

XML による DICOM データセットの記述

二反田 慎* 金子 邦彦** 牧之内 顕文***

*九州大学大学院システム生命科学府システム生命科学専攻 〒812-8581 福岡県福岡市
東区箱崎 6-10-1

**九州大学大学院システム情報科学研究科 〒812-8581 福岡県福岡市東区箱崎 6-10-1

***久留米工業大学工学部 〒830-0052 福岡県久留米市上津町 6-10-1

E-mail: *nitanda@db.is.kyushu-u.ac.jp, **kaneko@is.kyushu-u.ac.jp,
*** akifumi@cc.kurume-it.ac.jp

あらまし 医用画像機器から得られる DICOM データは、各スライス毎に 1 データファイルであり、1 回の撮影で複数のデータファイルが得られる。DICOM データの XML 記述において、こうしたデータの XML 記述では同一の撮影データについて共通情報をまとめると共に、バイナリデータ（画像など）は ASCII コード等 XML で扱える文字コードに変換する必要がある。本研究では NCI(National Cancer Institute)が公開している DICOM データセットの XML 記述について述べる。

キーワード DICOM, XML

DICOM dataset described by XML

Makoto NITANDA* Kunihiko KANEKO** and Akifumi MAKINOUCHI***

*Graduate School of Systems Life Sciences, Kyushu University 6-10-1 Hakozaki,
Higashi-Ku Fukuoka, 812-8581 Japan

**Graduate School of Information Science and Electrical Engineering, Kyushu University
6-10-1 Hakozaki, Higashi-Ku Fukuoka, 812-8581 Japan

***Kurume Institute of Technology, 2228-66 Kamitsu-machi, Kurume-shi Fukuoka, Japan
830-0052

E-mail: *nitanda@db.is.kyushu-u.ac.jp, **kaneko@is.kyushu-u.ac.jp,
*** akifumi@cc.kurume-it.ac.jp

Abstract The DICOM data obtained from medical Imaging equipment is one data file at each slice, and two or more data files are obtained from one taking a picture. For describing XML about the DICOM data, it is necessary to bring common information together about the same taking a picture data, and to convert the binary data (image etc.) into the character-code such as ASCII code etc that can be treated with XML. The XML description in the DICOM data set that NCI (National Cancer Institute) discloses is described in this research.

Keyword DICOM, XML

1. はじめに

DICOM データは被験者等の情報、及び被験者の CT や MRI などによる撮影画像からなるバイナリデータであり、各情報は Data Element 単位

で表現されている。

DICOM データの表現に XML ドキュメントを用いると、Data Element のデータを XML ドキュメントの要素で表現を行うことで各情報の構造

が明確になる。また XML データベースも発達してきているため、DICOM の表現には有効なデータ形式である。

このような考えから DICOM データの XML 表現に関する先行研究 [1][2] が行われてきた。本研究では、先行研究では行われなかったバイナリデータ (画像など) の XML 化を行うと共に、1 回の撮影で得られた CT や MRI などの画像からなる DICOM データセットの共通属性をまとめることで、1 つの XML ドキュメントで記述する。

先行研究 [1][2] 及び本研究での XML は well-formed であり、DTD (Document Type Definition) を定めているため valid である。これにより、XML データベース [3] を用いることで X-Path などによる各情報の問い合わせや XML の管理を行うことに繋げていくことが出来る。更に Web ブラウザ上で XSL と Java アプレットを併用することにより、XML の各要素を動的に操作して閲覧することが可能になる (図 1)。

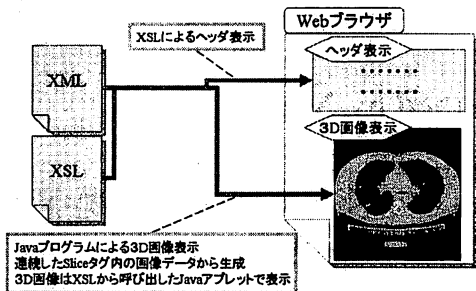


図 1. XML による、Web ブラウザ上での DICOM データ閲覧

2. DICOM の構造

DICOM データは被験者の情報や CT や MRI などによる被験者の撮影画像、その画像のフォーマット等の情報を持つ。それらの情報は Data Element 単位で区切られており、DICOM データはこれらから構成される。

Data Element は Tag, Value Representation, Value Length, Value Field の 4 つからなる。

Tag は Group Number とそれに続く Element Number の各 16bit の数の対であり、Data Element の Tag を識別することでどの種類のデータを扱っているかを特定できる。Value Representation はデータの種類を示す 2 文字の文字列で示される (但し、Data Element のうち Transfer Syntax

が "1.2.840.10008.1.2" の場合 VR は暗黙的になり、Tag の Group Value が "0008" 以降のものについては、DICOM データ内には VR は記述されずに省略されている)。Value Length は後述の Value Field のバイト列の長さを示す 16bit または 32bit の文字列である。どちらであるかは VR に依存する。被験者の情報や画像の形式、画像データ等は Value Field に格納されている。

3. XML 記述

3.1. XML 文字コード

XML の文字のエンコードは標準で UTF-8 である。UTF-8 では Unicode の 1 文字を 1~6 バイトの可変長の数値 (バイト列) に変換するようになっているが、現在定義されている Unicode 文字を UTF-8 で表現した場合、最長で 4 バイトのバイト列に変換されることになる。

3.2. Data Element と XML タグの対応

DICOM の 1 つの Data Element が XML のひとつの要素に対応する。

[1] では XML 要素のタグ名を全て "dicom_tag" で統一し、属性に tag と VR と name (規格に基づく正式名称) を設けて、Value Field をタグ値としている。一方、[2] では Data Element の種類毎にタグ名を設けて記述を行っている。

本研究では先行研究 [1][2] を踏襲し、タグ名は Data Element 内の Tag の数値の対が示す名称 (DICOM 規格に基づく) を用いることにしている。例えば Data Element にて Tag="0200,1300" の場合は XML のタグ名を "Implementation_version_name" とする。また、Value Representation と Value Length は属性とする。最後に、Value Field のバイト列を XML のタグ値とする。

これを図 2 に示す。

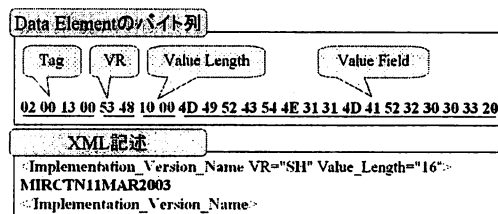


図 2. Data Element と XML タグの対応

3.3. バイナリデータの文字列変換

Value Field が画像などのバイナリデータの場合

合 UTF-8 文字コードに変換する必要がある。変換手順は以下の通りである。

1. バイナリデータの bit 列の先頭から 6bit 分取り出す
2. 最初の 6bit 分を抜き出す
3. 1bit 目が 0, 2bit 目が 1, そして 3bit~8bit 目に抜き出した 6bit 分を付け足したものを 1 文字目とする
4. バイナリデータの bit 列の 7~12bit, 13~19bit, 20~24bit 目についても同様に付け足して 2 文字目, 3 文字目, 4 文字目を生成する。但し文字コードが 0x7f になると制御コードの DEL になってしまう為, これは例外的に 0x3f(文字は"?")で置き換える。

画像の bit 列と変換後の bit 列を図 3 に示す。

表 1 は 0x00~0x7f までの文字コード表だが, 2bit 目を 0 にしたら大半が制御コードに相当してしまうので適用できない。1bit 目を 0, 2bit 目を 1 にしているのは制御コードを避け, XML のタグ内で許される文字コードのみの中で文字列変換を行うためである。従って, 変換後に出現する文字コードは表 2 の通りである。

以上で, 画像データも XML においてタグ値として扱うことが出来るようになる。

変換前は 6 bit のデータが変換後は 8 bit になるため, 変換後のバイナリデータのサイズは変換前の 4/3 倍に増加することになる。

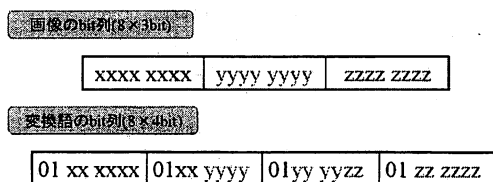


図 3. 画像の bit 列と変換後の文字列の対応

	0	1	2	3	4	5	6	7
0	NUL	DLE	SP	0	@	P	`	p
1	SOH	DC1	!	1	A	Q	a	q
2	STX	DC2	"	2	B	R	b	r
3	ETX	DC3	#	3	C	S	c	s
4	EOT	DC4	\$	4	D	T	d	t
5	ENQ	NAK	%	5	E	U	e	u
6	ACK	SYN	&	6	F	V	f	v
7	BEL	ETB	'	7	G	W	g	w
8	BS	CAN	(8	H	X	h	x
9	HT	EM)	9	I	Y	i	y
A	LF	SUB	*	:	J	Z	j	z
B	VT	ESC	+	;	K	[k	{
C	FF	IS4	,	<	L	¥	l	
D	CR	IS3	-	=	M]	m	}
E	SO	IS2	.	>	N	^	n	~
F	SI	IS1	/	?	O	_	o	DEL

表 1. 文字コード一覧

DICOM のデータ	対応する文字
0x40	@
0x41~0x5a	A~Z
0x5b~0x60	[, ¥,], ^, _ , `
0x61~0x7a	a~z
0x7a~0x7e	{, , }, ~

表 2. 変換語に現れる文字コード一覧

4. データセット単位での XML 記述

4.1. 構造

被験者の撮影を行う場合, 身体の対象部分において座標をずらしつつ撮影された画像が 1 つの DICOM データセットとなる。本研究ではこのデータセットを 1 つの XML ドキュメントに変換する。

XML ドキュメント構造は以下のように定義した。

- ルートタグを DICOM3D とする
- DICOM3D の下に Common と Slice タグを設ける
 - Common は DICOM データセット共通の情報を持つ Data Element を子とする
 - Slice は各 DICOM 固有の情報を持つ Data Element のみを子とし, 属性に

何枚目かを示す番号 No を持つ

- それぞれの Data Element のタグは各場合に
に応じて Common または Slice タグの子
として記述される

従って、1つの DICOM データは、XML では
Common 部分と Slice 部分に分割される。元の
DICOM データを復元する場合、Common の子孫
にある全ての要素と Slice の子孫にある全ての
要素をまとめる。

被験者の氏名や日付、画像のフォーマット等は
全てデータが同じであるため、これらは
Common の子孫とする。

このようにして定義した構造を図 4 に示す。
ルートの DICOM3D は Common と Slice を子と
する。Common は被験者の ID 等、内容が共通の
Data Element を子とし、各 Slice は画像データ
やスライス番号等、それぞれの DICOM データ
固有の Data Element を子とする。また、Slice が
子に持つ Data Element の要素の種類は同じである。

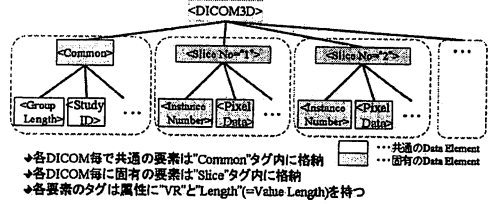


図 4. DICOM データセットの
全体的な XML 構造

4.2. 変換の流れ

DICOM データは Data Element 単位からなる
ため、バイト列を先頭から逐次読み込み、Tag,
VR, Value length, Value Field の順に以下の 1 ~
4 を行ってから出力を行うことを繰り返す流れ
をとる。

1. Tag として各 2byte ずつ読み込み、その
値からタグの名称を特定する
2. 次に VR として 2byte 読み込み、その値
をそのまま出力する。但し、Transfer
Syntax が "1.2.840.10008.1.2" の場合、Tag

Tag	VR	Common	名称	Tag	VR	Common	名称
0002 0000	UL	○	Group_Length	0018 0060	DS	○	KVP
0002 0001	OB	○	File_Meta_Information_Version	0020 000D	UI	○	Study_Instance_UID
0002 0002	UI	○	Media_Storage_SOP_Class_UID	0020 000E	UI	○	Series_Instance_UID
0002 0003	UI		Media_Storage_SOP_Instance_UID	0020 0010	SH	○	Study_ID
0002 0010	UI	○	Transfer_Syntax_UID	0020 0011	IS	○	Series_Number
0002 0012	UI	○	Implementation_Class_UID	0020 0012	IS	○	Acquisition_Number
0002 0013	SH	○	Implementation_Version_Name	0020 0013	IS		Instance_Number
0008 0008	CS	○	Image_Type	0020 0032	DS		Image_Position
0008 0016	UI	○	SOP_Class_UID	0020 0037	DS	○	Image_Orientation
0008 0018	UI		SOP_Instance_UID	0020 0052	UI	○	Frame_of_Reference_UID
0008 0020	DA	○	Study_Date	0020 1040	LO	○	Position_Reference_Indicator
0008 0023	DA	○	Content_Date	0020 1041	DS		Slice_Location
0008 0030	TM	○	Study_Time	0028 0002	US	○	Samples_per_Pixel
0008 0033	TM		Content_Time	0028 0004	CS	○	Photometric_Interpretation
0008 0050	SH	○	Accession_Number	0028 0010	US	○	Rows
0008 0060	CS	○	Modality	0028 0011	US	○	Columns
0008 0070	LO	○	Manufacturer	0028 0030	DS	○	Pixel_Spacing
0008 0090	PN	○	Referring_Physician_Name	0028 0100	US	○	Bits_Allocated
0008 1030	LO	○	Study_Description	0028 0101	US	○	Bits_Stored
0010 0010	PN	○	Patient_Name	0028 0102	US	○	High_Bit
0010 0020	LO	○	Patient_ID	0028 0103	US	○	Pixel_Representation
0010 0030	DA	○	Patient_Birth_Date	0028 1052	DS	○	Rescale_Intercept
0010 0040	CS	○	Patient_Sex	0028 1053	DS	○	Rescale_Slope
0018 0050	DS	○	Slice_Thickness	7FE0 0010	OB		Pixel_Data

表 3. 実験に用いた DICOM に存在する Data Element の種類(Tag, VR, Common の分類, 名称)

の Group Value="0008"以降は VR は省略されるため、この場合はこのステップは行わない

3. Value Length を読み込むが、これはデータ型によって 2byte か 4byte かどちらかになる。これは VR によって決まっているので、VR を読み込んだ際にあらかじめ判定を行う必要がある
4. 最後に Value Length で指定された分の byte 列を読み込み、それを Value Field として出力する
 - (ア) 文字列データの場合はそのままタグ値として出力する
 - (イ) 画像データの場合は 3.3 で述べた変換を行った後、タグ値として出力する

このようにして Data Element 毎に処理を行い、1つの Data Element 内のデータを処理し終わる度に XML ドキュメントに要素を1つずつ出力する。

上記のような処理は Common に相当する Data Element からまず行い、その後各 Slice タグの属性に定められている番号に対応した DICOM データファイルを順次読み込み、それぞれ固有の Data Element を XML ドキュメントの要素として出力する。

5. 実験

DICOM データセットは肺の画像データベースを扱っている National Cancer Institute の Lung Imaging Database Consortium で公開されている DICOM データセットを使用している[5]。このデータセットは CT スキャンにより撮影された肺の腫瘍部分に当たる DICOM データ(25~62 ファイル)を1症例のセットとし、それが計 23 症例用意されている。これらの DICOM に含まれる Data Element は表3の通りである。この表では Data Element における tag と VR と名称を示す。また、Common に丸がついているものは、4章において Common タグの子に指定されるものである。

1つの症例分を示す DICOM データセットを入力とし、1つの XML ドキュメントに出力を行った。

6. 結果

変換の結果として XML ドキュメントを付録として添付する。この XML ドキュメントは、出力結果のうち Common と Slice の1番目までを載せた、全体の一部分である。また、画像データは非常に長くなるため先頭部分のみ記述している。Well-formed であることを Web ブラウザ上(IE6)で構文チェックを行い、構文的に正しいことを確認できた。

7. まとめ

本編では DICOM データセットの XML 記述について複数の DICOM データを1つの XML ドキュメントに変換することで、画像データなどのバイナリデータも文字列変換により XML 内に埋め込んだ。

文 献

- [1] Mika J. Torhola, "Medical Imaging Data Representation with DICOM-XML", EuroPACS-MIR 2004 in the Enlarged Europe, pp.257-260, Finland
- [2] Tirado-Ramos A, Hu J, Lee KP, "Information Object Definition-based Unified Modeling Language Representation of DICOM Structured Reporting: A Case Study of Transcoding DICOM to XML", Journal of the American Medical Informatics Association Volume 9 Number 1, pp63-71, 2002
- [3] Apache Xindice
<http://xml.apache.org/xindice/>
- [4] National Electrical Manufacturers Association, "DICOM Part 5: Data Structures and Encoding", Digital Imaging and Communications in Medicine (DICOM), ps3.5-2004 edition.
- [5] National Cancer Institute, Lung Image Database Consortium (LIDC)
<http://imaging.cancer.gov/programsandresources/InformationSystems/LIDC>

