

# Array and Table Definitions

The array definition function is used to define the occurrences and layout of an array (X-arrays are not supported).

Array definition is initiated by the field command .A or by issuing the line command . .E and then marking the required field with the function code A.

The table definition function is used to define the occurrences and layout of more than one array at the same time. The arrays must begin in the same map line.

Table definition is invoked by the line command . .A.

This section contains information on the following:

- Array Definition
- Table Definition

## Array Definition

The upper portion of the following screen is displayed for the purpose of array definition:

```

Name #001                               Top Dim 1_____ 1_____ 1_____
-----
Dimensions                               Occurrences   Starting from   Spacing
0 . Index vertical                       1__           _____    0   Lines
0 . Index horizontal                      1__           _____    1   Columns
0 . Index (h/v) V                         1__           _____    0   Cls/Ls

001  --010---+-----+-----030---+-----+-----050---+-----+-----070---+-----
      .AXXXXXX

      Please enter starting name .AXXXXXXXXXXXXXXXXXXXXX

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Mset  Exit          --   -   +           <   >       Let
    
```

You can specify the following:

Field	Explanation
<b>Top Dim</b>	<p>Indicates the top dimension of the array; that is, the highest occurrence (from left to right) in the first, second and third dimension.</p> <p>If a field defined in a program is used to define the map array, the upper bounds of that field (user-defined variable or database field), as defined in the program, are used; these cannot be overwritten on the array definition screen.</p> <p>If you select a map array from a data definition in another Natural object, the dimensions of the map array must not exceed the dimensions shown in this field.</p> <p>If you do not select a map array from a data definition, the dimensions of the map array must not exceed the dimensions as defined in the Natural program.</p>
<b>Dimensions</b>	An array can have up to three dimensions. The order in which the dimensions of the array are mapped to the map layout is determined by the values entered to the left of the <b>Index</b> operands.
<b>Occurrences</b>	The number of occurrences to be defined for a dimension.
<b>Starting from</b>	<p>The starting index value for a dimension. A numeric value can be used, or a variable name can be used to indicate that the actual value is supplied in the Natural program which invokes the map definition.</p> <p>If the variable is not defined otherwise as a field in the map, it is assumed to be of Natural data format/length N7. If so, it can be edited using PF9 in the <b>Field and Parameter Definition</b> screen.</p> <p><b>Note:</b> Removing a <b>Starting from</b> value from an array implies that the variable is removed from the map, too, unless it is a map field or it is associated with any other map field as a <b>Starting from</b> value or help parameter. To edit <b>Starting from</b> values, press PF9 in the <b>Field and Variable Definitions - Summary</b> screen.</p>
<b>Spacing</b>	The number of blank lines (for vertical dimensions) or blank columns (for horizontal dimensions) to be inserted between each dimension occurrence.

## Examples of Array Definitions

### Example 1:

A one-dimensional array consisting of 10 vertical occurrences with 2 blank lines to be inserted between neighboring occurrences.

Name #001	Top Dim 10_____	1_____	1_____
-----			
Dimensions	Occurrences	Starting from	Spacing
1 . Index vertical	10_	_____	2 Lines
0 . Index horizontal	1_	_____	1 Columns
0 . Index (h/v) V	1_	_____	0 Cls/Ls

**Example 2:**

Same as *Example 1* except that the array is to be horizontal.

Name #001	Top Dim 10	1	1
-----			
Dimensions	Occurrences	Starting from	Spacing
0 . Index vertical	1__	_____	0 Lines
1 . Index horizontal	10_	_____	1 Columns
0 . Index (h/v) V	1__	_____	0 Cls/Ls

**Example 3:**

A two-dimensional array. The first dimension consists of 10 vertical occurrences with 1 blank line between neighboring occurrences. The second dimension consists of 5 horizontal occurrences with 2 blank columns between neighboring occurrences.

Name #001	Top Dim 10	5	1
-----			
Dimensions	Occurrences	Starting from	Spacing
1 . Index vertical	10_	_____	1 Lines
2 . Index horizontal	5__	_____	2 Columns
0 . Index (h/v) V	1__	_____	0 Cls/Ls

**Example 4:**

Same as *Example 3* except that the order of the dimensions is reversed.

Name #001	Top Dim 5	10	1
-----			
Dimensions	Occurrences	Starting from	Spacing
2 . Index vertical	10_	_____	1 Lines
1 . Index horizontal	5__	_____	2 Columns
0 . Index (h/v) V	1__	_____	0 Cls/Ls

**Example 5:**

A three-dimensional array. The first dimension consists of 3 vertical occurrences with 1 blank line between neighboring occurrences. The second dimension consists of 5 horizontal occurrences with 2 blank columns between neighboring occurrences. The third dimension consists of 2 occurrences, expanded vertically within each occurrence of the first dimension.

Name #001	Top Dim 3	5	2
-----			
Dimensions	Occurrences	Starting from	Spacing
1 . Index vertical	3__	_____	1 Lines
2 . Index horizontal	5__	_____	2 Columns
3 . Index (h/v) V	2__	_____	0 Cls/Ls

**Example 6:**

An example of using **Starting from**. The first dimension consists of 10 vertical occurrences starting from index I. I is defined in the map editor with Natural data format/length N7 by default. The second dimension consists of 5 horizontal occurrences starting from the index 3.

Name #001	Top Dim 10_____ 5_____ 1_____		
-----			
Dimensions	Occurrences	Starting from	Spacing
1 . Index vertical	10_	I_____	1 Lines
2 . Index horizontal	5_	3_____	2 Columns
0 . Index (h/v) V	1_	_____	0 Cls/Ls

**Example 7:**

An example of making a two-dimensional display from a one-dimensional array. The array consists of 40 elements. It is displayed in two columns with 20 lines each. This is achieved by specifying 0 as the horizontal index.

Name #001	Top Dim 40_____ 1_____ 1_____		
-----			
Dimensions	Occurrences	Starting from	Spacing
1 . Index vertical	20_	_____	0 Lines
0 . Index horizontal	2_	_____	10 Columns
0 . Index (h/v) V	1_	_____	0 Cls/Ls

**Table Definition**

A table of one or more arrays which all begin in the same map line is defined with the . .A line command. When you enter the . .A line command, the following screen is invoked:

14:41:47	***** NATURAL MAP EDITOR *****				2007-10-22
- Array Table Definition -					
Main Index:	Vert. Occur.	1	Starting from _____	Spacing 0	Lines
Second Index:	Direction(H/V)	V	_____	0	Cls/Ls
Third Index:	Direction(H/V)	V	_____	0	Cls/Ls
-----					
Name of Variable (truncated)	Col Pos	Ind1	Dimension Size Ind2	Ind3	Order 2. 3. M S T Occ Occ
-----					
#001	2	1	1	1	
#002	25	1	1	1	
-----					
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10---PF11---PF12---					
Help Mset Exit -- - + Let					

**Note:**

When applying the `. . A` command to arrays which were not defined by an `. . A` command but by an `. A` command, may result in a modification or even a destruction of these arrays.

The example screen above contains the following fields:

<b>Field</b>	<b>Explanation</b>
<b>Main Index</b>	The number of vertical occurrences, the starting position and the number of lines to be skipped between dimension occurrences.
<b>Second Index</b>	The direction (horizontal or vertical), the starting position and the number of lines/columns to be skipped between dimension occurrences.  The second dimension only applies if one of the arrays has more than one dimension. In this case the second dimension can be displayed either horizontally (in which case there must be enough space in the line for all selected occurrences) or vertically (in which case there must be enough lines on the map to display main dimension times second dimension occurrences, including line spacing).
<b>Third Index</b>	The direction (horizontal or vertical), the starting position and the number of lines/columns to be skipped between dimension occurrences.  The third dimension only applies if one of the arrays has more than two dimensions. In this case the third dimension can be displayed either horizontally (in which case there must be enough space in the line for all selected occurrences) or vertically (in which case there must be enough lines on the map to display main dimension times second dimension times third dimension occurrences, including line spacing).
<b>Name of Variable</b>	All names of field arrays contained in the table are displayed.
<b>Col Pos</b>	The column position in which the field is located. This is displayed for informational purposes only.
<b>Dimension Size</b>	The upper bounds <b>Ind1</b> , <b>Ind2</b> and <b>Ind3</b> of an array.  The dimensions of the array defined in the map must not exceed the dimensions of the corresponding array defined in the Natural object that invokes the map.
<b>Order</b>	The order in which the dimensions are to be defined: <b>M</b> , <b>S</b> and <b>T</b> correspond to Main, Second and Third.
<b>2. Occ.</b>	The number of occurrences to be defined for the second index.
<b>3. Occ.</b>	The number of occurrences to be defined for the third index.

**Example of a Table Definition**

This is an example of defining map fields that correspond to the following program definition:

```
DEFINE DATA
  1 ARRAY1 (A3/1:10)
  1 ARRAY2 (A5/1:10,1:2)
  1 ARRAY3 (A7/1:10,1:2,1:3)
END-DEFINE
```

**Table Definition:**

```

14:41:47          ***** NATURAL MAP EDITOR *****          2006-07-24
                    - Array Table Definition -

Main  Index:  Vert. Occur.  1   Starting from _____ Spacing 0   Lines
Second Index:  Direction(H/V) V   _____                0   Cls/Ls
Third  Index:  Direction(H/V) V   _____                0   Cls/Ls
-----
Name of Variable      Col      Dimension Size      Order 2.  3.
(truncated)          Pos Ind1      Ind2      Ind3      M S T Occ Occ
-----
ARRAY1                3  10          1          1          1
ARRAY2               32  10          2          1          1 2  2
ARRAY3               58  10          2          3          1 2 3  2 3

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help Mset Exit          --  -  +                          Let
    
```

ARRAY1 is a one-dimensional array with ten occurrences. The first two occurrences are expanded in the table.

ARRAY2 is a two-dimensional array. The first index consists of ten occurrences and the second index consists of two occurrences. The first two occurrences of the first index and both occurrences of the second index are expanded in the table.

ARRAY3 is a three-dimensional array. The first index consists of ten occurrences, the second index consists of two occurrences and the third index consists of three occurrences. The first two occurrences of the first index, both occurrences of the second index and all three occurrences of the third index are expanded in the table.

**Table Layout:**

