Digital Imaging and Communications in Medicine (DICOM)

Volume Rendering Volumetric Presentation States

Prepared by:

DICOM Standards Committee, Working Group 11, Presentation State
1300 N. 17th Street, Suite 900
Rosslyn, Virginia 22209 USA

Developed pursuant to Work Item 2008-04-C

VERSION: Working - 25 Jan 2016

Table of Contents

Scope and Field of Application

DICOM has added SOP Classes for representing Planar MPR Volumetric Presentation States (see DICOM PS 3.3 Section A.X.x1). This supplement extends the family of Volumetric Presentation States by adding two additional SOP Classes to represent Volume Rendering Volumetric Presentation States – one restricted to a single input volume and one allowing multiple input volumes.

Volume Rendering is a data visualization method where a 2D render view through volume data is created. Voxels (volume sample points) are assigned a color and an opacity (alpha), and for each XY coordinate of the render view the output pixel value is determined by accumulating the set of non-transparent voxels samples along the z-axis.

The Volume Rendering Volumetric Presentation State also provides for alpha compositing (blending) of multiple volumes and/or segmented volumes into a single volume dataset in preparation for the Volume Rendering operation.

Volume Rendering generally consists of a number of steps, many of which are parametrically specified in the Volume Rendering SOP Classes. Steps that are usually implemented by proprietary algorithms are not described in this supplement, and are implementation-specific. The processing steps are:

- Segmentation, or separating the volume data into groups that will share a particular color palette. Segmentation objects are specified as cropping inputs to the Volumetric Presentation State.
- Gradient Computation, or finding edges or boundaries between different types of tissue in the volumetric data. Gradient Computation used is an implementation decision outside the scope of the Volumetric Presentation State.
- Resampling of the volumetric data to create new samples along the imaginary ray behind each pixel in the output two-dimensional view, generally using some interpolation of the values of voxels in the neighborhood of the new sample. The interpolation method used is an implementation decision outside the scope of the Volumetric Presentation State.
- Classification of ray samples to assign a color and opacity to each sample. Classification parameters are specified in the Volumetric Presentation State.
- Shading or the application of a lighting model to ray samples indicating the effect of ambient, diffuse, and specular light on the sample. Basic shading parameters are specified in the Volumetric Presentation State.
- Compositing or the accumulation of samples on each ray into the final value of the pixel corresponding to that ray. The specific algorithms used are outside the scope of the Volumetric Presentation State.

Refer to Section A.X.x3.1 for a list of the parameters that are specified in the Volumetric Presentation State.

The process used in the creation of this standard started with the collection of clinical use cases representing a large number of modalities and interested clinical specialties. From these clinical use cases, technical requirements were identified and clearly defined. These technical requirements go on to drive the definition of the actual standard.

The result of application of a Volumetric Presentation State is not expected to be exactly reproducible on different systems. It is difficult to describe the display and render algorithms in enough detail in an interoperable manner, such that a presentation produced at a later time is indistinguishable from that of the original presentation. While Volumetric Presentation States use established DICOM concepts of grayscale and color matching (GSDF and ICC color profiles) and provides a generic description of the different types of display algorithms possible, variations in algorithm implementations within display devices are inevitable and an exact match of volume presentation on multiple devices cannot be guaranteed. Nevertheless, reasonable consistency is provided by specification of inputs, geometric descriptions of spatial views, type of processing to be used, color mapping and blending, input fusion, and many generic rendering parameters, producing what is expected to be a clinically acceptable result.

Questions and Open Issues

1. Is the existing Render Shading specification sufficient? For example, are there use cases that require multiple light sources or the specification of other detailed shading parameters?
2. Is there a use case for multiple shading models (tied to one or more specific inputs as is done with segmentation)? Is it be sufficient to set the surface characteristics for each input (i.e. related to the “shininess” parameter)?
3. Is there a conventional way of blending volume rendered and intensity projection views together (e.g. MIP is treated as 100% opaque above a certain threshold and 0% opaque below)?
4. Depth color mapping: Do we want to support a 2D color map, where the second dimension is the z-level of the sample being mapped? This would allow the render view to distinguish shallow structures from deep structures by a different color palette. If depth color mapping is supported, would it be represented by a different Render Algorithm value, plus also limited to a different SOP Class? Or just a qualifying attribute and map? Or should it just be an implementation decision to perturb the single map in hue and/or intensity for deeper structures?
5. Is there agreement that the Volume Blending transfer function in PS 3.4 X.2.2 be “B-atop-A”, or is another scheme preferable?
6. Do we need a more-limited single-input monochrome-only SOP Class (like Planar MPR)? WG11 believes that there are minimal use cases in modern equipment for such a SOP Class.
7. Must the cropping used for inputs 2-n of the Segmented Volume Rendering Volumetric Presentation States be constrained to include reference to a segmentation object? Even though this is the most common case, there are currently no constraints on the types of cropping to be used.
8. Should INCLUDE_SEQ and EXCLUDE_SEQ Enumerated Values of Volume Cropping Method () be excluded from the (basic) Volume Rendering Volumetric Presentation State SOP Class?
9. Should the presentation state specifiy how the result of the classification and blending steps is used to derive the final output of the shader? Implementations could differ in using the results for the different reflection characteristics.
10. Should the enumerated values of Compositing Method (0070,1206) be constrained for Volume Rendering? They currently include all values from Planar MPR plus VOLUME_RENDERED.
11. Regarding Presentation Animation Style (0070,1A01) of SWIVEL, does the smoothness of the swivel motion need to be more tightly prescribed by the standard? It is currently a recommendation that the implementation “smooth” the transition in direction (such as by using sinosoidal motion in the swivel), but it is essentially an implementation choice. Should the specific motion be specified? Is another element required to specify the specific characteristics of the motion (sinosoidal, bounce, etc.) if more than one method is desirable?

Assumptions Used and Closed Issues

1. Is it necessary to create different SOP Classes?

   Three SOP Classes are defined

2. Is there a need for separate algorithms for deriving the voxel color and opacity from multiple inputs in the compositing stage? Can they be derived using the same logic for both, or are the mechanisms for determining opacity different from those determining color?

   Separate table for Opacity for each VPS input. Leave in the Relationship module for the present – may decide to move the a later stage when those modules are defined. This is a documentation concern only – the opacity maps are a characteristic of the input regardless of which module includes the elements.

Changes to NEMA Standards Publication PS 3.2-2011
Digital Imaging and Communications in Medicine (DICOM)
Part 2: Conformance

Item #1: Add SOP Classes to PS3.2 Table A.1-2

Table A.1-2
UID VALUES

UID Value	UID NAME	Category
…		
1.2.840.10008.5.1.4.1.1.11.x3	**Volume Rendering Volumetric Presentation State Storage SOP Class**	**Transfer**
1.2.840.10008.5.1.4.1.1.11.x4	**Segmented Volume Rendering Volumetric Presentation State Storage SOP Class**	**Transfer**
1.2.840.10008.5.1.4.1.1.11.x5	**Multiple Volume Rendering Volumetric Presentation State Storage SOP Class**	**Transfer**
…		

Changes to NEMA Standards Publication PS 3.3-2011

Digital Imaging and Communications in Medicine (DICOM)

Part 3: Information Object Definitions

Item #2: Add references to PS 3.3 Section 2

2.6 Other References

[Phong 1975] Communications of the ACM. B. T. Phong 1975, 18 6 311-317 "Illumination for computer generated pictures"

[Porter-Duff 1984] SIGGRAPH '84 Proceedings of the 11th annual conference on Computer graphics and interactive techniques. T. Porter and T Duff 1984, 253-259 "Compositing Digital Images"

Item #3: Add IODs to PS3.3 Table A.1-1

IODs **Modules**	**Volume Rendering Volumetric Presentation State**
Patient	**M**
Clinical Trial Subject	**U**
General Study	**M**
Patient Study	**U**
Clinical Trial Study	**U**
General Series	**M**
Clinical Trial Series	**U**
Presentation Series	**M**
Frame Of Reference	**M**
General Equipment	**M**
Enhanced General Equipment	**M**
Volumetric Presentation State Identification	**M**
Volumetric Presentation State Relationship	**M**
Volume Cropping	**M**
Presentation View Description	**M**
Render Geometry	**M**
Render Shading	**U**
Render Display	**M**
Volumetric Graphic Annotation	**U**
Graphic Annotation	**U**
Graphic Layer	**C**
Presentation Animation	**U**
SOP Common	**M**

Item #4: Add sections to PS3.3 Annex A.X

A.X.x3 VOLUME RENDERING VOLUMETRIC PRESENTATION STATE INFORMATION OBJECT DEFINTION

A.X.x3.1 Volume Rendering Volumetric Presentation State Description

The Volume Rendering Volumetric Presentation State Information Object Definition (IOD) specifies information that defines a Volume Rendering presentation from volume datasets that are referenced from within the IOD.

It includes capabilities for specifying:

a. spatial registration of the input datasets
b. cropping of the volume datasets by a bounding box, oblique planes and segmentation objects
c. the generation geometry of volume rendered reconstruction
d. shading models
e. scalar to P-Value or RGB Value conversions
f. compositing of multiple volume streams and one volume stream with segmentations
g. clinical description of the specified view
h. volume and display relative annotations, including graphics, text and overlays
i. membership in a collection of related Volumetric Presentation States intended to be processed or displayed together
j. the position within a set of sequentially related Volumetric Presentation States
k. animating of the view
l. reference to an image depicting the view described by the Volumetric Presentation State

The Volume Rendering Volumetric Presentation State IOD is used in three SOP Classes as defined in PS3.4 Storage Service Class: the Volume Rendering Volumetric Presentation State SOP Class used for rendering a single Volume input into a render view, the Segmented Volume Rendering Volumetric Presentation State SOP Class used for rendering a single Volume with one or more croppings amalgamated into a rendered view, and the Multiple Volume Rendering Volumetric Presentation State SOP Class used for rendering a multiple Volumes each with optional croppings amalgamated into a rendered view.

A.X.x3.2 Volume Rendering Volumetric Presentation State IOD Module Table

Table A.X.x3-1
VOLUME RENDERING VOLUMETRIC PRESENTATION STATE IOD MODULES

IE	Module	Reference	Usage
Patient	Patient	C.7.1.1	M
	Clinical Trial Subject	C.7.1.3	U
Study	General Study	C.7.2.1	M
	Patient Study	C.7.2.2	U
	Clinical Trial Study	C.7.2.3	U
Series	General Series	C.7.3.1	M
	Clinical Trial Series	C.7.3.2	U

	Presentation Series	C.11.9	M
Frame of Reference	Frame of Reference	C.7.4.1	M
Equipment	General Equipment	C.7.5.1	M
	Enhanced General Equipment	C.7.5.2	M
Presentation State	Volumetric Presentation State Identification	C.11.x1	M
	Volumetric Presentation State Relationship	C.11.x2	M
	Volume Cropping	C.11.x3	M
	Presentation View Description	C.11.x4	M
	Render Geometry	C.11.x6	M
	Render Shading	C.11.x7	U
	Render Display	C.11.xB	M
	Volumetric Graphic Annotation	C.11.x9	U
	Graphic Annotation	C.10.5	U
	Graphic Layer	C.10.7	C Required if Graphic Layer (0070,0002) is present in Volumetric Presentation State Relationship module, Volume Graphic Annotation, or Graphic Annotation module
	Presentation Animation	C.11.xA	U
	SOP Common	C.12.1	M

A.X.x3.3 Volume Rendering Volumetric Presentation State IOD Content Constraints

A.X.x3.3.1 Presentation Input Restrictions

Presentation Input Type (0070,1202) shall have a value of VOLUME.

PS 3.4 Section B.5.1.z for SOP Class-specific Presentation Input Restrictions..

A.X.x1.3.2 Render Style

Render Style (0070,x601) shall have a value of VOLUME_RENDER.

A.X.x3.3.3 Graphic Annotation Module

The values of the following attributes, if present, shall be DISPLAY:

- Bounding Box Annotation Units (0070,0003)
- Anchor Point Annotation Units (0070,0004)
- Graphic Annotation Units (0070,0005)

The specified annotation is associated with the specified View and not with the input data, and may have clinical relevance only to the specified View. Therefore, if an application alters the View from that defined by the Presentation State, annotation may no longer be clinically correct.

See PS 3.1 Annex X "Usage of Annotations in Volumetric Presentation States (Informative)" for guidance on usage of the graphic annotation styles available in this IOD.

A.X.x3.3.4 Volumetric Presentation State Reference Coordinate System

All SOP Instances referenced in the Volumetric Presentation State Relationship Module shall be implicitly or explicitly registered to the Volumetric Presentation State Reference Coordinate System. See C.11.x2.3.

Item #5: Changes to PS 3.3 Annex A

A.X.x1.3 Planar MPR Volumetric Presentation State IOD Content Constraints

A.X.x1.3.1 Presentation Input Restrictions

Presentation Input Type (0070,1202) shall have a value of VOLUME.

If the value of Pixel Presentation (0008,9205) is MONOCHROME, the Volumetric Presentation State Input Sequence (0070,1201) shall have only a single item.

Enumerated values for Compositing Method (0070,1206) shall be AVERAGE_IP, MAXIMUM_IP, and MINIMUM_IP.

Item #6: Add to PS 3.3 Annex C

C.11.x6 Render Geometry Module

Table C.11.x6-1 contains Attributes that describe the geometry of the render view.

Table C.11.x6-1
RENDER GEOMETRY MODULE ATTRIBUTES

Attribute Name	Tag	Type	Attribute Description
Render Style	(0070,x601)	1	Style of rendering performed. Enumerated Values: VOLUME_RENDER
Render Projection	(0070,x602)	1	Projection style. Enumerated Values: ORTHOGRAPHIC PERSPECTIVE
Viewpoint Position	(0070,x603)	1	Position of the viewpoint in volume space. Encoded as three coordinates (x,y,z) in the Volumetric Presentation State Reference Coordinate System..
Viewpoint LookAt Point	(0070,x604)	1	Point the viewpoint is looking at. Encoded as three coordinates (x,y,z) in the Volumetric Presentation State Reference Coordinate System..
Viewpoint Up Direction	(0070,x605)	1	Up direction. A vector encoded as three coordinates (x,y,z) in the Volumetric Presentation State Reference Coordinate System..
Render Field of View	(0070,x606)	1	Extent of the viewing field of view in mm. 6-tuple of values X_{left}, X_{right}, Y_{top}, Y_{bottom}, $Distance_{near}$, $Distance_{far}$ within a coordinate system relative to the Viewpoint (the Viewpoint Coordinate System). See C.11.x6.1.

Compositing Method	(0070,1206)	1	The rendering method used. Enumerated values: AVERAGE_IP: A method that projects the mean intensity of all interpolated samples that fall in the path of each ray traced from the viewpoint to the plane of projection. MAXIMUM_IP: A method that projects the interpolated sample with maximum intensity that fall in the path of each ray traced from the viewpoint to the plane of projection. MINIMUM_IP: A method that projects the interpolated sample with minimum intensity that fall in the path of each ray traced from the viewpoint to the plane of projection. VOLUME_RENDERED: Volume Rendering is a data visualization method where a 2D render view through volume data is created. Voxels (volume sample points) are assigned a color and an opacity (alpha), and for each XY coordinate of the render view the output pixel value is determined by accumulating the set of non-transparent voxels samples along the z-axis.

C.11.x6.1 Render Field of View

The Render Field of View defines the region of the volume data that may appear on the screen.

The viewpoint is positioned and oriented within the Volumetric Presentation State Reference Coordinate System (VPS-RCS) by Viewpoint Position (0070,x603), Viewpoint LookAt Point (0070,x604), and Viewpoint Up Direction (0070,x605). This position and orientation sets up a Viewpoint Coordinate System (VCS), which is a right-hand coordinate system in which the viewpoint is positioned at (0,0,0) and is looking at a point at (0,0,-z) and the up direction is along the +y axis.

Render Field of View (0070,x606) is positioned within the VPS- RCS defined with the following coordinate values in the Viewpoint Coordinate System:

- X_{left}, X_{right}, specify the coordinates of the left and right vertical clipping planes of at $Distance_{far}$.
- Y_{top}, Y_{bottom} specify the coordinates for the top and bottom horizontal clipping planes at $Distance_{far}$.
- $Distance_{near}$, $Distance_{far}$ specify the distances to the near and far depth clipping planes. Both distances must be positive.

In the case of a Render Projection (0070,x602) value of ORTHOGRAPHIC, Render Field of View (0070,x606) defines a rectangular cuboid with dimensions (X_{right} -X_{left}) by (Y_{top} -Y_{bottom}) by ($Distance_{far}$ - $Distance_{near}$) as shown in Figure C.11.x6.1:

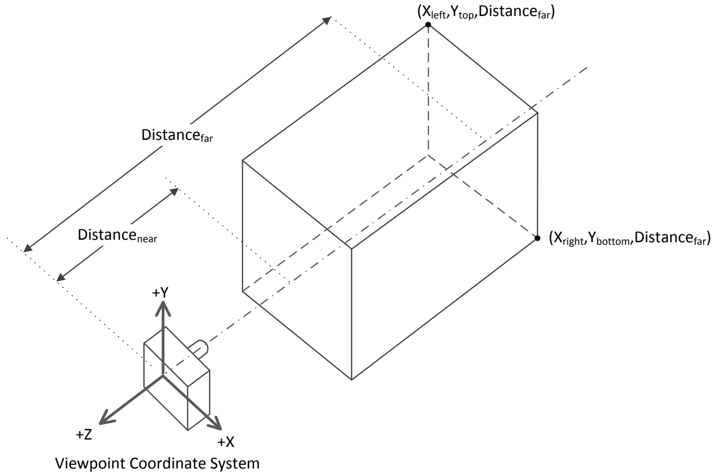

Figure C.11.x6-1: ORTHOGRAPHIC Field of View Geometry

In the case of a Render Projection (0070,x602) value of PROJECTION, Render Field of View (0070,x606) defines a frustum in which the far rectangular is larger than the near rectangular. The extent of the near rectangle is established by the four points where rays originating at the viewpoint position and intersecting the X_{left}, X_{right}, Y_{top}, and Y_{bottom} intersect the plane at the $Distance_{near}$ distance from the viewpoint (Figure C.11.x6-2):

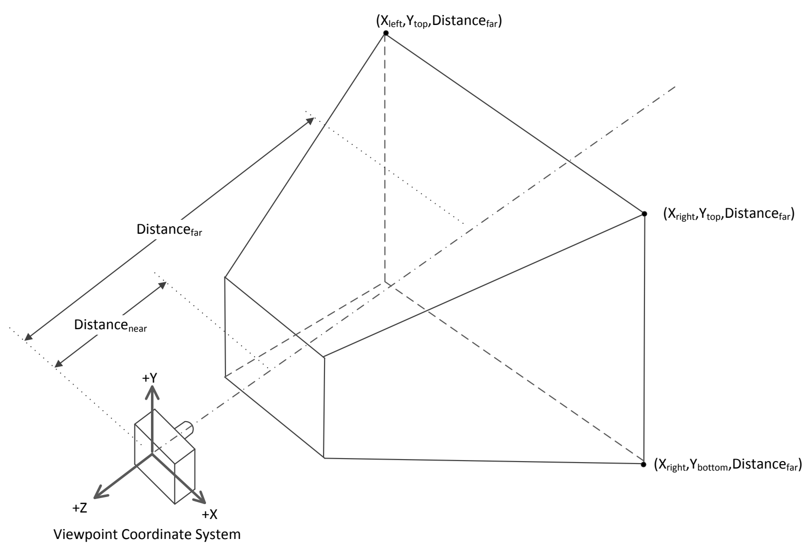

Figure C.11.x6-2: PERSPECTIVE Field of View Geometry

C.11.x7 Render Shading Module

Shading enhances the visual perception of a volume by adding reflection characteristics.
Table C.11.x7-1 contains Attributes that describe the shading used in a render view.
If the Render Shading Module is omitted for a render view an unshaded volume rendering shall be performed.

> Note: This module assumes a Phong shading model. An implementation may use any appropriate shading model, translating these parameters into a similar meaning in the chosen shading model.
> The generation mechanism for the surface normals which are required for the Phong shading model is not specified by DICOM.

Table C.11.x7-1
RENDER SHADING MODULE ATTRIBUTES

Attribute Name	Tag	Type	Attribute Description
Shading Style	(0070,x701)	1	Enumerated Values: SINGLESIDED DOUBLESIDED See C.11.x7.1
Ambient Reflection Intensity	(0070,x702)	1	Intensity of the ambient reflection in the relative range 0.0 to 1.0, inclusive.
Light Direction	(0070,x703)	1C	A vector encoded as three coordinates (x,y,z). Required if Diffuse Reflection Intensity (0070,x704) or Specular Reflection Intensity (0070,x705) is present.
Diffuse Reflection Intensity	(0070,x704)	3	Intensity of the diffuse reflection in the relative range 0.0 to 1.0, inclusive. If absent, no diffuse reflection is used in the view.
Specular Reflection Intensity	(0070,x705)	3	Intensity of the specular reflection in the relative range 0.0 to 1.0, inclusive. If absent, no specular reflection is used in the view.
Shininess	(0070,x706)	3	Specifies the roughness of the rendered surfaces, in the relative range 0.0 to 1.0, inclusive. A value of 0.0 representing a rough surface and a value of 1.0 representing the smoothest surface which can be generated by the implementation. Note: In theory, the range of shininess is from 0 to infinity. However, in practice each implementation has a finite upper limit for shininess. The implementation is expected to multiply this value by its upper limit value and use the result in its shading algorithm for shininess. If absent, the shininess is an implementation decision.

C.11.x7.1 Shading Style

The Shading Style (0070,x701) determines if the shading calculations are performed for all voxels, based on the facing of the surface normal which has been generated for that voxel:

- SINGLESIDED: only “front-facing” voxels are shaded.
- DOUBLESIDED: “front-facing” and “back-facing” voxels are shaded.

Note: “Front-facing” voxels are those with a negative dot product between the surface normal which has been generated for that voxel and the vector between the Viewpoint Position (0070,x603) and the ViewPoint LookAt Point (0070,x604). “Back-facing” voxels are those with a positive dot product.

C.11.xB Render Display Module

Table C.11.xB-1 specifies the attributes that define the transformations of the processed Volumetric Presentation State inputs into a single VPS display space, as described in the Volumetric Presentation State pipelines in PS 3.4 Section X.2.2.

Table C.11.xB-1
RENDER DISPLAY MODULE ATTRIBUTES

Attribute Name	Tag	Type	Attribute Description
Pixel Presentation	(0008,9205)	1	Grayscale or color space of the Presentation State output. Enumerated Values: MONOCHROME Output consists of P-Values TRUE_COLOR Output consists of PCS-Values
Volume Stream Sequence	(0070,xB01)	1	Volumetric source input streams combined through Volume Blending. One or more items shall be included in this sequence.
>Volumetric Presentation Input Source UID	(0070,xB02)	1	Identifier of the source data set processed in this volume stream. See C.11.x2.4.
>Presentation State Classification Component Sequence	(0070,1801)	1	Sequence of classification components in which the order of items is significant. Each classification component converts one or two processed inputs into a single RGB output. One or more items shall be included in this sequence. See C.11.x8.2.
>>Component Type	(0070,1802)	1	Type of component. Enumerated values: ONE_TO_RGBA TWO_TO_RGBA See C.11.x8.2.

>>Component Input Sequence	(0070,1803)	1	Description of the input or inputs to this component. One item shall be present in this sequence if Component Type (0070,1802) has a value of ONE_TO_RGBA. Two items shall be present in this sequence if Component Type (0070,1802) has a value of TWO_TO_RGBA.
>>> Volumetric Presentation Input Index	(0070,1804)	1	Value of Volumetric Presentation Input Number (0070,1207) for this input in the Volumetric Presentation State Input Sequence (0070,1201)
>>>Bits Mapped to Color Lookup Table	(0028,1403)	3	The number of most significant bits of each value of Pixel Data (7FE0,0010) from this frame contributing to the Palette Color Lookup Table input. If absent, Bits Stored (0028,0101) bits of each value of Pixel Data (7FE0,0010) from this frame contributes to the Palette Color Lookup Table input. See C.7.6.23.3.
>>RGB LUT Transfer Function	(0028,140F)	1C	Specifies the mapping that takes place between the input value and RGB output. Enumerated values: EQUAL_RGB Output is R=G=B = input value TABLE Output is RGB LUT values See C.11.x8.1. Required if Pixel Presentation (0008,9205) has a value of TRUE_COLOR.
>>Alpha LUT Transfer Function	(0028,1410)	1	Specifies the transformation that is used to create the Alpha input to a compositor component (if present) and the opacity value for use in the Volume Rendering compositor. Enumerated values: NONE Output = 1 (opaque) for all input values IDENTITY Output = input value TABLE Output = output of the Alpha LUT

>>Red Palette Color Lookup Table Descriptor	(0028,1101)	1C	Specifies the format of the Red Palette Color Lookup Table Data (0028,1201). The second value (first stored pixel value mapped) shall be zero. See C.7.6.3.1.5. Required if RGB LUT Transfer Function (0028,140F) has a value of TABLE.
>>Green Palette Color Lookup Table Descriptor	(0028,1102)	1C	Specifies the format of the Green Palette Color Lookup Table Data (0028,1202). The second value (first stored pixel value mapped) shall be zero. See C.7.6.3.1.5. Required if RGB LUT Transfer Function (0028,140F) has a value of TABLE.
>>Blue Palette Color Lookup Table Descriptor	(0028,1103)	1C	Specifies the format of the Blue Palette Color Lookup Table Data (0028,1203). The second value (first stored pixel value mapped) shall be zero. See C.7.6.3.1.5. Required if RGB LUT Transfer Function (0028,140F) has a value of TABLE.
>>Alpha Palette Color Lookup Table Descriptor	(0028,1104)	1C	Specifies the format of the Alpha Palette Color Lookup Table Data. The second value (first stored pixel value mapped) shall be zero. See C.7.6.3.1.5. Required if Alpha LUT Transfer Function (0028,1410) has a value of TABLE.
>>Palette Color Lookup Table UID	(0028,1199)	3	Palette Color Lookup Table UID. See C.7.9.1. Note: Including the Palette Color Lookup Table UID in Presentation States that use the same palette is helpful to the display application that is rendering several related presentations together.
>>Red Palette Color Lookup Table Data	(0028,1201)	1C	Red Palette Color Lookup Table Data. See C.7.6.3.1.5. Required if RGB LUT Transfer Function (0028,140F) has a value of TABLE and Segmented Red Palette Color Lookup Table Data (0028,1221) is not present.
>>Green Palette Color Lookup Table Data	(0028,1202)	1C	Green Palette Color Lookup Table Data. See C.7.6.3.1.5. Required if Red Palette Color Lookup Table Data (0028,1201) is present.

>>Blue Palette Color Lookup Table Data	(0028,1203)	1C	Blue Palette Color Lookup Table Data. See C.7.6.3.1.5. Required if Red Palette Color Lookup Table Data (0028,1201) is present.
>>Alpha Palette Color Lookup Table Data	(0028,1204)	1C	Alpha Palette Color Lookup Table Data. See C.7.6.3.1.5. Required if Alpha LUT Transfer Function (0028,1410) has a value of TABLE and Segmented Alpha Palette Color Lookup Table Data (0028,1224) is not present.
>>Segmented Red Palette Color Lookup Table Data	(0028,1221)	1C	Segmented Red Palette Color Lookup Table Data. See C.11.x8.5. Required if RGB LUT Transfer Function (0028,140F) has a value of TABLE and Red Palette Color Lookup Table Data (0028,1201) is not present.
>>Segmented Green Palette Color Lookup Table Data	(0028,1222)	1C	Segmented Green Palette Color Lookup Table Data. See C.11.x8.5 Required if Segmented Red Palette Color Lookup Table Data (0028,1221) is present.
>>Segmented Blue Palette Color Lookup Table Data	(0028,1223)	1C	Segmented Blue Palette Color Lookup Table Data. See C.11.x8.5 Required if Segmented Red Palette Color Lookup Table Data (0028,1221) is present.
>>Segmented Alpha Palette Color Lookup Table Data	(0028,1224)	1C	Segmented Alpha Palette Color Lookup Table Data. See C.11.x8.5 Required if Alpha LUT Transfer Function (0028,1410) has a value of TABLE and Alpha Palette Color Lookup Table Data (0028,1204) is not present.

Presentation State Compositor Component Sequence	(0070,1805)	2	RGBA Compositor components. Each RGBA Compositor component combines pairs of RGBA values to produce a single RGBA value. The order of items is significant. If there are more than one compositor component, the components are chained such that the output of one compositor component is an input to the next compositor component. The number of items in this sequence shall be the one less than the number of items in Volume Stream Sequence (0070,xB01). See C.11.x8.3.
>Weighting Transfer Function Sequence	(0070,1806)	1	Transfer functions used to derive the weighting factors for each of the two RGB inputs from both input Alphas. Each function is represented by the formula $f(Alpha_1, Alpha_2) = WeightingFactor$ The function is specified in the form of a table. Two items shall be included in this sequence. The order is significant. The first item specifies the weighting factor for RGB1 and the second item specifies the weighting factor for RGB2. See C.11.x8.4.
>>LUT Descriptor	(0028,3002)	1	Specifies the format of the LUT Data (0028,3006) in this Sequence. The first value (number of entries in the LUT) shall be an even power of two or zero indicating 2^{16}, so that there are an even number of bits in the LUT input. The third value (number of bits in the LUT Data) shall be 8. See C.11.1.1.
>>LUT Data	(0028,3006)	1	LUT Data

Presentation LUT Shape	(2050,0020)	1C	Presentation LUT transformation. Enumerated Values: IDENTITY No further translation necessary; input values are P-Values; INVERSE Output values after inversion are P-Values See C.11.6.1.2. Required if Pixel Presentation (0008,9205) has a value of MONOCHROME.
ICC Profile	(0028,2000)	1C	An ICC Profile encoding the transformation of device-dependent color stored pixel values into PCS-Values. When present, defines the color space of the output of the Volumetric Presentation State. See C.11.15.1.1 Required if Pixel Presentation (0008,9205) has a value of TRUE_COLOR.

Item #7: Changes to PS 3.3 Annex C

C.11.x2 Volumetric Presentation State Relationship Module

Table C.11.x2-1 contains Attributes that describe sets of inputs to a presentation state and how each input is to be displayed in the presentation.

Table C.11.x2-1
VOLUMETRIC PRESENTATION STATE RELATIONSHIP MODULE ATTRIBUTES

Attribute Name	Tag	Type	Attribute Description
Volumetric Presentation State Input Sequence	(0070,1201)	1	Inputs to the Presentation State. Each item represents one input. One or more items shall be included in this sequence.
>Volumetric Presentation Input Number	(0070,x207)	1	Identification number of the input. Values shall be ordinal numbers starting from 1 and monotonically increasing by 1 within the Volumetric Presentation State instance.
>Presentation Input Type	(0070,x202)	1	Type of input. See C.11.x2.1. Enumerated Value: VOLUME

Attribute Name	Tag	Type	Attribute Description
>Input Sequence Position Index	(0070,x203)	1C	Position of this input data within a set of sequential inputs. Multiple inputs may share the same value. Note: For example, CT and PET inputs. Distinct values shall be ordinal numbers starting from 1 and monotonically increasing by 1 within the Volumetric Presentation State instance. See C.11.x2.2. Required if Presentation Animation Style (0070,xA01) is present with a value of INPUT_SEQ. Note: The inputs of the sequence are typically temporally related.
>Referenced Image Sequence	(0008,1140)	1C	The set of images comprising this input volume. One or more items shall be included in this sequence. See C.11.x2.1 for constraints on objects referenced by this sequence. Required if Presentation Input Type (0070,x202) has a value of VOLUME.
>>Include 'Image SOP Instance Reference Macro' Table 10-3			
>Referenced Spatial Registration Sequence	(0070,0404)	1C	A reference to a Spatial Registration Instance that is used to register the referenced inputs. Only one item shall be included in this sequence. All images referenced by the Referenced Image Sequence (0008,1140) of this item of the Volumetric Presentation State Input Sequence (0070,x201) shall be referenced by the Spatial Registration instance. See C.11.x2.3. Required if the Frame of Reference UID (0020,0052) value of the Images referenced by the Referenced Image Sequence (0008,1140) of this item of the Volumetric Presentation State Input Sequence (0070,x201) does not match the Frame of Reference UID (0020,0052) value of this Presentation State instance. May be present otherwise.
>>Include SOP Instance Reference Macro Table 10-11			
>Include VOI LUT Macro Table C.11-2b			
>Crop	(0070,x204)	1	Specifies whether to crop this input. Enumerated Values: YES NO

Attribute Name	Tag	Type	Attribute Description
>Cropping Specification Index	(0070,x205)	1C	Values of Cropping Specification Number (0070,x309) of the item in the Volume Cropping Sequence (0070,x301) specifying the cropping methods to be applied to this input. Required if Crop (0070,x204) has a value of YES.
>Compositing Method	(0070,1206)	1C	The rendering method for this input. Enumerated values: AVERAGE_IP: A method that projects the mean intensity of all interpolated samples that fall in the path of each ray traced from the viewpoint to the plane of projection. MAXIMUM_IP: A method that projects the interpolated sample with maximum intensity that fall in the path of each ray traced from the viewpoint to the plane of projection. MINIMUM_IP: A method that projects the interpolated sample with minimum intensity that fall in the path of each ray traced from the viewpoint to the plane of projection. Required if MPR Thickness Type (0070,1502) ~~**has**~~ **is present with** a value of SLAB.
>Volumetric Presentation Input Source UID	**(0070,xB02)**	**1C**	**Identifier of the source data set. The UID may be shared among multiple Volumetric Presentation State Input Sequence (0070,1201) items.** **See C.11.x2.4.** **Required if Render Style (0070,x601) is present. May be present otherwise.**

C.11.x2.4 Volumetric Presentation Input Source UID

Unique Identifier of a specific set of source data. Two items of Volumetric Presentation State Input Sequence (0070,1201) that have identical values of Volumetric Presentation Input Source UID (0070,xB02) shall have equal values of the following elements:

- Presentation Input Type (0070,1202)
- Referenced Image Sequence (0008,1140)
- Referenced Spatial Registration Sequence (0070,0404)

C.11.x8 Volumetric Presentation State Display Module

Table C.11.x8-1 specifies the attributes that define the transformations of the processed Volumetric Presentation State inputs into a single VPS display space, as described in the Volumetric Presentation State pipeline in PS 3.4 Section X.2.1.

Table C.11.x8-1
VOLUMETRIC PRESENTATION STATE DISPLAY MODULE ATTRIBUTES

Attribute Name	Tag	Type	Attribute Description
…			
>Component Type	(0070,1802)	1	Type of component. Enumerated values: ONE_TO_RGBA TWO_TO_RGBA See C.11.x8.2 ~~**for description of the components corresponding to each selection**~~.
…			
Presentation State Compositor Component Sequence	(0070,1805)	2C	~~**Sequence of**~~ RGB Compositor ~~**components in which the order of items is significant**~~. Each RGB Compositor component combines ~~**together**~~ pairs of RGB values to produce a single RGB value. **The order of items is significant.** If there is more than one compositor component, the components are chained such that the output of one compositor component is an input to the next compositor component. The number of items in this sequence shall be **one less than** the number of items in Presentation State Classification Component Sequence (0070,1801) ~~**minus one**~~. See C.11.x8.3. Required if Pixel Presentation (0008,9205) has a value of TRUE_COLOR.

>Weighting Transfer Function Sequence	(0070,1806)	1	Transfer functions **used to derive the weighting factors for each of the two RGB inputs from both input Alphas.** **Each function is** ~~**each**~~ represented by the formula $f(Alpha_1, Alpha_2) = WeightingFactor$ ~~**used to derive the weighting factors for each of the two RGB inputs from both input Alphas.**~~ The function is specified in the form of a table. ~~**Two items shall be included in this sequence to produce weighting factors for RBG1 and RGB2 inputs, respectively.**~~ **The order is significant. The first item specifies the weighting factor for RGB1 and the second item specifies the weighting factor for RGB2.** See C.11.x8.4.
>>LUT Descriptor	(0028,3002)	1	Specifies the format of the LUT Data (0028,3006) in this Sequence. The first value (number of entries in the LUT) shall be an even power of two or zero indicating 2^{16}, so that there are an even number of bits in the LUT input. The third value (number of bits in the LUT Data) shall be 8. See C.11.1.1.
>>LUT Data	(0028,3006)	1	LUT Data ~~**in this Sequence**~~.
…			

C.11.xA Presentation Animation Module

Table C.11.xA-1 contains Attributes that describe animation of the presentation.

Table C.11.xA-1
PRESENTATION ANIMATION MODULE ATTRIBUTES

Attribute Name	Tag	Type	Attribute Description
Presentation Animation Style	(0070,1A01)	1	Animation style ~~**intended by the source.**~~ Defined Terms: INPUT_SEQ PRESENTATION_SEQ CROSSCURVE **FLYTHROUGH** **SWIVEL** See C.11.xA.1 ~~**for description of terms**~~.
Recommended Animation Rate	(0070,1A03)	3	Recommended rate at which the inputs shall be displayed. Shall have a value greater than zero. See C.11.xA.1 ~~**for units**~~.
Animation Curve Sequence	(0070,1A04)	1C	Curve describing the trajectory of a flythrough animation. Only one Item shall be included in this sequence. Required if Presentation Animation Style (0070,1A01) is CROSSCURVE **or FLYTHROUGH**.
>Number of Volumetric Curve Points	(0070,150C)	1	Number of (x,y,z) points in Volumetric Curve Points (0070,150D).
>Volumetric Curve Points	(0070,150D)	1C	Coordinates of points on the curve in the Volumetric Presentation State Reference Coordinate System, in mm. One triplet (x,y,z) shall be present for each point in the curve. Note: The points on the curve are samples. It is an implementation decision how the points are connected.
Animation Step Size	(0070,1A05)	1C	Distance in mm along the curve the display moves in one step. Required if Presentation Animation Style (0070,1A01) has a value of CROSSCURVE **or FLYTHROUGH**.
Swivel Range	**(0070,xA06)**	**1C**	**Range in which a volume rotates back-and-forth around the swivel axis, in degrees. The initial position is at the midpoint of the swivel range.** **See C.11.xA.1.** **Required if Presentation Animation Style (0070,1A01) is SWIVEL.**

…

C.11.xA.1 Presentation Animation Style

The presence of Presentation Animation Style (0070,1A01) indicates that a form of view animation is intended by the creator of the Presentation State, and the value of the attribute indicates the nature of such animation. See PS 3.4 Section X.3.2 for further description of the various presentation animation styles.

Values of Presentation Animation Style (0070,1A01) are:

- …
- **FLYTHROUGH: Indicates that the render view parameters be stepped along the curve defined in Animation Curve Sequence (0070,1A04) at the rate specified by Recommended Animation Rate (0070,1A03) in steps per second. Presentation Animation Style (00070,1A01) value of FLYTHROUGH shall be present only if Render Projection (0070,x602) is present, Viewpoint Position (0070,x603) is on the curve, and the ray from the Viewpoint Position (0070,x603) to the Viewpoint LookAt Point(0070,x604) is tangent to the curve at the starting point of the animation.**

  The viewpoint (position, lookAtPoint, upDirection) changes dynamically throughout the flythrough animation. Initially, the viewpoint position is set to Viewpoint Position (0070,x603), viewpoint lookAtPoint is set to Viewpoint LookAtPoint (0070,x604), and viewpoint upDirection is set to Viewpoint Up Direction (0070,x605). Also, the following derived vectors will be used in the calculation of viewpoint upDirection at each step of the animation:

  - $V_{origLOOK}$ **is the vector from Viewpoint Position (0070,x603) to Viewpoint LookAt Point (0070,x604)**
  - V_{origUP} **is unit vector in the direction Viewpoint Up Direction (0070,x605)**

  A lookVector V_{look} **at each animation step is defined as the vector from the current viewpoint position ending at the current viewpoint lookAtPoint. The length of this vector** L_{look} **is always the distance between Viewpoint Position (0070,x603) and Viewpoint LookAtPoint (0070,x604).**

  From the initial view, viewpoint (position, lookAtPoint, upDirection) changes at each step of the animation as follows:

  - **Viewpoint lookAtPoint steps along the curve by Animation Step Size (0070,A105) millimeters.**
  - V_{look} **is the vector of length** L_{look} **tangent to the curve at the new viewpoint lookAtPoint ending at viewpoint lookAtPoint and pointing in the direction of the most recent step along the curve.**
  - **Viewpoint position is the point at the origin of the new** V_{look}**, which is always at a distance** L_{look} **from viewpoint lookAtPoint.**
  - **Viewpoint upDirection adheres to the cross-product relationship**

    $$V_{origLOOK} \times V_{origUP} = V_{look} \times V_{up}$$

    where V_{up} **is the unit vector in viewpoint upDirection**
  - **The "viewpoint coordinate system" is defined at each animation step as the right-hand coordinate system (**V_{look}**,** V_{up}**, (**$V_{look} \times V_{up}$**)) with origin viewpoint position.**
  - **Render Field of View (0070,x606) defines the render view throughout the animation relative to the viewpoint coordinate system.**
- **SWIVEL: Indicates that the rendered volume rotates around the "swivel axis", which is defined as the axis parallel to the Viewpoint Up Direction (0070,x605) intersecting the**

Viewpoint LookAt Point (0070,x604). The rendered volume rotates back and forth in the angular range specified by Swivel Range (0070,xA06) at a rotational frequency specified by Recommended Animation Rate (0070,1A03) in full rotations per second. The display application should provide for smooth (rather than abrupt) changes in direction as the swivel approaches the limits of the swivel range. Shall not be used unless Render Projection (0070,x602) is present.

Changes to NEMA Standards Publication PS 3.4-2011
Digital Imaging and Communications in Medicine (DICOM)

Part 4: Service Class Specifications

Item #8: Change to PS 3.4 Section 2

2 Normative References

…

[Porter-Duff 1984] SIGGRAPH '84 Proceedings of the 11th annual conference on Computer graphics and interactive techniques. T. Porter and T Duff 1984, 253-259 "Compositing Digital Images"

Item #9: Add SOP Classes to PS3.4 Annex B

B.5 Standard SOP Classes

Table B.5-1
Standard SOP Classes

SOP Class	SOP Class UID	IOD Specification (defined in PS 3.3)
Volume Rendering Volumetric Presentation State Storage SOP Class	1.2.840.10008.5.1.4.1.1.11.x3	Volume Rendering Volumetric Presentation State
Segmented Volume Rendering Volumetric Presentation State Storage SOP Class	1.2.840.10008.5.1.4.1.1.11.x4	Volume Rendering Volumetric Presentation State
Multiple Volume Rendering Volumetric Presentation State Storage SOP Class	1.2.840.10008.5.1.4.1.1.11.x5	Volume Rendering Volumetric Presentation State

B.5.1.x Planar MPR Volumetric Presentation State Storage SOP Classes

The requirements of Annex X**.2.1** apply to the following SOP Classes:

…

B.5.1.z Volume Rendering Volumetric Presentation State Storage SOP Classes

The requirements of Annex X.2.2 apply to the following SOP Classes:

- **Volume Rendering Volumetric Presentation State SOP Class**
- **Segmented Volume Rendering Volumetric Presentation State SOP Class**
- **Multiple Volume Rendering Volumetric Presentation State SOP Class**

Volume Rendering Volumetric Presentation State SOP Class shall use the Volume Rendering Volumetric Presentation State IOD and include a single item in Volumetric Presentation State Input Sequence (0070,x201) and a single item in Volume Stream Sequence (0070,0B01). Values of Volume Cropping Method (0070,1301), if present, shall be limited to BOUNDING_BOX or OBLIQUE_PLANES.

Commented [JML1]: Insert real tag numbers for attributes from Sup156.

Segmented Volume Rendering Volumetric Presentation State SOP Class shall use the Volume Rendering Volumetric Presentation State IOD and include a single item in Volume Stream Sequence (0070,0B01).

Commented [JML2]: Constrain all inputs to be the same volume

Multiple Volume Rendering Volumetric Presentation State SOP Class shall use the Volume Rendering Volumetric Presentation State IOD and include two or more items in Volume Stream Sequence (0070,0B01).

Commented [JML3]: Constrain all inputs of a Volume Stream to be the same volume.

Item #10: Modifications to PS3.4 Annex I

I.4 Media Storage Standard SOP Classes

Table I.4-1
Media Storage Standard SOP Classes

SOP Class	SOP Class UID	IOD Specification (defined in PS 3.3)
Volume Rendering Volumetric Presentation State Storage SOP Class	**1.2.840.10008.5.1.4.1.1.11.x3**	**Volume Rendering Volumetric Presentation State**
Segmented Volume Rendering Volumetric Presentation State Storage SOP Class	**1.2.840.10008.5.1.4.1.1.11.x4**	**Volume Rendering Volumetric Presentation State**
Multiple Volume Rendering Volumetric Presentation State Storage SOP Class	**1.2.840.10008.5.1.4.1.1.11.x5**	**Volume Rendering Volumetric Presentation State**

Item #11: Modifications to PS3.4 Annex X

X.1. Overview

X.1.1 Scope

The Volumetric Presentation State Storage SOP Classes extend the functionality of the Storage Service class (defined in Annex B) to add the ability to convey an intended Volumetric Presentation State or record an existing Volumetric Presentation State. The SOP Classes specify the information and behavior that may be used to present (display) images that are referenced from within the SOP Classes.

They include capabilities for specifying:

a. spatial registration on the input datasets
b. cropping of the volume datasets by a bounding box, oblique planes and segmentation objects
c. the generation geometry of volumetric views
d. shading models
e. scalar to P-Value or RGB Value conversions
f. compositing of multiple renderings
g. compositing of multiple volume streams and one volume stream with segmentations
h. clinical description of the specified view

i. volume and display relative annotations, including graphics, text and overlays

j. membership to a collection of related Volumetric Presentation States intended to be processed or displayed together

k. the position within a set of **sequentially** related Volumetric Presentation States

l. animation of the view

m. reference to an image depicting the view described by the **Volumetric** Presentation State

Each Volumetric Presentation State corresponds to a single view (equivalent to an Image Box in a Hanging Protocol or Structured Display). If multiple Volumetric Presentation States are intended to be displayed together (e.g. a set of orthogonal MPR views) these Presentation States can be grouped by assigning them to a Display Collection. However, any detailed information about how a set of views should be presented can only be described by a Structured Display instance or a Hanging Protocol.

The Planar MPR Volumetric Presentation State refers to the multi-planar geometry and grayscale or color image transformations that are to be applied in an explicitly defined manner to convert the stored image pixel data values in a Composite Image Instance to presentation values (P-Values) or Profile Connection Space values (PCS-Values) when an image is displayed on a softcopy device.

The Volume Render Volumetric Presentation State represents a view of volume data consisting of a number of steps, many of which are parametrically specified in the Volume Rendering SOP Classes. Steps that are usually implemented by proprietary algorithms are not described in this supplement, and are implementation-specific. The processing steps are:

- **Segmentation, or separating the volume data into groups that will share a particular color palette. Segmentation objects are specified as inputs to the Volumetric Presentation State.**
- **Gradient Computation, or finding edges or boundaries between different types of tissue in the volumetric data. Gradient Computation used is an implementation decision outside the scope of the Volumetric Presentation State.**
- **Resampling of the volumetric data to create new samples along the imaginary ray behind each pixel in the output two-dimensional view, generally using some interpolation of the values of voxels in the neighborhood of the new sample. The interpolation method used is an implementation decision outside the scope of the Volumetric Presentation State.**
- **Classification of ray samples to assign a color and opacity to each sample. Classification parameters are specified in the Volumetric Presentation State.**
- **Shading or the application of a lighting model to ray samples indicating the effect of ambient, diffuse, and specular light on the sample. Basic shading parameters are specified in the Volumetric Presentation State.**
- **Compositing or the accumulation of samples on each ray into the final value of the pixel corresponding to that ray. The specific algorithms used are outside the scope of the Volumetric Presentation State.**
- **Conversion to presentation values (P-Values) or Profile Connection Space values (PCS-Values) when an image is displayed on a softcopy device.**

The result of application of a Volumetric Presentation State is not expected to be exactly reproducible on different systems. It is difficult to describe the display and render algorithms in enough detail in an interoperable manner, such that a presentation produced at a later time is indistinguishable from that of the original presentation. While Volumetric Presentation States use established DICOM concepts of grayscale and color matching (GSDF and ICC color profiles) and

provides a generic description of the different types of display algorithms possible, variations in algorithm implementations within display devices are inevitable and an exact match of volume presentation on multiple devices cannot be guaranteed. Nevertheless, reasonable consistency is provided by specification of inputs, geometric descriptions of spatial views, type of processing to be used, color mapping and blending, input fusion, and many generic rendering parameters, producing what is expected to be a clinically acceptable result.

...

Item #12: Append to PS 3.4 Annex X.2

X.2.2 Volume Rendering Volumetric Transformation Process

The Volume Rendering Volumetric Presentation State Storage SOP Classes support a set of transformations to produce derived volumetric views of volume input data.

The Volume Rendering Volumetric Presentation State Storage SOP Class defines a volumetric view from a single volume input to produce a volume render view. The sequence of transformations from volumetric inputs into PCS-Values is explicitly defined in the reference pipeline described in Figure X.2-1a.

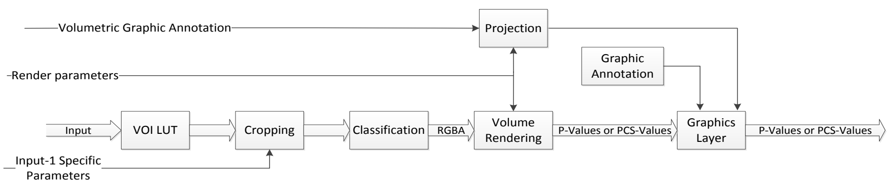

Figure X.2-1a: Volume Rendering Volumetric Pipeline

The Segmented Volume Rendering Volumetric Presentation State Storage SOP Class defines a volumetric view from a single volume dataset with optional segmentation croppings, each colored separately and blended into the volume to be rendered. The sequence of transformations from volumetric inputs into PCS-Values is explicitly defined in the reference pipeline described in Figure X.2-1b.

There is a single item in the Volume Stream Sequence (0070,xB01) for instances of this SOP Class.

The classified segmented volumes are blended in lowest to highest priority order using B-over-A blending of the RGB data and the corresponding opacity (alpha) data. The first item in the Presentation State Classification Component Sequence (0070,1801) is the base upon which subsequent items are Segmentation cropped and B-over-A blended with it.

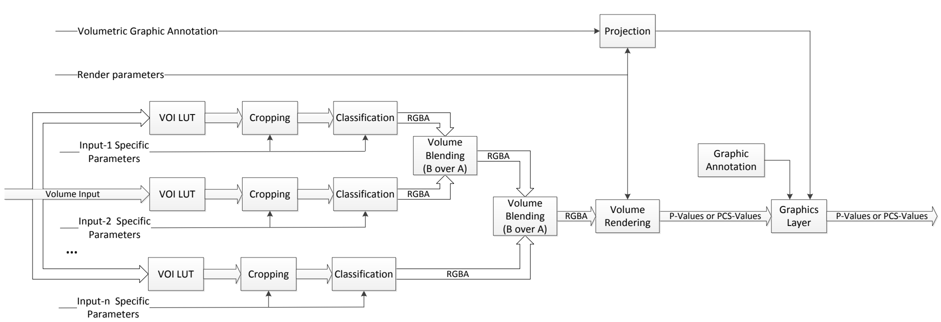

Figure X.2-1b: Segmented Volume Rendering Volumetric Pipeline

The Multiple Volume Rendering Volumetric Presentation State Storage SOP Class defines a volumetric view from more than one volume input. The sequence of transformations from volumetric inputs into P-

Values or PCS-Values is explicitly defined in the reference pipeline described in Figure X.2-1c. The specific algorithms for volume rendering may differ, but must result in a similar appearance.

Each item in the Volume Stream Sequence (0070,xB01) produces one input to a RGBA Compositor.

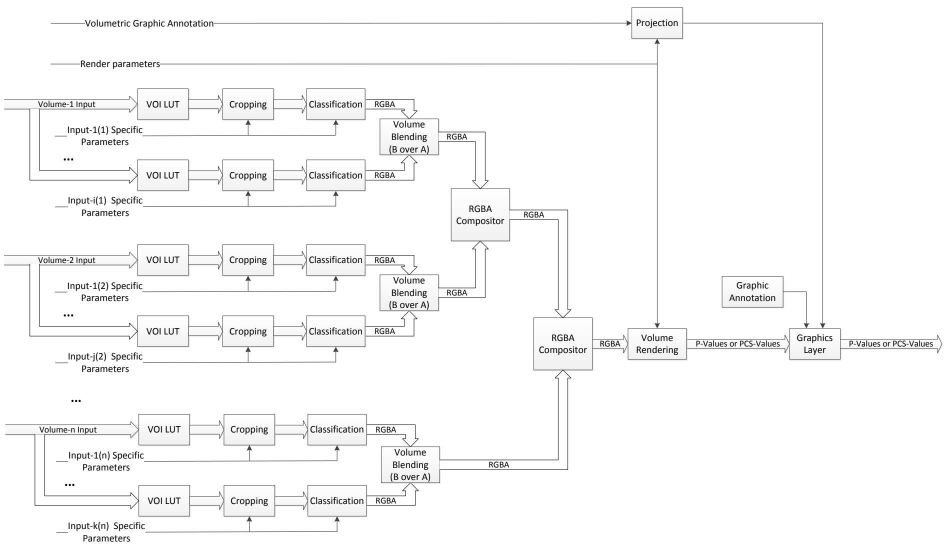

Figure X.2-1c: Multiple Volume Rendering Volumetric Pipeline

The transformations defined in the Volume Rendering Volumetric Presentation State Storage SOP Classes replace those that may be defined in the Referenced Image SOP Instances. If a particular transformation is absent in a Volume Rendering Volumetric Presentation State Storage SOP Classes then it shall be assumed to be an identity transformation, and any equivalent transformation, if present, in the Referenced Image SOP Instances shall NOT be used instead.

The presentation-related Attributes of the Volume Rendering Volumetric Presentation State Storage SOP Classes are immutable. They shall never be modified or updated; only a derived SOP Instance with a new SOP Instance UID may be created to represent a different presentation.

X.2.2.1 Volumetric Inputs, Registration and Cropping

A Volumetric Presentation State can take multiple volumes as input. A volume is defined in PS 3.3 section C.11.x2.1. The same source data can be referenced in more than one input.

The VOI LUT is applied to the input data.

The input volumes may or may not be in the Volumetric Presentation State Reference Coordinate System (VPS-RCS). If they are not, they shall be registered into the VPS-RCS.

The input volumes shall be cropped as specified by the value of Crop (0070,1204) and items in the Volume Cropping Sequence (0070,1301).

X.2.2.2 Volumetric Transformations

X.2.2.2.1 Volume Rendering Volumetric Presentation State

The Volume Rendering transformation requires a volume that is in the Volumetric Presentation State Reference Coordinate System (VPS-RCS).

X.2.2.3 Voxel Compositing

X.2.2.3.1 Voxel Compositing Foundation

The Volumetric Presentation State Display Module defines the algorithms used to transform the result of the volumetric processing on the input data into an output of P-Values or PCS-Values for display.

If Pixel Presentation (0008,9205) is MONOCHROME, then Presentation LUT Shape (2050,0020) provides the transform to output P-Values.

If Pixel Presentation (0008,9205) is TRUE_COLOR, then Presentation State Classification Component Sequence (0070,1801) describes the conversion of each processed input into an RGB data stream, and Presentation State Compositor Component Sequence (0070,1805) describes the compositing of these separate RGBA data streams into a single RGB data stream. This single RGB data stream is then processed as described by ICC Profile (0028,2000) to produce output PCS-Values.

X.2.2.3.1.2 Compositor Components

- RGBA Compositor: This component accepts two RGBA inputs and composites the data into a single RGB output and a single Alpha output.

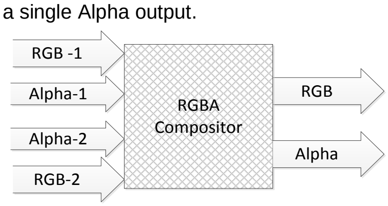

Figure X.2-5a: RGBA Compositor Component

X.2.2.3.1.3 Volume Rendering Transform Component

This component transforms an RGBA volume into a volume render view. This component is implementation dependent, but generally includes processing steps such as gradient computation to find normals of use in the shading operation, resampling of volume data, shading according to the parameters in the Render Shading module, and compositing of the resampled data to produce the final volume render view.

The transformation to P-Values or PCS-Values is performed after Volume Rendering to generate P-Values or PCS-Values output, respectively.

X.2.2.3.2 Internal Structure of Components

X.2.2.3.2.2 Internal Structure of RGBA Compositor Component

Weighting transfer functions that compute the weighting factors used by the Compositor Function as a function of $Alpha_1$ and $Alpha_2$ values are specified as weighting look-up tables (LUTs) in the RGB and Opacity Compositor component:

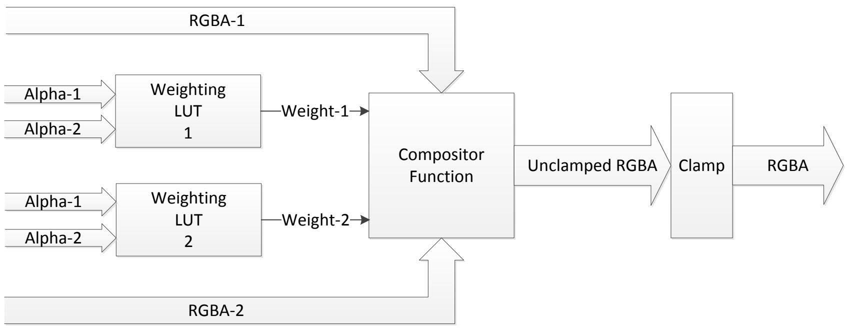

Figure X.2-8: Internal Structure of RGB and Opacity Compositor Component

Because each Weighting LUT uses both Alpha values in determining a weighting factor, they allow compositing functions that would not be possible if each weighting factor were based only on that input's Alpha value. See PS 3.17 Section Y.5 for typical usage of the Weighting LUTs.

The input bits to the Weighting LUTs are obtained by combining the two Alpha inputs, with half the input bits obtained from each Alpha input:

- In the case of the first compositor component corresponding to the first item in Presentation State Compositor Component Sequence (0070,1805), the Alpha from the classification component corresponding to the first item in the Presentation State Classification Component Sequence (0070,1805) provides the most significant bits of the Weighting LUT inputs, while the Alpha from the classification component corresponding to the second item in the Presentation State Classification Component Sequence (0070,1805) provides the least significant bits of the Weighting LUT inputs.
- In the case of subsequent compositor components, the Alpha from the classification component corresponding to the next item in the Presentation State Classification Component Sequence (0070,1805) provides the least significant bits of the Weighting LUT inputs, while the most significat bits of the Weighting LUT inputs are computed as one minus the Alpha from the classification component corresponding to the next item in the Presentation State Classification Component Sequence (0070,1805).

The integer outputs of the Weighting LUTs are normalized to the range 0.0 to 1.0, and the Compositor Function combines the normalized R, G, B, and A (each component called "Color" = C_x) input values as follows:

$$C_{out} = (C_1 * Weight_1) + (C_2 * Weight_2)$$

The sum of the normalized $Weight_1$ and $Weight_2$ shall be no greater than 1.0.

The color input values are normalized because the number of output bits from the RGB Palette Color Lookup Tables and the Alpha Lookup Tables may be different in each classification component.

The output of the compositor shall be range-limited ("clamped") to ensure that the outputs are guaranteed to be within a valid range of color and alpha values regardless of the validity of the weighting transfer functions. This isolates subsequent compositor components, the Volume Rendering Transform component and the Profile Connection Space Transform from overflow errors.

Changes to NEMA Standards Publication PS 3.6
Digital Imaging and Communications in Medicine (DICOM)
Part 6: Data Dictionary

Item #13: Add the following rows to PS3.6 Section 6

6 Registry of DICOM data elements

Note: For attributes that were present in ACR-NEMA 1.0 and 2.0 and that have been retired, the specifications of Value Representation and Value Multiplicity provided are recommendations for the purpose of interpreting their values in objects created in accordance with earlier versions of this standard. These recommendations are suggested as most appropriate for a particular attribute; however, there is no guarantee that historical objects will not violate some requirements or specified VR and/or VM.

Tag	Name	Keyword	VR	VM
(0070,x601)	Render Style		CS	1
(0070,x602)	Render Projection		CS	1
(0070,x603)	Viewpoint Position		FD	3
(0070,x604)	Viewpoint LookAt Point		FD	3
(0070,x605)	Viewpoint Up Direction		FD	3
(0070,x606)	Render Field of View		FD	6
(0070,x701)	Shading Style		CS	1
(0070,x702)	Ambient Reflection Intensity		FD	1
(0070,x703)	Light Direction		FD	3
(0070,x704)	Diffuse Reflection Intensity		FD	1
(0070,x705)	Specular Reflection Intensity		FD	1
(0070,x706)	Shininess		FD	1
(0070,xA06)	Swivel Range		FD	1
(0070,xB01)	Volume Stream Sequence		SQ	1
(0070,xB02)	Volumetric Presentation Input Source UID		UI	1

Item #14: Add the following rows to PS3.6 Annex A Table A-1

UID Value	UID Name	UID Type	Part
1.2.840.10008.5.1.4.1.1.11.x3	**Volume Rendering Volumetric Presentation State Storage SOP Class**	**SOP Class**	**PS 3.4**

1.2.840.10008.5.1.4.1.1.11.x4	Segmented Volume Rendering Volumetric Presentation State Storage SOP Class	SOP Class	PS 3.4
1.2.840.10008.5.1.4.1.1.11.x5	Multiple Volume Rendering Volumetric Presentation State Storage SOP Class	SOP Class	PS 3.4

Changes to NEMA Standards Publication PS 3.17-2011
Digital Imaging and Communications in Medicine (DICOM)
Part 17: Explanatory Information

Item #15: Append to Section Y.3

Y.3.X Highlighting Areas of Interest in Volume Rendered View

Y.3.X.1 User Scenario

A tumor in a volume has been identified and segmented. In volume render views the tumor is highlighted while preserving information about its relationship to surrounding anatomy.

Figure Y.3.X-1
Volume Rendered View with Tumor Highlighted

Y.3.X.2 Encoding Outline

The tumor is stored in a Segmentation IOD instance. A single Segment marks the corresponding area in the volume.

To create a Volume Rendered view which shows the tumor in a different color the VR VPS IOD instance defines via the Volumetric Presentation State Display Module a volumetric pipeline with 2 inputs.

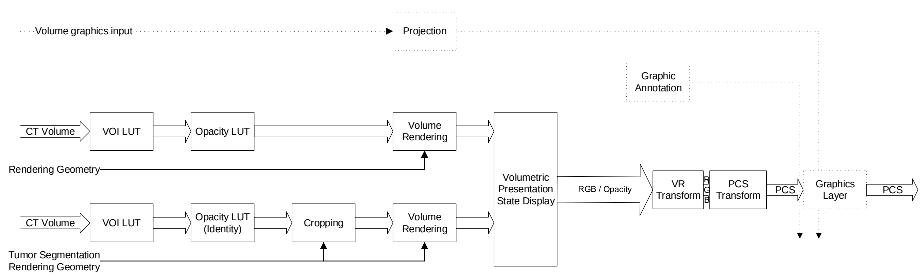

Figure Y.3.X-2
Planar MPR VPS Pipeline for Colorizing the Lung Nodule Categories

The same CT volume data set is used as input for all sub pipelines:

The first input to the Volumetric Presentation State (VPS) Display module provides the full (uncropped) volume rendered view of the anatomy in the display. This will provide the backdrop to the segmented inputs to be subsequently overlaid by compositor components of the Volumetric Presentation State Display pipeline.

The same input data and a single set of render geometry parameters defined in the Rendered Geometry module are used to generate each VPS Display module input; only the cropping is different. The Volume Cropping module for each of the other inputs specifies the included segment used to crop away all parts of the volume which do not belong to the tumor.

From this cropped volume Volume Rendered views are generated, which are then composited with the unsegmented input within the Volume Presentation State Display module. (see PS3.4 Section X.2.2.3.1.1).

In the Volumetric Presentation State Display Module the Presentation State Classification Component Sequence (0070,1801) defines scalar-to-RGB-and-Opacity transformations for mapping each voxel to RGBA and Opacity

For the volume rendered tumor view an RGB and opacity transfer function maps each voxel included in the segment to a fixed color value and maximum opacity.

Presentation State Compositor Component Sequence (0070,1805) in the Volumetric Presentation State Display Module then creates a RGB Compositor Component which composite the two sets of voxels into one. The first RGB Compositor performs “Partially Transparent A over B” compositing as described in Section Y.5.2 by passing through the Alpha of input 2 as Weight-2 and 1-Alpha of input 2 as Weight-1.

Figure Y.3.X-3 shows the complete pipeline for the lung nodule example:

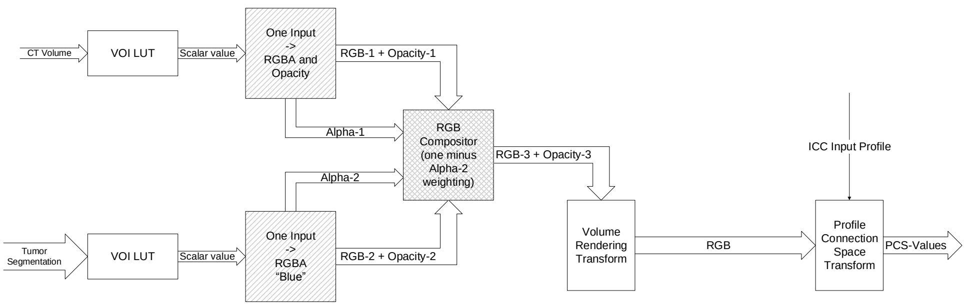

Figure Y.3.X-3
Tumor highlighting example pipeline

It is envisioned that display applications provide user interfaces for manipulating the Alpha LUT Transfer Functions for each input of the pipeline, allowing the user to control the visibility of the highlighting of the tumor.