo XML, por ser a interface de comunicação entre o sistema e o sistema DX-25 da DIXTAL.

O arquivo XML armazenado pelo sistema DX-25, deverá conter informações sobre o paciente, parâmetros de captura do eletrocardiógrafo, além dos traçados fornecidos pelo exame.

A estratégia utilizada foi modelar o arquivo XML seguindo o modelo de um *Waveform* IOD.

O objeto de informação que utilizamos foi o *General ECG IOD*, por possuir uma abrangência maior em relação ao número de canais.

Para validação da estrutura e dos elementos do documento XML foi criado um XML *Schema*.

4.2 Modelagem OO do Waveform IOD

Um IOD *Waveform* é formado por uma tabela de módulos que representam a constituição do seu objeto de informação. Dentre estes módulos, existem aqueles que são tipo M (*Mandatory Modules*), que são aqueles que são obrigatórios para a implementação do IOD. Por questões óbvias, na modelagem das classes só foi utilizado módulos tipo M.

Durante o processo de modelagem, tentou-se seguir de maneira mais fiel possível o modelo de informação DICOM. O resultado disto é um conjunto formado por 52 classes, contidas em 10 pacotes.

4.3 Implementação do Sistema

O sistema se trata de uma simples aplicação modo texto, que é o resultado da implementação do modelo OO gerado na segunda fase do projeto. Para executar a aplicação, basta entrar com seu nome na linha de comando, seguido do nome do arquivo XML. O *parser* XML irá ler o arquivo XML e efetuar a validação do documento XML. Feito isto, será instanciado um objeto da classe *GeneralECGXMLReader* que irá ler os valores dos elementos do documento XML e preencher o objeto *GeneralECGIOD*. Após isto, uma instância da classe *DicomOutputStream* irá gerar um arquivo com extensão DCM, que é a representação do exame no padrão DICOM.

4.3.1 Identificação

Um aspecto importante no desenvolvimento da aplicação foi conseguir uma raiz perante uma autoridade de padronização. Esta raiz é utilizada para representar a empresa, e assim poder gerar os identificadores únicos (UID). A entidade de padronização escolhida foi a *Internet Assigned Numbers Authority* (IANA) e raiz fornecida

foi 1.3.6.1.4.1.19491. Dessa forma, foi possível gerar o identificador para a instância SOP (*Service Object Pair Class*) sem problemas, além de outros identificadores.

4.4 Validação

Para verificar se o arquivo gerado atendia corretamente à especificação do padrão DICOM, optou-se por fazer testes com duas aplicações bem conhecidas e respeitadas: o DCMTK - DICOM *Toolkit* do instituto alemão OFFIS e o PixelMed Java DICOM *Toolkit* criado pelo respeitado David Clunie, uma das pessoas que ajuda a desenvolver o padrão.

- **DCMTK - DICOM *Toolkit*:** Dentro desse conjunto de ferramentas, foi utilizado para realizar o teste de conformidade da estrutura do documento DICOM a aplicação *dcmdump*. Se algum elemento de dados estiver sido gravado incorretamente, é possível identificar através deste aplicativo.
- **PixelMed Java DICOM *Toolkit*:** Dentro desse conjunto de ferramentas, foi utilizado o *ECGViewer*, uma aplicação que permite a visualização de exames de ECG. Através deste aplicativo é possível verificar se as coordenadas dos pontos do plano cartesiano tinham sido armazenadas corretamente, através da plotagem dos sinais do ECG.

5 Considerações Finais

Através da implantação do software, é possível mensurar que o objetivo do trabalho tenha sido alcançado, pois o software apresentou os resultados almejados, conseguindo gerar de forma eficiente os arquivos em conformidade com o padrão DICOM.

A validação obtida foi realizada através de testes com outras aplicações compatíveis com o padrão e com a utilização do software em ambientes como clínicas e hospitais, possibilitando a conclusão do projeto de acordo com os objetivos propostos.

Apesar do foco do trabalho ter sido a conversão de exames da empresa DIXTAL, seu campo de

aplicações é vasto e pode ser utilizado para conversão de sinais de qualquer empresa, desde que esta armazene os exames no formato XML, seguindo o modelo de documento utilizado no trabalho.

Além disso, a flexibilidade da ferramenta desenvolvida permite que pequenas adaptações e adição de novos módulos ampliem mais seu campo de atuação, podendo efetuar conversões para outras modalidade DICOM.

Com relação às contribuições à DIXTAL, o sistema se mostra uma ótima solução, pois possibilita a conversão de exames em equipamentos já existentes. Utilizando DICOM ao invés do padrão proprietário que era utilizado, a empresa consegue atingir um número maior de clientes e se coloca no mesmo patamar de grandes empresas como SIEMENS, GE entre outras.

Existe grandes potenciais em utilizar o padrão DICOM no contexto de clínicas e hospitais. Isto é verificado claramente em um ambiente PACS (*Picture Archiving and Communications System*). De qualquer forma, estes potenciais devem ser explorados, mesclando-se outras tecnologias, para se obter ferramentas e soluções interessantes e criativas, que venham a contribuir com a área de informática médica.

Referências

BORTOLUZZI, M. K. *Desenvolvimento e implementação de um editor de documentos estruturados no padrão DICOM Structured Report*. Dissertação (Mestrado em Ciências da Computação) — Universidade Federal de Santa Catarina, Florianópolis, 2003.

SAMPAIO, S. C. *Modelagem e implementação orientada a objetos de um cliente de rede para banco de dados de imagens médicas digitais utilizando o padrão DICOM 3.0*. Dissertação (Mestrado em Ciências da Computação) — Universidade Federal de Santa Catarina, Florianópolis, 1999.

SOCIEDADE BRASILEIRA DE CARDIOLOGIA. *O que é eletrocardiograma e para que serve?* Maio 1996. Disponível em: <<http://prevencao.cardiol.br/sbc-funcor/cuide/coracaonline/Responline.asp?P1=40&CodSes=2>> Acesso em: Maio, 2004.

TRANCHESI, J. *Eletrocardiograma normal e patológico*. São Paulo: Atheneu, 1972.