



**UNIVERSIDADE FEDERAL DE JUIZ DE FORA  
INSTITUTO DE CIÊNCIAS EXATAS  
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO**

**A UTILIZAÇÃO DO PADRÃO DICOM EM SISTEMAS DE  
ARMAZENAMENTO E COMUNICAÇÃO DE IMAGENS  
MÉDICAS DIGITAIS (PACS)**

**Rodrigo Alexandre Santos Cangussu**

JUIZ DE FORA  
MARÇO, 2012

**A UTILIZAÇÃO DO PADRÃO DICOM EM SISTEMAS DE  
ARMAZENAMENTO E COMUNICAÇÃO DE IMAGENS  
MÉDICAS DIGITAIS (PACS)**

**RODRIGO ALEXANDRE SANTOS CANGUSSU**

Universidade Federal de Juiz de Fora  
Instituto de Ciências Exatas  
Departamento de Ciência da Computação  
Bacharel em Ciência da Computação

Orientador: Prof. Dr. Rodrigo Weber dos Santos

JUIZ DE FORA  
MARÇO, 2012

# A UTILIZAÇÃO DO PADRÃO DICOM EM SISTEMAS DE ARMAZENAMENTO E COMUNICAÇÃO DE IMAGENS MÉDICAS DIGITAIS (PACS)

Rodrigo Alexandre Santos Cangussu

MONOGRAFIA SUBMETIDADA AO CORPO DOCENTE DO INSTITUTO DE CIÊNCIAS EXATAS DA UNIVERSIDADE FEDERAL DE JUIZ DE FORA COMO PARTE INTEGRANTE DOS REQUISITOS NECESSÁRIOS PARA OBTENÇÃO DO GRAU DE BACHAREL EM CIÊNCIA DA COMPUTAÇÃO.

Aprovada por:

---

Prof Dr. Rodrigo Weber dos Santos  
(Presidente)

---

Prof Dr. Ciro de Barros Barbosa

---

Prof Dr. Marcelo Lobosco

JUIZ DE FORA, MG – BRASIL  
MARÇO, 2012

## **Agradecimentos**

Aos meus pais “Seu” Leno e Dona Biga, por me incentivarem no desenvolvimento intelectual durante todo o tempo. O resultado que colho é fruto do empenho e dedicação de vocês.

À minha namorada (esposa) Paula por me apoiar incondicionalmente estando presente em todos momentos.

À minha filha Jennifer por trazer um maior sentido a todas as coisas.

Ao meu orientador por ajudar na elaboração deste trabalho.

À banca examinadora que aceitou prontamente o chamado para a apresentação.

A esta instituição de ensino pela minha formação, me fornecendo os conhecimentos necessários para a elaboração deste estudo.

## SUMÁRIO

<b>INTRODUÇÃO</b> .....	9
<b>CAPÍTULO I O padrão DICOM</b> .....	11
1.1 Imagens médicas digitais.....	11
1.2 Organização do padrão 3.0.....	12
1.3 Modelo de Informação.....	13
1.3.1 Definição de Objeto-Informação.....	13
1.3.2 Classes de Serviço.....	13
1.3.3 Par Serviço-Objeto.....	14
1.4 Modelo de comunicação.....	14
1.4.1 Estabelecimento da Associação.....	15
1.4.2 Formato da mensagem.....	15
1.4.2.1 Conjunto de Comando (Command Set ).....	16
1.4.2.2 Conjunto de Dados (Data Set).....	16
1.4.3 Sintaxe de Transferência.....	17
1.4.4 Identificadores Únicos (UID).....	17
1.5 Persistência de objetos.....	18
1.5.1 Formato de Arquivo.....	18
1.5.2 Espaço de Nomes.....	19
1.6 Acesso Web a objetos persistentes DICOM.....	19
1.7 Hospedagem de aplicações ( <i>Application Hosting</i> ).....	20
<b>CAPÍTULO 2 Os Sistemas de Armazenamento e Distribuição de Imagens Médicas Digitais (PACS)</b> .....	22

2.1 Radiologia Digital.....	22
2.1.1 PACS/RIS/HIS.....	23
2.2 Os componentes de um PACS.....	23
2.2.1 Gateway de aquisição de imagens .....	24
2.2.2 Apresentação de imagens.....	24
2.2.3 Controlador PACS e servidor de armazenamento de imagens.....	25
<b>CAPÍTULO 3 Implementação de Sistemas de Armazenamento e Comunicação de Imagens Médicas Digitais utilizando DICOM.....</b>	<b>26</b>
3.1 Materiais e métodos.....	27
3.2 Resultados.....	28
3.2 Discussões.....	30
<b>CONCLUSÃO.....</b>	<b>32</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>33</b>
<b>APÊNDICE A – Roteiro de Implementação.....</b>	<b>37</b>

## LISTA DE FIGURAS

Figura 1.1 Arquitetura de protocolos do modelo de comunicação DICOM.....	14
Figura 1.2 Mensagem DICOM.....	16
Figura 1.3 Arquivo DICOM.....	18
Figura 1.4 <i>file-set DICOM</i> .....	19
Figura 1.5: Acesso a objetos persistentes via aplicação hospedada.....	28
Figura 2.1 Sistema de armazenamento e distribuição de imagens médicas (PACS).....	24
Figura 3.1a Tela da aplicação cliente PACS para o envio de arquivos de imagem.....	27
Figura 3.1b Tela da aplicação cliente PACS para o envio de arquivos de imagem.....	28
Figura 3.2: Tela da aplicação cliente para a pesquisa e recuperação de arquivos.....	29
Figura 3.3 Saída apresentada no Conquest em sistema Windows após acesso web.....	29
Figura 3.4 ImageJ.....	30

## LISTA DE SIGLAS

*ACR American College of Radiology*  
*ACSE Association Control Service Element*  
*AE Application Entity*  
*ANSI American National Standards Institute*  
*API Application Programming Interface*  
*DICOM Digital Imaging and Communication in Medicine*  
*DIMSE DICOM Message Service Element*  
*DIMSE-C DICOM Message Service Element-Composite*  
*DIMSE-N DICOM Message Service Element-Normalized*  
*FSC File-set Creator*  
*FSR File-set Reader*  
*FSU File-set Updater*  
*HIS Hospital Information System*  
*HL7 Health Level 7*  
*HTTP Hypertext Transfer Protocol*  
*ID Identifier*  
*IOD Information Object Definition*  
*ISO International Standards Organization*  
*JPEG Joint Photographic Experts Group*  
*NEMA National Electrical Manufacturers Association*  
*OSI Open Systems Interconnection*  
*ISO International Organization for Standardization*  
*PACS Picture Archiving and Communication Systems*  
*RIS Radiology Information System*  
*RSNA Radiological Society of North America*  
*SOP Service-Object Pair*  
*TCP/IP Transmission Control Protocol/Internet Protocol*  
*UID Unique Identifier*  
*VR Value Representation*

## **Resumo**

O presente trabalho procura mostrar a aplicação do padrão DICOM em Sistemas de armazenamento e transmissão de imagens médicas digitais (*PACS – Picture Archiving and Communication Systems*). A metodologia utilizada foi a realização de experimentos através da implementação de um modelo de sistema que realiza o armazenamento, transmissão e visualização de imagens e dados a ela associados, como dados do paciente, do equipamento, entre outros que compõem o arquivo de exame. Os resultados demonstraram que o DICOM possibilita eficientemente a interação entre as aplicações de um PACS através da definição de objetos, serviços e protocolos de comunicação de dados.

## INTRODUÇÃO

Amplamente utilizadas em instituições hospitalares, as imagens radiológicas são ferramentas importantes na prática da medicina, por representar uma forma de visualização não-invasiva de órgãos, tecidos, ossos e outras estruturas do corpo humano, no apoio a diagnósticos, tratamentos e planejamento de cirurgias.

A obtenção de imagens em filme-écran através de processamento químico, é realizada desde a descoberta dos raios X em 1895, e ainda é a forma predominante de obtenção de imagens clínicas. No entanto, nos últimos 20 a 30 anos, os avanços no uso da computação possibilitou a utilização de tecnologias digitais para a aquisição, o armazenamento, o acesso e a visualização destas imagens. O armazenamento de filmes requer um grande espaço nos departamentos de radiologia. Geralmente os exames mais antigos são estocados em local separado do departamento, dificultando o acesso e muitas vezes inviabilizando-o em áreas geograficamente distantes. A utilização do formato digital proporciona um panorama de redução do espaço físico requerido, custo de material, redução do trabalho manual tradicional do manuseio de filmes e facilidade de acesso aos exames. Adicionalmente, técnicas de processamento de imagens podem ser aplicadas para adicionar valor ao diagnóstico, possibilitando uma melhor visualização da imagem através de algoritmos para o realce de contornos, suavização, inversão de cores, ou outras alterações nos atributos da imagem.

Entretanto, o uso cada vez maior de equipamentos digitais para o diagnóstico torna crítica a realização das tarefas de armazenamento e distribuição das imagens, bem como outras informações a elas associadas como o nome do paciente, modalidade de diagnóstico utilizada, posição de obtenção da imagem e muitos outros elementos utilizados na radiologia digital. Desta forma, estabeleceu-se a necessidade da criação de uma estrutura computacional que possibilitasse a troca de dados de imagens de forma consistente dentro

do ambiente clínico. Em resposta a esta necessidade, surgiu o conceito de PACS (*Picture Archiving and Communication System*). Um PACS é construído com o intuito de gerenciar o grande volume de dados gerados pelos diferentes equipamentos de aquisição de imagens, oferecendo funções de armazenamento, recuperação, processamento, visualização e disponibilização dos dados através de redes de transmissão. Para isto, foi necessário padronizar um modelo de informação e comunicação para que os dispositivos e aplicações compartilhem da mesma visão sobre os dados do sistema. Esta padronização é realizada através da adoção do padrão DICOM (*Digital Imaging Communication in Medicine*), utilizado atualmente pela maioria dos fabricantes de equipamentos médicos e aplicações que compõem um PACS. O padrão DICOM define o contexto da informação compartilhada em um PACS através da especificação de objetos e serviços equivalentes às tarefas presentes no fluxo de trabalho de um ambiente radiológico.

Este trabalho mostra como a utilização do padrão DICOM possibilita a construção dos PACS, revisando e demonstrando, através da implementação de um modelo cliente-servidor, os principais conceitos do DICOM 3.0 e dos sistemas de armazenamento e comunicação de imagens médicas.

O trabalho está organizado da seguinte forma: o Capítulo 1 aborda o padrão DICOM, seus protocolos e definições; o Capítulo 2 descreve a arquitetura e aplicação dos PACS em um ambiente radiológico; no Capítulo 3 é demonstrada a utilização do DICOM através da realização de experimentos realizados com a implementação de um modelo de sistema PACS, utilizando tecnologias de código aberto.

## **1 O padrão DICOM**

O padrão DICOM é um conjunto de normas para o processamento de imagens médicas em formato digital. Facilita a interoperabilidade entre aplicações, especificando: um conjunto de protocolos de rede a serem utilizados pelos dispositivos (aplicações); um modelo de informação; a sintaxe e a semântica de comandos utilizados para processar os objetos do modelo (NEMA, 2011).

### **1.1 Imagens médicas digitais**

As imagens obtidas por modalidades de aquisição de imagem podem ser representadas em formato digital. Uma imagem digital é representada como uma matriz retangular de *pixels* (*picture elements*), pequenos pontos de diferentes cores que formam a imagem atual. Uma imagem de tomografia computadorizada típica é uma matriz de 512 *pixels* de largura e 512 *pixels* de altura. Dispondo estes elementos em sequência teremos 262.144 *pixels* (512 x 512), que podem ser mapeados em *bytes* e organizados em um arquivo em formato definido pelo padrão, no qual a imagem é apenas parte do seu conteúdo. O suporte a imagens no DICOM se estende a formatos que utilizam algoritmos usuais de compressão tais como RLE, JPEG, JPEG2000, JPEG-LS, e outros (Pianykh,2009,p.86-90).

## 1.2 Organização do padrão DICOM 3.0

Como lembra Pianykh (2009), o padrão DICOM foi concebido em 1983 por um comitê formado pelo *American College of Radiology (ACR)* e a *National Electronics Manufacturer's Association (NEMA)*, sendo publicado em 1985 e denominado ACR-NEMA 1.0, o qual continha muitos erros e imperfeições. Diante disto, subcomitês foram criados para se dedicar a aprimorar partes específicas e foi liberada em 1988 uma segunda versão, a ACR-NEMA 2.0. Em 1993, uma terceira versão foi apresentada, em resposta a necessidades identificadas com o uso da versão anterior, bastante limitada quanto a comunicação de dados. O padrão passou a ser chamado DICOM 3.0, ou simplesmente DICOM. Atualmente, o padrão DICOM consiste em 20 volumes, nomeados da seguinte forma:

- PS 3.1: *Introduction and Overview*
- PS 3.2: *Conformance*
- PS 3.3: *Information Object Definitions*
- PS 3.4: *Service Class Specifications*
- PS 3.5: *Data Structures and Encoding*
- PS 3.6: *Data Dictionary*
- PS 3.7: *Message Exchange*
- PS 3.8: *Network Communication Support for Message Exchange*
- PS 3.9: *Point-to-Point Communication Support for Message Exchange (Retired)*
- PS 3.10: *Media Storage and File Format for Media Interchange*
- PS 3.11: *Media Storage Application Profiles*
- PS 3.12: *Media Formats and Physical Media-for-Media Interchange*
- PS 3.13: *Print Management Point-to-Point Communication Support (Retired)*
- PS 3.14: *Gray Scale Standard Display Function*
- PS 3.15: *Security Profiles*
- PS 3.16: *Content Mapping Resource*
- PS 3.17: *Explanatory Information*
- PS 3.18: *Web Access to DICOM Persistent Objects*
- PS 3.19: *Application Hosting*
- PS 3.20: *Transformation of DICOM to and from HL7 standards*

Estes documentos estabelecem informações detalhadas sobre o modelo das estruturas de dados e serviços necessários à intercomunicação de equipamentos de aquisição de imagem médica, estações de trabalho para visualização/processamento de imagens, unidades de armazenamento e impressão (NEMA,2011).

### **1.3 – Modelo de informação**

#### **1.3.1 Definição de Objeto-Informação**

Um IOD (*Information Object Definition*) é um modelo abstrato de dados utilizado para especificar objetos reais da prática clínica. Possibilita às entidades de aplicação partilhar uma visão comum das informações que serão trocadas. Um IOD representa uma classe de objetos reais que partilham a mesmas propriedades e atributos, podendo ser de dois tipos: normalizado ou composto.

Os IOD's normalizados representam uma única entidade no modelo. São alguns dos IOD's normalizados: *Patient*, *Study*, *Print Queue*. Os IOD's compostos combinam os normalizados de forma a representar diferentes entidades (NEMA,2011).

#### **1.3.2 Classes de Serviço**

As Classes de Serviço correspondem aos tipos de tarefas que são implementadas pelas aplicações, para que possam executar serviços sobre determinado objeto. Quando um dispositivo ou aplicação oferece um serviço, ele atua como provedor de classe de serviços (*Service Class Provider - SCP*). Caso o dispositivo utilize algum serviço, atua como um usuário de classe de serviços (*Service Class User – SCU*) (NEMA,2011).

Dependendo do caso, uma aplicação pode atuar como SCP, SCU ou ambos. Como exemplo, consideremos a classe de serviços *Storage Service Class*. Um servidor de armazenamento atua como *Storage SCP* ao receber os objetos para o armazenamento. Por outro lado, atua como SCU quando responde a uma solicitação de recuperação de algum objeto (Pianykh,2009).

### 1.3.3 Par Serviço-Objeto

As classes de serviço e os IOD's são combinados para formar a unidade fundamental do DICOM, a classe SOP (*Service-Object Pair*) (NEMA,2011).

Desta forma, um objeto-informação (IOD) CT produzido por uma modalidade de tomografia computadorizada, por exemplo, pode ser combinado com uma classe de serviço de armazenamento (*Storage Service Class*) para formar a classe SOP CT *Image Storage*, cujo envio através da interface de rede equivale a uma solicitação para que seja feito o armazenamento da imagem (Pianykh,2009).

### 1.4 Modelo de Comunicação

O modelo adotado no padrão para a comunicação de mensagens é baseado no modelo de referência para Interconexão de Sistemas Abertos (*Open System Interconnection - OSI*) desenvolvido pela *International Organization for Standardization - ISO*, com o propósito de padronizar a comunicação entre sistemas de processamento heterogêneos (ISO,2011).

Através de uma camada de interface, as aplicações (AE's - *Application Entities*), estabelecem/terminam associações, e transferem mensagens, isolando a camada de aplicação DICOM da pilha de protocolos de comunicação das camadas como representado na Figura 1.1 (NEMA,2011).

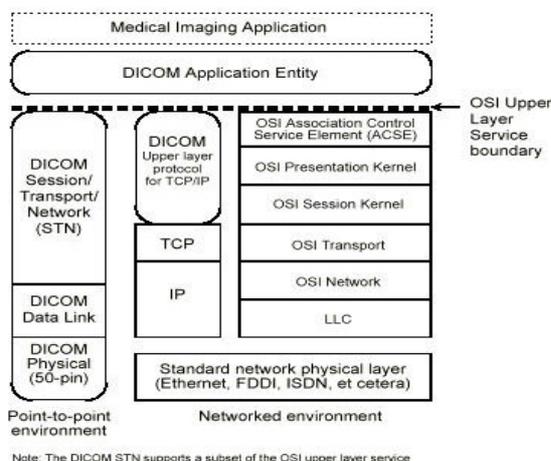


Figura 1.1: Arquitetura de protocolos do modelo de comunicação DICOM

(fonte: <http://203.71.232.36/hsir/medical/content/dicom3.htm>)

### 1.4.1 Estabelecimento da associação

A troca de mensagens começa com o estabelecimento de uma associação utilizando o protocolo ACSE (*Association Control Service Element*). O ACSE é o elemento básico da camada de aplicação. Ele é o responsável pelo estabelecimento e pela liberação da associação entre duas entidades de aplicação (AE's).

Após a associação é utilizado o protocolo DIMSE (*Dicom Message Service Element*) para a troca de mensagens para a troca de mensagens. Este protocolo vai oferecer as condições de operação dos serviços DIMSE-C e DIMSE-N. O DIMSE-C atende a operações associadas a instâncias SOP compostas e o DIMSE-N atende a operações e notificações associadas a instâncias SOP normalizadas. A Tabela 1.1 apresenta a relação destes serviços.

Name	Group	Type
C-STORE	DIMSE-C	operation
C-GET	DIMSE-C	operation
C-MOVE	DIMSE-C	operation
C-FIND	DIMSE-C	operation
C-ECHO	DIMSE-C	operation
N-EVENT-REPORT	DIMSE-N	notification
N-SET	DIMSE-N	operation
N-GET	DIMSE-N	operation
N-ACTION	DIMSE-N	operation
N-CREATE	DIMSE-N	operation
N-DELETE	DIMSE-N	operation

Tabela1.1: DIMSE Services

(fonte: <http://www.dabsoft.ch/dicom/7/7.5/>)

### 1.4.2 Formato da mensagem

As mensagens trocadas entre aplicações DICOM (AE's) através de uma interface de rede são compostas de um conjunto de comando (*Command Set*) seguido por um conjunto

de dados (*Data Set*). O *Command Set* é usado para indicar a operação que será executada no conjunto de dados. Esta estrutura é mostrada na Figura 1.2 (NEMA,2011).

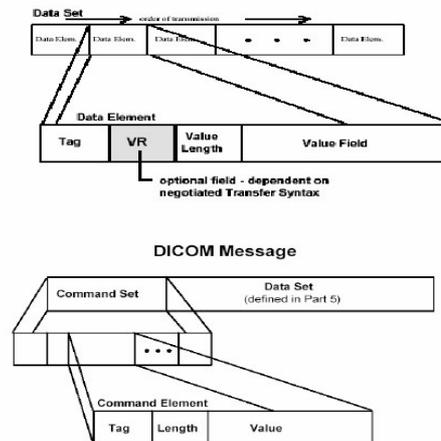


Figura 1.2: Mensagem DICOM

(fonte: <http://medical.nema.org/standard.html>)

#### 1.4.2.1 Conjunto de Comandos (Command Set)

Um *Command Set* é composto por Elementos de Comando. Cada Elemento de Comando é formado por três campos: *Tag* (etiqueta), *Value Length* (tamanho do valor) e *Value Field* (campo do valor). Os elementos são dispostos em ordem crescente de acordo com o número da etiqueta (*Tag*), que o identifica unicamente no Conjunto de Comando. A etiqueta é um hexadecimal de 32 bits (dois inteiros sem sinal), no formato (gggg,eeee), 2 bytes representando o número do grupo e 2 bytes para representar o número do elemento. O campo *Length* contém o tamanho do campo *Value* e o campo *Value* contém o valor do Elemento de Comando (NEMA,2011).

#### 1.4.2.2 Conjunto de Dados (Data Set)

Um *Data Set* é composto por Elementos de Dados. Os elementos de dados contém os valores dos atributos do objeto DICOM. Cada elemento de dado é composto de três ou quatro campos: *Tag* (etiqueta), *VR* (representação de valor), *Value Length* (tamanho do valor) e *Value Field* (campo do valor). O campo *VR* pode ser explícito ou implícito, e

contém 2 *bytes* para especificar o tipo de dado e a sintaxe de transferência. Quando implícito, os dados são utilizados na sintaxe de transferência padrão utilizada (*implicit VR Little Indian*). Quando o VR for explícito, a representação dos dados será de acordo com o valor do campo VR.

### 1.4.3 Sintaxe de Transferência

A sintaxe de transferência é a uma descrição formal do modo como os VR's dos elementos de dados são apresentados e como sua codificação é realizada (RSNA,1997).

As sintaxes de transferência básicas estão organizadas segundo dois critérios: forma de disposição dos *bytes*, em *Big* ou *Little Endian*; e, presença ou não, do Valor de Representação de cada um dos dados que compõem um arquivo de exame, *Explicit* ou *Implicit*. Desta forma, são definidos três tipos básicos de codificação: *Big Endian Explicit*, *Little Endian Explicit* e *Little Endian Implicit*. Isto permite às aplicações negociar as técnicas de codificação dos dados (NEMA,2011).

### 1.4.4 Identificadores Únicos (UID)

O UID (*Unique Identifier*) é um número de identificação único utilizado nas mensagens, associado a uma organização e a uma função. Este número é composto por duas partes (UID = <prefixo (org root) >. <sufixo>) e deve ter um limite máximo de 64 caracteres incluindo os separadores.

O prefixo identifica uma organização (fabricante, organização de pesquisa, NEMA). A parte <sufixo> do UID representa a aplicação, por exemplo, transferência de imagens de tomografia na função de usuário, e deve ser único dentro do prefixo <org root>. Isto implica que a organização identificada como raiz (org root) é responsável pela garantia da unicidade do sufixo.

A organização responsável pela definição e registro destes UID's é a NEMA. Outros prefixos privados podem ser utilizados por outras organizações. Neste caso, somente devem ser utilizados quando não houver UID's DICOM referentes a aquela função

desejada. O prefixo (org root) “1.2.840.10008” é reservado para itens do padrão DICOM, e não deve ser utilizado para definir itens privados (NEMA,2011).

## 1.5 Persistência

### 1.5.1 Formato de Arquivo

A definição do arquivo DICOM fornece uma forma de armazenar um conjunto de dados relativo a um par serviço-objeto (SOP). Cada arquivo pode conter informação associada a uma única instância SOP. Por definição, não é possível armazenar somente a imagem, que é apenas uma parte da informação. Um arquivo DICOM é formado por um cabeçalho (*Dicom File Meta Information*) seguido do corpo do arquivo. O cabeçalho é formado por três partes contendo informações textuais ligadas à imagem. A primeira parte é um preâmbulo de 128 bytes, que não possui uso definido e pode ser utilizado pelas aplicações da forma que desejarem. A segunda parte é uma sequência de 4 bytes com os caracteres “DICM”. A terceira parte é um conjunto de *File Meta Elements*, contendo informações sobre o conjunto de dados que segue o cabeçalho.

O corpo do arquivo contém o *Data Set*, que agrupa o conjunto de elementos de dados relativos à instância SOP relacionada ao IOD, como na Figura 1.3.

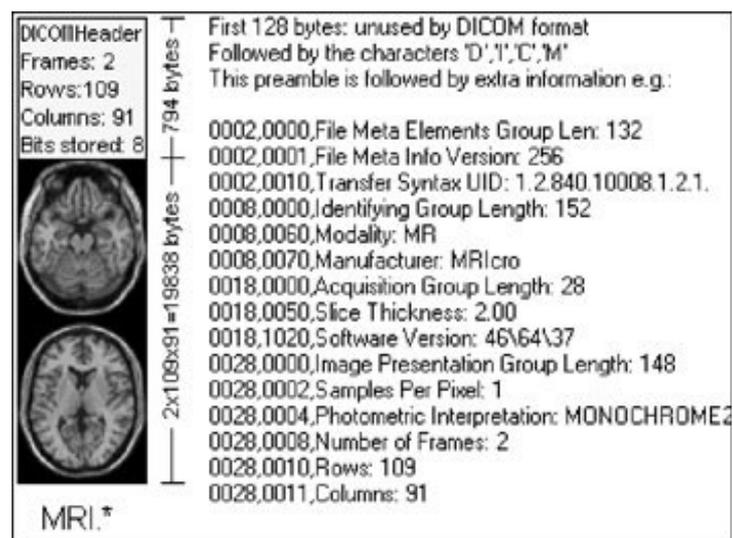


Figura 1.3: Arquivo DICOM

(fonte:[http://www.ercim.eu/publication/Ercim\\_News/enw58/acuna.html](http://www.ercim.eu/publication/Ercim_News/enw58/acuna.html))

## 1.5.2 Espaço de nomes

O conjunto de arquivos que compartilham o mesmo espaço de nomes no sistema representa o *file-set DICOM*, representado na Figura 1.4. Este *file-set* contém um arquivo especial, o DICOMDIR, implementado segundo as definições do padrão, organizando os arquivos inseridos segundo uma determinada hierarquia permitindo a recuperação e pesquisa de forma mais eficiente.

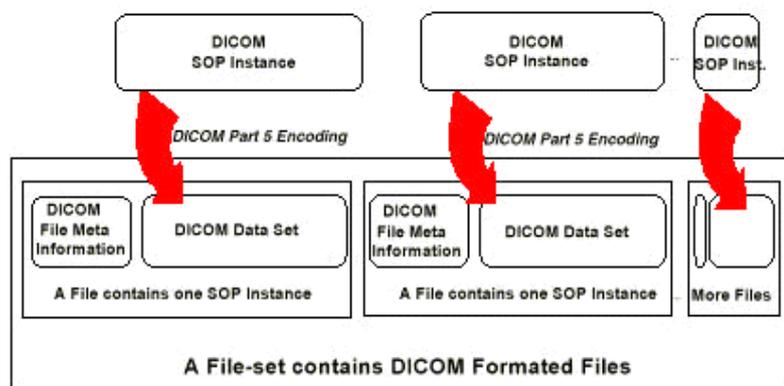


Figura 1.4: *file-set DICOM*

(fonte: [http://dicomlookup.com/html/03\\_10PU.html](http://dicomlookup.com/html/03_10PU.html))

Desta forma, cada exame é um estudo (study), e possui uma pasta equivalente no sistema de arquivos. Todas as imagens de um mesmo estudo possuem o mesmo valor na tag *Study Instance UID* (0020,000D). Alguns exames podem requer a obtenção de imagens em várias posições, ou níveis de exposição. Neste caso, o estudo é dividido em séries (Series). As imagens da mesma série possuem o mesmo valor na tag *Series Instance UID* (0020,000E), sendo representadas por um IOD. Portanto, mesmo que um objeto seja separado dos outros, é possível saber a que exame ele pertence, extraíndo do arquivo os elementos de dados que identificam a instância SOP.

## 1.6 Acesso Web a objetos persistentes DICOM

O acesso a objetos persistentes DICOM pode ser realizado através de protocolos *web*. O WADO (*Web Access to DICOM Persistent Objects*) define um serviço para acessar objetos através de requisições HTTP/HTTPS. Uma requisição utiliza o UID da instância do

objeto para referenciá-lo no servidor. Todos os elementos de dados suportados podem ser fornecidos como parâmetros em requisições *web*. A recuperação de uma região de uma imagem, convertida para JPEG2000, redimensionada, poderia produzir a seguinte requisição HTTP:

```
https://YourHospitalServer/imageaccess.js?requestType=WADO
&studyUID=1.2.250.1.59.40211.12345678.678910
&seriesUID=1.2.250.1.59.40211.789001276.14556172.67789
&objectUID=1.2.250.1.59.40211.2678810.877991027.89977.2
&contentType=image%2Fjp2;level=1,image%2Fjpeg;q=0.5
&columns=400
&rows=300
&region=0.3,0.4,0.5,0.5
&windowCenter=-1000
&windowWidth=2500
```

### **1.7 Hospedagem de aplicações (*Application Hosting*)**

Uma aplicação (*Hosting System*) fornece a infra-estrutura na qual uma segunda aplicação (*Hosted Application*) é executada e interage com o ambiente externo, de forma a oferecer serviços diversificados como acesso, segurança, e banco de dados. Um conjunto de métodos (API) é definido de forma a suportar diversas aplicações hospedadas, definindo uma interface comum (independente de tecnologia), para o controle e troca de dados do modelo DICOM (NEMA,2011). A Figura 1.5 apresenta a sequência da interação entre as aplicações em acesso a objetos persistentes.

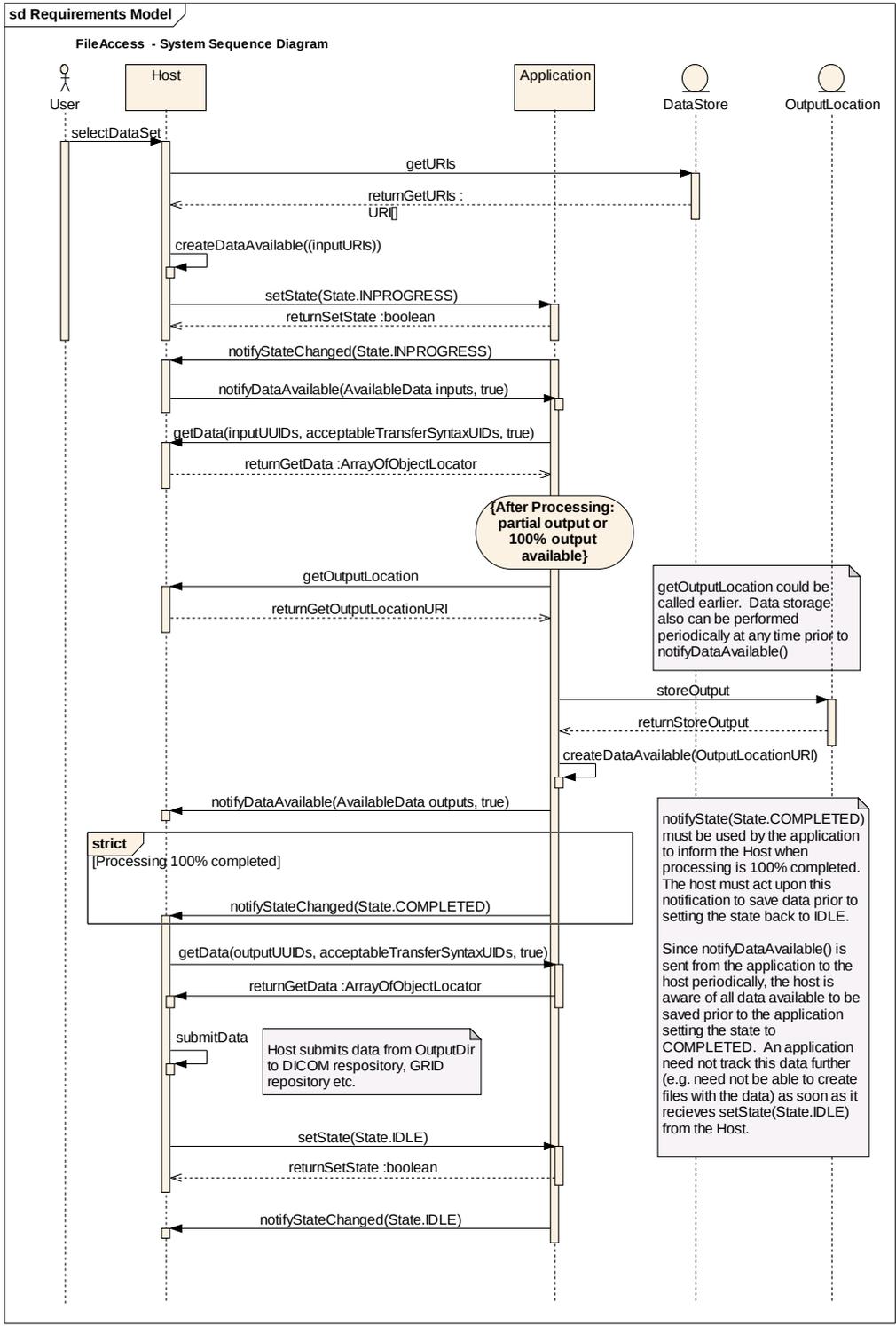


Figura 1.5: Acesso a objetos persistentes via aplicação hospedada.

(fonte: <http://medical.nema.org/standard.html>)

## **2 Os Sistemas de Armazenamento e Comunicação de Imagens Médicas Digitais (PACS)**

Segundo Huang (2010), o termo PACS (*Picture Archiving and Communication System*) se refere a um sistema de computação utilizado para capturar, armazenar, distribuir e exibir dados e imagens obtidas por modalidades de diagnóstico digitais, tais como Tomografia Computadorizada, Ressonância magnética e outros.

Esta tecnologia trouxe para a radiologia, e a medicina em geral, diversas vantagens sobre a operação convencional baseada em filme, dentre elas a redução de custos e a possibilidade de processamento para agregar valor ao diagnóstico (Huang,2010,p.387-406).

### **2.1 Radiologia digital**

O conceito de comunicação de imagens médicas digitais foi introduzido na década de 70, mas somente em 1982, após conferência internacional (*First International Conference and Workshop on Picture Archiving and Communication Systems*), em *Newport Beach*, Califórnia, este conceito, assim como o de PACS, passou a ser reconhecido de modo mais abrangente (Huang,2010,p.1-5).

A radiologia digital, segundo Siegel (2001), é possibilitada pela integração de sistemas em departamentos de radiologia formando a base para um serviço de radiologia “*filmless/paperless*” onde filme e papel são substituídos em total ou grande parte por formatos digitais.

### 2.1.1 PACS/RIS/HIS

Em uma instituição de saúde, um departamento de radiologia digital tem dois componentes: um sistema de informação em radiologia (RIS – *Radiology Information Management System*) e um Sistema de Imagem Digital (PACS). RIS é um subconjunto do Sistema de Informação Hospitalar (HIS – *Hospital Information System*) (Huang,2010,p.387-406).

Azevedo-Marques (2011) destaca que:

“a integração RIS/PACS” diminui o risco de inconsistências, através da redução do número e interfaces entre bases de dados com grande redundância de informação, proporcionando um ambiente de trabalho rápido e seguro para consulta de laudos radiológico e visualização de imagens associadas.”

Para que esta integração seja possível, são necessários padrões de comunicação bem definidos. O IHE (*Integrating the Health Care Enterprise*) estabelece um conjunto de arcabouços técnicos (*frameworks*) que define como os padrões devem ser implementados de forma a satisfazerem as necessidades da prática clínica (AZEVEDO-MARQUES,2011).

O foco do IHE é estabelecer formas otimizadas de fluxo de informação no âmbito de instituições de saúde com base em padrões de comunicação, como o DICOM e o HL7 (IHE, 2011).

O HL7 (ANSI) tem por missão definir padrões para troca, gerenciamento e integração de dados que propiciem o cuidado clínico de pacientes, além do gerenciamento, fornecimento e avaliação de serviços de saúde, tratando apenas informação textual (HL7, 2011).

### 2.2 Os Componentes de um PACS

Um PACS típico consiste de um gateway para a aquisição de dados e imagens, um controlador PACS/Servidor de armazenamento, e estações de exibição integradas por uma rede digital. Os módulos de software desta infra-estrutura incorporam o entendimento e a cooperação suficiente para permitir que os componentes trabalhem juntos de forma a compor um sistema integrado (Huang,2010,p.17).

Uma arquitetura típica de um PACS é mostrada na Figura 2.1.

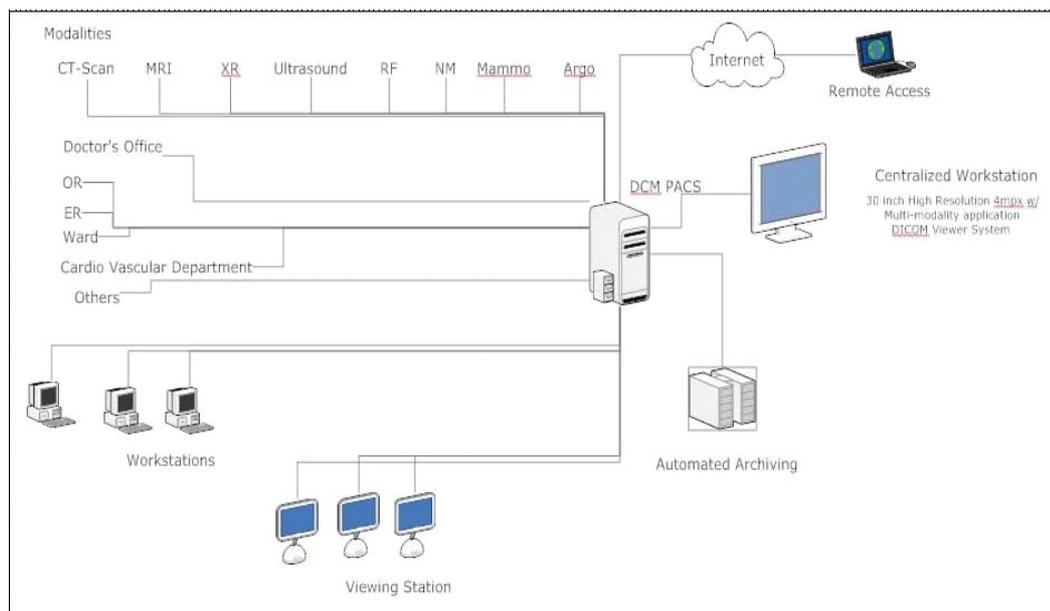


Figura 2.1: Sistema de armazenamento e distribuição de imagens médicas (PACS)  
 (fonte: [www.abfm.org.br/rbfm/publicado/rbfm\\_v3n1\\_131-9.pdf](http://www.abfm.org.br/rbfm/publicado/rbfm_v3n1_131-9.pdf))

### 2.2.1 Gateway de aquisição de imagens

É a parte do sistema responsável por obter a imagem digital a partir do exame realizado. Os dispositivos de aquisição modernos possuem computadores acoplados que, além de armazenarem localmente as imagens adquiridas, funcionam como um canal de comunicação (*gateway*) com o PACS, executando três tarefas primárias: aquisição da imagem através do equipamento radiológico, conversão dos dados ao padrão DICOM e encaminhamento do estudo que contém as imagens para o servidor PACS ou estações de trabalho (Huang,2010,p.301).

### 2.2.2 Apresentação de imagens

As estações de visualização de imagens são compostas basicamente por quatro componentes de hardware: um computador, uma placa de vídeo, monitores e dispositivos de armazenamento. Dependendo da aplicação, as características da estação podem se diferenciar. Por exemplo, uma estação diagnóstica para uso do radiologista precisa de monitores de maior resolução e contraste. O software instalado deve ser capaz de recuperar

arquivos em servidores PACS e exibir as imagens e dados recuperados, além de fornecer funções básicas de manipulação de imagens como janelamento e medidas de distâncias e ângulos (Huang,2010,p.349).

### **2.2.3 Controlador PACS e servidor de armazenamento de imagens**

O servidor do PACS é a peça fundamental de sua arquitetura e pode ser dividido em dois componentes principais: o controlador PACS e o servidor de armazenamento de imagens. O primeiro consiste em equipamentos e programas que controlam a comunicação e todo o fluxo de dados no PACS. O segundo é responsável pelo armazenamento, segurança e integridade dos dados das imagens recebidas. Em termos de hardware, o servidor de arquivamento de imagens é composto dispositivos de armazenamento e conexões de rede. Os principais dispositivos para armazenamento em um PACS incluem discos magnéticos (*Hard Disk - HD*), conjunto de discos independentes (*Redundant Array of Inexpensive Disks - RAID*), fitas magnéticas (*Digital Linear Tape - DLT*), além de CDs e DVDs. Devido à grande demanda por velocidade de acesso e confiabilidade, geralmente se utiliza RAID no PACS para armazenamento de dados que não necessitem estar disponíveis por um período de tempo muito longo. Por outro lado, fitas magnéticas são comuns para efetuar cópias de segurança, quando o armazenamento é permanente ou deve ser mantido por muitos anos (Huang,2010,p.325).

### **3 Implementação de Sistemas de Armazenamento e Comunicação de Imagens Médicas Digitais utilizando DICOM**

Um PACS é implementado utilizando o padrão DICOM como infra-estrutura de comunicação entre as aplicações que compõem o sistema. As tecnologias existentes para esta tarefa podem ser classificadas em duas categorias: proprietárias ou de software livre.

Neste capítulo é apresentada a demonstração de uma arquitetura que utiliza código livre, consistindo em um modelo que pode ser implementado a baixo custo em relação a soluções proprietárias, além de fornecer melhor adaptabilidade, pela estrutura em módulos independentes; portabilidade, por executar em diferentes sistemas operacionais; e interoperabilidade, pela comunicação entre diferentes sistemas computacionais. Neste modelo cliente-servidor, as imagens e informações associadas podem ser localizadas e recuperadas, a partir da aplicação cliente, através dos atributos (elementos de dados) que compõem o arquivo de imagem. As imagens podem ser visualizadas e editadas localmente utilizando o software ImageJ (IMAGEJ, 2011). Os arquivos enviados ao servidor são armazenados em banco de dados relacional. A comunicação com o servidor é realizada também por interface web, que permite a localização, visualização e recuperação das imagens através de requisições HTTP. As imagens podem ser recuperadas em formatos usuais de compressão como jpeg, gif e outros formatos habilitados no servidor.

#### **3.1 Materiais e métodos**

Para esta demonstração foi utilizado o servidor Conquest DICOM e uma aplicação cliente foi implementada através da construção de uma interface de usuário em linguagem

de programação Java. Conquest é um servidor DICOM desenvolvido no *Netherlands Cancer Institute*, por Marcel van Herk e Lambert Zijp, com base em um código de domínio público (UCDMC DICOM) desenvolvido inicialmente no *Medical Center of the University of Califórnia* por Mark Oskin. O servidor implementa a maioria das funcionalidades DICOM, como armazenamento e disponibilização de arquivos, além de compressão de imagens e conexões HTTP (CONQUEST, 2011).

A aplicação cliente foi construída na IDE (*Integrated Development Environment*) Eclipse 3.5.2. A IDE Eclipse é um projeto *open source* desenvolvido inicialmente pela IBM e oferece diversas facilidades aos programadores na construção do código fonte (ECLIPSE, 2011).

Foi utilizada a biblioteca Dcm4che, distribuída sob os termos da GNU, que implementa o padrão DICOM em linguagem de programação java (DCM4CHE, 2011).

Para a visualizar os arquivos de imagem foi utilizado o software ImageJ. O ImageJ é um software para processamento de imagens, desenvolvido em Java por *Wayne Rasband* no *National Institute of Mental Health, USA* (IMAGEJ, 2011).

Para os testes do sistema, foram utilizados dois computadores conectados em rede através de um roteador em uma rede ethernet cabeada: um computador Intel Core2 duo 1.66 Ghz com 2 Gb de memória RAM e sistema operacional Linux Ubuntu 10.10 e um outro computador Intel Celeron 2.4 Ghz com 4 Gb de RAM e sistema operacional Windows 7. Foram atribuídos aos mesmos os endereços IP 192.168.2.7 e 192.168.2.5, respectivamente. O servidor Conquest DICOM foi instalado no sistema Linux e a aplicação cliente no computador com sistema Windows. Posteriormente, com o intuito de avaliar o comportamento sistema em diferentes sistemas operacionais, o servidor foi instalado no computador Windows e o cliente instalado no computador Linux. Foram realizadas associações entre as aplicações para a pesquisa e transferência de arquivos.

A funcionalidade de pesquisa e recuperação é implementada utilizando o pacote dcm4che através das bibliotecas mvstudy.jar, movestudy.jar, dcmsnd.jar e dcmrcv.jar, que por sua vez implementam os DIMSE's C-FIND, C-MOVE e C-STORE. As imagens recebidas são armazenadas em um *file-set* DICOM criado com a utilização de dcmdir.jar, em subdiretórios com nomes derivados do objeto (PatientName, StudyID e SeriesNumber). O nome do arquivo será o valor do atributo InstanceNumber. Na inicialização, a aplicação passa a aguardar conexões de outras AE's (Application Entities) através da porta 104 da

máquina local, de forma a armazenar as instâncias SOP recebidas, encaminhando-as ao DICOMDIR.

### 3.2 Resultados

Inicialmente, foi realizado um acesso web a partir da máquina Windows ao servidor instalado em Linux, através do endereço <http://192.168.2.7/cgi-bin/dgate>, que retornou os exames armazenados no servidor conforme apresenta a Figura 3.1a:

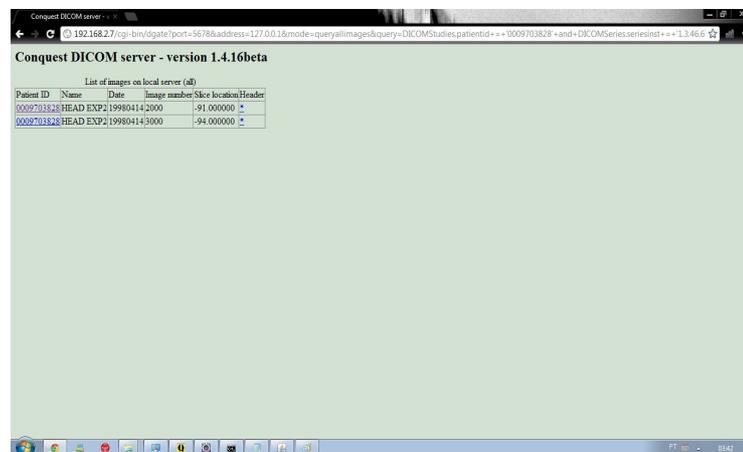


Figura 3.1a: Acesso web ao servidor Conquest DICOM

A figura 3.1b apresenta a saída no servidor Conquest após o acesso web a partir de <http://192.168.2.5>.

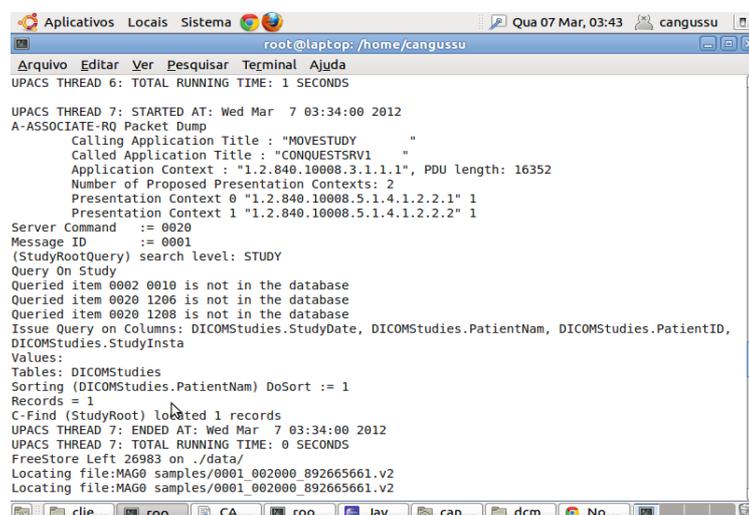


Figura 3.1b: Saída apresentada no servidor Conquest em sistema Linux após acesso web

Em seguida, utilizando a aplicação cliente, um arquivo foi recuperado do servidor para o sistema de arquivos local ao cliente. Através da aba Pesquisar/Recuperar foi informado os dados necessários à conexão: o endereço IP do servidor, a porta na qual atende a aplicação (5678), e o AETITLE (CONQUESTSRV1), que identifica o servidor na rede DICOM. A Figura 3.2 exibe a entrada dos dados para a recuperação.

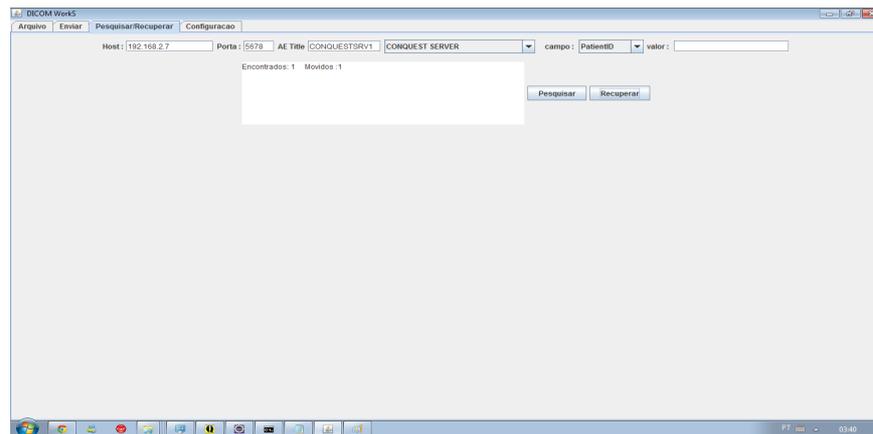


Figura 3.2: Tela da aplicação cliente para a pesquisa e recuperação de arquivos

Finalmente, com o servidor Conquest instalado no computador com sistema operacional Windows, foi enviado um arquivo a partir da aplicação cliente instalada no computador Linux. A saída apresentada no servidor é mostrada na Figura 3.3.

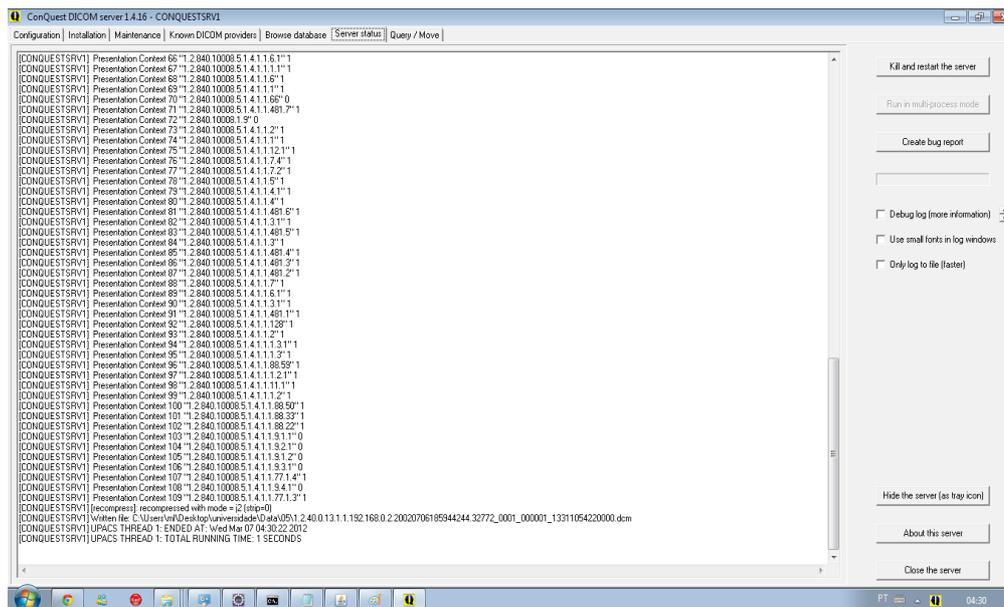


Figura 3.3: Saída apresentada no servidor Conquest em sistema Windows após acesso web

Para complementar o módulo cliente, o aplicativo ImageJ (Figura 3.4) foi utilizado para permitir a visualização das imagens e realizar algum processamento sobre elas.

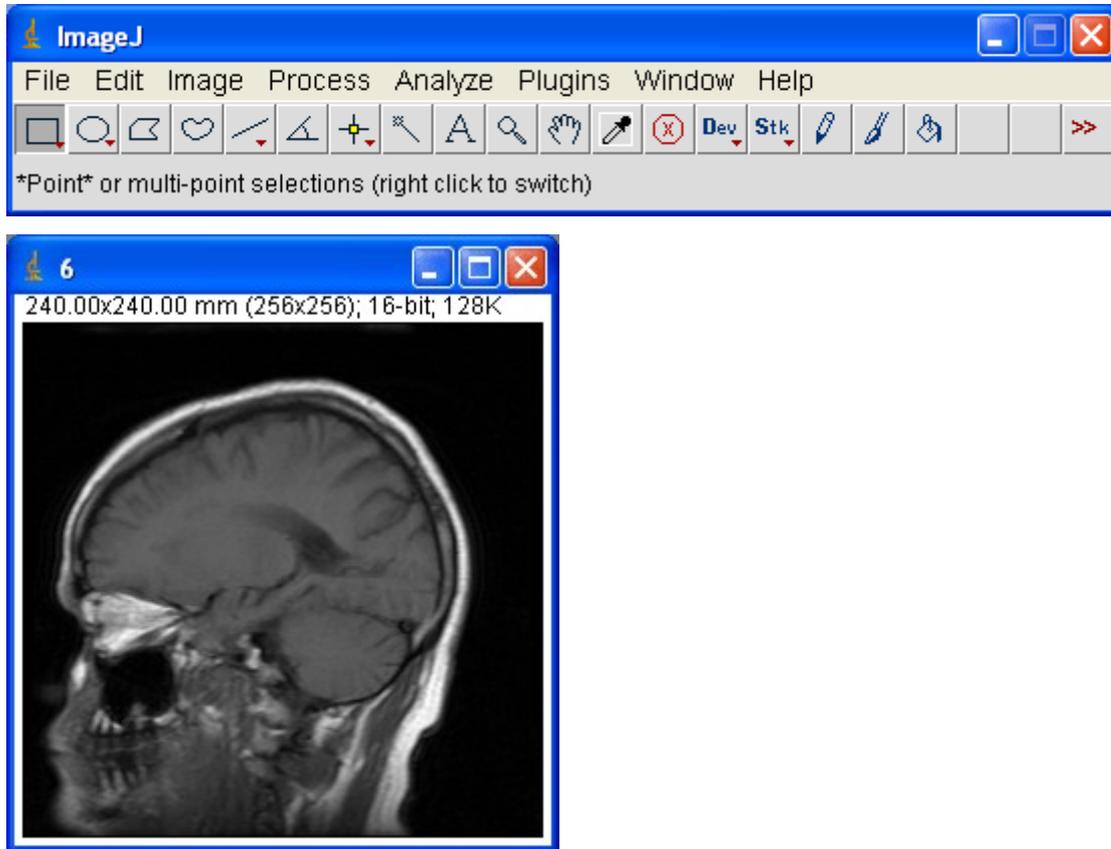


Figura 3.4: ImageJ

### 3.3 Discussões

A arquitetura implementada utilizou tecnologias de código livre, consistindo em um modelo de baixo custo, multiplataforma e com boa escalabilidade, para a implantação de um PACS.

A aplicação cliente pode ser utilizada para a pesquisa, envio, recuperação e visualização de imagens médicas digitais em servidores em conformidade com o padrão DICOM. Sua estrutura pode ser adaptada para funcionar em conjunto com modalidades de diagnóstico digital, através de uma conexão de rede e da configuração do dispositivo de forma que as imagens obtidas sejam encaminhadas para a porta 104 da estação de trabalho. Como o cliente aguarda indefinidamente por conexões com AE's que enviem instâncias

SOP *Storage*, o estudo recebido será devidamente armazenado segundo a hierarquia de pastas estabelecida no DICOMDIR, possibilitando a sua localização no sistema de arquivos local à estação, para visualização ou encaminhamento ao servidor.

O serviço WADO (*Web Access Data Persistent Objects*) define o acesso web a objetos persistentes DICOM, e é fornecido na maioria dos servidores atuais, como o Conquest. Este serviço diminui os custos de implantação dos PACS uma vez que possibilita a comunicação HTTP com um servidor através de um navegador web, geralmente presente de forma nativa na maioria dos sistemas operacionais.

## CONCLUSÃO

Este trabalho demonstrou a utilização do padrão DICOM para a implantação de Sistemas de Armazenamento e Transmissão de Imagens (PACS). Foi realizada a implementação de um modelo de sistema PACS com a utilização de ferramentas de código aberto em um ambiente cliente-servidor. Alguns objetos e serviços definidos no modelo DICOM foram utilizados através de aplicações instaladas em diferentes sistemas operacionais. O resultado demonstrou que as definições do padrão conferem às aplicações o conhecimento necessário do contexto da informação a ser compartilhada, independente de plataforma de hardware ou software. Isto possibilita a construção de um sistema integrado composto por diversos sistemas heterogêneos como dispositivos de aquisição, servidores de armazenamento, estações de trabalho locais ou remotas, entre outros.

Com os testes realizados foi possível concluir que o padrão DICOM viabiliza de maneira efetiva a comunicação no PACS, razão pela qual é adotado atualmente pela maioria dos fabricantes de equipamentos de diagnóstico por imagem. A principal contribuição deste trabalho consiste na utilização de tecnologias de código aberto proporcionando maior portabilidade através de melhorias no código ou adaptações necessárias ao comportamento esperado do sistema. O trabalho pode ser expandido em trabalhos futuros com a adição de novos componentes como um módulo de integração RIS/PACS, para aumentar a consistência dos dados dos pacientes e exames, ou um servidor CAD (*Computer Aided Diagnosis*) que utiliza o computador para a análise de imagens funcionando como uma “segunda opinião” para a tomada de decisões. A segurança nestes sistemas também é um importante tema a ser abordado, uma vez que a integração cria a necessidade de conferir diferentes níveis de privilégio aos usuários de forma a evitar acessos indevidos a determinados dados do paciente ou exame.

## REFERÊNCIAS BIBLIOGRÁFICAS

NEMA. **The DICOM standard**. Disponível em <<http://medical.nema.org/standard.html>>. Acesso em 10/01/2012.

HUANG, H.K. **PACS and imaging informatics: basic principles and applications**. Wilmington: Wiley-Liss, 2010.

PYANYKH, O. S. **Digital Imaging and Communications in Medicine (DICOM)**. Boston: Springer, 2009.

RSNA. **A nontechnical introduction to DICOM**. Disponível em: <<http://www.rsna.org/technology/dicom/intro/index.cfm>>. Acesso em: 25/04/2011.

Siegel EL, Kolodner R.M. **Filmless radiology: state of the arte and future trends**. Berlin: Springer-Verlag, 1999.

Azevedo-Marques, PM. **Integração RIS/PACS no Hospital das Clínicas de Ribeirão Preto: uma solução baseada em “web”**. Radiologia Bras. 2005.

Caritá EC, Seraphim E, Honda MO, Azevedo-Marques PM. **Implementação e avaliação de um sistema de gerenciamento de imagens médicas com suporte à recuperação baseada em conteúdo**. Radiologia Brasileira. 2008.

IMAGEJ. **ImageJ: Image Processing and Analysis in Java**. Disponível para download em <http://rsbweb.nih.gov/ij/download.html>. Acesso em 25/04/2011.

CONQUEST. **Conquest DICOM software 1.4.16 released**. Disponível para download em <http://ingenium.home.xs4all.nl/dicom.html>. Acesso em 25/04/2011.

ECLIPSE. **Eclipse IDE for Java EE Developers**. Disponível para download em <http://www.eclipse.org/downloads/packages/eclipse-ide-java-ee-developers/indigosr2>. Acesso em 25/04/2011.

DCM4CHE. **DCM4CHE: Open Source Clinical Image and Object Management**. Disponível em: <Http://www.dcm4che.org>. Acesso em: 25/04/2011.

## GLOSSÁRIO

*Application Entity*: uma aplicação DICOM em uma rede.

*Application Entity Title*: nome que identifica a aplicação na rede.

*Big-Endian*: formato onde o byte mais significativo é colocado inicialmente na memória ou enviado primeiro.

*Called AE Title*: o título da entidade de aplicação chamada, ou seja, o nome da entidade de aplicação (Application Entity) que recebeu uma requisição de associação.

*Calling AE Title*: o título da entidade de aplicação que requisita uma associação.

*Cliente*: o usuário de um serviço oferecido por algum servidor.

*Command*: o comando é uma solicitação para se realizar alguma operação em um Objeto de Informação, ou a informação a respeito de alguma mudança no estado de um Objeto de Informação.

*Command Element*: o elemento de comando é a codificação de um parâmetro de um comando.

*Command Set*: parte da mensagem DIMSE, na qual está especificada a operação a ser executada ou a informação de uma mudança de estado.

*Data Element*: o elemento de dado contém o valor codificado de um atributo de um Objeto de Informação do mundo real.

*Data Element Tag*: a etiqueta do elemento é composta de 32 bits e o identifica unicamente no conjunto.

*Data Set*: o conjunto de dados contém os valores de uma instância de um objeto de informação (*Information Object*) do mundo real. Ele é um conjunto de elementos de dados.

*Data Elements*: encapsula os valores que definirão uma determinada instância de

*DICOMDIR File*: o arquivo DICOMDIR é único e obrigatório dentro de um conjunto de arquivos (*File Set*) que contém o diretório de armazenagem da classe SOP.

*DICOM File Format*: o formato do arquivo DICOM é um meio para encapsular em um arquivo o conjunto de dados (*Data Set*) que representa uma Instância SOP de um objeto de informação DICOM (*DICOM Information Object*).

*DIMSE*: o elemento de serviço de mensagem DICOM (*DICOM Message Service Element*).

*Endian*: o endian define a ordem dos *bytes* na memória ou durante uma transferência na rede. Existem dois métodos de ordenação de valores de dados multi-byte em um computador: big-endian e little-endian. Para que duas aplicações se comuniquem corretamente, elas precisam negociar qual a ordenação dos *bytes* que será utilizada.

*General Public License*: a GPL é uma Licença Pública GNU que permite que o usuário de um software de código aberto possa alterar o código, mas não pode comercializar o que foi desenvolvido utilizando este código.

*Hypertext Transfer Protocol (HTTP)*: o protocolo de transferência de hipertexto é usado nas transações da Web. O hipertexto é o conteúdo das páginas web e o protocolo é o que permite a transferência dos dados da página.

*Information Object Definition*: a definição do objeto de informação é a definição dos atributos de uma entidade do mundo real, descrevendo as suas características.

*IOD Composto*: a definição do objeto de informação composto representa partes de informações de várias classes de objetos do mundo real.

*IOD Normalizado*: a definição do objeto de informação normalizado representa informações de uma classe simples de objetos do mundo real.

*Little-Endian*: formato onde o byte menos significativo é inicialmente colocado na memória ou enviado primeiro.

*SCP*: é uma aplicação que fornece um serviço DICOM.

*SCU*: é uma aplicação que utiliza um serviço DICOM.

*Service Class*: a classe de serviço é a descrição estruturada de um serviço suportado pelas entidades de aplicação DICOM (DICOM Application Entities) que cooperam em uma arquitetura Cliente/Servidor.

*Service Object Pair (SOP) Class*: combinação de um conjunto específico de elementos de serviço de mensagem DICOM e uma determinada definição de objeto de informação (*Information Object Definition – IOD*).

*Transfer Syntax*: a sintaxe de transferência é a descrição do método de codificação usado para um conjunto de dados (Data Set) DICOM, isto é, a definição da estrutura dos elementos de dados, da ordem dos *bytes* e do padrão de compressão da imagem.

*Unique Identifier (UID)*: string de caracteres formado por algarismos decimais e por pontos.

## APÊNDICE A – Roteiro de implementação

O roteiro de implementação a seguir foi testado no sistema operacional Linux Ubuntu 10.04.

O servidor Conquest possui também uma versão para o sistema operacional Windows.

A aplicação cliente pode ser executada em sistemas Linux e Windows devido a natureza multiplataforma conferida pela tecnologia Java.

### Aplicação servidor:

#### 1.1) Instalação:

- Baixar o servidor Conquest (conquestlinux1416.tar.gz) disponível para download em <http://ingenium.home.xs4all.nl/dicom.htm>

- Instalar dependências:

```
$sudo aptitude install mysql-server libmysqlclient-dev
```

```
$sudo aptitude install g++
```

```
$sudo aptitude install apache2
```

- Configura MySQL:

```
$mysql -u root -p
```

```
>CREATE DATABASE conquest;
```

```
>GRANT ALL ON conquest.* to <user@'hostname'> IDENTIFIED BY  
'<password>';
```

- Instalar o servidor Conquest:

```
$ tar xvf conquestlinux1416.tar.gz
```

```
$ cd jpeg-6c
```

```
$ ./configure
```

```
$ sudo make install
```

```
$ cd ../jasper-1.900.1.6ct
```

```
$ ./configure
```

```
$ sudo make install
```

```
$ cd ..
```

```
$ sudo ./maklinux_mysql
```

## 1.2) Configuração

\$ nano dicom.ini

- Alterar as configuração abaixo de acordo com o endereço do servidor

Mysql:

SQLHost = <host>

SQLServer = <database>

Username = <mysql\_user>

Password = <mysql\_password>

MAGDevice = /opt/conquest/data

- Alterar as configurações abaixo de acordo com o endereço do servidor

HTTP

WebServerFor = <server\_address>

WebScriptAddress = http://<server\_address>/cgi-bin/dgate

KfactorFile = dicom.sql

### Aplicação cliente:

- Baixar a IDE Eclipse disponível para download em <http://www.eclipse.org/downloads/packages/eclipse-ide-java-ee-developers/indigosr2>

- Importar para o ambiente a pasta do projeto ClienteDicom disponível no cd.

- Para iniciar a aplicação, execute a classe ClienteDicom.

- Para modificar a interface gráfica, utilizar os arquivos ClienteDicom.java e Painel.java.

- Para modificar as funções de pesquisa, recuperação e armazenamento de arquivos, alterar o código-fonte importado da biblioteca dcm4che, como a seguir:

DcmDir.java: criação do file-set DICOM.

DcmSnd.java: envio de arquivos.

DcmRcv.java: armazenamento de arquivos.

MoveStudy.java: pesquisa de arquivos.

Obs.: Acessar <http://dcm4che.org> para obter a versão mais recente do pacote dcm4che.

- Instalar ImageJ:

```
$ gunzip ij123-x86.tar.gz
```

```
$ tar xvf ij123-x86.tar
```

```
$ cd ImageJ
```

```
$ ./run
```