

AVSを手軽に使うための可視化システム

“VisPlus”の紹介

名古屋大学情報基盤センター

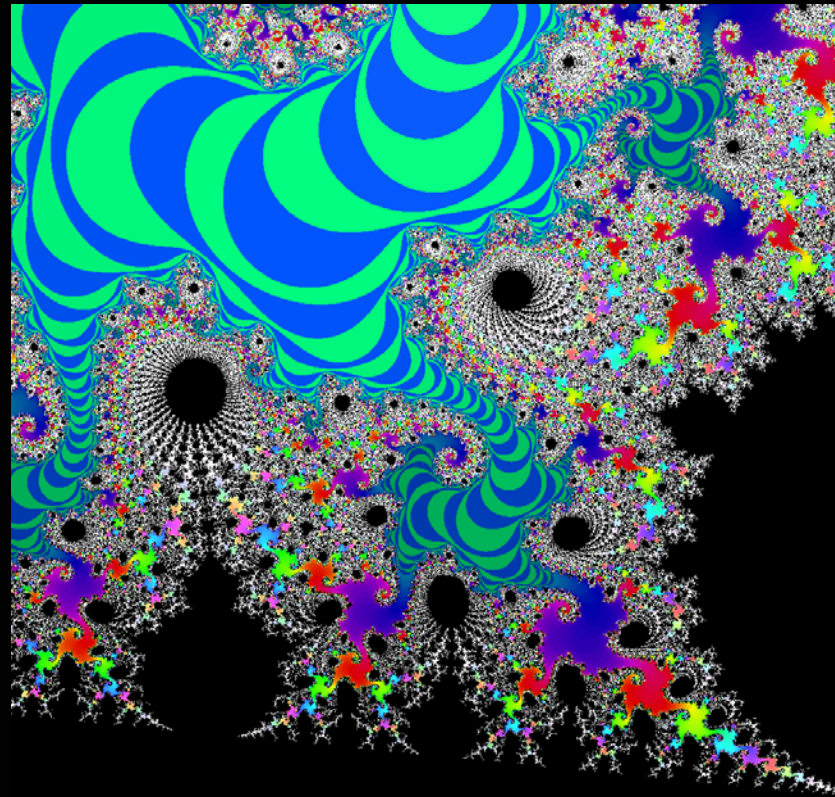
高橋 一郎



# アウトライン

---

- ◆ AVSとは
- ◆ VisPlusとは
  - ・ Vツール
  - ・ Vアプリ
  - ・ ユーティリティ
- ◆ 事例紹介
- ◆ デモ
- ◆ 今後の開発予定



# AVSとは

---

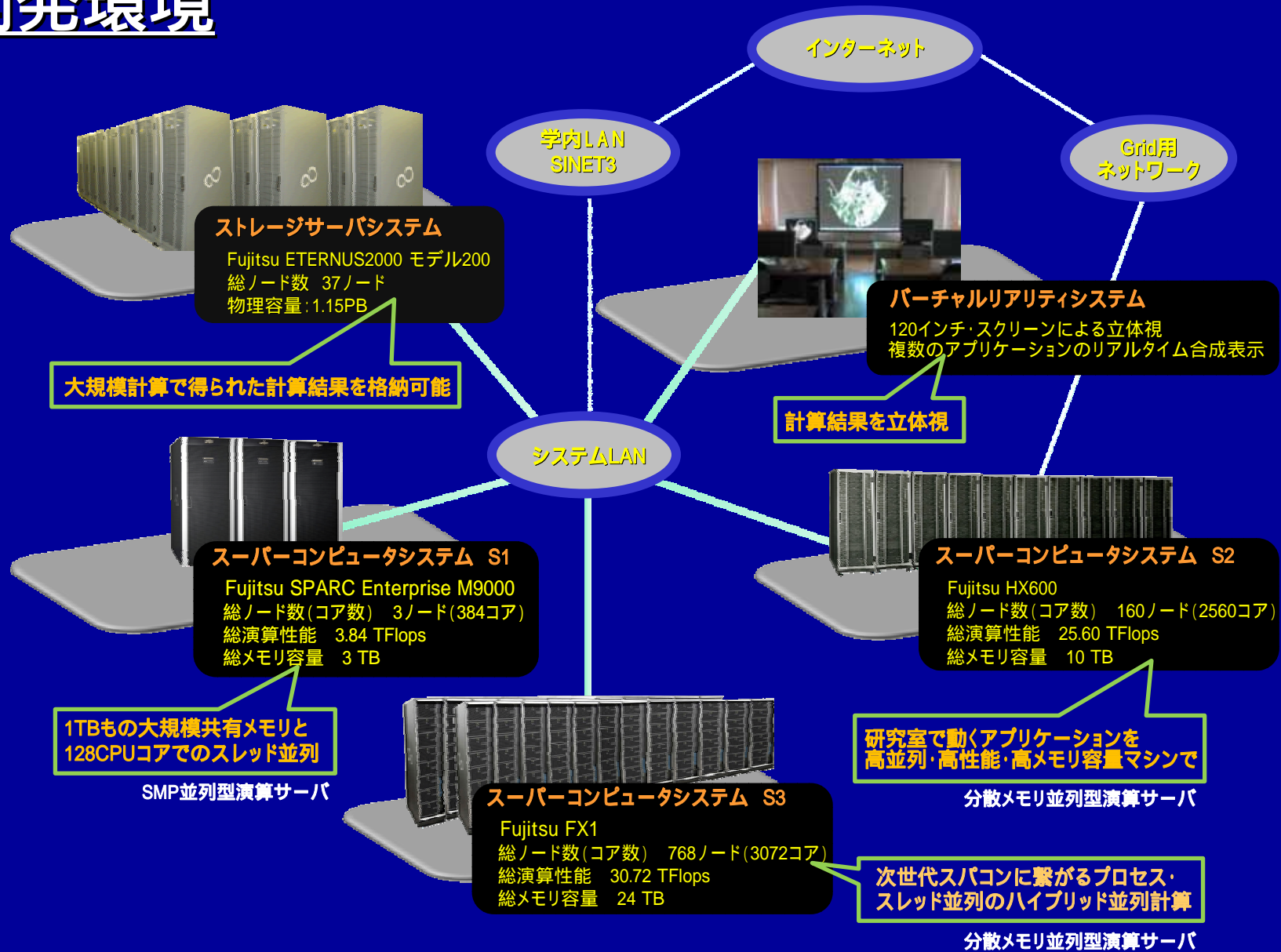
## グラフィックス・アプリケーションの開発 / 可視化ツール

### AVS/Express シリーズ

Viz , Developer , PST, PCE , MPE

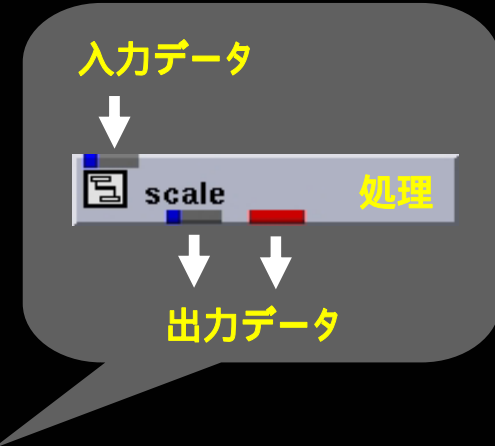
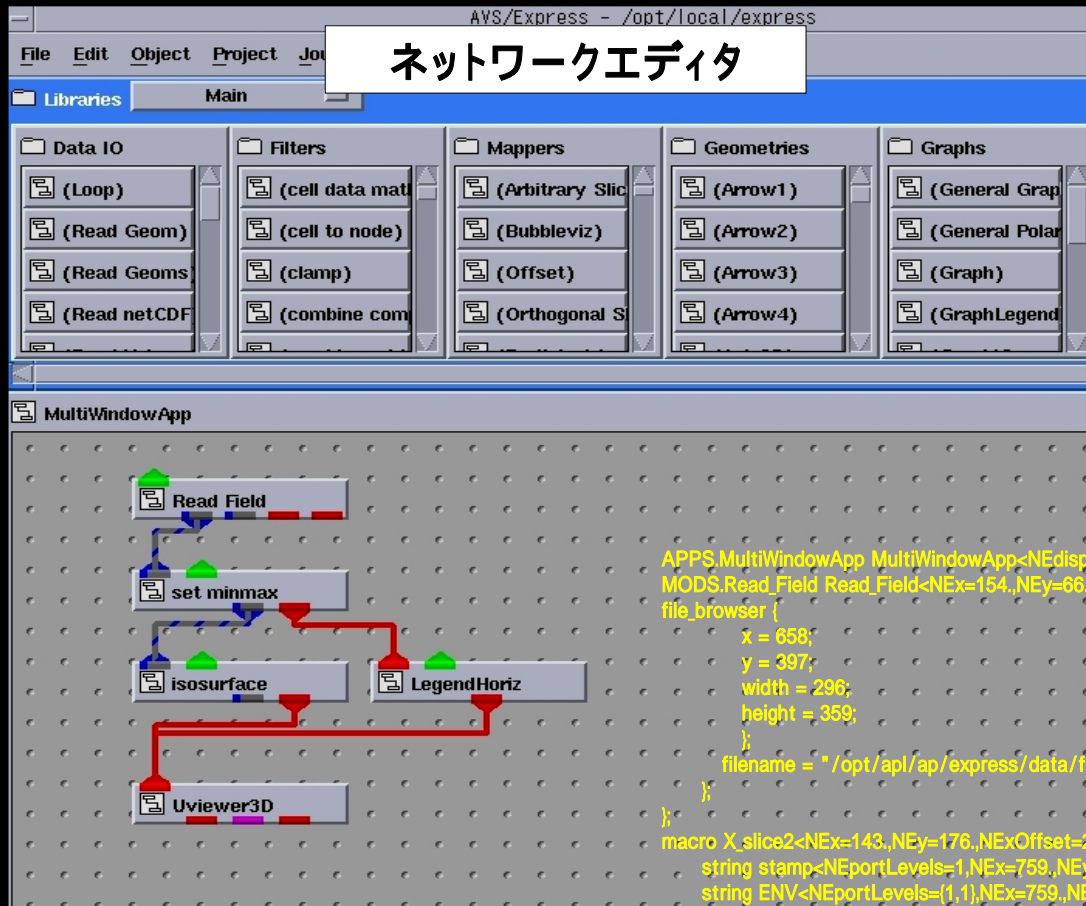
- ✧ アプリケーションの開発機能(GUIも含む)
- ✧ 表現手法やディスプレイへの表示機能が豊富
- ✧ 画像処理機能(保存, キャプチャ等)
- ✧ オープンプラットフォームで動作

# 開発環境



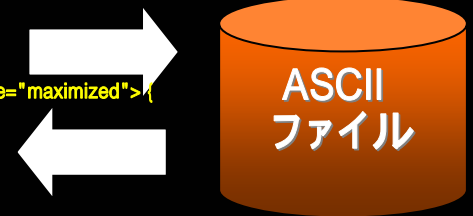
# プログラミング

モジュール：機能を部品化



AVSのカーネルの中間言語  
“V言語”で記述

```
APPS.MultiWindowApp MultiWindowApp<NEdisplayMode="maximized"> {  
  MODS.Read_Field Read_Field<NEx=154.,NEy=66.> {  
    file_browser {  
      x = 658;  
      y = 397;  
      width = 296;  
      height = 359;  
    };  
    filename = "/opt/apl/ap/express/data/field/hydrogen.fld";  
  };  
}  
  
macro X_slice2<NEx=143.,NEy=176.,NEOffset=204.0803223,NEOffset=223.2580261,NEscalingFactor=0.8408735991> {  
  string stamp<NPortLevels=1,NEx=759.,NEy=-198.> = "07/07/07";  
  string ENV<NPortLevels=(1,1),NEx=759.,NEy=-132.>[] => str_array(getenv("MACHINE"), "c");  
  int PSW<NPortLevels=(1,1),NEx=759.,NEy=-165.> => ((strcmp(ENV[0], "p") == 0) + 1);  
  int XF30<NPortLevels=(1,1),NEx=759.,NEy=-99.> => switch(.PSW,30,24);  
  int XS55<NPortLevels=(1,1),NEx=759.,NEy=-66.> => switch(.PSW,55,33);  
  int DNO<NPortLevels=1,NEx=759.,NEy=0.> => _X_option.sliderD.value;  
  float VAL<NPortLevels=1,NEx=759.,NEy=33.> => (1. / pow(10.,(DNO + 1)));  
  float Dsize<NPortLevels=(1,1),NEx=759.,NEy=66.> => ((abs(.MAX[.ANO1]) + abs(.MIN[.ANO1])) / 10.);
```



# VisPlusとは

---

VisPlusシステムは AVSを使いこなすために開発した  
フリーソフトウェア

- ◆ Vツール

AVSのV言語で開発した高機能モジュール

- ◆ Vアプリ

AVSのV言語で開発したアプリケーションプログラム

- ◆ ユーティリティ・プログラム

入力データの生成や外部プログラムとの連携を  
とるためのプログラム

# VisPlusシステム

解析	データ
実験	データ
計測	データ
医療画像	データ
イメージ	データ
形状	データ

データ入力

## AVS Express

ユーティリティ  
プログラム

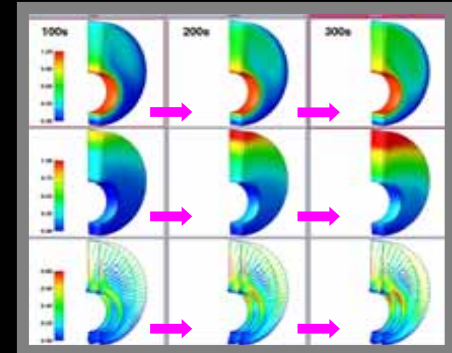


データ出力

可視化

外部プログラム

数値データ(物理量,座標値)  
画像データ  
動画データ (mpeg,AVI)  
3D動画データ(GFA)  
3D形状データ (STL)



マルチビュー



マルチモニター



大画面 & 立体視

# 動作環境

---

## ◆ Vツール 及び Vアプリ

AVS Expressが必要

オープンプラットフォームで動作

## ◆ ユーティリティ・プログラム

Fortranコンパイラが必要

UNIX OS , Linux OS上で動作



# ユーティリティとは

---

## ◆ AVSにデータを読み込むための入力支援プログラム

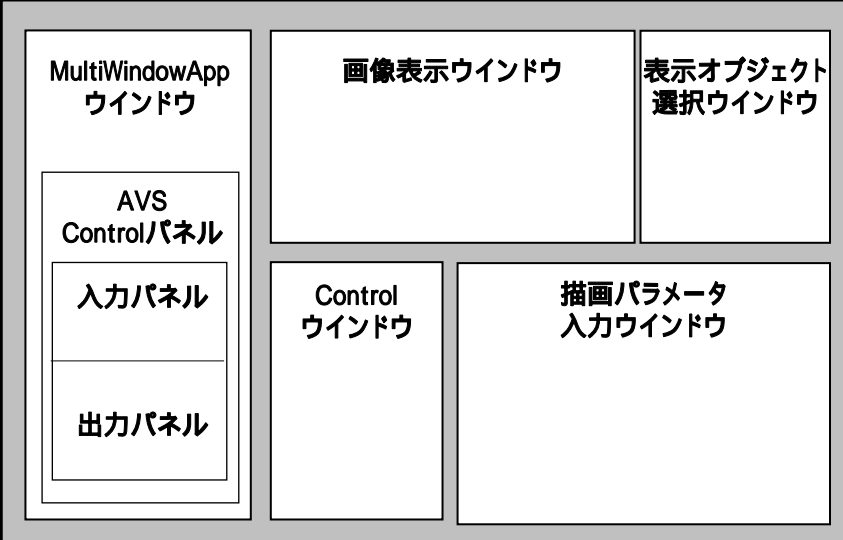
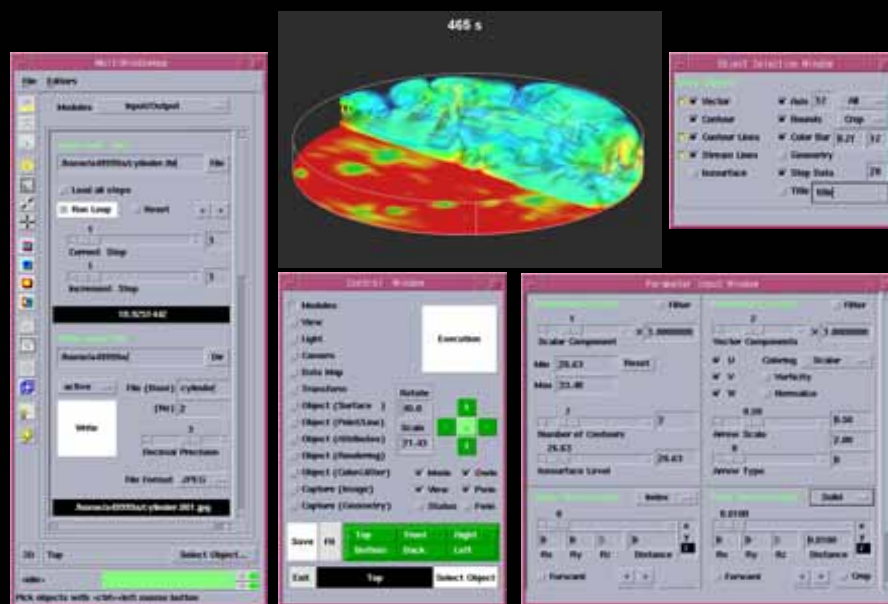
- ・ AVSヘッダーファイル生成
- ・ CSVデータをもとに入力データ生成
- ・ ISVアプリケーションの解析結果をもとに入力データ生成  
ISVアプリケーションの利用支援
- ・ スライス画像データ(DICOM, RAW)をもとに入力データ生成

## ◆ AVSの出力データを外部で利用するための支援プログラム

- ・ 成分データ, 格子座標データの抽出
- ・ 形状データの抽出・形式変換(STLなど)
- ・ フライスルー可視化のためのルートデータの作成

# Vアプリとは

- ◆ V言語で記述したアプリケーションの雛形の集まり
- ◆ ノンプログラミングで商用アプリケーションのように利用できる
- ◆ カスタマイズが容易、移植性、オープンプラットフォームで動作
- ◆ 可視化結果を保存する機能がある

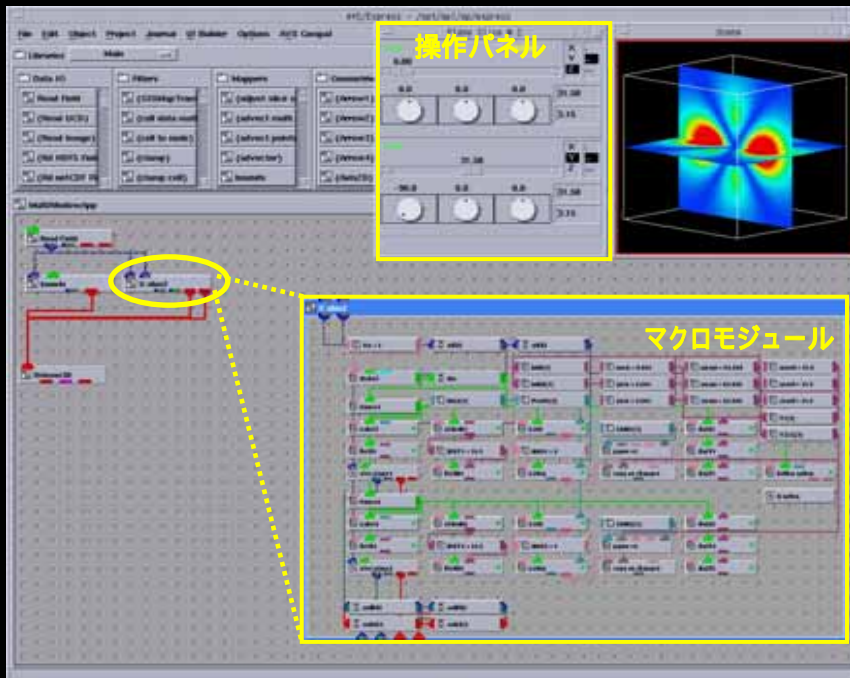


# Vツールとは

モジュール数 : 150  
種類 : 60

目的: 容易なプログラミング

- ◆ V言語で記述した高機能モジュールの集まり
- ◆ カスタマイズが容易、移植性、オープンプラットフォームで動作
- ◆ 複数データ, 複数成分データ, 複数領域, 多視点,  
複数の3次元描画ウィンドウ(立体視対応)が扱える

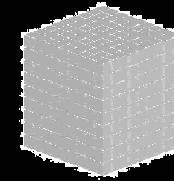


```
APPS.MultiWindowApp MultiWindowApp<NEdisplayMode="maximized"> {  
  MODS.Read_Field Read_Field<NEx=154.,NEy=66.> {  
    file_browser {  
      x = 658;  
      y = 397;  
      width = 296;  
      height = 359;  
    };  
    filename = "/opt/apl/ap/express/data/field/hydrogen.fld";  
  };  
};  
macro  
  X_slice2<NEx=143.,NEy=176.,NExOffset=284.0803223,NEyOffset=223.2580261,NEscaling  
  Factor=0.8408735991> {  
  string stamp<NEportLevels=1,NEx=759.,NEy=-198.> = "07/07/07";  
  string ENV<NEportLevels=(1,1),NEx=759.,NEy=-132.>[] =>  
  str_array(getenv("MACHINE"),"c");  
  int PSW<NEportLevels=(1,1),NEx=759.,NEy=-165.> => ((strcmp(ENV[0],"p") == 0) + 1);  
  int XF30<NEportLevels=(1,1),NEx=759.,NEy=-99.> => switch(.PSW,30,24);  
  int XS55<NEportLevels=(1,1),NEx=759.,NEy=-66.> => switch(.PSW,55,33);  
  int DNO<NEportLevels=1,NEx=759.,NEy=0.> => .X_option.sliderD.value;  
  float VAL<NEportLevels=1,NEx=759.,NEy=33.> => (1. / pow(10.,(DNO + 1)));  
  float Dsize<NEportLevels=(1,1),NEx=759.,NEy=66.> => ((abs(.MAX[.ANO1]) +  
  abs(.MIN[.ANO1])) / 10.);  
  }  
};
```

# モジュールの種類

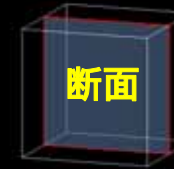
- ◆ 入力モジュール
- ◆ フィルターモジュール
- ◆ 作図モジュール
- ◆ ビューワモジュール
- ◆ 出力モジュール
- ◆ ファイル名生成モジュール
- ◆ 制御モジュール
- ◆ フライスルーモジュール

データの読み込み



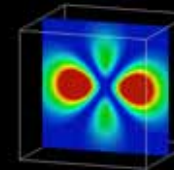
3次元データの読み込み

データの選択 / 加工 / 変換



2次元データの切り出し

作図  
表現手法の選択



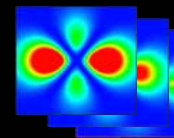
コンター図作成

画像表示  
ビューワ



ディスプレイ表示

可視化結果の保存

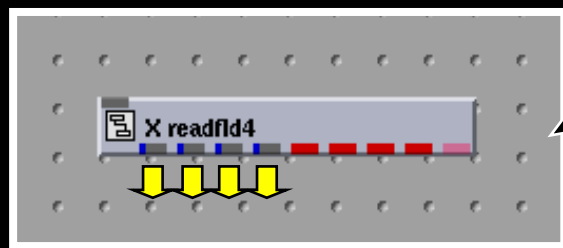


画像データ保存  
数値データ保存

3次元データの断面コンター表示の手順

# 「時系列データを読み込む」モジュール

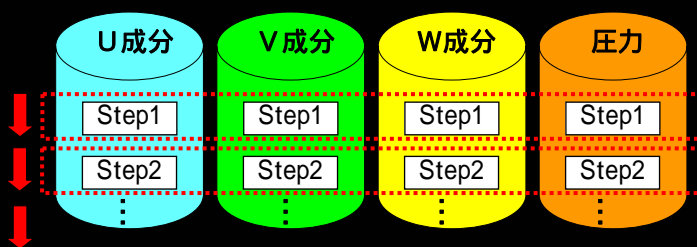
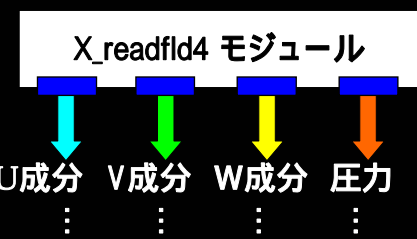
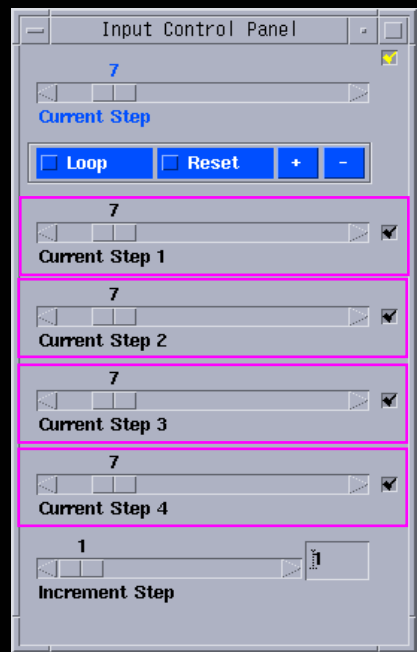
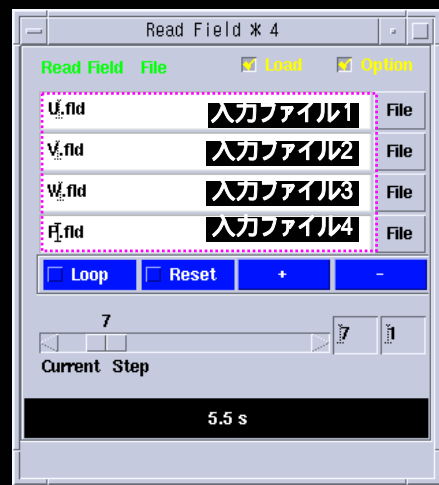
## X\_readfld4 モジュール



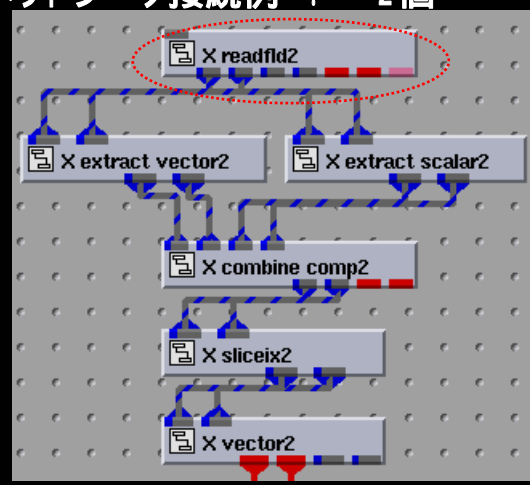
同期・非同期操作

4個の時系列データからステップデータを読み込む。

## X\_readfld4操作パネル

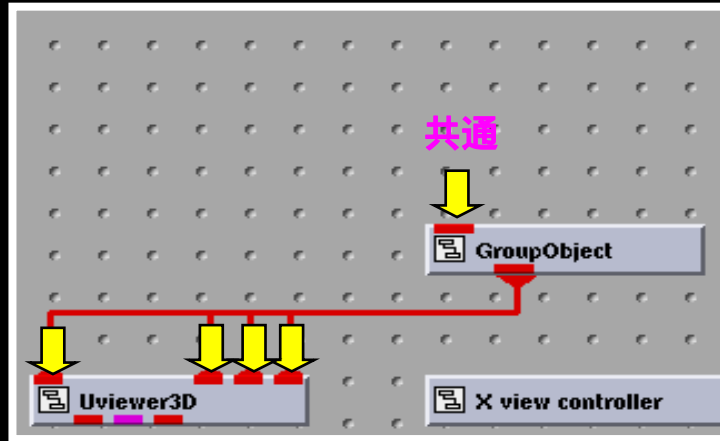


ネットワーク接続例 : 2個

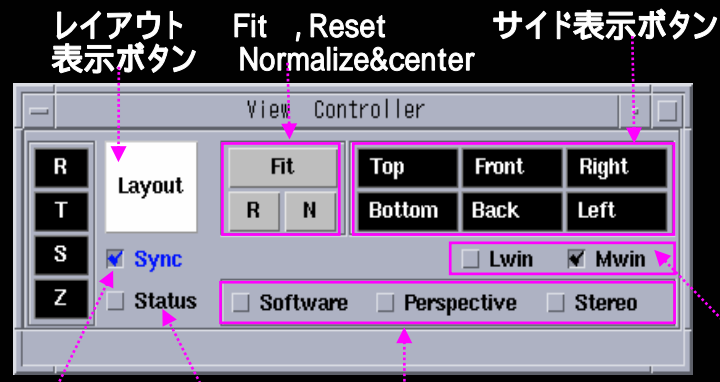


# 「マルチビューモジュール」

ネットワーク接続例 :

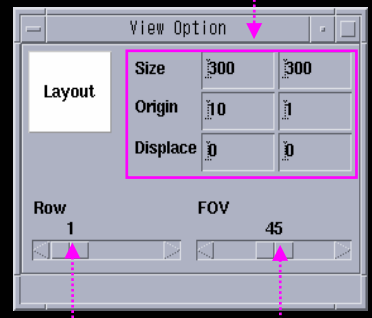


複数の3次元の描画ウィンドウ(シーン)を利用するためのモジュールで、幾何変換操作の同期・非同期の選択や、ウィンドウのレイアウトが行える。立体視にも対応。



同期操作ボタン  
レイアウト表示ボタン  
Fit, Reset Normalize&center  
サイド表示ボタン  
ソフトウェアレンダリングボタン  
透視投影表示ボタン  
ステレオ表示ボタン

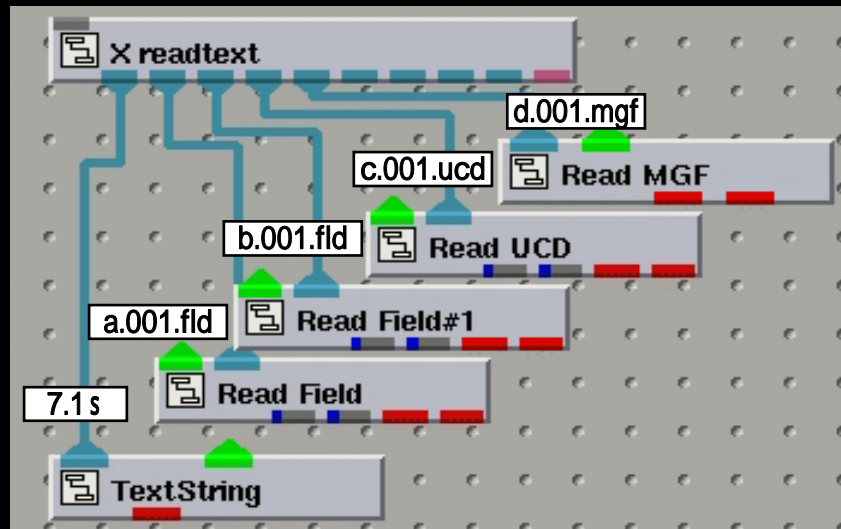
ウィンドウサイズ, Window1の位置, ウィンドウの重なり具合



レイアウト行数  
視野角度(透視投影)  
Lwin: View Optionウィンドウ  
Mwin: MultiWindowAppウィンドウ

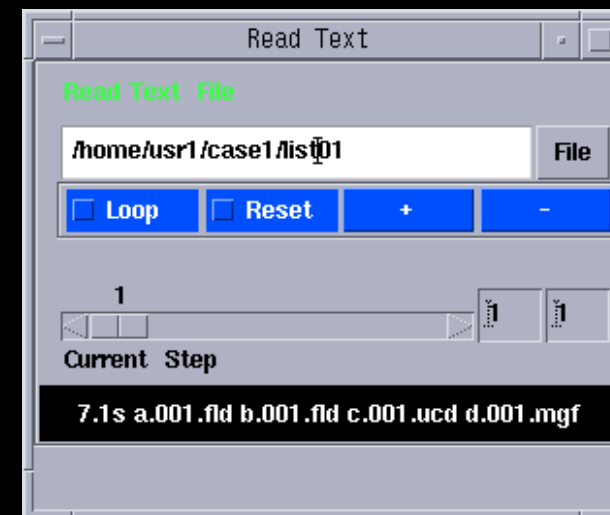
# 「CSVデータから項目データを読み込む」モジュール

ネットワーク接続例 :



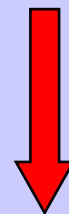
ファイル上に記録されている  
連番付きデータを行単位に  
読んで分解し、出力ポート  
に供給する。  
様々な形式のデータを組み  
合わせて読むことができる。

操作パネルの使用例 :



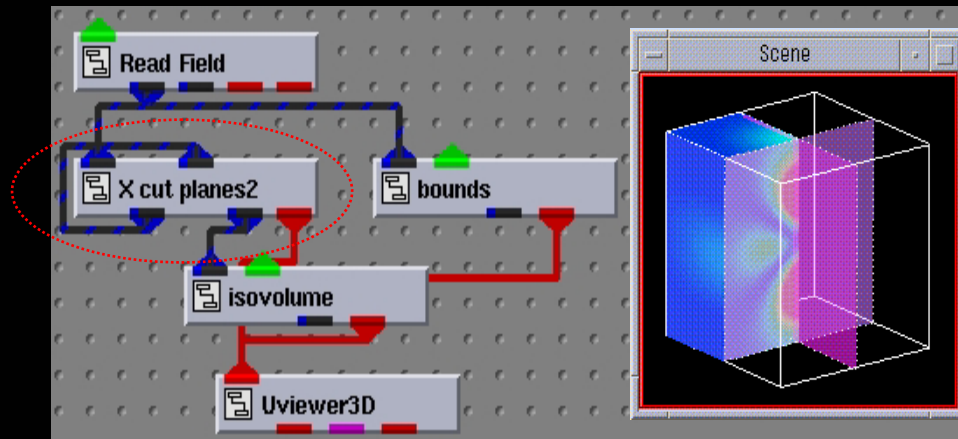
< CSVデータの内容 >

7.1s	a.001.fld	b.001.fld	c.001.ucd	d.001.mgf
7.2s	a.002.fld	b.002.fld	c.002.ucd	d.002.mgf
7.3s	a.003.fld	b.003.fld	c.003.ucd	d.003.mgf
⋮	⋮	⋮	⋮	⋮
項目 1	項目 2	項目 3	項目 4	項目 5



# 「面形状を使ってデータ領域を切り出す」モジュール

ネットワーク接続例 :



## 非同期操作モジュール

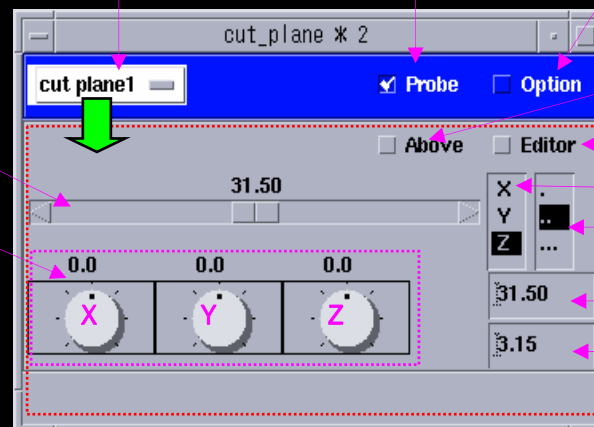
1回目に切り出したデータを、再度、2回目の入力ポート入力

操作パネルの使用例 :

操作パネル選択メニュー    プローブ    オプション  
(操作パネルの切り替え)    表示/非表示ボタン    パネル表示ボタン

カレントスライダー  
軸上の位置 (Distance)

スライス断面の回転操作  
ダイヤル



対称領域指定ボタン  
(面のの上側|下側)

TransformEditor表示ボタン  
スライス軸

軸上のスライス位置  
(min,center,max)

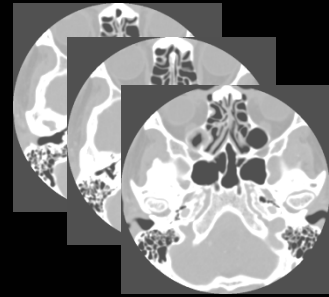
軸上の位置 (Distance)

カレントスライダーの刻み値



# 事例紹介

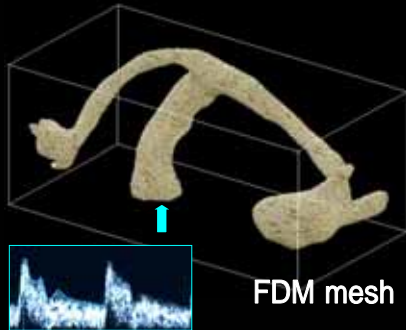
# 脳動脈瘤の血流解析シミュレーション



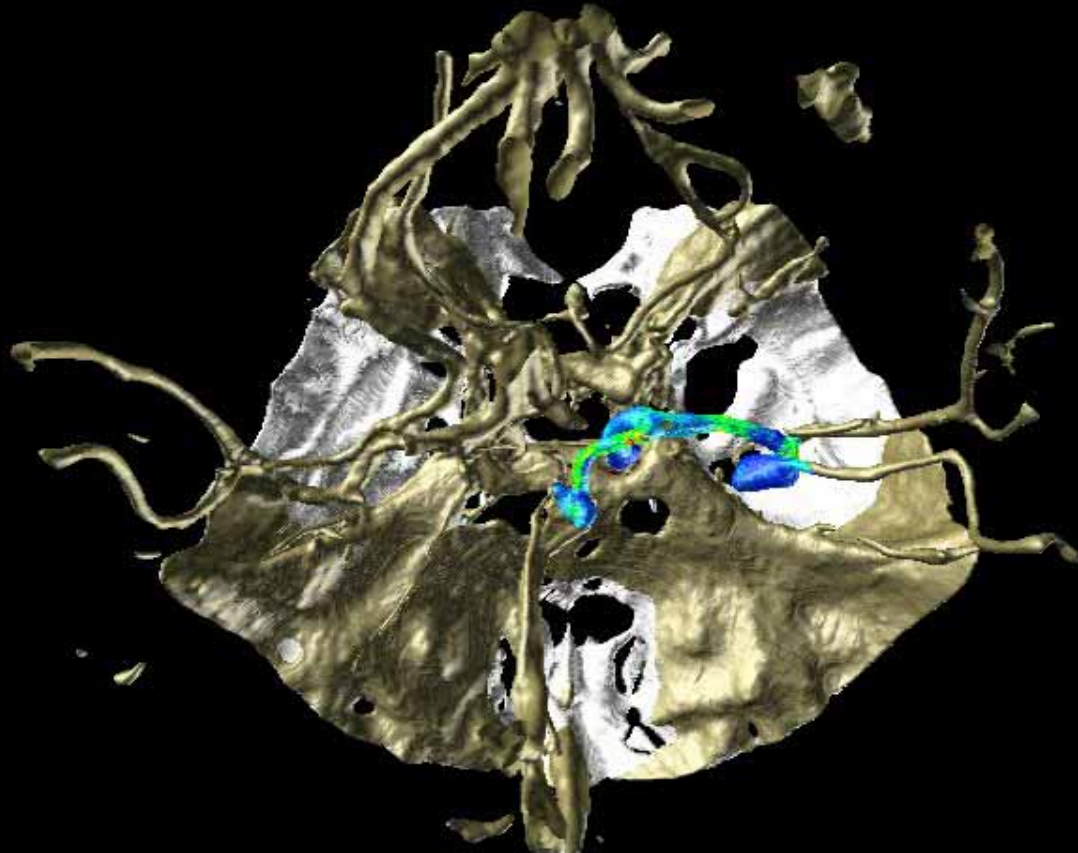
Slice Data



Voxel Data



FDM mesh



Visualization of results

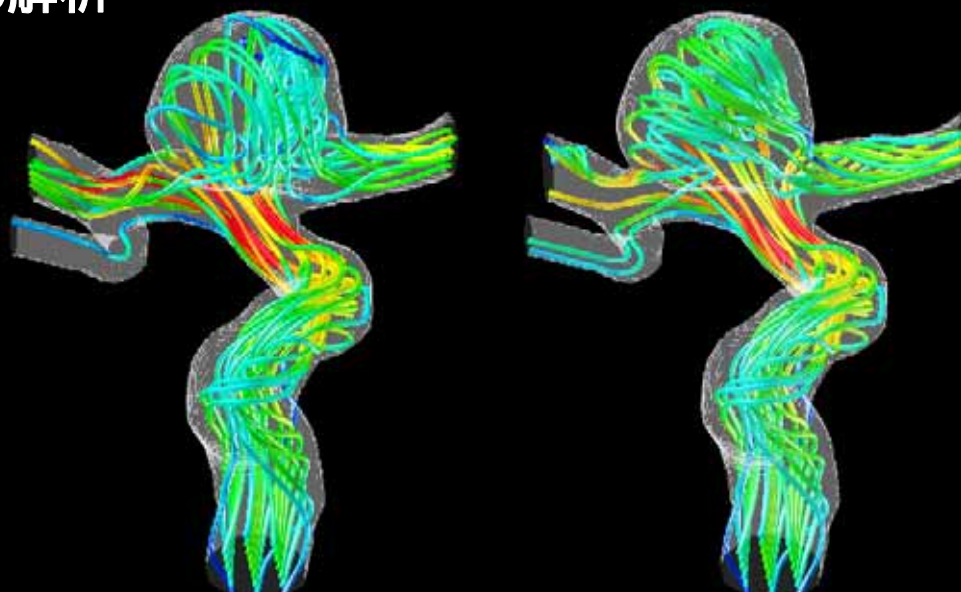
## 2つの時系列データの比較表示例

描画ウィンドウ数: 1

1つの描画ウィンドウに複数の図をレイアウト表示して、流れの比較を確認できる。



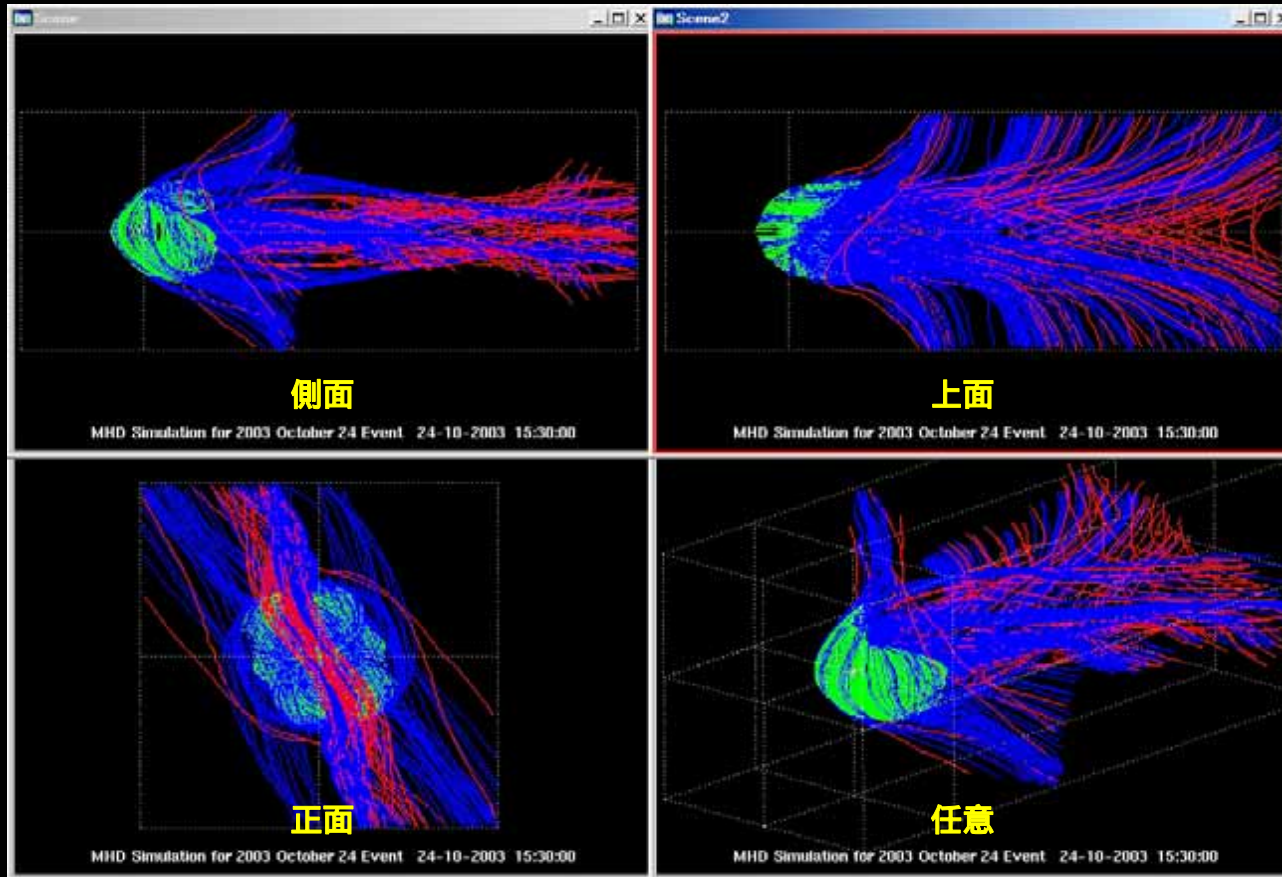
気管支分岐管モデルの流れ解析



脳動脈瘤の流れ解析(瘤部を移動)

# 多視点を使った可視化例

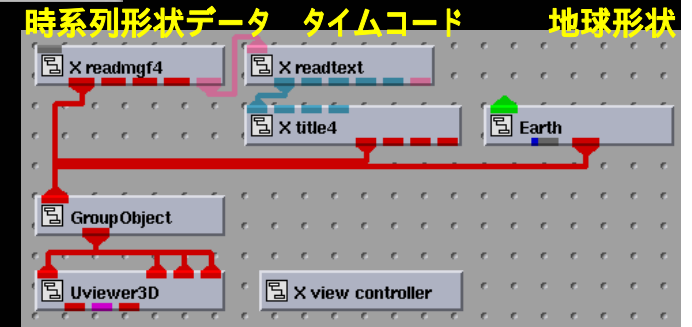
描画ウィンドウ数: 4



4つの描画ウィンドウに幾何変換操作を施して眺める方向を決め、時系列データを連続的に表示して太陽風によって複雑に変化する地球磁場と惑星間空間磁場の構造を確認できる。

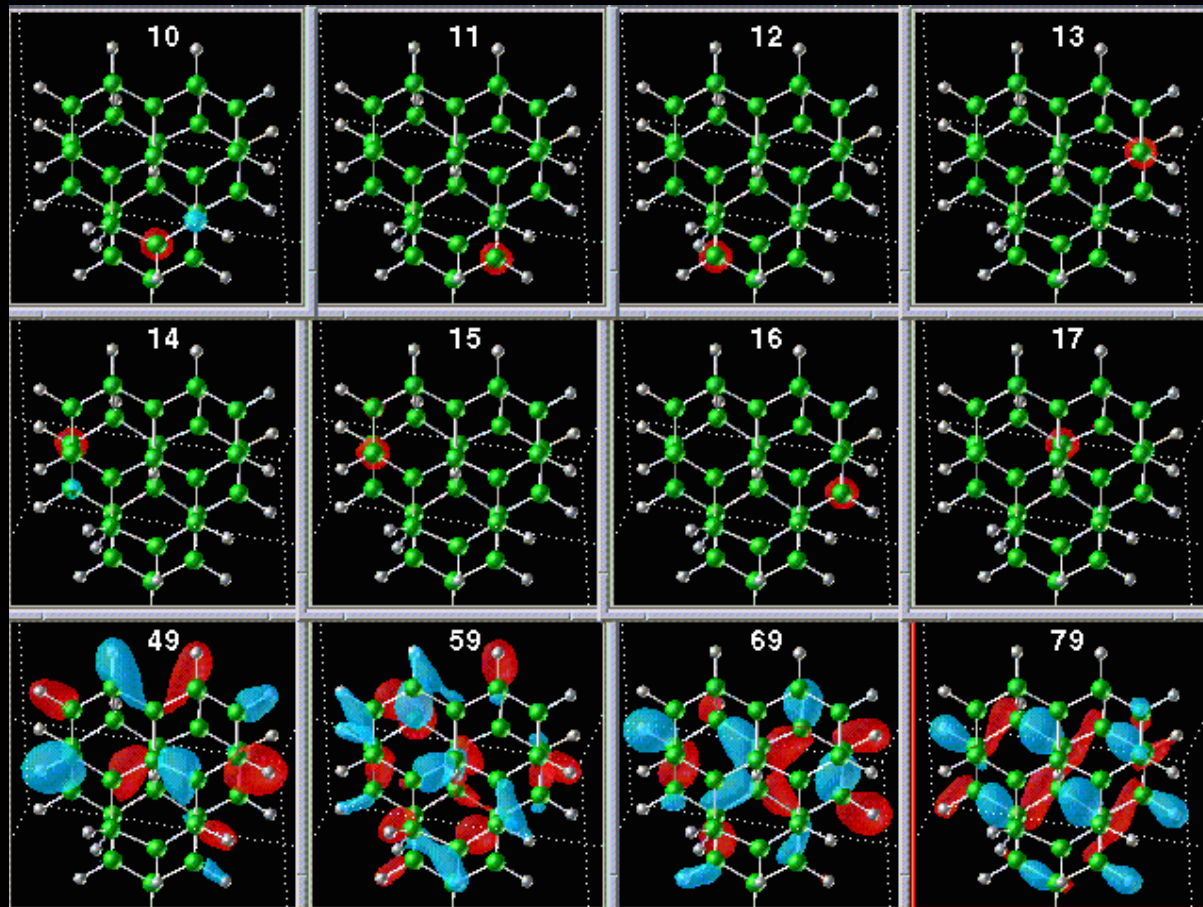
名古屋大学太陽地球環境研究所 荻野竜樹氏 提供

## 太陽風と地球磁気圏の相互作用のシミュレーション

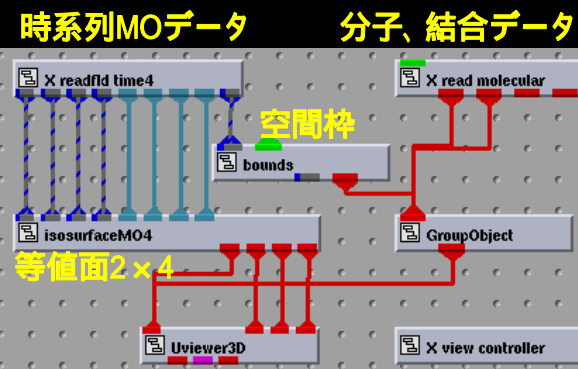


# 複数データの比較表示例

描画ウィンドウ数: 12



100を超えるの分子軌道を連続的に可視化し、短時間で解となる分子軌道を見つけることができた。

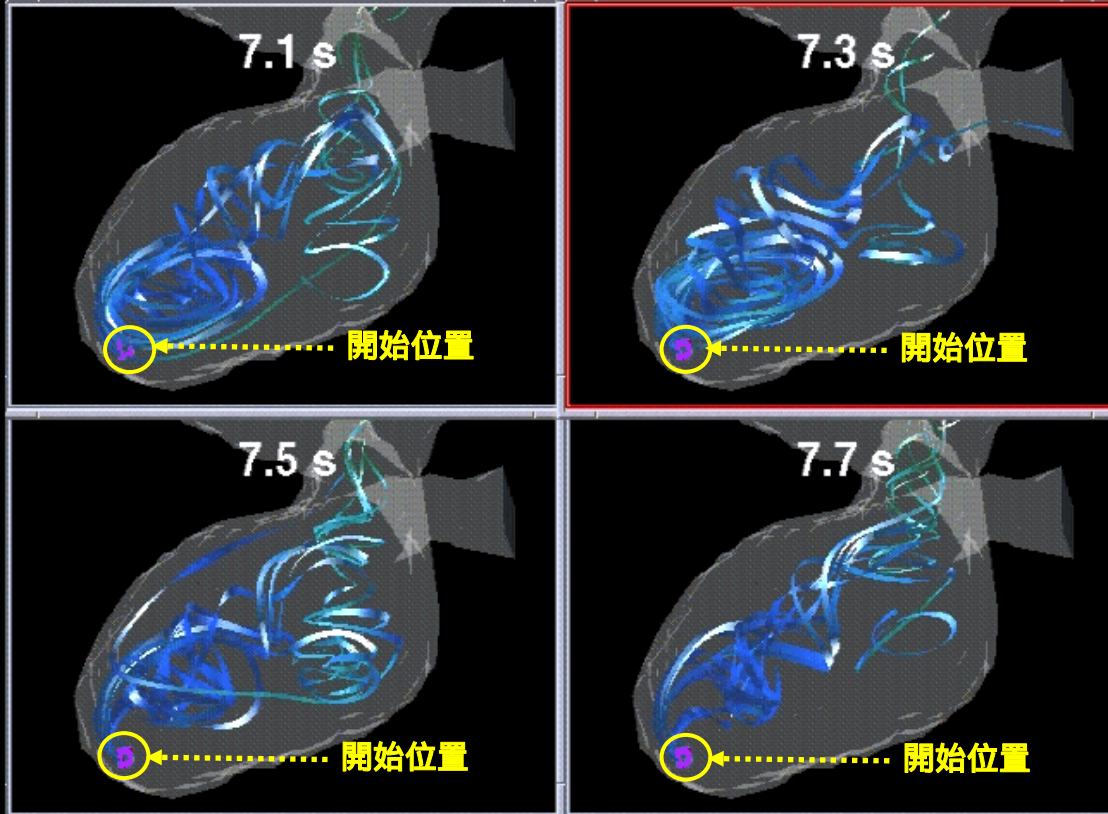


名古屋大学工学研究科 宮部祐三子氏 提供

グラファイトの分子軌道シミュレーション (Gaussianの解析結果)

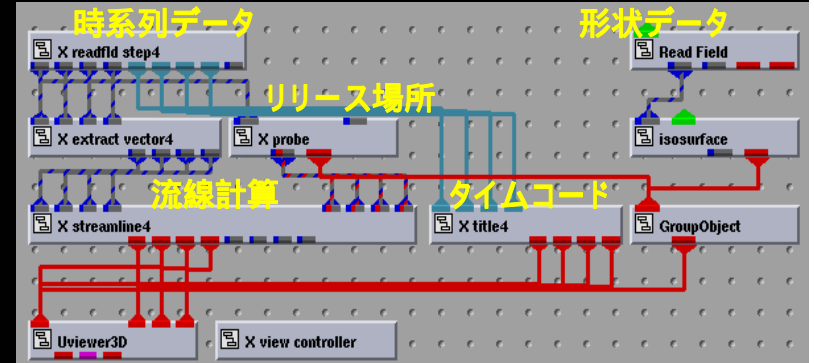
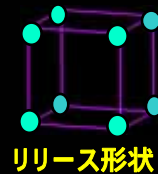
# 流線表示の同時操作例

同時操作数, 描画ウィンドウ数, 可視化データ数: 4



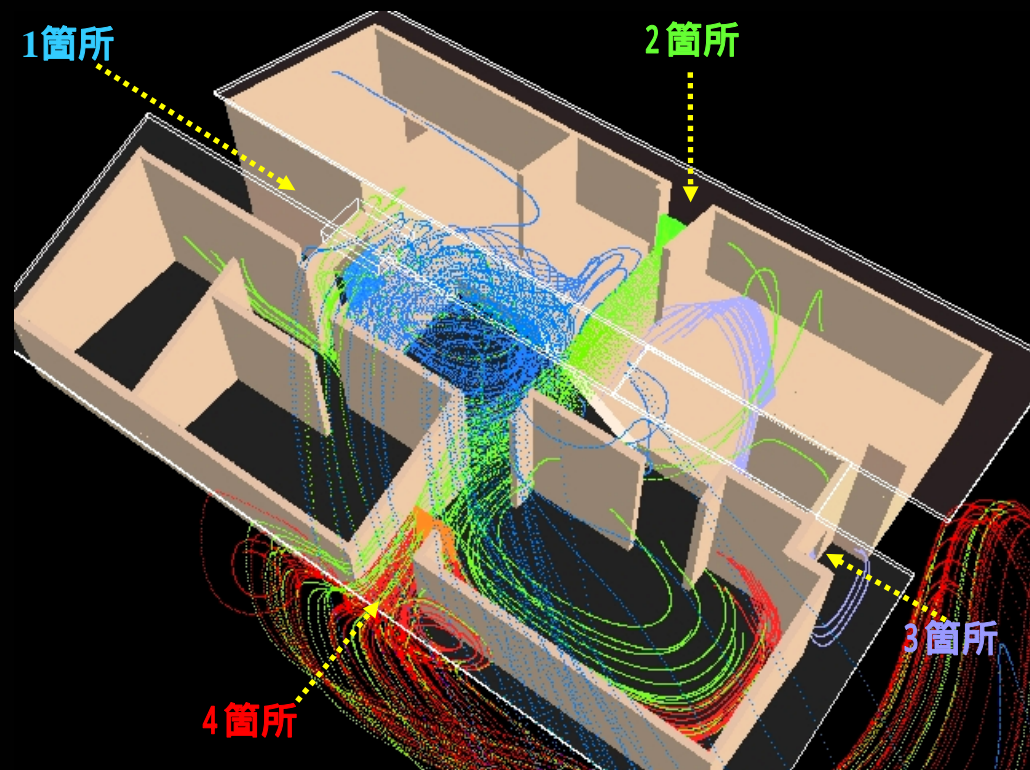
4個の描画ウィンドウに4時刻単位に流線表示モジュールを使ってリボン表示した様子を示す。この例では、描画ウィンドウ間で流線を開始する位置を同期させて一括操作することにより、時系列データ間の流れの比較や瘤内部の流れを確認できる。

## 3DCTをもとにした脳動脈瘤のシミュレーション



# パーティクル・リリース処理の同時操作例

同時操作数: 4

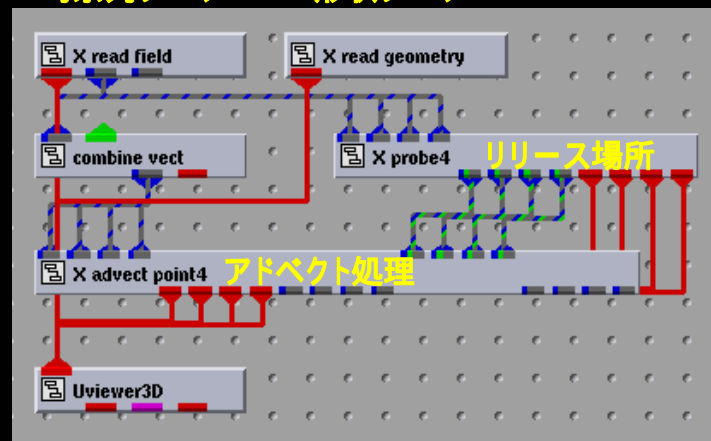


4個のパーティクル  
リリース・モジュール  
の同期を取って  
同じタイムステップで  
アニメーション表示して  
流れ場を確認できる。

熱帯気候の家の流れのシミュレーション

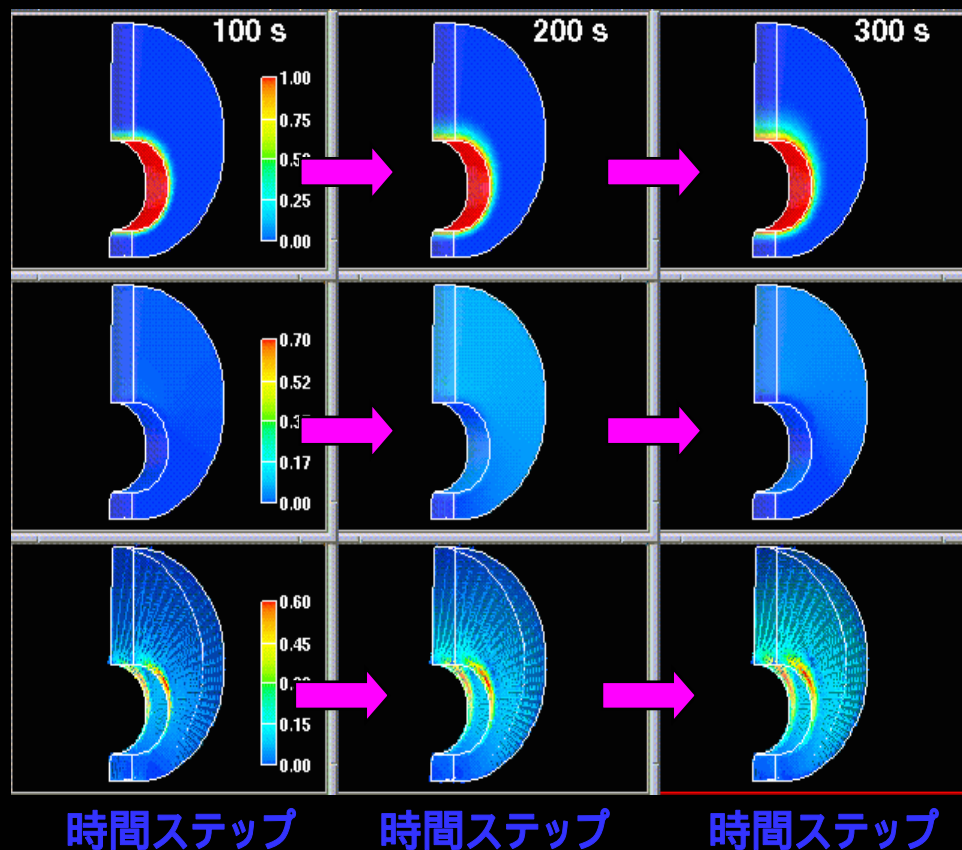
時系列データ

形状データ



# 複数成分の時系列データの可視化例

描画ウィンドウ数: 9



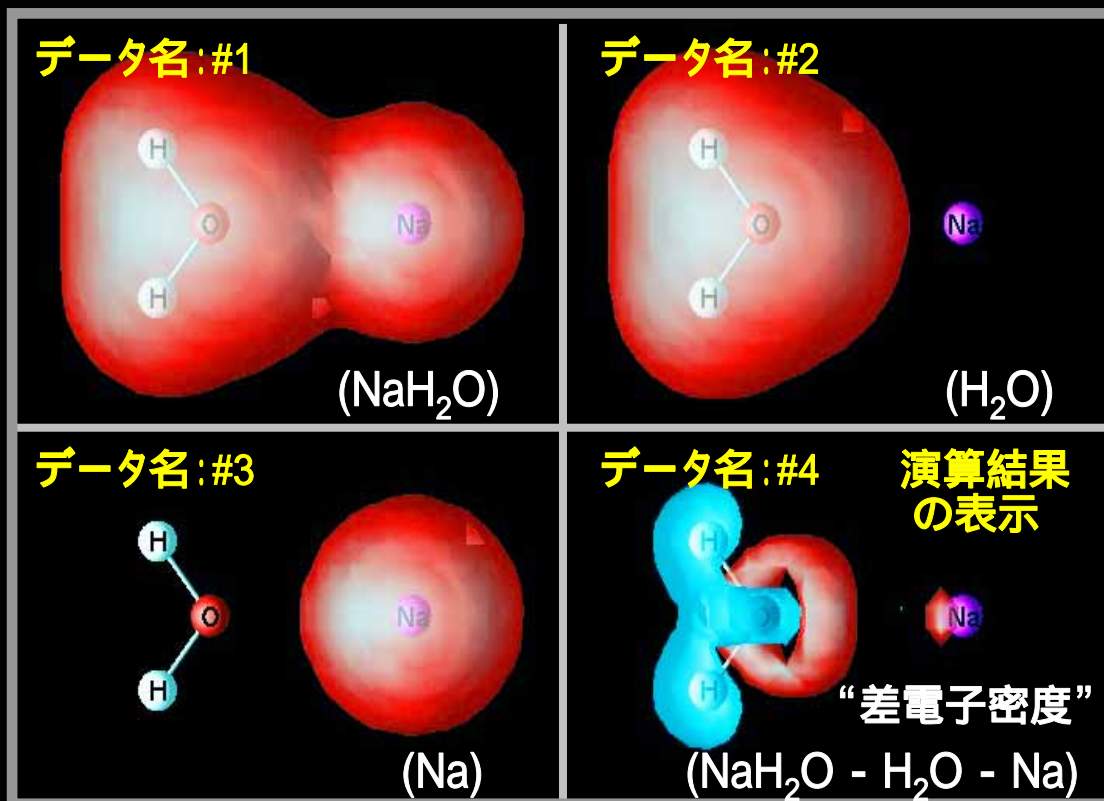
複数のステップデータと  
複数の成分データを  
同時に表示することにより、  
時系列データの動きと  
成分データ間の相関関係  
を確認できる。

## 二重円管内の熱対流のシミュレーション



# 数式演算処理の演算結果の可視化例

描画ウィンドウ数: 4



Gaussianによる分子軌道法計算結果を解析した例として差電子密度分布を求めたものを示す。

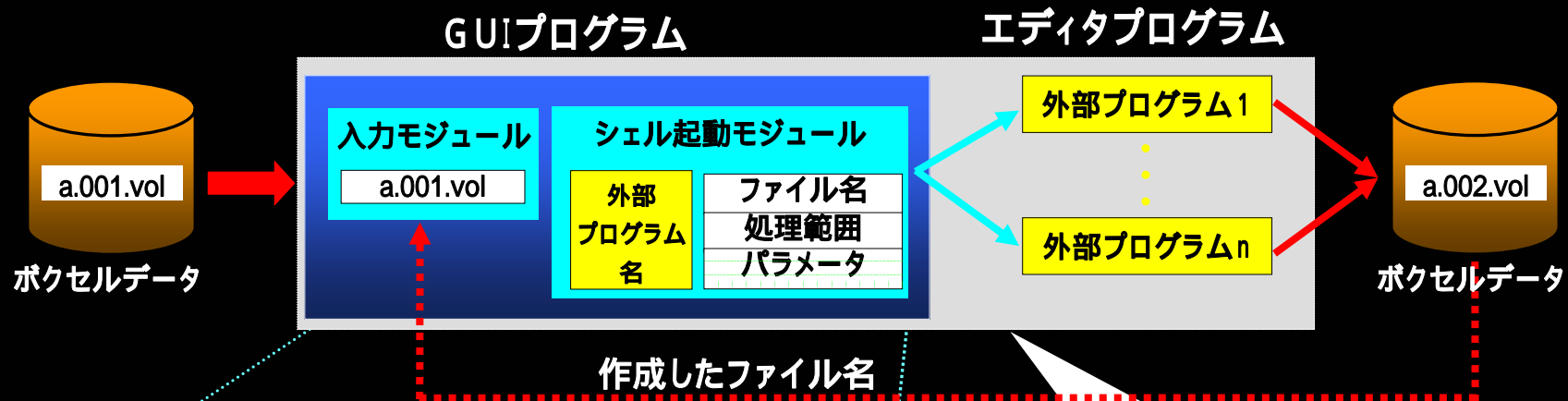
操作パネルから入力した数式を使って演算処理を施した様子を示す。得られた結果は、インタラクティブに描画ウィンドウで確認できる。

演算式:

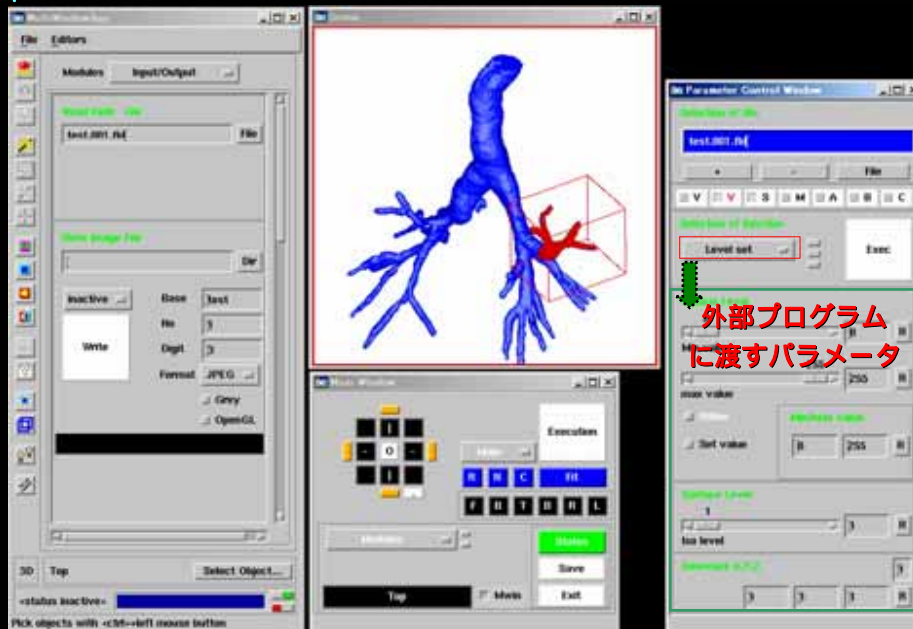
$$\#1 - \#2 - \#3 = \#4$$

数式演算処理を用いた電子密度の可視化

# GUI部分にAVSを使用した「ボクセルエディタ」の開発



可視化アプリケーションの操作パネル



エディタ本体は、AVSの Network Editor を使って作成。  
編集処理を行うプログラムは、機能別にFORTRANを使って作成し、エディタ本体からシェルプロシジャ起動モジュールを使って動的に呼び出す。

# 形状抽出

ボクセル  
データ

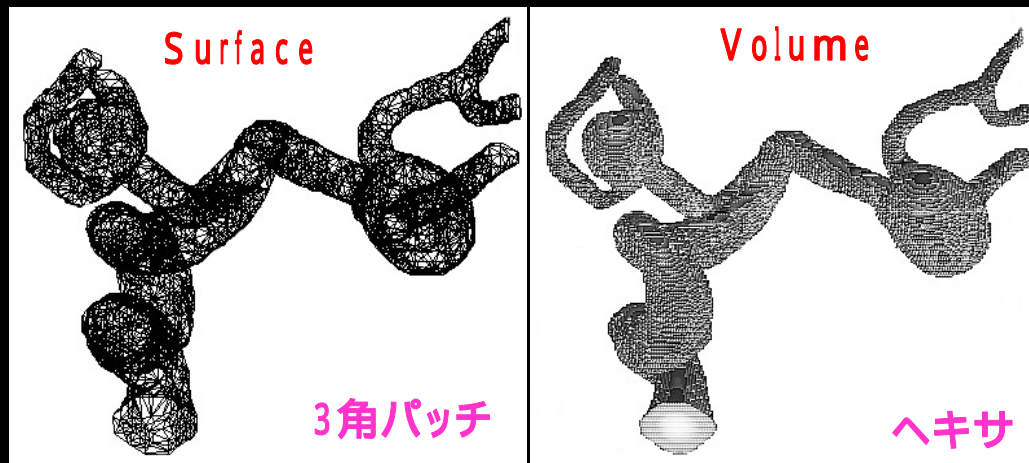
Vアプリ

形状出力

ユーティリティ等

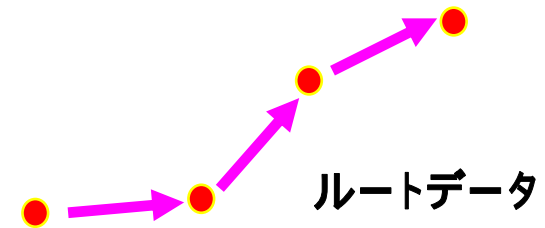
Format変換

- isosurface + reduction モジュール
- isovolume モジュール
- print field モジュール
- STL, Nastran
- AVS(UCD, Geometry)

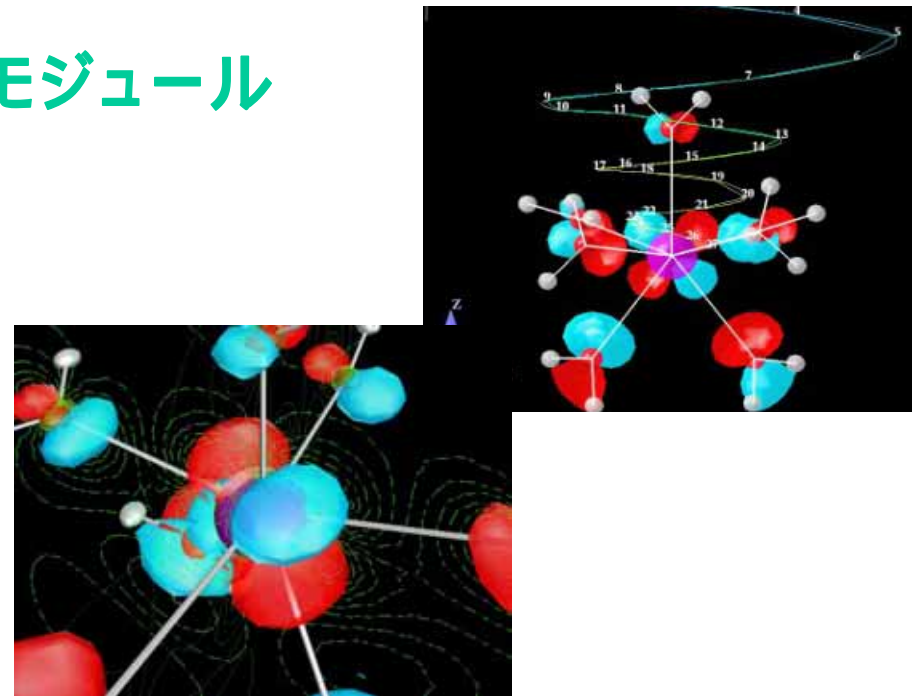


# フライスルー・モジュール

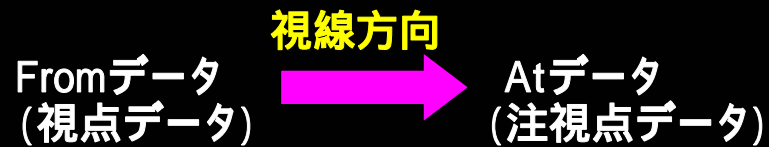
- カメラを移動させるモジュール



- オブジェクトを移動させるモジュール  
(ルート不要)

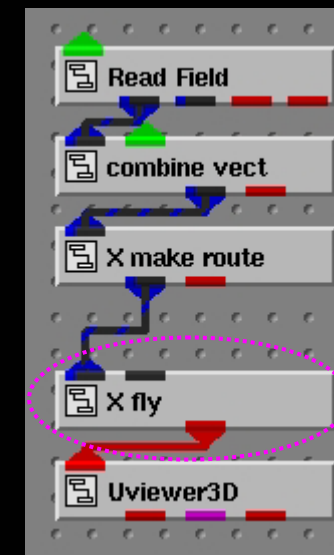
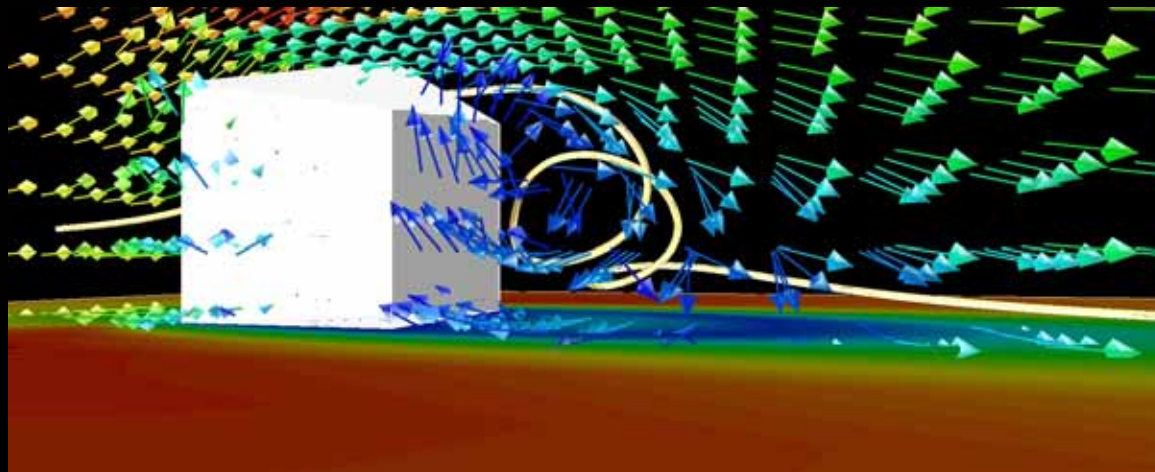
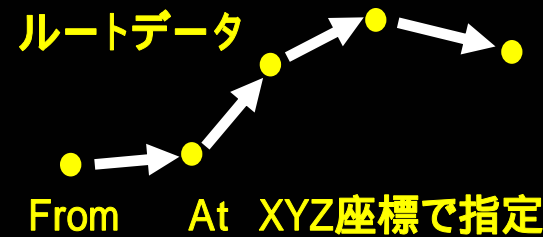
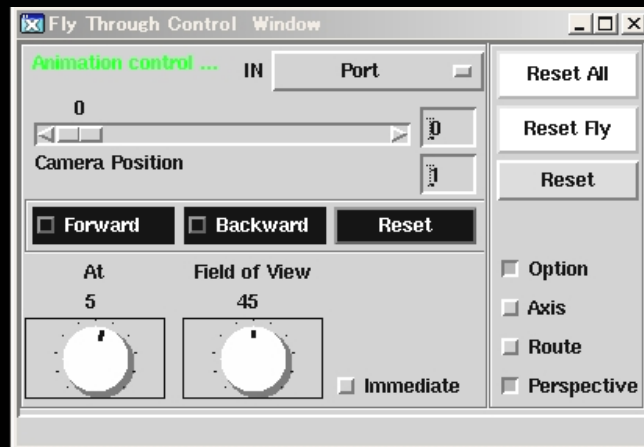


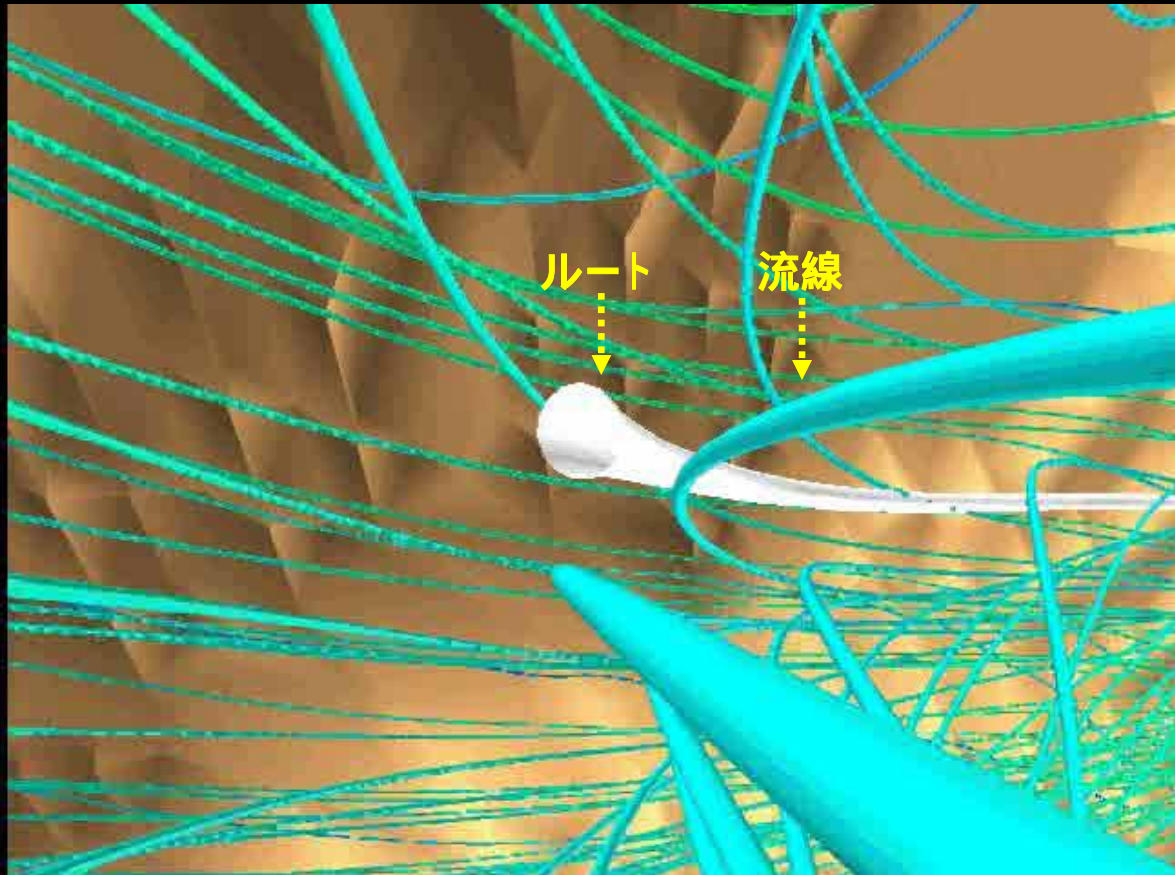
# 「カメラを移動させる方式」のフライスルー可視化



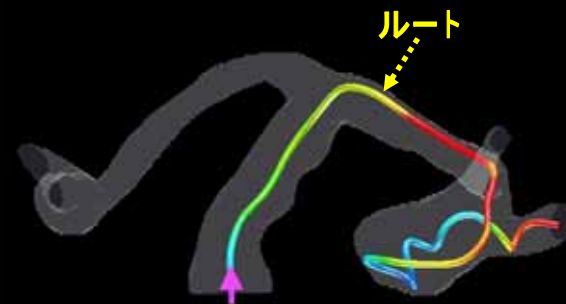
ルートデータ  
入力ポートから供給 (  $x,y,z$  )  
ファイルから供給 (  $x,y,z$   $x_1,y_1,z_1$ ,  $x_2,y_2,z_2$  )  
From At

カメラの向きの補正  
視線方向 (From At) の固定

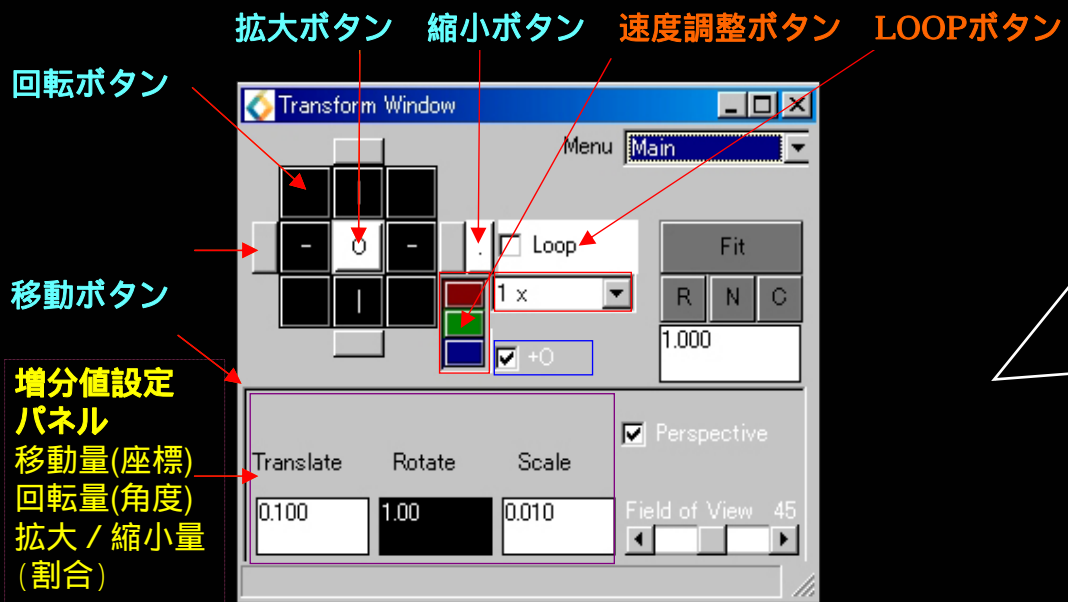




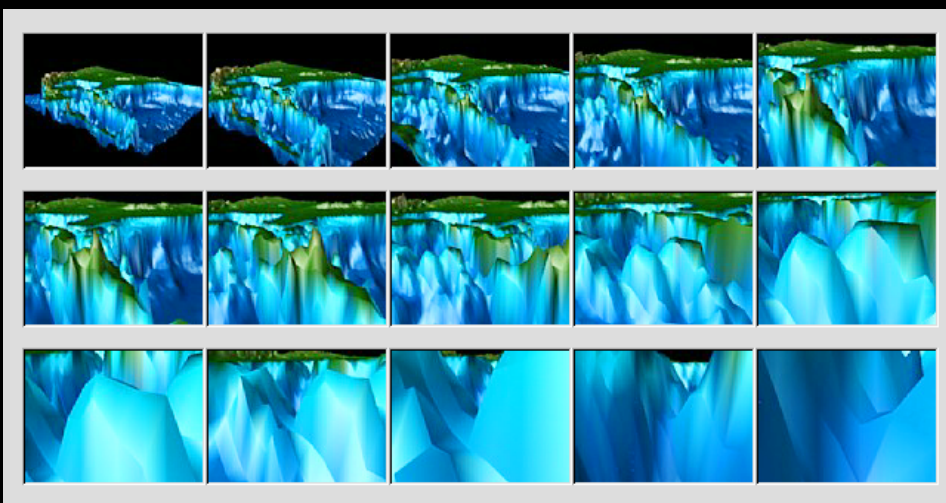
脳の血流解析シミュレーション  
速度の大きさに移動スピードを制御



# 「オブジェクトを移動させる方式」のフライスルー可視化例



操作パネル上の  
定量幾何変換機能と  
LOOP機能を使って、  
可視化空間を自由に  
フライスルーできる。



フライスルー・ルートは不要

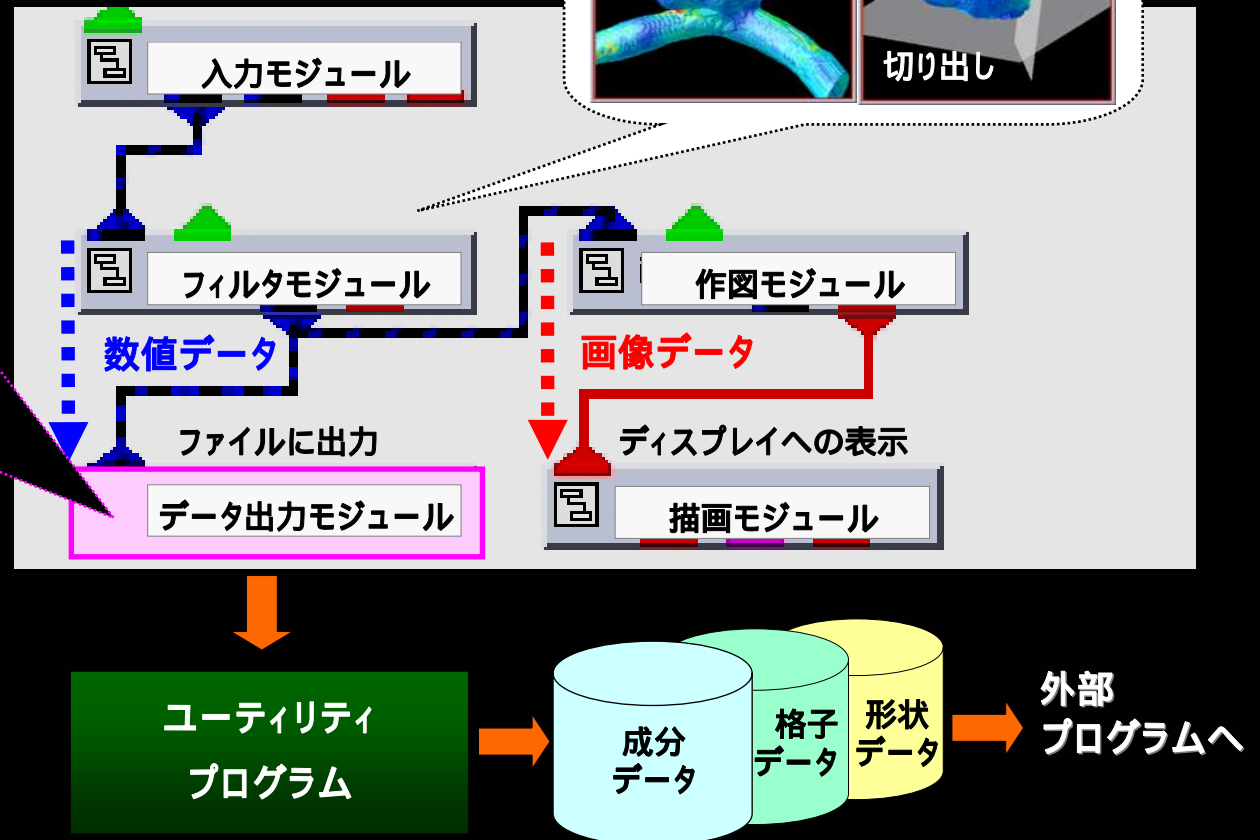
# 数値データ出力モジュール

フィルタモジュールを使って  
加工(切り出し・除去・演算  
などの処理)したデータ

↓  
連番ファイルで保存  
Case.001.dat, Case.002.dat, ...

保存ファイルの形式  
ASCII, Field, HDF, NetCDF

ネットワーク接続例 :





# Gaussian利用支援システム

Gaussianを使いこなすためのツール

“ジョブの投入～可視化”



計算の大規模化

膨大な数値データ

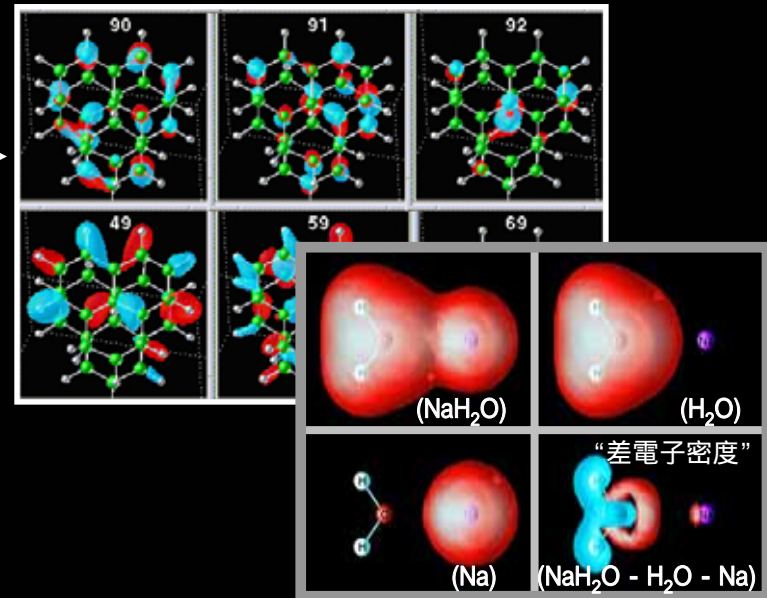
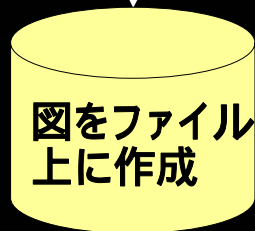
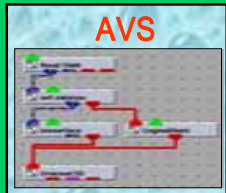
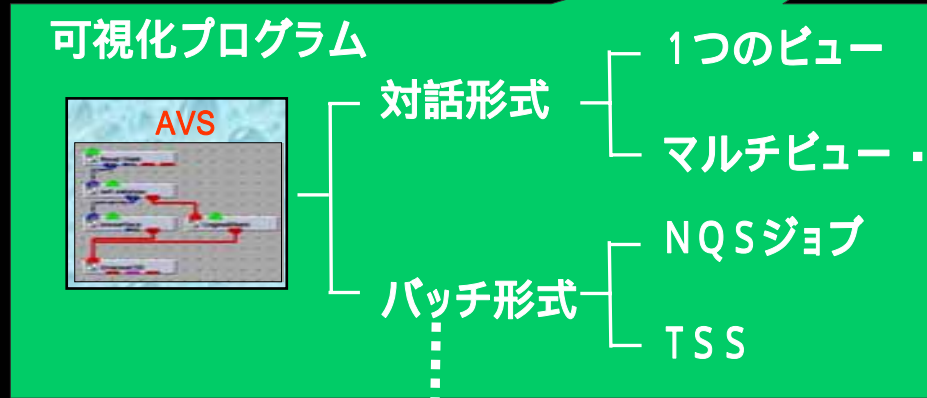
# 概要



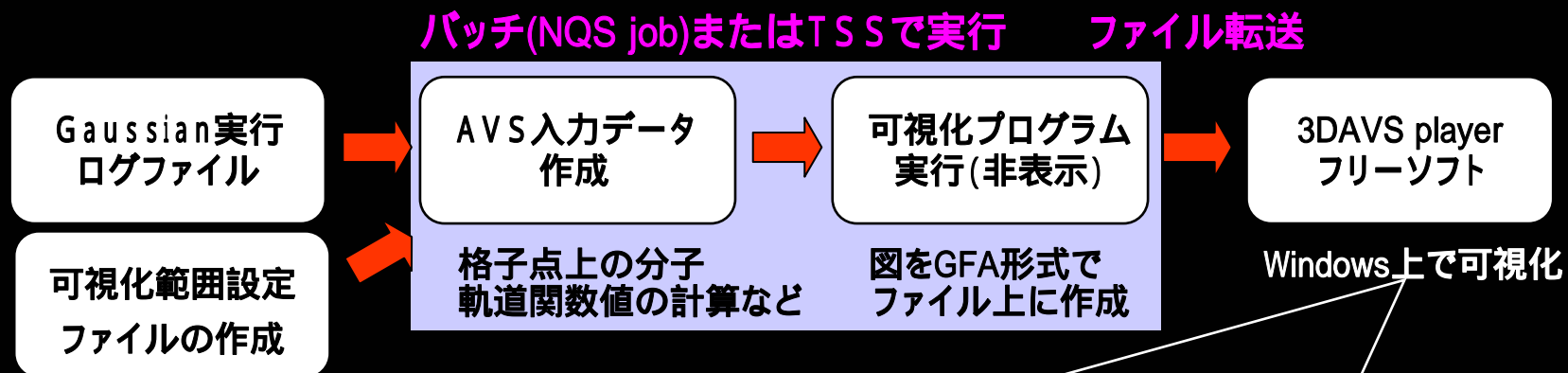
入力データの生成支援  
NQSジョブの投入支援

計算進行状況の確認

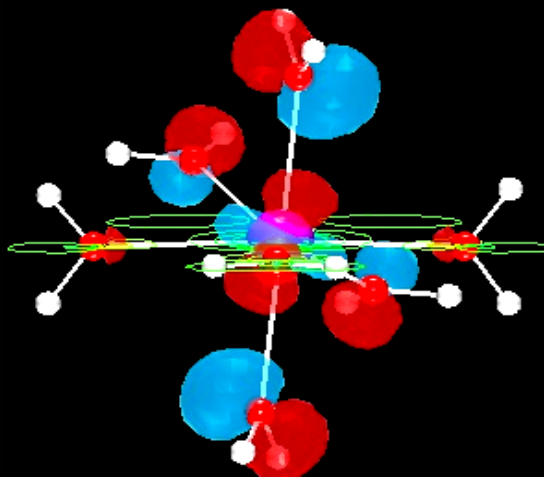
可視化データの絞込み  
可視化データ生成  
可視化(分子軌道、電子密度、振動解析など)



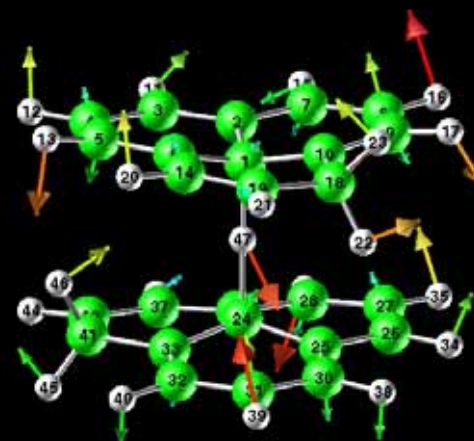
# バッチ可視化の適応例



指定した分子軌道のアニメーション表示



指定した振動サイクルのアニメーション表示



Gaussianの分子軌道や振動解析結果のバッチ可視化

## バッチ可視化の特徴

### ◆ 長 所

- 時間のかかる処理はバッチ処理で依頼 できる
- 回線速度の遅い利用環境から利用できる
- 行モードの端末エミュレータから利用できる

### ◆ 短 所

- 定型的な可視化しかできない
- 複雑なAVSの可視化プログラムは利用できない

### ◆ 必要条件

- ログインノードとNQSノードにAVSが必要

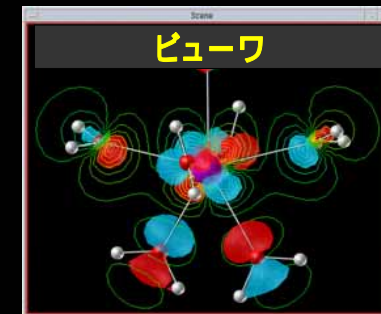
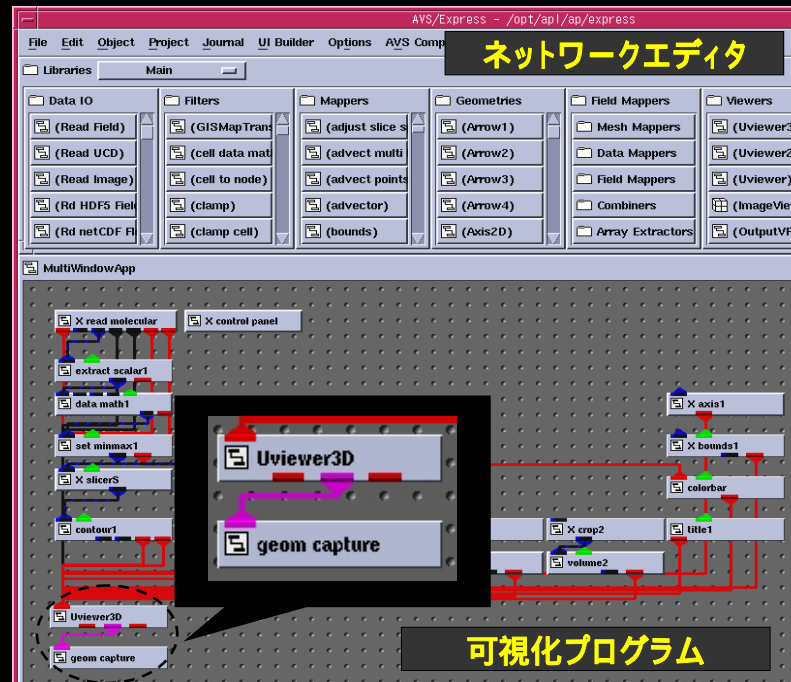
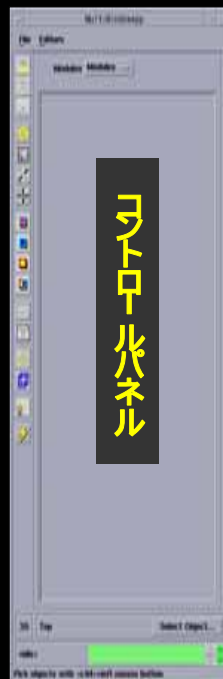
# バッチ可視化のしくみ

## 起動コマンドの形式

xp Vファイル -none -offscreen < 操作ファイル

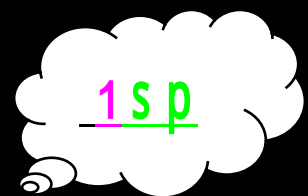
## 操作ファイルの内容:

ファイルの読み込み    パラメータの変更    図をファイルに保存    終了



◆ 分子軌道のデータの絞込み

特定の軌道情報      標準出力



g09cuto ログファイル名      原子番号の軌道の型

Molecular Orbital Coefficients

軌道番号 (nnnn) →

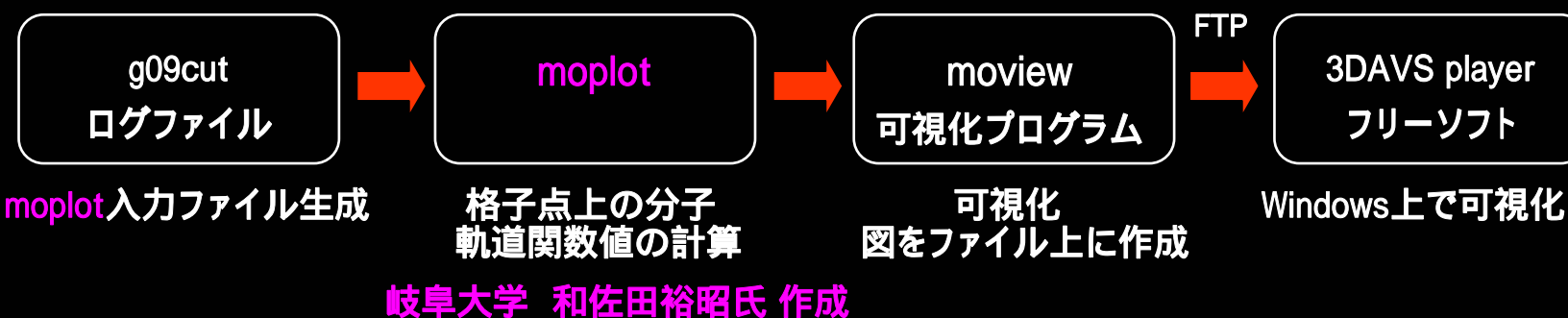
		1	2	3	4	5
		(A)--0	(A)--0	(A)--0	(A)--0	(A)--0
EIGENVALUES --		-14.41663	-14.41663	-3.93475	-2.16602	-2.16088
1	1 Pt 1S	0.00007	0.00040	0.99191	0.00000	0.00014
2	2S	0.00009	0.00052	-0.01746	0.00000	-0.00061
3	3S	0.00000	0.00000	0.00000	0.00000	0.00030
4	4PX	0.00000	0.00000	0.00000	0.99680	-0.00017
5	4PY	0.00000	0.00000	0.00000	0.00017	0.99725
6	4PZ	0.00000	0.00000	0.00000	0.00002	0.00013
7	5PX	0.00000	0.00000	0.00000	0.00395	0.00000
8	5PY	0.00022	0.00128	-0.00244	0.00000	0.00649
9	5PZ	0.00000	0.00000	-0.00001	0.00000	0.00000
10	6PX	-0.00067	0.00011	0.00000	-0.00052	0.00000
11	6PY	-0.00007	-0.00043	-0.00216	0.00000	-0.00073
12	6PZ	0.00000	0.00000	0.00000	0.00000	0.00000
13	7D 0	-0.00001	-0.00006	-0.01067	0.00000	0.00080
14	7D+1	0.00000	0.00000	0.00000	0.00000	0.00000
15	7D-1	0.00000	0.00000	0.00000	0.00000	0.00000
16	7D+2	0.00000	0.00001	0.00120	0.00000	0.00004
17	7D-2	0.00002	0.00000	0.00000	-0.00113	0.00000
18	8D 0	-0.00010	-0.00062	0.01327	0.00000	0.00064
19	8D+1	0.00000	0.00000	0.00000	0.00000	0.00000
20	8D-1	0.00000	0.00000	0.00000	0.00000	0.00000
21	8D+2	0.00002	0.00013	-0.00202	0.00000	0.00006
22	8D-2	0.00235	-0.00040	0.00000	-0.00448	0.00000
23	2 Cl 1S	0.00004	0.00007	-0.00042	-0.00348	-0.00325
24	2S	-0.00048	-0.00021	0.00856	-0.00009	-0.00090
:	:	:	:	:	:	:

分子軌道係数

軌道の型 ↓

注目する原子の軌道係数の型の分子軌道係数の大きな値

## ◆ 分子軌道の可視化

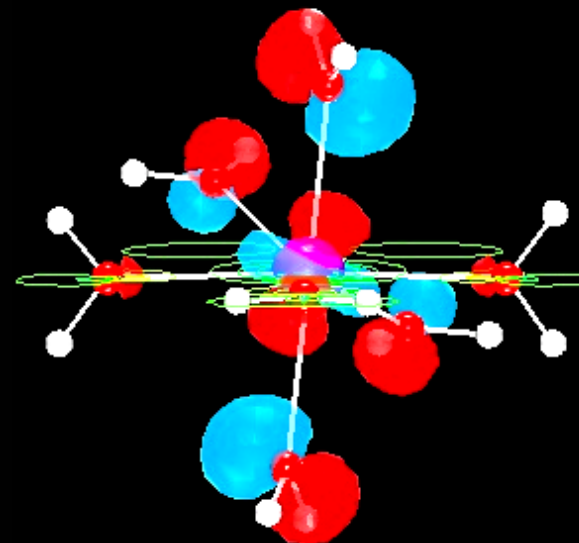


### 対話形式での可視化

g09cut case.log 1-50,200-250  
moview case または moview4 case

### バッチ形式での可視化

g09cut case.log 1-50,200-250  
moplotviewJ case



# ◆ 振動解析のデータの絞込み

振動スペクトル情報

標準出力

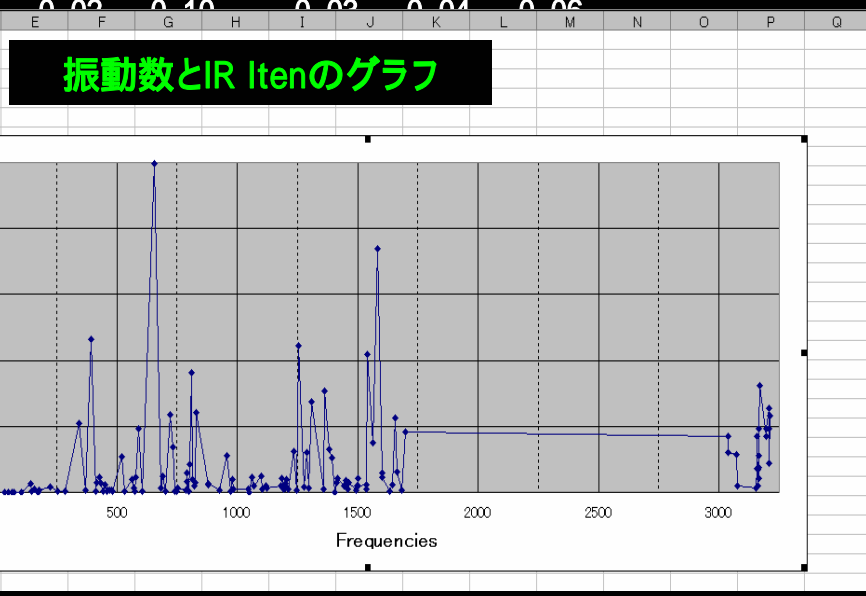
g03cutv

ログファイル名

Gaussianの出力ファイルの振動解析情報

		1 A2''			2 E'			3 E'		
Frequencies		--	36.5487		51.3892		67.5343			
Red. masses		--	4.8225		4.4608		5.2514			
Frc consts		--	0.0038		0.0069		0.0141			
IR Inten		--	0.3920		0.0717		0.3278			
Atom	AN	X	Y	Z	X	Y	Z	X	Y	Z
1	6	-0.01	-0.01	0.01	-0.01	-0.01	-0.01	0.04	0.06	0.05
2	6	-0.06	-0.03	-0.01	0.00	-0.01	-0.06	0.05	0.06	0.01
3	6	-0.05	-0.10	0.04	0.01	0.02	-0.10	0.02	0.04	0.06
4	6	0.01	-0.01	0.01						
5	6	0.07	-0.01	0.01						
6	6	0.07	-0.01	0.01						
7	6	-0.12	0.01	0.01						
8	6	-0.12	0.01	0.01						
9	6	-0.07	0.01	0.01						
10	6	-0.01	0.01	0.01						
11	1	-0.10	-0.01	0.01						
12	1	0.02	-0.01	0.01						
13	1	0.13	-0.01	0.01						
14	6	0.13	0.01	0.01						
15	1	-0.16	-0.01	0.01						
16	1	-0.16	0.01	0.01						
17	1	-0.07	0.01	0.01						
18	6	0.04	0.01	0.01						
19	6	0.13	0.01	0.01						
20	1	0.19	0.01	0.01						
:	:	:	:	:						

振動数 (nnn)





## ◆ 振動解析の可視化

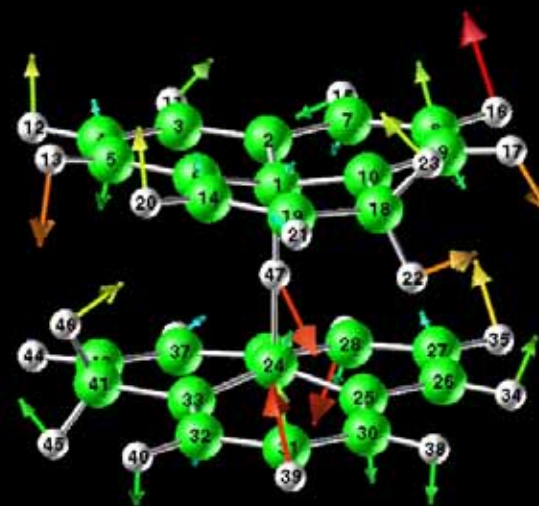


### 対話形式での可視化

```
g09vib case.log n 1-5 10-20  
movib case
```

### バッチ形式での可視化

```
g09vib case.log f 1000 4000  
movibJ case
```



# 今後の予定

---

- ◆ **新たなモジュールの開発**

  - 新しい可視化を研究者に提供

- ◆ **既存モジュールのチューニング**

  - 操作性の向上、機能追加

- ◆ **AVS/Express PST 用のモジュール開発**

  - 処理の高速化、大規模データ対応

**Thanks**