# AVS 入門 (1)

伊東栄典[†]

## 1 AVS の概要

近年では計算機を用いて流体解析，分子設計，地球物理，などさまざまな解析が行なわれています．計算機の高速化により，膨大なデータが入出力されて解析できるようになりました．一昔前までは，解析結果は折れ線グラフにする程度でしたので，表計算ソフトや自作の解析プログラムを用いることにより，解析結果の可視化を手作業で行なうことができました．しかし入出力されるデータが増大するにつれ，手作業でデータを加工することは困難になり，当然解析用のツールを開発する必要がでてくるようになりました．このような解析ツールの開発は，本来の研究とは関係が無いために正当な評価をされない作業になってしまいます．

AVS (Application Visualization System) [1]はこのような作業を支援してくれるシステムです．AVS は可視化システムという日本語名からもわかるように，さまざまなアプリケーションから出力されるデータを，目に見える形に描画してしてくれます．図 1 に AVS による可視化を示します．従来は，さまざまな解析プログラムに特化した可視化プログラムを開発する必要があったのですが (図 1 の左側)，AVS を用いれば AVS が可視化作業を一手に引き受けてくれます (図 1 の右側)．また，AVS には豊富な解析用のモジュールが用意されており，より多角的なデータ解析ができるように設計されています．従来視覚化できなかったデータを視覚化できるようになるため，解析時間の短縮と新たな発見を生み出すことが期待できます．AVS を用いることで，利用者は計算機の能力を問題解決のために十分に使える事ができるようになるでしょう．

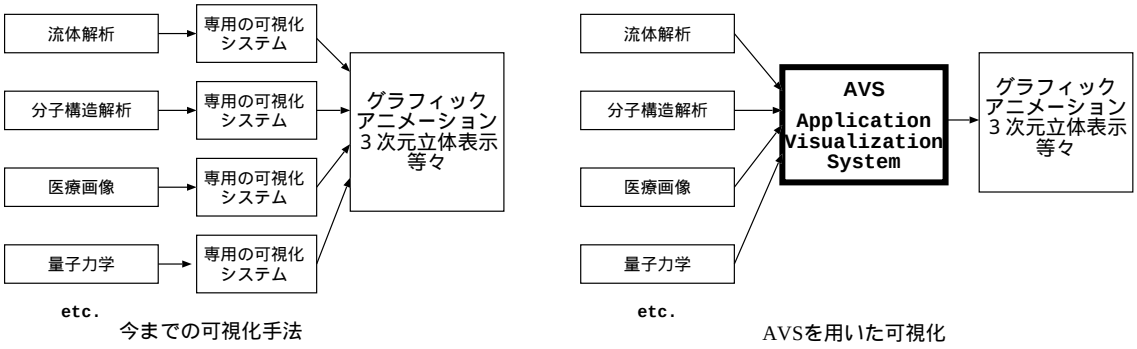


図 1: AVS による可視化

AVS は，元々科学技術計算結果の可視化のために開発されたものでしたが，その高度なグラフィック機能とビジュアル・プログラミング機能により広い分野において利用されています [1]．表 1 に現在利用されている分野を示します．

[†]九州大学大型計算機センター 研究開発部　E-mail:itou@cc.kyushu-u.ac.jp

[1]このソフトウェアを配布している会社も Advanced Visual Systems という名前になっています．
http://www.avs.com/

表 1: AVS 利用分野

| | | |
|---|---|---|
| ・医療画像 | ・有限要素法解析 | ・流体力学 |
| ・資源探査 | ・環境科学 | ・リモート ・センシング |
| ・量子力学 | ・分子設計 | ・CAD データ表示 |
| ・OR 分析 | ・金融データ解析 | ・一般画像処理 |
| ・計測 / 実験結果表示 | etc. | |

# 2 AVS 利用の開始と終了

九州大学大型計算機センター (以下センター) で AVS を利用するには以下の二つの方法があります．

- センターのワークステーション室で直接利用する
- ネットワーク経由で利用する

研究室などに置いてある手元の計算機で X Window System が利用可能ならば，ネットワーク経由でセンターの AVS を利用する事ができます．ただしネットワーク経由による利用の場合，利用者の端末となる計算機の性能により，AVS の性能が制限されるかもしれません．画面に描画される色の数は端末として使用している X Window System により制限されます．256 色しか表示できない X Window System だと，正確な画像は表示されないかもしれません．画像処理用のハードウェアが搭載されていなければ，画像の処理に時間がかかるため遅く感じられるかもしれません．データ量が多い画像の場合は，ネットワークの通信帯域も問題になるでしょう．

センターでは以下の計算機で AVS を利用することができます．

表 2: AVS を利用できる計算機

| 計算機名 | 機種 | ネットワーク経由の利用 |
|---|---|---|
| vhsgi.cc.kyushu-u.ac.jp | シリコングラフィックス i-station | × |
| vcsgi.cc.kyushu-u.ac.jp | 〃 | × |
| qgas-o1.cc.kyushu-u.ac.jp | 富士通 S-4/10 | × |
| qgas-o2.cc.kyushu-u.ac.jp | 〃 | × |
| qvisa.cc.kyushu-u.ac.jp | クボタ TITAN | ○ |
| kyu-cc.cc.kyushu-u.ac.jp | 汎用機 M-1800 | ○ |

## 2.1 利用の開始

利用の開始方法を示します．

- センターのワークステーション室で直接利用する
  利用したいワークステーションに，利用登録をしておく必要があります．利用登録をするコマンドは`touroku` です．オープン端末室の X 端末またはネットワーク経由で`kyu-cc` にログインして`touroku` コマンドを入力しましょう．

```
kyu-cc % touroku all ↵                    ← 全てのワークステーションに利用登録
```

これで kyu-cc 以外のワークステーションが利用できるようになります．センター棟 2 階に置いてあるワークステーションから AVS を利用して下さい．

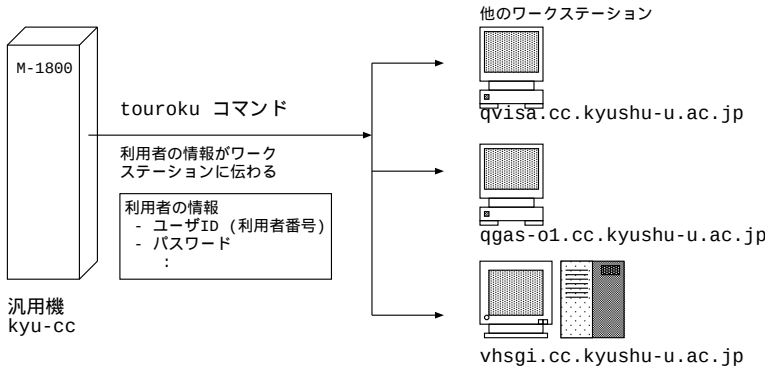


図 2: 直接利用する場合の準備

- ネットワーク経由で利用する
  ネットワーク経由で利用する場合，自分の目の前にあるワークステーション側で，他のワークステーションからの画面表示を許可するよう設定する必要があります．ここでは，自分の目の前にあるワークステーションの名前を *myws@mydomain* とし，遠隔利用している計算機を kyu-cc とします．

```
myws % xhost kyu-cc.cc.kyushu-u.ac.jp ↵
```

  つぎに，telnet などで遠隔利用している先の計算機側で，環境変数 DISPLAY に，自分の目の前にある計算機の名前を設定します．

```
kyu-cc % setenv DISPLAY myws@mydomain:0.0 ↵
```

  以上の設定で，kyu-cc の画面を *myws* に表示できるようになります．上記の設定の様子を図 3 に示します．

- 利用開始コマンド
  AVS の利用開始コマンドは avs です．コマンドプロンプトから avs と入力しましょう．

```
vhsgi % avs ↵
```

図 4 に示しているようなメニューウィンドウが画面の左端に表示されます．これが AVS の開始画面です．
図 4 に書いてあるように，最初のメインメニューから 5 つの項目 (サブシステム) が選択可能です．利用したい項目を選択して下さい．下から 2 番目にある [AVS application ] を選択すると，[AVS Demo ] と

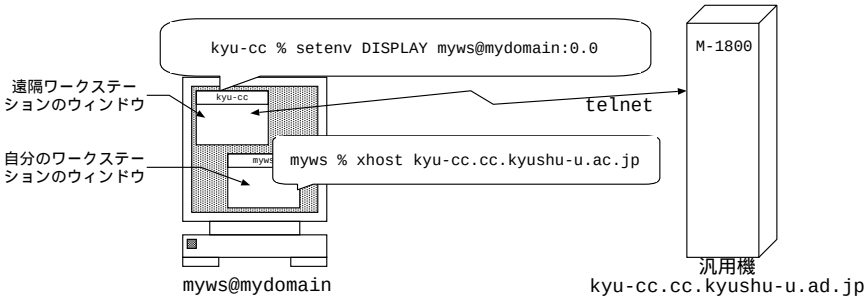


図 3: ネットワーク経由で利用する場合の準備

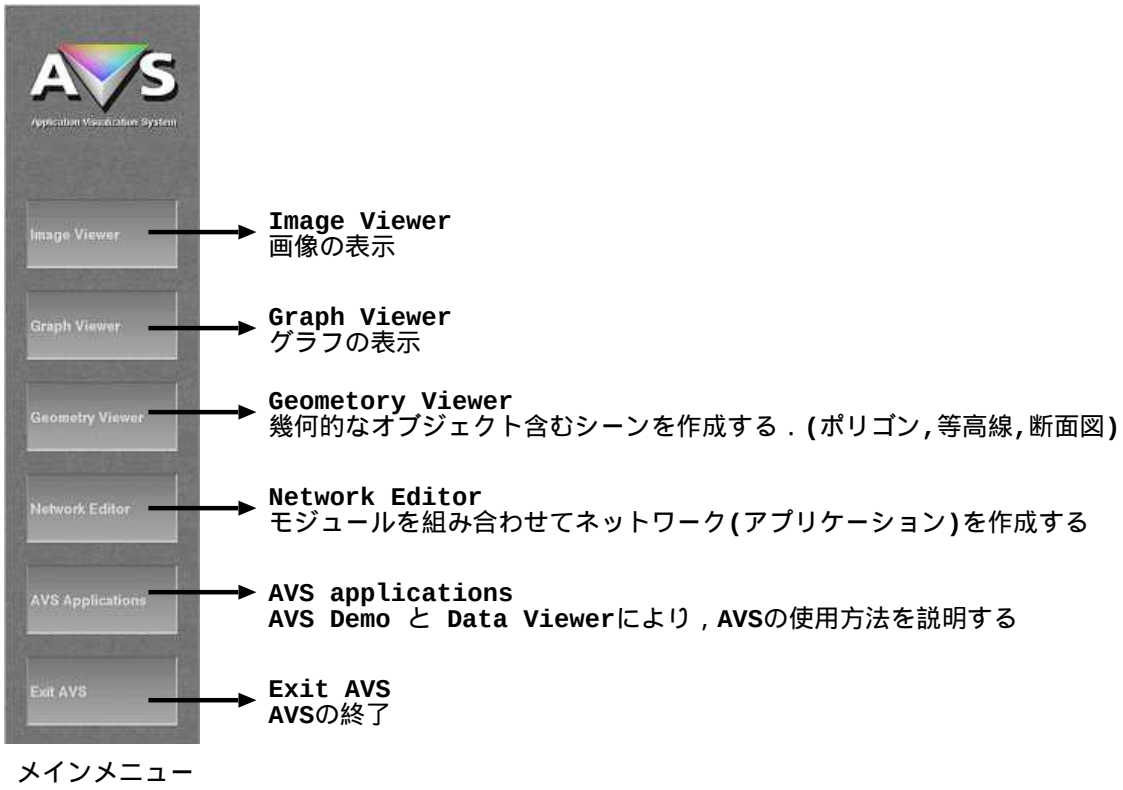


図 4: AVS の開始．メインメニュー

[Data Viewer ] が利用できます．これらは初心者が AVS の使い方を学んだり可視化の手法について学ぶためのツールです．

### 2.2 利用の終了

AVS の終了方法は簡単です．メニューウィンドウの一番下にある，[Exit AVS ] ボタンにマウスのポインタを合せ，マウスの左ボタンを押して下さい．すると AVS が終了します．各サブシステムからメインメニューに戻る場合は，サブシステムのメニュー画面の左上にある，[Close ] ボタンを押して下さい．

### 2.3 ヘルプ

サブシステムごとに操作方法やモジュールの説明のためのヘルプが用意されています．メインメニューからサブシステムを選択すると，それぞれのシステム画面が表示されます．サブシステムのウィンドウの左上にある [Help ] ボタンを押すことで，ヘルプが表示されます．

## 3 各サブシステムについての説明

前節で述べたように，AVS には 5 つのサブシステムがあります．最初の 3 つ，Image Viewer, Graph Viewer, Geometory Viewer は，画面表示を行なうシステムです．これら 3 つのビューワは，一般のビューワとさほど利用方法が異なるわけではありません．AVS に特徴的なものは 4 番目のネットワーク ・エディタです．3 つのビューワは，ネットワーク ・エディタから利用することも可能です．

- **Image Viewer**

  イメージ ・ビューワは画像を取り扱い，表示する場合に用います．一般の画像表示ソフトウェアと同じように，2 次元の画像を表示します．

- **Graph Viewer**

  グラフ ・ビューワはグラフを表示する場合に使います．グラフビューワでは 2 次元のグラフを生成することができます．

- **Geometry Viewer**

  3 次元の立体などを表示する場合に用います．ジオメトリ ・ビューワにより，幾何学的に定義されたオブジェクトを含む 「シーン」を構成することが可能です．オブジェクトは，プログラムまたは AVS の GEOM プログラムライブラリを使ったモジュールにより，あらかじめ用意しておく必要があります．

- **Network Editor**

  データを加工するアプリケーションを作成する場合に用います．AVS におけるデータを可視化する工程は，モジュールを継ぎ合せた 「ネットワーク」を用いて行なわれます．ネットワークやモジュールについては，4 節でより詳細に説明します．

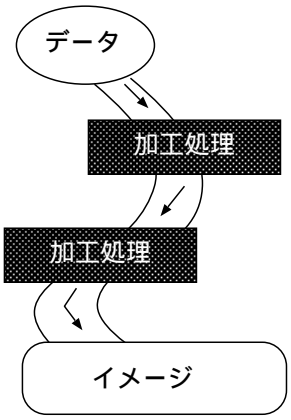


図 5: データフローの概念

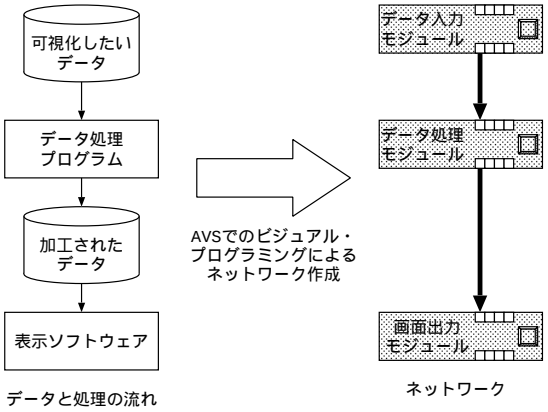


図 6: ビジュアル ・プログラミング

※ 以後 「ネットワーク」という言葉を，モジュールを継ぎ合わせて作成される **AVS** のデータ加工用のアプリケーションという意味で用います．

# 4 AVS アプリケーションの製作

AVS を特徴付けるものとして，以下の 2 つがあります．

- データ ・フロー ・アーキテクチャ (**Data flow architecture**)
- ビジュアル ・プログラミング (**Visual programming**)

データ ・フロー ・アーキテクチャとは，可視化したいデータの入力から可視化された出力までをデータの流れとみなし，流れの過程において種々の処理を加えるといった考え方です．図 5 にデータ ・フローの概念を示します．入力したデータは，種々の可視化のための処理の間を流れていき，最後に可視化された出力となります．可視化のための処理は幾つかの要素に分割されており，この要素をモジュールと呼んでいます．データ ・フロー ・アーキテクチャでは，データの流れは一方向ですので非常にわかりやすい処理構造となります．また，可視化のための処理をモジュールという要素に分割することで，モジュールの組合せや順序により様々な要求に応える可視化ツールを構成することができます．モジュールは，ソフトウェア資産として AVS に標準で付いてくるものや，ソフトウェア ・ベンダーの商品，AVS 利用者が自作して公開しているものなどとして蓄積されています．したがって，特殊な可視化処理でない限り，AVS を利用する際にモジュールの作成を行なう必要はありません．

AVS でのビジュアル ・プログラミングとは，前述のデータの流れとモジュールを GUI(Graphical User Interface) を用いて画面上で視覚的に配置 ・構成することによって可視化処理をプログラミングすることです．具体的には，アイコンとして抽象化したモジュールをマウスを用いて画面上に適当に配置し，モジュール間を線で繋いでいくことによりデータの流れを記述していきます．モジュールは，他のプログラミング言語における関数と違って完全に独立したものであり，すでにコーディングされているので，モジュール間を接続するだけで即実行できます．また，モジュールにはモジュール内で有効なパラメータを GUI を用いて設定することができます．

例えば 2 次元画像に何らかの加工を施して画面に表示する場合には，次のように作業を行なうでしょう．

(A) 画像データの読み出し → (B) 画像の加工 → (C) 加工された画像の表示

AVS では上記のような作業をデータ・フローとして捉えています．$A, B, C$ といった作業の間に，元データ，加工されたデータが存在します．それらが処理プログラム (モジュール) の間を流れていく，と考えるわけです．このデータと処理の流れを，AVS ではビジュアル・プログラミングによりそのままネットワークとして作成します．

AVS を用いてデータの可視化やデータ加工を行なう場合に必要な作業は，以下の 2 つです．

- ネットワーク・エディタによるアプリケーションの作成

  モジュールと呼ばれる部品を継ぎ合せ，データを加工するネットワークを，ネットワーク・エディタを用いて作成します．このネットワークはデータを加工するものとして扱われので，アプリケーションとも呼ばれます．

- 入力データの作成

  AVS での可視化の対象となる入力データは，利用者が用意する必要があります．この入力データを，AVS が読み可能な形式に変換する必要があります．入力データは，ファイルに記述することができます．

## 4.1 ネットワーク・エディタによるアプリケーションの作成

ネットワーク・エディタを用いた，アプリケーションの作成方法を示します．図 4 のメインメニューからネットワーク・エディタを選択すると，図 7 に示しているような画面が表示されます．これがネットワーク・エディタの起動画面です．

図 7 に示すように，ネットワーク・エディタは 4 つの部分から成っています．

- モジュール・コントロール・パネル

  ネットワークの実行と，モジュールのパラメータの操作を行なうことができます．

- エディタ・メニュー

  既存のネットワークの呼び出しや作成したネットワークの保存などを行なうことができます．

- モジュール・パレット

  使用可能なモジュールが格納されています．

- ワーク・スペース

  この領域でモジュールを継ぎ合せてネットワークを作成します．

### 4.1.1 モジュール

モジュールとは AVS における計算単位の事です．ほとんどのモジュールは，入力としてデータを受けとり，何らかの処理をデータに施した後にデータを出力する，という動作を行ないます．AVS でデータを加工，表示するアプリケーション (ネットワーク) を作成するには，モジュールを結合してネットワークを作成すれば良いのです．結合はモジュールからモジュールへのデータの流れ (Data Flow) を表わします．可視化の元となる入力データは，1 つ，もしくは複数のディスク上に配置されているファイルです．他にも，外部プログラムからの出力をモジュールの入力にすることも可能です．

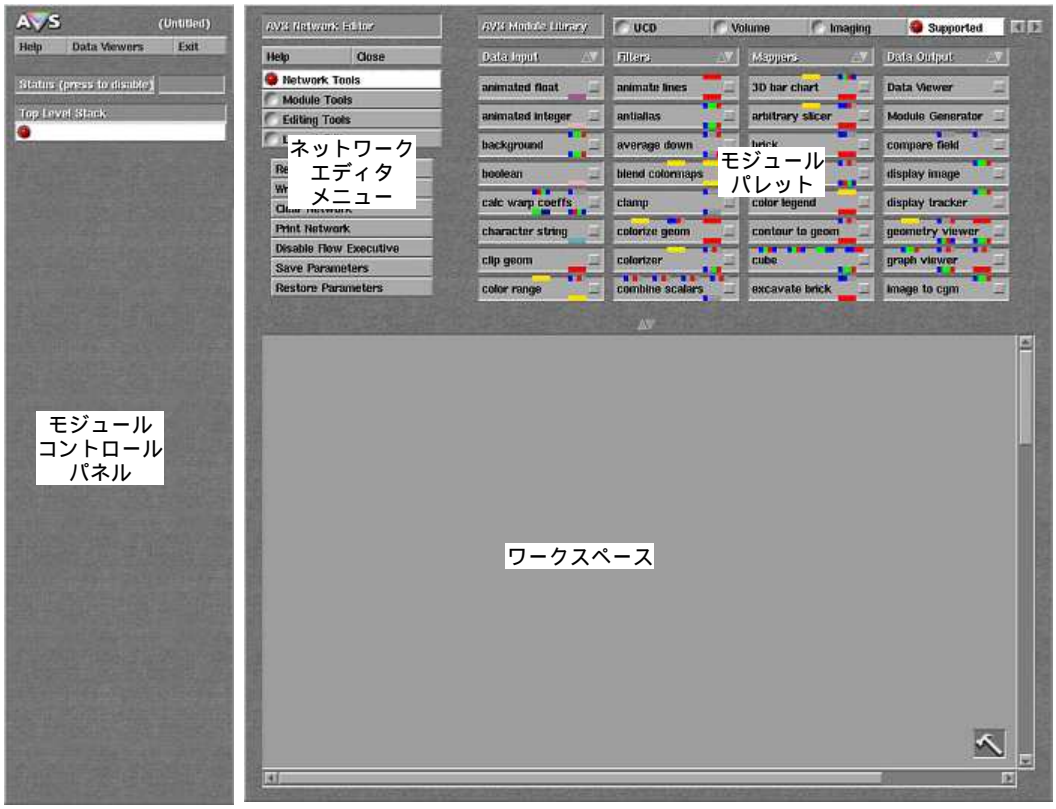


図 7: ネットワーク ・エディタの起動画面

ネットワーク ・エディタ上でモジュールは図 8 のような外見をしています．モジュールの上部には，モジュールへと流れ込むデータの入力ポートがあります．下部にはモジュールにより処理がなされたデータが流れ出る出力ポートがあります．右端にはディンプルと呼ばれるボタンがあります．

入出力ポートは，色の付いた小さな四角で表現されています．ポートは入出力されるデータの形式 (データ ・タイプ) に応じて色が付けられています．ポートの色とデータの対応を表 3 に示します．

モジュールの右端には，ディンプルというボタンがあります．ここにマウス ・ポインタを合せマウスの左ボタンを押すと，そのモジュールのパラメータを設定することができます．マウスの中ボタンや右ボタンを押すと，そのモジュールについての詳細な情報を表示するモジュール ・エディタ ・ウィンドウが表示されます．

モジュールは，機能により 4 つに分類することができます．それぞれの 4 種類が，モジュール ・パレットの 4 列に対応しています．

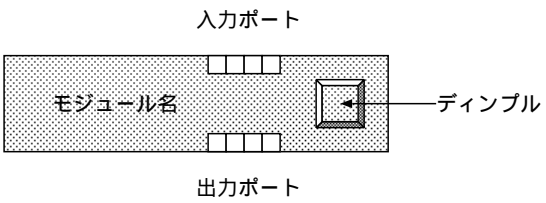


図 8: モジュール ・アイコン

表 3: 色とデータ ・タイプの対応

| 色 | データ ・タイプ |
|---|---|
| 赤 | ジオメトリ |
| 黄 | カラーマップ |
| ライト ・ブルー | ピックスマップ |
| 青, 赤, 緑, 灰の組合せ | フィールドデータ (イメージ , ボリュームを含む) |
| 橙 | 非構造格子データ (UCD) |

- **Data Input モジュール**

  可視化対象となるデータをネットワークに読み込むモジュールです . ファイルからデータの読み込みを行なうモジュールや , 任意の数値列やカラーマップを生成するモジュールもあります .

- **Filter モジュール**

  上流のモジュールから受け取ったデータが複数の成分を持つ場合に , そのデータの一部を選択するモジュールです . たとえば , 各成分に温度と圧力のデータを持つ場合には , フィルターモジュールを利用することで , どちらか一方のデータを選択して下流に流すという操作が行なえます .

- **Mapper モジュール**

  上流のモジュールから受け取ったデータを , ある特定の表現手法に基づいて変換し , そのデータを出力する機能を持つモジュールです . 例えば , ある 3 次元のデータから断面図や等高線を表示したい場合などに用います .

- **Data Output モジュール**

  ネットワークを経由して加工されてきたデータを受け取り , それを画像として表示したり , ファイルとして保存したりするモジュールです . 画像表示のモジュールには , [Image Viewer ] , [Graph Viewer ] , [Geometry Viewer ] といった AVS で最初から使用可能なビューワも含まれています . その他にも画像データを PostScript 形式で保存するモジュールなどもあります .

#### 4.1.2 ネットワークの作成

ワーク ・スペースでモジュールを継ぎ合せてネットワークを作成する方法を説明します . ネットワーク作成における作業は以下の 3 つです .

- モジュールの検索
- モジュールの配置 , 移動 , 削除
- モジュール間の接続と切断

・モジュールの検索

モジュールは , 図 7 に示しているモジュール ・パレットに表示されています . ここでモジュールを検索します . パレットに全てのモジュールが表示されているわけではありません . パレットを上下にスクロールさせて , 求めるモジュールを探して下さい . マウス ・ポインタを , 列の一番上に合せ , そこでマウスの

ボタンを押すことでスクロールします．マウスの右ボタンを押すと下に，左ボタンを押すと上にパレットがスクロールします．

・モジュールの配置，移動，削除

求めるモジュールが見つかったら，そのモジュールをパレットから取出し，ワークスペースに配置します．モジュールの上にマウス ・ポインタを置き，マウスの左ボタンを押したままマウスを動かします．この動作をドラッグといいます．モジュールをマウスの動きに合せて引っぱることができますので，マウスの左ボタンを押したままワークスペースまでマウスを動かし，そこでマウスのボタンを離します．するとモジュールがワークスペースに配置されます．ワークスペース内でのモジュールの移動も同様に行ないます．

ワークスペースからモジュールを削除する場合には，モジュールをワークスペース内にある，ハンマーアイコンまで持っていきます．するとモジュールがワークスペースから削除されます．モジュールの検索，配置，移動，削除の様子を図 9 に示します．

・モジュール間の接続と切断

必要なモジュールを配置したら，モジュール間のデータの流れを設定します．

モジュールとモジュールを接続する場合，マウスを使って入力ポートと出力ポートを接続します．例えばモジュール A の出力ポートからのデータを，モジュール B の入力ポートに接続する場合，(1) モジュール A の出力ポートにマウス ・ポインタを合せ，(2) マウスの中ボタンを押したまま，(3) マウス ・ポインタをモジュール B の入力ポートに移動させます．(4) そこでマウスの中ボタンを離すと，モジュール A とモジュール B が接続されます．モジュールを接続する場合には，同じタイプのデータ，すなわち同じ色のポートしか接続することはできません．

切断の操作は，接続の操作とほぼ同じです．例えばモジュール A の出力ポートとモジュール B の入力ポートに接続されている線を切断したい場合，(1) モジュール A の出力ポートにマウス ・ポインタを合せ，(2) マウスの右ボタンを押したまま，(3) マウス ・ポインタをモジュール B の入力ポートに移動させます．(4) そこでマウスの右ボタンを離すと，モジュール A からモジュール B への線が切断されます．

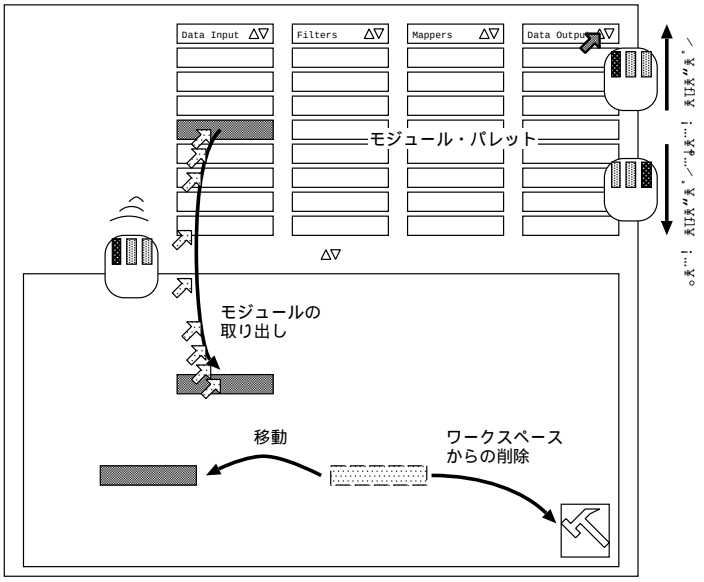


図 9: モジュールの検索，配置，移動，削除

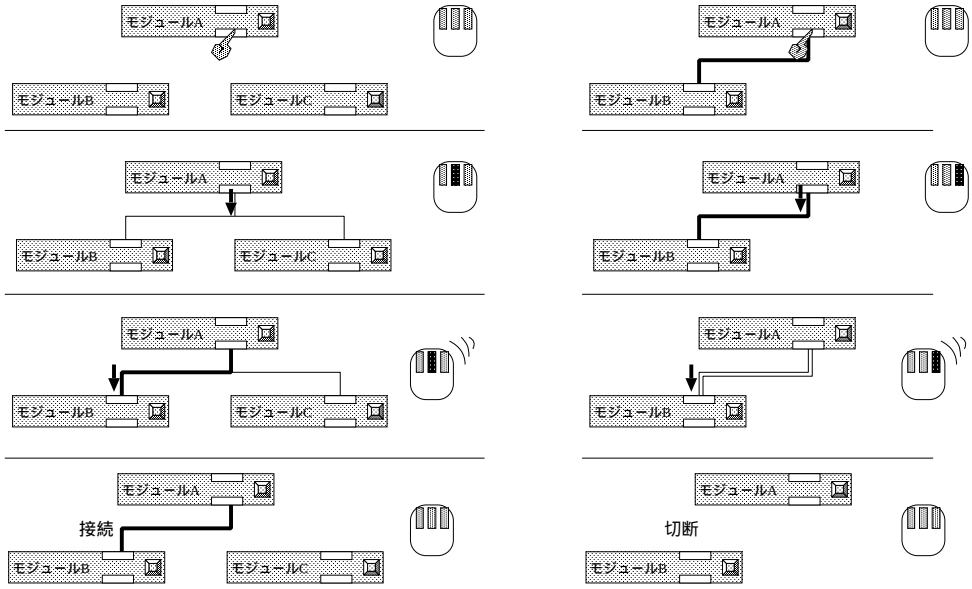


図 10: モジュールの接続と切断

#### 4.1.3 ネットワークの実行

ネットワークを作成したら，それを実行することで可視化などの操作を行なうことができます．ネットワークの最初にある [read image ] や [read ucd ] といったデータを読みだすモジュールで，ファイルなどをのデータを読み込む操作を行なうことで，ネットワークを実行することができます．データを読み込む操作については，第 5 節の中で説明しています．第 5 節の図 14 では，[read image ] モジュールでの画像データの読み込み方法について説明しています．参考にして下さい．

#### 4.1.4 ネットワークの保存と呼出し

一度作成したネットワーク (アプリケーション) は保存しておくことができます．ネットワークを保存しておけば，何度も利用することができます．ネットワークの保存と呼び出しは図 11 に示しているネットワーク ・エディタ ・メニューで行ないます．メニューの下の方に，[Write Network ] ボタンがあります．このボタンにマウスポインタを合せ，マウスの左ボタンを押すと，ファイル ・ブラウザが表示されます．さらに下の方にある [New File ] をマウスで選択すると，ファイル名を入力するためのウィンドウが表示されます．適切なファイル名を入力してください．名前を入力した後に ↵ キーを押すか，[O.K. ] ボタンを押すことで，ネットワークをその名前で保存することができます．

保存しておいたネットワークや，AVS に附属のネットワークは呼び出して使用することが可能です．エディタ ・メニューの下の方に，[Read Network ] ボタンがあります．このボタンにマウスポインタを合せ，マウスの左ボタンを押すと，ファイル ・ブラウザが表示されます．ファイル ・ブラウザを利用して求めるネットワークを探して下さい．ネットワークが見付かったら，そのネットワーク名が表示されている所にマウスポインタを合せて，マウスの左ボタンを押して下さい．

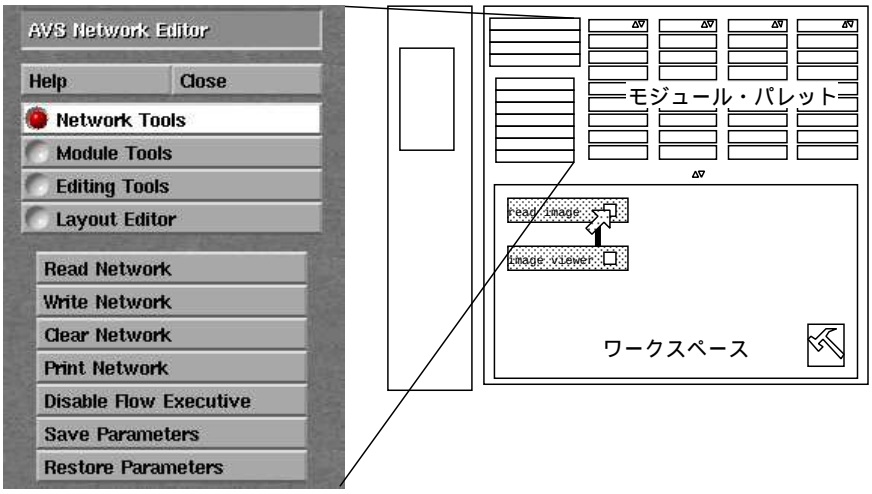


図 11: ネットワーク ・エディタ ・メニュー

## 4.2 入力データの作成

さまざまな科学技術計算用のソフトウェアから出力される解析結果を AVS に可視化させる場合，解析結果のデータを AVS が扱うことのできる形 (データ ・タイプ) に変更する必要があります．標準で AVS が扱うことが可能なデータ ・タイプには表 4 に示すような基本 (primitive) データ ・タイプと，表 5 に示すような集合データ ・タイプがあります．

基本データ ・タイプは，スカラー ・データや文字列のデータです．基本データ ・タイプは，集合データ ・タイプの基本的な構成要素やモジュールのパラメータに使用します．

表 4: 基本データ ・タイプ

| タイプ名 | 意味 |
|---|---|
| byte | 1 バイト (8 ビット) のデータ |
| integer | 1 ワードで表現する符号付き整数 |
| float | 単精度浮動小数点 |
| double | 倍精度浮動小数点 |
| string | それぞれが 1 文字を表わすバイトの配列 (文字列) |

## 4.3 AVS へのデータ入力

AVS へのデータの入力は，データ入力のモジュールを用いて AVS のデータ ・フォーマットに変換することによって行なわれます．図 12 に AVS へのデータ入力の概念図を示します．データがそのまま AVS の既存のデータ入力モジュールで読み込める場合は問題ありませんが，読み込めない場合の対処として以下の 2 通りの方針が考えられます．

(a) 何らかの方法で既存のデータ入力モジュールで読み込めるようにデータを修正する (図 12 中の 1 ~ 3 の手法 )

(b) データを AVS のデータ ・フォーマットに変換するモジュールを作成する (図 12 中の 4 の手法 )

表 5: 集合データ ・タイプ

| タイプ名 | 意味 |
|---|---|
| イメージ ・データ | 任意のサイズの 1 次元 , 4 ベクトル配列 (色相 , 彩度 , 明度 , 透明度) |
| フィールド ・データ | 配列の各位置に m 次元ベクトル ・データを持つ n 次元配列 ( m と n は整数) . 構造化データや 3 次元座標値など . |
| 非構造格子データ (UCD) | 節点と頂点により定義されるセルにより形成されるデータ . 点 , 線 , 三角形 , 四角形 , 四面体角錐 , 角柱 , 六面体要素タイプが利用可能 . 有限要素法など . |
| ジオメトリ ・データ | 3 次元の幾何データ . ポリゴン , メッシュ , 球など . |
| 分子 (PDB) データ | 分子の構造を表現するためのデータタイプ . 分子設計 , 量子力学計算など . |
| ユーザ定義データ | C の構造体のような表記によるデータタイプの定義が可能 . FORTRAN からも利用可能 . |

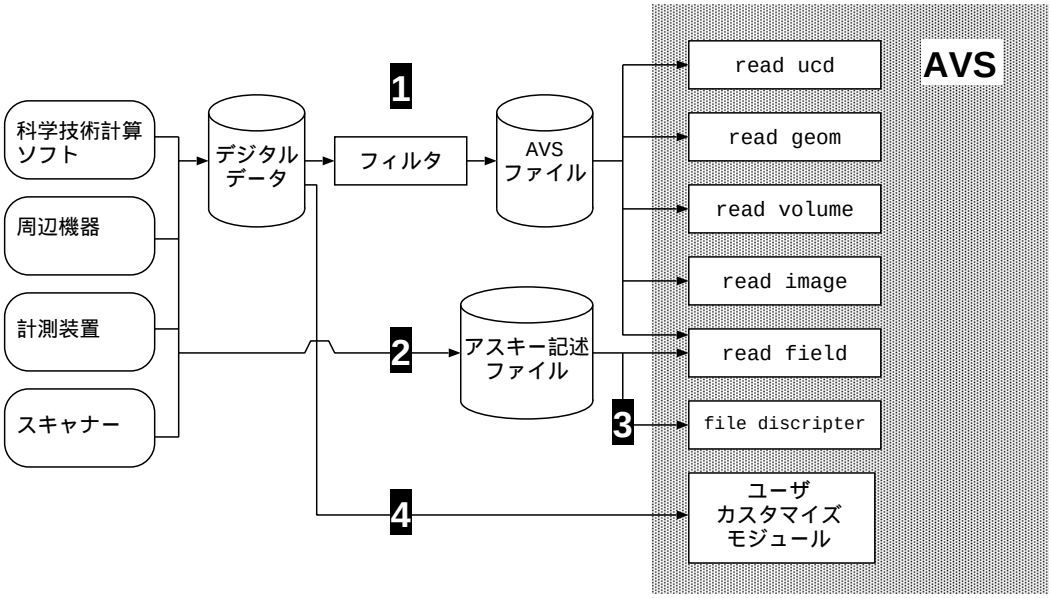


図 12: AVS へのデータ入力

1. フィルタ ・プログラムによるデータ入力

   可視化したい計算結果やスキャナ等の入力機器や実験測定器からのデータは , 一旦データファイルとしてディスク上に置かれます . このデータファイルをフィルタ ・プログラムにかけることによって AVS の各読み込みモジュールで読み込めるフォーマットに変換します . AVS では標準で表 6 の 5 つのデータ読み込みモジュールとそのデータ形式が用意されています . AVS ファイルのフォーマットへのデータ ・ファイルの変換は , 単純なフォーマット変換ですから , C や Fortran により簡単にフィルタ ・プログラムを作成できます . また , geom フォーマットは GEOM 関数を利用して作成できます . GEOM 関数に関しては 『Developer's Guide』[1] を参照して下さい . geom フォーマットへのフィルタ ・プログラムのサンプルは , `/usr/avs/filter` にもありますので参照して下さい .

表 6: AVS 標準入力データフォーマット

| フォーマット | AVS モジュール | 内容 |
|---|---|---|
| フィールド ・フォーマット | read field | 差分計算データ , 実験測定データ (構造格子型データ ) |
| イメージ ・フォーマット | read image | 画像データ |
| ボリューム ・フォーマット | read volume | 3D ボリュームデータ (画像データの積み重ね ) |
| geom フォーマット | read geom | 幾何データ (形状データ ) |
| UCD フォーマット | read ucd | 有限要素法データ (非構造格子型データ ) |

しかしながら , 実際にはフィルタ ・プログラムを作るよりは , 次に説明するアスキー記述ファイルを利用する方が便利です .

2. アスキー記述ファイルによるデータ入力

[read field ] モジュールの機能として , フィールド ・データの入力にアスキー記述ファイルを利用することができます . このアスキー記述ファイルに入力するフィールド ・データの書式を記述しておきます . そうすると , [read field ] モジュールがアスキー記述ファイルを解釈してデータの読み込みを行ないます . この機能は , アスキーとバイナリのどちらの形式のデータ ・ファイルに対しても有効です . ただし , この方法では , アスキー記述ファイルを解釈して読み込むためにオーバーヘッドがかかります .

3. **[file descripter ] モジュールによるデータ入力**

基本的には前述のアスキー記述ファイルの場合と同じです . [file descripter ] モジュールはアスキー記述ファイルを対話的に作成するインターフェースを提供しています . AVS の環境からアスキー記述ファイルを生成し , 同時にその記述に基づいてデータを取り込むことができます .

4. ユーザ ・カスタマイズ ・モジュールによるデータ入力

独自のモジュールを開発すれば , データを直接 AVS に取り込むことが可能です . 開発する場合には , モジュール作成および AVS でのデータの内部フォーマットの知識が必要となります . ただし , 独自に開発したアプリケーションやドライバーによって書き込まれたデータを読み込む際には , 書き込みの部分をそのまま読み込みモジュールに利用することができるので , それほど苦労はありません . むしろ変換のオーバーヘッドが無い分有利です .

AVS は , 内部データ ・フォーマットに対してそれぞれ C と Fortran で記述されたモジュールのソース ・プログラムを持っています . 新規モジュールを書くためのガイドとして , `/usr/avs/examples/` ディレクトリ以下に置いてあるサンプル ・ソース ・プログラムを利用することができます .

# 5 AVS の使用例 : 画像データに色々な加工を施して表示

実際にネットワークを作成して , データを可視化する操作の例を示します . 画像データをイメージ ・ビューワに表示する簡単なネットワークを例として示します . 利用するモジュールは , 画像データを読み込む [read image ] , 画像データをイメージ ・ビューワで表示する [image viewer ] の 2 つです . [read image ] モジュールは Data Input モジュールですのでモジュール ・パレットの左端の列に在ります . [image viewer ] は Data Output モジュールですので , 右端の列に在ります .

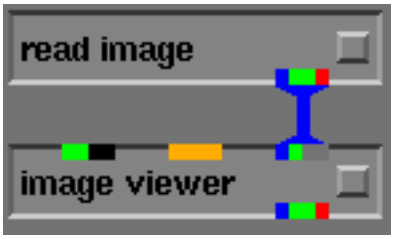


図 13: 画像を Image Viewer で表示するネットワーク

図 13 のようにネットワークを作成します．このネットワークを実行すると画像が表示されます．Data Input モジュールである [read image ] モジュールからファイルを読み込むことで，ネットワークが実行されます．画像ファイルの読み込みは，次の手順で行ないます．まず [read image ] モジュールのディンプルをマウスの左ボタンで押します．すると，図 14 のような [read image ] モジュールを操作するパネルが，左側のネットワーク・コントロール・パネル内に表示されます．図 14 のファイル・ブラウザを利用して表示させたいファイルを選択します．画像ファイルは .x で終る名前が付いています．表示させたいファイルにマウス・ポインタを合わせ，マウスの左ボタンを押すと画像ファイルが選択されます．するとネットワークが実行されて，イメージ・ビューワに画像が表示されます．

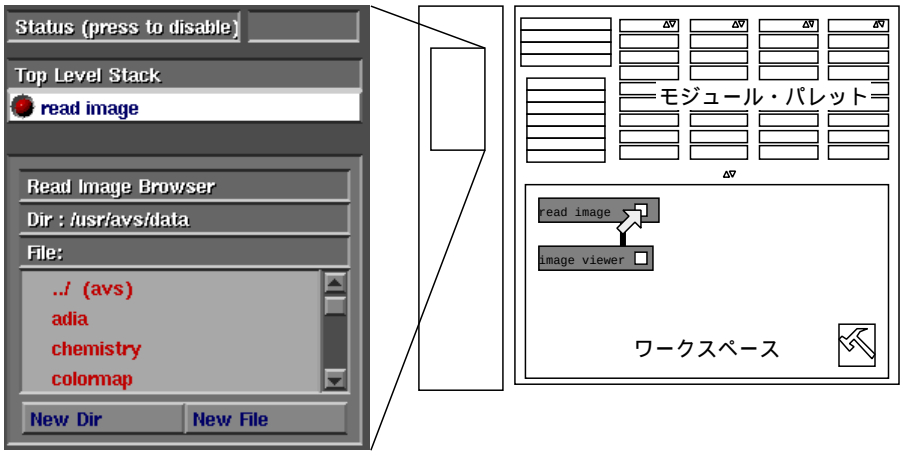


図 14: [read image ] モジュールのコントロール・パネル

図 13 のネットワークに Filter モジュールを追加してみましょう．Filter モジュールの追加例とその実行結果を，図 16，図 17，図 18 に示します．

## 6 おわりに

今回は AVS の利用について簡単な説明を行ないました．次回以降で，より詳細な使用法や実例を交じえた使用法を説明する予定です．AVS にはさまざまな利用方法がありますので，どんどん利用してみて下さい．

九州大学大型計算機センターの WWW ホームページからも，AVS についての説明を見ることができるようにしています．こちらも参考にしてください．九州大学大型計算機センターの WWW ホームページ

図 15: 図 13 のネットワークを実行 :mandrill.x を表示

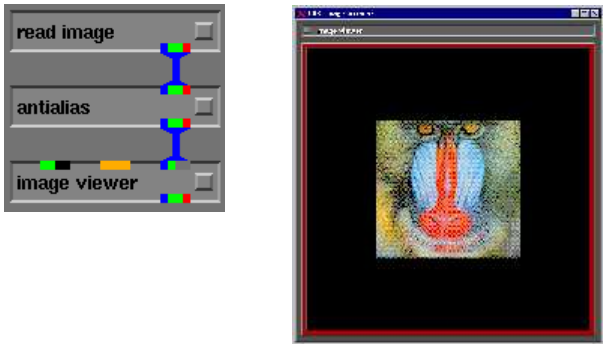


図 16: [antialias ] モジュールを追加．[antialias ] モジュールは画像を半分の大きさに変更します．

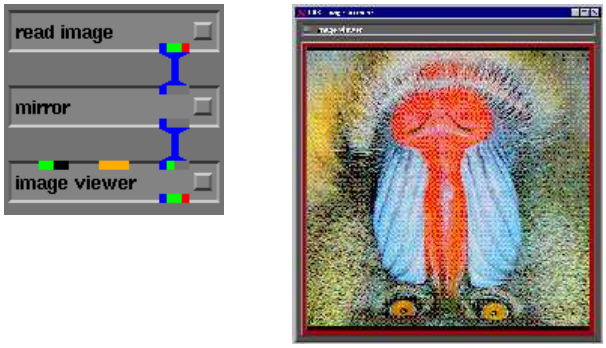


図 17: [mirror ] モジュールを追加．縦方向 (Y 軸) に反転．[mirror ] モジュールは，横 (X 軸) 方向または縦 (Y 軸) 方向に画像を反転するモジュールです．縦横の選択は，モジュール ・コントロール ・パレットで行ないます．

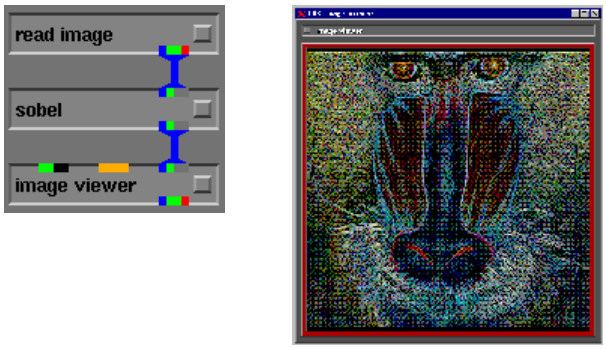


図 18:　[sobel ] モジュールを追加．[sobel ] モジュールは画像のエッジを見つけるモジュールです．

URL を以下に示します．

http://www.cc.kyushu-u.ac.jp/

参考文献の中に，インターネット上にある AVS に関連する WWW ページの URL を示しておきます．インターネットと WWW が利用可能な方はこちらも参考にして下さい．

## 参考文献

[1] “AVS ユーザーズ ・ガイド (ダイジェスト) 第 3 版,”
クボタコンピューター株式会社, 1992.

[2] “AVS ハンドブック – For The Better Visualization Environment–”,
クボタコンピューター株式会社, 1993.

[3] “AVS ハンドブック Vol.2 – For The Better Visualization Environment –”,
クボタコンピューター株式会社, 1993.

[4] “AVS ハンドブック Vol.3 – For The Better Visualization Environment – ,”
クボタコンピューター株式会社, 1993.

[5] “AVS User’s Guide, Release 4,”
ADVANCED VISUAL SYSTEMS Inc., 1992.

[6] “AVS Developters’s Guide, Release 4,”
ADVANCED VISUAL SYSTEMS Inc., 1992.

[7] “AVS Module Reference, Release 5,”
ADVANCED VISUAL SYSTEMS Inc., 1993.

[8] “ Advanced Visual Systems 社 WWW ページ, ”
http://www.avs.com/

[9] “ KTG クボタグラフィックス WWW ページ, ”
http://www.cssi.co.jp/hyper/KGT/kgt.html

[10] “富士通長野システムエンジニアリング, AVS サポートセンター WWW ページ, ”
http://www.fns.co.jp/avs/

[11] “大阪大学大型計算機センター AVS モジュール ftp サイト, ”
ftp://ftp.center.osaka-u.ac.jp/AVS-ug/avs_modules