Multiscale Spatiotemporal Visualisation

STREP # FP7-248032

MSV Deliverable

D2.1 – White paper on multiscale visualisation (draft) Addendum

Work package 2: Shared vision

Debora Testi (B3C)

23/12/2011

SEVENTH FRAMEWORK PROGRAMME

DOCUMENT INFORMATION

IST Project Number	FP7-248032	**Acronym**	MSV
Full title	Multiscale Spatiotemporal Visualisation: development of an open-source software library for the interactive visualisation of multiscale data		
Project URL	http://www.msv-project.eu		
Document URL	https://www.biomedtown.org/biomed_town/MSV/associates/reviewers/		
EU Project officer	Ivo Locatelli		

Deliverable Number	2.1	**Title**	White paper on multiscale visualisation
Work package Number	2	**Title**	Shared vision

Date of delivery	**Planned**	31-07-2010	**Actual**	02/01/2012
Status	Version v3		final ☒	
Nature	Prototype ☐ Report ☒ Demonstrator ☐ Other ☐			
Dissemination Level	Public ☐ Consortium ☒			

Authors (Partner)	Debora Testi (B3C)			
Responsible Author	Debora Testi		**Email**	d.testi@scsitaly.com
	Partner	B3C	**Phone**	+39-051-593543

Abstract (for dissemination)	This is an addendum to D2.1 in which the MSV partners have better identified the focus of the MSVTK library giving priorities to the identified challenges.	
Keywords	Shared vision, development focus	

Version Log

Issue Date	**Rev No.**	**Author**	**Change**
18/12/2011	V1	Debora Testi	First draft with all contents
23/12/2011	V2	Debora Testi	Revised version
02/01/2012	V3	Debora Testi	Final version

Project Consortium Information

Disclaimer: This document is property of the MSV Consortium. There is no warranty for the accuracy or completeness of the information, text, graphics, links or other items contained within this material. This document represents the common view of the consortium and does not necessarily reflect the view of the individual partners.

List of Abbreviations

SCS	SCS srl
BED	University of Bedfordshire
UPF	University POmpeu Fraba
KIT	Kitware
AUK	University of Auckland
VTK	Visualitation ToolKit
MAF	Multimod Application Framework

Summary

1 Introduction

During the first technical review meeting held on March 2011, the reviewers' panel asked to the MSV consortium to start identifying and to clarify the "focus" of the MSVTK library. An extract from the reviewers' report is here provided:

"*The FOCUS shall identify a tractable subset of challenges which will be addressed within the MSV-project, and relate this subset to the existing toolkit. Levels of final project achievement shall be identified in terms of additional toolkit functionality specified for example as use cases derived from the data sets identified in D3.1 ... A priority listing of implementation goals shall be given with an appropriate rationale in terms of benefits to end users. Risks shall be noted and appropriate mitigation plans outlined. Each item on the list should be associated with clear verifiable criteria for success.*"

As part of the project activites, the exemplary problems definition has been completed with a number of public data available[1] (see D3.2 for details). The information gathered on the potenial use cases together with the first analysis of the multiscale techniques (see D4.1 for details) have allowed the consortium to prioritarise the challenges, to define the implementation strategy for MSVTK, and to design the first demonstrators.

In the next section, the results of this analysis are reported in terms of priority given to challenges together with risk analysis and mitigation plans for each challenge.

2 Challenges and priorities

The list of challenges, as described in D3.1, has been revised according the data available to the consortium for the project results verification (as detailed in D3.2).

A number of example use cases have been collected. They represent quite a wide range of biomedical domains and, all together, they can be used to test all the identified challenges. This is the summary table mapping the challenges to the datasets available (as from the public wiki).

	Examples																														
Challenges	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	2.1	3.1	3.2	3.3	4.1	4.2	4.3	4.4	4.5	5.1	5.2	5.3	5.4	5.5	6.1	7.1	8.1	8.2	9.1	10.1	11.1	11.2	11.3
Ch1: Different spatial scales	x	x	x	x			x	x	x	x					x			x								x			x	x	x
Ch2: Registration issues	x		x		x				x	x	x	x	x	x	x	x	x	x		x	x								x	x	x
Ch3: Very large data	x		x				x	x	x	x					x	x	x	x					x						x	x	x
Ch4: Gaps between scales	x						x			x								x							x						x
Ch5: Heterogeneous data types	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x		x		x	x	x	x	x
Ch6: Heterogenous dimensionality	x							x	x									x													
Ch7: High dimensionality	x							x	x				x															x			
Ch8: Interactive visualisation									x	x										x			x	x	x				x	x	x
Ch9: Time varying issues		x	x	x				x	x																						x

[1] https://www.biomedtown.org/biomed_town/MSV/reception/wikis/Data

It is evident from the table, that not all the challenges are equally represented by the exemplary data collections. Thus, priority has been given to the challenges according to the number of available examples: addressing with MSVTK a more frequent challenge would give higher benefits to the community.

Based on this rationale, these are the challenges listed in priority order:

1. Heterogeneous data types
2. Registration issues
3. Different spatial scales
4. Very large data
5. Interactive visualisation
6. Gaps between scales
7. Time varying issues
8. High dimensionality
9. Heterogenous dimensionality

For what concerns heterogenous data types, all the end-users frameworks, which will be integrating MSVTK in the course of the project (MAF, GIMIAS, and VTK itself), already provide support to heterogenous data types. For this reason, no specific implementation is foreseen on this challenge within MSVTK, but the demonstrators will take into consideration the linkage between data of heterogenous types.

As already mentioned in previous documents, the registration problem is very dependent on the type of data to be registered and to the purpose of the registration itself. As MSVTK is meant to be a general-purpose library, the implementation of specific registration methods is out of the scope of the project. Manual registration techniques will be provided in the demonstrators to deal with use cases data which are not already registered in space.

Following these considerations, the first priorities to be addressed by MSVTK will be the spatial multiscale issue and the large data object management.

3 MSV strategy and implementation

The MSVTK will implement the click and zoom interaction paradigm: visual cues are provided for the positions of lower scale data with respect to the whole data, which leads to an intuitive interface for data navigation. MSVTK will use placeholders not only for the representation of lower-scale data, but also for hyperlinks to provide extra information such as documentation, etc. At the same time, it will also investigate the possibility of conveying meaningful information about the represented data through the icon/placeholder shapes and colors, with the aim to optimize information transfer and user's experience.

The MSVTK library is being implemented as an extension of VTK (Visualization ToolKit). The implementation has been designed to be as general as possible, and the library will provide a series of software elements, like widgets, that might be used later on in other software development projects to add support to the multiscale visualisation.

In particular, MSVTK relies and extends *vtkButtons*, which will be used to provide the visual cue to the user on the data available also at lower scales, and for the data navigation. A "button" is a geometry with a rectangular region that can be textured.

The button is divided into two regions: the texture region and the shoulder region. The points in both regions are assigned texture coordinates. The texture region has texture coordinates consistent with the image to be placed on it. All points in the shoulder regions are assigned a texture coordinate specified by the user. In this way the shoulder region can be colored by the texture.

As previously mentioned, at all scales there may be large datasets (size not possible to be dealt with on the available hardware), running into many GBs, so a proper management of these data types during the interaction has to be included. The MSV project will rely on previous work carried out to manage out-of-core data using a multi-resolution and bricking approach[2]. The possibility to apply the same technique to different data types is being investigated, together with its effective integration with the zoom-based navigation approach.

Also extensions to basic VTK functionalities to better deal with the time-varying data and/or information are being added.

4 Risks

In this section, for each challenge, the implementation choice is reported together with the associated risk and the identified mitigation plan.

Challenges	Implementation choice	Risk	Mitigation plan
Ch5: Heterogeneous data types	Use of vtkButtons and QtWidgets to convey linkages between heterogenous data types.	None	All the end-users frameworks, which will be integrating MSVTK in the course of the project (MAF, Slicer, GIMIAS) and VTK itself, already provide support to heterogenous data types. For this reason, no specific implementation is foreseen on this challenge within MSVTK, but the linkage among heterogenous data will be addressed into the MSV demonstrators.
Ch2: Registration issues	Manual registration tool.	None	The manual method, even if not very accurate, is general and does not rely on any specific set of data. Thus no problems are expected with this aspect. Specific registration methods will be then possibly added at end-user applications level by the system integrating MSVTK.

[2] A. Agrawal, J. Kohout, G. Clapworthy, N. McFarlane, F. Dong, M. Viceconti, F. Taddei and D. Testi (2010) Enabling the interactive display of large medical volume datasets by multiresolution bricking. *Journal of Supercomputing* 51(1):3-19

Ch1: Different spatial scales	Extension of vtkButtons to provide the identification of smaller/larger scale objects and implement the click and zoom interaction.	Moderate	In case problems will arise, the use of standard static glyph or icons to represent smaller scale objects will be considered.
Ch3: Very large data	Implementation of multi-resolution bricking.	Moderate	The proposed method was already included and tested in a previous MAF version; thus no particular risks are foreseen. The aim will be to increase its generality and possibily the number of supported data.
Ch8: Interactive visualisation	No specific implementation planned (vtkButtons use will be tested).	Low	All the end-users frameworks which will be integrating MSVTK in the course of the project (MAF, Slicer, GIMIAS) have already available interactive methods for visualisation. However, there might be cases in which the interaction is very slow due to hardware limits; in a specific demonstrator the use of vtkButtons to manage low and high resolution data will be tested.
Ch4: Gaps between scales	The same the different spatial scales.	Moderate	The use of the vtkButtons should allow the MSVTK library to work also in cases with gaps between scales.
Ch9: Time varying issues	Temporal zoom.	High	As very few datasets showing time multiscale issues are available, it might be that MSVTK would not be general enough to cover all possible end-users problems. Revision of the implementation can be carried out when other data examples are available.
Ch7: High dimensionality	Specific visualisation techniques.	High	As very few datasets showing high dimensionality issues are available, it might be that MSVTK would not cover this specific aspect. Further implementation analysis might be carried out when other data examples are available.
Ch6: Heterogenous dimensionality	No specific implementation planned.	Low	As point on the heterogenous data types.

5 Demonstrators

The MSVTK components are being used in the development of prototypes, which allow checking, on the collected exemplary problems, the efficacy of the proposed approach. Moreover, in order to verify the implementation generality, different prototypes applications are being developed integrating the MSVTK library tools in other frameworks like the Multimod Application Framework (MAF), GIMIAS, and VTK itself.

This is a first list of demonstrators the consortium is working on:

1. vtkButtons draft implementation: Aim is to visualise multiple datasets in which the use the vtkbuttons draft implementation within a MAF-based application is tested. The prototype will first re-implement the LHDL prototype trying to overcome its limitations, and then will be verified to other domains data. The user will be allowed to navigate 3D images datasets (CT scan at organ level, microCT scan, and nanoscan).
2. Large dataset support management: Original BED code is being generalised and integrated in MAF for the management of out-of-core volumes.
3. Electro-physiological dataset: An application designed using Qt, CTK and VTK will be dealing with a cardiological data example with the heterogeneous data type, sparse data, and time scale issues. Sparse points on the 3D heart surface are registered to ECG data, which are varying over a certain time frame. The 3D points are displayed in a VTK render window. The ECG(s) in a separate VTK chart view(s). Time is represented differently depending on the data.
4. Fiber multiscale visualisation of the myocardium: This example focuses on the challenge concerning integration of information in different spatial scales with a big gap between the two spatial scales.
5. Human anatomy interaction: The resolution and the large number of components with a lot of points and cells makes the standard rendering and the interaction very slow. The main MSVTK idea is to dynamically load, use, and manage different resolutions when interacting and rendering the scene and components. In this last case, the issue is not related to the nature of the data, but from the hardware limits. It is challenging to load all the data at once due to memory limits, and have an interactive frame rate (limited by the CPU). MSVTK will allow loading a low resolution of the entire anatomy by default, and then have every component (vtkButton) dynamic and clickable to increase/decrease its resolution.

The technical aspects of the demonstrators will be revised and detailed during their implementation. More details on the MSVTK general architecture will be provided in D5.1.