

AVS/Express 利用の手引き

Table of contents

1. はじめに	3
1.1. 利用できるバージョン	3
1.2. 概要	3
1.3. マニュアル	4
2. TSUBAMEでの利用方法	5
2.1. AVS/Expressの起動	5
2.2. ライセンス使用状況の確認	6
3. 基本的な操作方法	7
3.1. 利用できるデータ	7
3.2. 基本操作手順	7
3.3. サンプルデータの可視化	8
4. 可視化ネットワーク例	14
4.1. 直行断面塗りつぶしコンター図	14
4.2. 直行断面線コンター図	15
4.3. 直行断面ベクトル図	16
5. AVS/Express PCEにて利用できるデータファイルフォーマット	18
5.1. AVS Fieldデータフォーマット	18
5.2. AVS Fieldデータ例	20
5.3. AVS UCDデータフォーマット	21
改訂履歴	26

1. はじめに

本書は、AVS/Express、AVS/Express PCE を東京工業大学学術国際情報センターの TSUBAME3 で利用する方法について説明しています。また、TSUBAME3を利用するにあたっては、[TSUBAME利用の手引き](#) もご覧下さい。利用環境や注意事項などが詳細に記述されております。サイバネット株式会社ではAVS/Express、AVS/Express PCEに関するWebページを公開しています。下記のアドレスを参照してください。

- ・製品紹介ページ
- ・分野別の可視化事例ページ

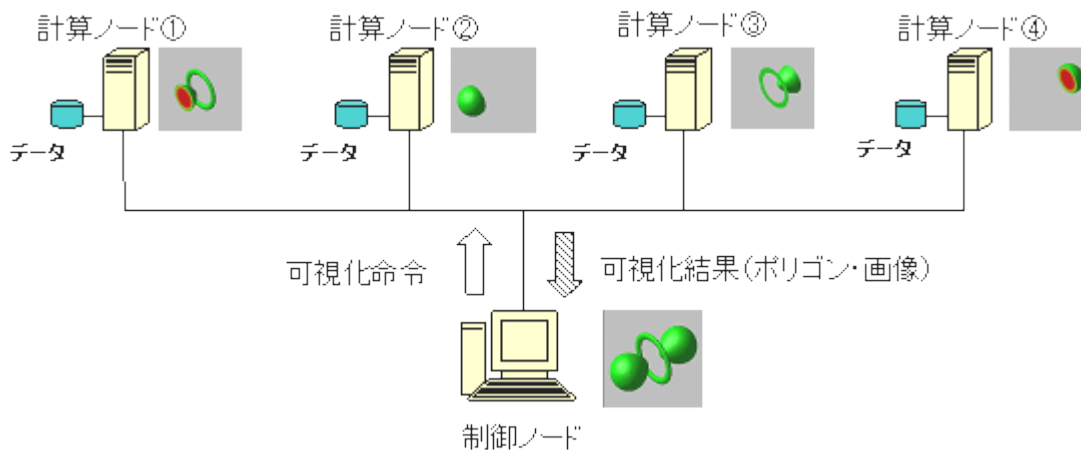
1.1. 利用できるバージョン

TSUBAME3で利用可能な最新バージョンについてはTSUBAME計算サービスWebサイトの[アプリケーション](#) ページをご確認ください。研究に支障がない限り、バグ修正の入っている最新版をご利用下さい。

1.2. 概要

AVS/Express PCE Parallel Cluster Edition はPCクラスタ対応版のAVS/Expressです。

クラスタ化された複数のLinuxマシンで、各マシンが持つ部分領域を可視化し、部分領域ごとに作成された可視化結果を合成し、全体領域での可視化結果を作成します。この部分領域のデータを読み込んで可視化処理を行うマシンを計算ノード、全体領域での可視化結果の作成および表示を行うマシンを制御ノードと呼びます。



- ・計算ノード データの読み込み 可視化まで、可視化処理の大部分を担当
- ・制御ノード 操作 可視化内容の定義 と可視化結果の表示のみ

制御ノードで定義した可視化内容は、可視化命令として計算ノードに送信されます。計算ノード側では、制御ノードより送信された可視化命令を受信し、定義された可視化内容に応じて可視化処理を実行します。計算ノードで可視化された結果は、奥行き情報付の画像、もしくはポリゴンで制御ノードに返信されます。 AVS/Express PCEの効果

PCクラスタを利用した並列計算やその他の大規模シミュレーションでは、出力される計算結果データも大規模である場合が多く、可視化を行う場合、これまでは、間引く、あるいは、部分領域のみを可視化するという手法が用いられていました。AVS/Express PCEでは、各計算ノードが持つ部分領域を可視化し、最終的な可視化結果は制御ノード上で作成するという構成になっています。そのため、並列計算や大規模シミュレーションの結果作成される大規模データを可視化する場合でも、高い精度を保ったまま、全体領域で可視化を実現することができます。

AVS/Express PCEには、以下の特長があります。

- ・可視化処理を複数の計算ノードで実行

データの読み込みからレンダリング処理まで、可視化処理のほとんどを、計算ノードで実行することにより大規模データの可視化を実現します。計算ノード複数台で処理することにより、1台では取り扱えないような大規模データを可視化することが可能です。

- ・分散データをそのまま可視化

部分領域ごとに分散して格納したデータ 各計算ノードのローカル・ディスクに保存されたデータ を制御ノードに集めることなく可視化することが可能です。共有ディスクに分割データを格納している場合でも、可視化することが可能です。可視化処理の自動化機能専用のスクリプトファイルを作成することにより、可視化処理を自動化することが可能です。

- ・様々な構成のシステムに対応可能

並列計算プログラム内での領域分割数と計算ノードの数が一致しない場合でも、可視化処理を繰り返すことにより、高精度・全体領域の可視化を実現します。

- ・AVS/Expressとのデータ互換

AVS/Express PCEはAVS/Expressのファミリー製品ですので、AVS/Express Viz、Developer等、他のAVS/Express製品とデータの互換性を保っており、データ資産をそのまま移行、流用することができます。操作方法も、ほぼ同じです。

1.2.1. 2種類の可視化方法 ポリゴンモードと画像モード

1.2.1.1. ポリゴンモード

ポリゴンモードは、AVS/Express PCE内で定義されるポリゴン情報を制御ノードに送信します。制御ノード上で表示オブジェクトを作成し、レンダリング処理を実行しますので、マウスによる幾何変換等のインタラクティブな操作が可能です。ただし、出力されるポリゴン数が多い場合、ポリゴン情報の転送と制御ノード上でのレンダリング処理に時間がかかる場合があります。

1.2.1.2. 画像モード

各計算ノードでデータの読み込みから、レンダリングまでの処理を行います。制御ノードに送信するのは奥行きが付加された可視化結果画像で、可視化した結果作成されるポリゴン数に関らず、常に一定量のデータが制御ノードに転送されます。ただし、制御ノードに送信されるのは、画像データなので、回転・拡大・縮小等の幾何変換を行う場合は、計算ノード側で、再度可視化処理を実行する必要があります。

1.3. マニュアル

マニュアルはTSUBAMEの次のディレクトリ下にあります。

アプリケーション名	ディレクトリ
AVS/Express 8.4	/apps/t3/sles12sp2/isl/avs/express84/runtime/help/ja/manual
AVS/Express PCE 8.4	/apps/t3/sles12sp2/isl/avs/express_pce84/doc

2. TSUBAMEでの利用方法

2.1. AVS/Expressの起動

[ログイン方法](#)を参考にログインノードにログイン後、[インタラクティブノード](#)を利用したX転送を参考にノードをX転送付きで確保して下さい。
以下以降の例では、全て計算ノードにログインした状態でいきます。

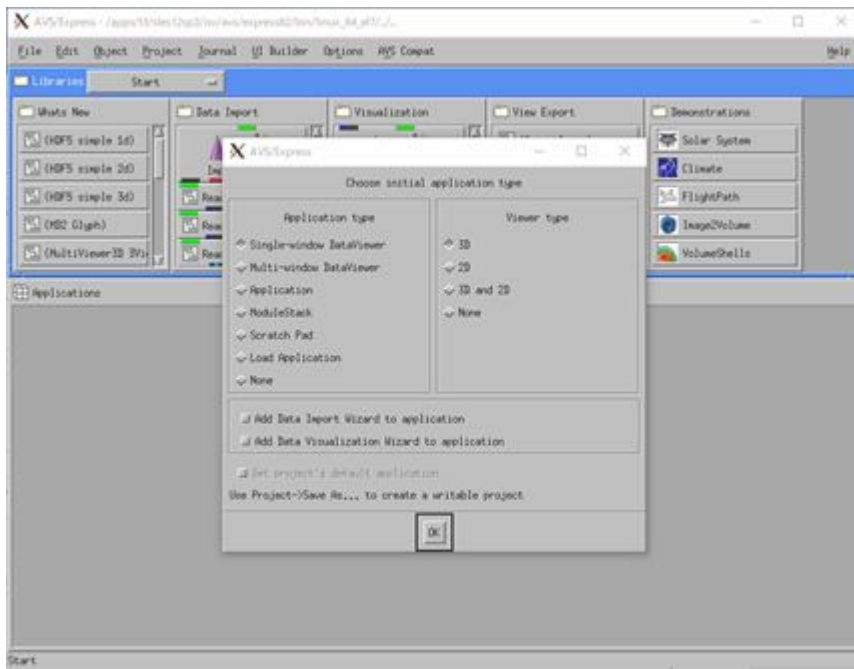
2.1.1. GUIの起動 AVS/Express

次のコマンドにより起動します。

```
$ module load avs/8.4
$ xp
```

起動に失敗する場合は、ハードウェアレンダリングを無効にして起動をお試しください。

```
$ module load avs/8.4
$ xp -nohw
```

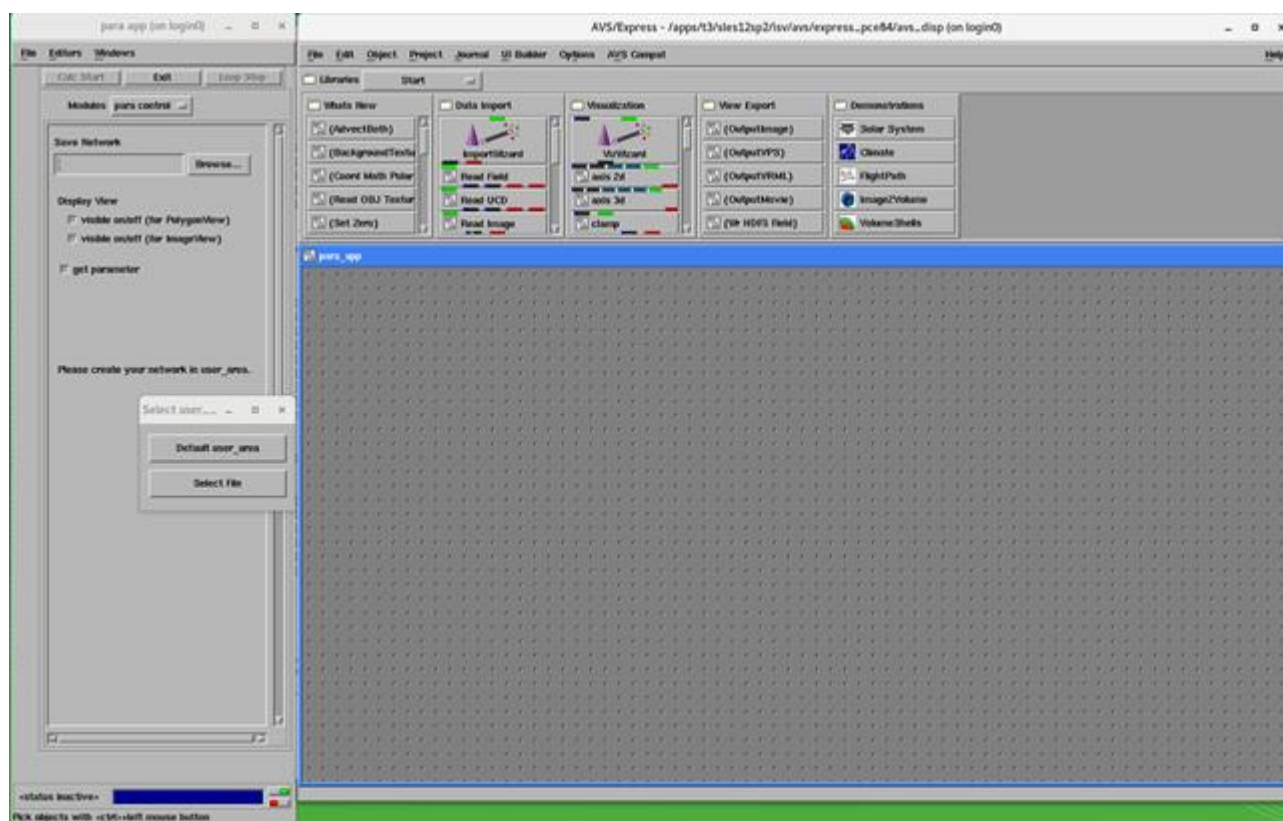


メニューバーのFile > Exitをクリックすると終了します。

2.1.2. GUIの起動 AVS/Express PCE

次のコマンドにより起動します。

```
$ module load avs
$ Xvfb :255 -ac -screen 0 1024x1024x24 &
$ para_start [コア数]
```



メニューバーのFile > Exitをクリックすると終了します。

2.2. ライセンス使用状況の確認

webブラウザで次のURLにアクセスすることにより、ライセンスの使用状況を確認できます。

<http://lice0.int.t3.gsic.titech.ac.jp:33333/STATUS>

ログインノード上でも以下のコマンドで確認できます。

```
w3m -dump http://lice0:33333/STATUS
```

3. 基本的な操作方法

ここから、AVS/Express PCE による可視化例となります。

3.1. 利用できるデータ

AVS/Express PCEを利用する場合、まず、以下のデータの準備をする必要があります。AVS/Express PCEで利用できるデータは、AVS/Expressと同様に大きく 2つのデータ構造に対応しています。

3.1.1. Fieldデータフォーマット 構造格子型

格子 メッシュ で表現できるデータで、1次元から3次元のデータに対応しています。



各座標値とその格子数 IJK で構成されます。格子が直交格子型の場合にも、必ず、両端点の座標値を指定する必要があります。差分法による解析結果などのデータに利用できます。

3.1.2. UCDデータフォーマット 非構造格子型

要素 セル で表現できるデータで、1次元から3次元のデータに対応しています。ポイント、ライン、3角形、4角形、4面体、6面体などの要素で構成されるデータです。各座標値とその要素タイプ、要素を構成する接続リストが必要です。物性値はその頂点上、もしくは要素の中心に定義することができます。有限要素法などのデータに利用できます。

※データの保存方法について

AVS/Express PCEで可視化を行うデータは、各計算ノードから参照可能なディスク上に作成する必要があります。その場合、全計算ノードのローカル・ディスクに同一名で保存するか、または、以下のように領域番号 ノード番号 を設定して、データを保存する必要があります。

領域番号 ノード番号 を設定してデータを保存する場合、「ファイル接頭辞」 「領域番号 ノード番号 %04d」 「ファイル拡張子」の形式でファイル名を設定します。

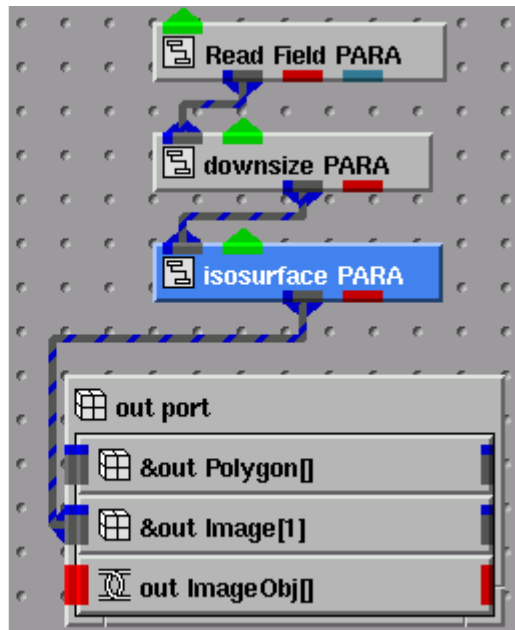
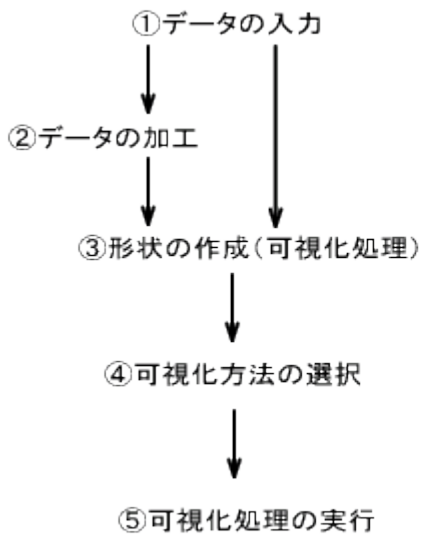
例 部分領域のFieldデータを、計算ノード 4台で可視化する場合

計算ノード①	sample.0001.fld	(部分領域①のデータ)
計算ノード②	sample.0002.fld	(部分領域②のデータ)
計算ノード③	sample.0003.fld	(部分領域③のデータ)
計算ノード④	sample.0004.fld	(部分領域④のデータ)

3.2. 基本操作手順

AVS/Express PCEでは、AVS/Expressと同様に、モジュールを組み合わせることでデータの可視化を行います。モジュールの接続方法など、基本的な操作はAVS/Expressと同じですが、AVS/Express PCEで可視化処理を実行する場合、並列対応を行っている専用のモジュールを使用する必要があります。

- AVS/ExpressPCE専用の並列対応モジュールは、特別なライブラリに登録され、全て、「_PARA」という名前が設定されています。例 Read_Field_PARA。
- 一般的にデータの可視化処理は以下のステップで行われます。



(1) データの入力

準備したデータを読み込みます。Read_Field_PARAやRead_UCD_PARAなどの読み込みモジュールを利用します。

(2) データの加工

必要に応じて、可視化処理の前準備を行います。例えば、複数のデータ成分から対象データを抜き出すextract_stalar_PARAモジュールや、ベクトル成分に対して大きさを計算するmagnitude_PARAモジュールなど、各種フィルタモジュールが準備されています。

(3) 形状の作成 可視化処理

等数値面を作成するisosurface_PARAモジュールや断面を作成するslice_orthobox_PARAモジュールが準備されています。

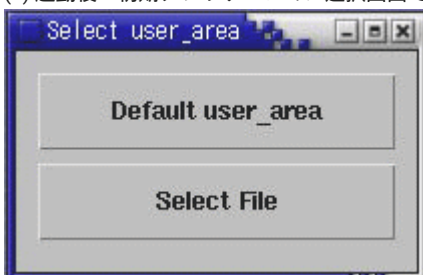
(4) 可視化手法の選択

可視化手法を選択します。ポリゴンモードと画像モードを選択することが可能です。

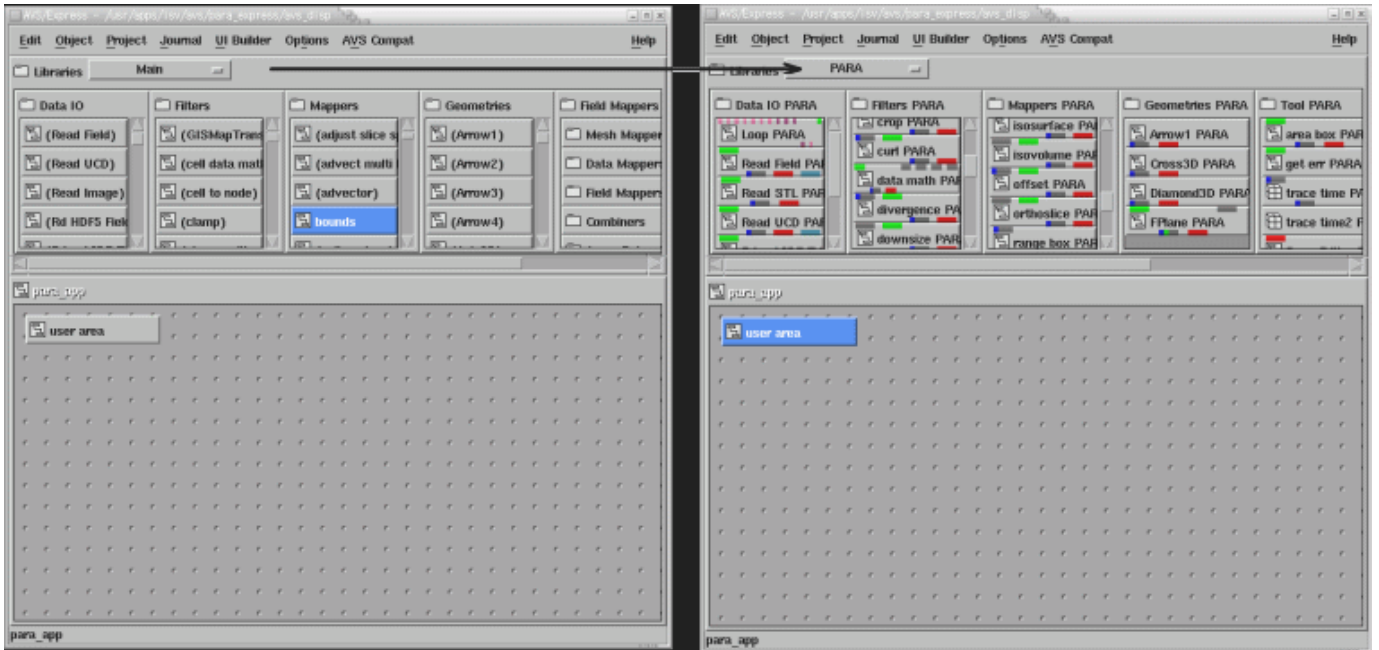
3.3. サンプルデータの可視化

サンプルデータを使った、簡単な可視化例を示します。以下に記述するサンプルデータは4領域のデータです。クラスタ情報ファイルを作成したのち、AVS/Express PCEを起動して下さい。

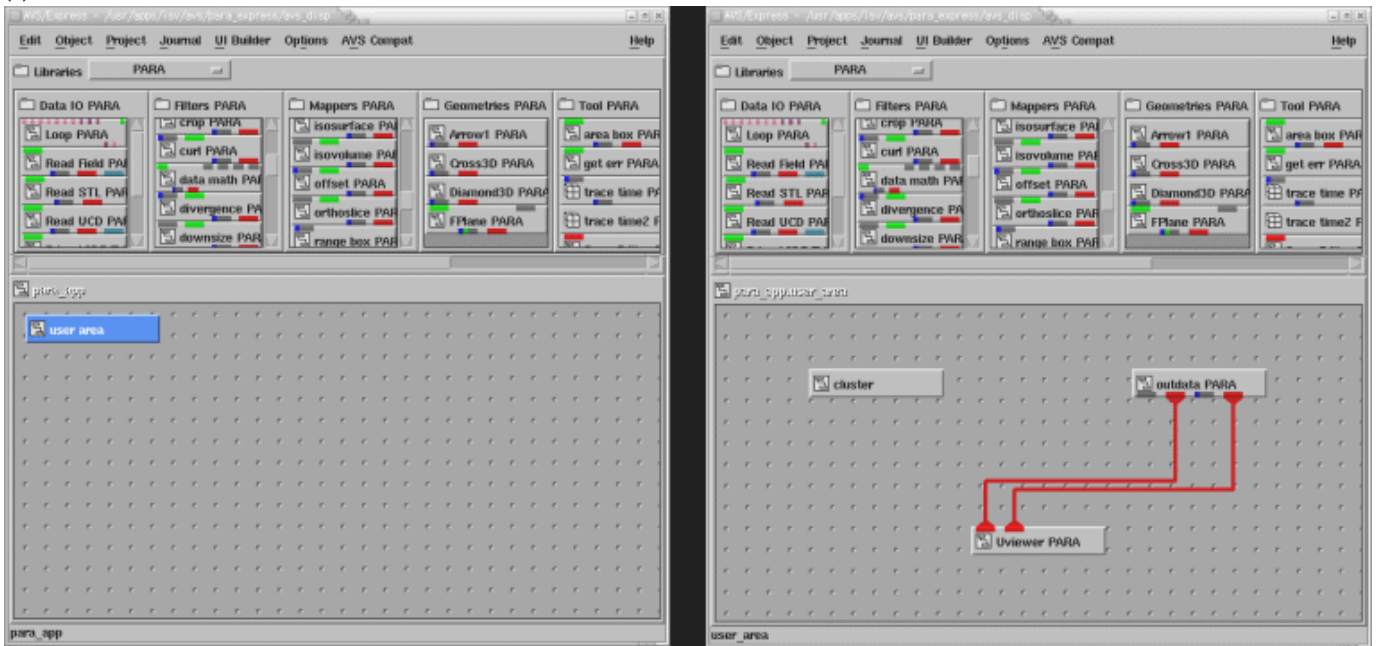
(1) 起動後、初期アプリケーション選択画面で「Default user_area」ボタンを押します。



(2) ライブラリページをMainからPARAに変更します。

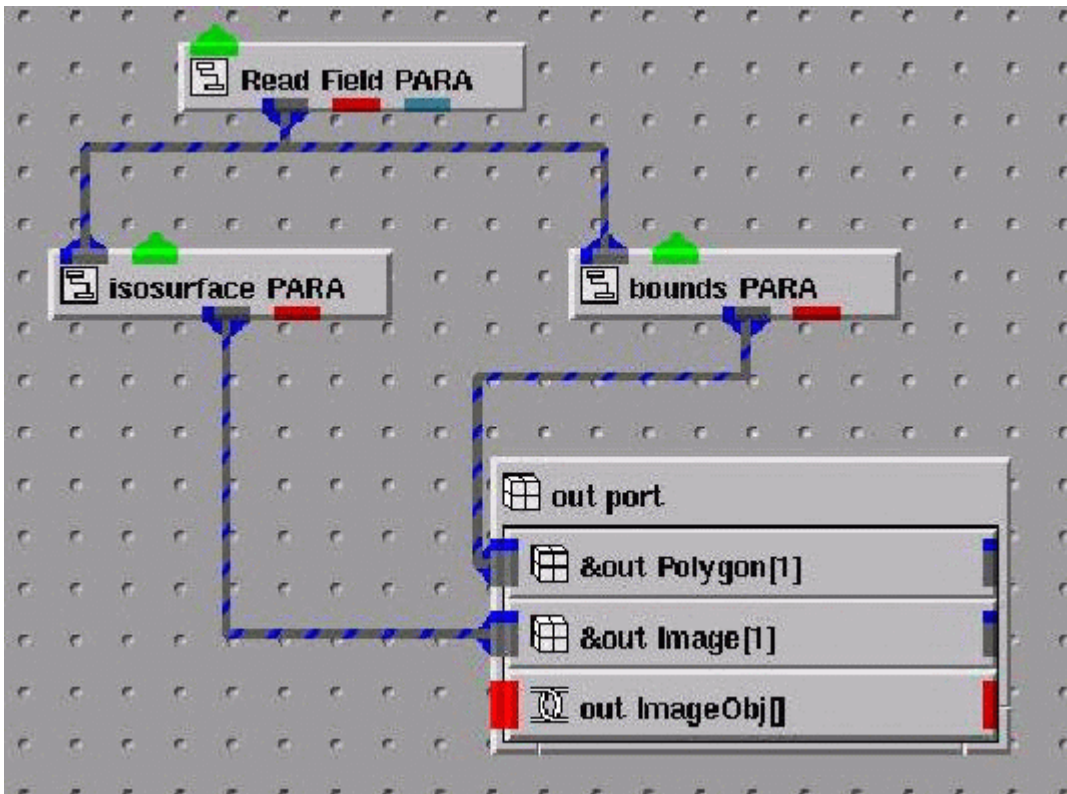


(3) user_areaモジュール→ clusterモジュールの順番で、モジュールをダブルクリックし、モジュールの内部に移動します。



(4) 以下のモジュールをインスタンスし、下図に示す可視化ネットワークを作成して下さい。

モジュール	ライブラリ	概要
Read_Field_PARA	Data_IO_PARA	Fieldデータの読み込み
bounds_PARA	Mappers_PARA	Fieldデータの領域枠(線・面)を作成
isosurface_PARA	Mappers_PARA	等値面を作成



<モジュールの接続方法>

マウスの左ボタンをポート モジュールの下の接続口 上で押したまま、接続するモジュールの方向に進めます。色が白く変わったら、マウスを放します。

<可視化手法の選択>

isosurface_PARAモジュールの出力ポートはout_Imageに接続されていますので、画像モードで可視化されます。 bounds_PARAモジュールの出力ポートはout_Polygonに接続されていますので、ポリゴンモードで可視化されます。

(5) データを読み込みます。 コントロール・パネル→「Editors」→「Modules」メニュー→「Read_Field_PARA」を選択すると、Read_Field_PARAモジュールのパラメータ設定画面が表示されます。「Browse...」ボタンを押し、ファイル・ブラウザを表示して、/apps/t3/sles12sp2/isy/avs/express_pce84/sample/field/4node/lobster.fldを選択して下さい。

注意点として、「Browse...」の隣にある入力ボックスに/apps/t3/sles12sp2/isy/avs/express_pce84/sample/field/4node/lobster.fldと入力してもファイル名として認識されません。必ずファイル・ブラウザを用いて、lobster.fldを選択し、OKボタンをクリックし確定してください。

なお、/apps/t3/sles12sp2/isy/avs/express_pce84/sample/field/4node/lobster.fldの実体ファイルは存在しません。実際は、lobster.0001.fld lobster.0004.fldが保存されており、これらは計算ノード4台で可視化するためのデータなのですが、AVS/Express PCE では、以下のような動作となります。

<制御ノード側>以下のように、存在しないファイル名を指定

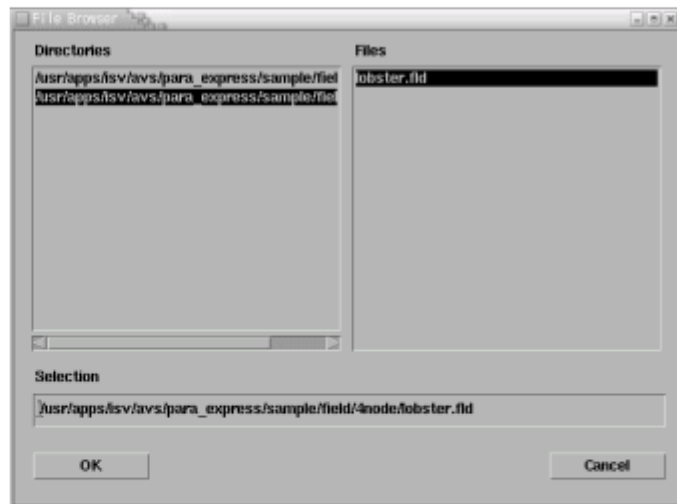
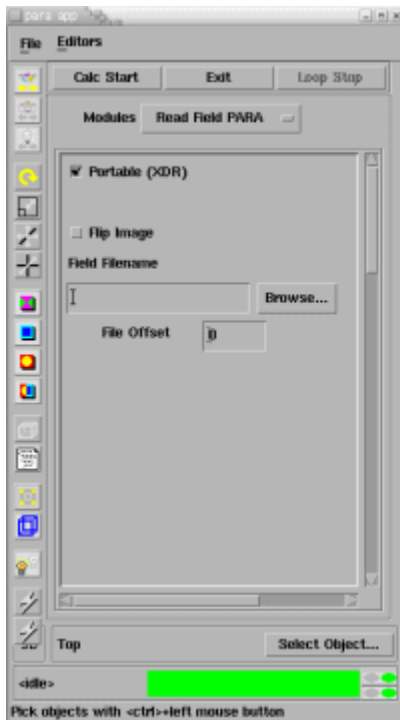
```
/apps/t3/sles12sp2/isy/avs/express_pce84/sample/field/4node/lobster.fld
```

<計算ノード側>

指定された/usr/apps/isy/avs/express_pce8.0B/linux_64_el5/express_pce80/sample/field/4node/lobster.fldというファイル名から、自分自身のマシン番号と一致するファイルの検索を行い、自動的に自分自身の担当領域のデータを読み込みます。

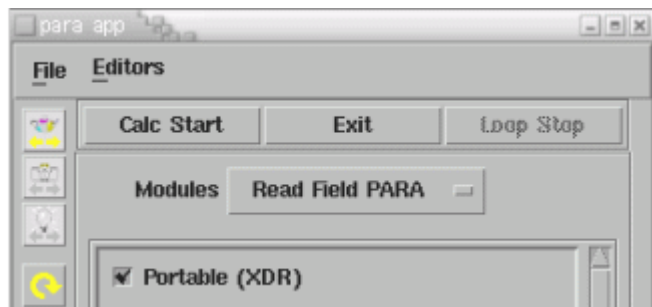
```
計算ノード①→lobster.0001.fld を読み込む。
計算ノード②→lobster.0002.fld を読み込む。
計算ノード③→lobster.0003.fld を読み込む。
計算ノード④→lobster.0004.fld を読み込む。
```

AVS Express PCEでは上記のようなファイル名変換機能が組み込まれておりますので、該当ディレクトリ下に、lobster.fld 番号のついていないファイル が存在しなくても問題はありません。

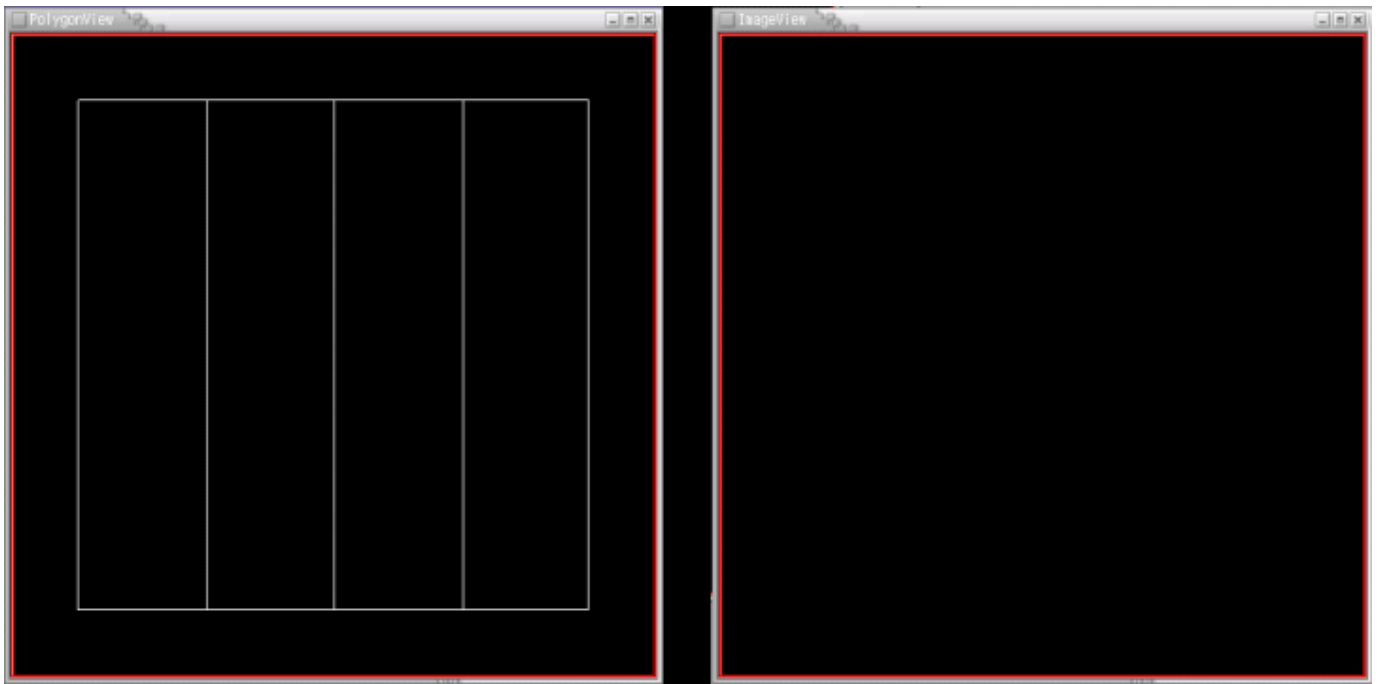


(6) 可視化処理を実行します。

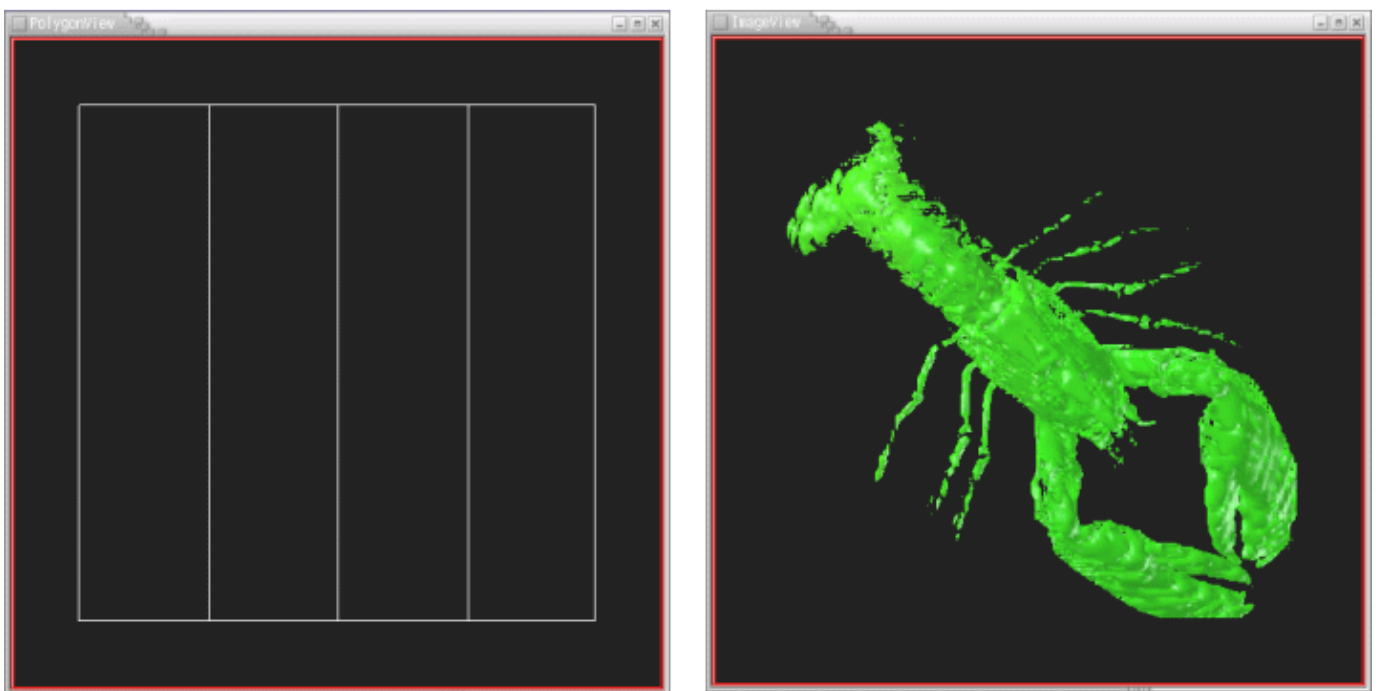
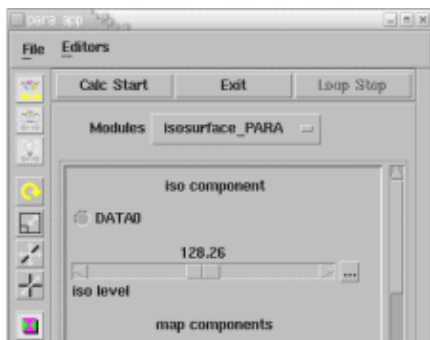
コントロール・パネル上部にある「Calc_Start」ボタンを押して下さい。



PolygonViewに境界枠が表示されます。



(7) 可視化パラメータの変更



コントロール・パネル→「Modules」メニュー→「isosurface_PARA」を選択すると、isosurface_PARAモジュールのパラメータ設定画面が表示されます。iso_levelスライダーに設定されている値を128に変更し、再度、「Calc_Start」ボタンを押して下さい。

Polygon Viewに境界枠、ImageViewに等値面が表示されます。PolygonViewはマウスで幾何変換することができます。操作方法はAVS/Expressと同じです。マウスで回転後、「Calc_Start」ボタンを押すことで、ImageViewの結果がその角度で更新されます。

4. 可視化ネットワーク例

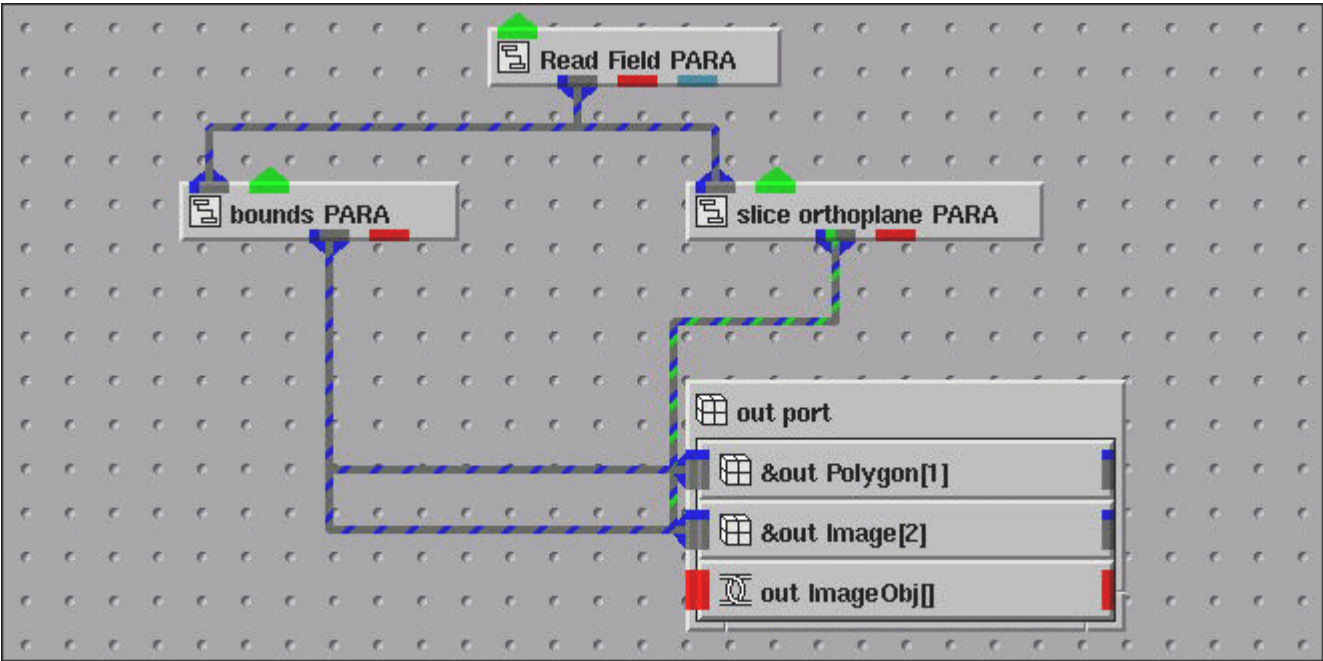
代表的なネットワーク例を示します。サンプルデータを使って操作してみることができます。なお、以下の例は全て4CPU使用した場合のサンプルです。

4.1. 直行断面塗りつぶしコンター図

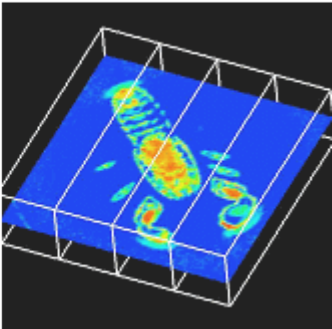
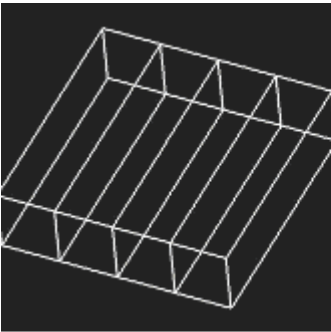
サンプルデータ

/apps/t3/sles12sp2/isy/avs/express_pce84/sample/field/4node/lobster.xxxx.fld

モジュール	ライブラリ	概要
Read_Field_PARA	Data_IO_PARA	Fieldデータの読み込み
bounds_PARA	Mappers_PARA	Fieldデータの領域枠(線・面)を作成
slice_orthoplane_PARA	Mappers_PARA	直行断面を作成



ネットワーク図



表示例 左側がPolygonView、右側がImageView

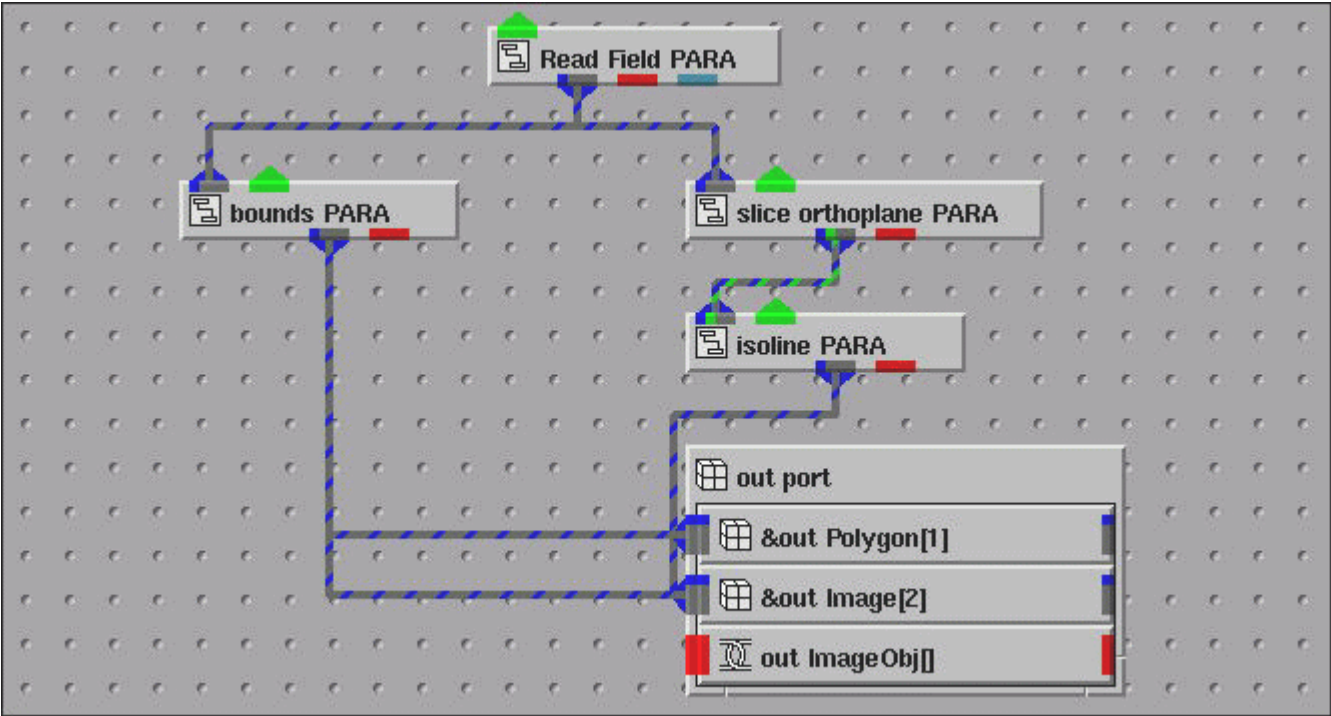
bounds PARA out_Polygon、 bounds PARA out_Image、 slice_orthoplane PARA out_Imageに接続します。コントロール・パネルにあるパラメータでIJKの方向(axis=0/1/2)と断面位置を指定できます。

4.2. 直行断面線コンター図

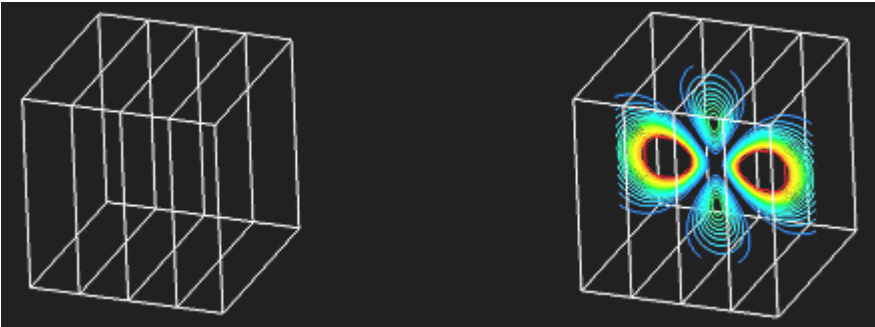
サンプルデータ

```
/apps/t3/sles12sp2/isv/avs/express_pce84/sample/volume/4node/hydrogen.xxxx.fld
```

モジュール	ライブラリ	概要
Read_Field_PARA	Data_IO_PARA	Field データの読み込み
bounds PARA	Mappers_PARA	Fieldデータの領域枠(線・面)を作成
slice_orthoplane PARA	Mappers_PARA	直行断面を作成
isoline PARA	Mappers_PARA	等値線 線コンター を作成



ネットワーク図



表示例 左側がPolygonView、右側がImageView

bounds PARA out_Polygon、 bounds PARA out_Image、 isoline PARA out_Imageに接続します。

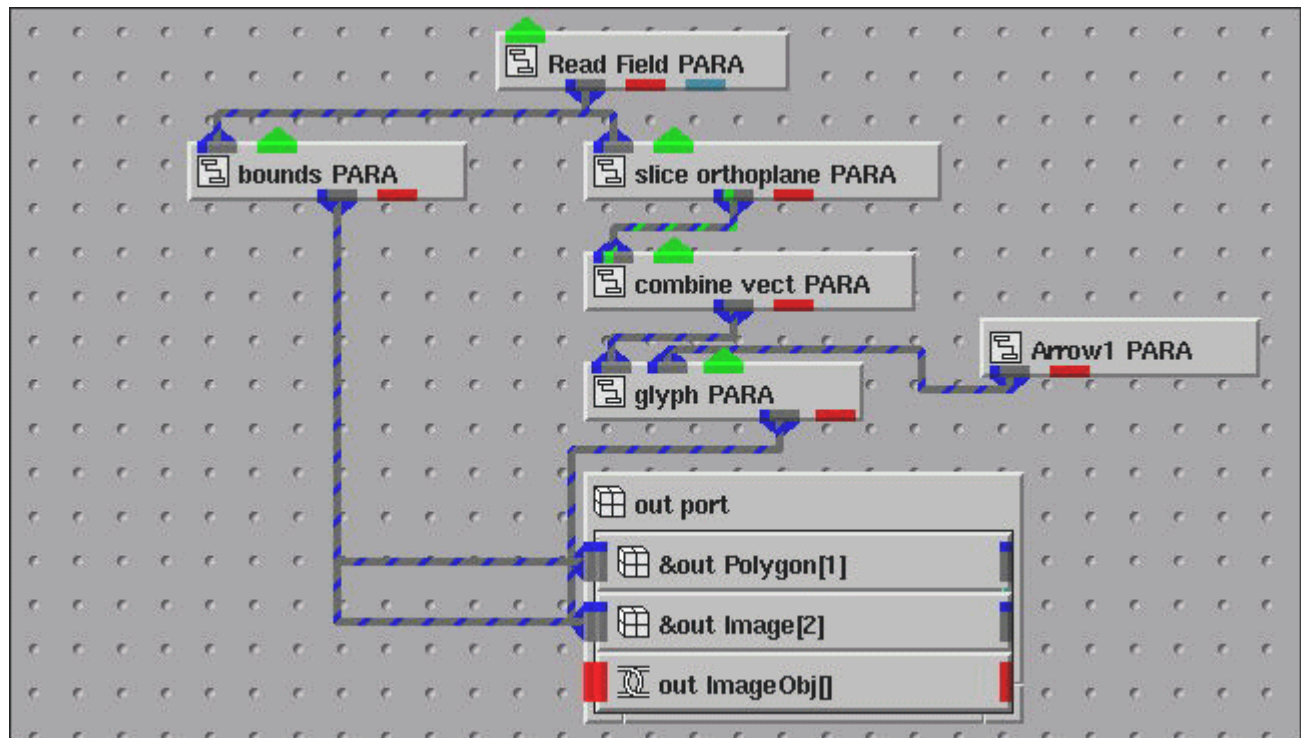
コントロール・パネルにあるパラメータでIJKの方向(axis=0/1/2)と断面位置、等値線の本数(number of contours)等を指定できます

4.3. 直行断面ベクトル図

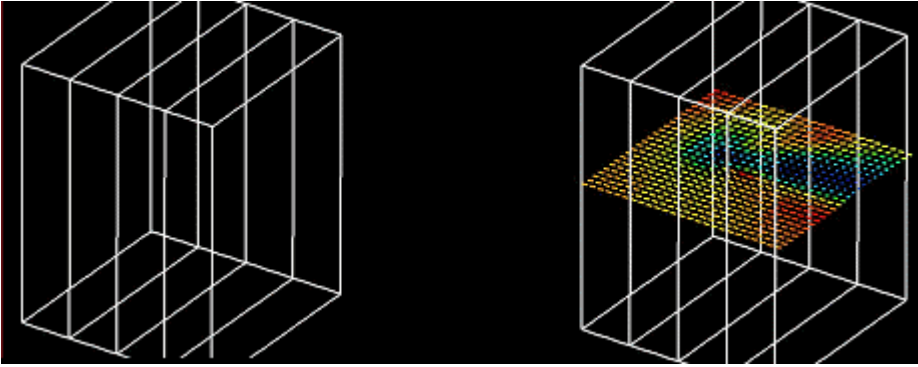
サンプルデータ

```
/apps/t3/sles12sp2/isy/avs/express_pce84/sample/vector/4node/wind.xxxx.fld
```

モジュール	ライブラリ	概要
Read_Field_PARA	Data_IO_PARA	Fieldデータの読み込み
bounds PARA	Mappers_PARA	Fieldデータの領域枠(線・面)を作成
slice_orthoplane PARA	Mappers_PARA	直行断面を作成
combine_vect PARA	Filters_PARA	ベクトルデータを作成
Arrow1 PARA	Geometries_PARA	矢印形状
glyph PARA	Mappers_PARA	ベクトル図を作成



ネットワーク図



表示例 左側がPolygonView、右側がImageView

bounds PARA out_Polygon、bounds PARA out_Image、glyph PARA out_Imageに接続します。コントロール・パネルにあるパラメータで断面のIJK方向(axis=0/1/2)、断面位置、グリフ(矢印)のスケール等を指定できます。

5. AVS/Express PCEにて利用できるデータファイルフォーマット

AVS/Express PCEで可視化を行う時に使用できるデータは、大きく分けて以下の2つのフォーマットが用意されています。

- AVS Field

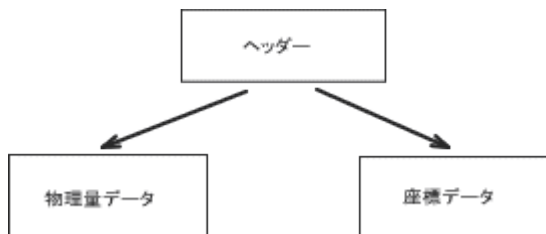
2次元や3次元の構造格子のデータを扱うフォーマットです。

- AVS UCD

Unstructured Cell Dataの略で、非構造格子型のデータを扱うフォーマットです。

5.1. AVS Fieldデータフォーマット

AVS Fieldフォーマットはヘッダー部と物理量データ部、形状データ部の3つのパートで作成されます。この3つのパートを一つのファイルとして取り扱うことができますが、作成するのに面倒なので、別ファイルとして取り扱うこともできます。形状が等間隔直交メッシュの場合は、形状データは必要ありません。データ部に書いてある内容をヘッダー部で説明するような形式になります。データ部はASCII形式でもバイナリ形式でも取り扱うことができます



5.1.1. ヘッダファイル・パラメータ

- ヘッダファイルのファイル名には必ず拡張子.fldが必要です。
- ヘッダファイルの最初の行は#AVSと書き込みます。

(2行目からの#はコメントですので#の後に続く文字は全て無視されます)

- ndim = 値(必須)

...フィールドの計算空間の次元数

- dim1 = 値(必須)
- dim2 = 値(必須、ndimの値に依存)
- dim3 = 値(必須、ndimの値に依存)

...各軸の次元サイズ

- nspace = 値 必須

...物理空間の次元数。データが存在する空間が 次元(x,y平面)であるか、3次元(x,y,z)であるかを記述します。

- veclen = 値(必須)

...各フィールド要素のデータ数

- data = {byte,integer,float,doubleの中の 1つが必須}

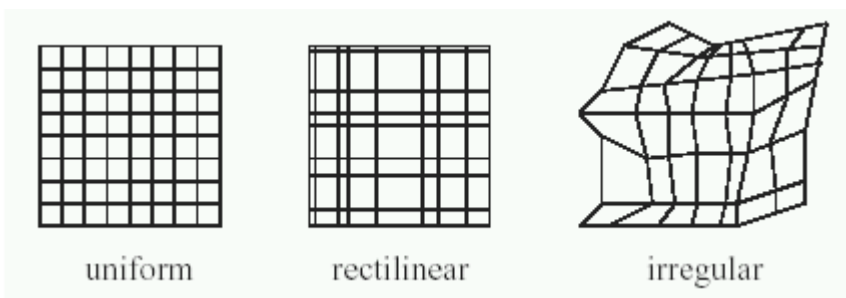
...各フィールド要素のデータ数

- data = {byte,integer,float,doubleの中の 1つが必須}

...データ・タイプ

- field = {uniform,rectilinear,irregularの中の 一つが必須}

...フィールドのタイプ



label = 文字列 オプション

...各要素にタイトルを付ける

- unit = 文字列(オプション)

...各要素に単位を付ける

以下のvariable・coordにはサブパラメータがあります

- variable(必須)

...データファイルのフォーマットを指定

- coord 必須 但し uniformフィールドタイプには不要

...座標情報のフォーマットを指定

variable・coordのパラメータコマンド

5.1.1.1. パラメータ

- n

データ要素数.座標値数を指定

- file

データ又は座標値が入っているファイル名の指定

- filetype,ascii,unformatted,binary

5.1.1.2. コマンド

- skip = n

asciiファイルでは読み飛ばす行数を指定 unformatted,binary では読み飛ばすバイト数を指定(skipのデフォルトは0)

- offset = m

最初のデータを読み始める前に読み飛ばすカラム数を指定 asciiファイルの時のみ使用 (offsetのデフォルトは0)

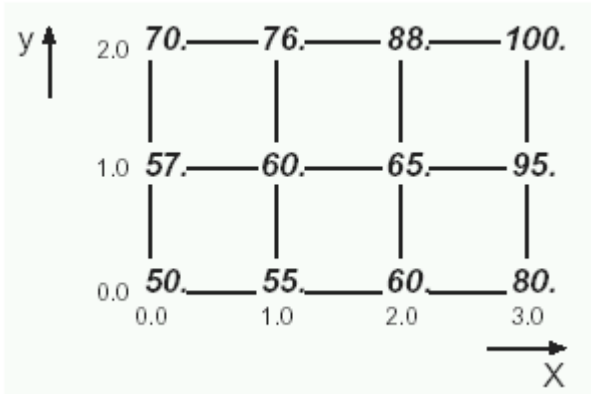
- stride = p

次のデータを読み込むための読み飛ばすカラム数を指定 (binaryと書式なしの場合はp x データ・サイズ) (stride のデフォルトは1)

5.2. AVS Fieldデータ例

5.2.1. スカラーデータ例

次に示すデータのFieldデータを作成します。



小さい文字の数値は座標値で、イタリックの文字はノード上の物理量(スカラー値)です。このデータの形状データおよび物理量のデータを以下に示します。

形状データ:scalar.geo

```
0.0  1.0  2.0  3.0  0.0  1.0  2.0  3.0  0.0  1.0  2.0  3.0
0.0  0.0  0.0  0.0  1.0  1.0  1.0  1.0  2.0  2.0  2.0  2.0
```

物理量データ:scalar.dat

```
50.  55.  60.  80.  57.  60.  65.  95.  70.  76.  88.  100.
```

上記のデータに対するAVS Fieldデータを以下に示します。

AVS Fieldデータ:scalar.fld

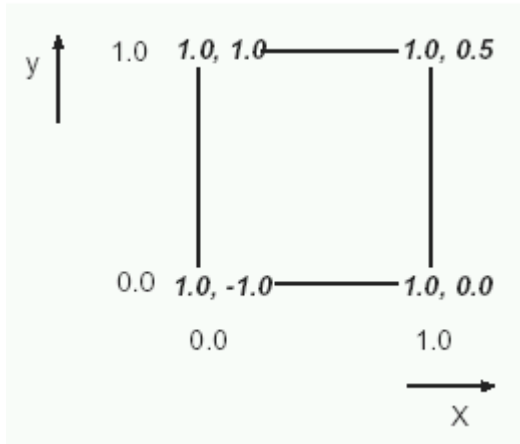
```
1  #AVS
2  ndim=2
3  dim1=4
4  dim2=3
5  nspace=2
6  veclen=1
7  data=float
8  field=irregular
9  coord 1 file=scalar.geo filetype=ascii skip=0 offset=0
10 coord 2 file=scalar.geo filetype=ascii skip=1 offset=0
11 variable 1 file=scalar.dat filetype=ascii skip=0 offset=0
```

各行の意味は以下のようになっています。

- 1行目 行目は必ず#AVSと書きます。
- 2行目 次元格子データなので、ndim=2となります。
- 3、4行目 i方向の次元数が4、j方向の次元数が3なので、dim1=4,dim2=3となります。
- 5行目 次元平面のデータなのでnspaceの値は2になります。
- 6行目 物理量が 種類だけなので、veclenの値は1になります。
- 7行目 物理量のデータが浮動小数点のデータなので、dataの指定はfloatになります。
- 8行目 このデータのタイプはuniformになりますが、各データのX,Y,Z座標値を与えるので、irregular形式として取り扱います。
- 9行目 形状ファイルavs.geoの中に1行目から1レコードで書かれてあり読み飛ばす必要はないので、skip=0,offset=0となります。
- 10行目 形状ファイルavs.geoの中に2行目から1レコードで書かれてあり始めの一行目だけを 読み飛ばすので、skip=1,offset=0 となります。
- 11行目 物理量のデータファイル avs.dat の中に、1行目から1レコードで書かれてあり 読み飛ばす必要はないので、skip=0,offset=0となります。

5.2.2. ベクトルデータ例

下記に示すデータのFieldデータを作成します。



小さい文字の数値は座標値で、イタリックの文字はノード上の物理量(ベクトル値)です。このデータの形状データおよび物理量のデータを以下に示します。

形状データ:vector.geo

```
0.0  1.0  1.0  0.0
0.0  0.0  1.0  1.0
```

物理量データ:vector.dat

```
1.0  1.0  1.0  1.0
-1.0  0.0  0.5  1.0
```

上記のデータに対するAVS Fieldデータを以下に示します。

AVS Fieldデータ vector.fld

```
1  #AVS
2  ndim = 2
3  dim1 = 2
4  dim2 = 2
5  nspace = 2
6  veclen = 2
7  data = float
8  field = irregular
9  coord 1 file=vector.geo filetype=ascii skip=0 offset=0
10 coord 2 file=vector.geo filetype=ascii skip=1 offset=0
11 variable 1 file=vector.dat filetype=ascii skip=0 offset=0 stride=1
12 variable 2 file=vector.dat filetype=ascii skip=1 offset=0 stride=1
```

各行の意味は以下のようになっています。 スカラーデータと違う部分に関してのみ説明します

3,4行目 i方向の次元数が2、j方向の次元数が2なので、dim1=2,dim2=2となります。

6行目 物理量は、U,Vの組みなので、veclenの値は2になります。

11行目 物理量のデータファイルavs.datの中に、 1行目から1レコードで書かれてあり 読み飛ばす必要はないので、skip=0,offset=0となります。

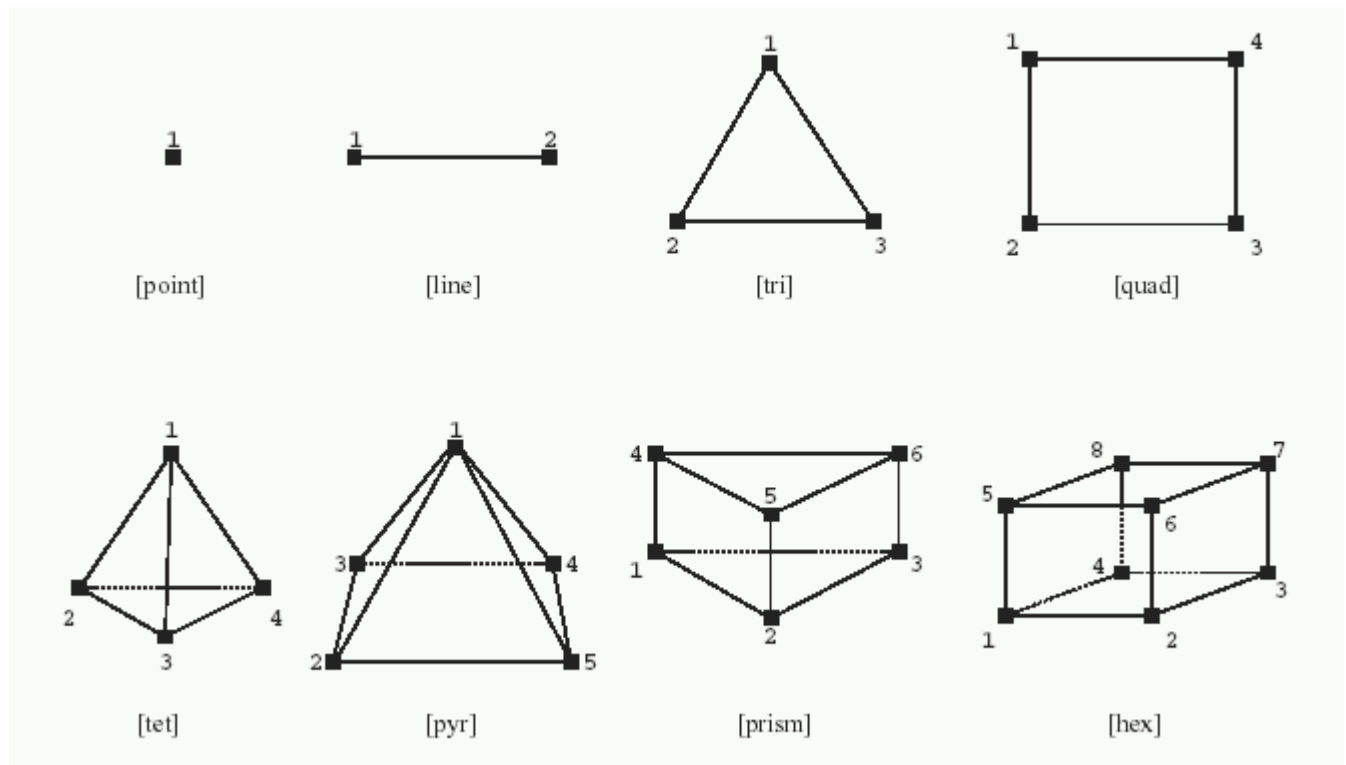
12行目 物理量のデータファイル avs.dat の中に、2行目から1レコードで書かれてあり 一行目だけを読み飛ばすのでskip=0,offset=0となります。

5.3. AVS UCDデータフォーマット

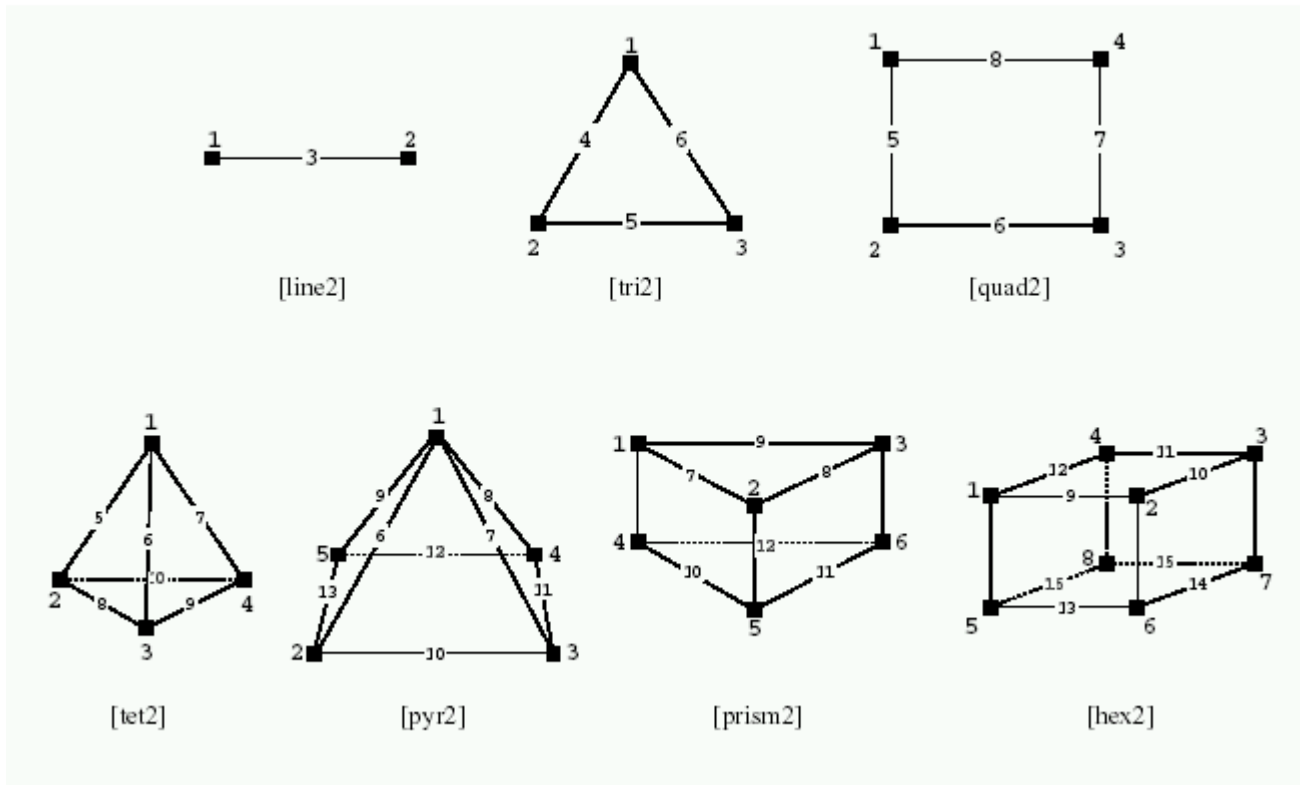
有限要素法等で計算された結果は、フィールドデータフォーマットと違い、構造格子状になっていません。三角形や、四角形などの要素型になっています。このようなデータは、UCDデータフォーマットで取り扱います。UCDデータフォーマットはセル(要素)とノード(節点)で構成され、ノードにデータを与えたり、セルにデータを持たせることができます。

5.3.1. セル 要素 の種類

データに使用できるセルタイプには次のものがあります。各ノードに示されている番号は、ノードの接続の順番を示しています。



上に示すセルに対して、2次要素としてノード間に対しても情報を持つことができます。



5.3.2. データの形式

UCDのファイルフォーマットは、ノードの座標値、データ等の情報を次に示すようにアスキー形式で作成します。

```
#コメント
<ステップ数>
<サイクルタイプ>
step1 <コメント>
<節点数> <要素数>
<節点番号1> <X> <Y> <Z>
<節点番号2> <X> <Y> <Z>
:
<要素番号1> <マテリアル番号> <要素タイプ> <要素の構成>
<要素番号2> <マテリアル番号> <要素タイプ> <要素の構成>
:
:
<節点データ数> <要素データ数>
<節点データ成分数> <第1成分ベクトル長> <第2成分ベクトル長>
<第1成分名>, <第1成分単位名>
<第2成分名>, <第2成分単位名>
:
:
<節点番号1> <節点データ1> <節点データ1>
<節点番号2> <節点データ2> <節点データ2>
:
:
<要素データ成分数> <第1成分ベクトル長> <第2成分ベクトル長>
<第1成分名>, <第1成分単位名>
<第2成分名>, <第2成分単位名>
:
:
<要素番号1> <要素データ1> <要素データ2> ...
:
:
step2 <コメント>
<節点数> <要素数>
<節点番号1> <X> <Y> <Z>
<節点番号2> <X> <Y> <Z>
:
:
<節点データ数> <要素データ数>
<要素データ成分数> <第1成分ベクトル長> <第2成分ベクトル長>
<第1成分名>, <第1成分単位名>
```

```

:
step3 <コメント>
:
:

```

ファイル形式の説明は次の通りです。

1. # データに関する記述の前((2)の部分以前)に存在する#で始まる行は、コメント行として使用することができます。
2. <ステップ数> ファイル内に書かれているデータのステップ数を記述します。
3. <サイクルタイプ> 繰り返しのタイプ。data,geom,data_geomから指定します。geomが指定されたときは(4) (7)の項目が繰り返され、dataが指定されたときは(4)から次のステップまでの項目が繰り返されます。data-geomが指定されれば、(4)から次のステップまでの項目が繰り返されます。
4. stepN 各ステップの先頭にはステップ番号を記述します。第 ステップはstep1,第 ステップはstep2...と記述する必要があります。ステップ番号の後には、そのステップに対するコメントを記述することができます。Read_UCDモジュールはこのコメントを出力するので、ビューワー・ウィンドウに表示させることも可能です。
5. <節点数>,<要素数>節点数および要素数。サイクルタイプがgeomまたはdata_geomの場合はステップ毎に記述されますが、geomの場合には、第 ステップと同じ値を第2ステップ以降に記述する必要があります。
6. <節点番号>、 „節点番号と節点の座標値を全ての節点に関して記述します。節点番号は整数値です。連続した番号でなくても構いませんが、重複しないようにして下さい。
7. <要素番号>整数値です。連続した番号でなくても構いませんが、重複しないようにして下さい。
8. <マテリアル番号>

要素をグループ分けするのに用いる整数値です。

・<要素タイプ>

要素のタイプを指定します。要素にはコーナーのみに節点を持つ 次要素と中間節点を持つ 次要素があります。

・<要素の構成>

要素を構成する節点の番号を記述します。この番号は(6)で記述したものに対応しています。

以上について、すべての要素に関して記述します。

5.3.3. スカラーデータ例

下記に示すデータのFieldデータを作成します。小さい文字の数値は座標値で、イタリックの細文字はノードの番号です。イタリックの太文字はセル上の物理量,スカラー値)です。このデータは、節点上ではなく、要素上に物理量が存在する例です。この図のUCDデータを以下に示します。

scalar.inp

```

1  1
2  data
3  step1
4  8  3
5  2  0.  0.  0.
6  2  1.  0.  0.
7  3  2.  0.  0.
8  4  3.  0.  0.
9  5  0.  1.  0.
10 6  1.  1.  0.
11 7  2.  1.  0.
12 8  3.  1.  0.
13 2  1 quad 1 2 6 5
14 2  1 quad 2 3 7 6
15 3  1 quad 3 4 8 7
16 2  1
17 2  1
18 scalar,
19 1 0.1
20 2 0.4
21 3 0.3

```

各行の意味は以下のようになっています。

1行目 ステップのデータなので、ステップ数は となります。

2行目 ステップ数が1なので、この部分はdataでもgeomでもdata_geomでも構いません。

3行目 全ノード数が8、セル数が3になります。

5-12行目 ノード番号と、ノードのX,Y,Z座標値を示しています。

13-15行目 各セルのセル番号と、セルタイプ、ノードの接続順番を指定します。

16行目 このデータは、節点上ではなく、要素上に1種類の物理量が存在するデータなので、節点データ数は0、要素データ数が1となります。

17行目 1種類のスカラー値のため、データ成分数が1、第1成分ベクトル長は1となります。

18行目 成分名 任意の文字列 scalar と、その単位名(任意の文字列)。単位名は、つけなくても構いませんが、成分名の後のコンマ","は必ずつけて下さい。

19-21行目 要素番号、要素データを各要素毎に記述します。

5.3.4. ベクトルデータ例

下記に示すデータのFieldデータを作成します。小さい文字の数値は座標値で、イタリックの細文字はノードの番号です。イタリックの大文字はセル上の物理量,ベクトル値です。このデータは、要素上ではなく、節点上に物理量が存在する例です。この図のUCDデータを以下に示します。

vector.inp

```

1  1
2  data
3  stepl
4  4  1
5  2  0.  0.  0.
6  2  1.  0.  0.
7  3  0.  1.  0.
8  4  1.  1.  0.
9  1  1 quad 1 2 4 3
10 3  0
11 1  3
12 Vector-component,
13 1 0.0 -0.5 0.0
14 2 0.5 -0.5 0.0
15 3 0.5 0.0 0.0
16 4 0.25 0.5 0.0

```

各行の意味は以下のようになっています。

1行目 1ステップのデータなので、ステップ数は1となります。

2行目 ステップ数が1なので、この部分はdataでもgeomでもdata_geomでも構いません。

3行目 全ノード数が4、セル数が1になります。

5-8行目 ノード番号と、ノードのX,Y,Z座標値を示しています。

9行目 各セルのセル番号と、セルタイプ、ノードの接続順番を指定します。

10行目 このデータは、要素上ではなく、節点上に 種類のベクトル値が存在するデータなので、節点データ数は3、要素データ数が0となります。

11行目 種類のベクトル値のため、データ成分数が1、第 成分ベクトル長は3となります。

12行目 成分名(任意の文字列:Vector-component)と、その単位名(任意の文字列J。単位名は、つけなくても構いませんが、成分名の後のコンマ","は必ずつけて下さい。

13-16行目 節点番号、節点データを各節点毎に記述します。

改訂履歴

改定日付	内容
2019/08/19	mkdocs版作成
05/23/2019	誤記修正
2018/10/26	「2.2 ライセンス使用状況の確認」を修正
2018/07/23	「2.1.5. GUIの起動 AVS/Express PCE 」にXvfbの指定を追加
2018/02/28	2.1.4にソフトウェアレンダリングモードでの起動方法を追記
2017/09/26	2.1.4および2.1.5のログイン方法を追記
2017/09/21	初版