Digital Imaging and Communications in Medicine (DICOM)

Supplement 228: Web Services and Protocol IOD for Volumetric Rendering

Prepared by:

DICOM Standards Committee, Working Group 27: Web Technologies

1300 N. 17th Street, Suite 900

Rosslyn, Virginia 22209 USA

Status: Public Comment, 28 June 2023

Developed pursuant to DICOM Work Item 2020-12-B

Table of Contents

1

Open Issues

1.	Would oblique planes be useful for user agents to modify anatomic orientation in addition to the perpendicular planes defined in Section 8.3.5.3.5? Should a set of projections be established based on clinical vocabulary to express oblique projections? Examples could be based on a common clinical vocabulary such as cardiac cath. projections (RAO, LAO, cranial, caudal) or a subset based on SNOMED (SCTID: 260419006). This supplement does not introduce oblique anatomic planes, as this would introduce a large number of projections, increasing complexity. Public comment is sought on this approach.
2.	What approach should be taken for selection of multi-phasic inputs and multi-volume rendering? This supplement currently includes 2 parameters, Volume Input Reference (Section 8.3.5.3.1) and Matching Attributes (Section 8.3.5.3.2) to facilitate the identification of subsets of inputs. The Volume Definition Module within the Volumetric Protocol (Section A.XX.3) also provides a means to identify and order multiple volumes. Are there other parameters or options that should be considered, such as: • Introduce a Volume Definition Object (similar to KOS)? • Add a service to identify instances that meet the Volume Input Requirements in section C.11.23.1 of PS3.3? • Let the origin server choose? • Some combination of solutions?
3.	Are the definitions sufficient in PS3.3 Section 3.17 “Multi-dimensional Definitions” of this supplement? Other terms are defined in the Scope and Field of Application of Supplements 156 and 190.
4.	Is the proposed approach for animation in web services sufficient? There is a strong case to include animation query parameters (e.g., rotate a 3D object to create a movie for ppt or multimedia report), however, the Presentation Animation Module may be too complex for some user agents. This supplement proposes the following parameters: Volumetric Curve Point Coordinates (8.3.5.3.11), Animation Step Size (8.3.5.3.12), and Animation Rate (8.3.5.3.13). Likewise, the Volumetric Rendering Protocol could be referenced by a user agent, as it includes the Presentation Animation module. Public comment is sought on this approach.

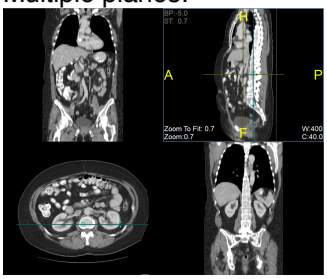
5.	Rendered MPR Volume and Rendered 3D Volume Resources limit the Target Resource to “either a Volumetric Presentation State Instance, or a collection of Image Instances or frames within Image Instances”. Should this definition be extended (potentially in a separate work item) to include the Surface Segmentation IOD, or should Segmentation IODs only be referenced in Volumetric Presentation States as annotation labels for segmentations of the volume data? See PS3.4, Section FF.2.4.1. Should this definition be extended to include encapsulated STL, OBJ, polygon mesh object types?
6.	Is the proposed approach of only specifying camera orientation parameters sufficient? In this supplement, camera orientation parameters (i.e., “viewpointposition”, “viewpointlookat”, or “viewpointup”) apply to rendered 3D and rendered MPR. This is inconsistent with the approach for Planar MPR Volumetric Presentation State IODs that specify orientation of the MPR slab as a direction cosine (x,y,z), in the MPR View Width Direction (0070,1507) and MPR View Height Direction (0070,1511) attributes. Using only camera orientation parameters simplifies parameters for the client, but diverges from the pattern in PS3.3, in which Volume Rendering and MPR each have their own orientation attributes. Public comment is sought on this approach.
7.	Is the proposed approach for zoom sufficient? Since the existing viewport scaling can be used for a 2D zoom, this supplement does not include query parameters corresponding to Render Field of View (0070,1606) or MPR View Width (0070,1508) and MPR View Height (0070,1512) attributes. Public comment is sought on this approach.
9.	The Rendered Volume Resource payload (Section 10.4.3.3.7) provides for an optional metadata representation in addition to the 2D Rendered Media Type. The metadata representation is enabled by the parameter "volumetricmetadata" and defined in a Response Message Body in Section X., an example is also included in Section B. Is the proposed approach of returning rendered attributes sufficient?
10.	Web services allow latitude in server behavior for (e.g. apply a default behavior, or return an error, see Section 8.3.5.3.3 for an example). To what extend such behavior need to be documented in the Conformance Statement? See Proposed modifications in Table PS3.2 N.5-76 of this Supplement for more information.

2

Closed Issues

1.	How to address tradeoffs in quality vs. size of video for lightweight devices (addressed with existing quality / scaling parameters) . **Response:** The existing quality parameter addresses this.
2.	What is the desired level of “interactivity” for the basic user agent? **Response:** pan, zoom, rotate, animate, windowing, set rendering method
3.	Do we need to support Volume Rendering protocols that can be applied in hanging protocols for a zero footprint viewer? **Response:** This is addressed in the Protocol IOD.
4.	What is the level of Ultrasound support? **Response:** MPR should address most US needs.
5.	How is vendor proprietary binary data supported **Response:** Vendor proprietary binary data is out of scope.
6.	Parameters potentially conflicting with Volumetric Protocol include render method, slab thickness and annotation. PS3.18 prescribes origin server behavior for conflicting presentation state parameters (Frame Number, Source Image Region, or Windowing) Should the Rendered Volume be as strict? Options include: 1. 1st apply the Protocol and 2nd apply overriding parameters? 2. Let the server choose to apply overriding parameters that are not in conflict 3. Return an error? **Response:** This supplement supports all 3 options.
7.	Should Anatomical Orientation, Viewpoint coordinates (3D) or MPR coordinates (MPR) be accepted as an overriding parameter for rendering a Volumetric Presentation State? This is inconsistent with the pattern of the 2D presentation state render behavior, however, could be useful to allow the user agent to set the desired orientation of a Volumetric Presentation State. One could also argue that, in Volume Rendering, the orientation is moving the camera and not changing the rendered object (resource). See 8.3.5.3.5 **Response:** Yes, Volume Rendering orientation is moving the camera and not changing the rendered object.
8.	Should either the Volumetric Protocol UID or the Volumetric Presentation State UID required as a mandatory parameter? **Response:** No, instances and frames are also allowed as a Target Resource.

9.	Should this service be extended to the URI Web Service PS18 9.x? or is it sufficient to limit it to the Retrieve Transaction of the Studies Web Service? **Response:** No, Retrieve Transaction Rendered MPR Volume Resources and Retrieve Transaction Rendered 3D Volume Resources apply to Studies Service and Resources.
10.	Is the approach for return media sufficient for multi-phasic volumes? Does the user agent need to know the number of images that will be returned in a multipart response? **Response**: Return media is be based on the Target Resource and rendering instructions (e.g., animated vs static rendering). See examples in this supplement.
11.	Is the proposed text in PS18 8.11 sufficient to address recognizable visual features that could be displayed by volumetric rendering? **Response:** Yes, this approach was deemed sufficient by WG-06 and WG-14.
12.	How should 4D animation (beating heart) be addressed? This is currently addressed in the Presentation Animation Style, PRESENTATION_SEQ in the Presentation Animation Module. **Response**: Assume 4D animation is desired when user agent selects multi volume input instances. Otherwise a simple volume resource would be selected.
13.	Should multiple static volumes (e.g., merging CT and PET) be supported in the basic case or reserved for advanced use cases? How should transparency be addressed? **Response**: This is out of scope for the basic user agent and handled by Volumetric Presentation States in the Multiple Volume Rendering Volumetric Presentation State Storage IOD.
14.	Should viewport scaling be allowed as an overriding parameter for Volumetric Presentation States? **No**
15.	Is it appropriate to apply an iccprofile to a color Volume Rendering? **Response**: Color space for the rendered image is not defined by DICOM. ICC Profile parameters may be embedded in compatible media formats returned by the origin server, however due to variation in applications that support ICC profiles, there is no guarantee of a standardized color space for rendered images.

16.	Does multi-planar reformat (MPR) describe one view of a reformatted plane or multiple synchronized views of multiple planes? The answer influences MPR parameters Single plane:  Multiple planes:  **Response**: The de-facto definition MPR refers to a single plane. This plane can be reconstructed in one of several arbitrary planes. The MPR endpoint returns a single planar reformat. The client may create a display (or hanging protocol) consisting of multiple spatiality related planar reformats. See PS3.17 XXX.3.2.1, which states “Planar MPR views are often displayed together with other spatially related Planar MPR views”.
17.	Should this service be extensible? For example, could an origin server offer a parameter to invoke a post processing application, such as automatic bone removal? **Response**: Not beyond the existing capability in PS3.18 Section 8.3 that allows Origin Servers to define additional query parameters.
18.	Should there be an API to save a Volumetric Presentation State, or is the existing Store Transaction (a.k.a. STOW) sufficient? **Response**: Store transaction is sufficient and minimizes complexity.
19.	The approach for the advanced use case in this supplement is to 1) utilize STOW to define Volumetric Rendering parameters by creating a Volumetric Presentation State IOD and 2) utilize the Retrieve Transaction for Rendered 3D/MPR Volume Resources. Is this an acceptable approach? **Response**: This supplement focuses on the basic use case. If there is additional support needed for advanced cases, they can be added later.
20.	Should 3D and MPR be in the same service, or distinct? This could be useful for conformance. **Response**: 3D and MPR has been separated into two services based on recommendation from WG-06.

21.	Should there be a patient instance based on the Volumetric Protocol, or is the Volumetric Presentation sufficient? This would be similar to a CT/XA Performed Procedure Protocol Storage Protocol that is applied to a Study. **Response**: The Volumetric Presentation state is a Patient Instance that suits this use case.
22.	This supplement establishes separate services for 3D and MPR study, series, instances and frames. The Volume Input Reference and Volume Input Criteria query parameters may be used to identify Volume Data temporal acquisitions across several series within a study. What constraints should be added to the study resource for usability? **Response**: This will be addressed in cp1978.
23.	Can any of the viewpoint values be left empty, or should they all be required as written (as in viewport scaling)? **Response**: Camera position query parameters were initially combined into a single parameter called “viewpoint”. This has been split into three parameters (Viewpoint Position, Viewpoint LookAt and Viewpoint Up) in order to allow the client to modify only one parameter if needed.
24.	Should this supplement establish the “volume” or the “input instances” as the Target Resource? This is important for clarity of the supplement and to differentiate 2D from 3D rendered resources for implementers. **Response:** Input Instances were kept as the Target Resource for consistency with the Volumetric Presentation States. “Volume Data” is referred to in a manner consistent with the pipeline for Volumetric Presentation states. This supplement also establishes a 4-step rendering pipeline: 1. Identify input instances that meet Volume Input Requirements, 2. resample input instances as Volume Data, 3. apply display algorithm and presentation parameters to the Volume Data, and 4. return new images as Acceptable Media Types in the response payload.
25.	Can volumetric rendering protocols be patient specific? Should the Volumetric Rendering Protocol include the Patient Specification Module as User Optional? **Response:** Volumetric Rendering Presentation states are patient specific.
26.	Should there be a resource that exposes the organized Volume Data as a bulkdata resource? **Response:** Not in this work item, but it could be considered for a future work item if there is interest.

3

Scope and Field of Application

This supplement introduces web services and a Volumetric Rendering Protocol IOD that facilitate volumetric rendering without having to specify the numerous and complex parameters required to do so.

The web services enable a user agent to request server-side 3D volumetric rendering. In this service, input instances with geometric consistency are reconstructed into volume data. Algorithm and display parameters are applied to the volume data in order to achieve the requested presentation, and lastly, the representation is encoded into one or more images of the requested media type and returned in a response payload to the user agent.

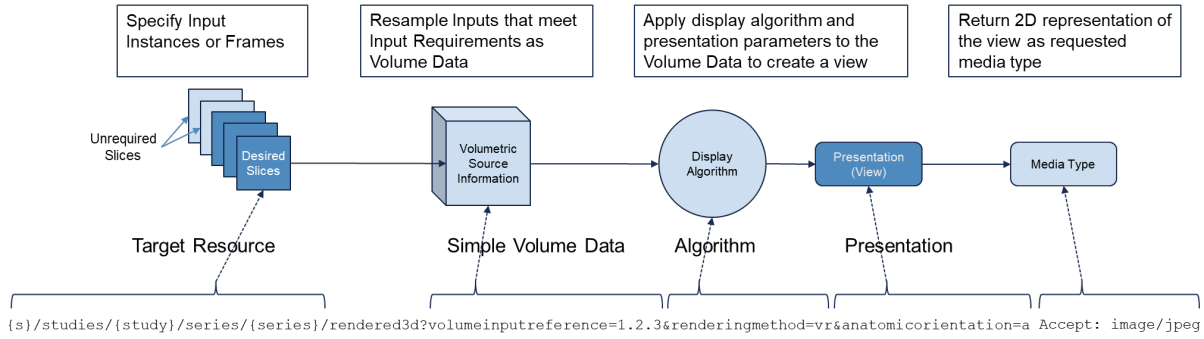


Figure 0.1 DICOMweb API Volumetric Rendering Pipeline

The Volumetric Rendering Protocol IOD is a non-patient instance belonging to the family of Defined Procedure Protocol IODs that organizes image set inputs into Volume Data, and specifies the Volumetric Transformations to be applied.

In this supplement, volumetric rendering refers to 3D volume rendering (VR), 3D MIP (maximum intensity projection), and multiplanar planar (MPR) rendering methods.

Use Case

For example, an ER stroke patient is referred for a CT. Non-contrast and CT angiogram images are acquired to rule out hemorrhage and intravascular thrombus, respectively. Images are reviewed on a zero-footprint viewer by ER physician. The viewer includes a hanging protocol that displays lossless JPEG-LS multi-frame images of thick slab axial MIPs of the CTA. This is the result of a RESTful service request with a pre-identified rendering mode, slab thickness and spacing. The ER physician interrogates images that best demonstrate the Circle of Willis.

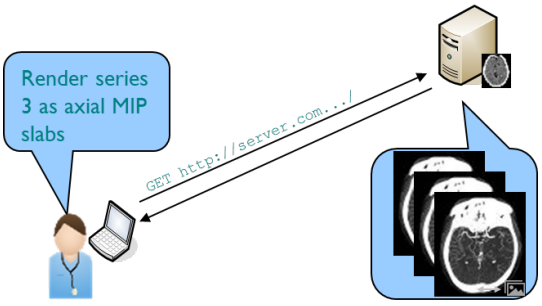


Figure 0.2 DICOMweb Volumetric Rendering Transactions

Relationship to Volumetric Presentation State

DICOMweb™ API for Server Volumetric Rendering can be used independently to render a volume of Input Instances at the time of image interpretation, or to render a Volumetric Presentation State, containing rendering parameters, presentation, graphic annotations, animation, cropping and segmentation that were performed prior to image interpretation.

User Agent Types

This supplement has considered two categories of user agents:

1. the advanced user agent, capable of defining and rendering the full breadth of parameters contained within DICOM Volumetric Presentation State IODs and Volumetric Rendering Protocol IODs, and
2. the basic user agent, capable of fundamental operations to identify the rendering type, select a rendering protocol, or manipulate the volumetric view and transformations.

The advanced use case is addressed in the creation and rendering of Volumetric Presentation States.

This supplement focuses on the requirements that satisfy the interoperability needs for the basic user agent with latitude for the origin server to apply implementation-specific default behaviors to handle aspects of the volumetric rendering pipeline that are not specified by the client and/or have not been standardized by industry.

Table 0-1. Examples of Functionality per User Agent Type

Basic Functions	Advanced Functions
• Pan • Zoom • Windowing • Set Quality • Rotate • Animate • Set Render Method	Available by referencing a Volumetric Presentation State or a Volumetric Rendering Protocol: • Display Color • Shading and Lighting • Crop • Merge Multiple Volumes • Define a Composite MPR • Annotate • Render projection or orthographic view • Render endoluminal view (i.e., fly through)

Invocation of implementation-specific post-processing applications often associated with 3D and 4D that could be negotiated between the user agent and origin server are out of scope for this supplement (e.g., vessel tracking, bone removal, deformable registration, etc.). PS3.18 Section 8.3 currently supports extensible, implementation-specific query parameters for this purpose.

DICOMweb API Volumetric Rendering Pipeline Description

Input Instances

In medical imaging, a volumetric rendering application requires 2D slice data input. In this supplement, each 2D component of that input is referred to as a frame. A frame is either a Legacy IOD Image Instance, or a frame within an Enhanced IOD Image Instance. Rendering algorithms depend on a degree of consistency within the collection of slice data: for example, a common patient frame of reference, pixel attributes (rows, columns, bit depth) and spatial alignment. Slices may possess Z-axis overlap and/or gaps but cannot contain duplicates.

The frames from which a volume is composed are typically generated by a modality. The IODs for those instances provide some flexibility in the organization of frames. As a result, frames may be organized in a single series or multi-frame container, or a single series or multi-frame container may include one or more collections of frames. DICOM defines the requirements for collections of frames that make up Volumetric Source Information in the Presentation Input Type Volume Input Requirements in PS3.3, Section C.11.23.1. For the simplest basic user agent use case, a Single Frame IOD Image Instances for a given volume are typically collected within a series, and Multi-frame IOD frames are typically collected within a single Instance.

Notes

1. A Single Frame IOD is an IOD that is not multi-frame, such as the legacy CT Image IOD.
2. A Multi-frame IOD is an IOD that is multi-frame, such as the Enhanced CT Image IOD.

Table 0-2 Identification of Volume Input Data

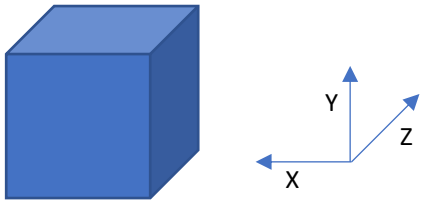
Source IOD Type	Target	Target Description	Target Resource
Multi-frame	Instance	An instance containing only a set of frames that satisfy Volume Input Requirements.	/studies/{study}/series/{series}/instances/{instance}
Multi-frame	Instance	An instance containing a set of frames with a subset that satisfies Volume Input Requirements. The subset is explicitly identified by the user agent.	/studies/{study}/series/{series}/instances/{instance}/frames/{frames}
Multi-frame	Instance	An instance containing a set of frames with a subset that satisfies Volume Input Requirements. The subset is identified by the origin server based on characteristics provided by the user agent.	/studies/{study}/series/{series}/instances/{instance}
Volumetric Presentation State	Instance	An instance containing references (in the Volumetric Presentation Input Set Sequence) to a set of frames that satisfy the Volume Input Requirements.	/studies/{study}/series/{series}/instances/{instance}
Single Frame	Series	A series containing only a set of instances that satisfy Volume Input Requirements.	/studies/{study}/series/{series}
Single Frame	Series	A series containing a set of instances with a subset that satisfies Volume Input Requirements. The subset is identified by the origin server based on characteristics provided by the user agent.	/studies/{study}/series/{series}

Volume Data

Simple Volume Data

Volumetric Source Information is used to compose volume data. Simple volume data consists of a single set of frames that meet the Presentation Input Type Volume Input Requirements (see Section C.11.23.1 of PS3.3). A simple volume is also referred to as 3D, in which each of the three dimensions represent a spatial axis (x, y and z).

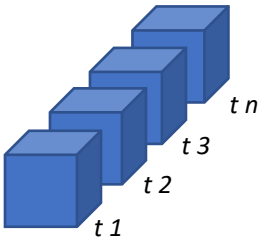
Figure 0.3 Three Dimensional Simple Volume



Multi-Volume Data

Multi-volume data consists of two or more simple volumes that are temporally related and rendered simultaneously. Each time point is represented as a simple volume that meets the Volume Input Requirements in Section C.11.23.1 of PS3.3.

Figure 0.4 Four Dimensional, Multi-Volume



Display Algorithm and Rendering Consistency

Softcopy DICOM presentation states (Grayscale, Color, Pseudo-Color, Blending, Advanced Blending, XA /XRF Grayscale Softcopy) are intended to maintain repeatable, consistent presentation of contrast and brightness, spatial and graphical operations between different hardcopy and softcopy devices.

Unlike Softcopy Presentation States, Volumetric Presentation States (Grayscale Planar MPR, Compositing Planar MPR, Advanced Blending, Volume Rendering, Segmented Volume Rendering, Multiple Volume Rendering) and the DICOMweb APIs for Server Volumetric Rendering describe the process of creating a new image rather than the application of parameters for displaying an existing one. Various volumetric rendering methods (e.g., Raycasting, Splatting, Shear-warp and 3D texture-mapping hardware-based approaches[1]) influence implementation-specific algorithm(s) and inevitably result in differences in the appearance of resulting images. For this reason, display consistency to the level offered in softcopy presentations states is not expected, rather reasonable consistency is provided by specification of inputs, geometric descriptions of spatial views, type of processing to be used, color mapping, and many generic rendering parameters in order to produce a clinically acceptable result[2].

Presentation

Presentation parameters define a single or initial view and optional animation parameters.

In a static view, a simple volume is rendered, presented in a fixed orientation and returned as a single frame media type. This is the most basic rendering, achieved through simple query parameters.

In an animated view, a simple volume is rendered. Animation is achieved by sequentially modifying the view to show animation (e.g., rotate around an axis) and returned as a multi-frame image or video media type. Even with a simple volume, animation has the potential to add several complex query parameters.

Post processing applications are capable of creating elaborate animations capable of combining fly-through, rotation, pan, zoom, transparencies, blending etc. To this end, the Volumetric Presentation state supports input sequence, presentation sequence, cross curve, fly-through and swivel. This supplement introduces a subset of animation parameters that specify volumetric curve point coordinates, the animation step size, and the animation rate. For more complex animations, such as fly-through, the Volumetric Presentation Animation Module within the Volumetric Rendering Protocol or Volumetric Presentation State should be used.

[1] http://dx.doi.org/10.1109/VV.2000.10009
[2] https://dicom.nema.org/medical/dicom/current/output/html/part17.html#sect_XXX.1

Returned Images

In the last step of the pipeline, new images are returned as Acceptable Media Types in the response payload. Table 0-3 defines the return media types for each presentation category based on the established Resource Categories in PS3.18.

Table 0-3. Volumetric Rendering Resource Categories

Volumetric Presentation	**Resource Category** (PS3.18 Table 8.7.2-1)	**Return Media Type** (PS3.18 Table 8.7.4-1)	**Example**
Static	Single Frame Image	image/jpeg image/gif image/png image/jp2	Lateral projection of a 3D ankle
Animated	Multi-Frame Image	image/gif	Rotating 3D ankle
Animated	Video	video/mpeg video/mp4 video/H265	Rotating 3D ankle

Volumetric Rendering Query Parameters

Query parameters are derived from existing Volumetric Presentation State attributes and used to identify Volumetric Source Information, control image set inputs, algorithm and presentation. Presentation States Parameter compatibility is designed for consistency with current PS3.18 `/rendered` overriding parameters for Presentation States and summarized in Table 0-4.

Table 0-4. Retrieve Rendered Volume Parameter Compatibility

Parameter	**Pipeline Reference** Figure 0.1	**Instances or Frames rendered as 3D**	**Instances or Frames rendered as MPR**	**Volume Rendering Presentation States**	**MPR Presentation States**
accept	presentation	p	p	p	p
annotation	presentation	p	p	p	p
charset	presentation	p	p	p	p
quality	presentation	p	p	p	p
viewport	presentation	p	p	p	p
window	presentation	p	p		
iccprofile	presentation	p	p		
volumeinputreference	volume data definition	p	p		
match	volume data definition	p	p		
volumetricprotocol	volume data definition, volumetric algorithm, volumetric presentation	p	p		
renderingmethod	volumetric algorithm	p	p		
orientation	volumetric presentation	p	p	p	p
viewpointposition	volumetric presentation	p	p	p	p
viewpointlookat	volumetric presentation	p	p	p	p
viewpointup	volumetric presentation	p	p	p	p
mprslab	volumetric presentation		p		
swivelrange	volumetric presentation	p	p		

volumetriccurvepoint	volumetric presentation	p	p		
animationstepsize	volumetric presentation	p	p		
animationrate	volumetric presentation	p	p		
volumetricmetadata	volumetric presentation	p	p		

Notes

1. Cells denoted with “p” represent a query parameter that is permitted for a given Target Resource
2. Empty cells represent query parameters that are unacceptable for a given Target Resource
3. Orientation and viewpoint may override the Volumetric Presentation State since the orientation of the camera changes with respect to the scene.
4. Volumetric parameters may be only applied by systems capable of performing and processing volumetric rendering.

Protocols

To avoid complexity for the basic user agent, the request URI does not include parameters defining 3D geometry, shading of display. Alternatively, the user agent may include a parameter to identify a non-patient instance Volumetric Rendering Protocol, or the origin server may apply a default protocol. Thumbnails are incorporated in the Volumetric Rendering Protocol IOD to facilitate selection.

It is anticipated that protocols will be managed in a manner consistent with exiting post-processing applications: the manufacturer provides a set of default protocols, the operator identifies a desired protocol by selecting a representative thumbnail view (typically selected by trial and error until the desired protocol is identified). Operators tend to rely on vendor supplied protocols and may occasionally apply minor modifications to the vendor protocols to suit their preference. Other than selection and the existing Store Transaction, this supplement does not introduce any additional web services to manage Volumetric Rendering Protocols.

Security

Other than potentially rendering recognizable visual features, Web services for Volumetric Rendering introduce no new security considerations. Please refer to controls for access control, authorization, and auditing addressed in PS3.15.

Modifications to PS3.18

Update PS3.18 Table 10.2-2 Resources by Transaction as follows:

In Table 10.3-2, the Target Resources permitted for each transaction are marked with M if support is mandatory for the origin server and O if it is optional. A blank cell indicates that the resource is not allowed in the transaction.

Table 10.3-2. Resources by Transaction

Resource	Retrieve	Store	Search
Studies Service			
All Studies		M	M
Study	M	M	
Study Metadata	M		
Study Bulkdata	O		
Study Pixel Data	O		
Rendered Study	M		
Rendered MPR Volume Study	**O**		
Rendered 3D Volume Study	**O**		
Study Thumbnail	O		
Study's Series			M
Study's Instances			M
All Series			M
Series	M		
Series Metadata	M		
Series Bulkdata	O		
Series Pixel Data	O		
Series' Instances			M
Rendered Series	M		
Rendered MPR Volume Series	**O**		
Rendered 3D Volume Series	**O**		

Series Thumbnail	O		
All Instances			M
Instance	M		
Instance Metadata	M		
Instance Bulkdata	O		
Instance Pixel Data	O		
Rendered Instance	M		
Rendered MPR Volume Instance	**O**		
Rendered 3D Volume Instance	**O**		
Instance Thumbnail	O		
Frames	M		
Rendered Frames	M		
Rendered MPR Volume Frames	**O**		
Rendered 3D Volume Frames	**O**		
Frame Thumbnail	O		
Bulkdata	M	M	

Add the following Sections after Section 10.4.1.1.6:

10.4.1.1.7 Rendered MPR Volume Resources

Rendered MPR Volume Resources (defined in Table 10.4.1.7-1) perform multiplanar reformatting, in accordance with the principles established for Planar MPR Volumetric Presentation States (see PS3.4, Section FF.2.1.1), representing a cross-section of a volume of slice data as an Euclidean plane. Rendered images are returned as Acceptable Media Types in the response payload.

Note

Reasonable consistency is provided to produce what is expected for a clinically acceptable result, however, there will be differences in origin server outputs.

The Target Resource shall be either a Planar MPR Volumetric Presentation State Instance, or a collection of Image Instances or frames within Image Instances, that conform to the Volume Input Requirements in Section C.11.23.1 of PS3.3.

Note

The Target Resource may be further refined using query parameters defined in Section 8.3.5.3 to meet Volume Input Requirements for Rendered Volume Resources (see PS3.3, Section C.11.23.1)

See Section 8.3.5.3 Query Parameters For Rendered Volume Resources.

Table 10.4.1.7-1. Retrieve Transaction Rendered MPR Volume Resources

Resource	URI Template
Rendered MPR Volume Study	/studies/{study}/renderedmpr
Rendered MPR Volume Series	/studies/{study}/series/{series}/renderedmpr
Rendered MPR Volume Instance	/studies/{study}/series/{series}/instances/{instance}/renderedmpr
Rendered MPR Volume Frames	/studies/{study}/series/{series}/instances/{instance}/{frames}/renderedmpr

10.4.1.1.8 Rendered 3D Volume Resources

Rendered 3D Volume Resources (defined in Table 10.4.1.8-1) perform rendering in accordance with the principles established for Volume Rendering Volumetric Presentation States (see PS3.4, Section FF.2.1.2), by applying thresholding, ray-casting, volume rendering, or other methods to display a volume of slice data as a three dimensional projection. Rendered images are returned as Acceptable Media Types in the response payload.

Note

Reasonable consistency is provided to produce what is expected for a clinically acceptable result, however, there will be differences in origin server outputs.

The Target Resource shall be either a Volume Rendering Volumetric Presentation State Instance, or a collection of Image Instances or frames within Image Instances, that conform to the Volume Input Requirements in Section C.11.23.1 of PS3.3.

Note

The Target Resource may be further refined using query parameters defined in Section 8.3.5.3 to meet Volume Input Requirements for Rendered Volume Resources (see PS3.3, Section C.11.23.1)

See Section 8.3.5.3 Query Parameters For Rendered Volume Resources.

Table 10.4.1.8-1. Retrieve Transaction Rendered 3D Volume Resources

Resource	URI Template
Rendered 3D Volume Study	/studies/{study}/series/rendered3d
Rendered 3D Volume Series	/studies/{study}/series/{series}/rendered3d
Rendered 3D Volume Instance	/studies/{study}/series/{series}/instances/{instance}/rendered3d
Rendered 3D Volume Frames	/studies/{study}/series/{series}/instances/{instance}/{frames}/rendered3d

Modify Table 10.4.1-5. Query Parameters by Resource as follows:

10.4.1.2 Query Parameters

Table 10.4.1-5. Query Parameters by Resource

Key	Resources	Usage		Section
		User Agent	Origin Server	

accept	All Resources	O	M	Section 8.3.3.1
	Rendered Volume Resources	**O**	**O**	
charset	Metadata Resources	O	M	Section 8.3.3.2
	Rendered Volume Resources	**O**	**O**	
annotation	Rendered Resources	O	M	Section 8.3.5.1.1
	Rendered Volume Resources	**O**	**O**	
quality	Rendered Resources	O	M	Section 8.3.5.1.2
	Rendered Volume Resources	**O**	**O**	
viewport	Rendered Resources	O	M	Section 8.3.5.1.3
	Rendered Volume Resources	**O**	**O**	
	Thumbnail Resources	O	O	
window	Rendered Resources	O	M	Section 8.3.5.1.4
	Rendered Volume Resources	**O**	**O**	
iccprofile	Rendered Resources	O	O	Section 8.3.5.1.5
	Rendered Volume Resources	**O**	**O**	
volumeInputreference	**Rendered Volume Resources**	**O**	**O**	**Section 8.3.5.3.1**
match	**Rendered Volume Resources**	**O**	**O**	**Section 8.3.5.3.2**
volumetricprotocol	**Rendered Volume Resources**	**O**	**O**	**Section 8.3.5.3.3**
renderingmethod	**Rendered Volume Resources**	**O**	**O**	**Section 8.3.5.3.4**
orientation	**Rendered Volume Resources**	**O**	**O**	**Section 8.3.5.3.5**
viewpointposition	**Rendered Volume Resources**	**O**	**O**	**Section 8.3.5.3.6**
viewpointlookat	**Rendered Volume Resources**	**O**	**O**	**Section 8.3.5.3.7**
viewpointup	**Rendered Volume Resources**	**O**	**O**	**Section 8.3.5.3.8**
mprslab	**Rendered MPR Volume Resources**	**O**	**O**	**Section 8.3.5.3.9**
swivelrange	**Rendered 3D Volume Resources**	**O**	**O**	**Section 8.3.5.3.10**
volumetriccurvepoint	**Rendered MPR Volume Resources**	**O**	**O**	**Section 8.3.5.3.11**
animationstepsize	**Rendered Volume Resources**	**O**	**O**	**Section 8.3.5.3.12**
animationrate	**Rendered Volume Resources**	**O**	**O**	**Section 8.3.5.3.13**

volumetricmetadata	**Rendered Volume Resources**	**O**	**O**	**Section 8.3.5.3.14**

Add the following Section after 10.4.3.3.6 Pixel Data Resource Payload:

10.4.3.3.7 Rendered Volume Resource Payload

The payload for a Rendered 3D Volume Resource (see Section 10.4.1.1.7) or a Rendered MPR Volume Resource (see Section 10.4.1.1.8) shall contain:

- a 2D representation of the rendered volume according to the parameters of the display algorithm,

and may also contain;

- a Rendered Volume Resources Response Module (see Annex X) corresponding to the request.

If both are returned, the payload shall be a multipart payload, otherwise the payload shall be single part. See Section B.x2 for an example.

Add the following Section after Section 8.3.5.2:

8.3.5.3 Query Parameters For Rendered Volume Resources

Query Parameters defined in this section control the creation of new 3D or MPR images based on Volume Data identified by the Target Resource.

The following rules pertain to all parameters defined in this section:

1. All parameters are optional for the user agent.
2. Not all parameters are required to be supported by the origin server.
3. These parameters only apply to resources that are images.

The set of transformations specified by the parameters in this section shall be applied to the images as if the parameters were a Volumetric Presentation State, that is, in the order specified by the applicable image rendering pipeline specified in Section FF.2 of PS3.4.

Table 8.3.5-2 shows the Query Parameters that may be used when requesting a Rendered Volume Representation.

Table 8.3.5-2. Retrieve Rendered Volume Query Parameters

Key	Values	Target Resource Category	Section
`volumeinputreference`	`uid or frame`	Image (single or multi-frame)	8.3.5.3.1
`match`	`; See attribute matching rules in Section 8.3.4.1`	Image (single or multi-frame)	8.3.5.3.2
`volumetricprotocol`	`uid`	Image (single or multi-frame)	8.3.5.3.3
`renderingmethod`	`"volume_rendered", "maximum_ip" , "minimum_ip" or "average_ip"`	Image (single or multi-frame)	8.3.5.3.4
`orientation`	`"a", "p", "r", "l", "h" or "f"`	Image (single or multi-frame) or Volumetric Presentation States	8.3.5.3.5

viewpointposition	px , py , pz	Image (single or multi-frame) or Volumetric Presentation States	8.3.5.3.6
viewpointlookat	lx , ly , lz	Image (single or multi-frame) or Volumetric Presentation States	8.3.5.3.7
viewpointup	ux , uy , uz	Image (single or multi-frame) or Volumetric Presentation States	8.3.5.3.8
mprslab	st	Image (single or multi-frame)	8.3.5.3.9
swivelrange	sr	Image (single or multi-frame)	8.3.5.3.10
volumetriccurvepoint	px , py , pz	Image (single or multi-frame)	8.3.5.3.11
animationstepsize	ss	Image (single or multi-frame)	8.3.5.3.12
animationrate	rt	Image (single or multi-frame)	8.3.5.3.13
volumetricmetadata	"yes"	Image (single or multi-frame)	8.3.5.3.14

Rendered Volume Resources have two mutually exclusive options to determine the initial orientation of the resampled Volume Data:

1. The “orientation” parameter establishes the standard anatomic position of the patient as viewed by the camera, and
2. camera orientation parameters (“viewpointposition”, “viewpointlookat”, or “viewpointup”) establish the camera position and direction as it views the patient.

When incorporating animation parameters, the initial frame is established by orientation parameters. The parameters “swivelrange”, "volumetriccurvepoint" and “animationstepsize” dictate subsequent frames. When animating multiple sets of temporally related, spatially co-located Volume Data (such as a multiphase acquisition), the initial frame's displayed phase is determined by the origin server.

There is no parameter to control the type of projection used during rendering. The origin server shall use Orthographic projection for Rendered 3D Volume Resources. See Section C.11.30.1 in PS3.3.

There is no parameter to explicitly control Render Field of View, MPR View Height or MPR View Width (see Sections C.11.30 and C.11.26 in PS3.3). The “viewport” parameter can be used to scale the returned media. See Section 8.3.5.1.3.

8.3.5.3.1 Volume Input Reference

The “volumeinputreference” parameter identifies the Instance, or Frame within an Instance, from which the origin server shall extract characteristics and identify additional Instances or Frames in the Target Resource with the same values for those characteristics. The user agent uses this parameter to identify a desired subset when the Target Resource is a superset of the intended Volume Data. The origin server shall identify a subset that conforms to the Volume Input Requirements for Rendered Volume Resources (see PS3.3, Section C.11.23.1).

The syntax of this parameter for a multi-frame image is:

```
%s" volumeinputreference =" uid "," frame
```

Otherwise it is:

```
%s" volumeinputreference =" uid
```

Where

`uid`	Is the Unique Identifier of the Volume Input Reference SOP Instance when the Target Resource is a series or study.
`frame`	Is the frame number within an Image Instance when the Volume Input Reference is an Enhanced IOD Image Instance.

Note

`uid` corresponds to Referenced SOP Instance UID (0008,1155) and `frame` corresponds to Referenced Frame Number (0008,1160) See Section 10.3 in PS3.3.

The origin server shall create Volume Data from instances or frames having characteristics identical to the Volume Input Reference based on implementation-specific logic.

If any of the following is true:

- the Target Resource is a Presentation State,
- valid Volume Data is not found based on the Volume Input Reference,
- the UID is not found in the Target Resource,
- the frame is not found in the Target Resource,
- a Match Attribute/Value pair is present in another parameter in the request,

then the origin server shall return a 400 (Bad Request) and may include an appropriate Status Report.

8.3.5.3.2 Match

The “match” parameter specifies common DICOM Attribute/Value pair characteristics of the Volume Data.

When the user agent identifies a Target Resource that is a superset of the intended Volume Data, it may identify Attribute/Value pairs that specify matching criteria to identify specific Instances or Frames in the Target Resource to resample as Volume Data. The resulting subset shall conform to the Volume Input Requirements for Rendered Volume Resources (see PS3.3, Section C.11.23.1).

See Section 8.3.4.1 for the syntax of this parameter.

The user agent may include the following Attributes in the parameter:

- Instance IE Attributes
- Private Data Element Tags and their corresponding Private Creator Element Tags

The origin server shall reconstruct Volume Data meeting the Volume Input Criteria.

If any of the following are true:

- the Target Resource is a Volumetric Presentation State,
- valid Volume Data is not found based on the Attribute/Value pair,
- the “volumeinputreference” parameter is also present,

then the origin server shall return a 400 (Bad Request) and may include an appropriate Status Report.

8.3.5.3.3 Volumetric Protocol

The "volumetricprotocol" parameter allows a Volumetric Rendering Protocol instance to be referenced. Volumetric rendering parameter values are extracted from the protocol and applied to the Volume Data.

The syntax of this parameter is:

```
%s"volumetricprotocol =" uid
```

Where

`uid`	Is the Unique Identifier of a Volumetric Rendering Protocol SOP Instance. See PS3.3 Section 7.13.x.

The origin server shall retrieve the instance corresponding to the specified UID, extract Rendered Volume parameters from that instance and apply them to the Target Resource.

If other query parameters are also specified, the origin server may:

- apply parameters in the Volumetric Rendering Protocol SOP Instance and then apply non-conflicting query parameters,
- apply parameters specified in the Volumetric Rendering Protocol SOP Instance and ignore all query parameters, or
- return 400 (Bad Request) and may include an appropriate Status Report.

If any of the following are true:

- the SOP Instance UID is not found,
- the parameter value is not a Volumetric Protocol SOP class,
- the Target Resource is a Volumetric Presentation State,
- the Target Resource does not meet the Volume Input Requirements in PS3.3 Section C.11.23.1,

then the origin server shall return a 400 (Bad Request) and may include an appropriate Status Report.

8.3.5.3.4 Rendering Method

The "renderingmethod" parameter specifies the display algorithm to be applied to the Volume Data.

The syntax of this parameter is:

```
%s"renderingmethod=" 1#( %s"volume_rendered" / %s"maximum_ip" / %s"minimum_ip" /
%s"average_ip" )
```

Where

`volume_rendered`	A method where each XY pixel of the rendered view is determined by accumulating the set of non-transparent voxel samples along a ray.
`maximum_ip`	A method that projects the interpolated sample with maximum intensity that falls 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 falls in the path of each ray traced from the viewpoint to the plane of projection.
`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.

Notes

1. These values correspond to the differently capitalized values of Rendering Method (0070,120D). See Sections C.11.23 and C.11.30 in PS3.3.
2. There is no parameter to control the type of projection used during rendering. Rendered 3D Volume Resources use Orthographic projection. See Figure C.11.30-1 in PS3.3.

If "renderingmethod" is not present, the origin server may apply a default rendering method, based on the resource, or alternatively, return 400 (Bad Request) and may include an appropriate Status Report.

If the Target Resource is a Volumetric Presentation State, the origin server shall return a 400 (Bad Request) and may include an appropriate Status Report.

8.3.5.3.5 Orientation

The "orientation" parameter specifies the patient's orientation as seen by the camera for the current 3D or MPR Volumetric Presentation View.

The syntax of this parameter is:

```
%s"orientation =" 1#( %s"a" / %s"p" / %s"r" / %s"l" / %s"h" / %s"f" )
```

Where

`a`	Anterior: The camera is viewing the patient from their anterior in the coronal plane, and viewpoint up is oriented to the patient's superior.
`p`	Posterior: The camera is viewing the patient from their posterior in the coronal plane, and viewpoint up is oriented to the patient's superior.
`r`	Right: The camera is viewing the patient from their right in the sagittal plane, and viewpoint up is oriented to the patient's superior.
`l`	Left: The camera is viewing the patient from their left in the sagittal plane, and viewpoint up is oriented to the patient's superior.
`h`	Head: The camera is viewing the patient from above in the axial plane, and viewpoint up is oriented to the patient's anterior.
`f`	Foot: The camera is viewing the patient from below in the axial plane, and viewpoint up is oriented to the patient's anterior..

Note

These values correspond to the differently capitalized values of the Patient Orientation (0020,0020) and Image Orientation (Patient) (0020,0037). See Section C.7.6.1.1.1 in PS3.3 and Section A in PS3.17.

If the Target Resource is a Volumetric Rendering Presentation State and any orientation query parameters are present, the origin server shall apply the query parameter(s) instead of the geometry attributes in the Multi-Planar Reconstruction Geometry Module, or the Volume Render Geometry Module.

Note

This is intended to allow the user to adjust orientation after viewing the initial orientation defined in the Volumetric Presentation State.

If both the "orientation" parameter and any of the camera orientation parameters (i.e., "viewpointposition", "viewpointlookat", or "viewpointup") are present, the origin server shall return a 400 (Bad Request) and may include an appropriate Status Report.

8.3.5.3.6 Viewpoint Position

The "viewpointposition" parameter specifies the position of the camera in the Viewpoint Coordinate System (VCS). See Section C.11.30.1 in PS3.3.

The syntax of this parameter is:

```
%s"viewpointposition =" px "," py "," pz
```

Where

`px, py and pz`	Position of the viewpoint in volume space. A point (x,y,z) in the VCS.

Note

This corresponds to the Viewpoint Position (0070,1603) attribute. See Section C.11.30 in PS3.3.

If the Target Resource is a Volumetric Rendering Presentation State and any orientation query parameters are present, the origin server shall apply the query parameter(s) instead of the geometry attributes in the Multi-Planar Reconstruction Geometry Module, or the Volume Render Geometry Module.

Any or all of the camera orientation parameters may be included. If any of the camera orientation Query Parameters are absent, the origin server may apply a default value (e.g.,

- set “viewpointposition” to the patient’s anterior,
- set “viewpointlookat” to the center of volume,
- set “viewpointup” to the patient’s superior),

or return a 400 (Bad Request) and may include an appropriate Status Report.

8.3.5.3.7 Viewpoint LookAt

The "viewpointlookat" parameter specifies the point that the camera is looking at within the Viewpoint Coordinate System (VCS). See Section C.11.30.1 in PS3.3.

The syntax of this parameter is:

```
%s"viewpointlookat =" lx "," ly "," lz
```

Where

`lx, ly and lz`	Viewpoint LookAt point (i.e., the point that the camera is looking at). A point (x,y,z) in the VCS.

Note

This corresponds to the Viewpoint LookAt Point (0070,1604) attribute. See Section C.11.30 in PS3.3.

If the Target Resource is a Volumetric Rendering Presentation State and any orientation query parameters are present, the origin server shall apply the query parameter(s) instead of the geometry attributes in the Multi-Planar Reconstruction Geometry Module, or the Volume Render Geometry Module.

8.3.5.3.8 Viewpoint Up

The "viewpointup" parameter specifies the vertical orientation of the camera within the Viewpoint Coordinate System (VCS). See Section C.11.30.1 in PS3.3.

The syntax of this parameter is:

```
%s"viewpointup =" ux "," uy "," uz
```

Where

`ux, uy and uz`	Viewpoint up direction (i.e., the direction that the top of the camera is pointing to). A vector (x,y,z) in the VCS.

Note

This corresponds to the Viewpoint Up Direction (0070,1605) attribute. See Section C.11.30 in PS3.3.

If the Target Resource is a Volumetric Rendering Presentation State and any orientation query parameters are present, the origin server shall apply the query parameter(s) instead of the geometry attributes in the Multi-Planar Reconstruction Geometry Module, or the Volume Render Geometry Module.

8.3.5.3.9 Slab Thickness

The "mprslab" parameter specifies the thickness of the MPR plane. This parameter results in an orthographic rendering with a defined thickness using the method defined by "renderingmethod". See PS3.3 Section C.11.26.1.1 for more information.

The syntax of this parameter for a Rendered MPR Volume is:

```
%s"mprslab =" st
```

Where

`st` Thickness of the Multi-Planar Reconstruction slab as a value greater than zero, in mm.

Notes

1. This corresponds to the MPR Slab Thickness (0070,1503) attribute. See Section C.11.26 in PS3.3.
2. The slab thickness of the returned media might not match the requested thickness due to the voxel size of the Target Resource.

If "renderingmethod" is not present, the origin server may apply a default rendering method, based on the resource and/or slab thickness, or alternatively, return 400 (Bad Request) and may include an appropriate Status Report.

If the Target Resource is a Volumetric Presentation State, the origin server shall return a 400 (Bad Request) and may include an appropriate Status Report.

8.3.5.3.10 Swivel Range

The "swivelrange" parameter specifies the angular range over which a rendered volume rotates around the swivel axis, which is defined as the axis parallel to the “viewpointup“ intersecting the “viewpointlookat”. The rendered volume rotates back and forth.

The syntax of this parameter is:

```
%s"swivelrange =" sr
```

Where

`sr` Range in which a volume rotates back-and-forth around the swivel axis, in degrees.

Note

This corresponds to the differently capitalized SWIVEL value of Presentation Animation Style (0070,1A01) and Swivel Range (0070,1A06). See Section C.11.29 in PS3.3 and Section FF.2.4.2 in PS3.4.

The origin server shall create an animation with a number of frames equal to Swivel Range divided by the "animationstepsize".

If the “swivelrange” parameter is present and the “animationrate” parameter is not present, the origin server shall determine the animation rate.

If the Target Resource is a Volumetric Presentation State, the origin server shall return a 400 (Bad Request) and may include an appropriate Status Report.

8.3.5.3.11 Volumetric Curve Point Coordinates

The "volumetriccurvepoint" parameter specifies coordinates of points on the animation 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. At least two points are required for an animation. See Section C.11.29.1 in PS3.3.

The syntax of this parameter is:

```
%s" volumetriccurvepoint =" px "," py "," pz
```

Where

`px, py and pz` Position of a point on the animation curve. A point (x,y,z) in the VPS-RCS, in mm.

Note

This corresponds to the Number of Volumetric Curve Points (0070,150D) attribute. See Section C.11.29 in PS3.3.

The origin server shall create an animation with a number of frames equal to the total distance of the Volumetric Curve divided by the "animationstepsize".

If the "volumetriccurvepoint" parameters are present and the “animationrate” parameter is not present, the origin server shall determine the animation rate.

If the Target Resource is a Volumetric Presentation State, the origin server shall return a 400 (Bad Request) and may include an appropriate Status Report.

8.3.5.3.12 Animation Step Size

The "animationstepsize" parameter specifies distance between animation steps, or frames, in a Volumetric Rendering animation.

For a swivel animation, the distance between steps is in degrees. For a Volumetric Curve, the distance between steps is in mm along the animation curve.

The syntax of this parameter is:

```
%s" animationstepsize =" ss
```

Where

`ss` The animation step size, an integer greater than zero.

Note

This corresponds to the Number of Animation Step Size (0070,1A05) attribute. See Section C.11.29 in PS3.3.

The origin server shall create an animation, with a number of frames equal to either:

- the “swivelrange” divided by the "animationstepsize", or

- the total distance of the Volumetric Curve divided by the "animationstepsize".

If " animationstepsize " is not present, and either “swivelrange”, or “volumetriccurvepoint” are present, the origin server may apply a default animation step size, or alternatively, return 400 (Bad Request) and may include an appropriate Status Report.

If the Target Resource is a Volumetric Presentation State, the origin server shall return a 400 (Bad Request) and may include an appropriate Status Report.

8.3.5.3.13 Animation Rate

The “animationrate” parameter specifies the rate at which an animated 3D or MPR Volumetric Presentation is displayed.

The syntax of this parameter is:

```
%s" animationrate =" rt
```

Where

`rt`	Rate in steps per second, an integer greater than zero.

Notes

1. This corresponds to Recommended Animation Rate (0070,1A03) in Section C.11.29 in PS3.3 and Section FF.2.4.2 in PS3.4.
2. Playback of the returned media on a client may or may not achieve the requested animation rate.

If " animationrate" is not present, and other animation parameters are present (e.g., “swivelrange”, “animationstepsize”, or “volumetriccurvepoint”), the origin server may apply a default animation rate, or alternatively, return 400 (Bad Request) and may include an appropriate Status Report.

If the Target Resource is a Volumetric Presentation State, the origin server shall return a 400 (Bad Request) and may include an appropriate Status Report.

8.3.5.3.14 Volumetric Metadata

The "volumetricmetadata" parameter specifies that, in addition to the requested 2D representation of the rendered volume, the response payload includes a Rendered Volume Response Module of the parameters applied by the origin server to generate the volumetric rendering.

The syntax of this parameter is:

```
%s"volumetricmetadata =" "yes"
```

Where

`yes`	Indicates that a Rendered Volume Response Module shall be present in the response payload.

The defined value is "yes". If this parameter is not present, no Response Module is requested.

The origin server shall return response payload containing a Rendered Volume Response Module, specified in Annex X, in addition to the requested media type.

If the Target Resource is a Volumetric Presentation State, the origin server shall return a 400 (Bad Request) and may include an appropriate Status Report.

Update PS3.18 Section 8.11 as follows:

8.11 Security and Privacy

It is very likely that DICOM objects contain Protected Health Information. Privacy regulations in the United States (HIPAA), Europe (GDPR), and elsewhere, require that Individually Identifiable Information be kept private. It is the responsibility of those implementing and deploying the DICOM Standard to ensure that applicable regulations for security and privacy are satisfied.

See, for example, [ONC Privacy Security Guide].

The DICOM PS3.10 File Format has security considerations that will apply whenever DICOM PS3.10 File format is used. See Section 7.5 in PS3.10.

Rendered Volume Resources (see Sections 10.4.1.1.7 and 10.4.1.1.8) may include recognizable visual features.

Update PS3.18 Section 12.1.1 Resource Descriptions as follows:

12.1.1 Resource Descriptions

An NPI Service manages resources from the same NPI Category. Target URIs have the following templates:

```
/{npi-name}
/{npi-name}/{uid}
```

Where

```
npi-name    = "color-palettes"
            / "defined-procedure-protocols"
            / "hanging-protocols"
            / "implant-templates"
            / "volumetric-rendering-protocols"
uid         ; is the Unique Identifier of an NPI Instance
```

Table 12.1.1-1 contains the templates for the NPI Resource Categories.

Table 12.1.1-1. Resource Categories, URI Templates and Descriptions

Resource Category	URI Template and Description	Corresponding IOD	Storage Class	Information Model
…				
Inventory	/inventories{/uid}	Section A.88 “Inventory IOD” in PS3.3	Section GG “Non-Patient Object Storage Service Class” in PS3.4	Section JJ.2 “Inventory Q/R Information Model” in PS3.4

Volumetric Rendering Protocol	**/volumetric-rendering-protocol{/uid}**	**Section A.XX "Volumetric Rendering Protocol IOD" in PS3.3**	**Section GG "Non-Patient Object Storage Service Class" in PS3.4**	**Section HH "Defined Procedure Protocol Query/Retrieve Service Classes" in PS3.4**

The NPI SOP Classes are listed in Table GG.3-1 "Standard SOP Classes" in PS3.4.

Update PS3.18 Section 12.6.1.2 Query Parameters as follows:

12.6.1.2 Query Parameters

The user agent shall supply, and the origin server shall support, the Common Query Parameters in Section 12.1.2.

The origin server shall support Query Parameters as required in Table 8.3.4-1.

The user agent shall supply in the request Query Parameters as required in Table 8.3.4-1.

For each Resource Category the origin server supports, it shall support the behaviors and matching key Attributes specified in the corresponding sections in Table 12.6.1-2.

Table 12.6.1-2. NPI Resource Search Attributes

Resource Category	Behaviors and Matching Key Attributes
Color Palette	Section X.6.1.2 "Color Palette Attributes" in PS3.4.
Defined Procedure Protocol	Section HH.6.1.2 "Defined Procedure Protocol Attributes" in PS3.4.
Hanging Protocol	Section U.6.1.2 "Hanging Protocol Attributes" in PS3.4.
Implant Template	Section BB.6.1.2 "Implant Template Attributes" in PS3.4.
Inventory	Section JJ.2.2 "Inventory Q/R Information Model Attributes" in PS3.4.
Volumetric Rendering Protocol	**Section HH.6.1.2 "Defined Procedure Protocol Attributes" in PS3.4.**

Add the following Sections after Section B.25:

B.x1 Render a Series into a 3D Volume as a JPEG

This example illustrates a request for a 3D volume rendering of a series of legacy instances, returned as a jpeg. Since no orientation is specified, the determination of orientation is left to the origin server.

```
GET /radiology
/studies/1.2.250.1.59.40211.12345678.678910
/series/1.2.250.1.59.40211.789001276.14556172.67789
/rendered3D?renderingmethod=volume_rendered

HTTP/1.1
Host: www.hospital-stmarco
Accept: image/jpeg

HTTP/1.1 200 OK
Content-Length: 79323
Content-Type: image/jpeg
```

<BINARY JPEG DATA>

B.x2 Render a Multi-frame Instance as a 3D Volume Rendering

This example illustrates a request for a 3D volume rendering of a multi-frame instance. The initial anterior view is swiveled 180 degrees left to right. A 20fps mp4 video and the Rendered Volume Response Module are returned. Since animation step size is not specified, the server determines one.

Note

The anterior orientation parameter is included the request, however the Rendered Volume Response Module requires camera orientation parameters are returned.

```
GET /radiology
/studies/1.2.250.1.59.40211.12345678.678910
/series/1.2.250.1.59.40211.789001276.14556172.67789
/instances/1.2.250.1.59.40211.2678810.87991027.899772.2
/rendered3D?renderingmethod=volume_rendered
&orientation=a
&swivelrange=180
&animationrate=20
&volumetricmetadata=yes

HTTP/1.1
Host: www.hospital-stmarco
Accept: multipart/related; type=video/mp4, multipart/related;
type=application/dicom+json

HTTP/1.1 200 OK
Content-Length: 15000
Content-Type: multipart/related; type=application/dicom+json
{
    "00720510": {
        "vr": "CS",
        "Value": ["3D_RENDERING"]
    },
    "0070120D": {
        "vr": "CS",
        "Value": ["VOLUME_RENDERED"]
    },
    "00801603": {
        "vr": "FD",
        "Value": [100,101,200]
    },
    "00801604": {
        "vr": "FD",
        "Value": [100,100,200]
    },
    "00801605": {
        "vr": "FD",
        "Value": [0,0,1]
    },
…
--MESSAGEBOUNDARY--
Content-Length: 3145728
Content-Type: video/mp4

<BINARY MPEG-4 DATA>
```

B.x3 Render Multiple Phase Series as an MPR

This example illustrates a request for an animated MPR rendering of a multi-phase series. Range matching of the Cardiac R-R Interval Specified (0018,9070) attribute specifies the desired phases to render as a temporal volume. Oblique orientation is specified using camera orientation parameters. The MPR has a nominal thickness, and windowed at a width of 400 and center of 40. A 30fps mp4 video is returned.

```
GET /radiology
/studies/1.2.250.1.59.40211.12345678.678910/renderedmpr?
CardiacRRIntervalSpecified=140-260
&renderingmethod=average_ip
&viewpointposition=532,38,126
&viewpointlookat=-532,-76,-154
&viewpointup=0,0,0
&animationrate=30
&window=400,40,linear

HTTP/1.1
Host: www.hospital-stmarco
Accept: video/mp4

HTTP/1.1 200 OK
Content-Length: 3145728
Content-Type: video/mp4

<BINARY MPEG-4 DATA>
```

Note

See PS3.4 Section C2.2.2 for Attribute Matching.

B.x4 Render One Phase of a Multi-phase Series as an MIP

This example illustrates a request for a static MPR rendering of one phase of a multi-phase series. A volume input reference is provided to identify the desired phase. Coronal orientation is specified using camera orientation parameters. The MPR MIP is 20mm thick, and windowed at a width of 700 and center of 100. A scaled jpeg is returned.

```
GET /radiology
/studies/1.2.250.1.59.40211.12345678.678910
/series/1.2.250.1.59.40211.789001276.14556172.67789/renderedmpr?
volumeinputreference=1.2.250.1.59.40211.2678810.87991027.899772.2
&renderingmethod=maximum_ip
&mprslab=20
&viewpointposition=100,101,200
&viewpointlookat=100,100,200
&viewpointup=0,0,1
&viewport=512,512,128,128,256,256
&window=700,100,linear

HTTP/1.1
Host: www.hospital-stmarco
Accept: image/jpeg

HTTP/1.1 200 OK
Content-Length: 79323
Content-Type: image/jpeg

<BINARY JPEG DATA>
```

626 *Update PS3.18 Table H-1. Resources and Methods as follows:*

627 **Table H-1. Resources and Methods**

Service	Resource	Transactions	Reference
Studies (see Section 10.1.1)			
	studies	Search for Studies Store Instances	Section 10.6 Section 10.5
	{StudyInstance}	Retrieve Study Store Study Instances	Section 10.4 Section 10.5
	metadata	Retrieve Study Metadata	Section 10.4
	renderedmpr	**Retrieve Rendered MPR Volume Study**	**Section 10.4**
	rendered3d	**Retrieve Rendered 3D Volume Study**	**Section 10.4**
	series	Search for Study Series	Section 10.6
	{SeriesInstance}	Retrieve Series	Section 10.4
	metadata	Retrieve Series Metadata	Section 10.4
	renderedmpr	**Retrieve Rendered MPR Volume Series**	**Section 10.4**
	rendered3d	**Retrieve Rendered 3D Volume Series**	**Section 10.4**
	instances	Search for Study Series Instances	Section 10.4
	{SOPInstance}	Retrieve Instance	Section 10.4
	metadata	Retrieve Instance Metadata	Section 10.4
	renderedmpr	**Retrieve Rendered MPR Volume Instance**	**Section 10.4**
	rendered3d	**Retrieve Rendered 3D Volume Instance**	**Section 10.4**
	frames	N/A	N/A
	{framelist}	Retrieve Frames	Section 10.4
	renderedmpr	**Retrieve Rendered MPR Volume Frames**	**Section 10.4**
	rendered3d	**Retrieve Rendered 3D Volume Frames**	**Section 10.4**
	instances	Search for Study Instances	Section 10.6
	series	Search for Series	Section 10.6
	{SeriesInstance}	N/A	N/A
	{instances}	Search for Instances	Section 10.6
	instances	Search for Instances	Section 10.6
	{BulkDataReference}	Retrieve Bulkdata	Section 10.4
Worklist (see Section 11.1.1)			
	workitems	Search for Workitem Create Workitem	Section 11.9 Section 11.4
	{Workitem}	Retrieve Workitem Update Workitem	Section 11.4 Section 11.6
	state	Change Workitem State	Section 11.7
	cancelrequest	Request Workitem Cancellation	Section 11.8
	subscribers	N/A	N/A
	{AETitle}	Subscribe Unsubscribe	Section 11.10 Section 11.11
	1.2.840.10008.5.1.4.34.5	N/A	N/A
	subscribers	N/A	N/A
	{AETitle}	Subscribe Unsubscribe	Section 11.10 Section 11.11

Service	Resource	Transactions	Reference
	suspend	Unsubscribe	Section 11.11
	1.2.840.10008.5.1.4.34.5.1	N/A	N/A
	subscribers	N/A	N/A
	{AETitle}	Subscribe Unsubscribe	Section 11.10 Section 11.11
	suspend	Suspend Worklist Subscription	Section 11.11
Non-Patient Instance (see Section 12.1.1)			
	color-palettes	N/A	N/A
	{uid}	Retrieve Store Search	Section 12.4 Section 12.5 Section 12.6
	defined-procedure-protocol	N/A	N/A
	{uid}	Retrieve Store Search	Section 12.4 Section 12.5 Section 12.6
	hanging-protocol	N/A	N/A
	{uid}	Retrieve Store Search	Section 12.4 Section 12.5 Section 12.6
	implant-templates	N/A	N/A
	{uid}	Retrieve Store Search	Section 12.4 Section 12.5 Section 12.6
	inventories	N/A	N/A
	{uid}	Retrieve Store Search	Section 12.4 Section 12.5 Section 12.6
	volumetric-rendering-protocol	**N/A**	**N/A**
	{uid}	**Retrieve** **Store** **Search**	**Section 12.4** **Section 12.5** **Section 12.6**

Add the following Section after Annex I:

X Rendered Volume Response Module

The Rendered Volume Response Module provides the user agent a representation of the parameters applied by the origin server to generate a volumetric rendering.

X.1 Response Message Body

Table X.1-1 defines the Attributes for referencing the rendering parameters that are contained in a Rendered Volume Resources Response Module in the response message body.

Note

These represent Query parameters that may be specified by the user agent in Rendered Volume Resources. See Section 8.3.5.3.

Table X.1-1. Rendered Volume Response Module Attributes

Attribute Name	Tag	Type	Attribute Description
Reformatting Operation Type	(0072,0510)	1	Reformatting operation to be applied to the Image Set. See Section C.XX-1 in PS3.3.
Rendering Method	(0070,120D)	1	Specifies the display algorithm to be applied to the Volume Data. See Section C.XX-1 in PS3.3.
Viewpoint Position	(0070,1603)	1	Position of the viewpoint in volume space. See Section C.11.30 in PS3.3.
Viewpoint LookAt Point	(0070,1604)	1	Point the viewpoint is looking at. See Section C.11.30 in PS3.3.
Viewpoint Up Direction	(0070,1605)	1	Vertical orientation of the view. See Section C.11.30 in PS3.3.
MPR Slab Thickness	(0070,1503)	1C	Required if Reformatting Operation Type (0072,0510) has a value of MPR and there is a specified thickness. See Section C.11.26 in PS3.3
VOI LUT Function	(0028,1056)	1C	Required if Rendering Method (0070,120D) is not VOLUME_RENDERED. See Section C.11.2.1.2 in PS3.3.
Window Width	(0028,1051)	1C	Required if Rendering Method (0070,120D) is not VOLUME_RENDERED. See Section C.11.2.1.2 in PS3.3.
Window Center	(0028,1051)	1C	Required if Rendering Method (0070,120D) is not VOLUME_RENDERED. See Section C.11.2.1.2 in PS3.3.
Swivel Range	(0070,1A06)	1C	Required for SWIVEL animations. See Section C.11.29.1 in PS3.3.
Animation Step Size	(0070,1A05)	1C	Required for SWIVEL or CROSSCURVE animations. See Section C.11.29.1 in PS3.3.
Recommended Animation Rate	(0070,1A03)	1C	Required for video media types. See Section C.11.29.1.

Modifications to PS3.2

Modify PS3.2 Section N.1.3.2 Studies Service as follows:

N.1.3.2 Studies Service

Table N.1-9 lists details on the support of the Studies Service.

[Complete Table N.1-9 to indicate support for the Studies Web Service]

Table N.1-9. Study Service

Service	Transaction	Resource	User Agent	Origin Server
Studies Web Service	Retrieve Capabilities			
	Retrieve (WADO-RS)	Study		
		Study Metadata		

		Study Bulkdata		
		Study Pixel Data		
		Rendered Study		
		Rendered MPR Volume Study		
		Rendered 3D Volume Study		
		Study Thumbnail		
		Series		
		Series Metadata		
		Series Bulkdata		
		Series Pixel Data		
		Rendered Series		
		Rendered MPR Volume Series		
		Rendered 3D Volume Series		
		Series Thumbnail		
		Instance		
		Instance Metadata		
		Instance Bulkdata		
		Instance Pixel Data		
		Rendered Instance		
		Rendered MPR Volume Instance		
		Rendered 3D Volume Instance		
		Instance Thumbnail		
		Frames		
		Rendered Frames		
		Rendered MPR Volume Frames		
		Rendered 3D Volume Frames		
		Frame Thumbnail		
		Bulkdata		
	…			

[If your product supports any Rendered Volume Resources, indicate supported SOP Classes in the "Process" column of Table N.1-1]

Add Volumetric Rendering Resources to PS3.2 Table N.5-72 as follows:

Table N.5-72. Resources Retrieve Transaction - User Agent

Resource	Comments
DICOM Instance Resources - See Resources path in Table 10.4.1-1 in PS3.18	
Study Instances	
Series Instances	
Individual Instance	
DICOM Metadata Resources - See Resources path in Table 10.4.1-2 in PS3.18	
Study Metadata	
Series Metadata	
Instance Metadata	
DICOM Bulkdata Resources - See Resources path in Table 10.4.1.5-1 in PS3.18	

Study Bulkdata	
Series Bulkdata	
Instance Bulkdata	
Bulkdata	
DICOM Pixel Data Resources - See Resources path in Table 10.4.1.6-1 in PS3.18	
Study Pixel Data	
Series Pixel Data	
Instance Pixel Data	
Frame Pixel data	
Rendered Resources - See Resources path in Table 10.4.1-3 in PS3.18	
rendered study	
rendered series	
rendered instance	
rendered frame	
rendered bulk	
Rendered MPR Volume Resources - See Resources path in Table 10.4.1.7-1 in PS3.18	
rendered mpr volume study	
rendered mpr volume series	
rendered mpr volume instance	
rendered mpr volume frames	
Rendered 3D Volume Resources - See Resources path in Table 10.4.1.8-1 in PS3.18	
rendered 3d volume study	
rendered 3d volume series	
rendered 3d volume instance	
rendered 3d volume frames	
Thumbnail Resources - See Resources path in Table 10.4.1-4 in PS3.18	
Study Thumbnail	
Series Thumbnail	
Instance Thumbnail	
Frame Thumbnail	

Add Volumetric Rendering Query Parameters to PS3.2 Table N.5-73 as follows:

Table N.5-73. Query Parameters for Retrieve Transaction - User Agent

Query Parameter	Supported Values	Comments
Accept	*[See examples in header parameters.]*	
Rendered Resource		
annotation	<<patient technique>>	
charset	<<UTF-8 ISO -8859-1 …>>	

quality		
viewport		
window		
iccprofile	<<no yes srgb adobergb rommrgb>>	
Rendered Volume Resources		
volumeinputreference		
match	**Attribute Values to address the search (matching key). See the supported DICOM Attribute in the Table N.5-84**	
volumetricprotocol		
renderingmethod	**<<volume_rendered** **maximum_ip** **minimum_ip** **average_ip>>**	
orientation		
viewpointposition		
viewpointlookat		
viewpointup		
mprslab		
swivelrange		
volumetriccurvepoint		
animationstepsize		
animationrate		
volumetricmetadata		
Thumbnail Resource		
charset	<<UTF-8 ISO-8859-1 …>>	
viewport		

Add Volumetric Rendering Header Fields to PS3.2 Table N.5-74 as follows:

655 **Table N.5-74. Header Fields for Retrieve Transaction - User Agent**

Header Field	Supported Values	Comments
Instance resource		
Accept	multipart/related; type="application/dicom"; transfer-syntax={uid}	See in the Overview section Table N.1-1 the supported DICOM SOP Classes / Transfer Syntaxes. Look for "Y" in the "UA" column.
	multipart/related; type="application/octet-stream"	
Metadata resource		
Accept	<<multipart/related; type="application/dicom+xml" multipart/related; type="application/dicom+json">>	
Bulkdata and Pixel Data resource		
Accept	Uncompressed: <<multipart/related; type="application/octet-stream">> Compressed: <<multipart/related; type="{media-type}">> supported {media-type} being <<image/jpeg image/x-dicom-rle image/x-jls image/jp2 image/jpx video/mpeg2 video/mp4>>	See details in Section N.5.3.2.1.2.
Rendered Resource		

Accept	<<image/jpeg image/gif image/png image/jp2 image/gif video/mpeg video/mp4 video/H265 text/html text/plain text/xml>>	See details in Section N.5.3.2.1.3.
Rendered Volume Resource		
Accept	**<<image/jpeg** **image/gif** **image/png** **image/jp2** **image/gif** **video/mpeg** **video/mp4** **video/H265** **multipart/related; type="application/dicom+xml"** **multipart/related; type="application/dicom+json">>**	**See details in Section N.5.3.2.1.3.**
Thumbnail Resource		

Accept	<<image/jpeg image/gif image/png image/jp2 image/gif video/mpeg video/mp4 video/H265 text/html text/plain text/xml>>	See details in Section N.5.3.2.1.3.
All Resources		
Accept-charset	<<UTF-8 ISO-8859-1 …>>	

656

657 *Add Volumetric Rendering Header Fields to PS3.2 Table N.5-74 as follows:*

658 **Table N.5-75. Resources Retrieve Transaction - Origin Server**

Resource	Comments
DICOM Instance Resources - See Resources path in Table 10.4.1-1 in PS3.18	
Study Instances	
Series Instances	
Individual Instance	
DICOM Metadata Resources - See Resources path in Table 10.4.1-2 in PS3.18	
Study Metadata	
Series Metadata	
Instance Metadata	
DICOM Bulkdata Resources - See Resources path in Table 10.4.1.5-1 in PS3.18	
Study Bulkdata	
Series Bulkdata	
Instance Bulkdata	
Bulkdata	
DICOM Pixel Data Resources - See Resources path in Table 10.4.1.6-1 in PS3.18	
Study Pixel Data	

Series Pixel Data	
Instance Pixel Data	
Frame Pixel data	
Rendered Resources - See Resources path in Table 10.4.1-3 in PS3.18	
rendered study	
rendered series	
rendered instance	
rendered frame	
rendered bulk	
Rendered MPR Volume Resources - See Resources path in Table 10.4.1.7-1 in PS3.18	
rendered mpr volume study	
rendered mpr volume series	
rendered mpr volume instance	
rendered mpr volume frames	
Rendered 3D Volume Resources - See Resources path in Table 10.4.1.8-1 in PS3.18	
rendered 3d volume study	
rendered 3d volume series	
rendered 3d volume instance	
rendered 3d volume frames	
Thumbnail Resources - See Resources path in Table 10.4.1-4 in PS3.18	
Study Thumbnail	
Series Thumbnail	
Instance Thumbnail	
Frame Thumbnail	

Add Volumetric Rendering Query Parameters to PS3.2 Table N.5-76 as follows:

Table N.5-76. Query Parameters for Retrieve Transaction - Origin Server

Query Parameter	Supported Values	Comments
Accept	[Supported Values are the same as for the Accept Header Field.]	
Rendered resource		
annotation	<<patient technique>> [Add additionally supported key word Values here.]	
charset	<<UTF-8 ISO-8859-1 …>>	
Quality		
Viewport		
Window		

iccprofile	<<no yes srgb adobergb rommrgb>>	
Rendered Volume resource		
volumeinputreference		
match	**Attribute Values to address the search (matching key). See the supported DICOM Attribute in the Table N.5-84**	
volumetricprotocol		***[Describe whether your product allows Query Parameters to supersede parameters within the Volumetric Rendering Protocol. See Section 8.3.5.3.3 in PS3.18]***
renderingmethod	**<<volume_rendered** **maximum_ip** **minimum_ip** **average_ip>>**	
orientation		
viewpointposition		
viewpointlookat		
viewpointup		
mprslab		
swivelrange		
volumetriccurvepoint		
animationstepsize		
animationrate		
volumetricmetadata		
Thumbnail resource		
charset	<<UTF-8 ISO-8859-1 …>>	
Viewport		

662

Add Volumetric Rendering Protocol to PS3.2 Table N.5-140 as follows:

Table N.5-140. Non-Patient Instance Web Service Storage SOP Classes

SOP Class Nname	SOP Class UID	User Agent	Origin Server	Comments
Hanging Protocol Storage	1.2.840.10008.5.1.4.38.1			
Color Palette Storage	1.2.840.10008.5.1.4.39.1			
Generic Implant Template Storage	1.2.840.10008.5.1.4.43.1			
Implant Assembly Template Storage	1.2.840.10008.5.1.4.44.1			
Implant Template Group Storage	1.2.840.10008.5.1.4.45.1			
CT Defined Procedure Protocol Storage	1.2.840.10008.5.1.4.1.1.200.1			
Protocol Approval Storage	1.2.840.10008.5.1.4.1.1.200.3			
Volumetric Rendering Protocol Storage	**1.2.840.10008.5.1.4.xxuid.1**			

Modify PS3.2 N.5.3.4.2.1 User Agent as follows:

N.5.3.4.2.1 User Agent

The Non-Patient Instance (NPI) Retrieve transaction as user agent can request resources listed in Table N.5-141

[Provide implementation specific details in the "Comments" column and indicate the supported {npi-name}. They can be:

- *color-palettes*
- *defined-procedure-protocols*
- *hanging-protocols*
- *implant-templates*
- ***volumetric-rendering-protocols****]*

Modify PS3.2 N.5.3.4.2.2 Origin Server as follows:

N.5.3.4.2.2 Origin Server

The NPI Web Service origin server supports resources listed in Table N.5-144 for the Retrieve Transaction:

[Provide implementation specific details in the "Comments" column and indicate the supported {npi-name}. They can be:

- *color-palettes*
- *defined-procedure-protocols*
- *hanging-protocols*
- *implant-templates*
- ***volumetric-rendering-protocols****]*

Modify PS3.2 N.5.3.4.3.1 User Agent as follows:

N.5.3.4.3.1 User Agent

For details regarding the IODs created by the system, see Section N.9.

The NPI Store Transaction user agent can request resources listed in Table N.5-147.

List the supported resources. Remove the non-supported resources rows.

[Provide implementation specific details in the "Comments" column and indicate what the supported {npi-name}. They can be:

- *color-palettes*
- *defined-procedure-protocols*
- *hanging-protocols*
- *implant-templates*
- ***volumetric-rendering-protocols****]*

Modify PS3.2 N.5.3.4.3.2 Origin Server as follows:

N.5.3.4.3.2 Origin Server

The NPI Store Transaction origin server receives POST requests to store or append to an existing resource on the server.

The user agent specifies the Target Resource as part of the URI and encapsulates the data in a multipart request body with a proper Content-Type (i.e., BINARY, XML or JSON).

The URI is composed by a Base URI: see Base URI for the origin server in Section N.6.3.4.

The NPI Store Transaction origin server supports resources listed in Table N.5-150.

[Provide implementation specific details in the "Comments" column and indicate what the supported {npi-name}. They can be:

- *color-palettes*
- *defined-procedure-protocols*
- *hanging-protocols*
- *implant-templates*
- ***volumetric-rendering-protocols****]*

Modify PS3.2 N.5.3.4.4.1 User Agent as follows:

N.5.3.4.4.1 User Agent

The NPI Search Transaction user agent can request resources listed in Table N.5-153.

[Provide implementation specific details in the "Comments" column and indicate what the supported {npi-name}. They can be:

- *color-palettes*
- *defined-procedure-protocols*
- *hanging-protocols*
- *implant-templates*
- ***volumetric-rendering-protocols****]*

Modify PS3.2 N.5.3.4.4.2 Origin Server as follows:

N.5.3.4.4.2 Origin Server

The NPI Search Transaction origin server receives GET requests to search for studies, series or instances.

[Specify here if this is a native or a DIMSE proxy implementation.]

The user agent specifies the Target Resource as part of the URI and the acceptable response Content-Type in the HTTP Header (i.e., dicom+xml or dicom+json).

The URI is composed by a Base URI: see Base URI for the origin server in Section N.6.3.4.

The Search Transaction origin server supports resources listed in Table N.5-157. *[Provide implementation specific details in the "Comments" column and indicate what the supported {npi-name}. They can be:*

- *color-palettes*
- *defined-procedure-protocols*
- *hanging-protocols*
- *implant-templates*
- ***volumetric-rendering-protocols****]*

Modifications to PS3.3

Modify the following definitions in PS3.3.17 as follows:

Multi-Planar Reconstruction (MPR)

Also called Multi-Planar Reformatting. A data visualization created by sampling volume data, typically represented by a stack of image planes, that lies in the neighborhood of the intersection of the volume with a plane, curved plane, slab or curved slab.

Planar Multi-Planar Reconstruction (Planar MPR)

An MPR where the samples are centered on a single plane intersected with the volume.

Volume Data

Data represented by a set of parallel XY planes whose positions are relative to each other that are arranged in a cartesian voxel grid.

Volumetric Presentation State (VPS)

A Presentation State that defines a transformation from 3D spatial input data (volume) to 2D spatial output data, with or without affecting other dimensions such as temporal.

Volumetric Presentation State Reference Coordinate System (VPS-RCS)

754 The Reference Coordinate System to which inputs to a Volumetric Presentation State are
755 registered and to which Attribute Values of a Volumetric Presentation State are referenced
756 (unless stated otherwise).

757 Volumetric Presentation View

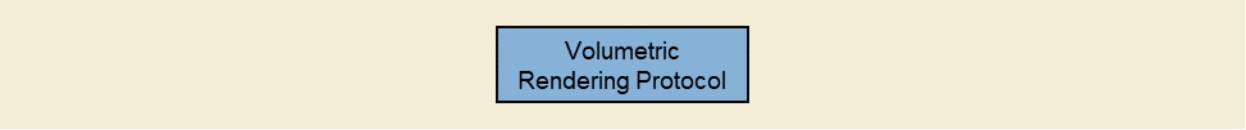
758 A presentation, with two spatial dimensions, of Volume Data.

759 *Add Volumetric Rendering Protocol to PS3.3 Section 7.13 as follows:*

760 **7.13.x Volumetric Rendering Protocol Information Entity**

761 A Volumetric Rendering Protocol Information Entity specifies standardized parameters for the multi-planar
762 reconstruction or volumetric rendering of a specific acquisition type. A Volumetric Rendering Protocol
763 definition includes descriptors that identify the Volumetric Rendering Protocol, input image set selector
764 constraints, an Icon providing a representation of the rendered volume resulting from the protocol, as well
765 as the 3D reformatting operation and rendering type.

766 The Volumetric Rendering Protocol IE does not have any relationships with other Information Entities.
767 See Figure 7.13-x

768 

769 **Figure 7.13-x. DICOM Model of the Real World - Volumetric Rendering Protocol**

770 *Add Volumetric Rendering Protocol IOD to PS3.3 Table A.1-9 as follows:*

771 **Table A.1-9. Composite Information Object Modules Overview - Protocols**

IODs **Modules**	**CT Performed Procedure Protocol**	**XA Performed Procedure Protocol**	**CT Defined Procedure Protocol**	**XA Defined Procedure Protocol**	**Protocol Approval**	**Volumetric Rendering Protocol**
Patient	M	M				
Clinical Trial Subject	U	U				
General Study	M	M				
Patient Study	U	U				
Clinical Trial Study	U	U				
General Series	M	M				
Clinical Trial Series	U	U				
Enhanced Series	M	M				
CT Protocol Series	M					
XA Protocol Series		M				
Frame of Reference	M	M				
General Equipment	M	M	M	M	M	**M**
Enhanced General Equipment	M	M	M	M	M	**M**
Protocol Context	M	M	M	M		**M**
Patient Protocol Context	U	U				

Clinical Trial Context			U	U		
Patient Specification			U	U		
Equipment Specification			M	M		
Instructions	U	U	U	U		**U**
Patient Positioning	U	U	U	U		
General Defined Acquisition			U	U		
Performed CT Acquisition	U					
Performed XA Acquisition		U				
General Defined Reconstruction			U	U		
Performed CT Reconstruction	U					
Performed XA Reconstruction		U				
Defined Storage			U	U		
Performed Storage	U	U				
Protocol Approval					M	
Volumetric Rendering Protocol						**M**
Volume Data Input Image Set Module						**U**
Volume Definition Module						**U**
Volume Render Geometry						**C**
Render Shading						**U**
Render Display						**C**
Multi-Planar Reconstruction Geometry						**C**
MPR Volumetric Presentation State Display						**C**
VOI LUT						**U**
Presentation Animation						**U**
SOP Common	M	M	M	M	M	**M**

Add new section for Volumetric Rendering Protocol IOD to PS3.3 Annex A Composite Information Object Definitions

A.XX Volumetric Rendering Protocol IOD

A.XX.1 Volumetric Rendering Protocol IOD Description

A Volumetric Rendering Protocol IOD is a non-patient instance belonging to the family of Defined Procedure Protocol IODs that specifies input image set selector constraints and a Rendered presentation of Volume Data. The Volumetric Transformations follow the Planar MPR Volumetric Transformations

specified in PS3.4 Section FF.2.1.1, or the Volume Rendering Volumetric Transformations specified in PS3.4 Section FF.2.1.2.

It includes capabilities for specifying:

a. protocol context
b. volumetric source information
c. grouping of volumetric source information into Volume Data
d. multi-planar or volume rendered geometry
e. optional volume rendered shading models
f. an exemplary thumbnail

A.XX.2 Volumetric Rendering Protocol IOD Entity-Relationship Model

Volumetric Rendering Protocol IOD uses the E-R Model specified in Section 7.13.X.

A.XX.3 Volumetric Rendering Protocol IOD Module Table

Table A.XX.3-1 lists the Modules that make up the Volumetric Rendering Protocol IOD.

Table A.XX.3-1. Volumetric Rendering Protocol IOD Modules

IE	Module	Reference	Usage
Procedure Protocol	Protocol Context	C.34.2	M
	General Equipment Module	C.7.5.1	M
	Enhanced General Equipment Module	C.7.5.2	M
	Instructions Module	C.34.7	U
Volumetric Rendering Protocol	Volumetric Rendering Protocol Module	C.XX.1	M
	Volume Data Input Image Set Module	C.XX.2	U
	Volume Definition Module	C.XX.3	U - Optional if the value of Volume Type (00gg,eee1) is TEMPORAL_VOLUME or MULTIVOLUME May be present otherwise.
Presentation State	Volume Render Geometry	C.11.30	C - Required if Reformatting Operation Type (0072,0510) in the Volumetric Rendering Defined Protocol Module equals 3D RENDERING
	Render Shading	C.11.31	U - Optional if Reformatting Operation Type (0072,0510) in the Volumetric Rendering Defined Protocol Module equals 3D RENDERING
	Render Display	C.11.32	C - Required if Reformatting Operation Type (0072,0510) in the Volumetric Rendering Defined Protocol Module equals 3D RENDERING
	Multi-Planar Reconstruction Geometry	C.11.26	C - Required if Reformatting Operation Type (0072,0510) in the Volumetric Rendering Defined Protocol Module equals MPR

	MPR Volumetric Presentation State Display	C.11.27	C - Required if Reformatting Operation Type (0072,0510) in the Volumetric Rendering Defined Protocol Module equals MPR
	VOI LUT	C.11.2	U - Optional if Rendering Method (0070,120D) in the Volumetric Rendering Defined Protocol Module is not VOLUME_RENDERED
	Presentation Animation	C.11.29	U
	SOP Common	C.12.1	M

794

795 *Add new section for Volumetric Rendering Protocol Modules to PS3.3 Annex C Information Module*
796 *Definitions*

797 **C.XX Volumetric Rendering Protocol Modules**

798 **C.XX.1 Volumetric Rendering Protocol Module**

799 Table C.XX-1.specifies the attributes of the Volumetric Rendering Defined Protocol Module

800 **Table C.XX-1 Volumetric Rendering Protocol Module**

Name	Tag	Type	Description
Reformatting Operation Type	(0072,0510)	1	Reformatting operation to be applied to the Image Set. Enumerated Values: MPR 3D_RENDERING

Rendering Method	(0070,120D)	1	Specifies the display algorithm to be applied to the Volume Data. Only one value shall be present. Enumerated Values: VOLUME_RENDERED A method where each XY pixel of the rendered view is determined by accumulating the set of non-transparent voxel samples along a ray. 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.
Icon Image Sequence	(0088,0200)	1	A image representing the type of output that would be generated by this Volumetric Rendering Protocol. Only a single Item is permitted in this Sequence.
>Include Table C.7-11b "Image Pixel Macro Attributes"			*See Section C.7.6.1.1.6*
Volume Organization Type	(00gg,eee1)	1	The Volume Data accepted as input to this Volumetric Rendering Protocol. See Section C.XX.1.1.1

C.XX.1.1 Volumetric Rendering Protocol Module Attribute Descriptions

C.XX.1.1.1 Volume Organization Type

Sequential acquisitions may result in multiple Volume Data Image Sets in which the same anatomical volume is imaged at multiple times in order to capture images of a non-cyclic, time varying event. For example, imaging of the uptake of a tracer or contrast in a specific organ over time.

The Volume Organization Type (00gg,eee1) characterizes the Volume Data input to Volumetric Rendering.

Enumerated Values:

VOLUME

a single Volume Data set at a single point in time

TEMPORAL_VOLUME

multiple temporally related Volume Data sets that are spatially co-located. Examples include:

- a sequence of cardiac volume acquisitions acquired through a heart cycle, or a
- a sequence of volume acquisitions during multiple phases of passage of a contrast agent.

MULTIVOLUME

multiple Volume Data sets that are spatially separated. See Figure C.7.6.16-4 for an example.

C.XX.2 Volume Data Input Image Set Module

Table C.XX-2 specifies the attributes of the Volume Data Input Image Set Module.

Table C.XX-2 Volume Data Input Image Set Module

Name	Tag	Type	Description
Volume Data Input Image Set Specification Sequence	(00gg,eee2)	1	Constraints on attributes, values, and/or value ranges for the set of Image Instances or frames that will make up the input data for volumetric rendering. See Sections C.11.23.1 and C.XX.2.1 One or more items are permitted in this Sequence.
>Include Table 10.25-1 "Attribute Value Constraint Macro Attributes"			The same Attribute shall not appear in more than one Item in the Sequence with the same values for Selector Sequence Pointer (0072,0052) and Selector Sequence Pointer Items (0074,1057).

C.XX.2.1 Volume Data Input Image Set Module Attribute Descriptions

C.XX.2.1.1 Volume Data Input Image Set Selector Sequence

The Volume Data Input Image Set Selector Sequence (00gg,eee2) identifies one or more acceptable image sets intended for one or more Volume Data sets. An Image Set may be an entire Study, Series or multi-frame instance, or a subset thereof, and is identified by procedure, anatomy, modality or other attribute values.

Note

1. The following Attributes from image IODs are examples of some possible values for Selector Attribute (0072,0026) of Volume Data Input Image Set Selector Sequence (00gg,eee2). This is not a complete list:
   - Image Type (0008,0008) or Frame Type (0008,9007)
   - Anatomic Region Sequence (0008,2218)
   - Acquisition Contrast (0008,9209)
   - Acquisition Time (0008,0032)
   - Contrast/Bolus Agent (0018,0010)
   - Body Part Examined (0018,0015)
   - Scanning Sequence (0018,0020)
   - Echo Time (0018,0081)
   - Echo Number(s) (0018,0086)

840 - Protocol Name (0018,1030)
841 - Trigger Time (0018,1060)
842 - Image Trigger Delay (0018,1067)
843 - Trigger Window (0018,1094)
844 - Echo Pulse Sequence (0018,9008)
845 - Phase Contrast (0018,9014)
846 - Effective Echo Time (0018,9082)
847 - Laterality (0020,0060)
848 - Image Laterality (0020,0062)
849 - Dimension Index Value (0020,9157)

850 2. The Selector Attribute Macro allows selection of Private Creator Attributes

851 ### C.XX.3 Volume Definition Module

852 Table C.XY-1 specifies the attributes of the Volumetric Definition Module

853 **Table C.XX-3 Volume Definition Module**

Name	Tag	Type	Description
Volume Data Organization Sequence	(00gg,eee3)	1	Constraints on the characteristics of Volume Data. See Section C.XX.3.1.1. One or more Items shall be included in this Sequence.
>Include Table 10.25-1 "Attribute Value Constraint Macro Attributes"			The same Attribute shall not appear in more than one Item in the Sequence with the same values for Selector Sequence Pointer (0072,0052) and Selector Sequence Pointer Items (0074,1057).
Volume Data Sorting Sequence	(00gg,eee4)	3	Sequence that defines sorting criteria to be applied to the result Volume Data Organization Sequence (00gg,eee3). Defines the order in which the display algorithm and presentation parameters are applied. See Section C.XX.3.1.2. One or more Items shall be included in this Sequence.
>Include Table 10-20a "Extended Selector Attribute Macro Attributes"			
>Sorting Direction	(0072,0604)	1	Sorting direction to be applied to the value(s) in the image set of the Attribute identified by the Extended Selector Attribute Macro Enumerated Values: INCREASING DECREASING

C.XX.3.1 Volume Definition Module Attribute Descriptions

C.XX.3.1.1 Volume Data Organization Sequence

The Items in the Volume Data Organization Sequence (00gg,eee3) define the grouping of Image Instances, or frames within Image Instances, identified in the Volume Definition Module, into collections of one or more sets of Volume Data.

C.XX.3.1.2 Volume Data Sorting Sequence

The Items in the Volume Data Sorting Sequence (00gg,eee4) define the order in which the display algorithm and presentation parameters are applied to Volume Data resulting from the Volume Data Organization Sequence (00gg,eee3). The sorting criteria may include the value of a numeric, date, or time Attribute that is expected to be present in each of the image objects in the Volume Data. A sorting direction shall be associated with each sorting criterion. If a textual Attribute is used for sorting, then the INCREASING sorting direction indicates alphabetical order, and DECREASING indicates reverse alphabetical order.

If a Code Sequence Attribute is used for sorting, then Code Meaning (0008,0104) shall be sorted alphabetically. If a string numeric Attribute is used for sorting (VR of IS or DS), then sorting shall be on the numeric value, and padding shall be ignored. When sorting by date or time Attribute, then sorting shall be on the temporal value, not the alphabetic string.

If there are multiple Items in the Volume Data Sorting Sequence (00gg,eee4), then the sorting operations shall be applied in Item order. The least rapidly varying Attribute for the sorting operation shall be the first Item in the Sequence.

Add new Protocol Directory Record Type to Section F.3.2.2

Table F.3-3. Directory Information Module Attributes

Attribute Name	Tag	Type	Attribute Description
>Directory Record Type	(0004,1430)	1	Defines a specialized type of Directory Record by reference to its position in the Media Storage Directory Information Model (see Section F.4). Enumerated Values: **PATIENT** … **DEF PERF PROT** **PRIVATE** Privately defined record hierarchy position. Type shall be defined by Private Record UID (0004,1432).

Add new Protocol Directory Record Type to Section F.4 and update Figure.

Editorial note: this needs to be synchronized with cp1715.

Table F.4-1. Relationship Between Directory Records

Directory Record Type	Section	Directory Record Types that may be included in the next lower-level directory Entity
(Root Directory Entity)		PATIENT, HANGING PROTOCOL, … **DEF PROC PROT,** PRIVATE
DEF PROC PROT	**F.5.4x**	**PRIVATE**

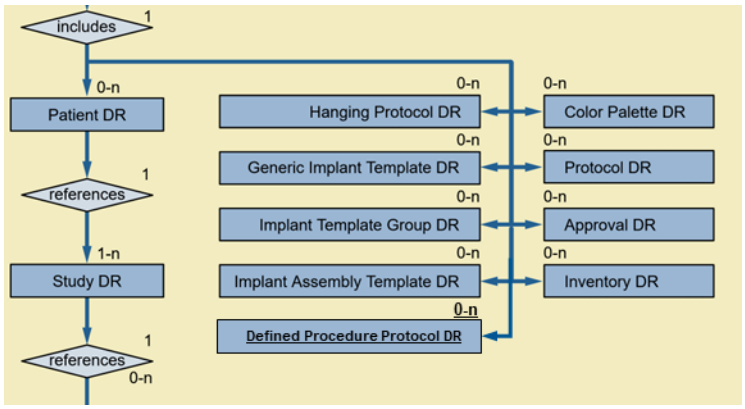


Figure F.4-1. Basic Directory IOD Information Model

Add new Protocol Directory Record Definition to Section F.5

F.5.4x Protocol Directory Record Definition

The Directory Record is based on the specification of Section F.3. It is identified by a Directory Record Type of Value "DEF PROC PROT". Table F.5-4x lists the set of keys with their associated Types for such a Directory Record Type. The description of these keys may be found in the Modules related to the Procedure Protocol IE of Protocol IODs. This Directory Record shall be used to reference a Protocol SOP Instance. This type of Directory Record may reference a Lower-Level Directory Entity that includes one or more Directory Records as defined in Table F.4-1.

Table F.5-4x. Protocol Keys

Attribute Name	Tag	Type	Attribute Description
Specific Character Set	(0008,0005)	1C	Required if an extended or replacement character set is used in one of the keys.
Protocol Name	(0018,1030)	1	
Instance Creation Date	(0008,0012)	1	
Instance Creation Time	(0008,0013)	2	
Any other Attribute of the Procedure Protocol IE Modules		3	

Modifications to PS3.4

Add Volumetric Rendering Protocol SOP Class to GG.3

GG.3 SOP Classes

The application-level services addressed by the Non-Patient Object Storage Service Class definition are specified in the SOP Classes specified in Table GG.3-1.

Table GG.3-1. Standard SOP Classes

SOP Class Name	SOP Class UID	IOD Specification (defined in PS3.3)
Hanging Protocol Storage	1.2.840.10008.5.1.4.38.1	Hanging Protocol IOD
Color Palette Storage	1.2.840.10008.5.1.4.39.1	Color Palette IOD
Generic Implant Template Storage	1.2.840.10008.5.1.4.43.1	Generic Implant Template IOD
Implant Assembly Template Storage	1.2.840.10008.5.1.4.44.1	Implant Assembly Template IOD
Implant Template Group Storage	1.2.840.10008.5.1.4.45.1	Implant Template Group IOD
CT Defined Procedure Protocol Storage	1.2.840.10008.5.1.4.1.1.200.1	CT Defined Procedure Protocol IOD
Protocol Approval Storage	1.2.840.10008.5.1.4.1.1.200.3	Protocol Approval IOD
XA Defined Procedure Protocol Storage	1.2.840.10008.5.1.4.1.1.200.7	XA Defined Procedure Protocol IOD
Inventory Storage	1.2.840.10008.5.1.4.1.1.201.1	Inventory IOD
Volumetric Rendering Protocol Storage	**1.2.840.10008.5.1.4.xxuid.1**	**Volumetric Rendering Protocol IOD**

Add new section for Volumetric Rendering Protocol to PS3.4 GG Non-Patient Object Storage:

GG.6.X Volumetric Rendering Protocol SOP Class

GG.6.X.1 Instance Creator

An implementation that conforms to the Volumetric Rendering Protocol Storage SOP Class as an SCU and is a SOP Instance creator shall state in its Conformance Statement:

- the Image Storage SOP Classes that are supported by the SCU and referenced in the Volumetric Rendering Protocol Storage SOP Class.

GG.6.X.2 Display Application

The following behavior shall be documented in the Conformance Statement of any implementation claiming conformance to a Volumetric Rendering Protocol Storage SOP Class as an SCP and interprets the contents of instances of the SOP Class to affect 3D rendering:

- the Image Storage SOP Classes that are supported by the SCP and referenced in the Volumetric Rendering Protocol Storage SOP Class.

Modifications to PS3.6

Update PS3.6 Table 6-1 Registry of DICOM Data Elements as follows:

Table 6-1. Registry of DICOM Data Elements

Tag	Name	Keyword	VR	VM	
…					
(00gg,eee1)	**Volume Type**	**VolumeType**	**CS**	**1**	
(00gg,eee2)	**Volume Data Input Image Set Specification Sequence**	**VolumeImageInputSetSelectorSequence**	**SQ**	**1**	
(00gg,eee3)	**Volume Organization Index Sequence**	**VolumeOrganizationIndexSequence**	**SQ**	**1**	
(00gg,eee4)	**Volume Data Sorting Sequence**	**MultivolumeVolumeRelationshipOrderSequence**	**SQ**	**1**	

Update PS3.6 Table A-1 UID Values as follows:

Table A-1. UID Values

UID Value	UID NAME	UID Type	Part
…			
1.2.840.10008.5.1.4.xxuid.1	**Volumetric Rendering Protocol Storage**	**SOP Class**	**PS3.4**

Modifications to PS3.15

Modify Section C.2 as follows:

C.2 Creator RSA Digital Signature Profile

The creator of a DICOM SOP Instance may generate signatures using the Creator RSA Digital Signature Profile. The Digital Signature produced by this Profile serves as a lifetime data integrity check that can be used to verify that the pixel data in the SOP instance has not been altered since its initial creation. An implementation that supports the Creator RSA Digital Signature Profile may include a Creator RSA Digital Signature with every SOP Instance that it creates; however, the implementation is not required to do so.

The signature shall use one of the RIPEMD-160, MD5, SHA-1 or SHA-2 family (SHA256, SHA384, SHA512) of hashing functions to generate a MAC, which is then encrypted using a private RSA key. All validators of digital signatures shall be capable of using a MAC generated by any of the hashing functions specified (RIPEMD-160, MD5, SHA-1 or SHA256, SHA384, SHA512).

As a minimum, an implementation shall include the following Attributes in generating the Creator RSA Digital Signature:

a. the SOP Class and Instance UIDs

b. the SOP Creation Date and Time, if present

c. the Study and Series Instance UIDs

d. any Attributes of the General Equipment Module **and the Enhanced General Equipment Module** that are present

…

x. any Attributes of the Protocol Context Module that are present

x. any Attributes of the Instructions Module that are present

x. any Attributes of the Volumetric Rendering Protocol Module that are present

x. any Attributes of the Volume Data Input Image Set Module that are present

x. any Attributes of the Volume Definition Module that are present

x. any Attributes of the Volume Render Geometry Module that are present

x. any Attributes of the Render Shading Module that are present

x. any Attributes of the Render Display Module that are present

x. any Attributes of the Multi-Planar Reconstruction Geometry Module that are present

x. any Attributes of the MPR Volumetric Presentation State Display Module that are present

x. any Attributes of the VOI LUT Module that are present

x. any Attributes of the Presentation Animation Module that are present

Modifications to PS3.17

Add the following Section after Section XXX.6:

XXX.x Scope of DICOMweb API for Server Volumetric Rendering

The web services enable a user agent to request server-side 3D volumetric rendering. In this service, input instances with geometric consistency are identified and reconstructed into volume data. Algorithm and display parameters are applied to the volume data in order to achieve the requested presentation, and lastly, the representation is encoded into one or more images of the requested media type and returned in a response payload to the user agent.

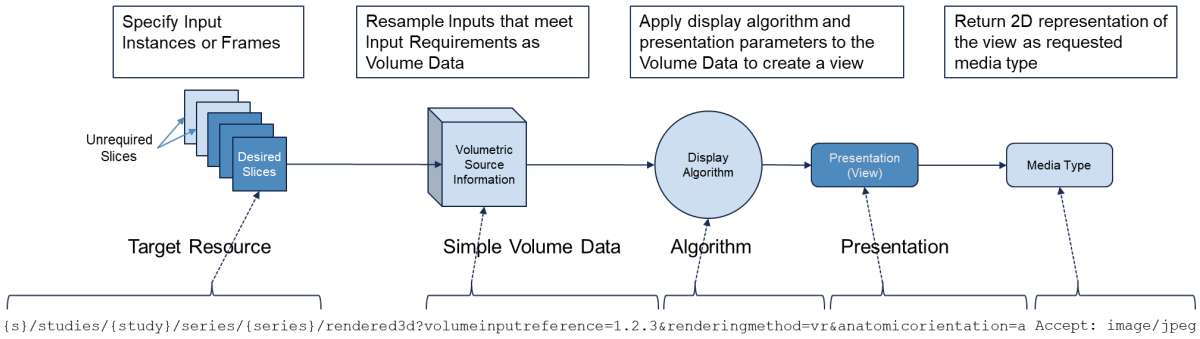


Figure XXX.x-1 Scope of DICOMweb API for Server Volumetric Rendering

XXX.x.1 Converting MPR Orientation to Viewpoint Attributes

The Rendered 3D and Rendered MPR camera orientation parameters for server-side Volumetric Rendering, like the Volume Rendering Volumetric Presentation State IOD, specifies orientation from the perspective of a camera in the VPS-RCS with three parameters consisting of:

- two points: “viewpointposition” and "viewpointlookat",
- and one vector, “viewpointup”.

Conversely, the Planar MPR Volumetric Presentation State IOD specifies orientation of the MPR slab as a direction cosine (x,y,z), in the MPR View Width Direction (0070,1507) and MPR View Height Direction (0070,1511) attributes.

MPR slab orientation attributes can be converted to camera attributes as follows:

Where:

T_{xyz} = coordinates of the MPR Top Left Hand Corner (0070,1505) in mm
W = MPR View Width (0070,1508) in mm
H = MPR View Height (0070,1512) in mm
X_{xyz} = values from the direction cosine of the MPR View Width Direction (0070,1507)
Y_{xyz} = values from the direction cosine of the MPR View Height Direction (0070,1511)
Z_{xyz} = the vector cross products of X_{xyz} and Y_{xyz}

`viewpointlookat` = $T_{xyz} + X_{xyz} * W / 2 + Y_{xyz} * H / 2$

`viewpointposition` = Viewpoint LookAt Point (0070,1604) + Z_{xyz}

`viewpointup` = Y_{xyz}

XXX.x.2 Animation Parameters

In an animated rendering, the user agent request includes:

- rendering parameters that establish the initial view, and
- animation parameters that specify:
  - volumetric curve point coordinates,
  - the animation step size,
  - and the animation rate.

See Section XXX.3.4.1 for an example.

For more complex animations, such as fly-through, the Volumetric Presentation Animation Module within the Volumetric Protocol or Volumetric Presentation State should be used.

XXX.y Scope of Volumetric Rendering Protocol IOD

The Volumetric Rendering Protocol IOD specifies criteria for, and organizes image set inputs into Volume Data, and specifies the Volumetric Transformations to be applied. This section provides examples of the Volumetric Rendering Protocol and Volume Data Input Image Set Modules. For examples or Procedure Protocol IE Modules refer to Section AAAA. For examples of Presentation State IE Modules, refer to Section XXX.3.

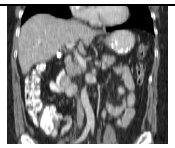
XXX.y.1 CT Temporal Volume Encoding Example

In this example, three CT acquisitions through the liver are obtained, each corresponding to a contrast phase (arterial, portal-venous and venous). All images are in a single series of Legacy CT Image objects. The scanner used to acquire the images increments Acquisition Number (0020,0012) for each contrast phase in the series:

1 = arterial
2 = portal-venous
3 = venous

A Volumetric Rendering Protocol is defined to identify contrast-enhanced input instances based on Protocol Name (0018,1030), and presence of Contrast/Bolus Agent (0018,0010). Instances are grouped into Volume Data based on Acquisition Number (0020,0012). Each resulting volume (i.e., phase) is rendered as a temporal MIP coronal slab, starting with the arterial contrast-enhanced phase. Once rendered, the operator can select each phase in the application user interface.

Table XXX.y-1. CT Temporal Volume Encoding Example

Name	Value
Volumetric Rendering Protocol Module	
Reformatting Operation Type	MPR
Rendering Method	AVERAGE_IP
Icon Image Sequence	
Volume Organization Type	TEMPORAL_VOLUME
Volume Data Input Image Set Module	
Volume Data Input Image Set Specification Sequence	
%item1	
>Selector Attribute Name	SOP Class UID
>Selector Attribute VR	UI
>Selector Attribute	(0008,0016)
>Selector Value Number	1
>Constraint Type	EQUAL
>Constraint Violation Significance	FAILURE
>Constraint Value Sequence	
>%item1	
>>Selector UI Value	1.2.840.10008.5.1.4.1.1.2
>%enditem	
%enditem	

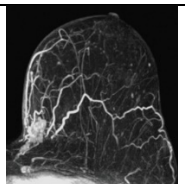
%item2	
>Selector Attribute Name	Protocol Name
>Selector Attribute VR	LO
>Selector Attribute	(0018,1030)
>Selector Value Number	1
>Constraint Type	EQUAL
>Constraint Violation Significance	WARNING
>Constraint Value Sequence	
>%item1	
>>Selector LO Value	"3-phase liver"
>%enditem	
%item3	
>Selector Attribute Name	Contrast/Bolus Agent
>Selector Attribute VR	LO
>Selector Attribute	(0018,0010)
>Selector Value Number	1
>Constraint Type	UNCONSTRAINED
>Constraint Violation Significance	WARNING
%enditem	
Volume Definition Module	
Volume Data Organization Sequence	
%item1	
>Selector Attribute Name	Acquisition Number
>Selector Attribute VR	IS
>Selector Attribute	(0020,0012)
>Selector Value Number	1
>Constraint Type	LESS_OR_EQUAL
>Constraint Violation Significance	FAILURE
>Constraint Value Sequence	
>%item1	
>>Selector IS Value	3
>%enditem	
%enditem	
Volume Data Sorting Sequence	
%item1	
>Selector Attribute Name	Acquisition Number
>Selector Attribute VR	IS
>Selector Attribute	(0020,0012)
>Selector Value Number	1
>Sorting Direction	INCREASING
%enditem	

XXX.y.2 MR Temporal Volume Encoding Example

In this example, Dynamic Contrast Enhanced (DCE) MR acquisition consisting of 5 phases is obtained through the breast. The first phase is non-contrast, phases 2-5 are contrast enhanced. All phases are encoded in single enhanced MR object. Phases are identified by the Temporal Position Index (0020,0100).

A Volumetric Rendering Protocol is defined to identify DCE input frames based on Image Type (0008,0008) and Temporal Position Index (0020,0100). Frames are grouped into Volume Data based on the Temporal Position Index (0020,0100). Each resulting volume (i.e., phase) is rendered as a temporal 3D MIP, starting with phase 2, the earliest contrast-enhanced phase. Once rendered, the operator can select each phase in the application user interface, or display a temporal loop demonstrating contrast enhancement over time.

Table XXX.y-2. MR Temporal Volume Encoding Example

Name	Value
Volumetric Rendering Protocol Module	
Reformatting Operation Type	3D_RENDERING
Rendering Method	MAXIMUM_IP
Icon Image Sequences	
Volume Organization Type	TEMPORAL_VOLUME
Volume Data Input Image Set Module	
Volume Data Input Image Set Specification Sequence	
%item1	
>Selector Attribute Name	SOP Class UID
>Selector Attribute VR	UI
>Selector Attribute	(0008,0016)
>Selector Value Number	1
>Constraint Type	EQUAL
>Constraint Violation Significance	FAILURE
>Constraint Value Sequence	
>%item1	
>>Selector UI Value	1.2.840.10008.5.1.4.1.1.4.1
>%enditem	
%enditem	
%item2	
>Selector Attribute Name	Image Type
>Selector Attribute VR	CS
>Selector Attribute	(0008,0008)
>Selector Value Number	3
>Constraint Type	EQUAL
>Constraint Violation Significance	FAILURE
>Constraint Value Sequence	

>%item1	
>>Selector CS Value	DYNAMIC
>%enditem	
%enditem	
Volume Definition Module	
Volume Data Organization Sequence	
%item1	
>Selector Attribute Name	Temporal Position Index
>Selector Attribute VR	UL
>Selector Attribute	(0020,0100)
>Selector Value Number	1
>Constraint Type	GREATER_OR_EQUAL
>Constraint Violation Significance	FAILURE
>Constraint Value Sequence	
>%item1	
>>Selector IS Value	2
>%enditem	
%enditem	
Volume Data Sorting Sequence	
%item1	
>Selector Attribute Name	Temporal Position Index
>Selector Attribute VR	UL
>Selector Attribute	(0020,0100)
>Selector Value Number	1
>Sorting Direction	INCREASING
%enditem	

1029