Database Design and Description for CDR DICOM

Schick Technologies, Inc.
30-00 47th Avenue
Long Island City, NY 11101

(718) 937-5765
(718) 937-5962 (fax)

PART NUMBER B1051050 REV. –

Part Number
B1051050 Rev. –

August 15, 2002

Printed in the United States of America

This document was originally prepared in English

Contents

Notice

Schick Technologies shall not be liable for errors contained herein or consequential damages in connection with the furnishing, performance, or use of this document. We reserve the right to revise this publication and to make changes to its content at any time, without obligation to notify any person or entity of such revisions and changes.

1. Introduction

1.1. Scope and Intended Audience

This document describes the content and format of database tables defined in CDR DICOM and the steps needed to incorporate these elements into Sybase and Microsoft SQL databases. The intended user of this document should be familiar with the design and function of SQL databases and their implementation in a client / server environment.

1.2. CDR DICOM and SQL Databases

The *CDRServer* application uses an SQL database to store information about patients and images and other CDR-related data objects. The database only contains information about images; the actual image file (in DICOM format) is stored separately on the server hard drive. This document describes the design of the database, including table and field descriptions as well as relationships between tables.

1.3. CDR DICOM Database Tables

The primary tables in the CDR Database are based on a simplified DICOM real world model as shown in **Figure 1**.

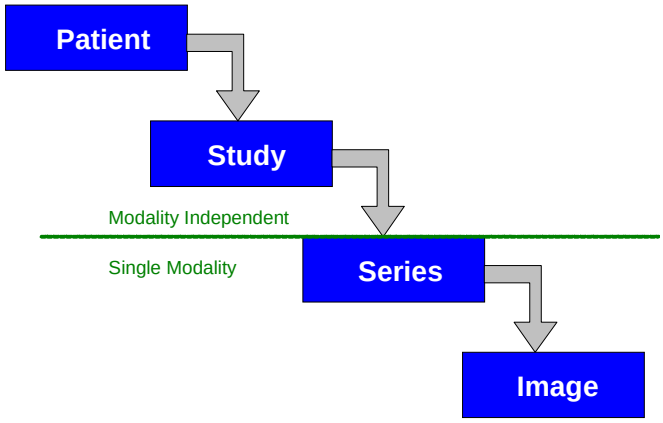

Figure 1. DICOM Real World Model

The corresponding tables in the CDR Database are **Patient**, **Study**, **Series** and **Image**. In addition, some supporting tables are **Volume**, **Workstations** and **Settings**. The **Volume** table stores information about specific storage volumes — hard drives, shared network drives, CD-ROMs, and other storage media. The **Workstations** table stores information about the workstations on the network that *CDRServer* recognizes. The **Settings** table is used to keep database version info and other useful information regarding the state of the database schema.

The fields for each of these tables are summarized on the following pages. In addition, **Figure 2** illustrates the one-to-many relationships among key fields in the tables.

1.3.1. Patient Table

Name	Type	Size	Description
PatientNumber	AutoNumber	4	Unique Identifier for Patient
NameDICOM	Text	64	DICOM formatted patient name LAST^FIRST
NameLast	Text	64	CDRPatient.LastName
NameFirst	Text	64	CDRPatient.FirstName
PatientID	Text	64	CDRPatient.IDNumber
PatientSex	Text	16	CDRPatient.Sex
BirthDate	Date/Time	8	CDRPatient.BirthDate
OtherID	Text	64	CDRPatient.OtherID
Comments	Memo	-	CDRPatient.Comments
DeleteFlag	Yes/No	1	Has patient been deleted?
EnteredDate	Date/Time	8	Date patient was entered
ModifiedDate	Date/Time	8	Date patient was last modified
PatientUID	Text	64	DICOM Unique ID for Patient

1.3.2. Study Table

Name	Type	Size	Description
StudyNumber	AutoNumber	4	Unique Identifier for Study
PatientNumber	Number (Long)	4	Link to Patient Table
StudyUID	Text	64	DICOM Unique ID of Study CDRStudy.UID
StudyDate	Date/Time	8	CDRStudy.Date
StudyID	Text	16	CDRStudy.StudyID
AccessionNumber	Text	16	CDRStudy.Accession
Description	Text	64	CDRStudy.Description
Comments	Memo	-	CDRStudy.Comments
ReferringPhysician	Text	64	CDRStudy.ReferringPhysician LAST^FIRST
ViewSetUID	Text	64	UID of Viewset Instance
ViewSetNumber	Number (Long)	4	Link to Image Table for Viewset entry
Exported	Yes/No	1	Has Study been exported?
Imported	Yes/No	1	Has Study been imported?
DeleteFlag	Yes/No	1	Has Study been deleted?
EnteredDate	Date/Time	8	Date Study was entered
ModifiedDate	Date/Time	8	Date study was last modified

1.3.3. Series Table

Name	Type	Size	Description
SeriesNumber	AutoNumber	4	Unique Identifier for Series
StudyNumber	Number (Long)	4	Link to Study Table
Modality	Text	16	CDRSeries.Modality
SeriesUID	Text	64	DICOM Unique ID of Series CDRSeries.UID
SeriesNumberDICOM	Text	12	Sequential number of Series in Study CDRSeries.Number
SeriesDate	Date/Time	8	CDRSeries.Date
PhysicianName	Text	64	CDRSeries.PhysicianName (formatted as LAST^FIRST)
OperatorName	Text	64	CDRSeries.OperatorName (formatted as LAST^FIRST)
SeriesDescription	Text	64	CDRSeries.Description
BodyPart	Text	16	CDRSeries.BodyPart
Exported	Yes/No	1	Has Series been exported?
Imported	Yes/No	1	Has Series been imported?
DeleteFlag	Yes/No	1	Has Series been deleted?
EnteredDate	Date/Time	8	Date Series was entered
ModifiedDate	Date/Time	8	Date Series was last modified

1.3.4. Image Table

Name	Type	Size	Description
ImageNumber	AutoNumber	4	Unique Identifier for Image
SeriesNumber	Number (Long)	4	Link to Series Table
ClassUID	Text	64	Class UID for DICOM Image
ImageUID	Text	64	DICOM Unique ID of Image CDRImage.UID
ImageNumberDICOM	Text	12	DICOM sequential number of image in series CDRImage.Number
ImageType	Text	255	CDRImage.ImageType
ImageFileName	Text	255	CDRImage.ImageFileName
ImageFileVolumeNumber	Number (Long)	4	Link to Volume Table for Image filename
ImageSource	Number (Long)	4	CDRImage.ImageSource
Width	Number (Long)	4	CDRImage.Width
Height	Number (Long)	4	CDRImage.Height
BitsPerPixel	Number (Long)	4	CDRImage.Depth
SensorType	Text	255	CDRImage.SensorType
SensorNumber	Text	16	CDRImage.SensorNumber
SensorMode	Text	255	CDRImage.SensorMode
MicronsPerPixel	Number (Long)	4	CDRImage.MicronsperPixel
MicronsPerPixelCalibrated	Number (Long)	4	CDRImage.MicronsPerPixelCal
ImageDate	Date/Time	8	CDRImage.ImageDate and ImageTime
AcquiredDate	Date/Time	8	CDRImage.AcqDate and AcqTime
CalibratedDate	Date/Time	8	CDRImage.CalibDate
Rotation	Number (Integer)	2	CDRImage.Rotation
HorizontalFlip	Yes/No	1	CDRImage.HorizontalFlip
Laterality	Text	16	CDRImage.Laterality
AnatomicRegion	Text	255	CDRImage.AnatomicRegion
AnatomicRegionModifier	Text	255	CDRImage.AnatomicRegionModifier
AnatomicStructure	Text	255	CDRImage.AnatomicStructure
AnatomicStructureModifier	Text	255	CDRImage.AnatomicStructureModifier
Comments	Memo	-	CDRImage.Comments
Exported	Yes/No	1	Has Image been exported?
Imported	Yes/No	1	Has Image been imported?
DeleteFlag	Yes/No	1	Has this Image been deleted?
EnteredDate	Date/Time	8	Date Image was entered
ModifiedDate	Date/Time	8	Date Image was last modified

1.3.5. Volume Table

Name	Type	Size	Description
VolumeNumber	Number (Long)	4	Unique ID of Volume
UNCPathName	Text	255	Drive Letter or UNC Share Name
CDRom	Yes/No	1	Is this volume a CD-ROM
Shared	Yes/No	1	Is this Volume shared
VolumeName	Text	255	Volume Label
VolumeSerial	Number (Long)	4	Volume serial number
FileSetID	Text	16	Name of File-Set Archive
FileSetUID	Text	64	Instance UID of File-Set Archive

1.3.6. Workstations Table

Name	Type	Size	Description
AETitle	Text	16	AE Title of workstation
IPAddress	Text	255	IP Address of workstation
Port	Text	255	Port used by workstation for listen
LocalAE	Text	16	AE Title used by server to initiate Association with workstation
MachineName	Text	255	Network machine name of workstation

1.3.7. Settings Table

Name	Type	Size	Description
Name	Text	255	Key Name of option
Type	Number (Long)	4	Data Type (0=String, 1=Number)
strValue	Text	255	String value of option
dwValue	Number (Long)	4	Numeric value of option

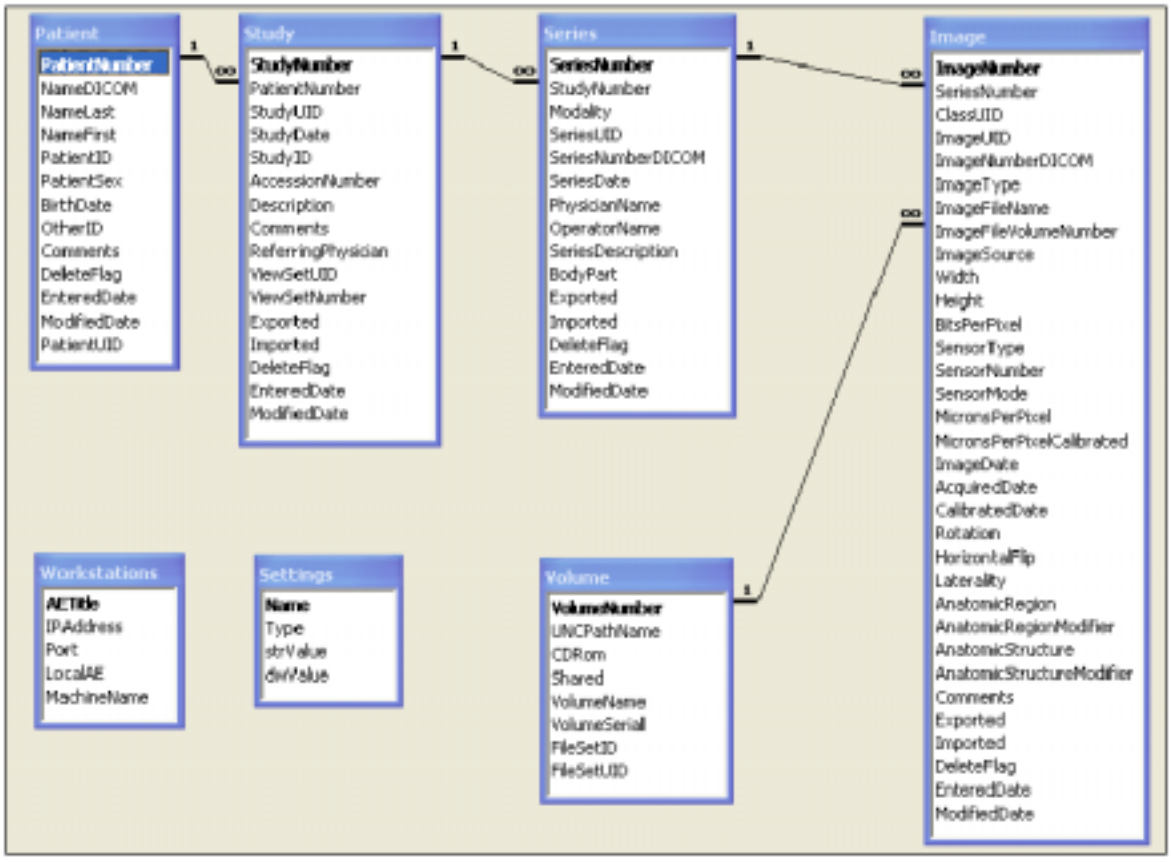

Figure 2. CDR Database Relationships

2. Step-by-Step Instructions

2.1. Setting Up Sybase SQL Anywhere for CDR DICOM

STEP 1

If you haven't done so already, please install the CDR DICOM Server application (CDRServer) before continuing with this procedure.

STEP 2

Using Sybase Central, create a new database.

STEP 3

Using Windows ODBC Administrator, create a new data source.

STEP 4

A. Browse the CDR DICOM CD and copy the Sybase script found in the ODBC folder. (The script is also supplied in **Section 3.1**.)

B. Start Interactive SQL and paste the script into the Command area. Click the **Execute** button.

STEP 5

A. In CDR DICOM Server, click **Database** > **Configure**. At the Database tab, click on Data Sources to connect the server to the SQL database.

B. Verify that your database connection is working correctly by importing patients, opening exams (after they've been added to the database), and making sure the database is able to connect to and function normally with registered workstations.

2.2. Setting Up Microsoft SQL Server for CDR DICOM

STEP 1

If you haven't done so already, please install the CDR DICOM Server application (CDRServer) before continuing with this procedure.

STEP 2

Using Enterprise Manager, create a new database.

STEP 3

Using Windows ODBC Administrator, create a new data source.

STEP 4

A. Browse the CDR DICOM CD and copy the SQL Server script found in the ODBC folder. (The script is also supplied in **Section 3.2**.)

B. Start SQL Query Analyzer and paste the script into the text area. Click the **Run** button (checkmark icon).

STEP 5

A. In CDR DICOM Server, click **Database** > **Configure**. At the Database tab, click on Data Sources to connect the server to the SQL database.

B. Verify that your database connection is working correctly by importing patients, opening exams (after they've been added to the database), and making sure the database is able to connect to and function normally with registered workstations.

3. Sample SQL Scripts

3.1. Sybase SQL Create Code

Below is a sample SQL Script to create a CDR Database generated from Sybase SQL Modeler. It includes the creation of tables, default values, indexes, and foreign keys.

```
create table Settings
(
    Name                        varchar(255)          not null,
    Type                        smallint                      ,
    strValue                    varchar(255)                  ,
    dwValue                     integer                       ,
    primary key (Name)
);

create table Workstations
(
    AETitle                     varchar(16)           not null,
    IPAddress                   varchar(255)                  ,
    Port                        varchar(255)                  ,
    LocalAE                     varchar(16)                   ,
    MachineName                 varchar(255)                  ,
    primary key (AETitle)
);

create table Volume
(
    VolumeNumber                integer               not null
        default AUTOINCREMENT,
    UNCPathName                 varchar(255)                  ,
    CDRom                       BIT                   not null
        default 0,
    Shared                      BIT                   not null
        default 0,
    VolumeName                  varchar(255)                  ,
    VolumeSerial                integer                       ,
    FileSetID                   varchar(16)                   ,
    FileSetUID                  varchar(64)                   ,
    primary key (VolumeNumber)
);

create table Patient
(
    PatientNumber               integer               not null
        default AUTOINCREMENT,
    NameDICOM                   varchar(64)           not null,
    NameLast                    varchar(64)           not null,
    NameFirst                   varchar(64)                   ,
    PatientID                   varchar(64)                   ,
    PatientSex                  varchar(16)                   ,
    BirthDate                   DATETIME                      ,
    OtherID                     varchar(64)                   ,
    Comments                    long varchar                  ,
    DeleteFlag                  BIT                   not null
        default 0,
    EnteredDate                 DATETIME              not null
        default CURRENT TIMESTAMP,
    ModifiedDate                DATETIME              not null
        default CURRENT TIMESTAMP,
    PatientUID                  varchar(64)                   ,
    primary key (PatientNumber)
);

create index PatientNameID_IK on Patient (NameLast asc, NameFirst asc, PatientID asc);

create table Study
(
```

```
    StudyNumber                 integer               not null
        default AUTOINCREMENT,
    PatientNumber               integer
        default AUTOINCREMENT,
    StudyUID                    varchar(64)                   ,
    StudyDate                   DATETIME                      ,
    StudyID                     varchar(16)                   ,
    AccessionNumber             varchar(16)                   ,
    Description                 varchar(64)                   ,
    Comments                    long varchar                  ,
    ReferringPhysician          varchar(64)                   ,
    ViewSetUID                  varchar(64)                   ,
    ViewSetNumber               integer
        default AUTOINCREMENT,
    Exported                    BIT                   not null
        default 0,
    Imported                    BIT                   not null
        default 0,
    DeleteFlag                  BIT                   not null
        default 0,
    EnteredDate                 DATETIME              not null
        default CURRENT TIMESTAMP,
    ModifiedDate                DATETIME              not null
        default CURRENT TIMESTAMP,
    primary key (StudyNumber)
);

create unique index StudyUID_IK on Study (StudyUID asc);

create table Series
(
    SeriesNumber                integer               not null
        default AUTOINCREMENT,
    StudyNumber                 integer
        default AUTOINCREMENT,
    Modality                    varchar(16)                   ,
    SeriesUID                   varchar(64)                   ,
    SeriesNumberDICOM           varchar(12)                   ,
    SeriesDate                  DATETIME                      ,
    PhysicianName               varchar(64)                   ,
    OperatorName                varchar(64)                   ,
    SeriesDescription           varchar(64)                   ,
    BodyPart                    varchar(16)                   ,
    Exported                    BIT                   not null
        default 0,
    Imported                    BIT                   not null
        default 0,
    DeleteFlag                  BIT                   not null
        default 0,
    EnteredDate                 DATETIME              not null
        default CURRENT TIMESTAMP,
    ModifiedDate                DATETIME              not null
        default CURRENT TIMESTAMP,
    primary key (SeriesNumber)
);

create unique index SeriesUID_IK on Series (SeriesUID asc);

create table Image
(
    ImageNumber                 integer               not null
        default AUTOINCREMENT,
    SeriesNumber                integer
        default AUTOINCREMENT,
    ClassUID                    varchar(64)                   ,
    ImageUID                    varchar(64)                   ,
    ImageNumberDICOM            varchar(12)                   ,
    ImageType                   varchar(255)                  ,
    ImageFileName               varchar(255)                  ,
    ImageFileVolumeNumber       integer
        default AUTOINCREMENT,
    ImageSource                 integer
              default 0,
    Width                       integer                       ,
    Height                      integer                       ,
```

```
    BitsPerPixel               integer                      ,
    SensorType                 varchar(255)                 ,
    SensorNumber               varchar(16)                  ,
    SensorMode                 varchar(255)                 ,
    MicronsPerPixel            integer                      ,
    MicronsPerPixelCalibrated  integer                      ,
    ImageDate                  DATETIME                     ,
    AcquiredDate               DATETIME                     ,
    CalibratedDate             DATETIME                     ,
    Rotation                   smallint                     ,
    HorizontalFlip             BIT                  not null
        default 0,
    Laterality                 varchar(16)                  ,
    AnatomicRegion             varchar(255)                 ,
    AnatomicRegionModifier     varchar(255)                 ,
    AnatomicStructure          varchar(255)                 ,
    AnatomicStructureModifier  varchar(255)                 ,
    Comments                   long varchar                 ,
    Exported                   BIT                  not null
        default 0,
    Imported                   BIT                  not null
        default 0,
    DeleteFlag                 BIT                  not null
        default 0,
    EnteredDate                DATETIME             not null
        default CURRENT TIMESTAMP,
    ModifiedDate               DATETIME             not null
        default CURRENT TIMESTAMP,
    primary key (ImageNumber)
);

create unique index ImageUID_IK on Image (ImageUID asc);

alter table Study
    add foreign key FK_STUDY_REF_102_PATIENT (PatientNumber)
       references Patient (PatientNumber);

alter table Series
    add foreign key FK_SERIES_REF_105_STUDY (StudyNumber)
       references Study (StudyNumber);

alter table Image
    add foreign key FK_IMAGE_REF_108_SERIES (SeriesNumber)
       references Series (SeriesNumber);

alter table Image
    add foreign key FK_IMAGE_REF_121_VOLUME (ImageFileVolumeNumber)
       references Volume (VolumeNumber);
```

3.2. Microsoft SQL Server Create Code

Below is a sample SQL Script to create a CDR Database compatible with Microsoft SQL Server. It includes the creation of tables, default values, indexes, and foreign keys.

```
create table Settings
(
    Name                    varchar(255)          not null,
    Type                    smallint                      ,
    strValue                varchar(255)                  ,
    dwValue                 integer                       ,
);

create table Workstations
(
    AETitle                 varchar(16)           not null,
    IPAddress               varchar(255)                  ,
    Port                    varchar(255)                  ,
    LocalAE                 varchar(16)                   ,
    MachineName             varchar(255)                  ,
);

create table Volume
(
    VolumeNumber            integer       IDENTITY (1,1)        not null,
    UNCPathName             varchar(255)                ,
    CDRom                   BIT                   not null
        default 0,
    Shared                  BIT                   not null
        default 0,
    VolumeName              varchar(255)                  ,
    VolumeSerial            integer                       ,
    FileSetID               varchar(16)                   ,
    FileSetUID              varchar(64)                   ,
);

create table Patient
(
    PatientNumber           integer        IDENTITY (1,1)       not null,
    NameDICOM               varchar(64)           not null,
    NameLast                varchar(64)           not null,
    NameFirst               varchar(64)                   ,
    PatientID               varchar(64)                   ,
    PatientSex              varchar(16)                   ,
    BirthDate               DATETIME                      ,
    OtherID                 varchar(64)                   ,
    Comments                text                  ,
    DeleteFlag              BIT                   not null
        default 0,
    EnteredDate             DATETIME              not null
        default CURRENT_TIMESTAMP,
    ModifiedDate            DATETIME              not null
        default CURRENT_TIMESTAMP,
    PatientUID              varchar(64)                   ,
);

create index PatientNameID_IK on Patient (NameLast, NameFirst , PatientID );

create table Study
(
    StudyNumber             integer          IDENTITY (1,1)     not null,
    PatientNumber           integer                       ,
    StudyUID                varchar(64)                   ,
    StudyDate               DATETIME                      ,
    StudyID                 varchar(16)                   ,
    AccessionNumber         varchar(16)                   ,
    Description             varchar(64)                   ,
    Comments                text                          ,
    ReferringPhysician      varchar(64)                   ,
    ViewSetUID              varchar(64)                   ,
    ViewSetNumber           integer                       ,
    Exported                BIT                   not null
```

```
        default 0,
    Imported                    BIT                   not null
        default 0,
    DeleteFlag                  BIT                   not null
        default 0,
    EnteredDate                 DATETIME              not null
        default CURRENT_TIMESTAMP,
    ModifiedDate                DATETIME              not null
        default CURRENT_TIMESTAMP,
);

create unique index StudyUID_IK on Study (StudyUID asc);

create table Series
(
    SeriesNumber                integer         IDENTITY (1,1)      not null,
    StudyNumber                 integer                         ,
    Modality                    varchar(16)                     ,
    SeriesUID                   varchar(64)                     ,
    SeriesNumberDICOM           varchar(12)                     ,
    SeriesDate                  DATETIME                        ,
    PhysicianName               varchar(64)                     ,
    OperatorName                varchar(64)                     ,
    SeriesDescription           varchar(64)                     ,
    BodyPart                    varchar(16)                     ,
    Exported                    BIT                   not null
        default 0,
    Imported                    BIT                   not null
        default 0,
    DeleteFlag                  BIT                   not null
        default 0,
    EnteredDate                 DATETIME              not null
        default CURRENT_TIMESTAMP,
    ModifiedDate                DATETIME              not null
        default CURRENT_TIMESTAMP,
);

create unique index SeriesUID_IK on Series (SeriesUID asc);

create table Image
(
    ImageNumber                 integer          IDENTITY (1,1)     not null,
    SeriesNumber                integer                         ,
    ClassUID                    varchar(64)                     ,
    ImageUID                    varchar(64)                     ,
    ImageNumberDICOM            varchar(12)                     ,
    ImageType                   varchar(255)                    ,
    ImageFileName               varchar(255)                    ,
    ImageFileVolumeNumber       integer                         ,
    ImageSource                 integer                         ,
    Width                       integer                         ,
    Height                      integer                         ,
    BitsPerPixel                integer                         ,
    SensorType                  varchar(255)                    ,
    SensorNumber                varchar(16)                     ,
    SensorMode                  varchar(255)                    ,
    MicronsPerPixel             integer                         ,
    MicronsPerPixelCalibrated   integer                         ,
    ImageDate                   DATETIME                        ,
    AcquiredDate                DATETIME                        ,
    CalibratedDate              DATETIME                        ,
    Rotation                    smallint                        ,
    HorizontalFlip              BIT                   not null
        default 0,
    Laterality                  varchar(16)                     ,
    AnatomicRegion              varchar(255)                    ,
    AnatomicRegionModifier      varchar(255)                    ,
    AnatomicStructure           varchar(255)                    ,
    AnatomicStructureModifier   varchar(255)                    ,
    Comments                    text                  ,
    Exported                    BIT                   not null
        default 0,
    Imported                    BIT                   not null
        default 0,
    DeleteFlag                  BIT                   not null
```

```
        default 0,
    EnteredDate                DATETIME              not null
        default CURRENT_TIMESTAMP,
    ModifiedDate               DATETIME              not null
        default CURRENT_TIMESTAMP,
);

create unique index ImageUID_IK on Image (ImageUID asc);

ALTER TABLE [dbo].[Image] WITH NOCHECK ADD
        CONSTRAINT [PK_Image] PRIMARY KEY  NONCLUSTERED
        (
                [ImageNumber]
        )  ON [PRIMARY] ,
        CONSTRAINT [IX_Image] UNIQUE  NONCLUSTERED
        (
                [ImageUID]
        )  ON [PRIMARY]
GO

ALTER TABLE [dbo].[Patient] WITH NOCHECK ADD
        CONSTRAINT [PK_Patient] PRIMARY KEY  NONCLUSTERED
        (
                [PatientNumber]
        )  ON [PRIMARY]
GO

ALTER TABLE [dbo].[Series] WITH NOCHECK ADD
        CONSTRAINT [PK_Series] PRIMARY KEY  NONCLUSTERED
        (
                [SeriesNumber]
        )  ON [PRIMARY] ,
        CONSTRAINT [IX_Series] UNIQUE  NONCLUSTERED
        (
                [SeriesUID]
        )  ON [PRIMARY]
GO

ALTER TABLE [dbo].[Study] WITH NOCHECK ADD
        CONSTRAINT [PK_Study] PRIMARY KEY  NONCLUSTERED
        (
                [StudyNumber]
        )  ON [PRIMARY] ,
        CONSTRAINT [IX_Study] UNIQUE  NONCLUSTERED
        (
                [StudyUID]
        )  ON [PRIMARY]
GO

ALTER TABLE [dbo].[Volume] WITH NOCHECK ADD
        CONSTRAINT [PK_Volume] PRIMARY KEY  NONCLUSTERED
        (
                [VolumeNumber]
        )  ON [PRIMARY]
GO

ALTER TABLE [dbo].[Workstations] WITH NOCHECK ADD
        CONSTRAINT [PK_Workstations] PRIMARY KEY  NONCLUSTERED
        (
                [AETitle]
        )  ON [PRIMARY]
GO

ALTER TABLE [dbo].[Settings] WITH NOCHECK ADD
        CONSTRAINT [PK_Settings] PRIMARY KEY  NONCLUSTERED
        (
                [Name]
        )  ON [PRIMARY]
GO

GRANT  SELECT ,  INSERT ,  UPDATE  ON [dbo].[Image]  TO [public]
GO

GRANT  SELECT ,  INSERT ,  UPDATE  ON [dbo].[Patient]  TO [public]
GO
```

```
GRANT  SELECT ,  INSERT ,  UPDATE  ON [dbo].[Series]  TO [public]
GO

GRANT  SELECT ,  INSERT ,  UPDATE  ON [dbo].[Study]  TO [public]
GO

GRANT  SELECT ,  INSERT ,  UPDATE  ON [dbo].[Volume]  TO [public]
GO

GRANT  SELECT ,  INSERT ,  UPDATE, DELETE  ON [dbo].[Workstations]  TO [public]
GO

GRANT  SELECT ON [dbo].[Settings]  TO [public]
GO

ALTER TABLE [dbo].[Image] ADD
        CONSTRAINT [FK_Image_Series] FOREIGN KEY
        (
                [SeriesNumber]
        ) REFERENCES [dbo].[Series] (
                [SeriesNumber]
        )
GO

ALTER TABLE [dbo].[Image] ADD
        CONSTRAINT [FK_Image_Volume] FOREIGN KEY
        (
                [ImageFileVolumeNumber]
        ) REFERENCES [dbo].[Volume] (
                [VolumeNumber]
        )
GO

ALTER TABLE [dbo].[Series] ADD
        CONSTRAINT [FK_Series_Study] FOREIGN KEY
        (
                [StudyNumber]
        ) REFERENCES [dbo].[Study] (
                [StudyNumber]
        )
GO

ALTER TABLE [dbo].[Study] ADD
        CONSTRAINT [FK_Study_Patient] FOREIGN KEY
        (
                [PatientNumber]
        ) REFERENCES [dbo].[Patient] (
                [PatientNumber]
        )

GO
```