LANDSCAPE Script Reference Manual

Hirohisa Mori
joshua@globalbase.org

Transrated by
Reiko Inoue Bendtsen c/o Suzaku Translations
(http://www.suzaku-translations.com/)

Edition 2009-12-19

Contents

Chapter 1

Overview

1.1 Abstract and Goal

The functionality of LANDSCAPE is implemented by means of various scripts running on the server. This manual explains the basic operations, including conversion of data to be dislosed, in particular image data. In addition, the reader may refer to chapters of old manuals (http://)for further information on contents creation. We intend to incorporate those old manuals in this manual one by one.

1.2 Human Requirements

Before you read this manual, you should install the LANDSCAPE server on an appropriate computer. Please refer to "LANDSCAPE Startup Manual" [1]for how to install the server.

In order to read this manual, you are technically speaking only required to know how to manipulate XLscript files; however, we do recommend that you understand enough of the language syntax for you to be able to understand the contents as well. Please refer to xlscript [2]"Protocol Reference Model of GLOBALBASE Architecture: The Autonomous Decentralized GIS" [**?**]as needed.

Moreover, you are required to have basic knowledge on how to search through GLOBALBASE coordinate systems and the principles of overlapping them. These technical details are explained in the GLOBALBASE Technical Materials [UNDEF REF (GLOBALBASE-tech)]. We recommend that you obtain first-hand familiarity with the COSMOS GLOBALBASE VIEWER [3]by trying it out yourself as well.

1.3 System Requirements

A POSIX-compliant machine is required. You will be able to use the functions of xlsv by installing LANDSCAPE GLOBALBASE VIEWER [1]on the machine.

1.4 Precautions on Server Operations

After the installation, the LANDSCAPE GLOBALBASE server is started and shut down with the launch-xl (Section 2.3.1) command (except for MacOSX) and XLServer (Section 2.3.1) . Some of the commands explained in this manual are only valid when the LANDSCAPE GLOBALBASE server is running. The availability is explained for each item; always make sure that the server is launched before using these commands.

Chapter 2

LANDSCAPE script/Reference

2.1 Abstract

Some of the commands explained in this manual are only valid when the LANDSCAPE GLOBALBASE server is running. The requirements are listed for each item; always make sure that the server is launched before using these commands.

It is possible to display errors if appropriate agents are specified as environmental data. For example, if the following is displayed in the script:

xl matrix.xl / import-pnm [pnm-filename] [mtx-filename]

it is possible to display errors by entering the following commands:

xl matrix.xl - - / import-pnm [pnm-filename] [mtx-filename]

2.2 Inheritance Information

Each script command uses functions that take advantage of various agents. Please refer to the section on agents as well.

2.3 Scripts

2.3.1 launch-xl (Launching the server)

■ **Prototype**

/etc/init.d/launch-xl start
/etc/init.d/launch-xl stop
/Library/StartupItems/XLServer/XLServer start (MacOSX)
/Library/StartupItems/XLServer/XLServer stop (MacOSX)

■ **Arguments**

start LANDSCAPE Launching the server
stop LANDSCAPE Stopping the server

■ **Environemnts Belongs**

Shell command

■ **Explaination**

These commands start or stop the LANDSCAPE server. They basically conform to the start script format of each OS. Immediately after installation of the server, they are automatically set to launch the LANDSCAPE server upon reboot. Refer to the manual of each OS for the explanation on how to turn this automatic launch on or off.

■ **Return Value**

None

■ **Error**

■ **References**

■ **Bugs**

2.3.2 makefile.xl (Regitration of contents)

■ **Prototype**

xl makefile.xl

■ **Arguments**

■ **Environemnts Belongs**

Agent xl [2]

LANDSCAPE must be launched before this command.

■ **Explaination**

This command registers or deletes contents at the moment when the contents are stored under the xldocs directory of the LANDSCAPE server or the contents are deleted. Contents are not delivered normally if they are simply placed in the directory without issuing this command.

This command is used in the following procedure:

1. Store contents under the xldocs directory.
2. Use the cd command or similar to move to the directory where the contents are stored.
3. Issue the command.

```
xl makefile.xl
```

If you issue this command in the the xldocs directory or the /usr/local/xl-gbs directory, all contents in the xldocs directory are registered.

■ **Return Value**

None

■ **Error**

■ **References**

■ **Bugs**

This command currently does not work properly for contents stored in the xldocs directory itself. This problem can be avoided by creating another directory inside the xldocs directory and storing the contents in the new directory.

2.3.3 p4m.xl (Creating mappings of 4 points or more)

■ **Prototype**

xl p4m.xl / [p4m-file]

■ **Arguments**

[p4m-file] Mapping file containing only point-map and meta data

Option

- reverse= **none/src/dest** The default is **none** .

  Support from ver.B.b16.05 and later.

■ **Environemnts Belongs**

Agent xl [2], Agent netmapper [UNDEF REF (netmapper)]

■ **Explaination**

In mappings involving 4or more corresponding points, it is not possible to match the coordinates via linear transformation. For this reason, this command uses a complementary method, where the entire map region is divided into multiple triangular areas (so-called TINs) and a different linear transformation is applied inside each TIN. Moreover, it is necesary to specify an approximate linear transformation for infinite distances so that a transformation can be applied to points placed at infinity in the coordinate systems. Although it is possible to specify these parameters and create a mapping file manually, it is usually easier to generate the mapping file automatically by specifying corresponding points and running the p4m.xl script.

In LANDSCAPE ver.B.b16.05 and later, a new reverse option has been added. If the axis directions of the mapping source and mapping destination are different, the reverse option is required. The reverse option is used to specify a mathematical coordinate system (x-axis direction from left to right, y-axis direction up) relative to a computer coordinate system (x-axis direction from left to right, y-axis direction down). For example, if the mapping source is a mathematical coordinate system, specify reverse=src. If the mapping destination is a mathematical coordinate system, specify reverse=dest. If both the mapping source and destination are mathematical coordinate systems, specify reverse=none.

First, create a file with the extension p4m. Basically, any extension except for map can be used. The structure of a p4m file is the same as that of a map file and contains point-map and meta data only. The first 3points can be considered as a map file, and the 4thpoint and onward can be considered as appended point-map data. Thus, if you already have a mapping file where a mapping of up to 3points has been specified, and you consider adding a 4thpoint to it, you can simply add the 4thpoint and rename the file extension to .p4m to create a p4m file. The example below shows the contents of a p4m file for a single system (Kyushu).

```
<?xml version="1.0" encoding="EUC-JP"?>
<map>
 <meta>
  <file type="xl"/>
  <src>
   xlp://isjhp2.nichibun.ac.jp:8080/kokudo/coord/01.crd
  </src>
  <dest>
   xlp://isjhp2.nichibun.ac.jp:8080/kokudo20000/coord/00.crd
  </dest>
  <dp>
   181
  </dp>
 </meta>
 <point-map>
  b1
  <list>
  6577.435059m
  30896.992188m
  </list>
```

<list>
146482.000000m
1294111.625000m
</list>
</point-map>
<point-map>
b2
<list>
-16864.001953m
34593.703125m
</list>
<list>
149568.734375m
1315266.750000m
</list>
</point-map>
<point-map>
b3
<list>
-20587.638672m
81065.656250m
</list>
<list>
189254.359375m
1318867.375000m
</list>
</point-map>
<point-map>
b4
<list>
3998.532959m
12131.631836m
</list>
<list>
130398.726563m
1296403.500000m
</list>
</point-map>
<point-map>
b5
<list>
-38258.191406m
24350.640625m
</list>
<list>
140765.75000m
1334532.000000m
</list>
</point-map>
<point-map>
b6
<list>
-15451.980469m
56954.250000m
</list>
<list>
168696.531250m

```
  1314098.375000m
  </list>
 </point-map>
</map>
```

Now save this contents in a file called and run the following script.

```
% xl p4m.xl - - / 01.p4m
```

Running the script generates the file, with the same name as the original except that the extension is changed to .map. However, this new file is a distributable mapping file in which TIN and infinite distance linear transformations are specified. The file contains the following.

```
<map>
 <meta>
  <file type="xl"/>
  <src>
   xlp://isjhp2.nichibun.ac.jp:8080/kokudo/coord/01.crd
  </src>
  <dest>
   xlp://isjhp2.nichibun.ac.jp:8080/kokudo20000/coord/00.crd
  </dest>
  <dp>
   181
  </dp>
 </meta>
 <horizontal-affen>
  <List>
   <List>
    0.002363
    0.854563
   </List>
   <List>
    -0.901667
    0.004435
   </List>
  </List>
  <List>
   120044.569714
   1299925.616167
  </List>
 </horizontal-affen>
 <point-map>
  ^"1"
  <list>
   -127929.444336
   -125736.416992
  </list>
  <list>
   12292.601562
   1414717.750000
  </list>
 </point-map>
 <point-map>
```

```
 ^"2"
 <list>
  -38258.191406m
  24350.640625m
 </list>
 <list>
  140765.750000m
  1334532.000000m
 </list>
</point-map>
<point-map>
 ^"3"
 <list>
  -127929.444336
  218933.705078
 </list>
 <list>
  306835.000000
  1416246.250000
 </list>
</point-map>
<triangle-map>
 ^"1"
 ^"2"
 ^"3"
</triangle-map>
<triangle-map>
 ^"2"
 ^"3"
 ^"1"
</triangle-map>
<triangle-map>
 ^"3"
 ^"2"
 ^"1"
</triangle-map>
<point-map>
 ^"4"
 <list>
  6577.435059m
  30896.992188m
 </list>
 <list>
  146482.000000m
  1294111.625000m
 </list>
</point-map>
<point-map>
 ^"5"
 <list>
  96248.687988
  218933.705078
 </list>
 <list>
  307364.656250
  1214112.375000
 </list>
```

```
</point-map>
<point-map>
 ^"6"
 <list>
  96248.687988
  -125736.416992
 </list>
 <list>
  12822.257812
  1212583.750000
 </list>
</point-map>
<triangle-map>
 ^"4"
 ^"5"
 ^"6"
</triangle-map>
<triangle-map>
 ^"6"
 ^"4"
 ^"5"
</triangle-map>
<triangle-map>
 ^"5"
 ^"4"
 ^"6"
</triangle-map>
<point-map>
 ^"7"
 <list>
  3998.532959m
  12131.631836m
 </list>
 <list>
  130398.726563m
  1296403.500000m
 </list>
</point-map>
<triangle-map>
 ^"1"
 ^"7"
 ^"6"
</triangle-map>
<triangle-map>
 ^"7"
 ^"6"
 ^"1"
</triangle-map>
<triangle-map>
 ^"6"
 ^"7"
 ^"1"
</triangle-map>
<point-map>
 ^"8"
 <list>
  -20587.638672m
```

```
  81065.656250m
 </list>
 <list>
  189254.359375m
  1318867.375000m
 </list>
</point-map>
<triangle-map>
 ^"5"
 ^"8"
 ^"3"
</triangle-map>
<triangle-map>
 ^"3"
 ^"8"
 ^"5"
</triangle-map>
<triangle-map>
 ^"8"
 ^"3"
 ^"5"
</triangle-map>
<triangle-map>
 ^"2"
 ^"8"
 ^"3"
</triangle-map>
<triangle-map>
 ^"3"
 ^"2"
 ^"8"
</triangle-map>
<triangle-map>
 ^"5"
 ^"8"
 ^"4"
</triangle-map>
<triangle-map>
 ^"8"
 ^"4"
 ^"5"
</triangle-map>
<triangle-map>
 ^"1"
 ^"2"
 ^"7"
</triangle-map>
<triangle-map>
 ^"2"
 ^"7"
 ^"1"
</triangle-map>
<triangle-map>
 ^"7"
 ^"4"
 ^"6"
</triangle-map>
```

```
<triangle-map>
 ^"6"
 ^"7"
 ^"4"
</triangle-map>
<point-map>
 ^"9"
 <list>
  -15451.980469m
  56954.250000m
 </list>
 <list>
  168696.531250m
  1314098.375000m
 </list>
</point-map>
<triangle-map>
 ^"9"
 ^"2"
 ^"8"
</triangle-map>
<triangle-map>
 ^"9"
 ^"8"
 ^"4"
</triangle-map>
<point-map>
 ^"10"
 <list>
  -16864.001953m
  34593.703125m
 </list>
 <list>
  149568.734375m
  1315266.750000m
 </list>
</point-map>
<triangle-map>
 ^"2"
 ^"7"
 ^"10"
</triangle-map>
<triangle-map>
 ^"9"
 ^"2"
 ^"10"
</triangle-map>
<triangle-map>
 ^"9"
 ^"10"
 ^"4"
</triangle-map>
<triangle-map>
 ^"7"
 ^"10"
 ^"4"
</triangle-map>
```

```
</map>
```

Once this file is recognized by the server, the mapping is completed.

```
% xl makefile.xl - -
```

■ **Return Value**
None
■ **Error**
■ **References**
■ **Bugs**

2.3.4 matrix.xl (Matrix Type Image Format Processing)

■ **Prototype**

xl matrix.xl / mtx-status [mtx-filename]

xl matrix.xl - - / pnm-status [pnm-filename]

xl matrix.xl / create-mtx [mtx-filename] [width] [height]

xl matrix.xl / import-pnm [pnm-filename] [mtx-filename]

xl matrix.xl / export-pnm [mtx-filename] [pnm-filename] splitmode=all

xl matrix.xl / export-pnm [mtx-filename] [pnm-filename] [x-offset] [y-offset] [width] [height] splitmode=part

xl matrix.xl / export-pnm [mtx-filename] [pnm-filename] [x-divide-nos] [y-divide-nos] splitmode=split

xl matrix.xl / import-pnm-over [pnm-filename] [mtx-filename] [x-offset] [y-offset]

xl matrix.xl / import-dted1 [target-dir] [dted1-filepath] [bib]

xl matrix.xl / block [mtx-filename] [jpg-filename] [level] [x] [y]

xl matrix.xl / copy-mtx [mtx-src-filename] [mtx-dest-filename] [channelNo.] ....

xl matrix.xl / scan-network [mtx-filename] **standard** [start-warp-point-file] [end-point-x] [end-point-y]

xl matrix.xl / scan-network-offset [mtx-filename] **standard** [start-warp-point-file] [end-point-x] [end-point-y] [dim-code-level] [dim-code-x] [dim-code-y]

xl matrix.xl - - / leveling-mtx [mtx-filaname]

■ **Arguments**

[mtx-filename] Matrix file name

[pnm-filename] PNM file name

[jpg-filename] JPEG file name

[dted1-filepath] File path of a DTED1 type file

[bib] A bib file with bib elements extracted from a crd file

[width] Image width, number of pixels

[height] Image height, number of pixels

[x-offset] Image saving start position. Number of pixels offset in the x direction of a matrix file

[y-offset] Image saving start position. Number of pixels offset in the y direction of a matrix file

[x-divide-nos] Number of image divisions in the x direction, for export-pnm

[y-divide-nos] Number of image divisions in the y direction, for export-pnm

[level] Hierarchy level of an image

[x] Coordinate position of an image, x coordinate of the matrix file

[y] Coordinate position of an image, y coordinate of the matrix file

[mtx-src-filename] Copy source matrix file name

[mtx-dest-filename] Copy destination matrix file name

[channelNo.] Channel number or the section of "Channel Flag (Matrix)" in the manual "COSMOS Reference Manual"[4]

[start-warp-point-file] File in which warp point data is saved. Warp point data refers to text data obtained by clicking "Edit" and then "Copy Place Info." the section of "Edit Menu" in the manual "COSMOS Reference Manual"[4]

[end-point-x] X-coordinate of the end of scan data. Base coordinate values of warp point data specified by [start-warp-point-file].

[end-point-y] Y-coordinate of the end of scan data. Base coordinate values of warp point data specified by [start-warp-point-file].

Option

- **Options in create-mtx**

  default_color= [default_color] The default is -1.

- **Options in import-pnm-over**

  transparent= [transparent_color] The default is 16777215.

  leveling= on/off The default is on.

- **Options in scan-network**

  meta= [bib file name]

  leveling= [on/off] The default is off.

- **Options in export-pnm**

  splitmode= [all/part/split] The default is all.

  level= integer The default is

■ **Environemnts Belongs**

Agent xl [2], gbmx [5]gbview [5]

■ **Explaination**

This script is used to perform operations related to matrix data. The script can currently only be applied to 8 bit RGB type images. The first argument indicates the operation, which is defined as follows.

- **mtx-status**

  This option returns the status information of a given matrix file. Please refer to the Matrix Format Specification for the detailed explanation. Since the status is the return value of the XLscript, it is necessary to enable the standard output.

  % xl matrix.xl - - / mtx-status

  This is why "- -" is inserted.

- **pnm-status**

  This option returns the status information of a given PNM file. Please refer to the netpbm Manual for the detailed explanation.

- **create-mtx**

  This option generates an empty matrix file of a given size (pixel size). It is equivalent to a totally transparent image when referenced.

  Specify a color conceptually assumed to be set before writing for the default_color option. This must be a RGBA 32-bitvalue. A=00 is completely opaque and A=FF is completely transparent.

- **import-pnm**

  This option converts a given PNM file into a matrix file with a given name. The image will have the same size in number of pixels.

- **export-pnm**

  In version ver.B.b17.01 or earlier, this argument converts a given matrix file to a PPM file with a given name. The image will have the same size in number of pixels. In version ver.B.b17.02 or later, a function for outputting images of various sizes has been added, which is explained below.

  From ver.B.b16.14, it is possible to specify the level option and convert data in each hierarchy in a matrix. The default is level=0. The number of pixels of each layer of the matrix varies depending on the settings of the matrix.

  This option converts a given matrix file to a PPM file with a given name. There are three formats. If only input file and output file names are specified, an image with the same number of pixels as the input matrix file is output to the output file. You can also specify the option **splitmode=all** as well. If you specify the **level** option, you can output images at any level.

  If you specify four values for [x-offset], [y-offset], [width], and [height] in addition to the input file and output file and specify **splitmode=part** , an image of size [width] times [height] is cut out starting at the given offset. The **level** option can be used. Moreover, the splitmode option can be omitted.

  If you specify [x-divide-nos], [y-divide-nos], and **splitmode=split** in addition to the input file and output file names, the image is divided into [x-divide-nos] parts in the x direction and [y-divide-nos] parts in the y direction and output into several PPM files. Since the image file is

automatically assigned the name [pnm-filename]- [index number in the x direction]- [index number in the y direction].ppm, no extension should be specified for [pnm-filename]. The **split-mode=on** option cannot be omitted.

- **import-pnm-over**

  This option embeds a given PNM file ([pnm-filename]) into an existing matrix file ([mtx-filename]). The embedding position is given by [x-offset], [y-offset].

  Specify colors to be eliminated for the transparent option as 24-bitRGB values. In other words, all parts of the image with the color specified by the transparent option will be assigned the originally written color. If this option is not specified, all colors are overwritten.

  The leveling=off option is valid in ver.B.b16.14. If this option is specified, leveling is not performed. It is a convenient option when importing files in a batch and determining their layers at the end.

- **import-dted1**

  This option reads a DTED1-type file and generates a matrix file, which is a 2-dimensionalarray of 16-bitsigned (complement of integers, along with object files such as .crd that reference it. For example, execute the following command.

```
xl matrix.xl - - / import-dted1 test 'area01/dted/*/*.dt1' bib.xl
```

  [dted1-filepath] shall be specified as a file path to a directory where a series of DTED1-type files are saved and a file path down to file name. Specifically, in case of the DTED1 format, the file name and the directory name including the file express the latitude and longitude directly and this information is thus essential; an error occurs if a file path is not specified. Note also that multiple files are specified in the example above by using wild cards for the parts corresponding to latitude and longitudae.

  bib.xl corresponding to [bib] is a bibliographic information file with the same format as worldfile-mtx.xl etc. This file contains bibliographic information of data to be disclosed, for example using the following format.

```
<?xl version="0.1" encoding="EUC-JP"?>
<bib xmlns:gb="xlp://isjhp1.nichibun.ac.jp:8080/gb_metadata">
   <gb:title type="text" data="Japan Ortho"/>
   <gb:creator type="text" data="Osaka City University"/>
   <gb:content.period type="W3C-DTF" data="2003-01-01 / 2003-01-01"/>
   <gb:issue.period type="W3C-DTF" data="2006-06-01"/>
   <gb:property type="gb-prop" data="photo"/>
   <gb:homepage type="URL" data="http://www.osaka-cu.ac.jp/"/>
</bib>
```

- **block**

  This option extracts a JPEG image at a given coordinate position ([level] [x] [y]) from a matrix file [mtx-filename]. The extracted image is saved in the file specified by [jpg-filename].

- **copy-mtx**

  This option generates a matrix file named [mtx-dest-filename] that contains the same header information as the matrix file specified by [mtx-src-filename] and copies only the channels given by [channelNo.] for all nodes within [mtx-src-filename]. It is possible to specify multiple channels for [channelNo.].

  If you want to generate a matrix file including only transmission data from a generated matrix file, you can issue the following command.

```
% xl matrix.xl - - / copy-mtx srcfile.mtx destfile.mtx MF_SEND MF_SEND_VISU MF_SEND_FILE
```

Note that the following channels are currently defined in matrix_RGB8.xl.

- **8** Transmission blocks (compressed jpeg)
- **9** Non-compressed RGB images

- **scan-network**

  This function scans maps on the network and creates a single matrix image file with the name specified by [mtx-filename].

  The scanning is started from warp point data saved in the file specified by [start-warp-point-file]. Warp point data is stored using the following format and can be obtained by selecting "Edit" and then "Copy Location Info" in COSMOS the section of "Edit Menu" in the manual "COSMOS Reference Manual"[4] . In order to obtain the warp point data required, use COSMOS' browse function to display the image you want to scan, zoom to the desired location to start scanning at the desired scanning resolution, select the menu item and save the image to a new file.

```
<warp-point>
    <tracking-time>197748854sec</tracking-time>
    <title>China</title>
    <type>2</type>
    <query><OR>
        <query qtype="URL" title="test-C118" id="0" active="off">
            <URL>xlp://gbs2.itakura.toyo.ac.jp:8080/wakashima/C118/C118.crd</URL>
        </query>
        <query qtype="property" id="0" active="off">
            <qualifier cond="part">xlp://isjhp1.nichibun.ac.jp:8080/gb_metadata 0 property
        </query>
        <query qtype="property" title="Archiology" id="0" active="off">
            <AND>
                <qualifier cond="part">xlp://isjhp1.nichibun.ac.jp:8080/gb_metadata 0 prope
                <qualifier cond="boundary">xlp://isjhp1.nichibun.ac.jp:8080/gb_metadata 0 c
            </AND>
        </query>
        .......
        </OR>
    </query>
    <resolution>2.500000 </resolution>
    <rotate>0.000000 </rotate>
    <center>130.000000  0.000000 </center>
    <base>xlp://localhost:8080/test/China.crd</base>
    <layers>
        <entry>xlp://localhost:8080/test/China.crd 256</entry>
    </layers>
</warp-point>
```

The matrix.xl script reads warp point data and sets the base coordinate system to the coordinate system specified by the base element. In this coordinate system, scanning is started from the coordinates specified by the center element of warp point data at the resolution given by the resolution element, and stopped at the coordinates specified by [end-point-x], [end-point-y]. [end-point-x], [end-point-y] must be larger than the coordinate values of the center element. Moreover, rotation is ignored; scanning is performed in parallel with the x or y axis.

The query element specifies search criteria for the coordinate systems. All coordinates matching the criteria are overlapped. In other words, it is useful when scanning a large coordinate system containing data represented in multiple sub-coordinate systems.

A matrix file specified by [mtx-filename] is saved as matrix data, with the scanning start position at the position of (0,0) and scanning end position at the position of the maximum X and Y pixel values.

Note that if no option is specified, the matrix will be saved without layers. It is possible to convert data of a connected image into a pnm file via export-pnm at this point. This also means that it is possible to save time for leveling and create a connected image.

In order to set layers in a matrix, you can specify leveling=on as an option first or execute the command without any options and then use the leveling-mtx function of matrix.xl to set the layers. By setting layers, you can create matrix data that can be published on the network.

If you specify meta= [meta data file name], crd, map and lst files can be created. By executing the makefile.xl (Section 2.3.2) script after operation, all information that can be published is generated. If you specify the meta option, the leveling option is automatically set to on. An example of a meta data file (bib file) is shown below.

```
<?xl version="0.1" encoding="EUC-JP"?>
<bib xmlns:gb="xlp://isjhp1.nichibun.ac.jp:8080/gb_metadata">
   <gb:title type="text" data="Japan Ortho"/>
   <gb:creator type="text" data="Osaka City University"/>
   <gb:content.period type="W3C-DTF" data="2003-01-01 / 2003-01-01"/>
   <gb:issue.period type="W3C-DTF" data="2006-06-01"/>
   <gb:property type="gb-prop" data="photo"/>
   <gb:homepage type="URL" data="http://www.osaka-cu.ac.jp/"/>
</bib>
```

- **scan-network-offset**

  This function scans maps on the network and creates a single matrix image with the name specified by [mtx-filename]. This function is different from scan-network in that the scanned data is saved from the position of the section of "Dimension Code (Matrix)" in the manual "COSMOS Reference Manual"[4] specified at the end of the argument.

- **leveling-mtx**

  This function is supported from ver.B.b16.14. It sets the layer in a given matrix [mtx-filename]. For example, it can be used when a matrix is composed by import-pnm without layers, in order to set the layer to the results in a batch at a later stage.

The matrix-type format supports long filenames. However, in order to use the long filenames, it must be permitted in your OS and file system as well.

In the future, layer support will be built into the raster format r64/cr and the vector format pdbp, which were employed during the early development of GLOBALBASE, and these formats will gradually be integrated into the matrix-type format.

■ **Return Value**

■ **Error**

■ **References**

■ **Bugs**

2.3.5 worldfile-mtx.xl (World File/Matrix/Converter)

■ **Prototype**

exl worldfile-mtx.xl / [image-path] [worldfile-prefix] [destination-directory] [base-unit] [bib-file]

■ **Arguments**

[image-path] Specification of an image file
[worldfile-prefix] Extension of a world file
[destination-directory] Save destination directory
[base-unit] Unit of a coordinate file
[bib-file] File containining bibliographical information

option

- **bgcolor=** [background-color] The default is 16777215.
- **backup=on,off,recover** The default is off.
- **backuploop=** [backuploop-count] The default is
- **add=on,off,info** The default is off.
- **leveling=on,off** The default is on.

■ **Environemnts Belongs**

Agent exl [UNDEF REF (exl)], Agent gbmx [5]

■ **Explaination**

This command converts a world file into matrix format and generates a coordinate system file and mapping file so that the file can be viewed.

[image-path] specifies an image file; wild cards can be used. When using wild cards, it is necessary to enclose them with ” such that the shell does not recognize them. [worldfile-prefix] must start from ”.”. [base-unit] specifies the unit of the coordinate system, which is automatically assumed to be the unit of the coordinate system in which the world file data is specified. [bib-file] specifies the bib information to be embedded in a coordinate system file. The command might for example be invoked as follows.

% xl worldfile-mtx - - / ’*/image/*.bmp’ .bpw image m image/bib.xl

An example og the output in image/bib.xl is shown below.

```
<?xl version="0.1" encoding="EUC-JP"?>
<bib xmlns:gb="xlp://isjhp1.nichibun.ac.jp:8080/gb_metadata">
   <gb:title type="text" data="Japan Ortho"/>
   <gb:creator type="text" data="Osaka City University"/>
   <gb:content.period type="W3C-DTF" data="2003-01-01 / 2003-01-01"/>
   <gb:issue.period type="W3C-DTF" data="2006-06-01"/>
   <gb:property type="gb-prop" data="photo"/>
   <gb:homepage type="URL" data="http://www.osaka-cu.ac.jp/"/>
</bib>
```

The operations carried out by worldfile-mtx function can basically be divided into 2phases.

1. Phase one

   The command scans the specified image file and world file to determine the size of the overall coordinate system based on the allocated positions and the size and generates img.crd,img.map,img.lst files.

2. Second phase

   Next, the command actually reads the image file and saves it in the matrix file img.mtx.

Since the names of the files generated in the example above are determined to be img.crd, img.map, img.lst, and img.mtx, **the save location must be an empty directory.**

bgcolor option

If you specify the bgcolor option, RGBA = 256* bgcolor + 255is specified as the mtx's default_color and transparent=bgcolor is set when reading a pnm file. If this option is not specified, the setting is automatically assumed to be bgcolor=0xffffff.

backup option

The backup option specifies whether or not to back up img.mtx in the second phase. If you specify off, the file is not backed up. If you specify on, backup is saved in img.mtx.b at regular intervals, and the backup information is saved in backup-list.xml at the same time. The frequency of backing up is specified by the backuploop option. If you specify backup=recover, the first phase is skipped and the operation starts from the stage where backup-list.xml is read and backed up at the end. Note that before using this functionality, **img.mtx.b must be copied to img.mtx in advance.**

If you specify backup=recover, the file is backed up. A backup is required for the recover option.

backuploop option

The backuploop option specifies the frequency at which an image file is backed up during the writing to img.mtx. If you specify backuploop=1 or less, the file is backed up every time the file is written in img.mtx.

If the command is abnormally terminated or forcefully terminated, an img.mtx file may still be created in the destination folder, but it may be corrupted. In this case, it is necessary to copy the backup file img.mtx.b to img.mtx and start the processing from the top.

add option

The add option provides functionality for adding (or overwriting) new image data to an existing matrix created by worldfile-mtx earlier. If you specify add=info, the first phase is executedn and information required for overwriting is acquired and saved in backup-list.xml. If the backup option was specified in pervious operation and backup-list.xml already exists, there is no need to execute add=info. If you specify add=on the next time, the first phase is skipped, backup-list.xml is read, the operation is started from the second phase and the data is overwritten.

leveling option

This option is valid for ver.B.b16.14 and later. If you set the leveling option to off, no layers are set. This option is useful when outputting a composed image in pnm format or similar and avoids carrying out unnecessary tasks.

■ **Return Value**

- [destination-directory]/image file
- backup-list.xml backup information file

■ **Error**

■ **References**

■ **Bugs**

As of ver.B.b15, the command supports only world files mapped in parallel with coordinate axes. The operations cannot be guaranteed if files are mapped at an angle to the coordinate axes.

In ver.B.b16.12, it was found that the worldfile-mtx.xl script was not included in the installer. This problem was solved in ver.B.b16.13.

2.3.6 import-images.xl (Conversion of Individual Image Files)

■ **Prototype**

xl import-images.xl / [image-format] [path-to-images] [destination-directory] [bib-file]

■ **Arguments**

[image-format] Image format after conversion. Either **r64** or **mtx** can be specified.
[path-to-images] Path to the conversion source image file. A wild card can be used.
[destination-directory] Save destination directory
[bib-file] A file containing bibliographic information

Option

- **process=all,wp** The default is **all** .

■ **Environemnts Belongs**

Agent exl [UNDEF REF (exl)]

■ **Explaination**

This command converts .tif, .jpeg, and .bmp files to formats that can be viewed with GLOBALBASE. It also generates .crd, .map and .lst files at the same time and saves them as is in a status where they can be viewed.

It not only saves image files, but also generates warp point files that reference the saved files. If you refer to this file from the Warp Point menu of COSMOS, you can choose coordinate systems of generated images from a list in the warp point menu and add them.

A [bib-file] example is shown below.

```
<?xl version="0.1" encoding="EUC-JP"?>
<bib xmlns:gb="xlp://isjhp1.nichibun.ac.jp:8080/gb_metadata">
   <gb:title type="text" data="Japan Ortho"/>
   <gb:creator type="text" data="Osaka City University"/>
   <gb:content.period type="W3C-DTF" data="2003-01-01 / 2003-01-01"/>
   <gb:issue.period type="W3C-DTF" data="2006-06-01"/>
   <gb:property type="gb-prop" data="photo"/>
   <gb:homepage type="URL" data="http://www.osaka-cu.ac.jp/"/>
</bib>
```

If you prepare a [bib-file] and execute the following command, an image coordinate system file is generated in [destination-directory]=kochizu1.

```
% xl import-images.xl - - / r64 '/home/gbs/data/nichibun/kochizu1/2813459.tif' kochizu1 bib.xl
```

When selecting either r64 or mtx as the file format, it is recommended to select r64 as a general guideline, as long as the non-compressed file size does not exceed 4Gbytes; for larger files, it is recommended to choose mtx. Although files stored in the r64 format must be smaller than 4Gbyte, the compression ratio is high and hence the network load required when transmitting images is small. Relatively speaking, the network load is higher when transmitting images using mtx is larger than when using r64, but the advantage is that the file size can be larger than 4Gbytes. In the future, as the image transfer capacities, performance and function of mtx files are improved, the preferred file format can be expected to shift toward the mtx format gradually.

Warp point menu files are saved in the directory where this script is executed under the file name wp.xl. Since the URL of each image coordinate system is calculated from the position of this wp.xl file, it is recommended to choose [destination-directory] as a directory under the directory where this script is executed.

If process=all is specified, all processes are executed. If process=wp is specified, only warp point menus are generated.

■ **Return Value**

- [destination-directory]/image file

- wp.xl warp point menu file

■ **Error**

■ **References**

■ **Bugs**

In versions ver.B.b16.10 or earlier, there is a bug where an error occurs when using the mtx format. Please use ver.B.b16.11 or later.

2.3.7 utm2bl-mtx.xl (Converting UTM Images to Ellipse Latitude/Longitude Coordinates)

■ **Prototype**

xl utm2bl-mtx.xl / [dest-mtx] [longitude-min] [longitude-max] [latitude-min] [latitude-max] [longitude-resolution] [latitude-resolution] [src-directory] [elipsoid-a] [elipsoid-b]

■ **Arguments**

[dest-mtx] Conversion source matrix file name
[longitude-min] Range of bl created, minimum value of longitude -180degrees to 180degrees
[longitude-max] Range of bl created, maximum value of longitude -180degrees to 540degrees
[latitude-min] Range of bl created, minimum value of latitude -90degrees to 90degrees
[latitude-min] Range of bl created, maximum value of latitude -90degrees to 90degrees
[longitude-resolution] Resolution in the longitude direction, dot/degree
[latitude-resolution] Resolution in the latitude direction, dot/degree
[src-directory] Conversion source directory
[elipsoid-a] Long diameter of ellipse (m)
[elipsoid-b] Short diameter of ellipse (m)

option

- **leveloing=on,off** The default is **on** .
- **part-step= [integer]** The default is **1** .
- **part-offset= [integer]** The default is **1** /

■ **Environemnts Belongs**

Agent xl [2]

■ **Explaination**

This command transforms matrix data represented in the UTM reference system into data represented in the ellipsoidal latitude/longitude coordinate system.

[longitude-min] and [longitude-max] indicate the range of the image after conversion in the longitude direction. Since it is also possible to convert data to a range including the date line, [longitude-max] can be up to However, [longitude-max]- [longitude-min] cannot exceed Similarly, [latitude-min] and [latitude-max] indicate the range in the latitude direction.

[longitude-resolution] and [latitude-resolution] determines the resolutions in the longitude and latitude directions, respectively. Note that it is not desirable to process high latitude areas and low latitude areas with the same resolution in the longitude direction. The resolution in the longitude direction must be lowered for high latitude areas. For this reason, it is normally desirable to convert a matrix file after dividing it into a southern high latitude area, an area around the equator and a northern high latitude area, and adjusting [latitude-resolution] for each area.

[src-directory] indicates the directory in which conversion source UTM data is saved. Each UTM zone data must be saved in a subdirectory with the zone name. Each subdirectory may contain img.crd, img.map, img.lst and img.mtx files and conform to the UTM coordinate format. Note that the name of the subdirectories must be composed as [zone number] or [zone number]S, [elipsoid-a]s, [elipsoid-a]N, or [elipsoid-a]n. Names without alphabetic characters at the end of a number or those with N or n indicate zones in the northern hemisphere while the names with S or s indicates zones in the southern hemisphere.

[elipsoid-a] and [elipsoid-b] determines the long and short diameters of a speroid. Set these values after checking the spheroid on which UTM is based.

If the option leveling=off is specified, leveling at the last stage is not performed. It is possible to use this option when executing utm2bl-mtx.xl several times and setting layers in a batch at the end.

The options part-step and part-offset are used to acquire UTM files in intervals rather than acquiring all the UTM files. If you specify part-step=2, for instance, every other UTM file is converted. If you specify part-offset=3, conversion is started from UTM3, etc. In general, since distortion of the UTM coordinate system is small at high latitudes, this option can be used for northern latitudes of 60degrees or higher, where even number UTM data is tied to odd number UTM.

Conversion method

The method of conversion when UTM data is given as data saved in multiple world files is explained briefly here.

1. Put together world file data in each UTM zone.

2. Convert world files of each zone to a single matrix data file set using the worldfile-mtx.xl (Section 2.3.5) script. At this point, the conversion destination directory must be set to the subdirectory name above.

   It is not necessary that the generated UTM data can be viewed. If you use the leveling=off option at this conversion, the last task of data leveling is not performed and the processing time can be shortened.

3. Then, use this script and convert the data to a single data list.

```
[geosage]$ ls
52  54 53n  bl
[geosage]$ xl utm2bl-mtx.xl - - / bl/img0.mtx -180 180 -60 60 3600 3600  .  6378137 6356752
....
[geosage]$ xl utm2bl-mtx.xl - - / bl/img1.mtx -180 180 60 90 1800 3600  .  6378137 6356752
....
[geosage]$ xl utm2bl-mtx.xl - - / bl/img2.mtx -180 180 -90 -60 1800 3600  .  6378137 635675
....
[geosage]$ ls bl
img0.mtx
[geosage]$
```

4. Finally, create a single crd file, and create a map file and lst file according to img0.mtx to img2.mtx and map all of them to the aforementioned crd file.

■ **Return Value**

- [dest-mtx] Matrix file

■ **Error**
■ **References**
■ **Bugs**

2.3.8 worldfile.xl (Conversion of World Files)

■ **Prototype**

xl worldfile.xl / direct [prefix] [unit] [crd]
xl worldfile.xl / csv [encoding] [filename]

■ **Arguments**

[prefix] Extension of conversion source data
[unit] Length unit of conversion source coordinate system
[crd] Mapping destination coordinate system file after conversion
[encoding] Character coding name of CSV files
[filename] CSV file name

Option

•

■ **Environemnts Belongs**

Agent xl [2]

■ **Explaination**

If you specify **direct** for the first argument, multiple world files are mapped to a single coordinate system [crd]. If you specify **csv** for the first argument, each world file read from a csv file [filename] written in a given format is mapped to a different coordinate system, and the coordinate system is further mapped to a coordinate system specified by [crd] based on the world file.

Conversion by the direct option is used when multiple world files are considered to be parts of a single image. On the other hand, conversion by the csv option can be used when multiple world files are completely independent separate images.

In case of conversion using the direct option, [prefix] specifies the extension of a world file image. It does not apply to image files with other extensions. [unit] indicates the unit of data in a world file. This unit must also be the base unit of the mapping destination coordinate system. [crd] indicates the mapping destination coordinate system.

```
$ xl worldfile.xl - - / direct tif m ginowan.crd
....
$
```

If the command above is issued, .lst and .cr files are generated for all the .tif world files in the working directory and .map file for mapping the .lst file to ginowan.crd based on the world file is generated.

In case of conversion using the csv option, specify the character code name used to descrive CVS files for [encoding]. [filename] indicates the target CSV file.

In a csv file, write each world file on a separate line and enter bibliographic information generated for each world file name (without extension) and world file. Insert a control command at the end of a file. Figure Fig.2.1 shows a cvs file opened in Excel.

Lines starting with "#" are comments and are ignored during processing. All the lines starting from "." in the last half of the file are commands. Please refer to section 3 for the detailed explanation on commands.

As a general rule, commands follow a syntax where one command is written on one line, the command name is written at the beginning of the line and options unique to bibliographic data written in each line are given in column B and onward. The commands used in this example are briefly explained below.

Specify a alphabet character type in the .character line. Characters in lines where small is specified are all converted to lower case. If no option is specified, no conversion is performed.

Specify a character string that must be placed before bibliographic data in the .pre.string line. No character strings are prefixed if no option is specified.

.qualifier indicates which line of bibliographic data shall correspond to which qualifier. If no qualifier is specified in a line, it is not included in the bibliographic data.

Specify the data type of the bibliographic data for .qtype.

Specify bibliographic data common to all coordinate systems for .qdata.

Specify a coordinate system URLL to which world files are mapped, regardless of other lines, for .wf.mapping.

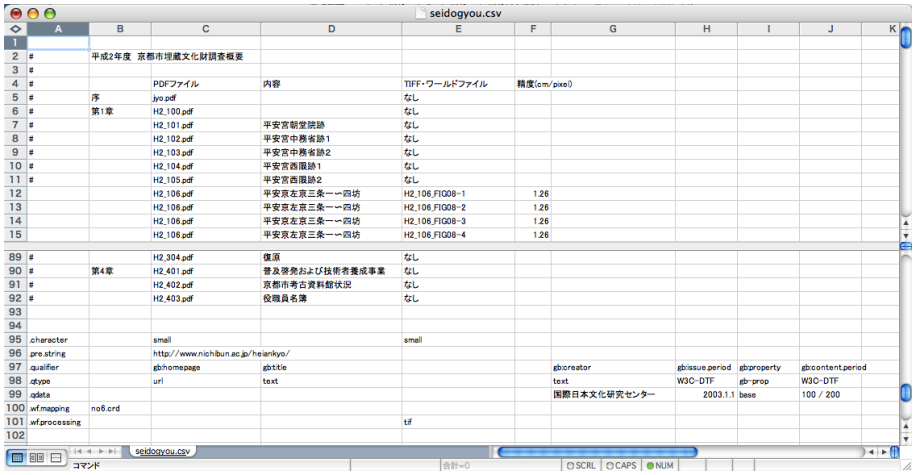

Figure 2.1: CSV file

Specify an image file extension in lines where names of world files to be processed are written for .wf.processing.

■ **Return Value**

- **In case the direct option is used**

  .lst, .cr and .map files corresponding to each world file

  A .map file specifies a mapping from a .lst file to [crd].

- **In case the csv option is used**

  .lst, .cr, .map(1) , .map(2) , and .crd files corresponding to each world file

  .map(1) specifies a mapping from a .lst file to a .crd file.

  .map(2) specifies a mapping from a .crd file to a [crd] file.

■ **Error**
■ **References**
■ **Bugs**

2.3.9 wfutils.xl (Tools Related to World Files)

■ **Prototype**

exl wfutils.xl / generate [.map URL] [.crd URL]
exl wfutils.xl / shift [load-worldfile] [save-worldfile] [x-rate] [y-rate] [x-shift] [y-shift]
exl wfutils.xl / split [load-worldfile] [splited-files trunc] [x-split-nos] [y-split-nos]

■ **Arguments**

[.map URL] Map file URL
[.crd URL] Coordinate system file URL
[load-worldfile] World file name to be read
[save-worldfile] World file name to be written
[splited-files truc] Base name of a divided file
[x-rate] Increase rate of resolution in the x direction
[y-rate] Increase rate of resolution in the y direction
[x-shift] Number of pixels to be moved in parallel to the x direction
[y-shift] Number of pixels to be moved in parallel to the y direction
[x-split-nos] Number of divisions in the x direction
[y-split-nos] Number of divisions in the y direction

Option

- **dir** = [forward/reverse] The default is forward.
- **prefix** =Prefix (including ".") The default is .tfw.
-

■ **Environemnts Belongs**

Agent exl [UNDEF REF (exl)]

■ **Explaination**

This command is supported from ver.B.b17.02.

This tool is used to create world files based on GLOBALBASE contents. It is used together with export-pnm of matrix.xl (Section 2.3.4) and so on. We did not employ a mechanism where world files are generated at the same time as export-pnm in order to allow adding world files to already generated image files at a later stage.

generate

If you specify generate in this script, world files are generated from GLOBALBASE contents. The mapping to be specified for [.map URL] must be an image object .lst file as the source and a coordinate system (.crd) as the destination. The file to be specified for [.crd URL], on the other hand, must be a coordinate system equivalent to the reference coordinate system on which the world files are based. With these specifications, this tool generates world files of .lst image data seen in the coordinate system [.crd URL].

It is possible to specify the **dir** option. If you specify dir=reverse, the order of the X and Y axes can be switched. Moreover, the prefix option specified the prefix attached to world file names.

shift

Starting from world files of images generated from a certain matrix of existing world files, [load-worldfile], this tool outputs data at low resolution from this matrix or creates world files from sub-parts extracted from the entire image.

Specify offset of an image at a certain level from the origin, seen from the level, for [x-shift] [y-shift]. Specify the values of 1/(1<<(level*dim_divide bit) for [x-rate] [y-rate]. In this example, the values shall be calculated assuming dim_divde bit=4. Note that, if the value is 1/4, it can also be specified as /4.

Specify the offset of the image whose resolution was decreased for [x-shift], [y-shift].

split

This tool is used to create a world file corresponding to an image file if one image is divided into multiple images using matrix.xl export-pnm or similar. Consider the case where the original image world file is [load-worldfile] and divided into [x-split-nos] portions horizontally and [y-split-nos] portions vertically. In order to check the image size after the division, specify [splited-file trunc]. This corresponds to [pnm-filename], which is specified by export-pnm of matrix.xl (Section 2.3.4) . Specify only the base name of the file. The file number and prefix are added to this. Multiple world files are created by referring to a series of files.

How to use (creating a single world image file)

1. Create a coordinate system that includes matrix images in such a way that they can be viewed with COSMOS. In this example, the coordinate system is called test.crd and the mapping that connects the matrix images to this coordinate system is called test.map.

2.
```
exl wfutils.xl - - / generate test..map test.crd prefix=.tfw
```

   If you execute the command above, a world file called test.tfw is generated.

3. From the matrix test.mtx connected via test.map, create a ppm file using matrix.xl export-pnm.

```
xl matrix.xl - - / export-pnm test.mtx test.ppm
```

4. Create a tiff image based on the generated test.ppm.

```
pnmtotiff test.ppm > test.tif
```

5. A world file is created based on the pair test.tif and test.tfw.

How to use (creating world image files whose resolution is reduced in some parts)

1. Create a coordinate system that include matrix images in such a way that they can be viewed with COSMOS. In this example, the coordinate system is called test.crd and the mapping that connects the matrix images to this coordinate system is called test.map.

2.
```
exl wfutils.xl - - / generate test.map test.crd prefix=.tfw
```

   If you execute the command above, a world file called test.tfw is generated.

3. From the matrix test.mtx connected via test.map, create a ppm file using matrix.xl export-pnm by specifying level=1 and partial image.

```
xl matrix.xl - - / export-pnm test.mtx test.ppm 1000 2000 500 500 level=1 splitmode=part
```

4. Create a tiff image based on the generated test.ppm.

```
pnmtotiff test.ppm > test.tif
```

5. Create a world file again from test.tfw based on the partial output above.

```
exl wfutils.xl - - / shift test.tfw tmp.tfw /4 /4 250 500
mv tmp.tfw test.tfw
```

   In case level=1, the image resolution is 1/4. Accordingly, the image offset becomes 1/4. Be aware that the exact ratio varies depending on the specification at matrix creation.

6. A world file is created with the pair of test.tif and test.tfw.

How to use (dividing one matrix image into multiple world image files

1. Create a coordinate system that includes matrix images in such a way that they can be viewed with COSMOS. In this example, the coordinate system is called test.crd and the mapping that connects the matrix images to this coordinate system is called test.map.

2.
```
exl wfutils.xl - - / generate test.map test.crd prefix=.tfw
```

   If you execute the command above, a world file called test.tfw is generated.

3. From the matrix test.mtx connected via test.map, extract images divided into 4in the x direction and 8in the y direction.

```
xl matrix.xl - - / export-pnm test.mtx test 4 8 splitmode=split
```

   Files named test-0-0.ppm, test-0-1.ppm, ... are generated.

4. Create a world file again from test.tfw based on the partial output above.

```
exl wfutils.xl - - / split test.tfw test 4 8 prefix=.tfw
mv tmp.tfw test.tfw
```

   The generated ppm file is read and the corresponding tfw is generated.

5. Create a tiff image based on the ppm file.

```
pnmtotiff test-0-0.ppm > test.tif
pnmtotiff test-0-1.ppm > test.tif
.....
```

6. A world file is created based on the pair test-X-Y.tif and test-X-Y.tfw.

■ **Return Value**

- **In case the generate option is used**

  File specified with [save-worldfile]

- **In case the shift option is used**

  File specified with [save-worldfile]

- **In case the split option is used**

  A file group specified with [splited-file trun]

■ **Error**
■ **References**
■ **Bugs**

2.3.10 prog.xl (Progress Display)

■ **Prototype**

xl prog.xl

■ **Arguments**

■ **Environemnts Belongs**

Agent xl [2]

■ **Explaination**

This command is supported from ver.B.b17.02.

Some scripts are equipped with the progress display function. These scripts generate .progressive files, which contain progress status logs. This script displays the contents of such files in readable form.

When a script that takes time to execute is run in one process, this command can be used to show the progress in another process; simply move to the same working directory and execute this command. Press ctrl+C to terminate the command.

■ **Return Value**

None.

■ **Error**

■ **References**

■ **Bugs**

2.3.11 csv-plot.xl (Conversion of CSV Plot Data to Matrix)

■ **Prototype**

xl csv-plot.xl [CSV-FILE] [CORRESPOND] [MATRIX] [X-DOTS] [Y-DOTS]

■ **Arguments**

[CSV-FILE] CSV file of plot data
[CORRESPOND] Corresponding conversion file
[MATRIX] Save destination matrix file
[X-DOTS] Number of dots in the x direction
[Y-DOTS] Number of dots in the y direction

Option

•

■ **Environemnts Belongs**

Agent exl [UNDEF REF (exl)]

■ **Explaination**

If you have installed LANDSCAPE, you can find sample data in /usr/local/xl-gbs/xlsamples/gbs/plot1. Please refer to it when you read the following.

This script converts plot data listed in a CSV file **[CSV-FILE]** to a matrix file with the format given by **[MATRIX]** . The plot data converted to matrix data covers the entire data set; if some of the plots cannot be displayed on the screen, the script behaves as follows.

1. Shrink icons
2. Change icons to simple colored points if they cannot be displayed even if they are shrunk
3. Merge points and display them as an area if there are many plots within one pixel on the screen

With these operations, an almost inifinite number of plots can be displayed.

While matrix data is basically an integer-type representation, plots tend to be sequential (real-valued) coordinate systems, for example plots against latitude/longitude. For this reason, a resolution must be specified for the matrix, and plot coordinates are converted to integers accordingly. That is, a real number coordinate system is divided into very small grids and plot positions are expressed as the grid's integer coordinate. **[X-DOTS]** **[Y-DOTS]** indicate the size of the integer grid.

Attributes assigned to one plot inside a matrix are identified by numbers starting from 0,e.g., the Nth attribute of a plot at coordinate XX,YY. Each attribute has a data type. This data type specifies how to merge plots if it becomes necessary to merge plots during the process of lowering the resolution of a matrix image (grid). To facilitate this, the following data types have been defined. First, teh basic data types are explained.

- **mxPgTYPE_NONE**

  NONE type, without values

- **mxPgTYPE_STRING**

  Character string type

- **mxPgTYPE_INT**

  Real number type

- **mxPgTYPE_RGBA**

  8-bitfull color type with transparency channel

In addition, each basic data type has a number of more detailed sub-types.

- **mxPgNONE**

  NONE type

- **mxPgSTRING**

  Character string type. It ignores character strings of upper level nodes of a matrix and assigns new character strings.

- **mxPgSTRING_INHERIT**

  Character string type. It inherits character strings of upper level nodes of a matrix (users are not permitted to use this type).

- **mxPgSTRING_ERASE**

  Character string type. It ignores character strings of upper level nodes of a matrix and assigns empty character strings (users are not permitted to use this type).

- **mxPgINT_ADD**

  Integer type. It assigns the total value of plots to be merged in lower level nodes to upper level nodes.

- **mxPgINT_AVG**

  Integer type. It assigns the average value of plots to be merged in lower level nodes to upper level nodes.

- **mxPgINT_MAX**

  Integer type. It assigns the maximum value of plots to be merged in lower level nodes to upper level nodes.

- **mxPgINT_MIN**

  Integer type. It assigns the minimum value of plots to be merged in lower level nodes to upper level nodes.

- **mxPgRGBA**

  RGBA type. It assigns the average value of each channel color to upper level nodes.

A CSV file **[CSV-FILE]** contains a simple list of attributes and does not indicate coordinates. Similarly, attributes of a CSV file has no type. For this reason, it is necessary to specify:

1. Which lines of a CSV file should be regarded as coordinate values
2. To which attributes of a matrix file the remaining lines of a CSV file should be assigned
3. Which type should be used at assignment

These specifications are made by a file specified by **[CORRESPOND]** . [CORRESPOND] is an XL script file, but only few instructions actually use it. /usr/local/xl-gbs/xlsamples/plot1/gaz.xl is one of them. When you open it, you can see that the only instructions used are ?xl, Define, and gmxPgPlotField.

Variables defined by Define

```
<Define> ^reso    3600 </Define>
```

The script above defines resolution in the reso variable, in this case 3600dots/degree. In gaz.csv in the example, the coordinates are specified in lines A and B as degrees while the matrix file defines 1second as 1dot. The command below, which is placed at the end of the file, indicates the number of lines at the beginning of the csv file to be ignored,

```
<Define> ^StartLine 1 </Define>
```

In this example, the first one line is ignored.

gmxPgPlotField

The following gmxPgPlotField specifies correspondence among each line of the csv file, coordinates of the matrix file and attribute numbers. gmxPgPlotField is a function of Agentgbmx. Please refer to the section of "gmxPgPlotField" in the manual "gbmx Agent Reference Manual"[5] for the detailed explanation of the function. It is used in the following 2ways in csv-plot.xlscript.

```
(A)
<gmxPgPlotField id="^mtx" csv="A" merge-type="^mxPgTYPE_INT" dim="0">
  ^Fields ^mxPgFDIT_DD ^reso (* 180 reso) 0</gmxPgPlotField>



(B)
(B-1)
<gmxPgPlotField id="^mtx" csv="C" merge-type="^mxPgTYPE_STRING" label="5">
  ^Fields utf-8 ([quote type="direct"] '([field name="LABEL"] ($ ___convert)))</gmxPgPlotField>
(B-2)
<gmxPgPlotField id="^mtx" merge-type="^mxPgTYPE_RGBA" label="0">
  ^Fields 0 0xff000000</gmxPgPlotField>
(B-3)
<gmxPgPlotField id="^mtx" csv="D" merge-type="^mxPgTYPE_STRING" label="1">
  ^Fields utf-8 () test-http </gmxPgPlotField>
```

(A) shows a method of specifying coordinates while (B) shows a method of specifying correspondence between attributes. The following XML attributes can be used.

- **id=" ^ mtx"**

  This XML attribute specifies the ID of a matrix and is fixed to this character string in case of csv-plot.xl.

- **csv**

  This XML attribute specifies a number of lines corresponding to attributes of a csv file, A,B,C,.... This XML attribute can be omitted.

- **label**

  It is used in pattern (B) ; the label XML attribute specifies the attribute number of a matrix. Lines in csv XML attribute correspond to a plot attribute number specified by the label XML attribute. If the csv XML attribute is omitted, the data value given by the raw data of gmxPgPlotField is saved in the label attribute numbers of all plots.

  Although the label XML attribute can be omitted, either the label or the dim XML attribute must exist.

- **dim**

  This XML attribute is used in pattern (A) and specifies the coordinate axis number of a matrix. As an intrinsic function of a matrix, it is possible to specify values up to the number of matrix dimensions - 1,but since plot-csv.xl supports only 2-dimensionalmatrices, the only permitted values are 0or 0specifies the x axis while 1specifies the y axis.

  Although the dim XML attribute can be omitted, either the dim or label XML attribute must exist.

- **merge-type** This specifies the detailed attribute type of a matrix.

Lastly, how the element data of gmxPgPlotField is specified varies depending on the data type given by the merge-type XML attribute.

- **mxPgNONE**

  None

- **mxPgSTRING**

  [Pointer to work area] [saved character code name] [conversion formula] [initialization character string: arbitrary]

  If **[conversion formula]** is set to (), no special processing is performed on the CSV data, which is simply set to the attribute value. If **[conversion formula]** is given as an XL conversion formula,

the CSV data converted by this formula is set to the attribute value. The example (B-1) above corresponds to this. The CSV data is stored in the "___convert" area. The actual conversion is executed when a CSV table is created for the first time; a $ symbol cannot be interpreted at this point and it is thus necessary to use direct quotes.

If no csv attributes are specified, **[initialization character string: arbitrary]** becomes necessary; this character string is passed to **[conversion formula]** and set for all plot attributes as a constant value.

- **mxPgINT_ADD mxPgINT_MAX mxPgINT_MIN**

  [Pointer to work area] [data type in CSV] [conversion resolution] [conversion offset] [count] [initialization character string: arbitrary]

  Data in a CSV file is first interpreted as the data type indicated by **[data type in CSV]** , which is read as double-precision floating point data. The detailed explanation on **[data type in CSV]** will be given later. The data read is converted by the formula ([read data]x [conversion resolution]+ [conversion offset]). Any fractions after the conversion are omitted, so that the converted output is an integer, and the result is set as the attribute data or coodinate value of the matrix. **[Conversion resolution]** and **[conversion offset]** must be given as double-precision floating point values.

  [Count] is ignored in these three data types; it is sufficient to enter 0for the attribute.

  If no csv attributes are specified, **[initialization character string: arbitrary]** becomes necessary; this character string is given to **[conversion formula]** and set for all plot attributes as a constant value.

- **mxPgINT_AVG**

  [Pointer to work area] [data type in CSV] [conversion resolution] [conversion offset] [count] [initialization character string: arbitrary]

  The method of calculating the attribute data is the same as above. **mxPgINT_ADD mxPgINT_MAX mxPgINT_MIN** This data and **[count]** is saved as attribute data, as a pair of two integers. In case of this type, in order to merge plots with higher layers of a matrix, an average of the plots is taken and set as a new plot attribute value. This average is calculated weighted by **[count]** and the **[count]** value in the higher layers is set to the total value of **[count]** of lower layers.

  If no csv attributes are specified, **[initialization character string: arbitrary]** becomes necessary; this character string is passed to **[conversion formula]** and set for all plot attributes as a constant value.

- **mxPgRGBA**

  [Pointer to work area] [count] [initialization RGBA: arbitrary]

  This function acquires RGBA data from CSV and assigns it to plot attributes. If it is given as hexadecimal, the format used in C language to specify pixel values, e.g. -16777216,must be used. **[Count]** works in the same way as integer type [count] and is used for weighting when calculating colors of higher layers. No weighting is performed if 0is specified.

  If no csv attributes are specified, **[initialization RGBA: arbitrary]** becomes necessary; this value is passed to **[conversion formula]** and set for all plot attributes as a constant value.

Integer type [Data type in CSV] The following values can be specified for this type.

- **mxPgFDIT_INT**

  The data type of CSV is integer (64-bitinteger).

- **mxPgFDIT_FLOAT**

  The data type of CSV is double-precision floating point.

- **mxPgFDIT_DD**

  The CSV data type is decimal point expression of latitude/longitude, for example means 135degress 30minutes.

- **mxPgFDIT_DDMMSS**

  The CSV data type is sexagesimal expression of latitude/longitude, for example means 135degrees, 50minutes, 12seconds.

How to use /usr/local/xl-gbs/xlsamples/gbs/plot1
When you want to disclose plot data, it is possible to disclose it using the following procedure.

1. Copy /usr/local/xl-gbs/xlsamples/gbs/plot1 to a disclosure point. For example, if the directory is myplot,
2. edit the img.crd bibliographic data in the myplot directory.
3. Preprare the CSV file of plot data to be disclosed and save it as a file in the myplot directory, for example as myplot.csv.
4. Edit gaz.xl file for myplot.csv to match attributes.
5. ```
   Execute xl csv-plot.xl - - / myplot.csv gaz.xl img.mtx [X-DOTS] [Y-DOTS].
   ```
6. ```
   Execute xl makefile.xl - -.
   ```

■ **Return Value**
■ **Error**
■ **References**
/usr/local/xl-gbs/xlsamples/gbs/plot1
■ **Bugs**

Chapter 3

CSV Commands for worldfile.xl Scripts Reference

3.1 Abstract

This is a reference manual of commands to be set in CSV files referenced when **csv** is specified for the worldfile.xl (Section 2.3.8) script.

All commands used in CSV files are specified as follows: command name first, followed by options. The option specified in each line affects data in the same line appearing before that.

This reference manual describes each command name in the "prototype" section and options in the "argument" section.

3.2 Inheritance Information

Please refer to section 2.3.8 .

3.3 Scripts

3.3.1 # (Comment)

■ **Prototype**

#

■ **Arguments**

Arbitrary

■ **Environemnts Belongs**

■ **Explaination**

No action is performed in the line.

■ **Return Value**

■ **Error**

■ **References**

■ **Bugs**

3.3.2 .character (Character Type Determination)

■ **Prototype**

.character

■ **Arguments**

- small
- capital

■ **Environemnts Belongs**

■ **Explaination**

Alphabetic characters appearing in lines where small is specified are all converted to lower case. Alphabetic characters appearing in lines where capital is specified are all converted to upper case.

■ **Return Value**

■ **Error**

■ **References**

■ **Bugs**

3.3.3 .pre.string (Prefixing)

■ **Prototype**

.character

■ **Arguments**

Character string to be prefixed

■ **Environemnts Belongs**

■ **Explaination**

In lines where character strings to be prefixed are specified, the character strings are attached at the start of the data.

■ **Return Value**

■ **Error**

■ **References**

■ **Bugs**

3.3.4 .post.string (Postposing)

■ **Prototype**

.character

■ **Arguments**

Character string to be posted

■ **Environemnts Belongs**

■ **Explaination**

In lines where character strings to be postfixed are specified, the character strings are attached at the end of the data.

■ **Return Value**

■ **Error**

■ **References**

■ **Bugs**

3.3.5 .qualifier (Qualifier Specification)

■ **Prototype**

.qualifier

■ **Arguments**

Qualifier name

■ **Environemnts Belongs**

■ **Explaination**

Lines with qualifier names are included in bibliographic information of the qualifier specified as bibliographic information of each coordinate system. Other lines are ignored.

■ **Return Value**

■ **Error**

■ **References**

■ **Bugs**

3.3.6 .qtype (Data Type Specification)

■ **Prototype**

.qtype

■ **Arguments**

Type name

■ **Environemnts Belongs**

■ **Explaination**

This command specifies the type of bibliographical information. The qualifier corresponding to the line to which the option is given determines the type given by the option. The option of this command must not be given to lines other than those to which the option is given by .qualifier (Section 3.3.5) .

■ **Return Value**

■ **Error**

■ **References**

■ **Bugs**

3.3.7 .qdata (Data Specification)

■ **Prototype**

.qdata

■ **Arguments**

Data

■ **Environemnts Belongs**

■ **Explaination**

This command specifies data common to all coordinate systems. The qualifier corresponding to lines to which the option is given becomes the data in all coordinate systems. The option of this command must not be given to lines other than those to which the option is given by .qualifier (Section 3.3.5) .

■ **Return Value**

■ **Error**

■ **References**

■ **Bugs**

3.3.8 .wf.mapping (Mapping Destination Specification)

■ **Prototype**

.wf.mapping

■ **Arguments**

Coordinate system URL

■ **Environemnts Belongs**

■ **Explaination**

The option can be specified at any lines. Moreover, this option does not affect other lines. Coordinate systems of all data are mapped to a coordinate system of a given URL according to the world file.

■ **Return Value**

■ **Error**

■ **References**

■ **Bugs**

3.3.9 .wf.processing (Specification of File Name line)

■ **Prototype**

.wf.processing

■ **Arguments**

Image file extension

■ **Environemnts Belongs**

■ **Explaination**

A line to which an image file extension is specified as an option indicates a line where a conversion source image file is saved. The image file name specified in the line must not have any extension.

■ **Return Value**

■ **Error**

■ **References**

■ **Bugs**

Bibliography

[1] Mori Hirohisa and Fujita Haruhiro. *LANDSCAPE Startup Manual*. GLOBALBASE PROJECT, 2006.

[2] Mori Hirohisa. *xl(standard) Agent Reference Manual*. GLOBALBASE PROJECT, 2006.

[3] Mori Hirohisa. *COSMOS Startup Manual*. GLOBALBASE PROJECT, 2006.

[4] Mori Hirohisa. *COSMOS Reference Manual*. GLOBALBASE PROJECT, 2006.

[5] Mori Hirohisa. *gbmx Agent Reference Manual*. GLOBALBASE PROJECT, 2006.

History

1. **Date:** 2009-12-19
   Generatting this manual。(Edition 2009-12-19)
2. **Date:** 2006-10-15
   Author: Hirohisa Mori **TargetVersion:** ver.B.b13
   Created this manual.
3. **Date:** 2007-04-08
   Author: Hirohisa Mori **TargetVersion:** ver.B.b15
   Added the matrix.xl script.
4. **Date:** 2007-04-18
   Author: Hirohisa Mori **TargetVersion:** ver.B.b15
   Added the p4m.xl script, as well as functions to worldfile-mtx.xl and matrix.xl.
5. **Date:** 2007-04-26
   Author: Hirohisa Mori **TargetVersion:** ver.B.b16
   Added the import-images.xl script and functions to matrix.xl.
6. **Date:** 2007-04-26
   Author: Hirohisa Mori **TargetVersion:** ver.B.b16.05
   Added a reverse option to p4m.xl.
7. **Date:** 2007-07-19
   Author: Hirohisa Mori **TargetVersion:** ver.B.b16.05
   Corrected the description of import-images.xl.
8. **Date:** 2007-07-22
   Author: Hirohisa Mori **TargetVersion:** ver.B.b16.10
   Corrected the description of worldfile.xl.
9. **Date:** 2007-09-07
   Author: Hirohisa Mori **TargetVersion:** ver.B.b16.13
   Corrected the description of csv-plot.xl.
10. **Date:** 2007-10-06
    Author: Hirohisa Mori **TargetVersion:** ver.B.b16.14
    Added the export-pnm option level to matrix.xl.
11. **Date:** 2007-10-18
    Author: Hirohisa Mori **TargetVersion:** ver.B.b16.14
    Added description of script utm2bl-mtx.xl.

12. **Date:** 2008-02-21

    Author: Hirohisa Mori **TargetVersion:** ver.B.b17.02

    Added the wfutils.xl script. Improved export-pnm in matrix.xl. Added the script prog.xl.

13. **Date:** 2009-01-30

    Author: Reiko Inoue Bendtsen **TargetVersion:** ver.B.b17.12

    Transrated this manual into English