ADAS 3D EXPORT FILE FORMATS SPECIFICATION

PRODUCT DETAILS:

ADAS 3D v2.12.1
www.adas3d.com

MANUFACTURER:

ADAS3D Medical
Rambla Catalunya 53, 4-H
08007 Barcelona - Spain
Tel: +34 933283964
www.adas3d.com
info@adas3d.com

| May 8, 2023 | Version 2.2 |

Table of Contents

ADAS VTK File Format Specification

Purpose

This section describes the VTK file format specification of the VTK files exported from ADAS 3D. The Visualization ToolKit (VTK) is an open-source software that is available in the following website: http://www.vtk.org/.

The VTK files exported from ADAS 3D can be imported to EP navigation systems, such as CARTO 3 version 6.0 (Biosense Webster) and Navigant™ version 5.0.6.27 (Stereotaxis).

Scope

This document assumes that the reader is familiar with the VTK file format and focuses on describing the specific VTK format, version and content that is exported from ADAS 3D.

References

VTK file format: http://www.vtk.org/VTK/img/file-formats.pdf

VTK viewer: A common viewer can be downloaded from:
http://www.paraview.org/paraview/resources/software.html

VTK exported file format specification

File Extension

The file extension of the exported files from ADAS 3D is ".vtk".

File Version

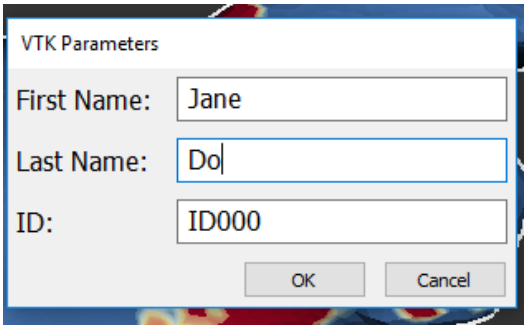
ADAS 3D exports legacy VTK files formats of version 4.1. The first line of the file is the following:

 # vtk DataFile Version 4.1

File Header

The second part of the file is the header. The header consists of a character string terminated by end-of-line character \n. The header is 256 characters maximum. The header is used to store patient information and any other pertinent information related to the ADAS Study where the VTK file was generated.

The patient information fields are filled with the information that the user introduces in the window displayed when exporting the data. Please see below an image showing this window:



File Header for CARTO

The header consists of a single line that includes the patient details as follows:

PatientData Jane Do 000

The first word is *PatientData* immediately followed by three fields separated by spaces defining the patient details in this order: first name, last name, patient Id.

File Header for Navigant

The header consists of 4 lines, each line terminated by the \r carriage return character.

PatientData [Patient name and/or Patient ID]\r

Structure [Anatomical structure name]\r

Provider [ADAS 3D version]\r

StudyDate [Date of the study]\r\n

Field	Description
PatientData [*string*]	Patient's name with ID information.
Structure [*string*]	Name of the anatomical structure contained in the surface file.
Provider [*string*]	Version of the ADAS 3D software that generated this VTK file.
StudyDate [*string*]	Date of study in the format 2019-01-01.

File Format

The third part is the file format. It indicates that it is a text file and not a binary file:

ASCII

Dataset Definition

The next part is the dataset definition. The dataset is always ***POLYDATA*** type.

DATASET POLYDATA

The Field Data

This part is only generated when exporting to Navigant. It includes the orientation of the data and the color information.

FIELD FieldData 8

Left 1 1 int

1

Anterior 1 1 int

-2

Superior 1 1 int

3

MmPerUnit 1 1 float

1

DefaultColor 1 3 int

255 255 255

Transparency 1 1 float

1

CLUTRangeMin 1 1 float

0

CLUTRangeMax 1 1 float

1

The only fields that are modified depending on each exported structure are DefaultColor and Transparency.

The Points Section

The next part is the points section, defining the points of the 3D structure. In the following example a 3D structure is described containing 4 points. Each point is defined by three float values defining its position (x, y, z) in space. The x, y, z coordinates are in millimeter units.

POINTS 4 float

-2.0 -1.0 0.0

2.0 -1.0 0.0

0.0 2.0 0.0

0.0 0.0 3.0

The Polygons Section

The next part is the polygons section. Only triangles are available in this section, although the VTK format supports also higher-level polygons.

The following example shows 4 triangles, where each triangle is defined using four numbers: the first number is 3 which specifies the number of points defining the polygon. The following three numbers are the triangle points indices (zero based).

POLYGONS 4 16

3 0 2 1

3 0 1 3

3 1 2 3

3 2 0 3

Scalars Section

The next part is the scalars. This part is optional. For each point, a float number between 0 and 1 is defined. The color (RGBA) that is used to visualize each point of the 3D structure is assigned using a lookup table that maps the float number to an RGBA color.

POINT_DATA 4

SCALARS my_scalars float

LOOKUP_TABLE custom_table

0 0 0 0.6

Look up Table Section

The next section is the look up table. This section is an optional one. The color scale defines a one-dimensional texture. This texture is used for mapping the texture coordinates range (between 0 to 1) to RGBA colors. The RGBA colors are defined by four float numbers between 0 to 1 where the last one (alpha) defines opacity where alpha=0 is transparent and alpha=1 is opaque. ADAS 3D will export alpha=1 for all colors (all colors have maximum opacity).

In this example a color scale is composed of five colors, red (1,0,0,1), green (0,1,0,1), blue (0,0,1,1), yellow (1,1,0) and pink (1,0,1,1). Since the color scale range is from 0 to 1, textures coordinates values from 0 to 0.2 will map to red, values from 0.2 to 0.4 will map to green, values from 0.4 to 0.6 will map to blue, values from 0.6 to 0.8 will map to yellow and values from 0.8 to 1 will map to pink.

LOOKUP_TABLE custom_table 5

1.0 0.0 0.0 1.0

0.0 1.0 0.0 1.0

0.0 0.0 1.0 1.0

1.0 1.0 0.0 1.0

1.0 0.0 1.0 1.0

Point Normals Section

Next, the point normals section is defined. For CARTO, the normals point towards the inside of the 3D structure and for Navigant, the normals point towards the outside of the 3D structure. The normals do not necessarily have to be unit vectors. However they cannot be zero vectors (0,0,0).

NORMALS Normals float

0 0 -1

0 -3 1

9 6 4

-9 -6 4

VTK Sample File

 # *vtk DataFile Version 4.1*

PatientData ADAS Example 1

ASCII

DATASET POLYDATA

POINTS 4 float

-2.0 -1.0 0.0

2.0 -1.0 0.0

0.0 2.0 0.0

0.0 0.0 3.0

POLYGONS 4 16

3 0 2 1

3 0 1 3

3 1 2 3

3 2 0 3

POINT_DATA 4

SCALARS my_scalars float

LOOKUP_TABLE custom_table

0.0 0.0 0.0 0.60

LOOKUP_TABLE custom_table 5

1.0 0.0 0.0 1.0

0.0 1.0 0.0 1.0

0.0 0.0 1.0 1.0

1.0 1.0 0.0 1.0

1.0 0.0 1.0 1.0

NORMALS Normals float

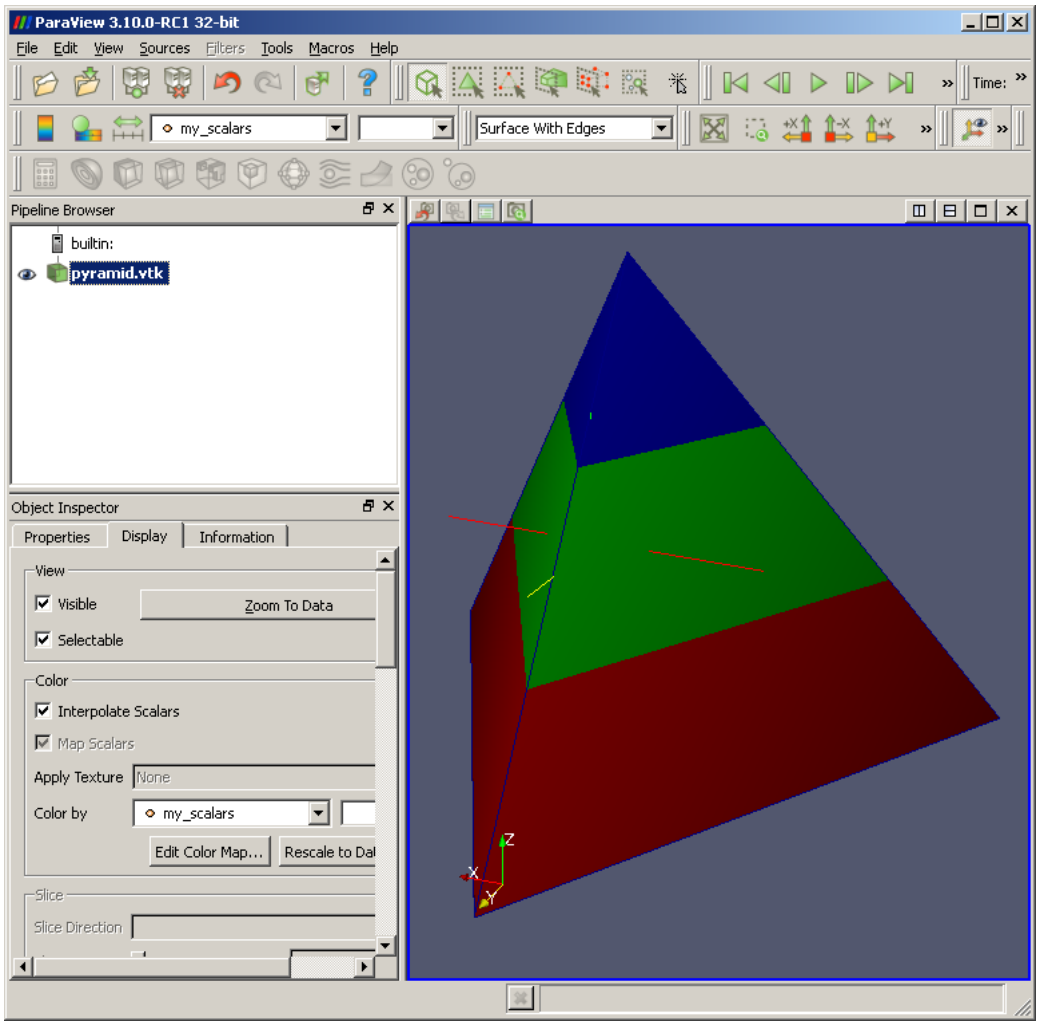
0 0 -1

0 -3 1

9 6 4

-9 -6 4

In this example all the base points are red since all of them have 0 as texture coordinates. The part of the color scale between 0 to 0.6 (containing the red, green and blue colors) is projected on the pyramid faces up to the head. See the print-screen below.



ADAS VTK Tables of definitions

Table 1: VTK File Header

Code	Description
#vtk DataFileVersion [version]	File version
PatientData [Firstname] [Lastname] [ID]	File header
ASCII	File Format
DATASET POLYDATA	Dataset Definition
# vtk DataFile Version 4.1 PatientData FirstName LastName ID ASCII DATASET POLYDATA	

Table 2: VTK File Header for Stereotaxis

Code	Description
#vtk DataFileVersion [version]	File version
PatientData [Patient name and Patient ID]\r Structure [Anatomical structure name]\r Provider [ADAS 3D version]\r StudyDate [Date of the study]\r\n	File header
ASCII	File Format
DATASET POLYDATA	Dataset Definition
FIELD FieldData 8	Field Data
# vtk DataFile Version 4.1 Provider ADAS 3D 2.8.0 PatientData FirstName LastName PatientID StudyDate 2020-11-30 Structure Aorta-DE-MRI ASCII DATASET POLYDATA FIELD FieldData 8 Left 1 1 int 1 Anterior 1 1 int -2 Superior 1 1 int 3 MmPerUnit 1 1 float 1 DefaultColor 1 3 int 127 0 255 Transparency 1 1 float	

0.6
CLUTRangeMin 1 1 float
0
CLUTRangeMax 1 1 float
1

Table 3: Points Section

Code	Description
Points [# of points] float	Points are 3D vertex coordinates
[X] [Y] [Z] [X] [Y] [Z] [X] [Y] [Z] …	Rows of coordinates – each three float numbers describe a vertex
POINTS 9743 float 68.8457 -77.5449 22.1253 69.4049 -77.1077 22.8217	

Table 2: Polygon Section

Code	Description
POLYGONS [# of polygons] [total number of values in the following rows]	
[# of vertices in polygon] [VertexD1] [Vertex D2] [VertexD3] … [VertexDn]	
VertexID – is the vertex's row number in the POINTS section	
POLYGONS 13429 53716 3 0 1 2 3 3 0 2 3 3 2 4	

Table 3: Scalars section

Code	Description
POINT_DATA [# of points we wish to color – should be the same as # of vertices]	
SCALARS [dataName – optional] [float]	
LOOKUP_TABLE tableName	Color lookup table: determines what color to give to each vertex by defining the color palette. If the palette is not defined, then the default VTK palette is used.
[Color for vertex0] [Color for vertex1] … [Color for vertexN]	
Color is a number of type "float" between 0 and 1, inclusive.	
POINT_DATA 9743 SCALARS my_scalars float	

```
LOOKUP_TABLE custom_table
0.849121 0.874268 0.874268 0.823975 0.823975 0.823975 0.823975
```

Table 4: Lookup table Section

Code	Description
LOOKUP_TABLE [name of table, used in last slide] [number of colors that follow]	
[Red] [Green] [Blue] [Alpha – Transparency]	
Each color/alpha is a “float” value between (0,1)	

```
LOOKUP_TABLE custom_table 6
1.0 0.0 1.0 1.0
0.0 0.0 1.0 1.0
0.0 1.0 1.0 1.0
```

ADAS DIF 5.0 File Format Specification

Purpose

This document specifies the format for Digital Image Fusion (DIF) files, version 5.0 exported from ADAS3D. The DIF supports the transfer of structures for import and use in the EnSite Precision system.

DIF Specification

DIF files are XML documents encoded in UTF-8.

DIF File Naming

DIF filenames use the ".xml" filename extension.

DIF XML Format

The structure of the ADAS 3D DIF 5.0 XML document is shown below. Each element and its corresponding attributes are described in subsequent sections.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<DIF>
        <DIFHeader>
                <Version> </Version>
                <VendorVersion> </VendorVersion>
                <PatientName> </PatientName>
                <PatientID> </PatientID>
                <PatientBirthDate> </PatientBirthDate>
                <PatientGender> </PatientGender>
                <StudyID> </StudyID>
                <SeriesNumber> </SeriesNumber>
                <StudyDate> </StudyDate>
                <StudyTime> </StudyTime>
                <Modality> </Modality>
                <RefPhysName> </RefPhysName>
                <StudyDesc> </StudyDesc>
                <SeriesDesc> </SeriesDesc>

                <OperatorName> </OperatorName>

                <OperatorComments> </OperatorComments>
                <SegmentationDate> </SegmentationDate>
                <SegmentationTime> </SegmentationTime>
        </DIFHeader>
        <DIFBody>
                <Volumes number="">
                        <Volume name="" color="">
                                <Vertices number=""> </Vertices>
                                <Normals number=""> </Normals>
                                <Polygons number=""> </Polygons>
                                <Maps numMaps="">
                                        <Map name="" idMap="" mapSurface="" refColorMap=""
                                        colorLow="" colorHigh="">
                                                <MaskData numElements=""> </MaskData>
                                                <MapData numElements="" dataName="" dataType=""
                                        scale=""> </MapData>
                                        </Map>
                                </Maps>
                        </Volume>
                </Volumes>
                <ColorMaps numColorMaps="">
```

```
                    <ColorMap name="" numElements="" idColorMap=""> </ColorMap>
                </ColorMaps>
                <Labels number="">
                    <Label name=""> </Label>
                </Labels>
                <ObjectMap> </ObjectMap>
        </DIFBody>
</DIF>
```

XML Declaration

DIF files contains an XML Declaration as the first line.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
```

DIF Element

The top level container element is **DIF**. It contains a **DIFHeader** element for header information about the patient and the scan, and a **DIFBody** element for the segmentation data.

```
<DIF>
        <DIFHeader> </DIFHeader>
        <DIFBody> </DIFBody>
</DIF>
```

DIFHeader Element

The required **DIFHeader** container element contains header information about the patient and the scan.

Parent Elements: **<DIF>**
Children Elements: **<Version>**, **<VendorVersion>**, **<PatientName>**, **<PatientID>**, **<PatientBirthDate>**, **<PatientGender>**, **<StudyID>**, **<SeriesNumber>**, **<StudyDate>**, **<StudyTime>**, **<SeriesDate>**, **<SeriesTime>**, **<Modality>**, **<RefPhysName>**, **<StudyDesc>**, **<SeriesDesc>**, **<OperatorName>**, **<OperatorComments>**, **<SegmentationDate>**, **<SegmentationTime>**

```
<DIFHeader>
        <Version> </Version>
        <VendorVersion> </VendorVersion>
        <PatientName> </PatientName>
        <PatientID> </PatientID>
        <PatientBirthDate> </PatientBirthDate>
        <PatientGender> </PatientGender>
        <StudyID> </StudyID>
        <SeriesNumber> </SeriesNumber>
        <StudyDate> </StudyDate>
        <StudyTime> </StudyTime>
        <SeriesDate> </SeriesDate>
        <SeriesTime> </SeriesTime>
        <Modality> </Modality>
        <RefPhysName> </RefPhysName>
        <StudyDesc> </StudyDesc>
        <SeriesDesc> </SeriesDesc>
        <OperatorName> </OperatorName>
        <OperatorComments> </OperatorComments>
        <SegmentationDate> </SegmentationDate>
        <SegmentationTime> </SegmentationTime>
</DIFHeader>
```

Version Element

The required Version element specifies the DIF version number for correct parsing by the EnSite Velocity System. This document specifies the format for DIF Version 5.0:

SJM_DIF_5.0

VendorVersion Element

The **VendorVersion** element is provided to store the current product and version identifier of the software used to generate the DIF file: **Galgo Medical S.L. - ADAS 3D 2.1.1**

PatientName Element

The **PatientName** element is provided to store the patient name.

PatientID Element

The **PatientID** element is provided to store a patient ID string.

PatientBirthDate Element

The **PatientBirthDate** element is provided to store the patient's birthdate.

PatientGender Element

The **PatientGender** element is provided to store the patient's gender. The value can be "M" (male), "F" (female) or "O" (other).

StudyID Element

The optional **StudyID** element is provided to store a study ID string. Use of study and series data fields follows DICOM convention where a series refers to a set of images and a study may consist of one or more series. DIF files are typically associated with a single image series.

Value: Freeform text of type: **xs:string**

SeriesNumber Element

The **SeriesNumber** element is provided to store a series number. A particular study may contain several image series.

Value: Freeform text of type: **xs:string**

StudyDate Element

The **StudyDate** element is provided to store the date of the study associated with the image series used to generate the DIF file.

Value: Date string of type: **xs:date**

StudyTime Element

The **StudyTime** element is provided to store the time of the study associated with the image series used to generate the DIF file.

Value: Time string of type: **xs:time**

SeriesDate Element

The SeriesDate element is provided to store the date of the image series.

Value: Date string of type: **xs:date**

SeriesTime Element

The SeriesTime element is provided to store the time of the image series.

Value: Time string of type: **xs:time**

Modality Element

The Modality element is provided to store the modality of the image series. For example: CT, MRI, etc.

Value: Freeform text of type: **xs:string**

RefPhysName Element

The **RefPhysName** element is provided to store the name of the referring physician.

Value: Freeform text of type: **xs:string**

StudyDesc Element

The **StudyDesc** element is provided to store a freeform text description of the study associated with the image series used to generate the DIF file.

Value: Freeform text of type: **xs:string**

SegmentationDate Element

The **SegmentationDate** element is provided to store the date of the image segmentation.

Value: Date string of type: **xs:date**

SegmentationTime Element

The **SegmentationTime** element is provided to store the time of the image segmentation.

Value: Time string of type: **xs:time**

DIF Body Subsection

The required **DIFBody** container element contains the 3-dimensional surface and any maps defined by the image segmentation.

```
<DIFBody>
        <Volumes> </Volumes>
        <ColorMaps> </ColorMaps>
        <Labels> </Labels>
        <ObjectMap> </ObjectMap>
</DIFBody>
```

Volumes Element

The required **Volumes** container element contains a sequence of **Volume** elements.

Parent Elements:	**<DIFBody>**
Children Elements:	**<Volume>**
Minimum Occurrences:	1
Maximum Occurrences:	1
Attributes:	**number** – Specifies the number of child elements of type **<Volume>**. use: Required type: **xs:unsignedInt** Constraints: The number of **<Volume>** child elements will be limited to 16
Value:	None

```
<Volumes number="">
        <Volume> </Volume>
</Volumes>
```

Volume Element

The **Volume** container element contains a sequence of elements that define geometric surface and map data. The surfaces are closed-manifold surfaces that consist entirely of triangles.

Parent Elements:	**<Volumes>**
Children Elements:	**<Vertices>**, **<Normals>**, **<Polygons>**, **<Maps>**
Minimum Occurrences:	0
Maximum Occurrences:	Unbounded
Attributes:	**name** – Specifies the name of the volume specified by the children elements. use: Required type: **xs:string** **color** – Specifies the color to display the surface associated with the volume. use: Required type: **DIF:color**
Value:	None

```
<Volume name="" color="">
        <Vertices> </Vertices>
        <Normals> </Normals>
        <Polygons> </Polygons>
        <Maps> </Maps>
</Volume>
```

Vertices Element

The **Vertices** element contains a list of vertices that are used to define polygons. Each vertex is a point in 3-D space representing the location where **Polygon** elements intersect. The number of vertices for each volume is limited to 15000.

Parent Elements:	**<Volume>**
Children Elements:	None
Minimum Occurrences:	1
Maximum Occurrences:	1
Attributes:	**number** – The number of vertices specified in the element. use: Required type: **xs:unsignedInt** Constraints: Less than 15000 vertices.
Value:	List of vertices used in defining polygons. Values are delimited by white space. type: **DIF:coordinate** Constraints: The number of vertices listed will match the value specified by the **number** attribute.

Normals Element

The **Normals** element contains a list of normal vectors. Each normal vector value specified is associated with a vertex value. The normal vector is assumed to point outward from the surface at the corresponding vertex.

Parent Elements:	**<Volume>**
Children Elements:	None
Minimum Occurrences:	0
Maximum Occurrences:	1
Attributes:	**number** – The number of vertices specified in the element.

	use: Required type: **xs:unsignedInt** Constraints: The value of the number attribute will match the number of vertices specified in the corresponding Vertices element.
Value:	List of normal vectors for each corresponding vertex. Values are delimited by white space. type: **DIF:coordinate** Constraints: The number of normals listed will match the value of the **number** attribute.

Polygons Element

The **Polygons** element contains a list of references to vertex indices. By specifying three vertices (by reference to an index assumed by the order of the corresponding **Vertices** elements), a triangle can be specified that defines a geometric surface in 3-D space. The sum of all polygons listed will create the closed geometric volume.

Parent Elements:	**<Volume>**
Children Elements:	None
Minimum Occurrences:	1
Maximum Occurrences:	1
Attributes:	**number** – The number of polygons specified in the element. use: Required type: **xs:unsignedInt**
Value:	List of triplets of vertex indices used to define polygons (triangles). The vertex index (one-based) refers to the position of the vertex within the corresponding Vertices element. Values are delimited by white space. type: **DIF:triangle** Constraints: • The vertices of a polygon will be ordered in a counter-clockwise direction when viewed from the outside of the volume defined by the vertices. • The number of polygons listed will match the value of the number attribute.

Maps Element

The optional Maps container element contains a sequence of Map elements.

Parent Elements:	**<Volume>**
Children Elements:	**<Map>**
Minimum Occurrences:	0
Maximum Occurrences:	1
Attributes:	**numMaps** – Specifies the number of child elements of type <Map>. use: Required type: **xs:unsignedInt**
Value:	None

```
<Maps numMaps="">
        <Map> </Map>
</Maps>
```

Map element

The **Map** container element contains elements that define a map. A map is a set of measurements associated with a geometrical position on a corresponding **Volume** element. For example, the measurement may be the MRI gray intensity level associated with a position on the geometrical surface.

Each map is associated with a vertex list within the corresponding Volume element. The default for these data points is the scalar type, but they can also be vectors or tensors.

Parent Elements:	**<Maps>**
Children Elements:	**<MaskData>**, **<MapData>**
Minimum Occurrences:	1
Maximum Occurrences:	Unbounded
Attributes:	**name** – Is used to specify a name for the map. use: Optional type: **xs:string** **idMap** – May be used to create a unique identifier for the map. use: Optional type: **xs:ID** **mapSurface** – Indicates that the values stored in the **MapData** element are associated with vertices stored in the **Vertices** element. use: Required type: Enumeration of type **xs:string**: { "**vertices**" } **refColorMap** – May be used to refer to a color map (stored in the ColorMap element with ID referenced by this attribute) that will be used to color the surface according to the map data. use: Optional type: **xs:IDREF** **colorLow** – Scalar value associated with the first color in the referenced color map. use: Required when **refColorMap** attribute is used for the current **Map**. type: **xs:decimal** **colorHigh** – Scalar value associated with the last color in the referenced color map. use: Required when **refColorMap** attribute is used for the current **Map**. type: **xs:decimal**
Value:	None

```
<Map name="" idMap="" mapSurface="" refColorMap="" colorLow="" colorHigh="">
        <MaskData> </MaskData>
        <MapData> </MapData>
</Maps>
```

MaskData

The optional **MaskData** element is a list of decimal values that indicate whether a value stored in the **MapData** element should be applied to the map. A value of 1.0 indicates that the corresponding value in the **MapData** element should be applied; a value of 0.0 indicates that the corresponding value in the **MapData** element should be ignored.

Parent Elements:	**<Map>**
Children Elements:	None
Minimum Occurrences:	0
Maximum Occurrences:	1
Attributes:	**numelements** – The number of values specified in the element. use: Required type: **xs:unsignedInt** Constrains: The number specified will match the number of corresponding **MapData** elements.
Value:	List of values to mask map data applied to the corresponding geometry. Values are delimited by white space. type: **xs:decimal** Constraints: • Values are restricted to the range [0.0, 1.0]. • The number of values will match the number specified in the **numElements** attribute.

MapData

The required **MapData** element is a series of scalar values that provide the map data for corresponding vertices or polygons.

Parent Elements:	**<Map>**
Children Elements:	None
Minimum Occurrences:	1
Maximum Occurrences:	unbounded
Attributes:	**numelements** – The number of values specified in the element. use: Required type: **xs:unsignedInt** Constraints: • The value for the **numElements** attribute will match the number of vertices or polygons (as specified by the corresponding **mapSurface** element) for the corresponding geometry.
Value:	List of map data values. Values are delimited by white space. type: **xs:decimal**

Colormaps Element

The optional **ColorMaps** container element contains a sequence of **ColorMap** elements.

Parent Elements:	**<DIFBody>**
Children Elements:	**<ColorMap>**
Minimum Occurrences:	0
Maximum Occurrences:	1
Attributes:	**numColorMaps** – Specifies the number of child elements of type **<ColorMap>**.

	use: Required type: **xs:unsignedInt**
Value:	None

<ColorMaps numColoarMaps="">
<ColorMap> </ColorMap>
</ColorMaps>

Colormap Element

The required **ColorMap** element contains a list of RGB values. The list is used as a lookup table to provide a color associated with scalar values stored in the **MapData** element. Further, the **Map** element includes **colorLow** and **colorHigh** attributes to indicate the range of data over which the color map is associated.

Parent Elements:	**< ColorMaps>**
Children Elements:	None
Minimum Occurrences:	1
Maximum Occurrences:	unbounded
Attributes:	**name** – May be used to specify the name for the color map. use: Optional type: **xs:string** **numElements** – The number of values specified in the element. use: Required type: **xs:unsignedInt** **idColorMap** – A unique identifier used to reference this color map from a **Map** element. use: Required type: **xs:ID**
Value:	List of sequential RGB value sets that define the color map. Value sets are delimited by white space. type: **xs:decimal** Constraints: • The number of value sets in the list will match the value specified by the corresponding **numElements** attribute.

Labels Element

The required **Labels** container element contains a sequence of **Label** elements. This element is not used and will always be 0.

ObjectMap Element

The objectMap element is not used and will always contain this information:

<ObjectMap>

<Rotation>1.000000 0.000000 -0.000000 0.000000 -1.000000 0.000000 -0.000000 0.000000 1.000000</Rotation>

<Translation>0.0000000 0.000000 0.0000000</Translation>

<Scaling>1.000000 1.000000 1.000000</Scaling>

<MD5Signature>00000000000000000000000000000000</MD5Signature>

</ObjectMap>

Example

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<DIF>
  <DIFHeader>
    <Version>SJM_DIF_5.0</Version>
    <VendorVersion>Galgo Medical S.L. - ADAS 3D 2.1.1</VendorVersion>
    <PatientName> </PatientName>
    <PatientID>FIBER7</PatientID>
    <PatientBirthDate>2002-03-01</PatientBirthDate>
    <PatientGender>M</PatientGender>
    <StudyID>0169592035</StudyID>
    <SeriesNumber>44</SeriesNumber>
    <StudyDate>2019-02-26</StudyDate>
    <StudyTime>15:11:35</StudyTime>
    <SeriesDate>2019-02-26</SeriesDate>
    <SeriesTime>15:11:35</SeriesTime>
    <Modality>MR</Modality>
    <RefPhysName></RefPhysName>
    <StudyDesc>ANGIORM VENAS PULMONARES Y VOLUMETRIA AU</StudyDesc>
    <SeriesDesc>3D TEST_5</SeriesDesc>
    <OperatorName></OperatorName>
    <OperatorComments></OperatorComments>
    <SegmentationDate>2019-02-26</SegmentationDate>
    <SegmentationTime>15:11:35</SegmentationTime>
  </DIFHeader>
  <DIFBody>
    <Volumes number="1">
      <Volume name="ENDO" color="">
        <Vertices number="10792">32.8181 ... 7.61853 </Vertices>
        <Normals number="10792">-0.446913 ... 0.768735 </Normals>
        <Polygons number="21580">3 ... 5900 </Polygons>
```

```
        <Maps numMaps="1">
          <Map name="Endocardium_colored" mapSurface="vertices" refColorMap="LUT40" colorLow="105.275962829589844" colorHigh="115.803596496582031">
            <MapData numElements="10792">152.75 ... 65.5235 </MapData>
            <MaskData numElements="10792">1 ... 1 </MaskData>
          </Map>
        </Maps>
      </Volume>
    </Volumes>
    <ColorMaps numColorMaps="1">
      <ColorMap name="LUT40" numElements="256" idColorMap="LUT40">0.270588 ... 0.14902 </ColorMap>
    </ColorMaps>
    <Labels number="0" />
    <ObjectMap>
      <Rotation>1.000000 0.000000 -0.000000 0.000000 -1.000000 0.000000 -0.000000 0.000000 1.000000</Rotation>
      <Translation>0.0000000 0.000000 0.0000000</Translation>
      <Scaling>1.000000 1.000000 1.000000</Scaling>
      <MD5Signature>00000000000000000000000000000000</MD5Signature>
    </ObjectMap>
  </DIFBody>
</DIF>
```