

MapObjects 2.1 file requirements for deploying applications

The following section updates the MapObjects file requirements information from pages 232–234 of Chapter 9, ‘Deploying Applications’, in *Building Applications with MapObjects*.

Core Microsoft components

You may need to install and register core Microsoft files so that MapObjects applications may function correctly. MapObjects was developed using Microsoft Foundation Classes (MFC) and Microsoft Visual Studio®. Therefore, the deployment of the MapObjects ActiveX control to a target computer relies on the presence of MFC files and other supporting libraries.

The files you need to deploy will vary depending on the Windows platform that you are installing to.

Note: throughout this section, those files marked with an asterisk should be registered on the target system. For full details on ActiveX registration, see the section in Chapter 9 of Building Applications with MapObjects.

Windows 2000

On the Windows 2000 platform, the required core Microsoft files may be assumed to already be present, and therefore you will not be required to install or update them.

Windows NT 4.0 and Windows 95/98

On the Windows NT and Windows 95 or 98 platforms, the files listed below are the core Microsoft files you will need to deploy for a successful deployment of your MapObjects 2.1 application:

Mfc42.dll*
Msvcirt.DLL
Msvcp60.DLL
Msvcrtdll
MsvcRt40.DLL
OleAut32.DLL*
OlePro32.DLL*
RegSvr32.exe
StdOle2.TLB

There are copies of all of these files on the MapObjects 2.1 CD-ROM in the ‘Microsoft’ folder. These versions of the files are part of the Microsoft Visual Studio Version 6.0 Service pack 4.

It is suggested that the files from the CD be used when deploying applications to a target computer in preference to those that are located in your own computer’s system directory.

These files should be installed into the target computer’s system directory, for example:

C:\WinNT\System32 (for Windows NT)

C:\Windows\System (for Windows95 and Windows98)

The files should always be version checked when installing and should only be installed if they have a more recent version than any existing file. The files should not be installed as shared DLLs. For more information on sharing files, see the ‘Sharing MapObjects components’ section in Chapter 9 of *Building Applications with MapObjects*.

Replacing and registering in-use files

It is likely that some of the files listed will already be present on the target computer. In addition, they may already be in use by Windows, having been loaded into memory by existing applications or services. This may cause problems when your installation tries to replace them.

If your installation software supports the replacement of in-use files, see the documentation provided with the software for details. Those installation programs that support the replacing of in-use files normally do so by copying the new files to a temporary location and then prompting the user to restart the computer. During this restart process the files are swapped so that when the computer restarts Windows the new files are in place, ready to be used.

If your installation software does not support the replacing of in-use files, you may use the following Windows Registry technique. This technique replaces the files and registers them automatically after a restart of the computer without the need for user interaction.

The Windows Registry has a key containing values that are executed upon restart and then deleted. This key is called the RunOnce key and is located in the HKEY_LOCAL_MACHINE hive in the Registry, as follows:

```
HKEY_LOCAL_MACHINE\SOFTWARE\  
Microsoft\Windows\CurrentVersion\RunOnce
```

The content of string Values in this key will be run at the command line during a restart of Windows. The value will then be deleted so that it does not occur during each subsequent restart.

Use your installation software to add a string Value to the RunOnce key, using a string similar to the one shown below to execute a command line register of a file:

```
Regsvr32.exe /s "C:\Program Files  
Common Files\ESRI\MO20.ocx"
```

The Regsvr32 application is located in the Windows system directory. It will attempt to self-register any OCX or DLL file that is passed as an argument. The /s switch runs the registration process in silent mode, which indicates that no message will appear during execution. If this switch is not included, a message will appear onscreen stating that the component has or has not been registered successfully.

If you are using this mechanism, then a separate Value is required for each DLL you wish to be registered. You must uniquely identify these Values. One suggestion for the Name of the data Value is to use an abbreviated version of your application name, suffixed with an integer incremented for each entry.

For example:

Name	Data
MyApp1	Regsvr32.exe /s "C:\Program Files\Common Files\ESRI\MO20.ocx"
MyApp2	Regsvr32.exe /s "C:\Program Files\Common Files\ESRI\Shape20.dll"
MyApp3	Regsvr32.exe /s "C:\Program Files\Common Files\ESRI\AiImage20.dll"
MyApp4	Regsvr32.exe /s "C:\Program Files\Common Files\ESRI\Mfc42.dll"

MapObjects file requirements

The following components should be installed to the ESRI sub folder of the target computer's Common Files folder, typically:

C:\Program Files\Common Files\ESRI

The location may vary from computer to computer however. You may wish to check the following registry Key during your installation to be sure that you are using the correct location of the Common Files folder:

HKEY_LOCAL_MACHINE\SOFTWARE\
Microsoft\Windows\CurrentVersion

This key has a Value named CommonFilesDir, which contains the location of the Common Files folder.

Core MapObjects components

All applications should be redistributed with at least the following core component files:

MO20.ocx*

AF20.dll

Shape20.dll*

Pe81.dll

Sg81.dll

Mtch.dll

ArcInfo components

If your application uses ArcInfo coverages, you must install the following data source component:

Coverage20.dll*

ArcSDE components

If your application has ArcSDE data then must install the following ArcSDE components. The components you will need to install depend on the version of the ArcSDE client that your application uses. One or more of these sets of components may be installed.

Note: The version of the ArcSDE client used by your application will be determined by the syntax your application uses to connect to SDE, for example:

```
pDataConn.Server = "sde81:<myserver>"
```

If you allow users to specify their own Server property, therefore, you must distribute the supporting files for all versions of SDE/ArcSDE.

SDE 3.0x

Sde30.dll

Sg.dll

Pe.dll

MOSde302.dll*

ArcSDE 8.0x

Sde80.dll

Sg80.dll

Pe80.dll

MOSde80.dll*

ArcSDE 8.1

Sde80.dll

Sg81.dll

Pe81.dll

MOSde81.dll*

Note: Sg81.dll and Pe81.dll are required by core MapObjects 2.1, and therefore these files are also included in the earlier section ‘Core MapObjects Components’.

Direct Connect components

In addition to the ArcSDE 8.1 client files listed above, if your application makes use of DirectConnect, you must install the following components:

SdeOra8iSrvr81.dll (if connecting to Oracle 8i)

SdeSqlSrvr81.dll (if connecting to SQL Server)

In addition, install the following files to a \Bin\ sub folder of the \Common Files\ESRI folder:

GSrvrOra8i81.dll (if connecting to Oracle 8i)

GSrvrSql81.dll (if connecting to SQL Server)

If your application intends to connect to Oracle, the target computer must also have an installation of the Oracle Net client.

If your application intends to edit via Direct Connect, the target computer must have access to an ArcSDE server license. Set the ESRI_LICENSE_FILE environment

variable to the name of the license server computer, prefixed with a ‘@’.

For example, for a license server computer called Dundas, you should ensure the following environment variable is set on the target computer:

Variable = ESRI_LICENSE_FILE

Value = @Dundas

VPF components

If your application uses VPF data, then you must install the following data source component:

VPF20.dll*

CAD components

If your application uses CAD data, you must install the following data source components:

CAD20.dll*

CADe.dll

CADpi.dll

Core image components

If your application has images (ImageLayers or GRIDs), you must install the following image source components:

Ailimage20.dll*

Mobase.dll

Mofeat.dll

Mogridio.dll

Dforrt.dll

Libtiff.dll

Elib.dll

Efileio.dll

Ergxlib.dll

Ethrlib.dll

Iso8211.dll

Also, if your application requires ERDAS IMAGINE® support, you must install the following folder into the folder where the Elib.dll is located:

MO_Image

You must also install all the sub folders and their contents. This should be the Common Files sub folder ESRI, that is:

...\Common Files\ESRI\MO_Image\Defaults

...\Common Files\ESRI\MO_Image\Devices

etc.

MrSID components

If your application uses MrSID™ ImageLayers, you must install the following data source component:

MoMrSID.dll*

Image catalog components

If your application uses image catalogs, then you will need to distribute the following component:

MolImageCat.dll*

Also, you will need to install support for the particular image types you wish to include in any catalogs. For example, core image support, MrSID support, etc.

ODBC Tables

ODBC may be used with MapObjects to connect to a variety of tabular data formats. If your application uses ODBC to connect to external data, then you will need to install the following component:

OdbcTbl20.dll*

Also, you must ensure that ODBC support and the appropriate ODBC drivers have been installed and configured on the target computer. ODBC support is installed as part of the Microsoft Data Access Components Version 2.5 installation (MDAC 2.5). If you require particular ODBC drivers, then these may need to be installed separately.

INFO Tables

If your application uses tables from INFO sources, then you will need to install the following component:

InfoTable20.dll*

ADO Tables

If your application makes use of MapObjects ADO support, then you will need to ensure that the following file is installed:

ADOTable20.dll*

You must also ensure that Microsoft ADO is installed on the target computer. If you need to install Microsoft ADO, this is best achieved by using the Microsoft MDAC 2.5 redistribution application (MDAC_TYP.exe). For more information on this refer to the article 'Redistributing Microsoft Data Access Components' in MSDN, found online at:

<http://msdn.microsoft.com/library/default.asp?URL=/library/psdk/dasdk/mdac2lyb.htm>.

StreetMap

If your application uses StreetMap data, then you will need to install the following component:

Street20.dll*

MemTable

If your application uses MemTable functionality, then you will need to include the following components:

MemTable21.dll*

MemTableServer.dll*

NADCON support

If your application makes use of projections with NADCON or HARN GeoTransformations then the following folder will need to be installed onto the target computer along with the sub folders NADCON and HARN and their contents:

PE_GRIDS

Geocoding support

The following folder and its contents should be installed if your application which uses US Address matching rules in MapObjects:

GeoRules

If you have international address matching rules, then these should be installed instead.

Fonts

If your application makes use of the ESRI fonts, for example, in a TextSymbol, then you will need to install the font files to the target computer. They will also need to be registered.

Please refer to the appropriate documentation for your installation program and platform for more information on installing and registering fonts. The following MSDN articles have helpful information:

http://msdn.microsoft.com/library/psdk/gdi/fontext_2ylq.htm

http://msdn.microsoft.com/library/psdk/buildapp/setup_9gms.htm

Code page

If your application makes use of MapObjects ISO code page conversion functionality—for example by using ISO shapefiles or ArcInfo Coverages—then you will need to install the following folder and its contents:

Codepage

MapObjects Helper Components

If your application uses the MapObjects Helper Components, MO2Legend and/or MO2Scalebar, then you will need to install the following components:

MO2Legend.ocx*

MO2ScaleBar.ocx*

Msvbvm60.dll*

Using the Visual Basic 6.0 Package & Deployment Wizard (SP3)

If you are using the ‘Visual Basic 6.0 Package & Deployment Wizard’ to install your application, then the wizard will determine which files are required to be installed in order to support a Visual Basic Executable. These files will include run-time Visual Basic support files as well as any files required by the application’s components.

The wizard program establishes which files are required by the components in your application, for example, MapObjects, by reading the vb6dep.ini dependency file. This file has a listing for every MapObjects DLL file and states which files are always required by MapObjects (e.g., AF20.dll) and which files are optional depending on the data sources that your application uses (e.g., CAD20.dll). The groups of optional files are listed in the sections previously.

If you wish to include optional components, then you should add the appropriate file(s) using the ‘Add’ button on the ‘Included files’ section of the wizard. When any of these files are added to the Setup, their dependencies are automatically added, also.

In addition to the files above, all of the core Microsoft files should be added to the installation at this time. During the installation process the Setup program will check if any of its system files are in use by Windows. If this is the case then it will force a reboot so that these files may be replaced before installing the main component of the application.

Note that earlier versions of the Package & Deployment Wizard may not operate as described here.