５－５ Field

５－５－１ ファイルの拡張子

ファイルの拡張子は .f l d です。

５－５－２ 読み込みモジュール

読み込みモジュールは Read_Field です。

５－５－３ フォーマット概略

Field データは、1 次元配列、2 次元配列、3 次元配列で定義されるは構造型のフィールド・データです。

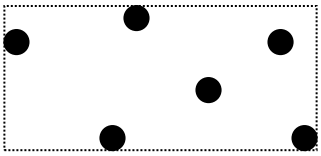

図５-4　計算空間が 1 次元、物理空間が 2 次元の Field データ

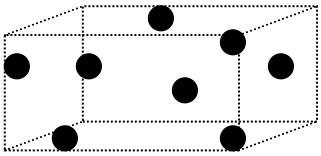

図５-5　計算空間が 1 次元、物理空間が 3 次元の Field データ

図５-6　計算空間が 2 次元、物理空間が 3 次元の Field データ

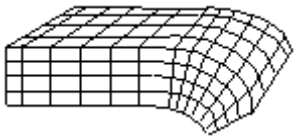

図５-7　計算空間が 3 次元、物理空間が 3 次元の Field データ

Field データファイルのフォーマットは図５-8 のようなデータの情報を記述するヘッダー部分、データ値が格納されるデータ部分、座標値が格納される座標部分から構成されます。

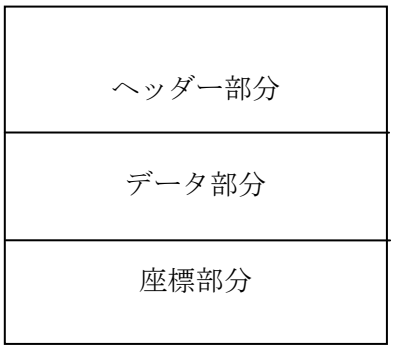

図５-8　Field データファイルの構成

さらにデータ部分、座標部分に関しては、直接バイナリで記述するタイプと、他のデータファイルを参照し、そのポインタだけを記述するタイプがあります。

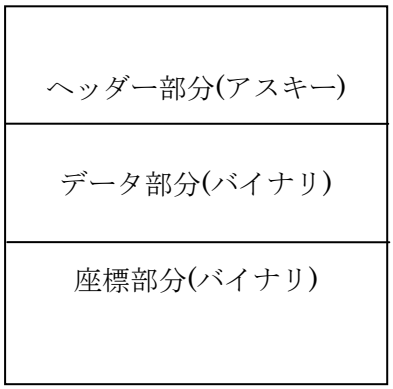

図 5-9-a　データを直接記述するタイプ

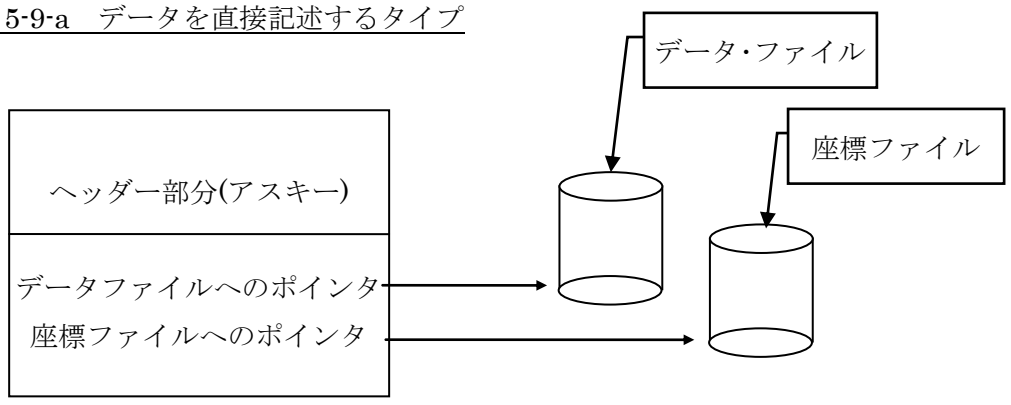

図５-9-b　データを参照するタイプ

５－５－４ フォーマット(ヘッダー部分)

ヘッダー部分では、Field データを定義するために必要な情報をアスキーで記述します。その情報を記述するには幾つか用意されたキーワードに値をセットします。

```
# AVS field file ---------- (1)
#
# This is a sample for "field 3D 3-vector uniform" ---------- (2)
#
ndim    =  3 ---------- (3)
dim1    =  30
dim2    =  20 ---------- (4)
dim3    =  10
nspace  =  3 ---------- (5)
veclen  =  3 ---------- (6)
data    =  float ---------- (7)
field   =  uniform ---------- (8)
label   =  vec_x vec_y vec_z ---------- (9)
```

図５-10　ヘッダー部の記述例

(1) **# AVS field file**

Field データファイルの先頭には必ずこのキーワードが必要です。

注) 大文字、小文字、空白を正確に記述しなければなりません。

(2) **#**

2 行目以降の#で始まる行はコメント行と見なされます。

(3) **ndim=<数値>**

計算空間での次元数を定義します。
離散的にデータが分布し、隣合うデータに規則性がないタイプは１次元として定義し、**ndim=1** とします。２次元格子として定義ができる場合は **ndim=2** 、３次元格子の場合は **ndim=3** とします。

(4) **dim1=<数値>　dim2=<数値>　dim3=<数値>**

各軸方向の格子数を定義します。
ndim=1 の場合は dim1 のみ、ndim=2 の場合は dim1,d dim2 を指定します。

(5) **nspace=<数値>**

データが存在する物理空間の次元数を定義します。
例えば、格子点の座標値をX,Yの２次元座標で定義するときは nspace=2、X,Y,Z の３次元座標で定義するときは nspace=3 となります。 field=irregular の場合以外は ndim の値と一致します。

(6) **veclen=<数値>**

格子点に存在するデータ成分の種類の数を定義します。

(7) **data=<文字列>**

データ成分の型を定義します。
<文字列>の部分には、以下のタイプからいづれかを設定します。
データ成分が複数ある場合、すべての成分は同じデータ型にしなければなりません。

short	byte	integer	float	double

(8) **field=<文字列>**

座標値に関する情報を定義します。座標値の持ち方によって<文字列>の部分には、以下のタイプからいづれかを設定します。

uniform	rectilinear	irregular

□uniform

直交等間隔の格子。

uniform 型の場合、座標情報は必要なく、ファイルの座標部分は不要になります。

□rectilinear

直交不等間隔の格子。

rectilinear 型の場合、代表となる各軸方向の座標値を定義します。

□irregular

すべての格子点の座標値を定義します。

uniform や rectilinear 型のデータの場合でも、すべての格子点の座標値を記述すれば irregular 型として定義することもできます。

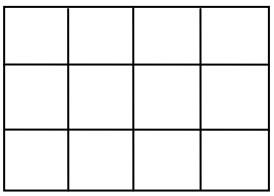

図５-11-a　uniform

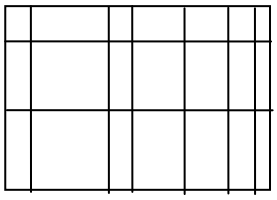

図５-11-b　rectilinear

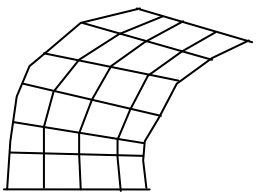

図５-11-c　irregular

⑼ **label=<文字列>　<文字列> ... （オプション）**

データ成分に対してラベルを付けます。

データ成分の種類が複数ある場合は<文字列>と<文字列>の間にスペースを入れて区切ります。

５－５－５ フォーマット(データを直接記述するタイプ)

ヘッダー部分に続いてデータ部分、及び座標部分を記述する場合は、下図のようになります。

注) ヘッダー部分とデータ部分を区切るためにセパレータ"^L^L"([Ctrl]+L [Ctrl]+L)が必要です。

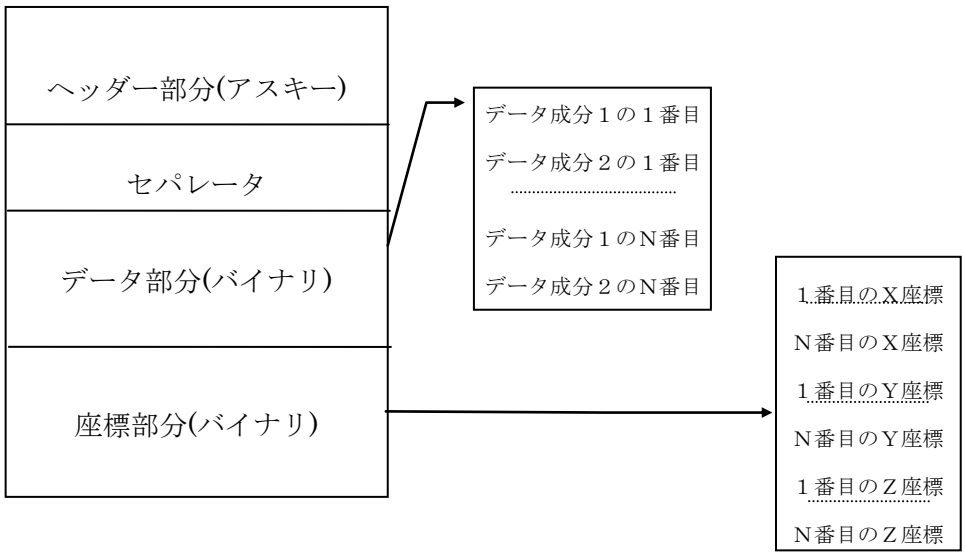

図５-12　データを直接記述するタイプ

５－５－６ フォーマット(データを参照するタイプ)

この場合、別ファイルからデータ値や座標値を取り込むことになります。その別ファイルはアスキーまたはバイナリのどちらでも構いません。

```
#AVS field file
ndim    = 3
dim1    = 3
dim2    = 3
dim3    = 3
nspace  = 3
veclen  = 2
data    = float
field   = irregular

variable 1 file=/tmp/data1 filetype=ascii skip=1 offset=0 stride=2
variable 2 file=/tmp/data1 filetype=ascii skip=1 offset=1 stride=2   (1)

coord 1 file=/tmp/data2 filetype=ascii skip=1 offset=0 stride=3
coord 2 file=/tmp/data2 filetype=ascii skip=1 offset=1 stride=3      (2)
coord 3 file=/tmp/data2 filetype=ascii skip=1 offset=2 stride=3
```

図５-13　データを参照するタイプの例

(1) **variable n file=**<文字>　**filetype=**<文字>　**skip=**<数値>　**offset=**<数値>　**stride=**<数値>

格子点上のデータの読み込み方を指定します。１行で記述されていなければなりません。各キーワードの意味は次の通りです。

☐ **variable**

このキーワードで始まる行は、データに関する情報を示します。

☐ **n**

データ成分を区別する番号で、1 から始まる整数値を指定します。

□ file=<文字>

読み込むファイル名を指定します。絶対パス名、相対パス名のどちらでも構いません。パソコンでの絶対パス名の指定は file=C:¥tmp¥data1 のように指定します。

□ filetype=<文字>

読み込むファイルのタイプをアスキーかバイナリかで指定します。

アスキーの場合：filetype=ascii バイナリの場合：filetype=binary（C プログラムで出力） filetype=unformated（Fortran の unformated で出力）

□ skip=<数値>

最初にどれだけファイルを読み飛ばすかを指定します。
指定しない場合のデフォルト値は skip=0

アスキーの場合： 最初に読み飛ばす行数 バイナリの場合： 最初に読み飛ばすバイト数

□ offset=<数値>

読み始める行の、さらに何カラム目から読み始めるのかを指定します。
指定しない場合のデフォルト値は offset=0

アスキーの場合： 読み飛ばす項目数 バイナリの場合： なし

□ stride=<数値>

データとデータの間をどれだけ読み飛ばすのかを指定します。

アスキーの場合： 読み飛ばす項目数 データとデータの間にスペースまたはタブが１つ以上ある場合、それを１つの区切りと見なします。 バイナリの場合： 読み飛ばす要素数 例えば３つの float 型のデータが並んでいる場合、stride=3 とすると 12 バイトごとにデータを読み込みます。

(2) **coord n file =**< 文字>**filetype =**< 文字>**skip =**< 数値>**offset =**< 数値>**stride =**< 数値>

格子点の座標データの読み込み方を指定します。１行で記述されていなければなりません。各キーワードの意味は基本的に variable と同じです

☐ **coord**

このキーワードで始まる行は、座標値に関する情報を示します。

☐ **n**

座標軸を区別する番号です。

n=1：X 座標
n=2：Y 座標
n=3：Z 座標

☐ **file=<文字>**

データ部分のキーワードと同じ意味です。

☐ **filetype=<文字>**

データ部分のキーワードと同じ意味です。

☐ **skip=<数値>**

データ部分のキーワードと同じ意味です。

☐ **offset=<数値>**

データ部分のキーワードと同じ意味です。

☐ **stride=<数値>**

データ部分のキーワードと同じ意味です。