

AIM@SHAPE

Advanced and Innovative Models And Tools for the development of Semantic-based systems for Handling, Acquiring, and Processing knowledge Embedded in multidimensional digital objects

IST NoE No 506766

a project financed by the



Sixth Framework Programme

Deliverable D1.2.2.1



Ontology for Shape Acquisition and Processing 3rd version

Circulation:	¹PU
Partner(s):	DISI (<i>Leader</i>), IMATI, ITI, WEIZMANN, MPII, UNIGE, TECHNION, UU
Authors:	L. Papaleo, R. Albertoni, M. Pitikakis, F. Robbiano, G. Vasilakis
Contributors:	T. Hassner, S. Hammann, L. Mocozet, W. Saleem, A. Tal, F. Ter Haar, R. Veltkamp
Version:	04
Stage:	FINAL (100%)
Date:	15 February 2007

¹ Please indicate the dissemination level using one of the following codes:

PU = Public

PP = Restricted to other programme participants (including the Commission Services).

RE = Restricted to a group specified by the consortium (including the Commission Services).

CO = Confidential, only for members of the consortium (including the Commission Services).

Copyright

© Copyright 2007 The AIM@SHAPE Consortium

consisting of:

CNR-IMATI-GE	C.N.R. – Istituto di Matematica Applicata e Tecnologie Informatiche Dept. of Genova, Italy
DISI	Università di Genova – Dipartimento di Informatica e Scienze dell'Informazione, Italy
EPFL	École Polytechnique Federale de Lausanne, Switzerland
FhG/IGD	Fraunhofer Institut für Graphische Datenverarbeitung, Germany
INPG	Institut National Polytechnique de Grenoble, France
INRIA	Institut National de Recherche en Informatique et Automatique, France
ITI-CERTH	Informatics and Telematics Institut – Center for Research and Technology Hellas, Greece
UNIGE	Université de Genève, Switzerland
MPII	Max-Planck-Institut für Informatik, Germany
SINTEF	Stiftelsen for industriell og teknisk forskning ved Norges Tekniske Høgskole, Norway
TECHNION	Technion – Israel Institute of Technology, Israel
UU	Utrecht University, Netherlands
WEIZMANN	Weizmann Institute of Science, Israel

This document may not be copied, reproduced, or modified in whole or in part for any purpose without written permission from the AIM@SHAPE Consortium. In addition to such written permission to copy, reproduce, or modify this document in whole or part, an acknowledgement of the authors of the document and all applicable portions of the copyright notice must be clearly referenced.

All rights reserved.

This document may change without notice.

Document History

Vers.	Issue Date	Stage	Content and changes
1	29 December 2006	35%	Initial draft for the deliverable and content structure.
2	15 January 2007	50%	First draft version.
3	30 January 2007	75%	Second draft version. Content reviewed, general structure of the ontology added
4	15 February 2007	100%	Document completed

Executive Summary

This document contains a description of the deliverable **D1.2.2.1** of the IST NoE AIM@SHAPE.

The deliverable ***D1.2.2.1 – Ontology for Shape Acquisition and Processing 3rd Version*** – is intended to provide the third version of the ontology the AIM@SHAPE partners have developed for formalizing the knowledge related to the Acquisition and Processing of Shapes. The task leader is **DISI** and has been supported by all the involved partners.

This document presents the main evolution steps regarding the Ontology for Shape Acquisition and Processing with respect to the second version proposed in the previous deliverable D1.2.2.1.

Table of Contents

1	INTRODUCTION	5
2	SUMMARY OF THE SHAPE ACQUISITION AND PROCESSING ONTOLOGY	5
3	USAGE SCENARIOS: DESCRIPTIONS, CQS AND ANSWERS VIA THE SAP ONTOLOGY	5
3.1	USAGE SCENARIO N.1 “WEBBY AND THE SCANNING SESSION”	6
3.2	USAGE SCENARIO N.2: “WEBBY AND THE SHAPE PROCESSING”	6
3.3	USAGE SCENARIO N.3: “WEBBY AND THE HISTORY OF A SHAPE”	7
3.4	NEW USAGE SCENARIO N.4: “WEBBY AND THE WORKFLOWS”	8
4	FINALIZATION OF THE COMMON ONTOLOGIES AND INTEGRATION WITH THE SAP ONTOLOGY	9
4.1	ACTIVITIES FOR THE COMMON ONTOLOGIES	9
4.1.1	<i>MultiResolutionModel</i>	10
4.1.2	<i>NonManifoldMesh</i>	11
4.1.3	<i>RasterData (RasterData2D, RasterData3D, RasterData4D, MRI, DynamicMRI)</i>	12
4.2	INTEGRATION OF THE COMMON ONTOLOGIES IN THE SAP ONTOLOGY.....	18
5	LIST OF COMPETENCY QUESTIONS RELATED TO THE SAP ONTOLOGY	20
5.1	COMPETENCY QUESTIONS ANSWERED	20
5.2	COMPETENCY QUESTIONS NOT ANSWERED	21
6	DISSEMINATION ACTIVITIES FOR THE SAP ONTOLOGY	22
7	CONCLUDING REMARKS	23

1 INTRODUCTION

This document presents the main evolution steps regarding the Ontology for Shape Acquisition and Processing with respect to the second version proposed in the previous deliverable D1.2.2.1. We first give a brief summary of the SAP ontology and then we show all the improvements made for its third version.

2 SUMMARY OF THE SHAPE ACQUISITION AND PROCESSING ONTOLOGY

The original ontology for Shape Acquisition and Reconstruction has been renamed to *Ontology for Shape Acquisition and Processing*. From now on we will refer to this ontology as the Shape Acquisition and Processing Ontology or SAP ontology (SAPO). **Shape models** and **tools** are treated as resources that can be uploaded and downloaded together with their metadata in the DSW. We report here briefly the domain of the ontology and the identified target applications:

DOMAIN:

- *The development, usage and sharing of hardware tools, software tools and shape data by researchers and experts in the field of acquisition and reconstruction of shapes.*

TARGET APPLICATIONS (as example):

- *Acquisition Planning:* the process of selecting and interconnecting the most suitable acquisition devices for the purpose of obtaining a satisfactory 3D acquisition.
- *Data validation:* determination if data are accurate, complete, or meet specified criteria.
- *Benchmarking:* activities related to the comparison of aspects of performance (of functions or processes) in accordance with best practices; identifying gaps in performance.
- *Testing:* testing the functionality and correctness of a process (algorithm, method, approach).
- *Multi-sensor Data Fusion:* the study of the means and tools for the integration of data coming from multiple sensors.
- *Data enhancement (automatic recovery):* the process of enhancing the quality of the data with respect to some specific constraints and requirements.
- *Documentation:* the process of augmenting a digital model with information related to its acquisition and to its early processing phases

3 USAGE SCENARIOS: DESCRIPTIONS, CQS AND ANSWERS VIA THE SAP ONTOLOGY

In the second deliverable presenting the ontology we have introduced the three usage scenarios we have identified, which have been treated as requirement guidelines to prove the correctness of the designed ontology. We briefly summarize them here.

As explained in the previous deliverable (2nd version of the ontology), the first two usage scenarios are related to the two different macro-aspects of the SAP ontology: the *Acquisition* and the *Processing* of a shape. The third scenario, instead, focuses on the

knowledge related to the history of a digital shape, and it outlines the case in which a researcher is interested in knowing how a given digital shape has been produced.

Note that the first usage scenario has been successfully presented at the review meeting in March 2006, while the other two scenarios have been only defined and not yet proven. In this document we will present how the evolution of the ontology into its third version is now able to answer all the CQs involved in the three usage scenarios. We also present a fourth usage scenario, which regards possible workflows involving tools and shape types.

3.1 Usage Scenario n.1 “Webby and the scanning session”

Description: “Webby” works for partner UU in the Scanning Group. He would like to compare the quality of two different Acquisition Systems owned by AIM@SHAPE partners. So he would like to find all the objects owned by UU that could be scanned also elsewhere [CQ #1]. He finds three different real objects, from which he chooses FROG, because it is movable and its weight is reasonably low. Webby puts the FROG in his knapsack but he still does not know where to go. He first would like to find documentation on the Acquisition Sessions already performed on the FROG and in particular which tricks have been used in those sessions [CQ #4 and CQ #3]. Because no trick has been ever used to acquire the FROG, and because the FROG is a light absorbent real object, he poses the question [CQ #2]. He browses all the Acquisition Devices found and he discovers that one is owned by “MPII”. So he plans his trip to Germany.

These are the CQs related to this usage scenario:

- CQ #1. Which are the Real Objects owned by “UU”?
- CQ #2. What are the Acquisition Systems able to scan a Real Object which is light absorbent?
- CQ #3. What tricks have been performed in order to scan the Real Object “FROG”?
- CQ #4. Under which Lighting Conditions has the real object “FROG” been scanned?

With this usage scenario, we have been able to prove that the SAP ontology is capable to answer a subset of the competency questions identified in the first Deliverable D1.2.2.1. This usage scenario has been successfully presented during the AIM@SHAPE review meeting that was held in Genova, March 30-31, 2006.

3.2 Usage Scenario n.2: “Webby and the shape processing”

Description: Webby has acquired at MPII (using the Acquisition System of MPII) the real object FROG owned by UU. He has produced a surface mesh called “MPIIFrog.off”. Webby now wants to evaluate the quality of this latter mesh compared to the one obtained at UU, but he does not have any tool to compute the *distance* between two shapes. He searches for an appropriate tool [CQ #2]. Browsing the obtained results he chooses the “Metro” tool since it can run on Linux (the OS he usually uses) [CQ #3]. “Metro” is able to load only “PLY” files [CQ #4], but the MPIIFrog mesh is in OFF format. For this reason, Webby searches for a converter from OFF to PLY [CQ #1]. He browses the results and chooses the “Off2Ply” tool, since it also works on Linux.

We list here some CQs strictly related to the usage scenario above:

- CQ #1. Is there a Converter tool able to convert an OFF file into a PLY file?
- CQ #2. Which are the tools that are able to compute the Euclidean distance between two shapes?

CQ #3. Which is the compilation platform for this tool?

CQ #4. Which format type does this tool support?

Thanks to this second usage scenario, we have been able to identify a subset of competency questions we are addressing now regarding the tasks and tools that can use/manipulate shapes. Note that, even if this scenario is simple in terms of CQs, the ontology is also able to answer CQs that are related to tools (from the common tool ontology), therefore enlarging the effective knowledge included in and demonstrated by the ontology, such as, for example:

CQ #5. Which are the tools with functionality “xyz”?

CQ #6. Which tool has been used for producing as output the shape “abc.ply”?

CQ #7. Which are the tools supporting the “ply” format?

CQ #8. Which type of shape does this tool support?

3.3 Usage Scenario n.3: “Webby and the history of a shape”

Description: Webby wants to learn more on the acquisition process and surface meshes in general. He searches for all the surface meshes with a given characteristic (for example “no self-intersections”...). He browses the obtained results and chooses the shape “*bimbaConNastrino.ply*”. He asks for the history of this shape, discovering all the operations made in order to obtain the chosen digital shape.

Here we list some CQs related to the third usage scenario:

CQ #1. Which are the *SurfaceMeshes* with no self-intersection?

CQ #2. What is the History related to the *SurfaceMesh* “ItalianWorldCup.wrl”?

CQ #3. Who are the owners of the *RealObjects* source of the *SurfaceMesh* “flyingPotamo.ply”?

The third usage scenario basically identifies an important competency question regarding the history of a digital shape: *Which is the History of this shape?* In the second Deliverable 1.2.2.1 we have presented our evolution of the ontology in order to model the knowledge related to the history of a digital shape. In the last section of this document we present how the new structure works as expected, providing the type of answer the ontology offers.

To conclude, we would like to remark that the properties *hasHistory* could be useful in general to both the other domain ontologies we are developing within the network, namely the *Virtual Humans* and the *Product Design* ontologies. All the partners involved are actually working on and evaluating possible improvements.

In Figure 1 we show how the SAP ontology can actually answer the CQs identified by the experts related to the history of a digital shape. The transitivity of *hasHistory* (*hasPreviousSession*) produces the list of *ToolSession* and *AcquisitionSession* which worked together for the creation (*hasOutput*) of the given shape.

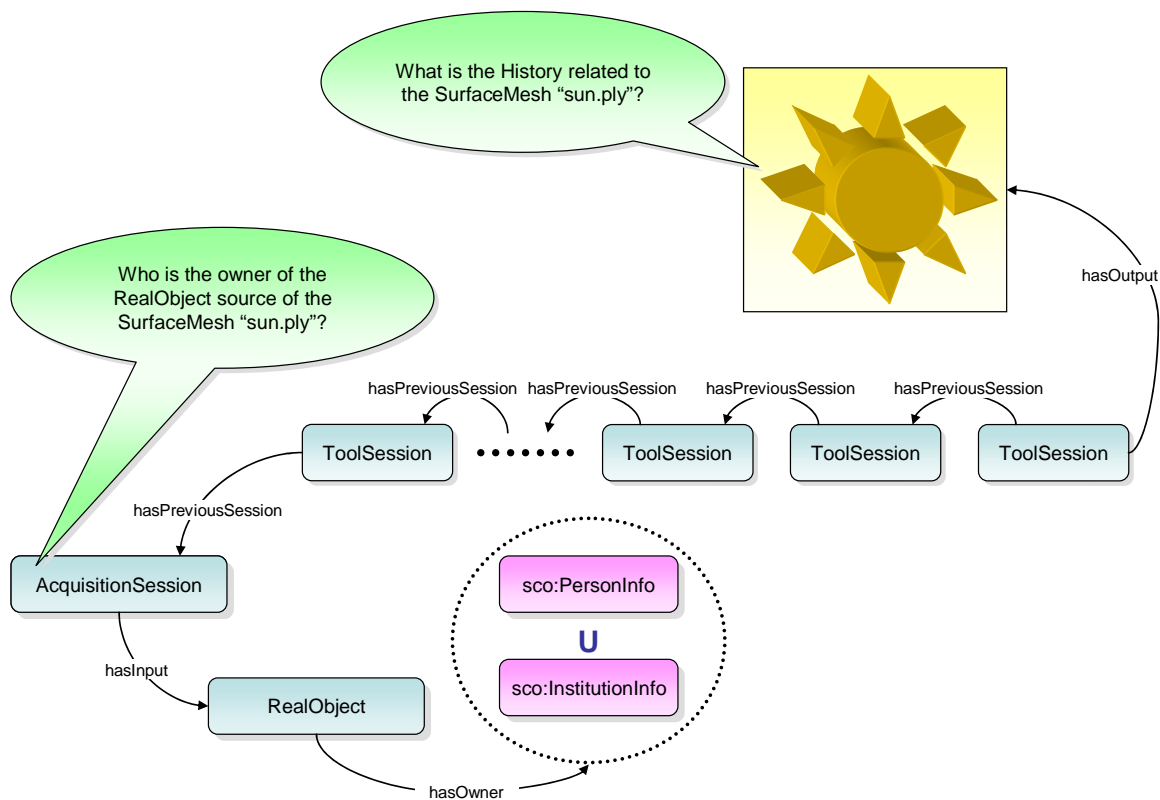


Figure 1 – An image showing two competency questions related to the history of a shape and the results obtained via the ontology

3.4 NEW Usage Scenario n.4: “Webby and the workflows”

Description: Webby is a young PhD student who just started his PhD in Shape Modeling and Reasoning. He does not know anything about the possible steps in surface reconstruction so he interrogates the ontology for learning about all the possible applicable shape processing steps starting from a *PointCloud* (the usual input in a shape reconstruction procedure), so that he can understand what can be done for creating new shapes starting from a cloud of points in space.

The involved competency questions are no more relative to a given shape in the DSW, but about the general concept of *ShapeType* (defined in our case in the common tool ontology). We list some of them here:

- CQ #1. What are the possible steps to perform starting with a point cloud?
- CQ #2. Through what steps is it possible to reach a surface mesh?
- CQ #3. Through what sequence of steps is it possible to reach a surface mesh starting from a point cloud?

The schema regarding steps as presented in the previous Deliverable D1.2.2.1 of the Ontology for Shape Acquisition and Processing, can easily answer the first CQ (we show the result in the last section of this document). Figure 2 shows the result from the ontology.

For CQs as the second and the third one listed above, we need to extend the ontology to support the extraction of the “history” of a given step in order to extract template workflows which satisfy the constraints given by the CQs. We also need to define the

concept of "sequence of steps". We are actually working in this direction for evolving the ontology which, in the end, will be able to answer such CQs. Recall that these types of competency questions are really interesting for the research community, since they actually require a formalization of an interdisciplinary knowledge regarding shapes and tools.

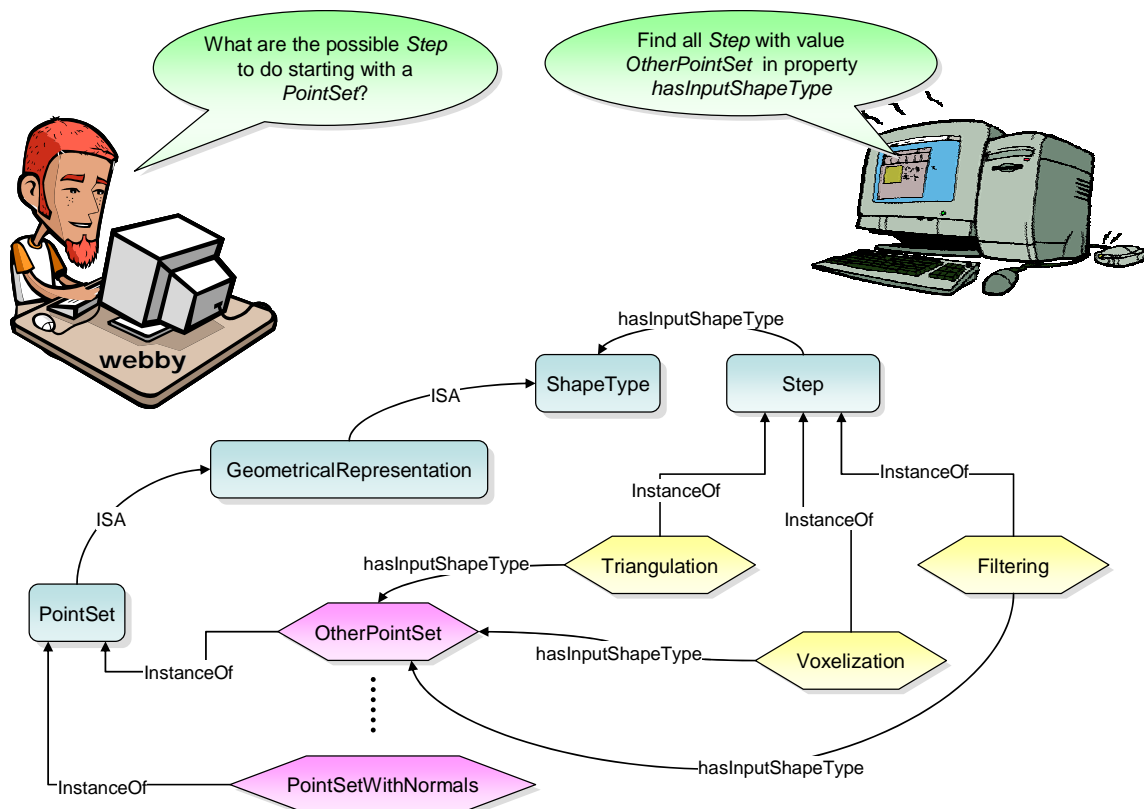


Figure 2 - The first question of the fourth usage scenario "webby and the workflows" with the answer that can be obtained interrogating the ontology as it is now. In yellow the instances of *Step* retrieved (*Triangulation*, *Voxelization*, *Filtering*), since they have as input shape type the instance of *OtherPointSet*

4 FINALIZATION OF THE COMMON ONTOLOGIES AND INTEGRATION WITH THE SAP ONTOLOGY

During the past few months of research activity a big effort has been put into finalizing the common ontologies (for both shapes and tools). In the following subsections, we will show our work for the common ontologies and the changes that were necessary for the SAP ontology in order to integrate it with the common ontologies.

4.1 Activities for the Common Ontologies

In the last months DISI, as task leader, has collected information from the partners involved in the Task 1.2.2 in order to propose new knowledge for both the common ontologies with all the necessary metadata. In particular, this task has identified the following classes (with all the necessary and relevant metadata) that now are part of the common shape ontology.

4.1.1 MultiResolutionModel

This class represents a shape model at different levels of details. Each instance in the shape repository is created automatically when a manifold surface mesh is uploaded in the repository. It has the following metadata

- *hasGranularity*
 - **Type:** integer
 - **Meaning:** a measure of granularity of elementary moves for resolution increase and decrease that this multiresolution mesh is able to support, it is the same as the number of tiles (triangles if surface mesh, tetrahedra if volume mesh) involved in a single simplification operation during the construction of this multiresolution mesh. More precisely, if a simplification operation removes a (bigger) set of triangles T_1 and replaces it with another (smaller) set of triangles, the granularity is the cardinality of set T_1 . A smaller granularity means that the multiresolution mesh is able to extract a wider variety of meshes at different resolutions.
- *hasMaximumResolutionMesh*
 - **Type:** relation (pointing either to a surface mesh or to a volume mesh)
 - **Meaning:** link to the original mesh from which this multiresolution mesh has been built through simplification.
- *hasMeshType*
 - **Type:** string in {surface, volume} (alternative: integer in {2, 3})
 - **Meaning:** the type of mesh that is represented by this multiresolution mesh, i.e., the type of mesh from which this multiresolution mesh has been built through simplification
- *hasNumberOfTilesAtMaxRes*
 - **Type:** integer
 - **Meaning:** number of tiles (triangles if surface mesh, tetrahedra if volume mesh) in the mesh at maximum resolution that can be extracted from this multiresolution mesh, it is the same as the number of tiles of the original mesh from which this multiresolution mesh has been built through simplification.
- *hasNumberOfTilesAtMinRes*
 - **Type:** integer
 - **Meaning:** number of tiles (triangles if surface mesh, tetrahedra if volume mesh) in the mesh at minimum resolution that can be extracted from this multiresolution mesh, it is the same as the number of tiles of the final mesh in the simplification process performed for building this multiresolution mesh.
- *hasNumberOfVerticesAtMaxRes*
 - **Type:** integer
 - **Meaning:** number of vertices in the mesh at maximum resolution that can be extracted from this multiresolution mesh, it is the same as the number of vertices of the original mesh from which this multiresolution mesh has been built through simplification.

- *hasNumberOfVerticesAtMinRes*
 - **Type:** integer
 - **Meaning:** number of vertices in the mesh at minimum resolution that can be extracted from this multiresolution mesh, it is the same as the number of vertices of the final mesh in the simplification process performed for building this multiresolution mesh.
- *hasSimplificationOperator*
 - **Type:** string in {edge collapse, vertex removal, triangle collapse, vertex-pair collapse, ...}
 - **Meaning:** the operator used to simplify the original mesh from which this multiresolution mesh has been built
- *maintainsTopology*
 - **Type:** boolean
 - **Meaning:** whether the topology of the original mesh (manifold condition, number of connected components, etc.) was maintained during the simplification process that has built this multiresolution mesh, or not; if yes, then all mesh representations extracted by this multiresolution mesh have the same topology as the original mesh; otherwise, this is not guaranteed. Note: some simplification operators intrinsically maintain topology, while others do not; so this information in some cases may be deduced from the simplification operator.
- *usesSimplificationTool*
 - **Type:** relation pointing to a softwaretool (if it exists)
 - **Meaning:** if the tool exists it is the one in the tool repository which has been used for the simplification

4.1.2 NonManifoldMesh

This class represents all the non-manifold meshes. Recall that, a manifold (with boundary) is a compact and connected subset S of the Euclidean space such that the neighbourhood of each point of S is homeomorphic to an open ball or to an open half-ball. Objects that do not fulfil this property at one or more points, are called non-manifold objects, and if they also contain parts of different dimensionalities, are called non-regular.

- *hasNumberOf1ConnectedComponents*
 - **Type:** integer
 - **Meaning:** a 1-connected component is any maximal submesh composed only of triangles which can be navigated by paths involving only triangles and edges shared by two triangles
- *hasNumberOf2Cycles*
 - **Type:** integer
 - **Meaning:** number of 2-cycles where the 2-cycles in a mesh are maximal components of the mesh bounding voids

- *hasNumberOfCells*
 - **Type:** integer
 - **Meaning:** number of simplices of maximal dimension (typically triangles or tetrahedra)
- *hasNumberOfConnectedSimplexesOfDim1*
 - **Type:** integer
 - **Meaning:** The number of connected components, which are not 1-connected and are composed of simplexes of dimension 1
- *hasNumberOfConnectedSimplexesOfDim2*
 - **Type:** integer
 - **Meaning:** The number of connected components, which are not 1-connected and are composed of simplexes of dimension 2
- *hasNumberOfNonManifoldEdges*
 - **Type:** integer
 - **Meaning:** The number of non-manifold edges, where a non-manifold edge is any edge shared by three or more triangles.
- *hasNumberOfNonManifoldVertices*
 - **Type:** integer
 - **Meaning:** The number of non-manifold vertices, where a non-manifold vertex is a vertex such that its star consists of several connected components.
- *hasNumberOfWireEdges*
 - **Type:** integer
 - **Meaning:** the number of wire edges that are dangling edges not bounding any triangle.
- *hasTopologicalDescription*
 - **Type:** URL
 - **Meaning:** a file providing the description of the different topologically meaningful parts into which a non-manifold mesh can be segmented by the application of *TopMesh* (a tool developed at DISI which is planned to become a web service of the DSW by the end of the project). The mesh is partitioned into the following classes of components (1) Graphs of wire-edges: a graph of wire-edges is any maximal connected component of the mesh consisting only of wire-edges, (2) Maximal 1-connected submeshes of triangles homeomorphic to a 2-manifold with boundaries obtained by splitting the mesh at non-manifold edges also called sheets, (3) 2-cycles also called shells

4.1.3 RasterData (RasterData2D, RasterData3D, RasterData4D, MRI, DynamicMRI)

These classes represent any regularly sampled dataset in 2-3-4 dimensions. For the sake of simplicity we list here the metadata for two of them, namely RasterData2D and

DynamicMRI

4.1.3.1 RasterData

- *hasDimensionX*
 - **Type:** integer
 - **Meaning:** the x dimension of the raster data
- *hasMeasurementType*
 - **Type:** string
 - **Meaning:** the type of raster data. The value can be one of the following Black&White | Greyscale | Indexed | RGB | Height (Depth - Distance - Range - Elevation) | XYZ

4.1.3.2 RasterData2D

- *hypernym:* RasterData
- *hasDimensionY*
 - **Type:** integer
 - **Meaning:** the y dimension

4.1.3.3 RasterData 3D

- *hypernym:* RasterData
- *hasDimensionY*
 - **Type:** integer
 - **Meaning:** the y dimension

4.1.3.4 MRI

- *hypernym:* RasterData3D
- *hasDimensionZ*
 - **Type:** integer
 - **Meaning:** the Z dimension
- *hasAcquisitionDate*
 - **Type:** string
 - **Meaning:** the date of acquisition
- *hasAcquisitionMatrix*
 - **Type:** string
 - **Meaning:** It defines the size of the Matrix n by m. It is in the form [n by m]
- *hasAcquisitionTime*
 - **Type:** string
 - **Meaning:** the time of acquisition

- *hasBitStored*
 - **Type:** integer
 - **Meaning:** the number of bits stored
- *hasEchoTime*
 - **Type:** float
 - **Meaning:** Echo time expressed in milliseconds (ms)
- *hasFlipAngle*
 - **Type:** float
 - **Meaning:** the flip angle expressed in degree
- *hasImageOrientation*
 - **Type:** string
 - **Meaning:** it is a double 3D vector x_1, y_1, z_1 and x_2, y_2, z_2 (in mm). It gives the orientation of the image with respect to the global coordinate system. It should be in the form $[x_1, y_1, z_1]$; $[x_2, y_2, z_2]$
- *hasImagePosition*
 - **Type:** string
 - **Meaning:** (x,y,z in mm - image origin) A 3D vector in the form $[x, y, z]$
- *hasMagneticFieldStrength*
 - **Type:** float
 - **Meaning:** Magnetic Field Strength expressed in Tesla
- *hasMRIAcquisitionType*
 - **Type:** string
 - **Meaning:** Possible values: | 2D | 3D | - 2D: the 3D volume is made of a stack of 2D acquisitions. - 3D: the 3D volume is made of one 3D volume acquisition.
- *hasMRIModality*
 - **Type:** string
 - **Meaning:** the modality of the MRI, it can be a value among the following - | CT | NM | MR | DS | DR | US | OT |
- *hasPatientAge*
 - **Type:** string
 - **Meaning:** the age of the patient of the MRI
- *hasPatientSex*
 - **Type:** string
 - **Meaning:** the sex of the patient
- *hasPatientWeight*
 - **Type:** float
 - **Meaning:** the weight of the patient in Kg

- *hasPixelSpacing*
 - **Type:** float
 - **Meaning:** indicates the pixel spacing in mm
- *hasReceivingCoil*
 - **Type:** string
 - **Meaning:** The coil. Possible values | surface | body |
- *hasRepetitionTime*
 - **Type:** float
 - **Meaning:** it indicates the repetition time in ms
- *hasSliceThickness*
 - **Type:** float
 - **Meaning:** the thickness of the slices in the MRI expressed in mm

4.1.3.5 Movie

- *hypernYm*: RasterData3D
- *hasCodec*
 - **Type:** string
 - **Meaning:** the codec used for the movie. Possible values: | Uncompressed | Indeo | Microsoft Video | mpeg | mpeg 4 | DivX | Cinepak | other
- *hasDimensionT*
 - **Type:** integer
 - **Meaning:** namely the number of frames in the movie
- *hasDuration*
 - **Type:** float
 - **Meaning:** the length of the movie (in sec)
- *hasFrameRate*
 - **Type:** integer
 - **Meaning:** it indicates the number of frames per second
- *hasSound*
 - **Type:** boolean
 - **Meaning:** it indicates if the movie has sound or not
- *hasStreaming*
 - **Type:** boolean
 - **Meaning:** it indicates if the movie has streaming or not
- *hasUncompressedFileSize*
 - **Type:** integer
 - **Meaning:** Optional, it indicates the size of the video file in bytes if saved

as uncompressed video

4.1.3.6 RasterData4D

- *hypernYm*: RasterData
- *no specific metadata*

4.1.3.7 DynamicMRI

- *hypernYm*: RasterData4D
- *hasDimensionZ*
 - **Type**: integer
 - **Meaning**: the Z dimension (the fourth dimension)
- *hasNumberOfAcquisitionPlane*
 - **Type**: integer
 - **Meaning**: It indicates the number of acquisition planes chosen for the session
- *hasNumberOfTemporalPositions*
 - **Type**: integer
 - **Meaning**: semantically equal to the number of frames
- *hasAcquisitionDate*
 - **Type**: string
 - **Meaning**: the date of acquisition
- *hasAcquisitionMatrix*
 - **Type**: string
 - **Meaning**: It defines the size of the Matrix n by m. It is in the form [n by m]
- *hasAcquisitionTime*
 - **Type**: string
 - **Meaning**: the time of acquisition
- *hasAcquisitionFrequency*
 - **Type**: string
 - **Meaning**: the frequency of acquisition
- *hasBitStored*
 - **Type**: integer
 - **Meaning**: the number of bit stored
- *hasEchoTime*
 - **Type**: float
 - **Meaning**: Echo time expressed in milliseconds (ms)

- *hasFlipAngle*
 - **Type:** float
 - **Meaning:** the flip angle expressed in degree
- *hasImageOrientation*
 - **Type:** string
 - **Meaning:** it is a double 3D vector x_1, y_1, z_1 and x_2, y_2, z_2 (in mm). It gives the orientation of the image with respect to the global coordinate system. It should be in the form $[x_1, y_1, z_1]$; $[x_2, y_2, z_2]$
- *hasImagePosition*
 - **Type:** string
 - **Meaning:** (x,y,z in mm - image origin) A 3D vector in the form $[x, y, z]$
- *hasMagneticFieldStrength*
 - **Type:** float
 - **Meaning:** Magnetic Field Strength expressed in Tesla
- *hasMRIAcquisitionType*
 - **Type:** string
 - **Meaning:** Possible values: | 2D | 3D | - 2D: the 3D volume is made of a stack of 2D acquisitions. - 3D: the 3D volume is made of one 3D volume acquisition.
- *hasMRIModality*
 - **Type:** string
 - **Meaning:** the modality of the MRI, it can be a value among the following - | CT | NM | MR | DS | DR | US | OT |
- *hasPatientAge*
 - **Type:** string
 - **Meaning:** the age of the patient of the MRI
- *hasPatientSex*
 - **Type:** string
 - **Meaning:** the sex of the patient
- *hasPatientWeight*
 - **Type:** float
 - **Meaning:** the weight of the patient in Kg
- *hasPixelSpacing*
 - **Type:** float
 - **Meaning:** indicates the pixel spacing in mm
- *hasReceivingCoil*
 - **Type:** string
 - **Meaning:** The coil. Possible values | surface | body |

- *hasRepetitionTime*
 - **Type:** float
 - **Meaning:** it indicates the repetition time in ms
- *hasSliceThickness*
 - **Type:** float
 - **Meaning:** the thickness of the slices in the MRI expressed in mm

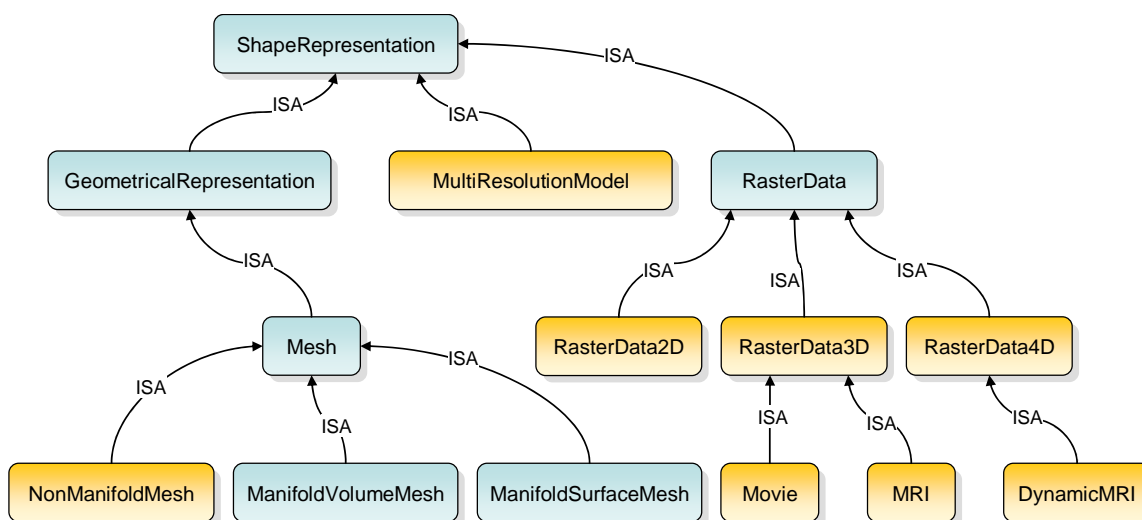


Figure 3 – A zoom on the class *ShapeRepresentation* in the common shape ontology showing (in yellow) those classes that have been proposed by the partners involved in Task 1.2.2 with all the metadata explained before.

Also, a large effort has been put in by all the partners involved in the task in order to validate the existing concepts, proposing in some cases changes and improvements. In Figure 3 we show the structure of the class *ShapeRepresentation* in the common shape ontology after the modifications proposed within the Shape Acquisition and Processing Ontology.

4.2 Integration of the Common Ontologies in the SAP ontology

At the end of the activity described in the previous session, the SAP ontology has been modified in order to be completely integrated with the two common ontologies. So all the concepts in the SAP ontology which had a precise correspondence to concepts in the common shape and/or tool ontology have been removed and the entire structure of the ontology has been updated appropriately.

Here we give some examples of the modifications that have been performed. We use the namespace **sap:** for classes and properties in the SAP ontology, **sco:** for those of the common shape ontology and **cto:** for those of the common tool ontology

- *sap:PersonInfo* class – removed and substituted with *sco:PersonInfo*. Note that all the properties in the SAP ontology which pointed to *sap:PersonInfo* have been modified to point to *sco:PersonInfo*.
- *sap:Institution* class – removed and substituted with *sco:InstitutionInfo*. Note that all the properties in the SAP ontology which pointed to *sap:Institution* have been modified to point to *sco:InstitutionInfo*.

- *sap:Tool* class – removed and *sap:SoftwareTool* class created as sub-class of both *sap:ProcessingSystem* and *cto:SoftwareTool*
- *sap:ToolSession.hasProcessingSystem* property – now has values in *cto:SoftwareTool* (instead of the old *sap:Tool*). Note that, since *sap:SoftwareTool* is a subconcept of *cto:SoftwareTool*, we are able to use as instances for *sap:ToolSession.hasProcessingSystem* also the already existing instances.
- *sap:ShapeData* – has been modified. Since it was basically equal to *sco:ShapeRepresentation* plus the information of the processing session which acquired it, it now has only two properties:
 - *hasBeenAcquiredBy* with values in *sap:ProcessingSession*
 - *hasShapeRepresentation* with values in *sco:ShapeRepresentation*

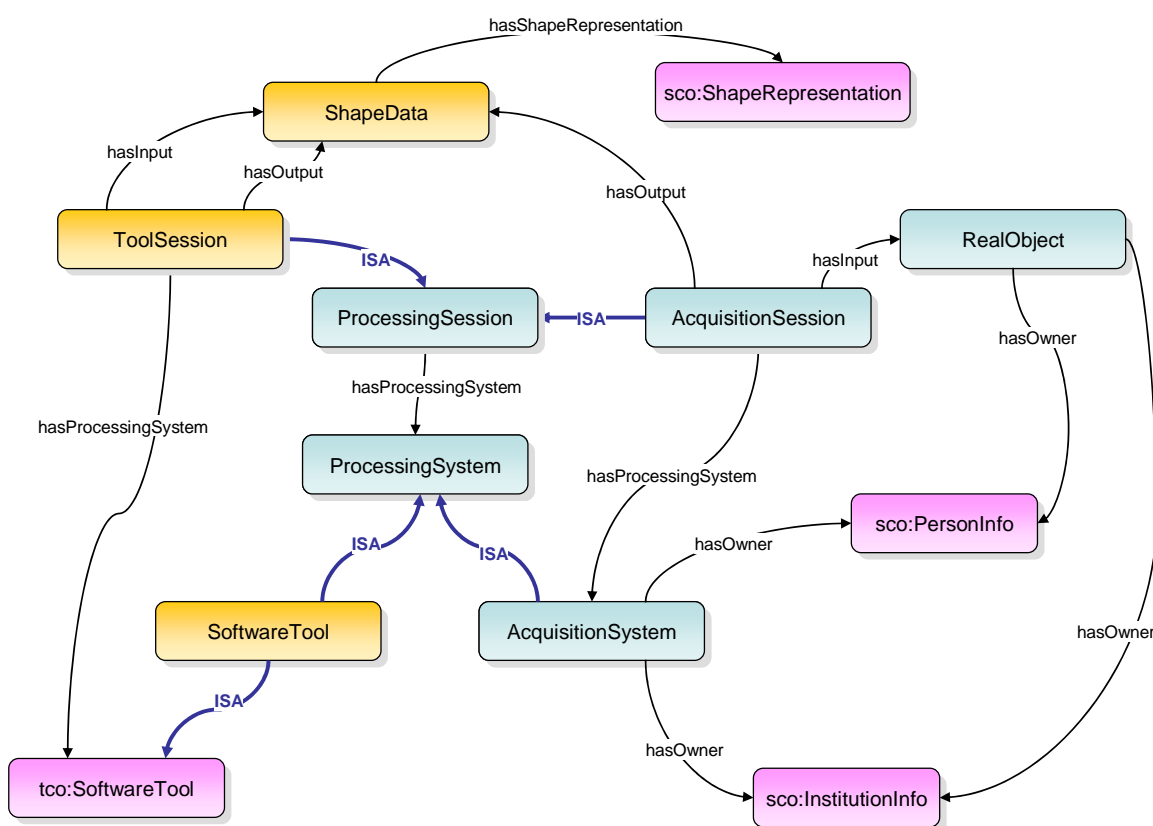


Figure 4 – a Picture showing the classes (in yellow) in the Shape Acquisition and Processing ontology which have mainly been changed by the integration with the common ontologies. In pink the classes from the common ontologies are depicted.

In Figure 4 we show a zoom on the classes in the Shape Acquisition and Processing Ontology which have been modified most by the integration with the Common Shape and Common Tool ontologies. In pink are depicted the classes belonging to the common ontologies and in yellow those classes in the SAP ontology which have been radically modified by the integration. Note that the class *SoftwareTool* in the SAP ontology is a sub-class of both *ProcessingSystem* and *tco:SoftwareTool*. In this way, the class *ToolSession* will be able to link to all the instances in the *tco:SoftwareTool* (basically all the tools in the tool repository) and also to all the instances of the SAP ontology.

5 LIST OF COMPETENCY QUESTIONS RELATED TO THE SAP ONTOLOGY

In this section we present all the formalised CQs. At this stage of development, it is possible to give an answer to some of them (5.1), while for other competency questions (5.2) no answer is possible at the moment. Further details in the specific sections below.

5.1 Competency Questions Answered

We list here a set of representative competency questions we are able to answer at this stage of the ontology development and how the answers can be retrieved.

CQ #1. Which are the Real Objects owned by "UU"?

- a. Select all the *RealObject*(s) which have *hasOwner* as UU

CQ #2. What are the Acquisition Systems able to scan a Real Object which is light absorbent?

- b. For each *AcquisitionSystem*, follow *hasAcquisitionDevice*.
- c. For each *AcquisitionDevice* found, follow *hasAcquisitionDeviceType* and check if *isAbleToScanLightAbsorbentObjects* is true. In this case select the *AcquisitionSystem*

CQ #3. What tricks have been performed in order to scan the Real Object "FROG"?

- d. Take "FROG", follow *hasBeenAcquiredBy*, for each *AcquisitionSessions* found follow *hasTrick*

CQ #4. Under which LightingConditions has the real object "FROG" been scanned?

- e. Take "FROG", follow *hasBeenAcquiredBy*, for each *AcquisitionSessions* found follow *hasEnvironmentConditions* and check the value of *hasLightingConditions*

CQ #5. Is there a Converter tool able to convert an OFF file into a PLY file?

- f. Check if there is any instance of the classes *tco:SoftwareTool* which is related to the instance *tco:ConversionBetweenFileFormatsAlgorithm* through the relation *tco:hasFunctionality*.

CQ #6. Which are the tools that are able to compute the Euclidean distance between two shapes?

- g. Check if there is any instance of the classes *tco:SoftwareTool* which is related to the instance *tco:EuclideanDistanceCompareAlgorithm* through the relation *tco:hasFunctionality*.

CQ #7. Which is the compilation platform for this tool?

- h. Select the instance representing the tool, then browse the relation *tco:hasCompilationPlatform*.

CQ #8. Which format type does this tool support?

- i. Starting from the instance of *tco:SoftwareTool* see the property *tco:supportsFormat*.

CQ #9. What is the History related to the SurfaceMesh "ItalianWorldCup.wrl"?

- j. Find the *ProcessingSession*, let's say PS_{start} , with *ItalianWorldCup.wrl* as output (via the relation *hasOutput*)
- k. Browse the relation *hasHistory* from PS_{start} , relying on the transitivity closure with respect to *hasHistory*, all *ProcessingSession*(s) which have contributed to obtain the shape are retrieved.

CQ #10. Who are the owners of the RealObjects source of the SurfaceMesh "flyingPotamo.ply"?

- l. Find all the *AcquisitionSession* in the history (found as before) of "flyingPotamo.ply"
- m. Browse the relation *hasInput* to find out the *RealObject* source for "flyingPotamo.ply"

CQ #11. What are the possible steps to do starting with a point cloud?

- n. Query *Step* instances which are related to the instance *PointCloud* through *hasInputShapeType*.

CQ #12. Which are the colored images? (rephrased as Which are the RasterData2D which have Color?)

- o. Find all the *RasterData2D* with *hasMeasurementType* as Color or RGB

CQ #13. Which images have dimension M by N? (rephrased as Which are the RasterData2D with dimension X equal to M and dimension Y equal to N?)

- a. Find all the *RasterData2D* with *hasDimensionX* as M and *hasDimensionY* as N

5.2 Competency Questions Not Answered

Here we list in red those CQs that still cannot be answered and in blue those that are still not investigated.

For the CQs not yet answered, note that some of them are strictly related to the analysis of a given digital shape. In the second deliverable D1.2.2.1 we outlined that, looking at some requirements defined by the involved partners (e.g. questions regarding MRI volumes, such as "*which anatomical region of interest (e.g., hip, knee, liver, etc.) is this?*" or "*find more examples of a selected body part*", or "*Is this a healthy organ?*"), it is clear that a formalization of the knowledge related to the **shape analysis** would be useful in order to "*close*" the digital shape lifecycle. We still need to understand how to integrate this knowledge in this cluster or maybe if another cluster (or sub-cluster) could be necessary. We are discussing this with the network partners in order to understand possible interests and to plan future activities.

CQ #14. Through what steps is it possible to reach a surface mesh?

- a. NOT YET ANSWERED – we are working on the extension of the ontology

CQ #15. Through what sequence of steps is it possible to reach a surface mesh starting from a point cloud?

- a. NOT YET ANSWERED – we are working on the extension of the ontology

CQ #16. What other shapes do images with horses contain?

- a. At this level of conceptualization of the common shape ontology we are not able to answer this question. We do not consider it at this moment.

CQ #17. What sub-shapes are contained in a face shape? What sub-shapes are contained in the sub-shapes? Etc.

- a. At this level of conceptualization of the common shape ontology we are not able to answer this question. We do not consider it at this moment.

CQ #18. What images contain elongated objects?

- a. At this level of conceptualization of the common shape ontology we are not able to answer this question. We do not consider it at this moment.

CQ #19. What images contain highly textured areas?

- a. At this level of conceptualization of the common shape ontology we are not able to answer this question. We do not consider it at this moment.

CQ #20. Which anatomical region of interest (e.g., hip, knee, liver, etc.) is this?

- a. At this level of conceptualization of the common shape ontology we are not able to answer this question. We do not consider it at this moment.

CQ #21. Is this a healthy organ?

- a. At this level of conceptualization of the common shape ontology we are not able to answer this question. We do not consider it at this moment.

CQ #22. Does this video contain articulated motion?

- a. At this level of conceptualization of the common shape ontology we are not able to answer this question. We do not consider it at this moment.

CQ #23. What actions are performed in this video?

- a. At this level of conceptualization of the common shape ontology we are not able to answer this question. We do not consider it at this moment.

CQ #24. What is the background (foreground) in this video?

- a. At this level of conceptualization of the common shape ontology we are not able to answer this question. We do not consider it at this moment.

6 DISSEMINATION ACTIVITIES FOR THE SAP ONTOLOGY

During the past months of research and development activities, DISI as task leader and all the involved partners have also worked on the preparation of material showing the results of the SAP Ontology to the industrial and research community.

In particular, material for the upcoming DVD on AIM@SHAPE has been prepared:

- 1) a short presentation of the SAP ontology in HTML
- 2) a long presentation of the SAP ontology in both HTML and PPT format
- 3) a video (both swf and avi format) showing the first usage scenario and its demonstration with the AIM@SHAPE semantic search engine

And also:

- 4) a long presentation in PPT format for the International Summer School of AIM@SHAPE held in Tallinn, Estonia, July 19-25, 2006
- 5) a poster and a presentation showing the main objectives and the results of the SAP ontology for the Interactive Demo Lab, organized at the International Visualization Conference 2006 (Baltimore, USA, October 28 – November 2, 2006).

The following papers have been prepared, submitted and presented in the last year

- 1) R. Albertoni, L. Papaleo, F. Robbiano, M. Spagnuolo, *Towards a Conceptualization for Shape Acquisition and Processing*, 1st Int. Workshop on Shapes and Semantics, Matsushima, JAPAN, June 17, 2006
- 2) L. De Floriani, A. Hui, L. Papaleo, *Topology-Based Reasoning on Non-Manifold Shapes*, 1st Int. Workshop on Shapes and Semantics, Matsushima, JAPAN, June 17, 2006
- 3) B. Falcidieno, M. Spagnuolo, M. Pitikakis, G. Vasilakis, A. Garcia-Rojas, L. Papaleo, *AIM@SHAPE: Research Advances and Future Contributions*, Poster at the 1st Int. Conference on Semantics And digital Media Technology (SAMT), Athens, GREECE, December 6-8, 2006

7 CONCLUDING REMARKS

In this deliverable, we have presented the third version of the SAP ontology. It is intended to produce knowledge support for researchers who have to face the shape lifecycle in general. We have identified that the acquisition of a shape is fundamental in this sense, but also that further processing has to be done to fulfil the users' needs, and documentation has to be produced in order to enrich the semantic impact of the shape.

The first version of the ontology already supported the requirements related to the Acquisition Session. In the second version we have formalized the knowledge related to shapes, tools and tasks by proposing a schema. This third version further improved the integration of the ontology with the common shape and tool ontologies, removed some inconsistencies in the general structure, and validated the second and the third usage scenarios.

Here, we briefly outline the major changes and improvements with respect to the previous versions of the ontology presented in Deliverable D1.2.2.1:

1. **Demonstration, via the answers to the identified CQs, of the three usage scenarios**, "Webby and the scanning session", "Webby and the shape processing", "Webby and the history of a shape".
2. **Identification of the fourth usage scenario "Webby and the workflows"**, which regards processing steps and shape types and aims at making explicit in the shape acquisition and processing ontology interdisciplinary knowledge relative to the creation/modification of shapes.
3. **Validation of the schema identified for formalizing the knowledge related to steps, tasks and tools processing digital shapes**, where we have shown with the formalization of the fourth usage scenario that the ontology needs to be extended and that the concept of "sequence of steps" must be defined. We are actually working in this direction.
4. **Validation and integration of the common shape and tool ontologies**, where the validation of the two common ontologies has been performed and

completed, proposing some changes, improvements and new concepts. Moreover, the integration with these two ontologies has been completed, changing the structure of the SAP ontology accordingly.

We want to remark again that, looking at some requirements defined by the involved partners (e.g. questions regarding MRI volumes, such as "*which anatomical region of interest (e.g., hip, knee, liver etc.) is this?*" or "*find more examples of a selected body part*", or "*Is this a healthy organ?*"), it is clear that a formalization of the knowledge related to the **shape analysis** would be useful in order to "*close*" the digital shape lifecycle. We still need to understand how to integrate this knowledge in this cluster or maybe if another cluster (or sub-cluster) could be necessary. We are discussing this with the network partners in order to understand possible interests and to plan future activities.

Moreover, in order to populate the knowledge base related to the SAP Ontology with meaningful instances, some collaboration sessions are expected, in which access to acquisition facilities will be granted to interested users (possibly also some not belonging to the AIM@SHAPE consortium) that are willing to document their acquisition sessions.