



MapXtreme 2008 v6.8.0 Release Notes

Americas:
Phone: 518 285 6000
Fax: 518 285 6070
Sales: 800 327 8627
Government Sales: 800 619 2333
Technical Support: 518 285 7283
www.mapinfo.com

© 2008 Pitney Bowes Software Inc.

All rights reserved. MapInfo and the MapInfo logo are trademarks of Pitney Bowes Software Inc. and/or its affiliates.

June 2008

These Release Notes provide information on enhancements and updates made to MapXtreme 2008 v6.8.0. It also covers bug fixes, known issues, and other important information for this release.

This document is a supplement to the MapXtreme 2008 Developer Guide. Refer to this document for release specific information and to the Developer Guide for overall product information and usage. The Developer Guide is available in three formats and from several locations, including:

- HTML version integrated into Visual Studio (in the Help Contents pane)
- PDF version accessible from the Start > All Programs menu under MapInfo > MapXtreme 2008 v6.8.0.0 > Learning Resources browser
- **PDF version** on the MapInfo website

List of Topics:

♦ What's New	2
♦ What's Changed	10
♦ Upgrading to MapXtreme 2008	11
♦ Bug Fixes	14
♦ Known Issues	17
♦ Documentation Topics	28

What's New

MapXtreme 2008 offers the following new features in version v6.8.0:

- **Supported Environments**
- **New Assemblies**
- **Visual Studio Support**
- **WFS Server Compliancy**
- **Raster Reprojection**
- **Time and DateTime Data Types**
- **Enhanced User-Defined Metadata Support**
- **LanguagePreference Key/Value Pair**
- **Border Control Properties for Printing and Exporting**
- **Contact Information Support on WMS Server**
- **Bi-Directional String Support in Curved Labels**

Supported Environments

This release of MapXtreme 2008 supports the following frameworks, development environments and operating systems. Items in **bold** are new for this release.

Operating Systems*	<ul style="list-style-type: none"> • Windows 2000 Professional SP 4 • Windows 2000 Server SP 4 • Windows 2000 Advanced Server SP 4 • Windows XP Professional SP 2 • Windows XP Professional Edition x64 † • Windows Server 2003 Standard Edition SP2 • Windows Server 2003 Standard Edition x64 ‡ • Windows Server 2003 Enterprise Edition SP2 • Windows Server 2003 Enterprise Edition x64 ‡ • Windows Server 2003 Web Edition SP2 • Windows Server 2008 Standard Edition (x86, x64) ‡ • Windows Server 2008 Enterprise Edition(x86, x64) ‡ • Windows Vista Ultimate (x86, x64) ‡
Development Framework and IDE Support ‡	<ul style="list-style-type: none"> • Microsoft .NET Framework 2.0 SP1 • Visual Studio 2005 SP1 (Professional and Standard Editions) • Visual Studio 2008 (Professional and Standard Editions)

* 32-bit systems only

† 32-bit emulation mode only via Microsoft's Windows-on-Windows (WOW64) technology

‡ Recommended development environments (IDE). Others can be used, however, the MapXtreme installer will not integrate its templates, samples, and help system.

Compilation on the 1.1 Framework is no longer supported. Visual Studio 2003, because it is built on the 1.1 Framework, is no longer supported.

See Chapter 2: Getting Started, in the MapXtreme 2008 Developer Guide for a complete discussion of system requirements, prerequisites and supported databases.

New Assemblies

MapXtreme 2008 includes two additional assemblies: **MapInfo.CoreEngine.Wrapper.dll** and **MapInfo.CommonDialogExt.dll**. These assemblies are automatically included in your project when you build using the MapXtreme templates. MapInfo.CoreEngineWrapper.dll is required for MapXtreme Web Site applications. Both assemblies are required for desktop applications.

If you are building MapXtreme applications from Visual Studio templates or from scratch, you must include these assemblies. Similarly, you will need to add them to existing MapXtreme applications. See [Upgrading to MapXtreme 2008](#) for more information about when and how to include these assemblies in your MapXtreme projects.

Visual Studio Support

MapXtreme 2008 v6.8.0 is built using the Microsoft .NET Framework 2.0 under Visual Studio 2005.

You can use either Visual Studio 2005 or 2008 to develop MapXtreme 2008 applications. You can choose to target .NET Framework versions 2.0, 3.0 or 3.5 and use any of their supported features when developing your MapXtreme application.

All sample applications that ship with MapXtreme are available in both Visual Studio 2005 and 2008. They are accessible from the Learning Resources browser from the Windows Programs menu. Look under Learning Resources in the left pane, and choose Sample Applications. These applications open in Visual Studio from the browser.

Visual Studio 2003 and .NET Framework version 1.1 are no longer supported.

Windows Presentation Foundation (WPF)

MapXtreme does not currently have native Windows Presentation Foundation (WPF) controls. However, you can develop MapXtreme 2008 WPF applications in Visual Studio 2008 by using the WindowsFormsIntegration assembly. This assembly allows WPF and Windows Forms to interoperate. The WindowsFormsIntegration assembly includes a WPF control called WindowsFormsHost that allows you to host Windows Forms controls within a WPF window.

You have two options when hosting a MapXtreme MapControl in a WPF window. You can either host it directly as a child of a WindowsFormsHost or indirectly through a Windows Forms UserControl. The second approach allows you to use Visual Studio to graphically design the Windows Forms portion of your application.

1. Drag a WindowsFormsHost control from the Visual Studio toolbox into the WPF window.

This adds a reference in your project to the WindowsFormsIntegration.dll assembly. It also adds the following namespace attribute to the Window element in your Window's xaml file:

```
xmlns:my="clr-namespace:System.Windows.Forms.Integration;assembly=WindowsFormsIntegration"
```

It also adds a WindowsFormsHost element to the xaml file:

```
<my:WindowsFormsHost Margin="0,0,6,18"
Name="windowsFormsHost1"ChildChanged="windowsFormsHost1_ChildChanged">
</my:WindowsFormsHost>
```

2. Create a Windows Forms UserControl in your project and drag a MapControl from the toolbox to the UserControl's design form.

If you are planning to access the control from WPF code-behind, you can choose to expose the control as a property of the UserControl. You can also choose to keep all of the MapXtreme code encapsulated within the UserControl.

3. To host the UserControl in your xaml Window, add the following namespace attribute to your Window element:

```
xmlns:app="clr-namespace:WpfApplication1"
```

where WpfApplication1 is the namespace containing the UserControl.

4. Add the following element as the WindowsFormsHost child:

```
<my:WindowsFormsHost Margin="0,0,6,18" Name="windowsFormsHost1"
    ChildChanged="windowsFormsHost1_ChildChanged">
    <app:UserControl1 x:Name="userControl1" />
</my:WindowsFormsHost>
```

You can then access your UserControl instance as a field of the Window class called userControl1 in the Windows code-behind file.

Windows Communication Foundation (WCF)

You can use MapXtreme 2008 under the WCF framework. There are, however, some things to keep in mind:

Current Session Object

MapXtreme 2008 does not currently have automatic session management in WCF as it does in ASP.NET. You are responsible for manually disposing the current Session object after every stateless OperationContract. For example, the following service contract implementation assures the current Session is disposed by using a try/finally block and calling static method Session.Dispose() within its finally block.

```
public string[] GetMapAliases ()
{
    try
    {
        List<string> aliases = new List<string>();
        foreach (Map map in Session.Current.MapFactory)
        {
            aliases.Add(map.Alias);
        }

        return aliases.ToArray();
    }
    finally
    {
        Session.Dispose();
    }
}
```

ASP.NET HTTP Module

ASP.NET MapXtreme applications use an ASP.NET HTTPmodule to manage the MapXtreme Session. This module cannot be used in WCF applications. Make sure that your app.config or web.config file in the WCF project *does not* contain the following:

```
<httpModules>
<add type="MapInfo.Engine.WebSessionActivator, MapInfo.CoreEngine,
Version=6.8.0.528, Culture=neutral, PublicKeyToken=93e298a0f6b95eb1"
name="WebSessionActivator" />
</httpModules>
```

3. Depending on how your WCF service is hosted, you might want to use MapXtreme pooling to control how many operations can concurrently use MapXtreme within a service instance. To use object pooling or workspace loading, add the following to the <configuration> element in your app.config or web.config file in your WCF project:

```
<configuration>
  <appSettings>
    <!--Use this setting to turn Session pooling on/off (true/false)-->
    <add key="MapInfo.Engine.Session.Pooled" value="true" />
    <!--Use this setting to save Session state automatically (HttpSessionState) or manually (Manual)-->
    <add key="MapInfo.Engine.Session.State" value="Manual" />
    <!--Use this setting to preload a workspace on Session creation-->
    <add key="MapInfo.Engine.Session.Workspace" value="c:\Program Files\MapInfo\MapXtreme\6.8.0\Samples\Data\World.mws;c:\Program Files (x86)\MapInfo\MapXtreme\6.8.0\Samples\Data\World.mws" />
  </appSettings>
```

For more information on state management, see Chapter 6 in the MapXtreme 2008 Developer Guide.

Raster Reprojection

Raster reprojection enables you to change the cartographic projection of a raster layer in a map. Many types of raster images can be reprojected. Some examples include satellite and aerophoto images, scanned maps, as well as grids, seamless raster tables, and WMS data.

In earlier versions of MapXtreme, only vector layers could be reprojected. If a raster layer was added to one or more vector layers in a map window, all the vector layers were reprojected into the projection of this raster layer. Also, if there was more than one raster layer in a map, the raster layer that took up the maximum screen space area determined the map window projection. Other raster layers were only stretched in both directions to fit to the sizes of that maximum-screen-coverage layer.

Now, you can control the reprojection of both raster and vector layers. When you add either a raster or vector layer to a map, the new layer is reprojected into the current map window projection.

When you change the projection of a map window that contains a combination of vector and raster layers, all the layers, both raster and vector, can be reprojected to the new map window projection.

You can access raster reprojection settings in the MapXtreme API, as well as in the user interface of the Workspace Manager's Layer Control. For information using raster reprojection in the API, see the MapXtreme Developer Reference. For information on the raster reprojection user interface, see the Workspace Manager chapter in the MapXtreme Developer Guide.

Raster reprojection properties can also be read into and from a workspace. MapXtreme can also write raster reprojection information from MapInfo Professional to an .mws workspace. This capability enables you to create your workspace in MapInfo Professional, and then load it into MapXtreme.

Note MapXtreme versions 6.7.1 and earlier ignore raster reprojection parameters.

WFS Server Compliancy

The MapXtreme 6.8.0 WFS Server has passed all of the compliance tests for the Basic profile of the OGC WFS 1.0.0 standard. Certification from the OGC has been applied for. See the OGC Compliance and Interoperability Testing & Evaluation Initiative ([CITE](#)) for more information about the standard and the tests.

Time and DateTime Data Types

Two new data types have been introduced Time, and DateTime. These new types increase MapXtreme's capability to handle date, time, and combined date and time data in queries and maps, as well as enhance interoperability among MapInfo products.

The addition of the new types has required some changes to the way that the Date type is implemented. The following table provides a quick comparison of the Date, DateTime, and Time implementation between MapXtreme 6.7.x and 6.8.

MapXtreme 6.7.x Type > Structure	MapXtreme 6.8 Type > Structure
--	Time > MapInfo.Data.Time
--	DateTime > System.DateTime
Date > System.DateTime	Date > MapInfo.Data.Date

Each of the types and their structures is explained below.

Time Type and Time Structure

The Time data type represents a particular moment in time in hours, minutes, seconds, and fractions of seconds. The MapInfo.Data.Time structure has been implemented to represent the MapInfo Time type, since .NET does not have a Time type structure.

The addition of the Time type provides better interoperability among MapInfo products that also use a Time data type, such as MapInfo Professional.

The smallest unit of fractions of seconds that the Time type accepts is 100 nanoseconds. The object values of the Time type range from 00:00 midnight to 23:59:59.9999999.

The MapInfo.Data.MIDataReader.GetTime() method works only on MapXtreme Time columns and returns a MapInfo.Data.Time value.

For more information on the Time type, see the MapInfo.Data.MIDbType enumeration in the MapXtreme Developer Reference.

DateTime Type

The DateTime data type represents combined date and time values. The DateTime type is matched to the .NET System.DateTime structure.

The smallest unit of fractions of seconds that the DateTime type accepts is 100 nanoseconds, the same as Time.

Note The smallest unit of fractions of sections that the MapInfo Professional DateTime type accepts is millisecond.

The object values of the DateTime type range from 12:00:00 midnight, January 1, 0001 A.D. to 11:59:59.9999999 pm, December 31, 9999 A.D. Any calculations or comparisons of DateTime instances are meaningful only when the instances are created in the same time zone.

DateTime Backward Compatibility

To ensure backward compatibility with earlier versions of MapXtreme, the MapInfo.Data.MIDataReader.GetDateTime method works with both the DateTime and Date types. In both cases, a System.DateTime value is returned. However, the MapInfo.Data.Column.DataType will reflect the actual data type, either Date or DateTime.

Date Type Changes and Date Structure

In earlier versions of MapXtreme, the Date type was mapped to Microsoft .NET System.DateTime. However, because System.DateTime is a closer match to the new DateTime type than the Date type, we have matched the DateTime type to System.DateTime as stated above, and implemented a new MapInfo.Data.Date structure to represent the MapInfo Date type. The new mapping can impact the behavior of applications created in MapXtreme 6.7.1 and earlier depending on how the application was written.

The range of acceptable values in the Date type has been changed so that the range is the same as the range of acceptable date values of the DateTime type. The values range from January 1, 0001 A.D. to December 31, 9999 A.D. This change also makes the acceptable values for the Date type consistent with .NET.

The MapInfo.Data.MIDDataReader.GetDate() method works only on MapXtreme Date columns and returns a MapInfo.Data.Date value.

To support backward compatibility, MapInfo.Data.MIDDataReader.GetDateTime method will also work on a Date type column and the answer will have midnight appended to it.

For more information on the Date type, see the MIDbType enumeration in the MapXtreme Developer Reference.

Data Source Support

The addition of the Time and DateTime types enables MapXtreme to make better use of server data that also has these data types. The DateTime type removes the limitations that existed in earlier versions of MapXtreme of reading DateTime data from servers using the MapXtreme Date type.

MapXtreme can read Date, DateTime, and Time data (and save it back if applied) on the supported data sources and data providers. The different data sources may have different type definitions on date/time, which may or may not match MapXtreme types exactly.

The new data types are supported for the following data sources:

- Mem tables
- Native tables (TAB files)
- ADO.NET
- Oracle via OCI
- MS SQL Server via ODBC
- Informix via ODBC

The ASCII and dBase, and Microsoft Access data sources are not supported.

Date and DateTime Support in Remote Databases

Remote databases may not support all the data types that MapXtreme supports. The table below shows the date and time-based types supported in native TAB files and in each supported database.

MapXtreme	Native TAB Files	ADO.NET	Oracle (OCI)	MS SQL Server	Informix
Date	Date		Date		Date
Time	Time				Time
DateTime	DateTime	DateTime	DateTime	DateTime	DateTime

Return Column Type and Value Changes in Remote Databases

The addition of the new data types has prompted some changes to the return column types and values for remote databases. The table below shows the return column type and value for MapXtreme 6.7.x and MapXtreme 6.8 for the supported types in each remote database.

Server / Data Type	MapXtreme 6.7.x Return Column Type / Value	MapXtreme 6.8.0 Return Column Type / Value
SQL Server/DateTime*	Date/System.DateTime	DateTime/System.DateTime
Informix/DateTime	Date/System.DateTime	DateTime/System.DateTime
Informix/Date	Date/System.DateTime	Date/MapInfo.Data.Date
Informix/Time	String/System.String	Time/MapInfo.Data.Time
Oracle/TimeStamp	Date/System.DateTime	DateTime/System.DateTime
Oracle/Date	Date/System.DateTime	Date/MapInfo.Data.Date

* SQL Server 2005 and earlier.

MapInfo SQL

The Time and DateTime types are not supported for MapInfo SQL functions. However, in MapInfo SQL functions that call/use a DateTime type, the function will return the date portion of the DateTime value. Please see the MapInfo SQL Reference for more information.

MapInfo Professional Tables

MapXtreme supports the Date, Time, and DateTime data types introduced in MapInfo Professional 9.0. See the MapXtreme Developer Reference for information on the new data types in the MapXtreme API.

Thematic Mapping and SQL Queries

Date, Time and DateTime fields can be used in a thematic map to shade the map according to time or datetime information in your table.

You can also use date and time information in SQL queries. Date, time, or datetime data can be a useful filter of your data. You can select data according to a specific period of time or that is within certain dates to obtain the desired segment of your data. You can then put the query results on a map and do further visual analysis such as create a theme or a buffer. See Chapter 11: Creating Expressions in the MapXtreme Developer Guide for more information.

Enhanced User-Defined Metadata Support

User-defined metadata is now supported for FeatureGeometry and Style column definitions. This metadata enables you to open a table as a spatial table and communicate with it in the absence of a MapCatalog. For example, you can explicitly identify FeatureGeometry and Style columns in result set schemas that are produced from TableInfoServer query definitions. Or you can use the metadata to a view table that you've created on the server when you do not want, or are unable, to use a MapCatalog.

The new ColumnHints property on the MapInfo.Data.TableInfoServer class enables you to specify the FeatureGeometry and Style column definition information that is used as hints to identify the columns. The new DefaultStyle property on the MapInfo.Data.GeometryColumn class supports the ColumnHints property in TableInfoServer. The DefaultStyle property enables a default style to be set for FeatureGeometry objects in the FeatureGeometry column. The DefaultStyle property explicitly provides the style information that would have to be

inferred for remote database tables from the MapCatalog. In the absence of any style information, the style specified in the DefaultStyle property is used to populate the Style column. The default style will also be used if there are any null values in the Style column.

Note: ColumnHints are not yet supported for defining TableInfoServer tables that will be modified through INSERT/UPDATE/DELETE commands. Database tables must be registered in the server's MapCatalog to support such modifications.

LanguagePreference Key/Value Pair

We now recognize a new key/value pair in the .config file that represents a language preference (when applied in the context of the Latin1 charset only). The key is named MapInfo.Engine.Session.LanguagePreference and is located within the <configuration><appSettings> section of the .config file. The associated value is a decimal locale identifier value to be given preference in the logic that establishes our default string normalizer - the class used for internal string comparison, ordering, and index utilization operations. The default LanguagePreference value is taken from the locale of the operating system.

A sample .config file might look like the following:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <appSettings>
    <add key="MapInfo.Engine.Session.LanguagePreference" value="1030" />
  </appSettings>
</configuration>
```

In this example, the value of "1030" corresponds to the hex value 0x0406 denoting Danish (Denmark), or locale name da-DK. As such, this will now allow the application to run in the Swedish (or Danish) language context even on an English operating system. Internal string comparisons, ordering and indexing will follow the Swedish/Danish language context.

Because this setting is contained in the application's .config file, the LanguagePreference value you set will only apply when running that application.

For a listing of locale identifiers, see [http://msdn2.microsoft.com/en-us/library/ms776260\(VS.85\).aspx](http://msdn2.microsoft.com/en-us/library/ms776260(VS.85).aspx).

Border Control Properties for Printing and Exporting

Two new properties in the MapInfo.Printing.ViewPrintDocument class, Border and BorderPen, enable you to change the border pen and turn the border on or off when you print maps or legends. These properties are also in the MapInfo.Mapping.MapExport and MapInfo.Mapping.LegendExport classes. In previous releases, it was not possible to remove the border within the MapInfo.Printing.MapPrinting class.

Contact Information Support on WMS Server

You can now add contact information such as contact name, organization and address to the WMS server configuration file and WMS capabilities xml when publishing a WMS server. A ContactInformation element has been added to the MapXtreme WMS schema (MXP_WMS_Configuration_1_1.xsd). In the schema, "ContactInformation" is optional, as it is in the OGC WMS Capabilities.

Bi-Directional String Support in Curved Labels

Curved labels are now supported in languages that read right to left. Support was added for the Uniscribe library that ships with Windows, which handles contextual glyph shaping and reordering bi-directional scripts such as Hebrew and Arabic. Its methods can analyze and break apart strings using complex scripts into separate runs which can then be reordered for display.

What's Changed

- **API Changes**
- **InfoTips**

API Changes

AutoScale Property Deprecated

The `AutoScale` property has been deprecated in the classes listed below. Please use the `AutoScaleMode` property instead.

- `MapInfo.Windows.BaseStandardDialog`
- `MapInfo.Windows.Dialogs.AreaStyleDlg`
- `MapInfo.Windows.Dialogs.FillStyleDlg`
- `MapInfo.Windows.Dialogs.LineStyleDlg`
- `MapInfo.Windows.Dialogs.SymbolStyleDlg`
- `MapInfo.Windows.Dialogs.TextStyleDlg`

InfoTips

The implementation of `InfoTips` has been changed so that they now display properly when the mouse hovers over a map feature. The new `InfoTipsDisplayDelay` property in the `MapInfo.Tools.MapTools` class provides additional control over when the `InfoTips` display.

The `MapInfo.Tools.MapTools` class now has three properties that control the appearance of `InfoTips`. They are:

- `InfoTipsEnabled`—Gets/Sets whether `InfoTips` will appear when the mouse is idle.
- `InfoTipTimerDelay`—Gets/Sets the amount of time (in milliseconds) that must elapse before the `InfoTip` appears. The default is 500.
- `InfoTipDisplayDelay`—Gets/Sets the amount of time (in milliseconds) that the `InfoTip` will be displayed. The default is 0, which means that the `InfoTip` will be displayed until the mouse is moved.

Note that as long as the mouse is moving, `InfoTips` will not appear. An `InfoTip` will only appear when the mouse is over a feature with a label and has been idle for at least the interval specified by `InfoTipTimerDelay`. The `InfoTip` will disappear when the mouse is moved or when the mouse has been idle for longer than the time specified by the `InfoTipDisplayDelay` provided that time is greater than zero.

You can prevent `InfoTips` from being displayed at all with:

```
mapControl1.Tools.InfoTipsEnabled = false;
```

You can set the delay before an `InfoTip` is displayed to one second with:

```
mapControl1.Tools.InfoTipTimerDelay = 1000;
```

You can set the length of time an `InfoTip` is displayed even though the mouse remains idle to 2.5 seconds with:

```
mapControl1.Tools.InfoTipDisplayDelay = 2500;
```

For more information on `InfoTips`, see the `MapInfo.Tools.MapTools` class in the *MapXtreme Developer Reference*. For more information on tools, see Chapter 8 in the *MapXtreme Developer Guide: Desktop Applications, Controls, Dialogs, and Tools*.

Upgrading to MapXtreme 2008

MapXtreme 2008 installs into its own directory using the form `\<installdir>\MapInfo\MapXtreme\6.x.x`, where X.x.x is the current release. It will not overwrite a previous version. This allows you to maintain side-by-side installations of different releases of the product.

A side-by-side installation requires that you copy your old license file to the location of the newest installation (Program Files\Common Files\MapInfo\MapXtreme\6.x.x.).

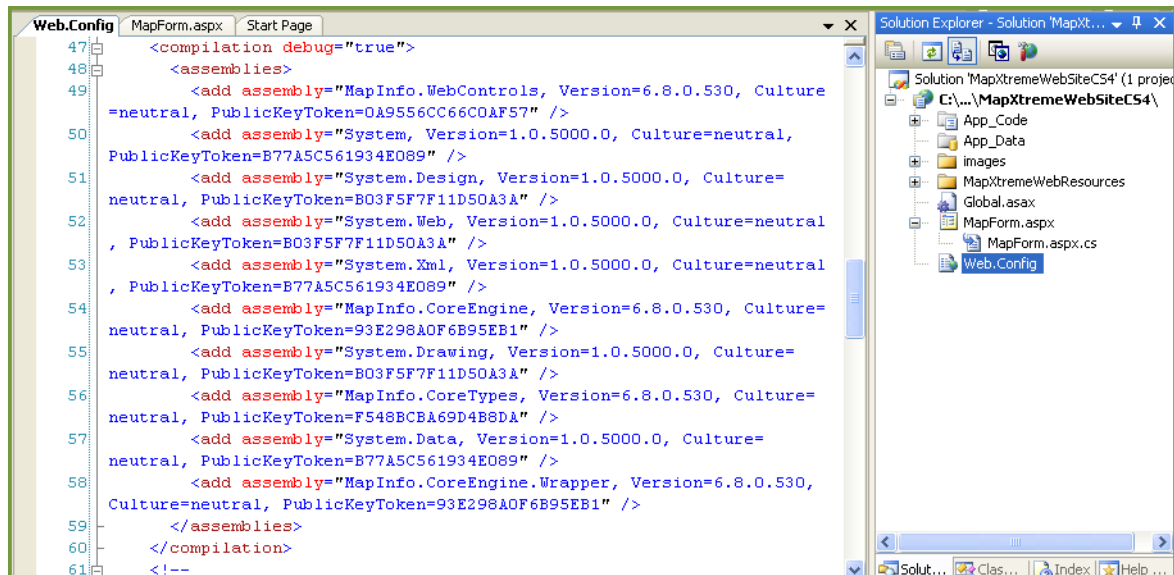
Note, when using a side by side install, you must close Visual Studio before opening a project of a different type. For web applications, you will also need to reset Internet Information Services (IIS). Use the **iisreset.exe** command in a console window or from the Start > Run menu option.

Updating Existing Web Sites

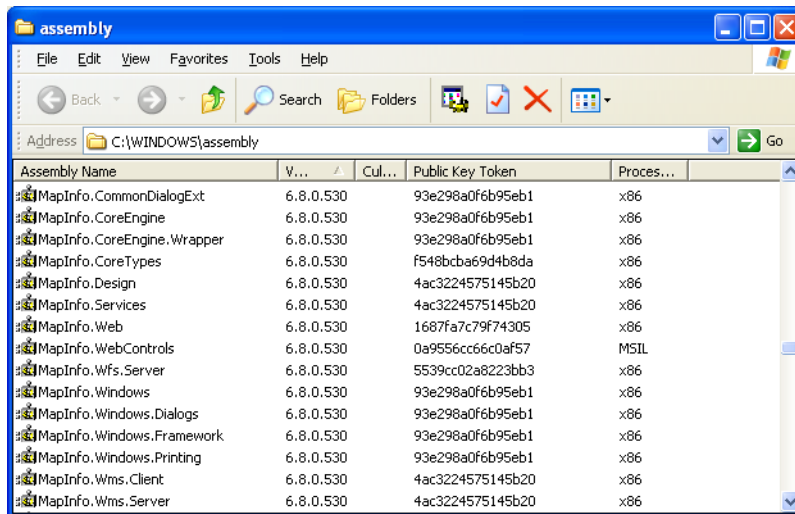
In order to run a web site created with a previous release of MapXtreme, you must edit your application's Web.config file to point to the new versions of the assemblies. The example below shows the four MapXtreme assemblies that must be included:

- MapInfo.WebControls
- MapInfo.CoreEngine
- MapInfo.CoreTypes
- MapInfo.CoreEngine.Wrapper

MapInfo.CoreEngine.Wrapper is new beginning with the MapXtreme 2008 v6.8.0 release. Any web application created with MapXtreme v6.7.1 or earlier, must include this assembly to run against MapXtreme 2008 v6.8.0.



Following the illustration above, edit your Web.config file to include the full version and build number and PublicKey token of the MapXtreme assemblies. To determine version number and PublicKey token, open the Run dialog from the Start menu and type `assembly`. The Global Assembly Cache (GAC) lists all assemblies installed on your system. The version and build number for MapXtreme 2008 v6.8.0 is 6.8.0.536.



You can also reach the Assembly window by navigating from Start > All Programs > Control Panel > Administrative Tools > .NET Framework 2.0 Configuration and choose Manage the Assembly Cache.

When you are finished editing, save your Web.config file and rebuild your web site.

Updating Existing Desktop Applications

Desktop applications created with a previous release of MapXtreme can be recompiled to work in the current release. There are several things you need to do first.

- Set Copy Local property to false
- Add new v6.8.0 assemblies to project
- Redirect assemblies to the v6.8.0 assemblies

Set Copy Local property to False

Verify that the Copy Local property for your web controls is set to False. There exists a known issue that when you drag and drop a MapXtreme desktop control onto a Windows form, references are loaded that have the Copy Local property set to True. MapXtreme references must point to assemblies residing in the Global Assembly Cache (GAC), not to the local bin path, which is what happens when Copy Local is set to True. For more on this issue, see [Copy Local Property](#).

Add New Assemblies to Project

For applications built using MapXtreme v6.7.1 or earlier, you must also add two assemblies: MapInfo.CoreEngine.Wrapper and MapInfo.CommonDialogExt. For more information, see [New Assemblies](#).

Redirecting MapXtreme Assemblies To Newer Versions

Microsoft provides several mechanisms to redirect assemblies. Review the information below. For more detailed information, see Microsoft's [.NET Framework Developer Center](#).

Application Configuration File

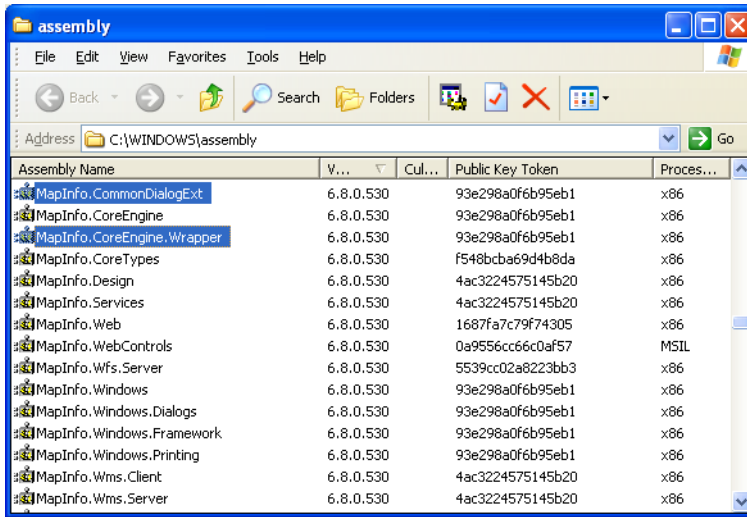
It is recommended that you use an application configuration file to accomplish assembly redirection. The configuration file must be located in the same directory as the application and is named after the application. For example, the configuration file for *myApp.exe* must be named *myApp.exe.config*.

The application configuration file overrides settings in the publisher's policy file.

To redirect assemblies, you must identify the version numbers and PublicKey tokens for the current release and add them to your application configuration file. The version number is in the form X.x.x.x, for example, 6.8.0.536.

To find the information, open the Run dialog from the Start menu and type `assembly`. The Global Assembly Cache (GAC) lists all assemblies installed on your system. You can also reach the Assembly window by navigating from Start > All Programs > Control Panel > Administrative Tools > .NET Framework 2.0 Configuration and choose Manage the Assembly Cache. This tool will also create an application configuration file, if one does not exist.

The illustration below highlights the two assemblies that must be added to desktop applications created with MapXtreme v6.7.1 or earlier.



Publisher Policy File

A publisher policy file containing redirection settings could be installed in the GAC with the assembly. However, this is not a supported MapInfo configuration.

Machine Configuration File

The Microsoft .NET Framework 2.0 Configuration tool can be used to update the machine configuration file, which is located in `\Microsoft.NET\Framework\v2.0.50727\Config` under your Windows installation directory.

Specifying redirection settings here will cause *all* applications referencing the assembly to use the upgraded version. Use this method of redirection carefully, since the machine configuration file overrides settings in both the application configuration file *and* the publisher's policy file.

Bug Fixes

The following topics were addressed during the development of this release of MapXtreme 2008:

- [Add Text Tool](#)
- [Affined or Rotated Images](#)
- [Anti-Aliasing](#)
- [Curved Labels](#)
- [Group Layers](#)
- [InfoTips](#)
- [Label Priority](#)
- [Layer Control](#)
- [MapInfo SQL](#)
- [Printing Maps](#)
- [StreetPro Sample Data](#)
- [Tables](#)
- [Translucency](#)
- [Web Designer](#)
- [WMS Client](#)
- [WMS Server](#)

Add Text Tool

When you use the Add Text tool to enter text on a layer, using the “y” key no longer cancels the tool. In previous releases, pressing the “y” key as you typed your text cancelled the activation of the tool.

Affined or Rotated Images

MapXtreme now displays and reprojects raster images that are affined or rotated in British National Grid (BNG) correctly. In previous releases, raster images that were in the Long/Lat coordinate system and had an affine transformation lost the affine transformation information when the image was converted to a different coordinate system using metadata keys.

Anti-Aliasing

Workspace Manager no longer crashes when anti-aliasing is turned on in data that contains single node polylines.

Curved Labels

The option that activates curved labels in the Position tab of Workspace Manager has been renamed to better identify what the option does. In previous releases, the option was named Parallel to Multiple Segments. It is now called Curve labels along segments.

Group Layers

Reordering layers from a group layer to outside the group layer now works correctly. Revisions to the MapInfo.Mapping.LayersBase.Move methods corrects the problem of layers in group layers reordering unpredictably when they are moved.

Group layers now work correctly with the All Layers option in the View Entire Layer dialog box in the Workspace Manager. In previous releases, choosing the All Layers option when there was a group layer present would produce an exception.

InfoTips

Changes in the MapInfo.Tools.MapTools class corrects the problem of InfoTips not displaying when the mouse is hovering over a map feature. The following list summarizes how InfoTips behave:

- When the mouse hovers over a feature with a label, after a period specified by the InfoTipTimerDelay property, the label will appear in an InfoTip instance. After a period specified by the InfoTipDisplayDelay property, the InfoTip instance will be hidden.
- When the mouse subsequently moves, the InfoTip instance will be hidden even if the InfoTipDisplayDelay time interval has not expired.
- If the InfoTipDisplayDelay time is zero, the InfoTip instance will remain visible until the next MouseMove event.
- The InfoTip will be hidden when the mouse moves over a feature with no label or is not over a feature or moves off the map control.

The InfoTipDisplayDelay property is new with this release. For more information, see [InfoTips](#).

Label Priority

In Workspace Manager, label priority settings on an .MWS workspace that was created in MapInfo Professional are now saved. In previous releases, Per-Label Priority Expression settings specified in the Workspace Manager Rules tab were not saved in workspaces that were created in MapInfo Professional.

Layer Control

At certain <appsettings> in the web.config file, the LayerControl crashed when you added a layer from a temporary table to a map. This problem has been fixed. In previous releases, this occurred when the <appsettings> in the web.config file for the application were set as follows:

```
<add key="MapInfo.Engine.Session.Pooled" value="false" />
<add key="MapInfo.Engine.Session.State" value="HttpSessionState" />
<sessionState mode="StateServer" stateConnectionString="tcpip=xxx.x.x.x:xxxx"
sqlConnectionString="data source=xxx.x.x.x;userid=sa;password=" cookieless="false"
timeout="20" />
```

Note that these settings are not the recommended settings for MapXtreme 2005. The project templates and samples use pooling, manual Session.State and InProc sessionState mode. This gives you more control over what is saved for each HTTP session. See Chapters 5 and 6 in the MapXtreme 2005 Developer Guide for information on best practices for web applications and managing state.

MapInfo SQL

The MI_EngineVersion function was reporting the major version number as "1". The function now correctly reports "6" as the major version number.

Printing Maps

Two new properties in the MapInfo.Printing.ViewPrintDocument class, Border and BorderPen, enable you to change the border pen and turn the border on or off when you print maps or legends. These properties are also in the MapInfo.Mapping.MapExport and MapInfo.Mapping.LegendExport classes. In previous releases, it was not possible to remove the border within the MapInfo.Printing.MapPrinting class.

StreetPro Sample Data

The StreetPro sample data now displays in Workspace Manager with the correct symbols. In previous releases, the fonts needed to create the symbols were missing.

Tables

CacheStorageType

The CacheSettings.StorageType property is now updated properly when an internal override to the cache storage type occurs. In previous releases, the CacheSettings.StorageType property was not updated in certain cases. Specifically, when CacheSettings were enabled as part of the TableInfo definition, and Native was specified as the CacheStorageType on the CacheParameters, if any errors occurred during the creation of the Native cache file, an attempt was made to create the cache using MemTable storage instead. For large tables, this override created a high likelihood of much more memory being used than anticipated. In addition, when this fallback to MemTable storage occurred, the CacheStorageType property of the CacheParameters was not being updated to reflect the change.

TableInfo.CreateFromFile Method

The TableInfo.CreateFromFile method now associates indices to the correct fields/columns. In earlier releases, the index to field/column association was incorrect in certain cases.

TableInfoServer.KeyType Property

Setting the TableInfoServer.KeyType property to Implicit now works correctly when the TableInfoServer.KeyColumns property is set. Previously, the KeyType property was internally set to Explicit after the TableInfoServer.KeyColumns property was set.

Translucency

Translucency no longer causes colors to invert in objects with holes at low zoom levels. In earlier releases, zooming in on objects with holes with translucency enabled caused the colors of the object and the hole to invert, so that the hole took on the color of the object and the object took on the color of the hole (white). This issue also occurred when antialiasing (GDI Plus) was enabled, and with printing, regardless of whether translucency or antialiasing were enabled. These cases have also been fixed.

When translucency and anti-aliasing are enabled in MapXtreme, transparency settings on raster images now work correctly. In previous releases, it was not possible to use raster transparency settings with translucency and anti-aliasing because activating translucency and anti-aliasing would override the raster transparency settings. If translucency and anti-aliasing were not enabled, raster transparency settings worked properly.

Web Designer

References to the WebControls assembly are now added to the web.config file with the other required MapXtreme assemblies. In previous releases, web applications could not be compiled because the web.config file did not reference WebControls.

WMS Client

The problem of WMS client Get Capabilities requests returning false "Invalid XML Format" errors in the MapInfo.Wms.ClientUtilities.GetCapabilities method has been corrected. To fix the problem, the XmlTextReader.UrlResolver has been set to null and the XmlUrlResolver has been removed. The XmlTextReader performs an XML well-formedness check with the DTD, but it does not validate the DTD. Because the WMS capabilities parser checks the XML elements in the proper order based on the WMS <version>spec/DTD, the initial DTD well-formedness check performed by the XmlTextReader is not necessary.

WMS Server

The problem of the WMS CapabilitiesWriter outputting SRS tags only in WMS v 1.1.1 format has been corrected. This issue occurred even when v. 1.1.0 was specified in the web.config file and the appropriate v. 1.1.0 DTD was used. This created XML that would not validate against the v. 1.1.0 DTD. To correct the problem, references to dtdVersion in both the CapabilitiesWriter and ExceptionWriter have been removed. XML now validates against the v. 1.1.0 DTD.

Known Issues

The issues that follow are those identified since the previous release of this product. They are either currently under investigation or in active development seeking a resolution.

- ◆ [64-bit Systems and Platform Target x86](#)
- ◆ [Switching Between .NET Framework 1.1 and 2.0 on 64-bit Windows Systems](#)
- ◆ [ADO.NET Serialization](#)
- ◆ [Data Access](#)
- ◆ [Enhanced Rendering](#)
- ◆ [Graticules](#)
- ◆ [HTML and CSS Validation Errors in Visual Studio](#)
- ◆ [LayerControl](#)
- ◆ [Labels](#)
- ◆ [Legends](#)
- ◆ [Licensing](#)
- ◆ [Mapping](#)
- ◆ [Persistence](#)
- ◆ [Printing](#)
- ◆ [Raster Images](#)
- ◆ [Sample Applications](#)
- ◆ [Serialization](#)
- ◆ [Styles](#)
- ◆ [Tables](#)
- ◆ [Tools](#)
- ◆ [Visual Studio](#)
- ◆ [Web Applications](#)
- ◆ [Web Controls](#)
- ◆ [WMS/WFS](#)
- ◆ [Workspace Manager](#)
- ◆ [Miscellaneous Issues](#)

64-bit Systems and Platform Target x86

On a 64-bit machine, you may get a BadImageFormatException when trying to run a MapXtreme 2008 desktop application. Since MapXtreme runs in 32-bit emulation mode on 64-bit systems (WOW64), you must assign the application you are building in Visual Studio to use the Platform Target x86. In the project properties, click on the Build tab and choose x86 for the Platform Target.

Switching Between .NET Framework 1.1 and 2.0 on 64-bit Windows Systems

MapXtreme 2008 is a 32-bit application that runs on 64-bit systems using Microsoft's WOW64 technology. It is built on top of the .NET Framework 2.0. MapXtreme Web Sites or ASP.NET Web Sites or Applications run under ASP.NET 2.0.

If you have a side-by-side installation of MapXtreme that must run under the .NET Framework 1.1 (v6.6 and earlier), follow the instructions below to switch from one framework to the other. See <http://support.microsoft.com/kb/894435> and <http://support.microsoft.com/kb/911720> for more information.

Using MapXtreme (v6.6 or earlier) on the .NET Framework 1.1 on a 64-bit System

1. Execute the following script to enable 32-bit mode:

```
cscript %SYSTEMDRIVE%\inetpub\adminscripts\adsutil.vbs SET W3SVC/AppPools/Enable32bitAppOnWin64 1
```

2. Execute the following command to install the version of ASP.NET 1.1 and to install the script maps at the IIS root and under:

```
%SYSTEMROOT%\Microsoft.NET\Framework\v1.1.4322\aspnet_regiis.exe -i
```

3. Under Web service extensions in IIS Manager, ensure ASP.NET version 1.1.4322 is set to Allowed.

Using MapXtreme (v6.7 or later) on the .NET Framework v2.0 on 64-bit System

1. Execute the following script to enable 32-bit mode:

```
cscript %SYSTEMDRIVE%\inetpub\adminscripts\adsutil.vbs SET W3SVC/AppPools/Enable32bitAppOnWin64 1
```

2. Execute the following command to install the version of ASP.NET 2.0 (32-bit) and to install the script maps at the IIS root and under:

```
%SYSTEMROOT%\Microsoft.NET\Framework\v2.0.50727\aspnet_regiis.exe -i
```

3. Under Web service extensions in IIS Manager, ensure ASP.NET version 2.0.50727 (32-bit) is set to Allowed.

ADO.NET Serialization

Because of situations in which we cannot reliably re-establish references to a shared DataTable instance (resulting in disjoint copies), the automatic serialization of TableInfoAdoNet-based tables is currently unsupported. Chapter 10 in the MapXtreme 2008 Developer Guide outlines a manual procedure that can be used until this capability can be properly and reliably supported. Please check here in the Release Notes for the latest information and updates pertaining to that feature.

Data Access

Update Query for MS Access

Executing an update query on a MS Access table fails to insert style and object. For example:

```
insert into custpointswest (Obj, MI_Style) values ( newobject, newstyle)
```

However, if you only insert the Obj field, the query works, as below:

```
insert into custpointswest (Obj) values ( MI_Point  
(-121.331658, 38.546608, 'EPSG:4326'))
```

MS Access Failed Connections

MapXtreme Web applications that access Microsoft Access databases may experience failed connection attempts. This is due to a limitation in Microsoft's Jet database engine when used in a high-stress, 24 x7 server environment. The Jet database engine, the engine behind MS Access, is limited to 64 concurrent connections. This includes the number of tables across all threads. In high-stress web environments, the actual number of successful concurrent connections may be less. For more information, see the following Microsoft technical publication:

<http://support.microsoft.com/default.aspx?scid=kb;EN-US;q222135>.

Microsoft recommends using MS SQL Server with IIS for web applications that require absolute data integrity or high user concurrency.

Catalog.Search Leaks

Catalog.Search using SearchResultProcessor leaks ResultSet tables in the Catalog (that is, extra ResultSetTables remain in the Catalog). The user also does not have the ability to determine which ResultSet table was created.

Error Creating Index During SQL Query

The composition of values from the fields referenced in a GROUP BY clause are used to create key values for a temporary table created internally during statement execution. An index is created for the key value column in the temporary table, and this index is currently limited to values with a maximum length of 254. To avoid index creation errors, please ensure that composite keys will not exceed this maximum length (that is, try to avoid long string columns in the GROUP BY clause).

Update Command

MapXtreme cannot perform a subselect or specify two tables in an update command. To work around this issue, you must iterate through one table and specify one feature at a time to do a query based on another table.

Windows Authentication and TableInfoServer

To connect to a SQL Server database via TableInfoServer using Windows Authentication, you must explicitly set the Toolkit to ODBC:

```
ti.Toolkit = MapInfo.Data.ServerToolKit.Odbc
```

You can not use an OLEDB based connection string directly.

Enhanced Rendering

Translucency Slider Controls

The translucency slider controls in the Style dialogs have no affect unless enhanced rendering (translucency) is enabled. When enhanced rendering is disabled, the translucency slider controls in the styles dialogs appear to be enabled, yet have no effect.

To enable translucency, in Workspace Manager, check the Enable Translucency under the Style tab. Programmatically, set the MapInfo.MapInfo.DrawingAttributes.EnableTranslucency property.

Panning

Using anti-aliasing with translucency will distort labeling when panning.

Range Theme and Dot Density Color Picker Dialogs

The color picker dialog used by ranged themes and dot density themes does not have support for translucency. This means that, when using this dialog, you cannot choose a translucent color value for the inflection point of a ranged theme nor a translucent dot color for a dot density theme. You can do it programmatically.

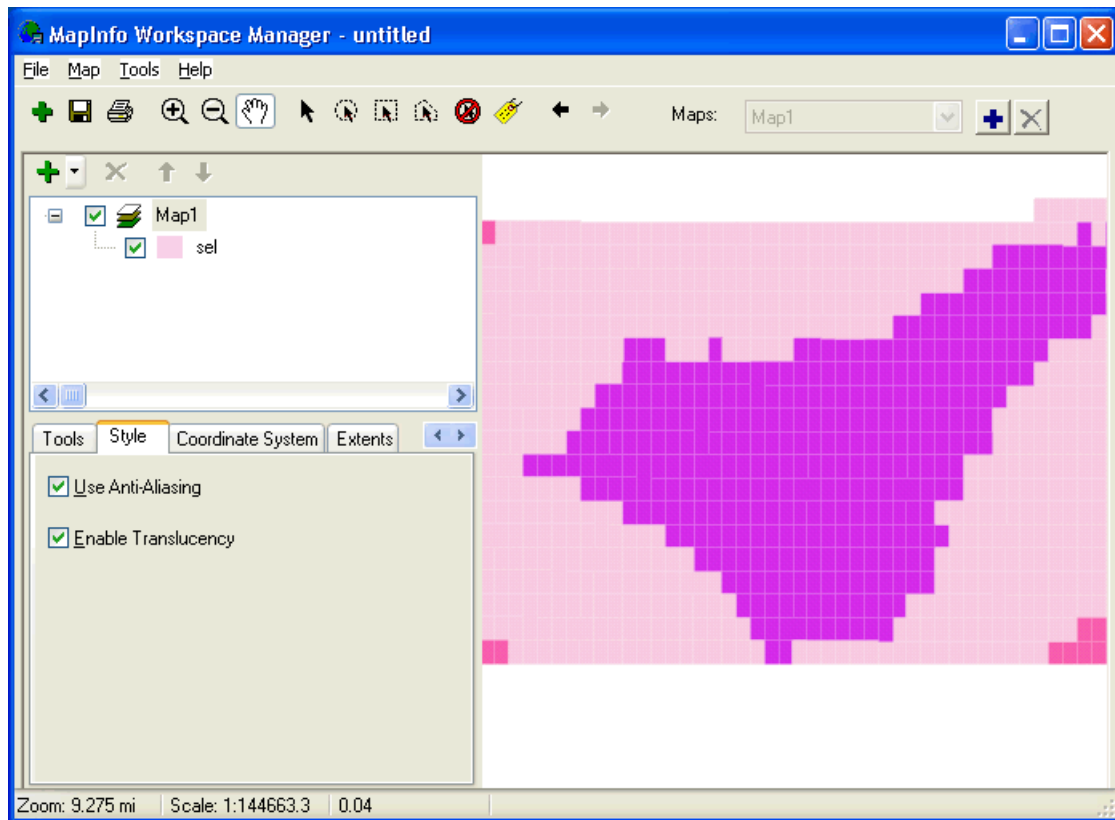
Note: The other MapXtreme 2008 style dialogs, such as LineStyle, AreaStyle, TextStyle and SymbolStyle, use a different color picker dialog that does support translucency.

Remote Desktop Issue

The quality of images in Enhanced Rendering mode is sometimes compromised when using Remote Desktop to run MapXtreme 2008. The white portions of the map may draw with a grey tint when you pan the map.

White Borders Caused by Anti-Aliasing

Turning on anti-aliasing causes layers with straight edges to be drawn with white lines. If you have a table that contains a series of rectangular features with the same boundaries and anti-aliasing is enabled, you see a light border around each feature, as pictured below:



In these instances you either should disable anti-aliasing or select borders with the same color as the feature.

Graticules

If a graticule is the first item inserted into a workspace, nothing appears and subsequent layers cannot be seen. To view the map and the graticule's grid lines, you must right-click the root node that corresponds to the map and select View Entire Map.

HTML and CSS Validation Errors in Visual Studio

When developing a Web Site in Visual Studio with the MapXtreme template, you may receive some erroneous HTML validation errors or warnings. These errors occur because Visual Studio is validating the HTML internally against the XHTML 1.0 Transitional standard. Visual Studio does not use the HTML 4.01 (loose) DOCTYPE, which is the defined target for web pages created from the MapXtreme template. However, simply setting this target to HTML 4.01 is still not enough to ensure HTML validation.

To ensure that your HTML validates, set the validation target in Visual Studio 2005/2008 to Internet Explorer 6.0. This setting validates against HTML 4.01 (loose) and is more compatible with the MapXtreme Web application. The validation target can be changed using the Tools > Options > Text Editor > HTML > Validation menu. This setting can also be changed on the HTML Source Editing toolbar (View>Toolbars>HTML Source Editing).

If you encounter CSS validation errors, check the validation level using the Visual Studio Style Sheet toolbar (View > Toolbars > Style Sheet). Set the level to CSS 2.0 to match the style sheet used in the MapXtreme template.

LayerControl

Edit Handles

Edit handles aren't displayed or hidden properly when using LayerHelper class to make a layer editable.

Edit handles also do not display properly when you select features in a layer, and then set the layer to be editable using the desktop LayerControl. The problem goes away the first time you redraw the map. Edit handles behave correctly as you select and deselect features.

Select Tool

The Select tool does not behave correctly in LayerControl when Allow Node Editing is turned on.

Labels

The rotation handles that display for labels and legacy text that are selected do not operate. Additionally, the anchor point position cannot be changed.

Label visible range is not inclusive at MaxZoom in some situations. The visible zoom level varies due to the coordinate system, possibly due to rounding calculations.

When label names are very long and a label priority is used to eliminate overlap, the resulting distribution of labels is too sparse. There are large areas of the map where labels could be drawn without overlapping, but did not.

Custom labels cause large memory usage when cloning a map.

Curved Labels

When curved labels is on the label tool does not draw the labels curved. This happens when the street layer visibility is turned off.

Curved labels with the default setting of not allowing overlapped labels results in too many curved labels not displaying.

Zooming far in on a curved label will make it cross the street segment.

Clicking and dragging a curved label does not work as expected.

Legends

BarTheme Legend

The BarTheme legend does not update to show scale after the map zoom level is changed. In place of the scale, a "Not drawn to scale" message displays.

Legend Row Visibility

The LegendFrame.Rows.Visible property, when set to 'false', does not turn off the legend row as expected.

Duplicate Legends

Re-opening a workspace with a theme and saving it creates two legends in the workspace.

Appended Custom Legends

Saving a map and an appended custom legend to a workspace results in an error.

Licensing

The licensing in MapXtreme 2008 is unable to determine if an application is running as a desktop or web application when deployed as a Web Service. For web applications use a desktop runtime license along with your web runtime license.

Mapping

Exporting to EMF

Exporting to EMF and viewing the EMF file in Microsoft Word displays more of the map than was originally exported.

Panning

When panning a map that uses a scale range for visibility, white patches display over the map,

Map Center Point Incorrect After Pan and Zoom

ASP.NET session state management problems can occur when using StateServer and SqlServer under .NET 2.0 framework. When performing a zoom or pan operation on the map, the new center point of the map image is offset incorrectly. This problem was encountered only in Visual Studio 2005 using the 2.0 Framework.

Exporting to a Higher Resolution

Exporting a map with labels to a higher resolution increases the size of the labels.

Opening Many Windows

Opening many map windows (40 or more) causes the map to draw incorrectly.

Performance

AddColumns

MapXtreme 2008's performance can be very slow when you are adding columns to tables with more than 200,000 records. We recommend that you create the columns collection yourself (instead of passing in NULL). Include the geometry column only if you need it. Do not index the temporary columns unless you need to.

Inserting records into a TableInfoServer table using MICommand takes a long time for a large number of records.

Zoom

Zooming in and out on layers is much faster in MapX 5.x than it is in MapXtreme 2008.

Persistence

The MapInfo.Persistence.WorkspacePersistence.Save method produces an error when attempting to save a view table to a workspace that is defined using MapInfo SQL aggregations. The error indicates that a key that already exists is being added to the dictionary. This issue is an internal problem caused by how MapXtreme handles tables with dependencies, i.e., view tables and ResultSet tables, and tables that have temporary columns, such as an aggregation column.

Printing

Adornment Positioning

Adornments are not positioned correctly when printing.

When changing the map to print to full page and you have a title adornment, it doesn't seem that you can get to the size of the map on the page in order to change the location of the title adornment.

Translucent Rasters

There is a limitation with printing translucent rasters. EnableTranslucency must be enabled to print translucent rasters; however, this is not required for screen display or exporting.

Print Preview Unhandled Exception

Print Preview throws an unhandled exception in Workspace Manager with HP LaserJet printers and 2.0 Framework.

Note: The Print Preview exception has a Continue and Quit button. If you click the Quit button, Workspace Manager closes. If you click Continue, Workspace Manager opens successfully. However, you will not be able to print successfully to this printer until you restart Workspace Manager.

Raster Images

Raster Reprojection Fails at 180 degrees

Raster reprojection is not working properly when the rotation angle is 180.0 or -180.0 degrees. It works when the rotation angle is ± 180.00001 or greater.

ESRI Grid and ERDAS IMAGINE Files

MapXtreme 2008 is unable to open ESRI Grid or ERDAS IMAGINE (.IMG) raster files in Workspace Manager.

Distortion in Side-by-Side Images

When attempting to generate seamless side-by-side images of a map, a slight distortion in the resulting images occurs if a raster image is added, even when the map's projection matches that of the raster. A side effect of adding a raster image is that it always determines the current coordsys to use. It also affects where the standard parallel is set which slightly skews the display. To eliminate the distortion, simply regenerate the second image.

Sample Applications

World.MWS

In the HelloWorldWeb sample application, the ocean layer in the world.mws map is corrupt (many dots appear in the ocean). This happens when you have an adornment attached to map.

Serialization

You cannot deserialize an ASCII PointRef table if the reference table is closed. You must first deserialize the pointRef table and then close the reference table.

When using ranged themes, do not clear the theme before deserializing it. This will result in differences in the theme properties between the serialized and deserialized theme.

Styles

The black halo for a Text style is not displaying correctly. This is observable in the TextStyleDlg and StyleSampleBitmapButton object (button on the Style tab). As a workaround, choose a very dark gray as the halo instead of black.

The Symbol style for a Graduated Symbol theme is not properly reflected in the Layer Control. Graduated Symbols appear correctly on the map but are represented by a different symbol in the Layer Control.

Disposing CompositeStyles

CompositeStyles do not contain a way to free the resources used, causing memory leaks under some conditions. In the following code example, to avoid a memory leak, set the CompositeStyle to null.

```
private MapInfo.Styles.AreaStyle _style = new MapInfo.Styles.AreaStyle();
for (int i = 0; i < 1000000; i++)
{
    MapInfo.Styles.CompositeStyle cs = new MapInfo.Styles.CompositeStyle(_style);
}
cs.AreaStyle=null;
```

Tables

WriteTabFile()

TableInfoView.WriteTabFile() doesn't write out the CharSet property, resulting in the default value (Latin1) being used.

Alias

You cannot change the alias of a FeatureCollection returned from a Search. The exception string and online documentation does not include this information.

SpatialSchemaXY

Update operations on a MapXtreme Catalog for a Microsoft Access database fail when a SpatialSchema object is used.

Vertical Mapper GRD Files

MapXtreme 2008 locks a Vertical Mapper GRD file during use, preventing further operations on the file outside of MapXtreme 2008. For example, the file cannot be edited with the VM SDK method vmWriteGRDRow nor can it be deleted from a user's system through Windows Explorer.

ASCII and Other File-Based Table Formats

MapXtreme is unable to open ASCII files that have non-TXT file extensions. The DataPath property on TableInfoAscii cannot reliably determine where the file is located when the extension is other than .TXT.

MapXtreme is unable to open supported file-based formats when the related .TAB file is located in a different directory from the base file.

Tools

Rectangle Select Tool

The RectangleSelect tool in MapXtreme 2008 may draw slower than expected when the map contains Oracle Spatial layers. To avoid this issue, clear the Show InfoTips check box for the layer(s) in Workspace Manager.

Firefox and Netscape

Tools do not work properly in Firefox or Netscape when creating a new web application from the ASP.NET web application template in Visual Studio 2005 and Visual Studio 2008.

Visual Studio

Web Site Projects and Web Application Projects

In Visual Studio 2005, Microsoft introduced a web project development model that no longer required the use of a project file for containing web application files. This project type, called a Web Site, is created from a template located under File > New > Web Site menu. The previous web model, called a Web Application, was removed from the product. Its project-based template was accessible under the File > New > Project menu.

In Visual Studio 2005 Service Pack 1, Microsoft re-introduced the Web Application development model. In Visual Studio 2005 SP1 and Visual Studio 2008, you now have a choice of web development models: Web Site, under the New Web Site menu, and Web Application, under the New Project menu.

This release of MapXtreme 2008 does not provide a template that uses the re-introduced Web Application model. To build a web project using a MapXtreme template, you must use the Web Site development model under the New Web Site menu. All web sample applications included in this release were built as Web Sites.

For Microsoft-based ASP.NET web projects, you can build either a Web Site or Web Application.

Web Setup Project

If you create your own deployment for a MapXtreme Web Site project in Visual Studio 2005 or Visual Studio 2008, you will need to add the MSMs manually. The web setup project does not detect the assemblies that are referenced within Web.config; thus the assemblies (and the corresponding MSMs) are not detected as dependencies.

To add the MSMs manually, right-click on your web setup project and choose Add: Merge Module. Select MapInfoCoreEngine_6.x, MapInfoMXTConfig_6.x, MapInfoWeb_6.x and any other MSMs you need.

MapForm Designer

The MapForm designer in Visual Studio 2005 and Visual Studio 2008 throws errors when you load a map into a MapControl, then close the map before rebuilding the application. This happens regardless of the type of template (MapXtreme or Windows) or programming language (C# or VB) you use. As workaround, be sure to rebuild your application after loading a map.

Desktop Application Deployment

An issue exists when you are building your own installer for a desktop application, Visual Studio does not include the following dependencies and their policies:

- Microsoft_VC80_CRT_x86.msm, policy_8_0_Microsoft_VC80_CRT_x86.msm
- Microsoft_VC80_DebugCRT_x86.msm, policy_8_0_Microsoft_VC80_DebugCRT_x86.msm

VisualStudio 2005 Service Pack 1 and Visual Studio 2008 use an updated C runtime library. The custom installer might not work if deployed to a machine without the updated C runtime library.

Web Applications

MapXtreme Assembly References and Microsoft's ASP.NET Web Application Template

This issue affects users of MapXtreme 2008 (v6.8.0) who have side-by-side installations of MapXtreme integrated with Visual Studio 2005 SP1 or Visual Studio 2008 or both.

When using the ASP.NET Web Application template (File > New > Project) and you drag and drop MapXtreme web controls onto your form, the Visual Studio designer automatically adds references to the MapXtreme assemblies. These references are added to several files: the web project file (.csproj or .vbproj), the web application web configuration file (Web.config), and the web page itself (.aspx). It is possible that one or more of these assembly references has an incorrect assembly version number.

For example, you have two versions of MapXtreme installed on your system (v6.7.1 and v6.8.0) and you add a MapXtreme LayerControl from v6.8.0 onto your ASP.NET form. Your web project file may reference the v6.7 version of the LayerControl while the Web.config and .aspx files reference the v6.8.0 LayerControl. This will throw errors when you debug your application. There has been no established pattern for this behavior.

This does not occur when you are using the MapXtreme Web Site template (located under File > New > Web Site).

To resolve this assembly mismatch, you must manually correct the version numbers in each file to match the version of the web control you used. To determine the full version number for each installation of MapXtreme, view the assemblies in the Global Assembly Cache (GAC). Assemblies for v6.7.1 display version number 6.7.1.503. Assemblies for v6.8.0 display 6.8.0.

Copy Local Property

Dragging and dropping a MapXtreme desktop control onto a Windows Form loads references that have the Copy Local property Set to TRUE. This behavior is exhibited in Visual Studio 2005 with SP1. MapXtreme does not run assemblies from the local bin path; it runs assemblies from the Global Assembly Cache (GAC). The workaround for this issue is to set the Copy Local property to FALSE.

The Copy Local property determines whether a reference is copied to the local bin path. At run time, a reference must exist in either the Global Assembly Cache (GAC) or the output path of the project. If this property is set to true, the reference is copied to the output path of the project at run time. MapXtreme's assemblies reside in the GAC, so any reference to an assembly must point to the GAC. This means that your project's Copy Local property should be set to FALSE.

The same behavior exists for ASP.NET Web applications when dragging and dropping MapXtreme web controls onto a form. The same workaround applies here: Set the Copy Local property to FALSE.

This does not occur when using the MapXtreme Web Site template or sample applications, which are based on the MapXtreme Web Site template. Web site projects have all the assembly references contained in the Web.config file. There is no notion of a Copy Local property in the Web.config. For more information on the difference between these two types of projects, see [Web Site Projects and Web Application Projects](#).

Web Template Web.config for Non-English Operating Systems

On some non-English operating systems, the web.config file that is generated by the MapXtreme 2008 Web Site template does not contain the correct path to the world.mws file. The correct path is C:\Programmer\MapInfo\MapXtreme\6.x.x\Samples\Data\World.mws.

This issue is a factor in the web sample applications that ship with MapXtreme 2008. See the Readme.rtf that accompanies each web sample application. Look under the section entitled "Installation and Configuration Issue."

Browser Back Button and pre-v6.5 Web Controls

In a web application, if you zoom in on one area of a map, click the Back button, and zoom in on another area of the map, the previous zoom view displays. The Forward and Back browser buttons are disabled. This applies to the pre-v6.5 web controls only.

Pre-v6.5 Web InfoTool

The pre-v6.5 web InfoTool selects features incorrectly and inconsistently. The expected behavior is that the tool selects objects that intersect the select point within a certain pixel tolerance. Currently it selects objects that have a minimum bounding rectangle (MBR) that enclose the clicked point. These MBRs may have larger or smaller tolerances. This behavior does not exist in the JavaScript-enabled Web MapControl.

Web Controls

MapAlias Is Required at Design Time

You must set a MapAlias value every time you add a MapControl to a WebForm. If the MapAlias is not set at design time, your application will not run properly, resulting in a crash. Further, if you change the MapAlias value at runtime, an error will result from the application looking for the original MapAlias.

The MapXtreme 2008 web template and samples provide a default MapAlias called "Map1".

MapAlias and Session Management

Changing to a different MapAlias for a MapControl at runtime restores incorrect session values. The default session management settings in MapXtreme 2008's project templates and sample applications only store changes to the HTTP session, rather than the entire session. As stated in the above known issue, the MapAlias is the key to tying the changes to the correct session. If the MapAlias is changed at runtime, the session will return state that was for the previous MapAlias.

PointSelectionTool

The web PointSelectionTool cursor hotspot is not at the expected location (top of finger). This makes it difficult to click precisely on an object.

WMS/WFS

Workspace Manager errors when you select Map > View Entire Layer for a WMS Table version 750.

Workspace Manager

Changed values are not included when a workspace is saved. For example, if you change one of the zoom values on the Visibility tab then immediately click Save, the change is not preserved upon exiting Workspace Manager and reopening the workspace. However, if you click away from the control containing the changed value (that is, lose focus) prior to clicking Save, the changed value is saved.

Miscellaneous Issues

MessageBox

Putting a message box in a window (form) constructor prevents marquee zoom in and zoom out tools from working. To avoid this situation, put the MessageBox call in the form's Load event handler method.

Documentation Topics

The following items are topics that are unclearly documented or missing from the MapXtreme Documentation set (Developer Guide and Developer Reference).

F1 Help for C# Applications

The Visual Studio F1 Help button yields an “Information not found” topic when you highlight the entire line of a MapXtreme member in a C# application. For example, if you highlight `MapInfo.Geometry.LineString`, the help topic for `LineString` will not display. If you highlight only the word “`LineString`” in the same line and click F1, the correct Help topic displays. This is an issue with Visual Studio.

In Visual Basic applications, the F1 Help button yields the correct help topics regardless if you highlight the entire line or only a portion of it.

MapXtreme Web Site Template

The MapXtreme Visual Studio 2005 and 2008 Web Site templates open to the Source view that shows the HTML for `webform1.aspx`. This behavior is dictated by a user preference in Visual Studio under the Tools > Options menu in which the user can choose between Source View and Design View. MapXtreme templates honor the user preference over the built in preference to display in Design View.

Special Characters in Field Names

MapXtreme does not support special characters in field names unless it meets the following criteria:

The following symbols are valid in a field name *only* if they exist at the end of the identifier and they are not the only character in the field name.

- \$
- @
- %
- !
- &

To verify that a field name is valid, MapXtreme now checks the name against a table of invalid characters. Anything not found in the table, is acceptable for a field name. That means fields names may contain alpha and numeric characters a-z, A-Z and 0-9.

TableInfoServer.Temporary Property

In the MapXtreme Developer Reference, it incorrectly states that you can set `MapInfo.Data.TableInfoServer.Temporary` to true. For this table type, creating a temporary table is not supported. `TableInfoServer` now throws an exception if `Temporary` is set to true.