

# サイエンティフィック ビジュアライゼーションの 最新動向

岩手県立大学 小山田 耕二

koyamada@soft.iwate-pu.ac.jp

サイエンティフィックビジュアライゼーション（以降可視化）という言葉は、数値シミュレーションに関係する科学技術者の間で浸透しているものの、一般的な情報処理技術者にとってはまだよく分からないというのが現状である。そこで、本稿では、まず、可視化についてその背景、経緯、そしてその技術の基本について解説する。続いて、可視化の最新動向について3つの話題を紹介する。1つ目は、1999年発表された安価な専用ボードの紹介を含むボリュームレンダリングの最新動向である。2つ目は、米国のAccelerated Strategic Computing Initiative（以降ASCI）プロジェクトで使われているような大規模データの可視化技術である。3つ目は、WWWベースの可視化技術である。最後に、可視化対象データについての内容記述について少しふれる。

## 可視化とは？

サイエンティフィックビジュアライゼーション（以降可視化）というのと何やら難しそうだが、部屋の温度分布を目に見えるようにするための技術といえは少しは分かりやすいだろうか？ 床に近い、しかも日当たりの悪い所では温度も低いだろうし、天井付近でしかも照明装置近辺だと温度も高そうである。温度計を手にして計っていけば、空間の各点で温度値は違うはずである。可視化技術は、このような3次元空間において定義された数値データを分かりやすく表示するために用いられる。

可視化の意義を最初に明確な形で唱えたのは、1987年にACMから出された“Visc (Visualization in Scientific Computing) レポート”であろう。この中で可視化について、“Visualization offers a method for seeing the unseen.”<sup>1)</sup>と記述している。“See the unseen”という言葉に集約されているように可視化は、本来ならば見ることでできないデータを人間に見える形で提示する技術である。最近では、可視化の対象として、データベースに格納された多次元のデータも対象とするようになり、特に“情報可視化 (Information Visualization)”という研究領域を形成している。本稿では、3次元空間で定義された数値データの可視化を対象とする。情報可視化について興味のある読者は情報処理学会誌Vol.41, No.4で組まれた特集“ビジュア

ルな情報検索”を一読することをおすすめる。

Viscレポート出版以降、CGの世界最大規模の国際会議“ACM-SIGGRAPH”では、多くの基本となる可視化手法が発表された。1987年から1988年にかけて、等値面生成法、ボリュームレンダリング法が次々と発表された。これらの手法は、今日、多くの研究のベースとなっている。ACMでは、1992年以降隔年ごとに可視化に特化した国際会議を開催している。IEEEでは、1990年以降毎年可視化に特化した国際会議を開催している。2000年は10月9日から13日までソルトレイクシティで行われた (<http://www.erc.msstate.edu/conferences/vis00/index.html>)。また、最近、「可視化」に特化した論文誌“Transactions on Visualization and Computer Graphics”を出版するようになった。国内においては、1991年に情報処理学会論文誌 (Vol.32, No.5) で特集を行った。以降、いろいろな学会において「可視化」の特集が組まれてきた。

全体の流れでいえるのは、映像産業向けに開発されたCG技術が可視化技術のシーズとなっていることであろう。Viscレポートがこの動きを加速したと考えるのが自然である。写実的な雲や惑星爆発といったシーン生成のためのCG技術が1980年代に開発され、それらがボリュームレンダリング法を生み出したといえる。また、画像処理で使われるコンボリューション技術は、Cabralらが提案したLine Integral Convolution (LIC) 法<sup>4)</sup>のベースとなってい

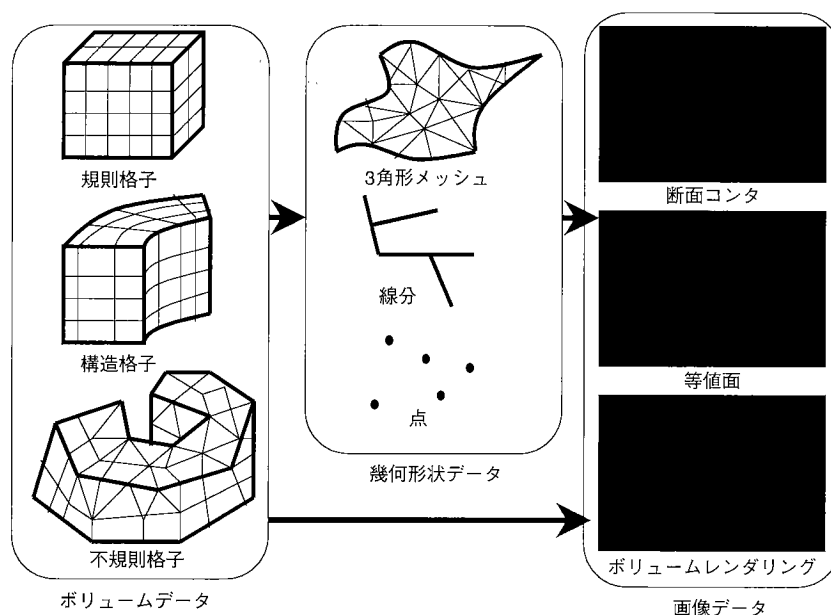


図-1 可視化プロセス

る。ごく最近提案された可視化手法には、非写実的なCG表現手法の流れを汲むものもある。ブラウン大学のR. M. Kirbyらは、イラストレータの描画法のエッセンスを取り入れて速度場を分かりやすく表現する可視化手法を開発した。今後は、可視化対象の拡大(ベクタ場, テンソル場), 大規模化, 広域化への対応といった観点で研究が進められていくと予想する。

### どうやって可視化する？

可視化の対象となるのは、3次元空間で定義された数値データ(以降ボリュームデータ)である。3次元空間は、多くの場合、格子と呼ばれる小立体に分割され、数値データは、格子の頂点(以降節点)で定義されている。数値データは、大きく分けてスカラー・ベクタ・テンソルに分類することができる。格子の形状には、規則格子、構造格子、不規則格子がある(図-1参照)。規則格子は、各辺が座標軸に平行している直方体である。構造格子は、規則格子を歪めて作成したものである。不規則格子では、対象となる3次元空間を格子分割する上で6面体格子のほかに4面体格子・5面体格子なども用いられる。規則格子のうち全格子が立方体であるものを特にボクセルデータと呼ぶ。

可視化は、連続する2つの変換プロセスと考えると分かりやすい(図-1参照)。第1の変換は、ボリュームデータから幾何形状データへの変換である。幾何形状は、3角形メッシュ、線分、点といった、いわゆる3次元グラフィクスソフトウェアの表示プリミティブを意味する。第2の変換は、幾何形状データから画像データへの変換である。この変換は一般にOpenGLに代表される3次元グラフィクスライブラリの描画関数によって行われる。ボリュームレ

ンダリングでは、ボリュームデータが直接画像データに変換される。

ここでは、スカラーデータの表示技術に焦点を絞って、可視化の基本について説明する。スカラーデータの表示技術は、ボリュームデータを可視化する上で基本となる表示技術である。ベクタデータやテンソルデータは、何らかの操作を施すことによりスカラーデータに変換することができるからである。図-1に数値シミュレーションで計算した人体からの塵埃の分布(スカラーデータ)を3つの可視化手法(断面コンタ、等値面、ボリュームレンダリング)で表示したものを示す。

#### ■断面コンタ表示

スカラーボリュームデータの定義された空間に平面を配置し、その面上でスカラーデータをマッピングすることにより、スカラーデータの定義された平面を作成することができる。この平面上でスカラーデータ値を適切な色データに変換し、その色で塗りつぶす表示手法のことを断面コンタ表示という。スカラーデータからカラーデータへの変換は、あらかじめユーザーがテーブル形式で指定しておく。断面コンタ表示は、ある特定領域におけるスカラーデータの全体的分布を知りたい時に利用される。

#### ■等値面表示

等値面は、スカラーデータの定義されたボリューム内でデータ値が同じである点の集合として定義される。等値面表示技術は、数値シミュレーション結果において、スカラーデータの全体的な分布を知りたい時によく使われる。また、医用画像処理分野では、ある軸に沿って撮影された複数枚のCTスキャンデータから骨、筋肉、臓器の形状を再構成するために用いられる。

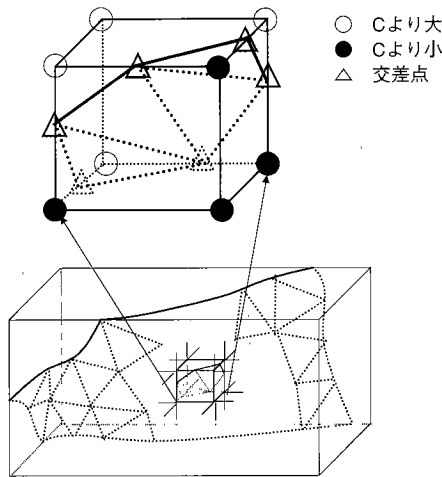


図-2 3角形メッシュで表現された等値面

等値面表示技術では、あるスカラー値Cが与えられた時、全節点において定義されているスカラー値とCとを比較し、その大小で節点を分類する。格子ごとにその稜線の両端の節点を調べ、その分類がお互いに異なっているような場合、その稜線が等値面と交差すると判定する。交差点の座標の計算には、稜線上で線形補間を適用する。このようにして計算された交差点を使っていくつかの3角形を定義する(図-2参照)。この操作をすべての格子に対して行うことにより、等値面を3角形メッシュとして生成することができる。等値面表示において標準的に利用されるマーチングキューアルゴリズムでは、格子単位でのすべての分類について3角形生成パターンをテーブル化している。

等値面表示技術では、原則的に、等値面が交差するかどうかの判定を全格子について行うことになる。N個の格子から構成されるボリュームデータから等値面を計算する場合、等値面と交差する格子の数は $O(N^{2/3})$ である。Nが膨大になった場合、全格子を対象とするような処理はボトルネックとなってしまふ。この問題を解決するために等値面の生成に先立って、補助的なインデックスを作成する手法が提案されている<sup>5)</sup>。

### ■ボリュームレンダリング表示

ボリュームレンダリングでは、ボリュームデータは、発光する一種の密度雲として表現される<sup>2)</sup>。スカラー値に対して、色と不透明度をどのように対応させるかはユーザがテーブルの形(伝達関数)の形で指定する。不透明度は、密度を表現し、その値が高いと対応するスカラー値が強調されることになる。

ボリュームレンダリングでは、視点と各画素を結ぶ半直線(レイ)上でサンプリングポイントを配置する。サンプリングポイントは、レイセグメント(レイのうち隣接する格子面で仕切られた部分)の中心に配置されることが多い(図-3参照)。サンプリングポイントにおいてスカラー値を評価し、その評価値に対して色と不透明度を割り当て、

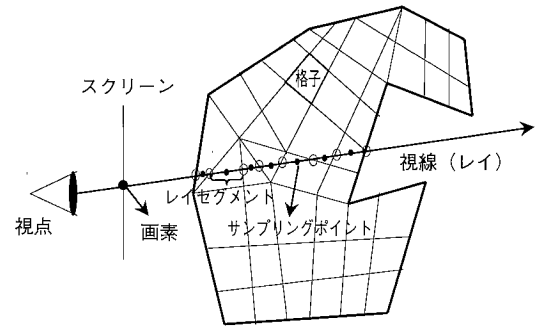


図-3 サンプリングポイントの配置

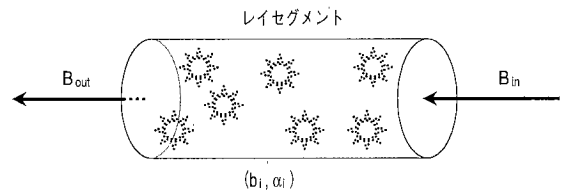


図-4 レイセグメントを通過する光束

それらを加算していき、2次元スクリーン上に投影する。ボリュームレンダリングで用いられている色と不透明度の合成法は、半透明表示法に基づいている。1つのレイセグメントを通過する光束を考えると、i番目のレイセグメントを通過した光束の明るさ $B_{out}$ は、i番目のレイセグメントに侵入した光束の明るさ $B_{in}$ とレイセグメント自体の明るさ $b_i$ および不透明度 $\alpha_i$ から次のように求まる(図-4参照)。

$$B_{out} = b_i \alpha_i + B_{in} (1 - \alpha_i)$$

不透明度 $\alpha_i$ が1であると侵入してきた光束は、完全にブロックされる。0となっているとレイセグメントはまったくの透明体となる。上式を視点からスクリーン上の画素を通りボリュームを通過する光線に沿って、n個のレイセグメントに視点方向から適用していくと、最終的に求まるピクセルの明るさBは、

$$B = \sum_{i=1}^n b_i \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_j)$$

となる。この式は、あるレイセグメントの明るさ $b_i \alpha_i$ が、視点とそのレイセグメントとの間に存在するレイセグメントによって吸収され、減衰しながら視点に $\prod (1 - \alpha_j)$ の割合で到達することを示している。

ボリュームレンダリングの対象となる空間は、格子から構成されているので、この加算計算では、視線上で隣接する格子を次々と探索することになる。この探索には、視線と格子面との交点計算を含むため、全格子面を対象にレイトレーシング計算を行っているといってもよい。

### ボリュームレンダリングを高速に

一般的に全格子面の数は格子の数の数倍となるため、ボリュームレンダリングは莫大な計算時間が必要となる。こ

の計算時間を削減するため、高速化に関するさまざまな試みが行われている。この高速化は、大きく分けて、汎用グラフィクスハードウェアを利用するもの、そして専用ハードウェアによるものに分けることができる。

### ■汎用グラフィクスボードを使って

汎用グラフィクスハードウェアには、イメージ合成のための $\alpha$ ブレンディング機能を持ったものが多い。この機能を使って部分的に半透明な、すなわち各頂点で異なる不透明度を持つ3角形を描画処理することが可能である。与えられたボリュームデータをこのような3角形を用いて近似し、ボリュームレンダリングの高速化を実現する手法が報告されている。

ボリュームデータを構成する各格子をスクリーン上に投影し、そのシルエットを3角形に分割することは比較的容易である。Shirleyらは、このような3角形の各頂点を通過するレイセグメントにおいて、色、不透明度データを計算し、これらをその頂点に割り当てた。彼らは、これら3角形を視線から遠い順に重ね合わせ表示することにより、ボリュームレンダリングの高速化を図る方法を提案した。この手法では、視点が変更されるごとに格子のソート処理を必要とする。

筆者らは、視点からの距離を一定値ごとに増加させながら等距離面を計算し、それらを距離の大きいものから順に重ね合わせることでボリュームレンダリング計算を実現した(図-5参照)<sup>3)</sup>。等距離面は、一種の等値面であり、3角形メッシュで構成される。この3角形の各頂点で評価したスカラ値をもとに色、不透明度をその頂点に割り当てた。この場合、レイセグメントは、レイのうち隣接する等距離面で仕切られた部分となる。Shirleyらの手法で生成される3角形と違って、すべての3角形が視線と垂直となり、不適切な線形補間に起因する歪みは生じない。4年後、同様のアイデアがオハイオ州立大学の研究グループから提案された<sup>6)</sup>。1998年に発表されたボリュームレンダリング用ライブラリOpenGL Volumizerのコアアイデア(<http://kirii.8m.com/volumizer/index.html>)も同様である。

### ■専用ボードの出現

ボリュームレンダリング用専用ハードウェアとしては、Kaufmanらが開発したCubeが有名である。Pfisterらは、このCubeをベースとしたパーソナルコンピュータ向けボリュームレンダリング専用システムVolumeProを開発した。VolumeProシステムは、PCIカード・3Dグラフィックカード・ソフトウェアから構成される。PCIカードには、4つのレンダリングパイプラインを持つvg500チップが搭載されている。vg500チップには、0.35 $\mu$ 技術を用いて3,200,000個の論理トランジスタが実装されており、125MHzで動作する。したがって、毎秒 $4 \times 125 \times 10^6$ の格子データを処理することができる。これは $256^3 \times 30$ にほ

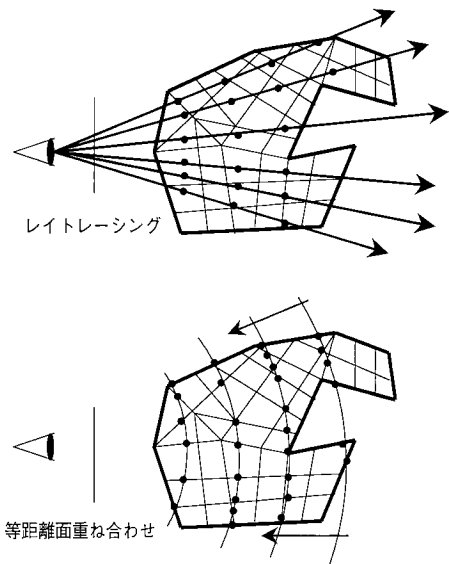


図-5 等距離面重ね合わせによるボリュームレンダリングの実現

ぼ等しい。すなわち、 $256^3$ の規則格子データを毎秒30フレームでレンダリング処理できることになる(今のところ平行投影だけ)。

このボードの実質の生みの親Kaufmanは“Volume Graphics”という考え方を提唱している。この考えでは3次元空間で定義された数値データだけでなく、多角形、線分、点といった幾何形状もボクセルデータとして表示しようとするものである。これに対して、従来の3Dグラフィクスは“Surface Graphics”という。“Surface Graphics”では描画速度が幾何形状の数、すなわちシーンの複雑さに依存するが、“Volume Graphics”における描画時間はシーンの複雑さに依存しない。今後、映像産業分野を含めて“Volume Graphics”が浸透していく可能性が出てきた。

### ベクタデータの可視化

ベクタデータの可視化手法として最も単純で多く用いられているのは、3次元空間でいくつか場所を指定して、その点でのベクタデータの方向・大きさを表す矢印を描かせることである。しかし、この矢印表示手法は、矢印の向きに曖昧性があったり、複数平面を設定した場合、奥行き方向に矢印が重なったりして理解が困難となる場合がある。

ベクタデータを可視化するもう1つの方法として、流線(stream line)表示がよく使われている。流線は、各点での接線がそこでのベクタデータに平行であるような曲線である。この流線計算には、開始点が必要となる。やみくもに開始点を設定して流線計算を行ってもベクタデータの分布を適切に表現することはできない。そこで、適切な開始点を決めることが重要となってくる。与えられたベクタボリュームデータにおいて、ベクタ値がゼロと

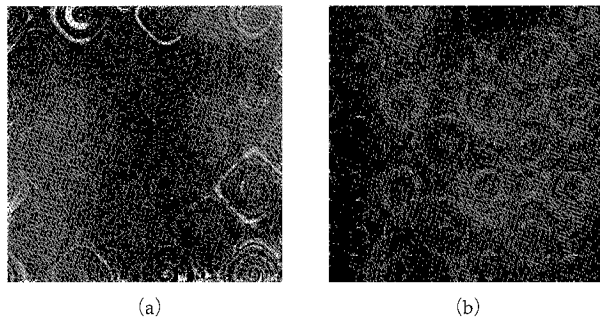


図-6 2次元ベクタ場の可視化

なる点(特異点)は、渦中心等ベクタ場を理解する上で重要な役割を果たす。特異点付近を開始点とするような流線をまとめて表示すると一目でベクタ場を理解できるような画像ができあがる。これをFlow Topology表現と呼ぶ。

2次元ベクタ場の可視化に対しては、冒頭でふれたLIC法が有効である。LIC手法では、ホワイトノイズ画像を入力データとし、ベクタ場から計算される局所流線に沿った畳み込み積分計算を行い、ベクタの方向に対してコヒーレンスのある画像(ベクタボリュームの場合にはボリュームデータを)を出力する(図-6(a)参照)。この結果、ベクタ場の大局的な傾向や複雑な構造を直感的に理解することが可能となる。3次元ベクタデータの場合は、3次元画像(ボクセル)が生成され、表示には、一般にボリュームレンダリングを利用することになる。

実際の流れ場を可視化する場合、適切なトレーサを流していくことにより直観的な流れ場表現を行うことができる。この手法をそのままコンピュータ上で実現するのも有効である。適切な間隔で配置されたドットを拡散物質の投入場所と考え、与えられた速度場をもとにその物質濃度について移流拡散方程式を解き、結果をそのまま画像化したのが図-6(b)である。Beckerらは、流れ場の流入口からテクスチャ座標そのものを流し込み、あるパターン画像をテクスチャマッピングした。

## テンソルデータの可視化

テンソル(2階テンソル)は、古典物理学、特に力学、弾性力学、流体力学、電磁気学などにおいて重要な概念として位置付けられており、数値シミュレーション結果にも出力されることが多い。テンソルデータはそのままの形で評価されることは少なく、いったん、スカラ、ベクタデータに変換された上で表示されることが多い。一般的にテンソルデータはその特性方程式の固有値、固有ベクタに対応する主値や主軸ベクタに変換されることが多い。

有限要素法による構造解析では、解析結果として応力テンソルが出力されることが多い。構造物の強度評価の指標として使われる材料の降伏点は応力テンソルの主値

となっているので、構造物の表面または断面で応力テンソルの主値のカラー表示を行うことが多い。

応力テンソルの主軸ベクタを利用した可視化手法として超流線表示法が提案されている。超流線表示法では、テンソルデータから3つの主軸を求め、そのうち対応する主値が最大となる主軸を選択する。すべての節点でこの主軸ベクタを計算し、ベクタデータ可視化手法を使って表示を行う。外力のかかっている点から超流線を表示することによりかけられた力が物体の中でどのように伝播していくかが明らかになる。

MRI画像を加工することによって得られる拡散テンソルボリュームデータは、生体組織における水分子の移動容易な方向を表現する。ユタ大学のKindlmannらは、拡散テンソルの主軸ベクタを用いて、移動容易な方向の観点で、生体組織を“直線/平面/等方”的の3つに分類した。彼らは、ボリュームレンダリングを利用し、水分子の移動容易な方向が“直線/平面/等方”的となる領域を強調表示した。

## 大規模化するデータ

米国のAccelerated Strategic Computing Initiative(以降ASCI)プロジェクトでは、現在超並列計算機を使って、10億個程度の格子を使うような大規模シミュレーションが行われている。このような大規模シミュレーションの出力する膨大なデータをいかに効率よく可視化するかに大きな関心が寄せられている。

大規模シミュレーション結果の可視化処理の特徴は、ボリュームデータが単一ノードの物理メモリに収まりきらないということである。このため可視化処理において、ややもすれば、ディスク入出力を伴うスワッピングが頻発することになる。ディスク入出力においてデータアクセスに要する計算クロック数は、物理メモリにおけるそれに比べて桁違いに大きい(10,000倍程度)ため、計算時間の増大をもたらす。Coxらは、Unixのシステムコールmmap()を使い、ボリュームデータファイル全体を物理メモリにロードするのではなく、必要になった時点で初めて物理メモリにページ単位でロードされるようにした。ただ、ボリュームデータが仮想メモリを超えてしまうような場合には無力である。この手法は商用のDBMSシステムに用いられてきたものである。

Schroederらは、ストリーミング技術を使い、物理メモリだけを使って可視化処理を行うことを提案した。彼らは可視化処理で使用される計算機の物理メモリや処理アルゴリズムを調べて、可視化処理すべての局面でスワッピングが起らないようなサイズを算出し、与えられたボリュームデータをこのサイズに分割した。データフロー型可視化システムにこの手法を実装し、ストリーミング技術を使わない場合に比べ、40倍程度の計算速度向上がもたらされるとの報告を行っている。

## 広域化する可視化

数値シミュレーション技術の向上の結果、その利用分野が拡大すると同時にその解析結果の恩恵を受ける人も増えている。たとえば、営業担当者が解析結果を使って、顧客に商品をプレゼンテーションするケースも出始めていると聞く。WWWのサーバに解析結果ファイルを格納し、これをインターネットを介して可視化する仕組みを構築すれば、多くの人々が、WWWのクライアント端末を使ってその恩恵にあずかることが可能になる。

(株)ソフトウェアクレイドルは、WWWサーバに格納された計算流体力学シミュレーション結果をクライアント側で表示するためのプロトタイプシステムを開発した。ローエンドクライアント端末においても実用的な「可視化」を実現するには、サーバ側でできる限り幾何形状のデータ量を削減しておく必要がある。このため、本プロトタイプシステムでは、可視化処理を断面コンタ表示に限定し、データ付き3角形メッシュを大きな1枚のテクスチャ付き4角形で表現する手法<sup>7)</sup>を採用している。

図-7はシステムの概要図である。クライアントではWEBブラウザ経由で現在サーバで保存されているボリュームデータの一覧を表示する。ユーザは、Javaアプレットをダウンロードし、断面の指定などに必要とされる可視化パラメータを入力する。入力されたパラメータは、サーバ側でロードされているServletに渡される。Servletは、Java Native Interface (JNI) 経由でC言語で開発された可視化用Native Moduleを呼び出す。このModuleは、ボリュームデータからテクスチャ付き4角形をVRML形式で生成する。生成されたVRMLファイルは、クライアントのブラウザ上で実行されているJavaアプレットに転送され、そこからExternal Authoring Interface (EAI) を用いてVRMLビューワに受け渡される。

## ボリュームコンテンツの普及

インターネットのブロードバンド化が進んでくるとデジタルコンテンツの配信(デリバリ)が盛んになってくる。さらに、Kaufmanの提唱する“Volume Graphics”が定着してくるとコンテンツデリバリの対象としてボリュームデータが新たに加わることになるであろう。

WWWサーバに格納された多くのボリュームデータの中から望みのボリュームデータにアクセスするためには、コンテンツの内容記述を行っておくことが重要である。コンテンツの内容記述として、最も単純なのは、ボリュームデータを低解像度でボクセル化し、サムネイルボクセルを一覧形式で表示することであろう。次に考えられる内容記述としては、ボリュームデータに関する何らかの特徴である。3次元スカラ場での極点といった特異点であれば自動生成することが可能である。さらに、数値シミュレーションでモデル化される物理現象の意味(温度の

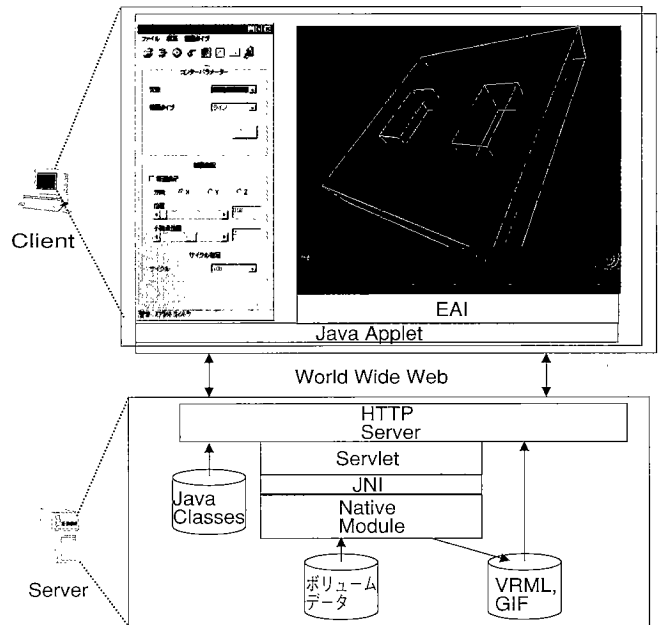


図-7 WWWベースの可視化システム実現例

吹きだまりが時間とともにx方向に移動するなど)を自動抽出することができれば、それをもとにナレーション付きのアニメーションを生成することも可能となる。

ASCIプロジェクトのメンバはボリュームデータの物理的な管理に忙殺され、データから有効な知見を抽出するという本質的な仕事に時間を費やすことが困難であった。この問題を解決するために彼らは、Scientific Data Management (以降SDM。 <http://www.ca.sandia.gov/asci-sdm/cgi-bin/sdmframedisplay.cgi/asci-sdm/index2.html>を参照)というプロジェクトを開始した。このSDMプロジェクトでは、メタデータ(ボリュームデータの内容を記述したデータ)の生成とその管理に焦点を当てた活動を行っている。今後の可視化研究の1つの方向性を示唆するものといえよう。

### 参考文献

- 1) McCormick, B. H., DeFanti, T. A. and Brown, M. D. (ed.): Visualization in Scientific Computing, Computer Graphics, Vol.21, No.6 (1987).
- 2) Sabella, P.: A Rendering Algorithm for Visualizing 3D Scalar Fields, Computer Graphics, Vol.22, No.4, pp.51-58 (1988).
- 3) Koyamada, K., Uno, S., Doi, A. and Miyazawa, T.: Fast Volume Rendering by Polygonal Approximation, Vol.15, No.4, pp.535-544 (1991).
- 4) Cabral, B. and Leedom, L.: Imaging Vector Fields using Line Integral Convolution, Computer Graphics, Vol.27, pp.263-272 (1993).
- 5) Itoh, T. and Koyamada, K.: Automatic Isosurface Propagation by Using an Extrema Graph and Sorted Boundary Cell Lists, IEEE Transactions on Visualization and Computer Graphics Vol.1, No.4, pp.319-327 (1995).
- 6) David, M., Reed, R. Y., Asish, L., Po-Wen, S. and Naeem, S.: Hardware Assisted Volume Rendering of Unstructured Grids by Incremental Slicing, Proceedings of the 1996 Symposium on Volume Visualization, pp.55-62 (1996).
- 7) 小山田耕二: VRMLを用いたCFD解析結果の可視化技術, 日本機械学会論文集B, Vol.66, No.646, pp.1259-1265 (2000).

(平成12年9月29日受付)