



MÄK Plan View Display 2.5.1 Release Notes

This release note contains the following release-specific information for Plan View Display Release 2.5.1:

Overview	2
Systems Supported	2
New Features and Updates	2
Documentation Updates	2
Changes to the PVD API	2
Creating Overlay Objects	3
Terrain API	3
Bug Fixes	3
Known Problems	4

Overview

Plan View Display 2.5.1 is a maintenance release for Plan View Display 2.5. It fixes problems in the Plan View Display 2.5 release. All of the information in *MÄK Plan View Display 2.5 Release Notes* applies to this updated release. Therefore this release note only documents bug fixes and documentation corrections for release 2.5.1.

Systems Supported



If you run the Plan View Display on IRIX in 8-bit color mode (the default), performance may be severely compromised. To run on IRIX, set the color depth to 24-bit mode.

New Features and Updates

This release has the following new features and updates:

- The *DtTerrainDatabase* class has a new member function, `terrainHeight()`. This function allows you to get the altitude at a specified location or latitude and longitude.
- Release 2.5 changed the way that geometric features from CTDB databases were added to the terrain. Release 2.5.1 reverts to the way features were added in release 2.4. Geometric feature data is, once again, stored in the terrain group node of the terrain database.

Documentation Updates

This section contains updates to *MÄK Plan View Display 2.5 Release Notes* and to the *MÄK Plan View Display 2.5* manuals.

Changes to the PVD API

This section contains changes to the PVD API that were not included in *MÄK Plan View Display 2.5 Release Notes*.

Changes to Existing Classes

The `myCurrentObjectType` member variable of the *DtCreationHandler* class has been changed from a `DtObjectType` to a `DtEntityMapperKey`. If you used to use the `matchPattern()` member function of `DtObjectType`, you can now use a similar function in `DtEntityMapperKey`.

If you want to match the pattern against a `DtObjectType`, you can call the `matchPattern()` member function on `myCurrentObjectType` as follows:

```
myCurrentObjectType.matchPattern(DtEntityMapperKey(-1,
    object.superType(), object.kind(), object.domain(), object.country(),
    object.category(), object.subCategory(), object.specific(),
    object.extra()));
```

Deprecated Objects

- ♦ *DtPvdMiniMap*
- ♦ *DtRouteConfigWidget*.

Creating Overlay Objects

The following information about overlays and overlay objects was not included in *MÄK Plan View Display User's Guide*:

- ♦ If you add an overlay object to the map without creating an overlay, the Plan View Display automatically creates an overlay called Default Layer, and places the overlay object on the Default Layer.
- ♦ If you create an overlay and do not place any objects on it, it does not get saved.

Terrain API

The following code in section 11.3 of *MÄK Plan View Display User's Guide* is incorrect:

```
if (terrainDb->intersect(chord, intersectRecord, DtGdbNodeIRT_POINT |
    DtGdbNodeIRT_INORMAL ))
```

It should read as follows:

```
DtGdbNodeDtIntersectRecordType flag =
    (DtGdbNodeDtIntersectRecordType)(DtGdbNodeIRT_IPOINT |
    DtGdbNodeIRT_INORMAL);

// Try to intersect the terrain to get the left triangle
if(!terrainDb.intersect(chord, intersectionRecord, flag))
```

Bug Fixes

This release fixes the following problems in Plan View Display 2.5:

- ♦ Under certain circumstances, the working object cursor was not displayed while creating an overlay object. This is fixed.
- ♦ The Plan View Display no longer crashes if you try to open a GDB file that does not exist.
- ♦ MGRS coordinates were not being calculated to enough decimal places.
- ♦ Geodetic databases are now converted to geocentric properly when loaded through the GUI or converted using `tdbtool`.

- ♦ When you loaded a *.map* file that referred to multiple terrain files, only the first file was displayed. Now, all files are loaded correctly.
- ♦ Converting GDB files from the release 2.4 format to release 2.5 format resulted in lost terrain. Files are now converted properly.
- ♦ If you ran *tdbtool* with *-s 0,0* or *-s 0,0,0*, it did not delete any existing spatial indexes as it implied. Now, if the GDB file has a spatial index, it will be deleted if you specify *-s 0,0* or *-s 0,0,0*.

Known Problems

Plan View Display Release 2.5.1 has the following known problems:

- ♦ Using large filled polygons, such as minefields, in overlays can degrade performance.

Problems with Terrain Databases

- ♦ Using the *tdbtool* 's balancing option (*-b br,lf*) to balance certain terrains may cause the generated *.gdb* file to display incorrectly. This has to do with a draw ordering bug that is not currently corrected for all terrains. The resulting GDB file will still give correct intersection information, so it will still be suitable for simulation purposes.
- ♦ Balancing larger terrain databases can take an extremely long time.
- ♦ You cannot load vector data with geocentric databases. Vector data works only with geodetic or UTM databases.
- ♦ Do not set the spatial index Z value to be greater than 1. This can cause incorrect intersections.

Problems with Fonts and Icons

- ♦ On IRIX, icons may appear discolored or grainy unless the color depth is set to 24-bit.
- ♦ True Type fonts are not supported on SGI IRIX. Use bitmaps for entity icons.
- ♦ True Type fonts may not display properly until you reboot your computer. If entity icons are displayed as question marks, @ signs, or letters, reboot your computer to properly install the fonts.
- ♦ Uninstalling the Plan View Display in Windows via InstallShield uninstalls the entity fonts. If you have another application that uses these fonts, such as MÄK VR-Forces, the correct entity icons would not be displayed. You must manually install the fonts from their location in the other application that needs them.

Configuration and Protocol Problems

- ♦ If you are using the Plan View Display in a DIS exercise, you may experience problems with dropped packets and entities may time out. To avoid this problem, you can use asynchronous I/O by starting with the *-I* parameter.

- ♦ The Qt translation files (in *.translations*) for built-in Qt-level GUI items do not work properly. The translation files for the Plan View Display work correctly.

