

Parks Medical Electronics, Inc.
SonovaE Software DICOM Conformance Statement

1 Table of Contents

2	Introduction	4
2.1	Revision History	4
2.2	DICOM and SonovaE Software.....	4
2.3	Definitions, terms, and abbreviations	4
3	Implementation Model.....	6
3.1	Application Data Flow Diagram.....	6
3.2	Functional Definition of SonovaE Application Entity.....	6
3.3	Verification of Remote Real-World Activities	6
3.4	Remote Real-World Activities.....	7
3.5	Sequencing of Remote Real World Activities:.....	7
3.6	Support of Extended Character Sets.....	7
3.7	Codes and Controlled Terminologies	7
4	Application Entity Specifications.....	8
4.1	Association Establishment Policies	8
4.2	Number of Associations	8
4.3	Asynchronous Nature.....	8
4.4	Implementation Identifying Information	8
4.5	Association Acceptance Policy.....	9
4.6	Application Entity Title.....	9
4.7	Required and optionally applied Attributes	9
4.8	Structured Reporting.....	10
5	Communication Profiles	11
5.1	Supported Configuration Stacks (parts 8, 9)	11
5.2	TCP/IP Stack.....	11
5.3	Physical Media Support	11
6	Configuration.....	12
6.1	SCP Server File Storage Type	12
6.2	SCP Server AE Title / Presentation Address Mapping.....	12
6.3	DICOM SCP Configuration - Use	12

6.4	DNS Name	12
6.5	IP Address.....	12
6.6	Port	12
6.7	SCP AE Title	12
6.8	SonovaE AE Title	12
6.9	Local/Network Drive File Storage Type	13
6.10	Local/Network File Path	13
6.11	Change Path Dialog.....	13
6.12	Root UID Prefix.....	13
7	99VIP DICOM Structured Reporting Format Specification	14
7.1	Master Container.....	14
7.2	Questions Container	14
7.3	Test Container.....	14
7.4	Site Container	14
7.5	Pressure Container	15
7.6	Wave Container	15
8	99VIP Tags	16
9	SRT Location Tags	22

2 Introduction

Document Responsibility: Software Engineering

Document Used By: Sale & Marketing, Customer Service, Customers, DICOM Vendors

This document describes the conformance to the DICOM 3.0 standard for the FloLab vascular data acquisition and reporting systems from Parks Medical Electronics, Inc.

It shall establish the conformance specifications for the FloLab system only, and does not apply to other products offered by Parks Medical Electronics, Inc.

2.1 Revision History

Version	Date	By	Description
3.0	2005-01-07		Original
4.0	2008-03-31	Chad	SonovaE 5.4
5.5	2013-04-15	Gus	Update SR
5.6	2014-01-29	John	Update SR

2.2 DICOM and SonovaE Software

The DICOM standard provides a well-defined set of structures and protocols that allow inter-operability of a wide variety of medical imaging devices.

SonovaE Software provides support for connectivity to DICOM service class providers (SCP's). SonovaE Software will not support all features supported by the DICOM standard. This document clearly states the DICOM services and SOP classes that are supported by the SonovaE Software application included with the FloLab systems.

This document is written with respect to the adopted portions of the DICOM standard, Revision 3. The following sections of this document follow the outline specified in the DICOM Standard NEMA publication PS3.2.

SonovaE Software supports the Ultrasound Image Storage SOP as a Service Class User (SCU).

2.3 Definitions, terms, and abbreviations

AE **A**pplication **E**ntity The program (SonovaE) which implements DICOM.

DICOM The **D**igital **I**maging and **C**ommunications in **M**edicine standard.

DICOM files on the media.

DIMSE **D**ICOM **M**essage **S**ervice **E**lement

media the storage media hard disk

NEMA **N**ational **E**quipment **M**anufacturer's **A**ssociation

PDU **P**rotocol **D**ata **U**nit

RWA **R**eal **W**orld **A**ctivity

SCP **S**ervice **C**lass **P**rovider

SCU **S**ervice **C**lass **U**ser

SOP **S**ervice **O**bject **P**air

TCP/IP **T**ransmission **C**ontrol **P**rotocol/**I**nternet **P**rotocol

UID **U**nique **I**dentifier

3 Implementation Model

The Implementation Model identifies the SonovaE DICOM Application Entity (AE) and its relation to the Real World Activities (RWAs).

This statement covers the storage of DICOM images to networked DICOM Storage SCP devices, and local or network accessible hard disks, as specified by DICOM 3.0 specification, parts 3, 4, 5, 6, 10, 11, and 12.

3.1 Application Data Flow Diagram

The diagram in figure 1 represents the relationship between the Sonova Application Entity (rectangle left) and a remote DICOM storage SCU AE (circle right).

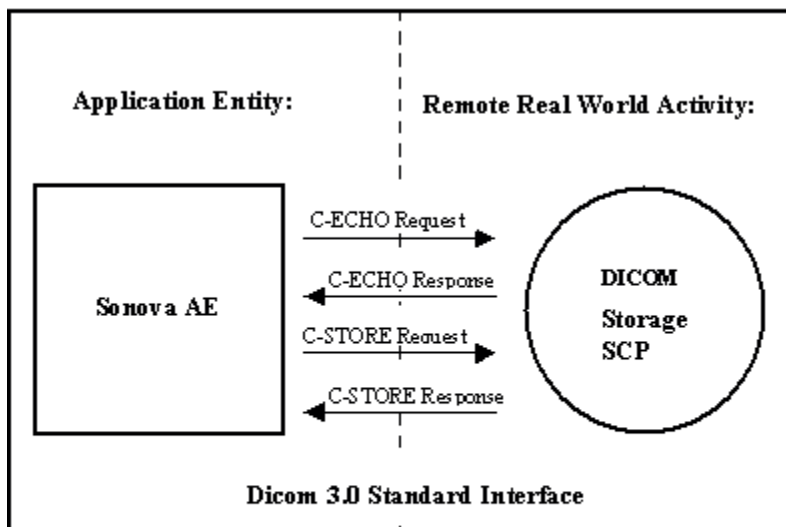


Figure 1 – Implementation Model

3.2 Functional Definition of SonovaE Application Entity

- Create DICOM Part 10 file (US-MF IOD) images from native report format.
- Verify remote SCP server and its status.
- Send DICOM image files to remote DICOM SCP.

3.3 Verification of Remote Real-World Activities

SonovaE AE performs the Verification Service Class as SCU to determine the ability of an application on a remote node to receive DICOM messages (C-ECHO DIMSE).

The SonovaE AE initiates an association to the Store SCP server and verifies its on-line status when the user selects “DICOM” and “Send (F12)”.

3.4 Remote Real-World Activities

SonovaE AE performs all of the functions to transmit images and associated data to DICOM SCPs (network servers and/or workstations). The SonovaE AE supports the Ultrasound Image Storage SOP class as an SCU.

3.5 Sequencing of Remote Real World Activities:

In order for remote processes to be able to provide the Real World Activity SCP services, the appropriate association must first be successfully opened. SonovaE AE initiates an association to the Store SCP server when the user selects “DICOM” and “Send (F12)”. If an association cannot be opened then the “Send (F12)” process will be aborted.

Once an association has been made (post “Send (F12)” initiation), SonovaE AE will create a DICOM part 10 cine image file from the native report format and transfer the DICOM file to the remote DICOM SCP (C-STORE). Once the transfer completes the (SonovaE AE - remote SCU AE) association will be closed.

3.6 Support of Extended Character Sets

The ISO IR 100 specific Character Set is supported.

3.7 Codes and Controlled Terminologies

The SonovaE SCU AE does not support any addition Codes or Controlled Terminologies.

4 Application Entity Specifications

The SonovaE AE provide Standard **Conformance** to the following **DICOM V3.0** SOP Classes as an SCU:

Information Object Definition	SOP Class UID	Transfer Syntax	Transfer Syntax UID
Ultrasound Image Storage	1.2.840.10008.5.1.4.1.1.6.1	Implicit VR Little Endian Uncompressed	1.2.840.10008.1.2
		Explicit VR Little Endian Uncompressed	1.2.840.10008.1.2.1
		JPEG Lossless	1.2.840.10008.1.2.4.57
Structured Reporting	1.2.840.10008.5.1.4.1.1.88.33		

Table 3.0-1: Supported SOP Classes

4.1 Association Establishment Policies

The SonovaE AE uses TCP/IP. The Maximum Length PDU negotiation is included in all association establishment requests. The maximum length PDU offered for an association initiated by SonovaE AE is: 64k

4.2 Number of Associations

The maximum number of simultaneous associations for the SonovaE AE is one. A separate association will be negotiated for each image transferred.

4.3 Asynchronous Nature

Asynchronous operations are not supported.

4.4 Implementation Identifying Information

Implementation Class UID: "1.2.826.0.1.3680043.2.827"

Implementation Version Name: "SONOVA V3"

4.5 Association Acceptance Policy

The SonovaE AE does not accept associations.

4.6 Application Entity Title

The default AE Title used is “SONOVA” but this can be modified in the DICOM Configuration Dialog.

4.7 Required and optionally applied Attributes

The following table denotes the attributes included in the Ultrasound Image

Object as implemented in the SonovaE AE. Attributes not listed are not used.

Tag	Attribute	Notes
0008,0000	Identifying Group Length	
0008,0008	Image Type	Set to “ORIGINAL\PRIMARY”.
0008,0012	Instance Creation Date	Set to date file was created
0008,0013	Instance Creation Time	Set to time file was created
0008,0014	Instance Creator UID	Unique Id (SonovaE assigned)
0008,0016	SOP Class UID	Set to US Image
0008,0018	SOP Instance UID	System assigned
0008,0020	Study Date	Set to the date that the study
0008,0023	Image Date	Set to the date the image file was
0008,0030	Study Time	Set to the time that the study
0008,0033	Image Time	Set to the time the image was
0008,0050	Accession Number	User assigned
0008,0060	Modality	Set to ‘US’
0008,0064	Conversion Type	
0008,0070	Manufacturer	Set to “Parks Medical Electronics,
0008,0080	Institution Name	Default is zero length
0008,0081	Institution Address	Default is zero length
0008,0090	Referring Physician’s Name	Default is zero length
0008,1010	Station Name	Not used at this time – zero length
0008,1030	Study Description	Set to SonovaE study type + study
0008,1040	Institutional Department Name	Not used at this time – zero length
0008,1060	Name of Physician(s) Reading	Default is zero length
0008,1070	Operators Name	Default is zero length
0008,1090	Manufacturer’s Model Name	Set to Simulation or FloLab 2100SX
0010,0000	Patient Group Length	
0010,0010	Patient’s Name	This field must be filled in by the
0010,0020	Patient ID	Set to unique system assigned
0010,0021	Issuer of Patient ID	Issuing authority institution name
0010,0030	Patient’s Birth Date	Default is zero length
0010,0040	Patient’s Sex	Default is zero length
0010,1010	Patient’s Age	Default is zero length
0018,0000	Acquisition Group Length	

0018,1000	Device Serial Number	Default is zero length
0018,1020	Software Version	Set to the SonovaE AE version
0018,1023	Digital Image Format Acquired	
0020,0000	Image Group Length	
0020,000D	Study Instance UID	System assigned
0020,000E	Series Instance UID	System assigned
0020,0010	Study ID	Set to unique system assigned
0020,0011	Series Number	System assigned
0020,0013	Image Number	
0020,0020	Patient Orientation	
0028,0000	Image Representation Group	
0028,0002	Samples Per Pixel	
0028,0004	Photometric Interpretation	Set to RGB
0028,0006	Planar Configuration	
0028,0010	Rows	
0028,0011	Columns	
0028,0100	Bits Allocated	
0028,0101	Bits Stored	
0028,0102	High Bit	
0028,0103	Pixel Representation	
0028,1050	Window Center	
0028,1051	Window Width	
0032,0000	Study Group Length	
0032,4000	Study Comments	default is zero length
7fe0,0000	Pixel Data Group Length	
7fe0,0010	Pixel Data	

4.8 Structured Reporting

SOP Class Name: Comprehensive SR Storage

SOP Class ID: 1.2.840.10008.5.1.4.1.1.88.33

5 Communication Profiles

5.1 Supported Configuration Stacks (parts 8, 9)

The SonovaE AE provides DICOM V3.0 TCP/IP Network Communication Support as defined in PS 3.8.

5.2 TCP/IP Stack

The SonovaE AE inherits the TCP/IP stack from the Windows operating system upon which it executes. Port number 104 is used as a default for DICOM communication but can be configured to use any port.

5.3 Physical Media Support

The SonovaE SCU is indifferent to the physical medium over which TCP/IP executes; it inherits this from the Windows operating system upon which it executes.

The FloLab system hardware supports IEEE 802.3 - 10BaseT (“twisted pair”).

6 Configuration

SonovaE AE can be configured to send DICOM images to a DICOM SCP store or send images for storage to a local or network available hard disk. The user can configure the storage type by selecting the menu choice “Utilities”, then “System Settings. . .”, and then “DICOMMWL Setup”. The default storage type is “Local/Network Drive”.

6.1 SCP Server File Storage Type

When the user selects file storage “SCP Server” they need to provide additional configuration settings for SonovaE AE to communicate with a remote DICOM SCP (refer to figure 1).

6.2 SCP Server AE Title / Presentation Address Mapping

The SonovaE DICOM Configuration is used for mapping of an Application Entity Title to a Presentation Address. A Presentation Address consists of a host name or IP address and port number. The information obtained in the SonovaE DICOM Configuration Dialog is used to establish associations to remote Application Entities.

6.3 DICOM SCP Configuration - Use

These radio buttons determine whether DHCP (Dynamic Host Configuration Protocol) is used to configure the network IP address, or a user specified IP address is used.

6.4 DNS Name

The DNS Name field is used to configure the remote computer name for the system that SonovaE AE will be sending DICOM images to.

6.5 IP Address

The IP Address field is used to configure the network IP address for the system that SonovaE AE will be sending DICOM images to.

6.6 Port

This field is used to configure the port number that the remote DICOM Store SCP application will be listening to for messages from SonovaE AE. The factory default is “104”.

6.7 SCP AE Title

The SCP AE Title field is used to configure the Application Entity title for the remote DICOM Store SCP application that SonovaE AE will be sending images to.

6.8 SonovaE AE Title

The SonovaE AE Title field is used to configure the Application Entity title used by SonovaE when requesting services from an SCP.

6.9 Local/Network Drive File Storage Type

When the user selects file storage “Local/Network” they need to provide drive and file path for SonovaE AE (refer to figure 1).

6.10 Local/Network File Path

The user can select the drive and file path using the “Change Path” button.

6.11 Change Path Dialog

The user can use this dialog to provide an available drive and path to where the images will be stored.

6.12 Root UID Prefix

The UIDRoot prefix field is used to change the unique ID Root prefix that is used to create file names for images.

If this edit is blank the SonovaE AE will use the system assigned unique UIDRoot.

7 99VIP DICOM Structured Reporting Format Specification

7.1 Master Container

CONTAINER: (LN, 18760-9, "Ultrasound Report")
TEXT: (99VIP, GUID, "Peripheral Vascular Report") =
"{98D8C047-5872-4F0B-9796-C7EE037B898B}"
CODE: (99VIP, type, "report type") = CODE: ([studyL, studyU, studyV, studyO],99VIP,[studyL]"")
TEXT: (99VIP, name, "report name") = data
CONTAINER: (DCM, 121118, "Patient Characteristics")
TEXT: (99VIP, notes, "Notes Patient") = data
CONTAINER: (DCM, 121060, "History")
TEXT: (99VIP, notes, "Notes History") = data
CONTAINER: (99VIP, questions, "Questions")
CONTAINER: (DCM, 121111, "Summary")
TEXT: (99VIP, notes, "Notes Summary") = data
CONTAINER: (DCM, 121070, "Findings")
*CONTAINER: (99VIP, test, "Test")

7.2 Questions Container

CONTAINER: (99VIP, questions, "Questions")
*CONTAINER: (99VIP, question, "Question")
TEXT: (99VIP, prompt, "Prompt") = prompt
TEXT: (99VIP, answer, "Answer") = answer

7.3 Test Container

CONTAINER:(,,"Tests")=SEPARATE>
CONTAINER: (99VIP, test, "Test") CODE:
(99VIP, type, "test type") = Code: data
TEXT: (99VIP, name, "test name") = data
TEXT: (99VIP, notes, "Notes {test name}") = data

7.4 Site Container

<contains CONTAINER:(,,"Findings")=SEPARATE>
CODE: (SRT, G-C0E3, "Finding Site") = Code:
CODE: (SRT, G-C171, "Laterality") = Code:
<contains CONTAINER:(,,"Artery Location")=SEPARATE>
TEXT: (99VIP, text, "Text") = Systolic Blood Pressure
TEXT:(,,"mmHg")="mmHg">
TEXT: (99VIP, grade, "Grade") = user entered grade
NUM: (99VIP, temperature, "Temperature") = Code: (UCUM, Cel or degF, "degrees")
CONTAINER: (99VIP, wave, "wave")

7.5 Wave Container

CONTAINER: (99VIP, wave, "wave")

NUM: (99VIP, "WaveSampleRate") = Code: (UCUM, Hz, "Hertz")

NUM: (99VIP, "WaveSweepSpeed") = Code: (UCUM, mm/s, "millimeter per second")

NUM: (99VIP, "WavePosition",) = Code (UCUM, mm, "millimeter")

NUM: (99VIP, "WaveSizePercent",) = Code: (UCUM, %, "percent")

NUM: (99VIP, "WaveScale50mm",) = Code: UCUM, mm, "millimeter")

TEXT: (99VIP, "WavePoints") = t0,t1,...,tN

8 99VIP Tags

```
public class CCodedEntryDictionaryVIP
{
    /// <summary>
    /// Set this to a 99* name.
    /// id of this coding scheme.
    /// Historically VIP started it all in 1985.
    /// </summary>
    public const string CodingSchemeDesignator = "99VIP";

    /// <summary>
    /// Responsible Organization for CodingSchemeDesignator.
    /// Set this to who is responsible for the codes.
    /// </summary>
    public const string ResponsibleOrganization = "ParksMed.com";

    /// <summary>
    /// USI SR js 8-20-13
    /// Generic code to give a "patients name"
    /// </summary>
    public const string CodeValuePatientName = "Patients Name";
    /// <summary>
    /// USI SR js 8-20-13
    /// Generic code to give a "patients ID"
    /// </summary>
    public const string CodeValuePatientId = "Patients ID";
    /// <summary>
    /// USI SR js 8-20-13
    /// Generic code to give a "patients dob"
    /// </summary>
    public const string CodeValuePatientDob = "Patients DOB";
    /// <summary>
    /// USI SR js 8-20-13
    /// Generic code to give a "patients age"
    /// </summary>
    public const string CodeValuePatientAge = "Patients Age";
    /// <summary>
    /// USI SR js 8-20-13
    /// Generic code to give a "patients gender"
    /// </summary>
    public const string CodeValuePatientGender = "Patients Gender";
    /// <summary>
    /// USI SR js 8-20-13
    /// Generic code to give a "study date"
    /// </summary>
    public const string CodeValuePatientStudyDate = "Study Date";
    /// <summary>
    /// USI SR js 8-20-13
    /// Generic code to give a "Room #"
    /// </summary>
    public const string CodeValuePatientRoom = "Room #";
    /// <summary>
    /// USI SR js 8-20-13
    /// Generic code to give a "Accession #"
    /// </summary>
    public const string CodeValuePatientAccession = "Accession #";
    /// <summary>
    /// USI SR js 8-20-13
    /// Generic code to give a "Exam By"
    /// </summary>
    public const string CodeValuePatientExamBy = "Examined By";
}
```



```

/// <summary>
/// USI SR js 8-20-13
/// Generic code to give a "Read By"
/// </summary>
public const string CodeValuePatientReadBy = "Read By";
/// <summary>
/// USI SR js 8-20-13
/// Generic code to give a "Ref1 By"
/// </summary>
public const string CodeValuePatientRef1By = "Referred1 By";
/// <summary>
/// USI SR js 8-20-13
/// Generic code to give a "Ref2 By"
/// </summary>
public const string CodeValuePatientRef2By = "Referred2 By";
/// <summary>
/// USI SR js 8-20-13
/// Generic code to give a "Description"
/// </summary>
public const string CodeValuePatientDescription = "Description";

////////////////////////////////////
/// <summary>
/// Generic code to give a "name"
/// </summary>
public const string CodeValueName = "name";

/// <summary>
/// Generic code to give a "type"
/// </summary>
public const string CodeValueType = "type";

/// <summary>
/// Generic code to give a "GUID"
/// </summary>
public const string CodeValueGUID = "GUID";

/// <summary>
/// GUID used with CodeValueGUID.
/// {98D8C047-5872-4F0B-9796-C7EE037B898B}
///
/// Identify that this report is a Peripheral Vascular Study.
/// Allows for quick ID that this is the study to Parse.
/// </summary>
public const string CodeValueReportVascularGUID = "{98D8C047-5872-4F0B-9796-C7EE037B898B}";

/// <summary>
/// Used to indicate an array item.
/// </summary>
public const string CodeValueIndex = "index";

/// <summary>
/// Root study type indicator.
/// Used with (99VIP, study, "study type") = studyL
/// </summary>
public const string CodeValueStudy = "study";

/// <summary>
/// study type Lower
/// </summary>
public const string CodeValueStudyL = "studyL";

/// <summary>
/// study type Upper
/// </summary>
public const string CodeValueStudyU = "studyU";

```

```
/// <summary>
/// study type Venous
/// </summary>
public const string CodeValueStudyV = "studyV";

/// <summary>
/// study type Other
/// </summary>
public const string CodeValueStudyO = "studyO";

/// <summary>
/// generic property of parent
/// property name will be in the CodeMeaning field
/// </summary>
public const string CodeValueProperty = "property";

/// <summary>
/// notes
/// </summary>
public const string CodeValueNotes = "notes";

/// <summary>
/// questions
/// </summary>
public const string CodeValueQuestions = "questions";

/// <summary>
/// question
/// </summary>
public const string CodeValueQuestion = "question";

/// <summary>
/// prompt
/// </summary>
public const string CodeValuePrompt = "prompt";

/// <summary>
/// answer
/// </summary>
public const string CodeValueAnswer = "answer";

///// <summary>
///// tests
///// </summary>
//public const string CodeValueTests = "tests";

/// <summary>
/// test
/// </summary>
public const string CodeValueTest = "test";
```

9 SRT Location Tags

```
/// <summary>
/// Get Coded Entry for a body location.
/// </summary>
/// <param name="codeSequence"></param>
/// <param name="BodyMap"></param>
/// <returns></returns>
public static CodedEntry GetCodedEntry(TagId.eGroupElement codeSequence, eBodyMap
BodyMap)
{
    string codeValue;
    string codeMeaning;

    switch (BodyMap)
    {
        case eBodyMap.brachial:
            codeValue = "T-47160";
            codeMeaning = "Brachial Artery";
            break;
        case eBodyMap.subclavian:
            codeValue = "T-46100";
            codeMeaning = "Subclavian Artery";
            break;
        case eBodyMap.axillary:
            codeValue = "T-47100";
            codeMeaning = "Axillary Artery";
            break;
        case eBodyMap.wristUlnar:
            codeValue = "T-47200";
            codeMeaning = "Ulnar Artery";
            break;
        case eBodyMap.wristRadial:
            codeValue = "T-47300";
            codeMeaning = "Radial Artery";
            break;
        case eBodyMap.femoral:
            codeValue = "T-47400";
            codeMeaning = "Common Femoral Artery";
            break;
        case eBodyMap.femoralSuperficial:
            codeValue = "T-47403";
            codeMeaning = "Superficial Femoral Artery";
            break;
        case eBodyMap.peroneal:
            codeValue = "T-47630";
            codeMeaning = "Peroneal Artery";
            break;
        case eBodyMap.popliteal:
            codeValue = "T-47500";
            codeMeaning = "Popliteal Artery";
            break;
        case eBodyMap.tibialPosterior:
        case eBodyMap.ankleTibialPosterior:
            codeValue = "T-47600";
            codeMeaning = "Posterior Tibial Artery";
            break;
        case eBodyMap.tibialAnterior:
        case eBodyMap.ankleTibialAnterior:
```

```

codeValue = "T-47700";
codeMeaning = "Anterior Tibial Artery";
    break;
case eBodyMap.pedisDorsalis:
case eBodyMap.anklePedisDorsalis:
codeValue = "T-47741";
codeMeaning = "Dorsalis Pedis Artery";
    break;
case eBodyMap.penile:
codeValue = "T-91000";
codeMeaning = "Penile";
    break;
case eBodyMap.thighHigh:
codeValue = "T-D07C0";
codeMeaning = "High Thigh";
    break;
case eBodyMap.thighLow:
codeValue = "T-D9100";
codeMeaning = "Low Thigh";
    break;
case eBodyMap.thigh:
codeValue = "T-D9100";
codeMeaning = "Thigh";
    break;
case eBodyMap.calf:
codeValue = "T-D9440";
codeMeaning = "Calf";
    break;
case eBodyMap.ankle:
codeValue = "T-D9500";
codeMeaning = "Ankle";
    break;
case eBodyMap.metatarsal:
codeValue = "T-D9707";
codeMeaning = "Metatarsal";
    break;
case eBodyMap.digit:
codeValue = "T-D03C0";
codeMeaning = "Digit";
    break;
case eBodyMap.forearm:
codeValue = "T-D8500";
codeMeaning = "Forearm";
    break;
case eBodyMap.wrist:
codeValue = "T-D0787";
codeMeaning = "Wrist";
    break;
case eBodyMap.InternalJugularVein:
codeValue = "T-48170";
codeMeaning = "Internal Jugular Vein";
    break;
case eBodyMap.SubclavianVein:
codeValue = "T-48330";
codeMeaning = "Subclavian Vein";
    break;
case eBodyMap.AxillaryVein:
codeValue = "T-49110";
codeMeaning = "Axillary Vein";
    break;
case eBodyMap.BrachialVein:

```

```

        codeValue = "T-D";
        codeMeaning = "Brachial Vein";
        break;
    case eBodyMap.CommonFemoralVein:
        codeValue = "G-035B";
        codeMeaning = "Common Femoral Vein";
        break;
    case eBodyMap.SuperficialFemoralVein:
        codeValue = "G-035A";
        codeMeaning = "Superficial Femoral Vein";
        break;
    case eBodyMap.PoplitealVein:
        codeValue = "T-49640";
        codeMeaning = "Popliteal Vein";
        break;
    case eBodyMap.PeronealVein:
        codeValue = "T-49650";
        codeMeaning = "Peroneal Vein";
        break;
    case eBodyMap.AnteriorTibialVein:
        codeValue = "T-49630";
        codeMeaning = "Anterior Tibial Vein";
        break;
    case eBodyMap.PosteriorTibialVein:
        codeValue = "T-49620";
        codeMeaning = "Posterior Tibial Vein";
        break;
    case eBodyMap.GreatSaphenousVein:
        codeValue = "T-49530";
        codeMeaning = "Great Saphenous Vein";
        break;
    default:
        return null;
    }
}

return new CCodedEntry(codeSequence, codeValue, SRT, codeMeaning);
}

```