Plan View Display 1.5-ngc Release Notes

This release note provides the following release-specific information for MÄK Plan View Display 1.5-ngc:

Document ID: PVD-1.5-2-011102

Supported Systems and System Requirements

Plan View Display 1.5 is being released for the following platforms:

- SGI IRIX 6.5 (Plan View Display 1.5-ngc only)
- PC with Red Hat Linux 6.0 (Linux 2.2.5) and 7.1
- PC with Windows NT 4.0, Windows 95/98/2000.

Platform and Compiler Requirements for Plug-ins

See the table of platforms and compilers in the *VR-Link 3.7-ngc Release Notes*.

System Requirements for PCs

To install the Plan View Display on a PC, you must have a system with the following minimum requirements:

- Processor: Intel Pentium 300 MHz or equivalent.
- RAM: The minimum required for your operating system, plus:
  - 32 MB for DIS exercises
  - 64 MB for HLA federates

Depending on the size of your terrain databases, you may need more RAM.

- Hard disk space - Approximately 120 MB. Most of this is for the terrain databases shipped with the Plan View Display. You may need considerably more space depending on the databases you plan to use.
- Two or more multiples of the amount of RAM for virtual memory. Running the Plan View Display in HLA may require increased amounts of virtual memory.

System Requirements for UNIX

To install the Plan View Display for UNIX, you must have a system with the following minimum requirements:

- RAM: 32 MB, 64 MB recommended

Depending on the size of your terrain databases, you may need more RAM.

- Hard disk space: Approximately 120 MB. Most of this is for the terrain databases shipped with the Plan View Display. You may need considerably more space depending on the databases you plan to use.

Linux Support

The Plan View Display will run on Red Hat Linux 6.0 and 7.1. To build plug-ins on Linux 7.1, you need to get the Linux 7.1 version of VR-Link and you need to use the Linux compiler's backward compatibility flag. Plug-ins built against Linux 7.1 are not compatible with the DMSO RTI NG.

Network Compatibility

HLA only

Plan View Display 1.5 was built against VR-Link 3.7-ngc and is compliant with:

- RPR-FOM 0.5, 0.7, 0.8, 1.0, and 2.0
- MÄK RTI 1.3.5-ngc
- DMSO RTI NG v1, v2, v3.0, v3.1, v3.2, and 4.0.

Plan View Display 1.5 is compatible with applications that use earlier versions of VR-Link if they support versions of the RPR FOM listed above.

DIS only

Plan View Display 1.5 supports DIS 2.0.4, IEEE 1278.1, 2.1.4, and IEEE 1278.1a, and can therefore interoperate with DIS applications of any of these versions.

RTI Support

The MÄK RTI is included on all MÄK product CDs. It is link-compatible with the DMSO RTI NG. Plan View Display 1.5 has been tested with DMSO RTI NG v1, v2, v3.0, v3.2, and 4.0.

If you use the MÄK RTI, be sure that the Plan View Display can find your FED file, and optional *rid.mtl* file, by putting them in the directory from which you are running, or by setting the environment variable RTI_CONFIG to the directory that contains them.

RTI Library Names

Plan View Display 1.5 expects library names that match those of DMSO RTI NG v3.2. If you run Plan View Display 1.5 with DMSO RTI NG v3.2, both the UNIX and PC versions work without any modification. If you want to run Plan View Display 1.5 with DMSO RTI NG v1 or v2, you must make the following changes:

- For UNIX, you must use the libraries *libRTI-NGv2.so* or *libRTI-NGv1.so*, which are supplied by MÄK in the *./lib* directory. These libraries correspond to DMSO RTI NG v1 and v2. To use Plan View Display 1.5 with RTI NG v1 or v2, copy the appropriate library to the DMSO RTI NG *lib* directory and rename it *libRTI-NG.so*. (You might want to rename the v3.x *libRTI-NG.so* library first.)
- For PCs, rename the libraries that come with RTI NG v1 or v2 to the names used by the RTI NG v3.2 libraries (*libRTI-NG.lib* and *libfedtime.lib*).

FOM Support

Plan View Display Release 1.5 has built-in support for versions 0.5, 0.7, 0.8, 1.0, and 2.0 of the RPR FOM. It also supports VR-Link's ability to support alternative FOMs through the FOM Mapper. The default version of the RPR FOM supported by the Plan View Display Release 1.5 is version 1.0.

As described in Chapter 6 of *Plan View Display User's Guide*, you can specify which FOM Mapping code to use with the **`-f`** startup option, in one of two ways:

```
Pvd -f { vlRPR05 | vlRPR07 | vlRPR08 | vlRPR10 | vlRPR20 }
Pvd -f library:init_data
```

If the argument to the **`-f`** option is one of the **`vlRPRxx`** keywords, the Plan View Display uses VR-Link's built-in RPR FOM mapper, configured for RPR FOM version 0.5, 0.7, 0.8, 1.0, or 2.0 as specified by the keyword. For example:

```
Pvd -f vrlRPR05
```

This command instructs the Plan View Display to use the built-in RPR FOM Mapper, configured for RPR FOM version 0.5.

Specifying A FED File that is Appropriate for the FOM Mapper

If you do not use the default version of the RPR FOM with the default FED file, you must specify the one you want to use. The Plan View Display comes with the following FED files:

- *VR-Link.fed*, based on RPR FOM 1.0 (default FED file and RPR FOM version)
- *VR-Link20.fed*, based on RPR FOM 2.0
- *VR-Link08.fed*, based on RPR FOM 0.8
- *VR-Link07.fed*, based on RPR FOM 0.7
- *VR-Link05.fed*, based on RPR FOM 0.5

The following example instructs the Plan View Display to use the RPR FOM 0.5 FOM Mapping code (**`-f vlRPR05`**), and also instructs it to use the *VR-Link05.fed* file (**`-x VR-Link05`**).

```
Pvd -f vrlRPR05 -x VR-Link05
```

New Features and Changes

This release of the Plan View Display has the following new features:

- Improved support for DTED databases
- Improved support for paper map records
- Support for extent records

Improved Support for DTED Databases

DTED Terrain Database Records

DTED data is a grid of elevations. By default, the Plan View Display shows a DTED database as projected onto UTM coordinates with white-on-green relief shading.

In a DTED terrain database record, you specify the subsampling rate, clipping latitude and longitude values, display parameters, and one or more DTED files. The syntax is as follows:

```
PvDtedMetafileRecord
{
    PostSpacing rate
    Projected flag
    LowColor red green blue
    HighColor red green blue
    LowColorHeight height
    HighColorHeight height
    ReliefShaded flag
    Origin latitude longitude
    MinMaxLatitude min_latitude max_latitude
    MinMaxLongitude min_longitude max_longitude
    SeaRectangleAdded flag

    File filename 1.dt0 (or .dt1, .dt2, .dt3, .dt4, or .dt5)
    File filename 2
    .
    .
    .
    File filename n
}
```

where:

- **PostSpacing** is an integer indicating the subsampling rate. The default value, 5, loads every fifth value along each dimension from the raw DTED data.
- **Projected** specifies whether or not the database should be projected onto UTM coordinates. The default value, True, projects the database. The value False means do not project the database.
- **LowColor** specifies the RGB values of the low color to be displayed. For relief shading, the low color is used for southwest facing polygons, and for height shading, the low color is used for polygons at the low color height and lower.
- **HighColor** specifies the RGB values of the high color to be displayed. For relief shading, the high color is used for northeast facing polygons, and for height shading, the high color is used for polygons at the high color height and above.
- **LowColorHeight** specifies the height, in meters, to use with the *LowColor* value in height shading. An interpolated value between the low color and the high color is used for polygons between the low-color height and the high-color height.
- **HighColorHeight** specifies the height, in meters, to use with the *HighColor* value in height shading. An interpolated value between the low color and the high color is used for polygons between the low-color height and the high-color height.
- **ReliefShaded** specifies whether or not to use relief shading. The default value, True, specifies relief shading. The value False specifies height shading.
- **Origin** specifies the latitude and longitude, in degrees, for the database origin. The default is to use the southeast corner of the first DTED file loaded.
- **MinMaxLatitude** allows you to clip the database from the north and south. No grid point with a latitude, in degrees, larger than the maximum or less than the minimum is loaded.
- **MinMaxLongitude** allows you to clip the database from the east and west. No grid point with a longitude, in degrees, larger than the maximum or less than the minimum is loaded.
- **SeaRectangleAdded** specifies whether or not to add a surface at sea level across the entire database. The default value, False, adds no additional surfaces to the database. This can be useful when DTED data is unavailable for an ocean area.
- **File** is a DTED filename. DTED files typically have extensions *.dt0* (for level 0 DTED), *.dt1* (for level 1 DTED), and so forth.

Improved Support for Paper Map Records

The following section updates "Paper Map Records," on page 4-18 in *Plan View Display User's Guide*.

Paper Map Records

The paper map record specifies the filename of the bitmap file, the coordinate system it uses, and the coordinates of the terrain database that it should overlay.

A paper map record uses an extent to specify a region of the world to view in Geodetic, UTM, or LCC coordinates. An extent is defined by two points, a southwest corner and a northeast corner to construct a "dummy" database that consists of a single, flat polygon. This extent can be used to provide underlying support for paper maps or to view entities in a particular section of the world in the absence of a database for that area.

Paper Map Bitmap Record

The bitmap record specifies the bitmap used to overlay the terrain database and its location. The syntax is:

```
PvPaperMapBmpRecord
{
    File filename.bmp
    CoordinateSystem {Geodetic | Utm | Lambert}
    SWCorner (X, Y)
    NECorner (X, Y)
}
```

where:

- **File** is the name of the bitmap to load. The file must use the bitmap format. (In other words, as used here, bitmap as not synonymous with raster file.)
- **CoordinateSystem** specifies the coordinate system that the **SWCorner** and **NECorner** extents are expressed in. The Plan View Display supports the following coordinate systems:
  - Geodetic
  - UTM
  - Lambert Conformal Conic (LCC, Lambert).

  UTM and LCC are only supported if the underlying terrain or extent is in the same coordinate system.
- **SWCorner** and **NECorner** specify the X and Y coordinates for the southwest corner and the northeast corner of the papermap. In the Geodetic coordinate system, the X and Y values define the corners in latitude and longitude (degrees). In the UTM and LCC coordinate systems, the X and Y values define the corners in meters.

Support for Extent Records

You can specify extent records in the Geodetic, UTM, and LCC coordinate systems. The syntax for each type of extent record is as follows:

```
PvExtentRecord
{
    CoordinateSystem Geodetic
    SWCorner (latitude, longitude)
    NECorner (latitude, longitude)
}

PvExtentRecord
{
    CoordinateSystem Utm
    Origin (latitude, longitude, altitude)
    SWCorner (X, Y)
    NECorner (X, Y)
}

PvExtentRecord
{
    CoordinateSystem Lambert
    Lam0 longitude
    Phi0 latitude
    Phi1 latitude
    Phi2 latitude
    SWCorner (X, Y)
    NECorner (X, Y)
}
```

where:

- **CoordinateSystem** specifies the coordinate system in which the SWCorner and the NECorner are expressed.
- **SWCorner** and **NECorner** specify the X and Y coordinates for the southwest corner and the northeast corner of the papermap. In the Geodetic coordinate system, the X and Y values define the corners in latitude and longitude (degrees). In the UTM and LCC coordinate systems, the X and Y values define the corners in meters.
- **Origin** specifies the UTM origin for the coordinate system in latitude (degrees), longitude (degrees), and altitude (meters). (UTM only)
- **Lam0** specifies the central longitude (degrees) for the LCC projection. (LCC only)
- **Phi0** specifies the central latitude (degrees) for the LCC projection. (LCC only)
- **Phi1** specifies the lower latitude (degrees) for the LCC projection. (LCC only)
- **Phi2** specifies the upper latitude (degrees) for the LCC projection. (LCC only)

New Command Options for the tdbtool

The tdbtool has the following command-line options:

```
tdbtool {input-file |
   -m input-file1 ... input-filen |
   -k levels [-a min-lod] input-file}
   [-b br,lf
   -f input-format
   -l lod-dist
   -r tolerance
   -s x,y[,z]
   -t
   -o output-file]
```

where:

- `-a` (only for use with `-k`) sets the distance to use with external OpenFlight files for the low level-of-detail, paged out representation.
- `-b` balances the tree, with branching *br* and leafing *lf*, for example, `-b 4,4`.
- `-f` specifies the input file format where *format* can be `gdb`, `flt`, `c4b`, or `dtd` (the default is to guess from the filename extension).
- `-k` keeps a given number of levels of a OpenFlight file as external files. This converts the top-level OpenFlight file to a master file and preserves the external structure of the OpenFlight file. It also causes the external OpenFlight files to be converted into matching GDB files). The `-k` argument implies `-f` flt.
- `-l` sets the level-of-detail distance to use for OpenFlight files. Default: 0.
- `-m` creates a master file that loads the input files on demand.
- `-o` specifies the output filename (default: *out.gdb*).
- `-r` reduces the complexity of the terrain by combining vertices within a *tolerance* distance of one another.
- `-s` generates a spatial index grid with cell dimensions *x*, *y*, and (for 3-D grids) *z* measured in database units.
- `-t` converts all polygons to triangles.

Documentation Updates

- The description of the syntax for paper map records and paper map RPF records on pages 4-18 and 4-19 contains an error. The coordinate system should be specified as Geodetic, not PvCoordinateSystemGeodetic. See "Improved Support for Paper Map Records", on page 7 for a complete update of the paper map record syntax.

Known Problems

This section lists problems that we have identified, but which do not interfere with your use of the Plan View Display:

- The information bar at the bottom of the canvas does not always display correct text about selected menu items.
- The Plan View Display may display the following message:

  ```
  Cannot allocate exact pen/brush color.
  ```

  To eliminate this message, try increasing the color depth of the display.

  To disable this message, run the Plan View Display with a lower notification level. Notification levels are described in Section 6.1 "Startup Options" in *Plan View Display User's Guide*.
- Platforms other than Windows NT may not correctly load the *world-hi.map* file. Use the *world-lo.map* file instead.
- The *ground.gdb* and world databases (New global geodetic view in the file menu) do not show their entire extent when first loaded. You must resize the window in order to see their complete extent.