



MÄK Plan View Display 2.9.1 Release Notes

This release note contains the following release-specific information for Plan View Display Release 2.9.1:

Systems Supported	2
Patching the Microsoft Visual C++ 6.0 Compiler	2
Building on Linux	3
Disk Space Requirements	3
Compiler Compatibility on Windows	3
MÄK Product Compatibility	3
Qt Release Compatibility	3
Third-Party Library Requirements	4
Network Compatibility	4
Backwards Compatibility	4
RPR FOM Versions Supported	5
New Features and Updates	6
Bug Fixes	6
Known Problems	7

Systems Supported

Plan View Display 2.9.1 is available for the platforms listed in Table 1. For the availability of other platforms, contact your MÄK salesperson. For toolkit users, application code must be built with the indicated compilers in order to link to Plan View Display libraries.

Table 1: Platforms supported

Operating System	Compiler	glibc
Red Hat Enterprise Linux Workstation 3	GNU C++ (GCC) 3.2.3	2.3.2
Red Hat Enterprise Linux Workstation 4	GNU C++ (GCC) 3.4.4	2.3.4
Red Hat Fedora Core 3	GNU C++ (GCC) 3.4.4	2.3.6
Windows 2000 SP2, XP	Microsoft Visual C++ 6.0, Service Pack 5 Microsoft Visual C++ 7.1	

Patching the Microsoft Visual C++ 6.0 Compiler

The Plan View Display is transitioning to use of the Standard Template Library, particularly for lists and map objects. Microsoft Visual C++ version 6.0 has a bug with the STL that is fixed with the supplied *xtree-msvc++patch* file. If you use MS Visual C++ version 6.0, you must install a patch to a standard header file. The patch comes from Dinkumware, Ltd (the company that developed the Standard C++ Library header files for Microsoft), but for convenience, we are distributing it with Plan View Display releases.

The patch fixes a bug that would cause applications to crash if they pass `std::sets` or `std::maps` from an application to a DLL or vice-versa, something that IEEE 1516 federates must do. For details about the patch go to http://www.dinkumware.com/vc_fixes.html.

To install the patch:

1. In the top level Plan View Display directory, is a file called *XTREE-msvc++patch*. Rename this file *XTREE*.
2. Make a back-up version of the original *XTREE*.
3. In the Visual Studio *include* directory (for example, *C:\Program Files\Microsoft Visual Studio\VC98\Include*), replace the original version of *XTREE* with the patched version.

Building on Linux

The example Makefiles require a patched version of gmake 3.80, which has been included for your convenience in the *./mkbin* directory. This version of gmake is based on version 3.80, but includes a memory fix that is needed to use our makefiles. The source code for the modified gmake is freely available from <http://ftp.mak.com/out/gmake3.80-patch1.tar.gz>. This patch will be included in future versions of gmake.

Before you compile the examples, go to the top level of your installation and create a symbolic link 'VR-Link' to your VR-Link installation and another called 'RTI' to your RTI installation.

Disk Space Requirements

A full installation of Plan View Display requires approximately 300 MB of disk space. Terrain databases use approximately 106 MB and documentation (primarily the class reference) uses 68 MB.

Compiler Compatibility on Windows

MÄK provides versions of product releases that have been compiled with Microsoft Visual C++ 6.0 and 7.1. When you run MÄK products together on the same computer, for example, the Plan View Display and the Stealth, we strongly recommend that you run versions compiled with the same compiler. Mixing products compiled with different versions of the compiler can result in program instability.

MÄK Product Compatibility

To build the Plan View Display, you must link with VR-Link 3.9.6. The Plan View Display does not include a VR-Link developer's license. You must purchase it separately. If you are building for HLA and want to link with the MÄK RTI, use MÄK RTI 2.x.

MÄK Plan View Display 2.9.1 is a parallel release to VR-Forces 3.9.1.

Qt Release Compatibility

If you want to do development using the Plan View Display API, you need to use Qt, a cross-platform GUI toolkit from Trolltech. The Plan View Display GUI was built with Qt release 3.3.5. The Plan View Display does not include a Qt license. You must purchase a license from Trolltech.

Third-Party Library Requirements

DtTerrainProfileWidget is based, in part, on the work of the Qwt project (<http://qwt.sf.net>). To compile examples or user projects, you must download the QWT distribution at <ftp://ftp.mak.com/out/qwt-0.4.1.zip>. This distribution is known to work with VR-Forces. Set the MAK_QWTDIR environment variable to point to the location of the QWT library.

Network Compatibility

HLA only

Plan View Display 2.9.1 was built against VR-Link 3.9.6 and is compliant with:

- ♦ RPR-FOM 1.0 and a subset of 2.0 (draft 6 and 17)
- ♦ MÄK RTI 2.4.2
- ♦ DMSO RTI NG 6.



If you need to run the Linux version of the Plan View Display with the DMSO RTI, contact MÄK technical support.

Plan View Display 2.9.1 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 2.9.1 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.

Backwards Compatibility

Plan View Display 2.9.1 is fully compatible with Plan View Display 2.9.

RPR FOM Versions Supported

Plan View Display 2.9.1 has built-in support for versions 1.0 and 2.0 (draft 6 and 17) of the RPR FOM. It also supports VR-Link's ability to support alternative FOMs through the FOM Mapper. By default, Plan View Display 2.9.1 uses RPR FOM 1.0.

If you are building the Plan View Display and you want to specify a version of the RPR FOM through code, pass the version number (for example, 2.0006) to the *DtRprFomMapper* constructor and pass the resulting object to the *DtExerciseConn* constructor. Also, make sure you are using a federation execution name that corresponds to the right FED file. For example:

```
DtExerciseConn conn("VR-Link20006", "MyApp", new DtRprFomMapper(2.0006));
```

In order to support RPR FOM 1.0, we have added the following extensions (which are supported in later versions of the RPR FOM):

- ♦ EnvironmentProcess
- ♦ GriddedData
- ♦ EmbeddedSystem.IFF
- ♦ EmbeddedSystem.IFF.NatoIFF
- ♦ EmbeddedSystem.IFF.NatoIFF.NatoIFFTransponder
- ♦ EmbeddedSystem.IFF.NatoIFF.NatoIFFInterrogator
- ♦ EmbeddedSystem.IFF.SovietIFF
- ♦ EmbeddedSystem.IFF.SovietIFF.SovietIFFTransponder
- ♦ EmbeddedSystem.IFF.SovietIFF.SovietIFFInterrogator
- ♦ EmbeddedSystem.IFF.RRB
- ♦ BaseEntity.AggregateEntity
- ♦ ObjectRoot.BaseEntity.PhysicalEntity.Lifeform.

For both RPR FOM 1.0 and 2.0 Plan View Display 2.9.1 relies on the LgrControl and ViewControl custom MÄK extensions.

New Features and Updates

Plan View Display 2.9.1 is a maintenance release. It has no new features. Please see *Plan View Display 2.9 Release Notes* for features introduced in Plan View Display 2.9.

Documentation Updates

Section 1.1.2 in *MÄK Plan View Display User's Guide* states that you can publish overlay objects to the network. This is not correct. Overlay objects are only visible locally.

Bug Fixes

This release fixes the following problems:

- A problem that occasionally caused the GUI to hang was diagnosed and corrected.
- The global minimap now displays the window bounding boxes as described in *MÄK Plan View Display User's Guide*.
- Problems caused by the Qt Assistant registry key following repeated installations and uninstallations have been diagnosed and a workaround is provided in the support FAQ (http://www.mak.com/product_support_faq.php#olh).
- Reflected entities with the ID 0:0:0 no longer hang the GUI.
- The system now respects MTL transparency values applied to color.
- Having two open 'Create Selection Group' dialog boxes no longer crashes the GUI.
- The receive buffer size no longer gets overridden in both front and back-end with hardcoded values.
- When you delete a point on an overlay object the correct point gets deleted. Previously the second point always got deleted.
- The Delete key now deletes selected line points for local overlay objects. Previously it deleted the entire object.
- If a circle overlay object is selected and you select another object from the echelon view, the circle no longer stays highlighted.
- The `-l` command line option now works.
- CADRG files can now be loaded even if a full set does not exist.
- The Location label is now correctly positioned.
- The `tdbtool -m` option now works.

Known Problems

Plan View Display Release 2.9.1 has the following known problems:

- ♦ 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.
- ♦ Vector clipping does not work properly with areal features.
- ♦ 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.
- ♦ Using large filled polygons, such as minefields, in overlays can degrade performance.
- ♦ You cannot load vector data with geocentric databases. Vector data works only with geodetic or UTM databases.
- ♦ Intervisibility works with linear and areal features, but not with point features.
- ♦ You cannot use translation files to translate entity labels.
- ♦ If you drag a toolbar onto the Utility Toolbar, you will not be able to dock that toolbar anywhere else in the Plan View Display window. 5080

