UCRL-TR-209775

LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

ViSUS: Visualization Streams for Ultimate Scalability

Valerio Pascucci

February 16, 2005

This work was performed under the auspices of the U.S. Department of Energy by University of California, Lawrence Livermore National Laboratory under Contract W-7405-Eng-48.

FY04 LDRD Final Report

ViSUS: Visualization Streams for Ultimate Scalability
LDRD Project Tracking Code: 02-ERI-003
Valerio Pascucci, Principal Investigator

Figure 1: Live demonstration of the ViSUS streaming and visualization infrastructure used to show the real-time monitoring of Miranda (in the figure) and Raptor simulation codes executed and visualized in real-time on Blue Gene/L (behind on the left). Picture taken at the IBM booth of the Supercomputing 2004 exhibition floor the day that Blue Gene/L was presented as the fasted supercomputer in the world.

Abstract

In this project we developed a suite of progressive visualization algorithms and a data-streaming infrastructure that enable interactive exploration of scientific datasets of unprecedented size. The methodology aims to globally optimize the data flow in a pipeline of processing modules. Each module reads a multi-resolution representation of the input while producing a multi-resolution representation of the output. The use of multi-resolution representations provides the necessary flexibility to trade speed for accuracy in the visualization process. Maximum coherency and minimum delay in the data-flow is achieved by extensive use of progressive algorithms that continuously map local geometric updates of the input stream into immediate updates of the output stream.

We implemented a prototype software infrastructure that demonstrated the flexibility and scalability of this approach by allowing large data visualization on single desktop computers, on PC clusters, and on heterogeneous computing resources distributed over a wide area network.

When processing terabytes of scientific data, we have achieved an effective increase in visualization performance of several orders of magnitude in two major settings: (i) interactive visualization on desktop workstations of large datasets that cannot be stored locally; (ii) real-time monitoring of a large scientific simulation with negligible impact on the computing resources available.

The ViSUS streaming infrastructure enabled the real-time execution and visualization (see Figure 1) of the two LLNL simulation codes (Miranda and Raptor) run at Supercomputing 2004 on Blue Gene/L at its presentation as the fastest supercomputer in the world. In addition to SC04, we have run live demonstrations at the IEEE VIS conference and at invited talks at the DOE MICS office, DOE computer graphics forum, UC Riverside, and the University of Maryland. In all cases we have shown the capability to stream and visualize interactively data stored remotely at the San Diego Supercomputing Center or monitor in real-time simulation codes executed on a cluster of PC's at LLNL.

1. Background and Motivation

The recent growth in size of large simulation datasets still surpasses the combined advances in hardware infrastructure and processing algorithms for scientific visualization. Only a few years ago the NIH's Visible Human project produced one of the largest datasets, a single mesh of 2048x1216x5871 elements (15 GB). Today we are already dealing with datasets like the Richtmyer-Meshkov instability study [M99] that computed 27000 time-steps for a 2048x2048x1920 grid (217 TB generated in one week of computation on the ASCI Blue Pacific supercomputer). The cost of storing and visualizing such a dataset is prohibitive, so that only one out of every hundred time-steps has been actually stored and visualized. By 2004 ASCI will enable physics simulations on massively parallel computers (100 TeraFLOPS computers) generating upwards of 10 TeraBytes per hour with a potential total output of several petabytes per simulation.

A major consequence of this trend is the reduced productivity of scientists due to the bottleneck posed by the visualization stage in the overall modeling-simulation-analysis activity. Scientific visualization remains the most effective way for a scientist to gain qualitative understanding of the results of a simulation. Unfortunately, it is increasingly more difficult to visualize the results of a simulation interactively, especially as a daily routine from a desktop workstation. This problem poses a fundamentally new challenge both to the development of visualization algorithms and to the design of visualization systems.

The efficiency of a visualization algorithm must be evaluated in the context of an end-to-end system instead of being optimized individually. Moreover, one cannot rely on fundamental assumptions like uniform memory access time (RAM computational model) because the datasets are too large to be kept in main memory or even stored on a local disk. At the system level one has to design the visualization process as a pipeline of modules that process the data in stages. This flow of data needs to be optimized globally with respect to the magnitude and location of available resources.

To address these issues we propose to develop new data-streaming techniques for progressive processing and visualization of large scientific datasets. The progressive processing paradigm combines on-line computing with the use of hierarchical representations. Our strategy is to exploit the coupling of time-critical algorithms and progressive multi-resolution data structures to realize an end-to-end optimized flow of data from the original source, such as remote storage or large scientific simulation, to the rendering hardware.

The classical quality-driven visualization will be complemented by a new time-driven interaction paradigm where an immediate coarse representation of a scene can be displayed to the user while evolving quickly into a high-resolution visualization. This progressive evolution of the visualization can be terminated at any point, if so desired. The scientist is guaranteed real-time interactivity in the navigation of large datasets and can make informed decisions as soon as the required quality is reached.

The implementation of this approach will enable two major visualization modalities that state-of-the-art systems are currently unable to support: (i) interactive visualization on desktop workstations of large datasets that cannot be stored locally; and (ii) real-time monitoring of the progress of a large scientific simulation from a remote workstation. These modalities target multiple phases in the process of generating and exploring very large simulation datasets where real-time user interaction can increase the productivity of scientists.

2. Proposed Work and Technical Approach

2.1. Progressive Computing

In recent years there has been a wealth of research in the area of multi-resolution geometric data structures. Such data structures provide considerable flexibility in trading accuracy for speed by representing a dataset as a virtually continuous range of approximations, from which a model of suitable complexity can be quickly extracted and displayed. For example Lindstrom [LK96] and Duchaineau [DM97] studied view-dependent simplifications of triangulated terrain surfaces. They locally adapt the approximation of the surface depending on its distance from the viewer. Taking advantage of the loss in perceived detail caused by foreshortening, they obtain surfaces with minimal triangle counts while maintaining the appearance of the original model.
Depending on the particular algorithm implemented (surface decimation, isosurface extraction, volume rendering, etc.), different multi-resolution models may be better suited for achieving optimal performance. This creates a problem because the practical benefits of using multi-resolution techniques can be diminished by the numerous data conversion stages involved in the computation pipeline. Figure 2 shows the simple case of isosurface visualization of a scalar field, involving three computing stages.

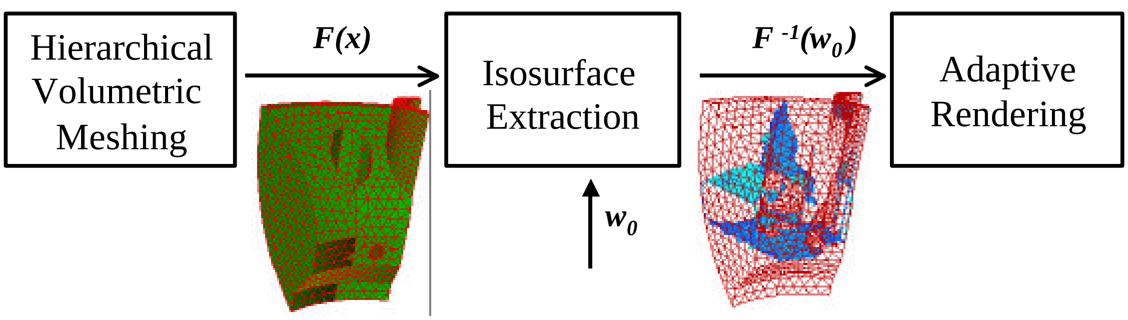


Figure 2: Pipeline of a visualization process. The overall data-flow efficiency is reduced by the conversions in data representation required by all the modules to improve their internal performance.

The scalar field *F(x)* is defined by a volumetric mesh with scalar values F_i assigned at the vertices v_i of the mesh. The user explores the data by selecting an isovalue w_0 and looking at the surface defined by $F^{-1}(w_0)$ (for example, regions of constant pressure). A hierarchical representation of the volumetric mesh is first built, a 2D multi-resolution representation of an isosurface is then extracted, and finally a view-dependent adaptive version of the isosurface is rendered.
Using multi-resolution representations at all stages of this sequence of computations is not sufficient for achieving efficiency in the overall data flow. For example, to optimize the rendering stage it is necessary to compute view-dependent adaptations of any isosurface that is visualized. If the required multi-resolution representation of each isosurface is built by decimation [LT99], then a full resolution representation of the isosurface must be constructed first. This would make any interactive exploration of the

data impossible, because each modification of the isovalue w_0 would require the computation of a new full resolution isosurface, which cannot be done at interactive rates.

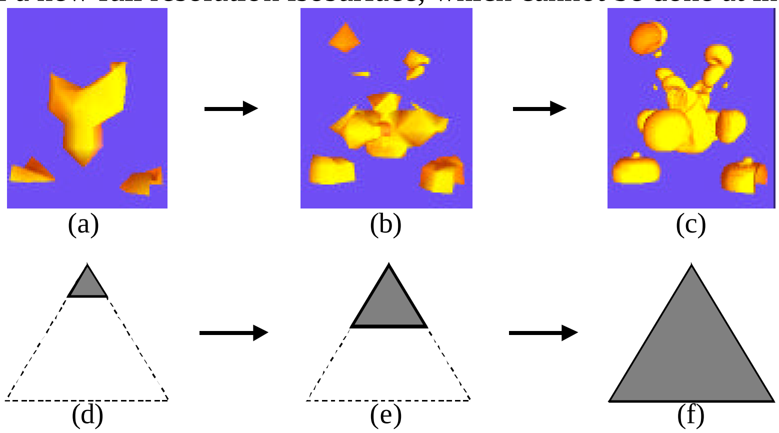


Figure 3: Progressive computation of an isosurface of the HiPIP (High Potential Iron Protein) dataset. The coarse resolution (a) can be rendered while being improved to become (b) and, at the end, (c). In (d) through (f), the hierarchical representation of the isosurface is represented by a triangle, where the solid gray part is the portion actually computed at a given stage and hence available to the rendering engine.

This problem can be solved in part by using the paradigm of *progressive computing* that Pascucci et al.[P00,PB00] have proven viable for the case of isosurface extraction.

Figure 3 shows the conceptual structure of the scheme. A coarse representation of the isosurface is built instantly. Then a multi-resolution representation is constructed progressively by successive insertions of new details. The input volumetric hierarchy is traversed once, from the coarse level to the fine level, independently of the number of different views displayed. During this traversal, a hierarchical representation of the isosurface is constructed, and maintained consistent at all times, so that the rendering engine can extract an adaptive model optimized with respect to the current viewpoint. No fine resolution isosurface needs to be built in the first place, and therefore some approximation of the output can always be generated without delay.

The application of the progressive computing paradigm allows redrawing the conceptual scheme of Figure 2 as in Figure 4. Each module builds a hierarchical representation of its output on-line while reading the input hierarchy. The advantages of this data-flow mechanism are enhanced when the scheme is implemented in a distributed environment because the penalty of having the data stored remotely is greatly reduced by the assumption of coherent access to the input hierarchy.

Several new progressive algorithms need to be developed. Volume rendering [DC88] is a fundamental visualization technique that complements isosurface extraction. It allows one to look at the entire volume of data by associating each point x with a color and transparency level dependent on the function value $F(x)$.

Progressive volume rendering is a major challenge when visibility occlusion is taken into account. While isocontours can be computed progressively we still need general-purpose surface simplification of other types of surfaces [BD00]. On-line volumetric decimation is necessary for the construction of multi-resolution representations of the datasets generated by large simulations.

For achieving maximum graphics performance, we need also to develop progressive schemes for graphic display of polygonal models tailored to the available graphics hardware.

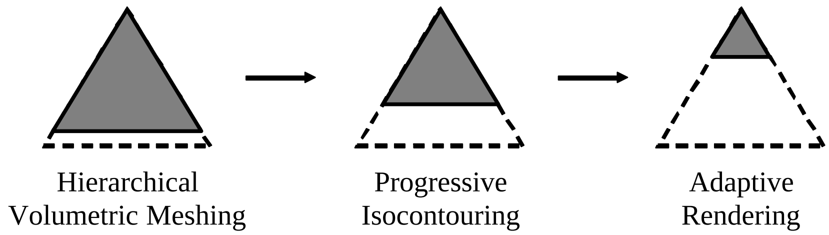


Figure 4: A sequence of three progressive modules allows optimization of the data-flow. Each hierarchical geometric model is represented by a triangle, where the solid gray part is the portion actually computed at a given stage and hence available to the next module. The rendering module is able to display an initial approximation of the result as soon as all the previous modules have started their computation, even if none of them has finished yet.

2.2. External Memory Data-structures and Distributed Computing

One of the main challenges that emerge when visualizing large datasets is the inability to load them into the main memory of the processing unit. This is mainly due to the difference in scale between the simulation computers and the visualization computers. A parallel computer with thousands of processors generates a dataset in days of execution using all its available memory. That same dataset then needs to be visualized interactively on a relatively small cluster of PCs, or even a single desktop workstation. Consequently a key component in the scalability of a visualization system is the ability to deal with data that is stored out-of-core or in external memory.

Out-of-core computing [V00] addresses the issues of algorithm redesign and external data layout restructuring necessary to enable data access patterns with minimal performance degradation. Research in this area is also valuable in parallel and distributed computing, where one has to deal with the similar issue of balancing processing time with data migration time. The solution to the out-of-core processing problem is typically divided into two parts: (i) algorithm analysis to understand the data access patterns and, when possible, redesign to maximize their locality; and (ii) storage of the data in secondary memory with a layout consistent with the access patterns of the algorithm, amortizing the cost of individual I/O operations over several memory access operations. We primarily focus our attention on cache-oblivious techniques [BDF00] since they optimize performance independently of the blocking factor of the secondary memory, and hence have the potential of increasing performance at all levels of the memory hierarchy of a modern supercomputer.

For the case of isosurface computation, Pascucci [BP99] showed how to create a static data partitioning that guarantees minimal data paging for out-of-core processing. This generalization of the optimal isocontouring algorithm [BPS96] also achieves very good load balancing when used in parallel. The main limitation of this approach is that it generates only fine resolution data. We are exploring a coupling with the decimation approach proposed by Lindstrom [Li00] to directly generate simplified versions of the output geometry. One important quality of this approach is the ability to simplify the topology of the input surface and to consequently guarantee a sufficiently small output even for extremely complex surfaces. For example, rendering an isosurface of the PPM dataset [M99] (see Figure 5) at interactive rates would greatly benefit from the reduction of its topological structure (number of tunnels, voids, and connected components). The minimum number of triangles necessary to represent the full topology of the surface can be prohibitive even for the fastest graphics hardware. The method in [Li00] has the ability to close small holes and merge nearby components, but generates a single decimated model at a fixed resolution, which is a major limitation in this context. In the next year we plan to remove this limitation to achieve full integration in the proposed multi-resolution framework.

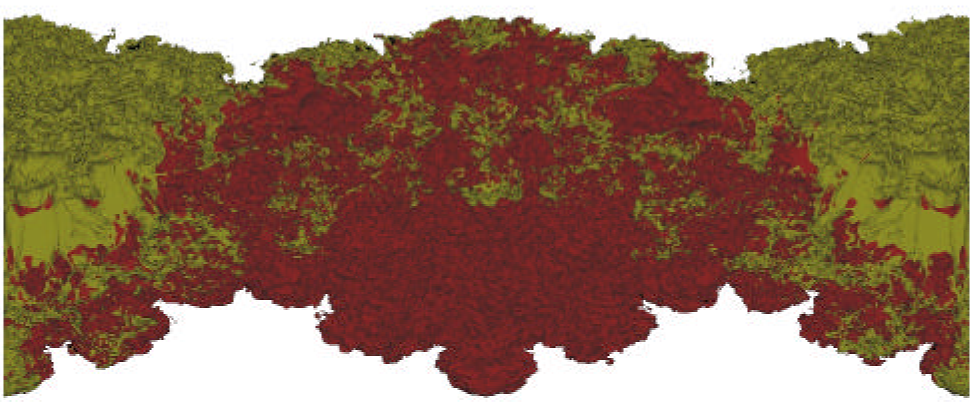


Figure 5: Isocontour extracted from a single time-step of the PPM simulation. The upper and lower faces of the surface are colored red and yellow. This surface contains 469M triangles and itself requires 9GB of disk space to store.

Ma [HM00, MC00] has developed algorithms for volume rendering of large datasets on clusters of PC's [MC00] and on large multiprocessor graphics workstations [HM00]. Several challenges still need to be addressed to achieve the level of scalability necessary for the needs of tera-byte simulation datasets.

The distributed system infrastructure that we want to enable is depicted in Figure 6. The system will involve three main types of components: Data Sources, Data Servers and Data Clients. These components will link a distributed, hierarchical, progressive data stream connecting directly the simulation code with different classes of visualization platforms. The Data Sources act as the data producer in the system and are typically the compute nodes of a large scientific simulation. In the next year we plan to experiment with the JEEP code (*ab initio* simulation of molecular dynamics [FG02]) as in situ simulation engine with Data Sources. The Data Servers constitute the central data processing components in the system. Each Data Server takes a data stream as input and acts as a filtering and storage agent. A Data Server outputs a data stream either to another Data Server or to a Data Client in response to a set of spatially bounded data queries. The Data Clients act as data stream sinks. We focus on Data Clients that perform scientific visualization with fundamental techniques like cut planes or performs progressive volume rendering and iso-surfacing. In general a Data Client makes spatially bounded queries to its upstream Data Servers in response to interactive user requests (GUI events). The Data Client then receives the data streams corresponding to the samples satisfying the queries and progressively renders the information received, either on a desktop workstation or on a PowerWall display. We will focus on improving current distributed rendering techniques [HB00] to take advantage of the progressive streaming framework, while using hardware accelerations in each local rendering engine.

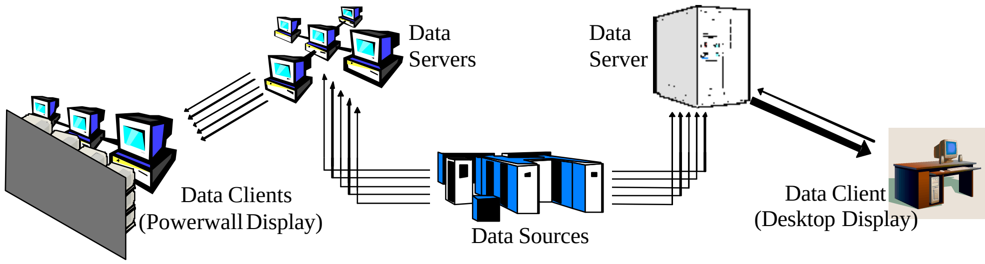
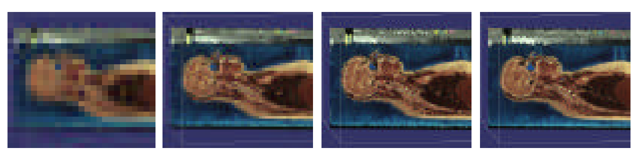


Figure 6. Schematic representation of the planned ViSUS distributed system with three types of components. The *Data Sources* create the input datasets and are typically the compute nodes in a large scientific simulation. The *Data Servers* are filters that take care of eventual transformations in the data representation and provide services like the long term storage. The *Data Clients* perform interactive queries access the data through the *Data Servers* for visualization purposes.

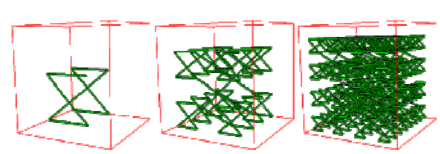
3. ACCOMPLISHMENTS AND MILESTONES

The ViSUS project involves novel research both at the theoretical level of designing new algorithms, data-structures, and, at the system level, a new streaming infrastructure. We report below a list of main technical accomplishments. In summary, we have successfully completed our milestones and released a software prototype that fully demonstrates the capabilities that where envisioned at the beginning of the project. In the process we have published 31 peer-reviewed papers (highlighted in the bibliography) and produced software tools for the visualization of 2D/3D scalar fields and surface meshes as well as a streaming infrastructure currently deployed on the fastest supercomputer in the world (Blue Gene/L).

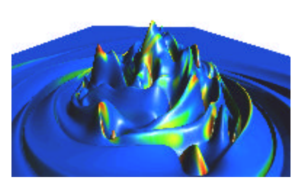
Technical Highlights



- *Progressive slicing of rectilinear grids*. The first progressive visualization technique that we have implemented is a simple slicing through rectilinear grids with planar cross sections. A sequence of atomic improvements (see figure on the right) is performed while maintaining a consistent texture to allow the display of a valid picture at any time.

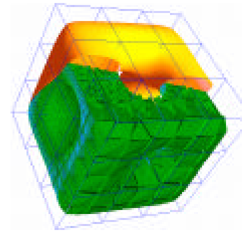


- *Cache-oblivious representation of rectilinear grids [PF01, PF02]*. We defined a cache oblivious data layout for rectilinear grids based on the recursive definition (shown on the right) of the Z-order space-filling curve, which allows achieving orders of magnitude of speed-up in data access when used in combination with a progressive visualization algorithm.



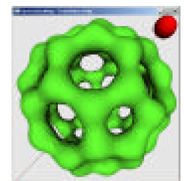
- *Hierarchical geometric models with reduced dependencies [LP02,LP01,ML02]*. We developed a novel multi-resolution representation for geometric models. This scheme reduces the complexity of the hierarchical representation and minimizes the cost of building view-dependent adaptive models.

- *Subdivision methods for general unstructured meshes [P02].* We are working on the fundamental problem of devising visualization techniques for general unstructured meshes and that are as efficient as those for rectilinear grids. To achieve this goal we introduced the first subdivision method for unstructured meshes of any dimension and cells of virtually any topological type.

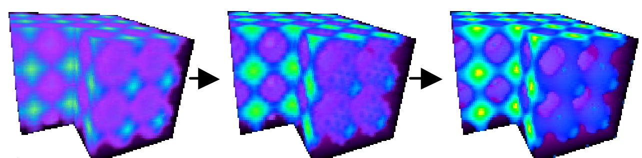


- *Memory insensitive surface simplification [LS01].* We have developed a technique dealing with both input and output triangulated surface models that are much larger that the main memory of the processing unit.

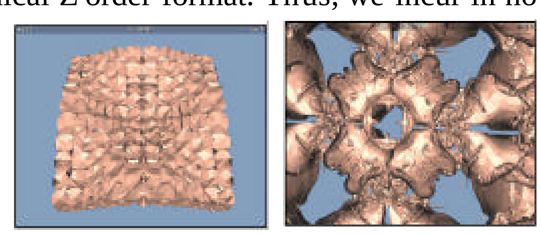
- *View dependent progressive isocontouring*. We developed [GD02] a system combining the original progressive isocontouring algorithm [PB00] with the multi-queue approach introduced in [DM97]. This allows combining the progressive computation of the isosurface with the ability to construct a minimal view-dependent representation.



- *Preliminary testing of the streaming infrastructure*. We tested the basic network components that will be needed for our streaming infrastructure (see Figure 5) and committed to a solution based on the use of direct socket connections.
- *Complete prototype streaming infrastructure*. We have implemented a complete end-to-end prototype of the ViSUS infrastructure [PL03], which realizes the diagram in see Figure 5. The Figure on the right shows the progressive visualization of a Computational Fluid Dynamics simulation (JEEP code) obtained for time step *t* while time step *t+1* is being computed. The system is based on the following three main components.

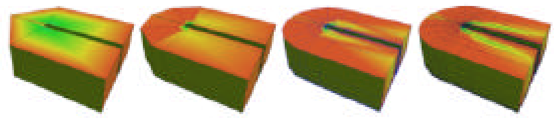


    - A remote viewer that allows interactive exploration for datasets that are retrieved on demand from a remote storage system.
    - A direct streaming module that connects the nodes of a scientific simulation to a set of Data Servers that are used for permanent storage. We have used the JEEP code in our performances tests. This module establishes *m*-to-*n* direct interconnect between *m* JEEP compute nodes (Data Sources) and *n* storage nodes (Data Servers).
    - A new algorithm for embedded the distributed preprocessing and streaming of rectilinear grids. For large rectilinear grids the data reordering of our multi-resolution Z-order layout can take 20 minutes (for a $2k^3$ grid) if performed locally on a Data Server. Our scheme allows computing the same reordering during the data streaming stage with negligible overhead. In this the compute nodes (Data Sources) store the data in a traditional way but the Data Servers receive the data already in our hierarchical Z-order format. Thus, we incur in no practical preprocessing cost.
- *External memory multi-resolution representations [L03]*. We have developed

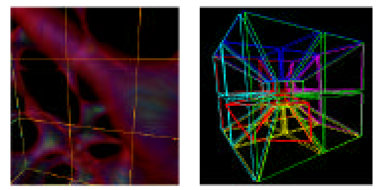


a new technique allowing construction and traversal of multi-resolution surface meshes in external memory. This method enables the interactive visualization of very large surfaces meshes ? not necessarily isosurfaces? on desktop workstations.

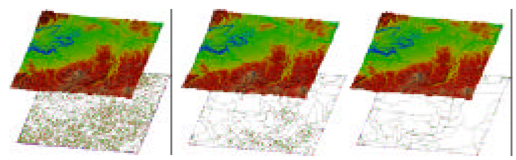
- We have worked [LP02b,LP03,LG04, IL03] on the definition and implementation of wavelet multi-resolution models to be combined with our new subdivision scheme for volumetric meshes. This step will complete the definition of our new semi-regular representation for general geometric models, enabling the future use of our out-of-core techniques for unstructured meshes.



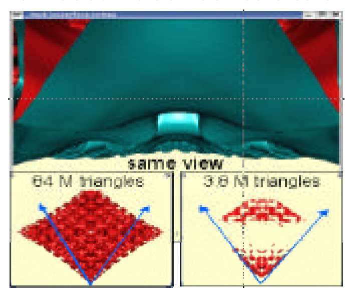
- We have developed a new image cache front-end decoupling the image display from its rasterization. This allows maintaining interactive rate of data exploration even for slow rendering algorithm. Moreover it allows easy parallelization of generic rendering techniques. We plan to implement scheduling strategies based on error tables [NL03] for dynamic data.



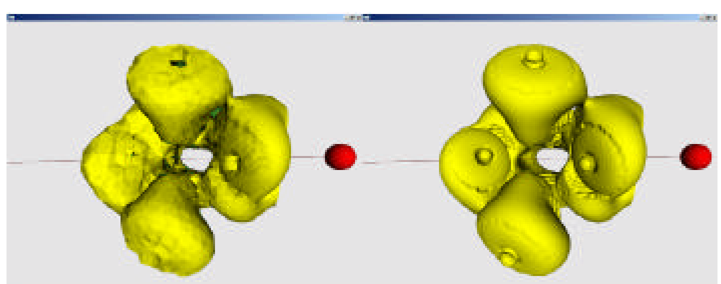
- We implemented topology-based multi-resolution techniques [PC03,CE03,EH03,P04] that provide high quality approximations and guidance in the data exploration process.

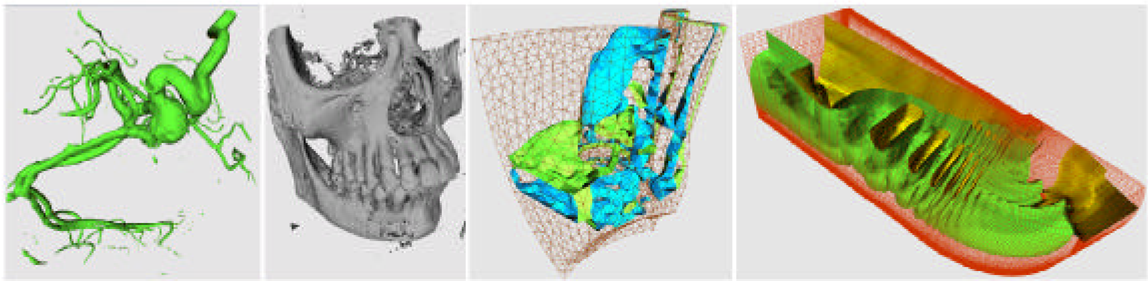


- We have developed a new technique that accelerate the isosurface construction process by taking advantage of occlusion culling [PLP04]. For very high complexity isosurfaces this is crucial to achieving improved performance since one major bottleneck is the repeated construction and drawing of portions of isosurfaces that are not visible to the user. The figure on the right shows the case of an isosurfaces from a simulation dataset showing how one can generate exactly the same view while reducing the triangle count of the rendered surface by a factor of twenty (from 84M triangles down to 3.8M triangles).

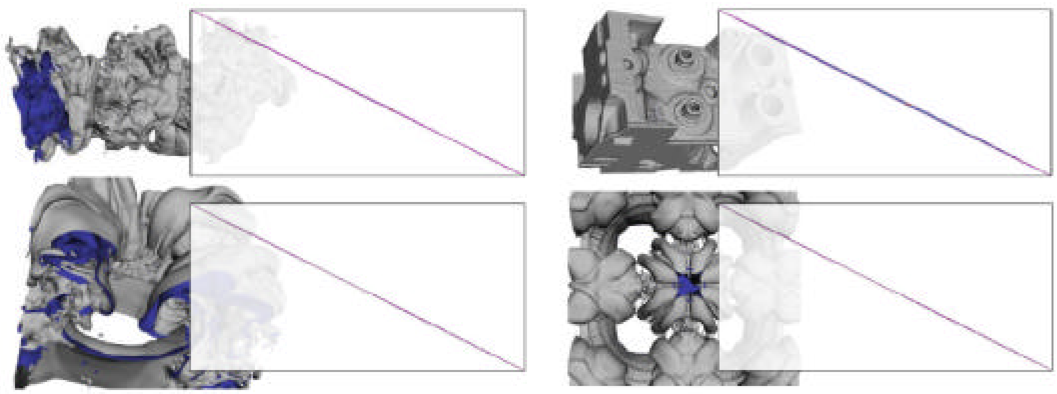


- We extended our nested error metric approach to enable fast and simple selection of view-dependent refinements of volumetric data [GD02]. This has been crucial for improving our progressive isosurfaces refinements from uniform for adaptive view dependent (see figure above with viewpoint marked by a red sphere). We achieve this goal without adding any storage or time overhead, which were previously required for fixing cracks in an adaptive mesh.





- We sped-up and simplified the computation of isosurfaces by taking advantage of the compute capabilities of Graphics Processing Units in modern commodity graphics hardware available on desktop workstations [P04b]. The figure above shows several examples of isosurfaces generated with this technique.



- We developed a new data layouts for unstructured meshes that allow for multi-resolution streaming of large triangulated models. This allows the remote visualization of surface meshes (e.g. material boundaries) computed with external memory tools [MIP04,IL04,ILG03].
- **Demonstrations:** the ViSUS streaming infrastructure enabled the real-time execution and visualization (see Figure 1) of the two LLNL simulation codes (Miranda and Raptor) run at Supercomputing 2004 on Blue Gene/L at its presentation as the fastest supercomputer in the world. In addition to SC04, we have run live demonstrations at the IEEE VIS conference and at invited talks at the DOE MICS office, DOE computer graphics forum, UC Riverside, and the University of Maryland. In all cases we have shown the capability to stream and visualize interactively data stored remotely at the San Diego Supercomputing Center or monitor in real-time simulation codes executed on a cluster of PC's at LLNL.

4. PRINCIPAL INVESTIGATOR AND REASEARCH TEAM

We have a team of scientists with proven strengths in the theoretical development and practical implementation of algorithms in all the aspects of scientific visualization. This includes computer graphics, data analysis, compression, and parallel/out-of-core computing. The members of the core team, Valerio Pascucci (PI), Peter Lindstrom, and Daniel Laney, have published peer-reviewed papers in top conferences and journals in the area of scientific visualization, computer graphics, computational geometry and data compression.

Valerio Pascucci earned his Ph.D. in Computer Science in May 2000. In his thesis he first proved the viability of the paradigm of progressive computing (central theme of this proposal) in scientific visualization with the design and implementation of the progressive isocontouring algorithm. His research in scientific visualization has lead to the first isocontouring algorithm that has optimal performance with minimal storage overhead. He has introduced the Hyper-Volume Rendering algorithm that extends the volume rendering approach to scientific datasets of dimension higher than three and the Contour Spectrum interface that analyzes scientific data to produce simple diagrams guiding the user in the selection of visualization parameters. Other research interests include foundations of representation schemes for geometric data and compression techniques.

Peter Lindstrom earned his Ph.D. in Computer Science in November 2000. In his thesis work he extensively studied error metrics for triangulated models both in geometric space and in image space. Fundamental contributions that are relevant for this project have been in view-dependent simplification of large terrains, out-of-core simplification of triangulated models and image-based mesh decimation and optimization.

Daniel Laney is finishing his Ph.D. in Applied Sciences (expected graduation in June 2002). His thesis research has been focused on adaptive refinement and progressive encoding of distance fields. He has been working also in multi-resolution wavelet encoding of level sets. His background in applied physics makes his particularly valuable in our effort to develop a streaming infrastructure that starts from and includes the simulation code.

5. COLLABORATORS

We have an on-going collaboration with the LLNL research scientists Mark Duchaineau, and Randall Frank, and with professors Bernd Hamann, Kenneth Joy, and Kwan-Liu Ma (Computer Science Department, University of California at Davis) on the themes of multi-resolution methods, wavelet representation, and computer graphics with particular focus on the problem of processing and visualizing of large scientific datasets. The university collaboration is found on continuous reciprocal visits and co-advising of Ph.D. students developing theses on topics of interest to the laboratory. The students spend usually the summer visiting the laboratory or are resident for the entire year as student employees after finishing their course requirements.

Mark Duchaineau was the leader of the SAVAnTS research project in CASC. His recent research includes multi-resolution representations and algorithms with application to high-performance interactive exploration, mechanical design, and wavelet compression. His research in adaptive and output-sensitive data reduction methods has lead to the definition of the ROAM algorithm for the view dependent simplification and on-line adaptation of surface meshes. Duchaineau was recently the co-recipient of the Gordon Bell Prize in Supercomputing for devising new compression methods to make possible the visualization of [M99], and he has similarly assisted with record-breaking molecular dynamics simulations on ASCI White.

Randall Frank is the former project leader of the ASCI/VIEWS visualization effort and currently working at CEI. He has experience in the design and implementation of innovative visualization systems, which is particularly valuable for the design and experimentation with the high level layer of our new infrastructure. Before joining LLNL

Frank was Senior Systems Architect in Research Systems Inc. and designed the visualization subsystem of IDL (Interactive Data Language). More recently he has been involved in the CROMIUM research project (collaboration with Stanford university) for the definition of a generic interface for parallel and distributed visualization systems.

Bernd Hamann is a full professor of computer science at the University of California, Davis, since 1995. Professor Hamann serves on the editorial board of the IEEE Transactions on Visualization and Computer Graphics and served as a papers co-chair/proceedings co-editor for the IEEE Visualization conferences in 1999 and 2000. Professor Hamann's main research and teaching interests are visualization, geometric modeling (or computer-aided geometric design), computer graphics, and immersive environments. His current research focuses on hierarchical representations and visualization methods for very large scientific data sets.

Kenneth Joy is a Professor and founding member of the Computer Science Department at the University of California, Davis, since 1983. Professor Joy's research and teaching areas are visualization, geometric modeling, and computer graphics. He is a well-known expert in solid modeling, multivariate splines, subdivision methods for general 3D grids, surface reduction and accelerated rendering techniques. He is a faculty researcher in the Center for Image Processing and Integrated Computing. This Center provides the research environment for the Graphics and Visualization Research Group at UC Davis, and it provides an interdisciplinary research environment where practical visualization problems from a variety of disciplines can be addressed.

Kwan-Liu Ma has been a Professor in the Computer Science Department at UC Davis since July 1999. Professor Ma had worked at ICASE, located at the NASA Langley Research Center, Virginia. His current research and teaching are in the areas of Scientific Visualization and Computer Graphics. Particularly valuable for this project is his research experience in parallel and distributed visualization systems. Professor Ma recently received the NSF Career Award and the Presidential Early Career Award for Scientists and Engineers (PECASE). He organized a workshop for NSF/DOE (May '99), as well as a course and a panel for SIGGRAPH '99, all in the area of Large-Scale Data Visualization. He has served as a guest-editor for the Sep/Oct 1999, Sep/Oct 2000, and Jul/Aug 2001 issues of IEEE Computer Graphics & Applications.

6. Strategic Alignment

The research work developed in this project has enabled new scalable visualization techniques that specifically target the needs of the scientists at the Laboratory to explore in real time very large datasets in several different computing environments, from a desktop workstation to a high-resolution power-wall display. The full deployment of these techniques will provide the Laboratory with the technology necessary to improve the level of interactivity of scientific visualization systems by orders of magnitude. Several major bottlenecks in the visualization process will be removed by the combined ability to produce instant coarse representations and streaming update information through a sequence of processing stages possibly distributed over a network. While our initial focus has been on large ASCI datasets, the technology developed applies in general to large scientific simulations (e.g. turbulent fluid mixing, meteorological simulations, combustion, etc.) or experimental setting that produce quickly large amounts of data.

The Laboratory will benefit by the availability of this new technology at two levels. At the deployment level, the improved efficiency in the use of hardware resources will reduce the cost of the necessary visualization hardware infrastructure. At the scientific level, the developed technology will enable scientists in the lab designing large simulation datasets to utilize their time more effectively, reducing the overall time of the design, simulation, and visualization cycle.

We will enable remote monitoring of large and expensive simulations, allowing for example to save computing resources through early termination and restart (with new initial conditions) of erroneous test simulations. For simulation codes with internal mechanisms for dynamic modification of their running conditions this would enable runtime steering of the simulation.

The out-of-core computing techniques that we have developed are relevant for a wide class of applications. Our focus on cache-oblivious approaches allows performance improvements at the various memory cache levels even for usual in-core processing. We plan to continue our close interaction with the simulation and data mining groups to facilitate the transfer of this technology and allow other efforts to benefit from the same performance gains.

7. CURRENT EFFORTS AND FUTURE DIRECTIONS (EXIT PLAN)

While the funding stage form the LDRD office is ended, the ViSUS project is currently very active. Our current main effort is devoted to the deployment of the capabilities to the end users. In particular we are focusing on the scalability of the streaming infrastructure to the full 64K nodes Blue Gene/L machine. This effort is mainly supported by the ASCI DVS group. Our release of the ViSUS library and IDX file format has allowed integration in other tools that may take advantage, at least in part, of the benefits inherent in using our hierarchical space-filling curves technology. In addition to the Terascale Browser (developed by ASCI DVS) two other visualization softwares are now capable to load directly IDX data: ViSIT, open source tool developed at LLNL, and Ensight, commercial tool developed by CEI.

The Miranda code is already dumping both vis and restart dumps in IDX format. Other simulation codes are in line to achieve the same goal. We are extending the streaming infrastructure to a multi-block format do enable direct dumps from AMR codes.

A new research direction that is being developed is to use and extend the IDX format for image processing purposes. This work will be supported by NGI and will be started during the current fiscal year for a period of three years (contract already in place).

Overall the seed funding providing by the LDRD office have provided us the opportunity do develop the core technology that we are currently deploying and expanding in new directions.

Bibliography (ViSUS publications in bold)

[BD00] K.Bonnell, M.A.Duchaineau, D.R.Schikore, B.Hamann and K.I.Joy, "Constructing Material Interfaces from Data Sets Containing Volume Fraction Information", *IEEE Visualization 2000*.

[BDF00] M.A.Bender, E.D.Demaine, and M.Farach-Colton, ``Cache-Oblivious B-Trees," *Proceedings of the 41st Annual Symposium on Foundations of Computer Science (FOCS 2000)*, Redondo Beach, California, November 12-14, 2000, pages 399-409.

[BDH00] Martin Bertram, Mark A. Duchaineau, Bernd Hamann and Kenneth I. Joy, "Bicubic Subdivision-Surface Wavelets for Large-Scale Isosurface Representation and Visualization," *Proceedings of IEEE Visualization 2000*, Salt Lake City, Utah, October, 2000.

[BEH04] P.-T. Bremer, H. Edelsbrunner, B. Hamann, and V. Pascucci. "A topological hierarchy for functions on triangulated surfaces". ***IEEE Transactions on Visualization and Computer Graphics*****, 10(4):385–396, July/August 2004. UCRL-JRNL-208700.**

[BP99] C.Bajaj, V.Pascucci, D.Thompson, X.Y. Zhang, "Parallel Accelerated Isocontouring for Out-Of-Core Visualization"**,** In Proceedings of *IEEE Parallel Visualization and Graphics Symposium, October 24-29,1999 San Francisco, CA*, pp.97-104.

[BPS04] R. Borgo, V. Pascucci, and R. Scopigno. "Massive data pre-processing with a cluster based approach". In Proceedings of the ***2004 Eurographics Symposium on Parallel Graphics and Visualization (EGPGV-04)*****, pages 67–74, Aire-la-Ville, Switzerland, June 10–11 2004. UCRL-PROC-208678.**

[BPS96] C.L. Bajaj, V. Pascucci and D.R. Schikore, "Fast isocontouring for improved interactivity". *Proceedings of the Symposium on Volume Visualization*, pages 39-46. IEEE Computer Society, October 1996.

[CE03] K. Cole-McLaughlin, H. Edelsbrunner, J. Harer, V. Natarajan, and V. Pascucci. "Loops in Reeb Graphs of 2-Manifolds". In Proceeding of the 19-th ACM Symposium on Computational Geometry 2003, pp344–350, June 2003. UCRL-JC-151933.

[CEH04] K. Cole-McLaughlin, H. Edelsbrunner, J. Harer, V. Natarajan, and V. Pascucci. "Loops in reeb graphs of 2-manifolds". ***Discrete and Computational Geometry*****, 32(2):231–244, July 2004. UCRL-JRNL-20870.**

[CP04] K. Cole-McLaughlin and V. Pascucci. "Multiresolution representation of topology". In Proceedings of the ***4th IASTED International Conference on Visualization, Imaging, And Image Processing (VIIP 2004)*****, pages 282–289, Marbella, Sapin, September 2004. UCRL-PROC-208680.**

[DC88] R.A.Drebin, L.Carpenter, and P.Hanrahan, "Volume Rendering", *Computer Graphics*, August 1988. **22**(4): p. 65-74.

[DM97] M.Duchaineau, M.Wolinsky, D.E.Sigeti, M.C.Miller, "ROAMing Terrain: Real-time Optimally Adapting Meshes". *Proceedings of IEEE Visualization '97*, 1997,, pp. 81-88.

[Du00] M.Duchaineau, "SAVAnTS: Scalable Algorithms for Visualization and Analysis of Terascale Science", Exploratory Research in the Disciplines, Computation Directorate, Lawrence Livermore National Laboratory, 2000.

[EH03] H. Edelsbrunner, J. Harer, V. Natarajan, and V. Pascucci, "Morse Complexes for Piecewise Linear 3-Manifolds", In Proceeding of the 19-th ACM Symposium on Computational Geometry 2003, pp361–370, June 2003. UCRL-PROC-208698.

[EHM04] H. Edelsbrunner, J. Harer, A. Mascarenhas, and V. Pascucci. "Time-varying Reeb graphs for continuous space-time data". In Proceedings of the *twentieth annual symposium on Computational geometry (SoCG '04)*, pages 366–372. ACM Press, 2004. UCRL-PROC-208697.

[EHNP04] H. Edelsbrunner, J. Harer, V. Natarajan, and V. Pascucci. "Local and global comparison of continuous functions". In Proceeding of the *IEEE conference on Visualization*, pages 275–280, October 2004. UCRL-PROC-208677.

[FG02] J.L.Fattebert, and F. Gygi, "Density Functional Theory for Efficient Ab Initio Molecular Dynamics Simulations in Solution," *J. Comput. Chem., in press* (2002). Also available as Lawrence Livermore National Laboratory technical report UCRL-JC-143326, April 2001.

[GD02] B. Gregorski, M. Duchaineau, P. Lindstrom, V. Pascucci, K. I. Joy, "Interactive View-Dependent Rendering of Large Iso-Surfaces", in Proceedings of IEEE Visualization Conference 2002, pp475-482, October 2002. UCRL-JC-146819.

[GP05] C. Guerra and V. Pascucci. "Line-based object recognition using Hausdorff distance: from range images to molecular secondary structure". *Image and Vision Computing*. To appear. UCRL-JRNL-208551.

[HB00] G.Humphreys, I.Buck, M.Eldridge and P.Hanrahan, "Distributed Rendering for Scalable Displays". *Proceedings of Supercomputing 2000*.

[HM00] C.Hofsetz and K.L. Ma, "Multi-threaded Rendering Unstructured-Grid Volume Data on the SGI Origin 2000". *3rd Eurographics Workshop on Parallel Graphics and Visualization*, EU, 2000.

[IL03] L. Ibarria, P. Lindstrom, J. Rossignac, and A. Szymczak, "Out-of-core compression and decompression of large n-dimensional scalar fields". In proceedings of *Eurographics 2003*, September 2003, pp. 343-348. UCRL-JC-151934-REV-1.

[IL04] M. Isenburg and P. Lindstrom. Streaming Meshes. LLNL Technical Report, April 2004. UCRL-CONF-201992.

[ILG03] M. Isenburg, P. Lindstrom, S. Gumhold, and J .Snoeyink, Large Mesh Simplification using Processing Sequences. IEEE Visualization 2003, October 2003, pp. 465-472. UCRL-CONF-200016.

[KO04] M. J. van Kreveld, R. van Oostrum, C. L. Bajaj, V. Pascucci, and D. R. Schikore. "Efficient contour tree and minimum seed set construction", Chapter in: *Surface Topological Data Structures: An Introduction for Geographical Information Science*, pages 71–86. John Wiley & Sons, May 2004. UCRL-BOOK-200018.

[L00] P.Lindstrom "Model Simplification using Image and Geometry-Based Metrics". Ph.D. thesis, Georgia Institute of Technology, November 2000.

[L03] P. Lindstrom. "Out-of-Core Construction and Visualization of Multiresolution Surfaces". In Proceedings of the ACM Symposium on Interactive 3D Graphics 2003, April 2003, pp. 93-102, 239. UCRL-JC-150863.

[LGP04] L. Linsen, J. T. Gray, V. Pascucci, M. A. Duchaineau, B. Hamann, and K. I. Joy. Hierarchical "Large-scale Volume Representation with 3rd-root-of-2 Subdivision and Trivariate B-spline Wavelets", Chapter in: *Geometric Modeling for Scientific Visualization*, pages 259–378. Mathematics and Visualization. Springer-Verlag, Heidelberg, Berlin, February 2004. UCRL-JC-146824

[Li00] P.Lindstrom, "Out-of-Core Simplification of Large Polygonal Models". *Proceedings of ACM SIGGRAPH 2000*, July 2000, pp. 259-262.

[LP01] P.Lindstrom and V. Pascucci. "Visualization of large terrains made easy". In Proceeding of *IEEE Visualization*, 2001. pp. 363–370. UCRL-JC-144753.

[LP02] P.Lindstrom and V. Pascucci. "Terrain Simplification Simplified: A General Framework for View-Dependent Out-of-Core Visualization," *IEEE Transactions on Visualization and Computer Graphics*, 8(3), July-September 2002, pp. 239-254. UCRL-JC-147847.

[LP02b] L. Linsen, V. Pascucci, M. A. Duchaineau, B. Hamann, K. I. Joy, "Hierarchical Representation of Time-varying Volume Data with 4th-root-of-2 Subdivision and Quadrilinear B-spline Wavelets", in Proceedings of 10-th Pacific Conference on Computer Graphics and Applications, pp 346–355, October 2002. UCRL-JC-151063.

[LP03] L. Linsen, B. Hamann, K. I. Joy, V. Pascucci, and M. A. Duchaineau. "Wavelet-based multiresolution with n-th-root-of-2". *Computing*, 72(1-2):129–142, April 2004. UCRL-JRNL-208695

[LK96] P.Lindstrom, D.Koller, W.Ribarsky, L.F.Hodges, N.Faust, and G.A.Turner. "Real-Time, Continuous Level of Detail Rendering of Height Fields". *Proceedings of ACM SIGGRAPH 96*, August 1996, pp. 109-118.

[LT99] P.Lindstrom and G.Turk, "Evaluation of Memoryless Simplification". *IEEE Transactions on Visualization and Computer Graphics*, 5(2), April-June 1999, pp. 98-115

[LS01] P.Lindstrom and C. Silva. "A Memory Insensitive Technique for Large Model Simplification". IEEE Visualization 2001, October 2001, pp. 121-126, 550. UCRL-JC-144550.

[M99] A. Mirin, “Performance of Large-Scale Scientific Applications on the IBM ASCI Blue-Pacific System”, *Proceedings of the Ninth SIAM Conf. of Parallel Processing for Scientific Computing*, Philadelphia, March 1999.

[MC00] K.L. Ma and D. Camp, “High Performance Visualization of Time-Varying Volume Data over a Wide-Area Network. *Proceedings of Supercomputing 2000*, November 2000**.**

[MIP04] A. Mascarenhas, M. Isenburg, V. Pascucci, and J. Snoeyink. “Encoding volumetric meshes for streaming isosurface extraction”. In Proceedings of the *International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT)*, pages 293–300, September 2004. UCRL-PROC-208696.

[ML02] P. Miller, P. Lindstrom, and A. Cook, “Visualizations of the Dynamics of a Vortical Flow”, winner of the video category at 20th Annual Gallery of Fluid Motion, American Physical Society, November 2002. UCRL-VID-150303.

[NL03] C. Nuber, E. C. LaMar, V. Pascucci, B. Hamann, K. I. Joy, "Using Graphs for Fast Error Term Approximation of Time-varying Datasets", In Proceedings of Eurographics VisSym, May 2003, Grenoble, France, pp 9-18. UCRL-JC-151965.

[P00] V. Pascucci, “Multi-resolution and Multi-dimensional Data-structures for Scientific Visualization”. Ph.D. thesis, Computer Science Department, Purdue University, May 2000.

[P02] V. Pascucci. “Slow growing subdivisions in any dimension: towards removing the curse of dimensionality”. In *Proceedings of Eurographics 2002*, pp.451-460. UCRL-ID-144257.

[P04] V. Pascucci. “Topology Diagrams in Scientific Visulization”, Chapter in: *Surface Topological Data Structures: An Introduction for Geographical Information Science*, pages 121–130. John Wiley & Sons, May 2004. UCRL-BOOK-200013.

[P04b] V. Pascucci. “Isosurface computation made simple: Hardware acceleration, adaptive refinement and tetrahedral stripping”. In Proceedings of the *Joint Eurographics -IEEE TVCG Symposium on Visualization (VisSym)*, pages 293–300, May 2004. UCRL-CONF-202459.

[PB00] V. Pascucci and C.L. Bajaj, “Time critical adaptive refinement and smoothing”. Proceedings of the ACM/IEEE Volume Visualization and Graphics Symposium, pp 33--42, Salt lake City, Utah, October 2000.

[PC03] V. Pascucci, K. Cole-McLaughlin, “Parallel Computation of the Topology of Level Sets”, Algorithmica, 38(1):249–268, October 2003. UCRL-JRNL-208707.

[PF01] V. Pascucci and R.J.Frank “Global Static Indexing for Real-time Exploration of Very Large Regular Grids”. In proceeding of *14th Annual Supercomputing conference*, 2001, Denver, Co. On-line proceedings http://www.sc2001.org/techpaper.shtml. UCRL-JC-144754.

[PF02] V. Pascucci and R.J.Frank “Hierarchical Indexing for Out-of-Core Access to Multi-Resolution Data”. Chapter in *Hierarchical Approximation and Geometrical*

***Methods for Scientific Visualization*. B. Hamann, G.Farin and H. Hagen editors, pp.225-242,Springer, Berlin, 2003. UCRL-JC-140581.**

[PL03] V. Pascucci, D. E. Laney, R. Frank, G. Scorzelli, L. Linsen, B. Hamann, F. Gygi, "Real-Time Monitoring of Large Scientific Simulations", In Proceedings of the 18-th annual *ACM Symposium on Applied Computing*, March, 2003, Melbourne, FL, pp.194-198. UCRL-JC-151094.

[PLP04] S. Pesco, P. Lindstrom, V. Pascucci, and C. Silva. "Implicit occluders". In Proceeding of the *IEEE/SIGGRAPH Symposium on Volume Visualization and Graphics*, pages 47–54, October 2004. UCRL-CONF-202186.

[PPG04] A. Paoluzzi, V. Pascucci, and G. Scorzelli. "Progressive bsp representation and boolean operations for dimension-independent polyhedra". In Proceeding of the *ACM Symposium on Solid Modeling*, pages 203–211. ACM, June 2004. UCRL-PROC-208679.

[SM97] W. Schroeder, K. Martin, and B. Lorensen, "The Visualization Toolkit: An Object-Oriented Approach To 3D Graphics", Prentice Hall, 1997.

[V00] J.S.Vitter, "External Memory Algorithms and Data Structures: Dealing with MASSIVE DATA", *ACM Computing Surveys*, March, 2000.