# High resolution visualization library for Exa-scale supercomputer

Wada, Yoshitaka
Department of Mechanical Engineering, Faculty of Science and Technology, Kindai University : Associate Professor

Murotani, Kohei
Department of Systems Innovations, University of Tokyo : Assistant Professor

Ogino, Masao
High Performance Division, Information Technology Center, Nagoya University : Associate Professor

Kawai, Hiroshi
Department of Computer and Media Engineering, Faculty of Engineering, Tokyo University of Science : Associate Professor

他

https://hdl.handle.net/2324/1546883

出版情報：MI lecture note series. 64, pp.71-79, 2015-09-18. 九州大学マス・フォア・インダストリ研究所
バージョン：
権利関係：

MATHEMATICAL PROGRESS IN EXPRESSIVE IMAGE SYNTHESIS 2015
**Sep 25-27, 2015, Nishijin Plaza Kyushu University, Fukuoka**

# High resolution visualization library for Exa-scale supercomputer

Yoshitaka Wada*
Department of Mechanical Engineering, Kindai University
Kohei Murotani†
Department of Systems Innovations, University of Tokyo
Masao Ogino‡
High Performance Division, Information Technology Center, Nagoya University
Hiroshi Kawai§
Department of Computer and Media Engineering, Tokyo University of Science, Suwa
Ryuji Shioya¶
Faculty of Information Sciences and Arts, Toyo University

**Abstract** This paper describes development of visualization library, which is named as LexADV_VSCG, for very large scale on the next generation supercomputer. At this moment, the next generation supercomputer, which is called as exa-scale computer, is not designed clearly. We predict the exa-scale computer and research the possibility of new software which is optimized for the exa-scale computer. In this work, we show how to visualize and deal with ultra large scale FE and particle based data with very fine resolution. LexADV_VSCG just provide only simple API for drawing and rendering triangles and lines including transparent and solid colors.

**Keywords:** Exa-scale computing, parallel finite element method, parallel particle-based method, ultra high resolution

## 1 Introduction

The large scale FE analysis would be bottleneck using supercomputer systems[1-4]. However a commodity computer system becomes popular and faster than ever and graphic accelerator becomes 20 times faster than 10 years before at moderate estimation. The Kei supercomputer can generate numerical computation result of over tera bytes. Even if using data compression technique, network bandwidth is not enough for data transmission through the Internet. In the next generation supercomputer, the data transmission problem will be obvious and we will not be able to handle post processing, which is an examination task using visualization software system after simulation, on outside computers. There are two ways to solve the problem. One is the data compression which

*wada@mech.kindai.ac.jp, 3-4-1 Kowakae, Higashiosaka, Osaka, Japan
†muro@sys.t.u-tokyo.ac.jp, 7-3-1 Hongo Bunkyo, Tokyo, Japan
‡masao.ogino@cc.nagoya-u.ac.jp, Furo-cho, Chikusa, Nagoya, Japan
§kawai@rs.tus.ac.jp, 5000-1 Toyohira, Chino, Nagano, Japan
¶shioya@toyo.jp, 2100 Kujirai, Kawagoe, Saitama, Japan

is reduced data set composed by coarse volume data or coarse facet data using parallel computer environment[5-8]. The data set becomes much smaller than the original data and we can handle the post processing with interactivity. The other is the direct visualization on the supercomputer which is sequential post processing after numerical simulation[9-11]. However computer visualization as a post processing in the finite element analysis provides important user experience. From a point of this view, visualization on the computer could not provide experience well in the past. In addition, the architecture of the next generation supercomputer is just determined. In the background, the authors have developed scientific visualization library: LexADV_VSCG[9,12-14] with high portability for any computer environment. In this study, several results are shown and a design and an actual implementation for the library through the visualized results are discussed.

# 2 LexADV_VSCG library

## 2.1 Fundamental ideas of design and implementation

Fundamental ideas of LexADV VSCG are as follows[3].

- Simple implementation for any environment: independent from OS
- Multiple images to be one image using z-buffer: parallel processing
- No other special hardware and libraries: independent from specific environment issue
- Handling very fine resolution by well-designed software: key idea for ultra large scale data
- Impressive image by well-designed algorithm and implementation: key idea using very fine resolution

These ideas are important keys for high reliability on the future supercomputer. As mentioned above, we don't know the actual next generation supercomputer very well. In order to ensure those ideas, we have decided rendering processes conducted by software without any other software library. LexADV_VSCG is a simple library software which provides application programming interface for scientific visualization. In short, LexADV_VSCG is usually integrated into a simulation code and doesn't use any specific hardware accelerators and software libraries which are developed by other research group and companies.
Image data structure has additional several bits to represent depth information and dot product by surface normal and view direction. The structure possess extended z-buffer information. The advantages of the extended z-buffer are suited for parallel computers on which the images are rendered in parallel. The images are usually stored at the local node storage or stay in local memory. For example, if a supercomputer has totally 50,000 cores, FE mesh also is subdivided into 50,000 parts. View camera is fixed at 200 locations, 10,000,000 images are rendered at least. We should merge all of these images into 200 images in this case. As a result, the merging process needs huge data transmission between nodes. Merging tasks are sequentially processed as shown in Fig. 1. If the tasks are processed for 16 times, $2^{16}$ images can be merged into one.
Visualization technique of LOD (Level Of Detail) is a practical idea to deal with a very fine FE mesh. Simplicity of handling the visualized results is important. LOD requires several level of data set for one analysis model. In case of massively parallel data, we have to manage hundreds of thousands of data sets. Furthermore, very large scale FE data can be generated on only a supercomputer. If computation from modeling to visualization is conducted on a supercomputer, computing cost would be larger than we expected at first. In order to overcome these difficulties, we have decided to generate $10^5 \times 10^5$ pixels image in parallel and we just merge hundreds of thousands of images into one on the supercomputer.

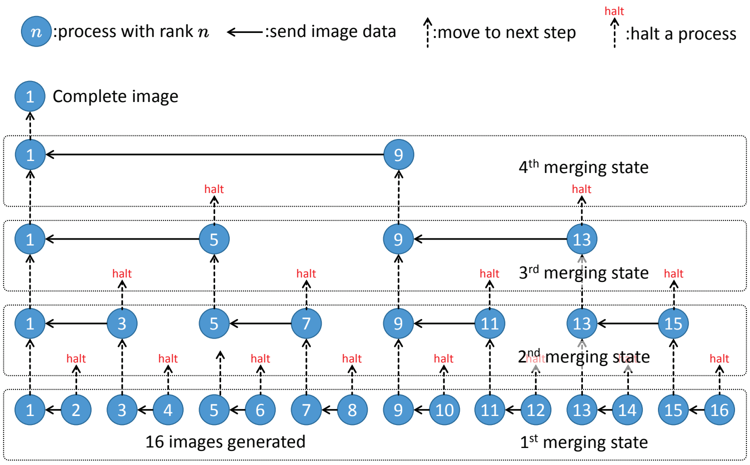



Figure 1: Image merging process in parallel environment.

## 2.2 Implementation and functions

LexADV_VSCG is implemented in C language which is compatible with C99 standard. Any kind of compiler on supercomputers usually doesn't provide the latest standard, because reliability and performance are required to compilers. Since a reliability is strongly demanded in scientific visualization, LexADV_VSCG needs no other library except for standard C library defined in C99 standard. Functions in the library are shown in table 1. The library provides very fundamental functions to draw and render triangles with gradated colors in accordance with specified physical quantity, i.e. von Mieses stress $\overline{\sigma}$ and principal stresses $\sigma_1, \sigma_2$ and $\sigma_3$ and so on. One of the target application of the library is particle based method, i.e. Moving Particle Semi-implicit (MPS) and Smoothed-Particle Hydrodynamics (SPH). A sphere is represented by a polyhedron which consists of 80 or 320 triangles. An example of particle visualization is shown in Fig.2. Total numbers of polygons affect total time to get an image, however it is important for users to choose fine or coarse visualization result. In the very fine image, differences between fine and coarse polyhedron do not make large difference of both of images.

# 3 How to get interactivity using the library

In the development of LexADV_VSCG library, the objective is clearly limited to applications to very large scale fine mesh and huge number of particles. The library doesn't provide such usual techniques for high quality rendering image, which are ray casting and glow shading. As a result of these limitation, the implementation goes back to fundamental techniques by drawing triangle facet and shading on the flat triangle. However, huge number of fine facet and particle which represented by triangle facets, the quality of the image rendered by the library is enough for scientific visualization with the supercomputer.

Interactivity of visualization is one of the most important issue in the scientific post processing. Regretfully the latest supercomputer doesn't provide any interactive connection with user's computer. All of programs are controlled by job controlling system on the supercomputer. LexADV_VSCG library is able to generate very fine image with $10^5 \times 10^5$ pixels resolution. The image has enough information with regards to fine FE mesh. Usual image viewer can easily handle the large image by

Table 1: Implemented fundamental functions for scientific visualization

| | Functions | Explanation |
|---|---|---|
| vscg_draw_triangle functions | draw_triangle_line | draw triangle line with same color and have no depth info |
| | draw_triangle_solid | draw solid triangle with same color and have no depth info |
| | draw_triangle_line_with_depth (z-buffer) | draw triangle line with color and have depth info |
| | draw_triangle_solid_with_depth (z-buffer) | draw solid triangle with same color and have depth info |
| | draw_triangle_gradated_with_depth | draw solid triangle with gradated color and have depth info |
| | draw_triangle_gradated_transparent | draw transparent triangle with gradated color and have no depth info |
| vscg_XXXX utility functions | allocate_image | allocate image buffer for drawing |
| | set_color_XXXX (setting default color) | generate colors for basic colors and color legend |
| | vector_XXXX (vector operations) | 2 and 3 dimensional vector operations |
| | accumrate_image (image operations) | merge two or more z-buffer images into one image |

| | Functions | Explanation |
|---|---|---|
| vscg_draw_particle functions | draw_particle_solid (regular icosahedron) | draw regular icosahedron as a sphere with depth info |
| | draw_particle_transparent_solid (regular icosahedron) | draw transparent regular icosahedron as a sphere with depth info |
| | – | – |
| vscg_XXXX utility functions | draw_vector_with_arrow | draw vector using arrow shape with depth info |
| | paint_color_legend | paint color legend for physical quantity distribution |
| | paint_color_legend_2_physical_quantities | paint color legend for 2 physical quantity distributions |

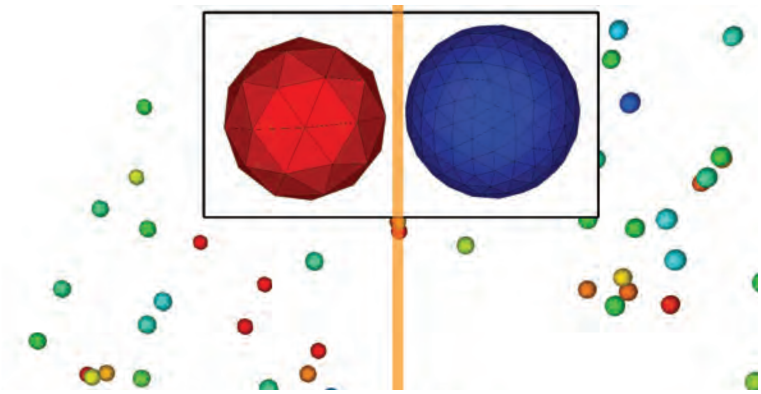


Figure 2: Particle representation by a polyhedron: coarse polyhedron in left side and fine polyhedron in right side.

shrinking to appropriate size as an engineer needs. We require interactivity of rotation of an object too. The system provides multiple viewpoints image generation as shown in Fig. 3. Simple spherical polar coordinate system and the description rule of the file name including the coordinate system are defined as described in Fig. 4. The rule is easily extended to other coordinate system, i. e. Euler's angle definition. There are two ways to describe rotation of an object in the library. First way

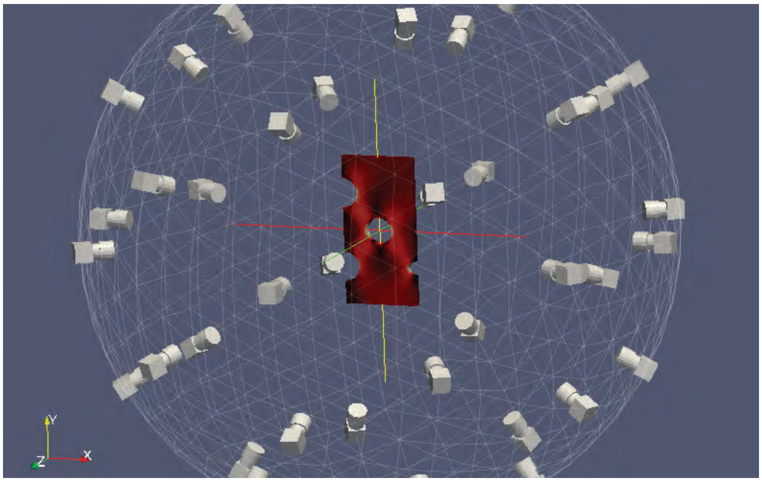



Figure 3: Multiple view points for off-line interactive visualization.

`{name}_{seq. num.}_{θ in deg.}_{φ in deg.}_{time step}.png`

- Name: arbitrary numbers and alphabets within 256 letters
- Sequential number: distinguishing analysis case by 8 digits
- $\theta$ and $\varphi$ in degree: angle in degree , i.e. 01050 = 10.50°
- Time step: time for dynamic analysis by 6 digits

Example:
`MechanicalPart_00000001_01050_00050_000000.png`

Figure 4: Notation rule for camera position in spherical surface.

directly describes rotation matrix. Second way is Rodrigues vector $(k_1, k_2, k_3)$. Rodrigues' rotation formula is described as follows

$$\mathbf{K} = \begin{pmatrix} 0 & -k_3 & k_2 \\ k_3 & 0 & -k_1 \\ -k_2 & k_1 & 0 \end{pmatrix} \tag{1}$$

$$\mathbf{R} = \mathbf{I} + (\sin\theta)\mathbf{K} + (\mathbf{1} - \cos\theta)\mathbf{K^2} \tag{2}$$

where $\theta$ is length of Rodrigues vector. Camera position is notated in spherical polar coordinate. The distance from the center of an object to camera is constant, which is 1. Amplitude $\theta$ is an angle between z-axis and radius vector. Amplitude $\varphi$ is an angle between x-axis and projected vector of radius vector to x-y plane. In actual implementation, the camera is fixed at $(0, 0, -1)$. In order to describe the notation, camera positions should be calculated by

$$\left\{ \begin{array}{c} x \\ y \\ z \end{array} \right\} = \mathbf{R^{-1}} \left\{ \begin{array}{c} 0 \\ 0 \\ -1 \end{array} \right\} \tag{3}$$

$$\left\{ \begin{array}{l} \cos\theta = \dfrac{z}{\sqrt{x^2+y^2+z^2}}(0 \leq \theta \leq \pi) \\ \cos\varphi = \dfrac{x}{\sqrt{x^2+y^2}}, \sin\varphi = \dfrac{y}{\sqrt{x^2+y^2}}(0 \leq \varphi \leq 2\pi) \end{array} \right. \tag{4}$$

where x, y and z are camera position in orthogonal coordinate system; $\theta$ and $\varphi$ are camera position in spherical polar coordinate system.

# 4 Results in numerical examples

## 4.1 Application to finite element analysis in structural problem

Visualization techniques are well designed and developed using GPGPU hardware and those can present impressive artificial images. However, usual design process and analysis process doesn't demand for such an impressive image but accurate and easy to understand. From a practical point of view, the library supports polygon with flat shading including alpha blending which represent transparency effect.

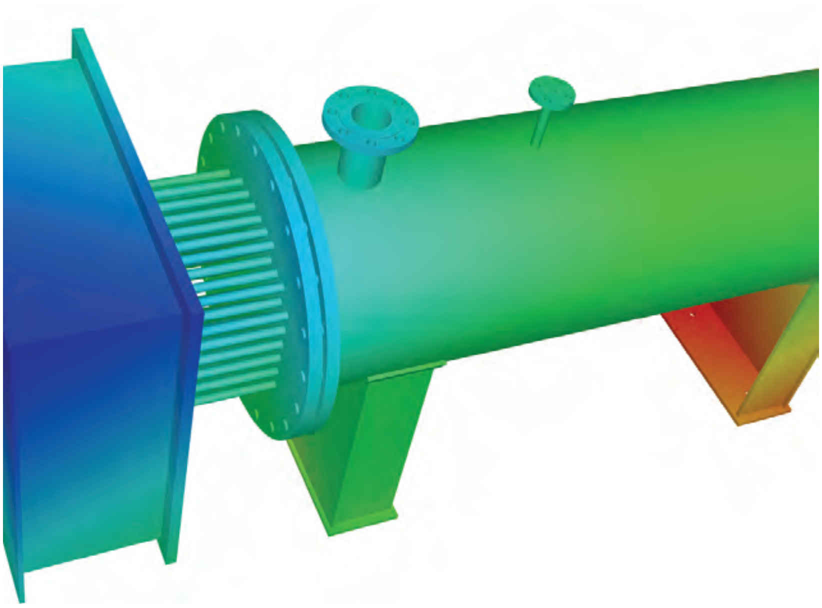
Figure 5 shows an example of high resolution FE analysis image, which is resized image from $10^5 \times 10^5$ pixels. In the image, black lines with 1 pixel width are drawn for representing finite elements, however no element lines cannot be seen in Fig. 5 (a). The $10^5 \times 10^5$ pixels image shows element lines as shown in Fig. 5 (b). Figure 5 (b) is magnified image from the $10^5 \times 10^5$ pixels image. Gradated color is very smoothly distributed in spite of linear interpolation of gradated color. Engineers want to see finite elements even if a very large scale mesh. Since manipulations of 2-dimensional image is processed very fast, magnification and reduction processes are done for a short time.

## 4.2 Application to particle based method in fluid dynamics

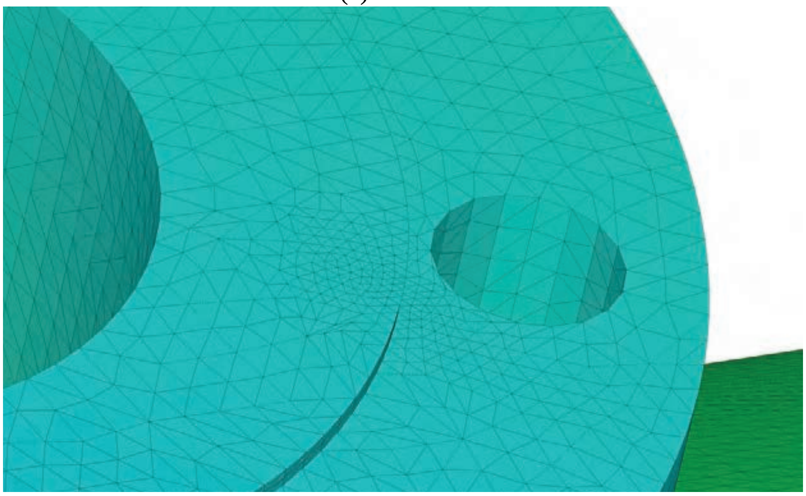
Figure 6 shows another example of very high resolution analysis by explicit MPS method. The visualization is conducted using transparent object and solid colored particles. Figure 6(a) shows 16,000 $\times$ 12,800 pixels image which is enough resolution to examine fluid flow velocity and vorticity. Figure 6(b) is magnified image of velocity from the original image. A large number of particles forms the 3-dimensional configuration of fluid surface in Fig. 6(b), since the sufficient resolution of images keeps shadow of each particle and automatically makes the surface formed.

# 5 Conclusions

We have developed library software to visualize scientific computation results on the next generation supercomputer. LexADV_VSCG generates very fine images on the parallel environment efficiently. Interactivity is one of the important issue for examination of analysis result. We have also proposed interactive operation way by simple representation of spherical polar coordinate system. In particular structural analysis engineers usually examine the result using interactive visualization software.

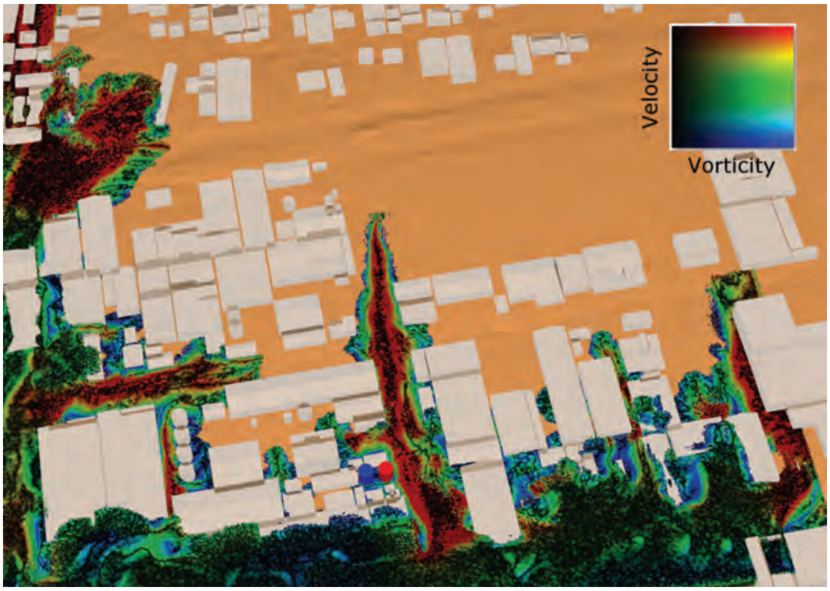


(a) whole view

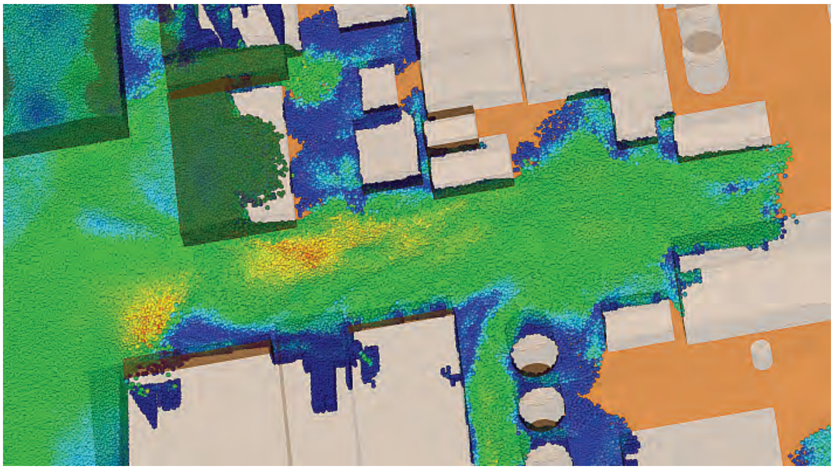


(b) magnified view

Figure 5: Visualized results for finite element model with $10^5 \times 10^5$ pixel resolution: (a) whole view, (b) magnified view.




(a) whole view of velocity and vorticity



(b) magnified view of velocity field

Figure 6: Visualized results for MPS analysis with $30,000 \times 20,000$ pixels resolution: (a) whole view of velocity and vorticity, (b) magnified view of velocity field at the same time step.

Basic visualization techniques are already implemented in the library. Huge amount of data, cannot be downloaded through network, is computed by supercomputers. Automated process to obtain the best visualization results is required for the next generation visualization software. The library just provides fundamental functions and reliability. The library is now developed in progress and is evaluated for more modifications to ensure reliability.

# Acknowledgment

This research was supported in the Basic Research Programs: Development of System Software Technologies for post-Peta Scale High Performance Computing by CREST, JST.

# References

[1] Nakamura, H., Fujishiro, I. and Takeshima, Y., 2000, Towards optimizing local feature metric for simplifying colored interval volumes, *In Proceedings of Work in Progress, IEEE Visualization 2000*, Salt Lake City, (2000)

[2] Fujishiro, I., et al, 2002, Parallel visualization of gigabyte datasets in GeoFEM, *Concurrency and Computation: Practice and Experience*, Vo.14 (2002)

[3] Okuda, H., et al, Parallel Finite Element Analysis Platform for the Earth Simulator: GeoFEM, *Proceedings of 3rd International Working Group Meeting*, (2003)

[4] J. Dongarra, et al, The International Exascale Software Project roadmap, *Int. J. High Performance Computing Applications*, pages 3-60 (2011)

[5] K. Ma, et al, In-situ processing and visualization for ultrascale simulations, *J. Phys.: Conf. Ser. Vol.78, No.012043*, 10 pages (2007)

[6] K. Ma, In Situ Visualization at Extreme Scale: Challenges and Opportunities, *Computer Graphics and Applications, IEEE*, Vol.29, Issue 6, pages 14-19 (2009)

[7] N. Fabian, K. Moreland, The paraview coprocessing library: a scalable, general purpose in situ visualization library, *IEEE Symposium on Large Data Analysis and Visualization, LDAV, IEEE*, pages 89-96 (2011)

[8] A. Artigues, et al, SCIENTIFIC BIG DATA VISUALIZATION: A COUPLED TOOLS APPROACH, *Int. J. Supercomputing frontiers and innovations*, Vol.1, No.3, pages 4-18 (2014)

[9] Y. Wada, et al, Development of high resolution visualization library for very large scale analysis, *Proceedings of 18th JSCES conference*, Vol.18 (2013)

[10] A. Kageyama, T. Yamada, An approach to exascale visualization: Interactive viewing of in-situ visualization, *Computer Physics Communications*, Vol. 185, Issue 1, pages 79-85 (2014)

[11] J. Ahrens, et al, An image-based approach to extreme scale in situ visualization and analysis *SC '14 Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 424-434 (2014)

[12] Y. Wada, et al, Development of high resolution visualization library for very large scale analysis, *Proceedings of JSME-KSME Joint Symposium on CM & CAE 2014*, (2014)

[13] Y. Wada, et al, Advanced high resolution visualization library VSCG for very large scale analysis, *Proceedings of 19th JSCES conference*, Vol.19 (2014)

[14] Y. Wada, et al, Implementation of polygon-based section generation for high resolution visualization library LexADV_VSCG, *Proceedings of 20th JSCES conference*, Vol.20 (2015)