



# MapXtreme 2004 发行版 6.1 开发人员指南补充手册

2005 年 1 月

## 说明内容:

---

◆ 简介 .....	1
◆ 升级到 MapXtreme 2004 发行版 6.1 .....	2
◆ 管理 MapInfo 会话状态 .....	3
◆ MapXtreme 2004 发行版 6.1 中的错误更正 .....	17
◆ 已知问题 .....	28

## 简介

---

### 本文档的适用对象

本文档是《MapXtreme 2004 开发人员指南》的补充手册。本补充手册不仅包含了升级指示，还涵盖了新的功能并介绍了未在现有《MapXtreme 2004 开发人员指南》中描述的重要技术。因此，所有 MapXtreme 2004 管理员都需要阅读本补充手册。

本补充手册还合并了 MapXtreme 2004 所有发行版的“发行说明”：

- 2004 年 4 月 30 日版本 6.0 (Build 75)
- 2004 年 6 月 30 日的有限发行版 (Build 78)
- 2004 年 12 月 3 日版本 6.1 (Build 104)

有关发行版 6.1 的最新信息包含在[第 17 页的 MapXtreme 2004 发行版 6.1 中的错误更正](#)一节中。当前正在进行调查或开发寻找解决方案的问题列示在[第 28 页的已知问题](#)。

## 升级到 MapXtreme 2004 发行版 6.1

---

卸载任何已安装在计算机上的 MapXtreme 试用版或 beta 版。要升级到 MapXtreme 2004 发行版 6.1, 请运行 Setup.exe 并遵循提示。可以同时安装发行版 6.1 和发行版 6.0 Build 75。如果不确定已安装了哪个内部版本号, 则选择**开始 > 运行**然后输入 **assembly**。

**注:** 如果已安装了有限发行版 (Build 78), 则必须将其卸载之后才能安装发行版 6.1。

### Build 75 桌面应用程序

如果升级到发行版 6.1, 则可以重新编译由 Build 75 创建的桌面应用程序。可能必须将控件重新增加到窗体上。

或者, 还可以将应用程序重定向为使用新的 MapInfo 程序集, 而不必重新进行编译。此项技术指示应用程序查找已引用程序集的升级版本。关于应用程序配置步骤的完整指示, 请参阅[第 9 页的重定向程序集版本](#)。

### Build 75 Web 应用程序

对于使用 Build 75 创建的 web 应用程序, 必须将项目版本号从旧的版本号更改为 1.1.0.104, 才能使用发行版 6.1 开发应用程序。有关更改项目号号的完整指示, 请参阅[第 7 页的创建或重新编译 Web 应用程序](#)。

### 重新定位许可文件

当前的 MapXtreme 2004 用户 (发行版 6.0) 必须将他们的许可文件复制到相应的 6.1 文件夹 (C:\Program Files\Common Files\MapInfo\MapXtreme\6.1) 中。

### Web 示例应用程序

对于 Web 示例应用程序, 必须创建适当的虚拟目录。创建用于 C# 示例应用程序的虚拟目录。对于 VB, 可以将为 C# 应用程序创建的虚拟目录更改为供 VB 应用程序使用的位置。

例如: 在默认安装过程中, 位于

C:\Program Files\MapInfo\MapXtreme\6.1\Samples\Features\ThematicsWeb\cs 中的 C# 示例应用程序 ThematicsWeb 具有已安装好的虚拟目录。应更改此虚拟目录以将它用于 Visual Basic 应用程序 ThematicsWeb。

1. 从“控制面板”中, 打开 IIS。在默认 Web 站点中找到 ThematicsWeb 文件夹。
2. 右键单击该文件夹, 然后将本地路径从:  
C:\Program Files\MapInfo\MapXtreme\6.1\Samples\Features\ThematicsWeb\cs 更改为  
C:\Program Files\MapInfo\MapXtreme\6.1\Samples\Features\ThematicsWeb\vb。
3. 重新启动 IIS。
4. 使用“任务管理器”停止 aspnet\_wp.exe 然后再重新启动它, 在 Windows 2003 操作系统中此 aspnet 进程名为 w3wp.exe。

对于其它 Visual Basic 示例应用程序, 请遵循相同的基本步骤。

## 示例数据安装程序

如果安装了 MapXtreme 2004 发行版 6.0 附带的示例数据，则应在使用发行版 6.1 免费示例数据安装程序安装提供的新示例数据之前卸载该示例数据。卸载还将删除客户更改的任何数据。

**注：** 数据卸载程序将会删除数据，即使这些数据已被客户改变。如果客户已更改了示例数据并且想要保留这些更改，则应在卸载之前将这些数据移动到另一位置。

如果 6.1 数据安装程序检测到旧的示例数据，则安装将会失败。

## Windows XP Service Pack 2

Microsoft Windows XP Service Pack 2 (SP2) 与 MapXtreme 2004 在一起会引发某些已知问题。这些问题通常都与增加的安全功能有关。默认情况下，Windows XP SP2 关闭了许多先前用于与其它计算机进行通讯的端口。另外，如果启用了 Windows 防火墙，则它将阻断与计算机的自动连接，并且这种行为会导致 MapXtreme 2004 发生问题。

Microsoft 提供的“知识库”文章描述了如何选择性地启用程序，以使它们与在安装 Windows XP SP2 之前一样运行。通过将 MapXtreme 2004 增加到“例外列表”中，可以避免同时使用 MapXtreme 2004 和 Windows XP SP2 可能遇到的困难。

请参阅 Microsoft 知识库文章，网址为：

<http://support.microsoft.com/default.aspx?kbid=842242>

有关 Windows XP Service Pack 2 (SP2) 如何影响 SQL Server 和 MSDE 的常见问题解答，请参阅：

<http://www.microsoft.com/china/sql/techinfo/administration/2000/security/>

## 管理 MapInfo 会话状态

### ISession 接口

MapXtreme 2004 MapInfo.Session 命名空间包含有 ISession 接口，定义了对应用程序核心（例如，地图与表）主要信息的访问。ISession 还管理 MapXtreme 应用程序所需资源的初始化。

在 ASP.NET 应用程序中，ISession 的状态对于每个 ASP.NET 会话都是惟一的，这是因为每个会话都表示一个客户端，而每个客户端都有其自己的状态。每当客户端发出请求时，该请求都会检索会话最后的状态、执行请求，然后再将新的状态返回到存储中。

WebSessionActivator 负责自动设置 ISession 实例，以便可以通过 Session.Current 静态属性访问，并且具有特定于正在处理的 ASP.NET 会话的正确状态。WebSessionActivator 是 IHttpModule，并且具有对特定于 ASP.NET 应用程序事件的访问权，这些事件定义请求的生命周期。下文描述了在以下这些事件中 WebSessionActivator 的行为：

- **BeginRequest**

WebSessionActivator 创建新的 ISession 实例（或如果在 web.config 文件中 MapInfo.Engine.Session.Pooled 设置为 true，则从池中获得一个实例）并将该实例存储在特定于请求的存储中，以便可以通过 Session.Current 属性访问 ISession。

如果在 web.config 中 sessionState 模式设置为 InProc 且 MapInfo.Engine.Session.State 设置为 HttpSessionState，则此步骤会被推迟到 AcquireRequestState，这是因为 ISession 实例是存储在内存中的，并且必须从内存中才可检索。

- ...

- **AcquireRequestState**

如果 MapInfo.Engine.Session.State 设置为 HttpSessionState 且 sessionState 模式设置为 StateServer/SQLServer，则会对位于特定于请求的存储中的当前 ISession 实例执行状态反序列化。

如果 sessionState 模式设置为 InProc，则 WebSessionActivator 会从内存（ASP.NET 会话状态）检索现有 ISession 实例并将该实例存储在特定于请求的存储中，以便可以通过 Session.Current 属性访问。

- ...

- **IHandler 执行**

在此步骤中执行特定于 web 页面或 web 服务的代码。可以通过 Session.Current 属性访问特定于请求的存储中的 ISession 实例。

- ...

- **PostRequestHandlerExecute**

如果 MapInfo.Engine.Session.State 设置为 HttpSessionState，则在 WebSessionActivator 将 ISession 实例存储在 ASP.NET 会话状态时发生此事件（不考虑 sessionState 模式设置）。

- ...

- **ReleaseSessionState**

ASP.NET 引擎根据 sessionState 模式设置将 ASP.NET 会话状态保存在内存中或将其序列化到服务器存储中。

- ...

- **EndRequest**

WebSessionActivator 处置 ISession 实例（或者，如果 MapInfo.Engine.Session.Pooled 设置为 true，则将其返回给池）。

如果 MapInfo.Engine.Session.State 设置为 HttpSessionState 且 sessionState 模式设置为 InProc，则改为从特定于请求的存储中停用 ISession 实例，但不会处置它，这是因为将对同一会话实例的下一请求重用该实例。

## 使用 Session.Dispose(HttpSessionState) 方法

本节描述了在何时以及如何从 Session\_End 中调用 Session.Dispose(HttpSessionState) 方法。

所有使用 MapXtreme 2004 web 控件的 web 应用程序默认使用以下 web.config 设置（即使未在 web.config 文件中出现）：

```
<add key="MapInfo.Engine.Session.State" value="HttpSessionState" />
<sessionState mode="InProc" />
```

第一个设置是特定于应用程序的设置，控制用于保存和恢复 MapInfo.Engine.ISession 实例状态的机制。可以通过 MapInfo.Engine.Session.Current 静态属性访问该实例。HttpSessionState 设置指示通过 ASP.NET 会话状态保存和恢复此会话。此状态通过当前的 HttpContext 公开并且该状态是一种 HttpSessionState。

第二个设置是 ASP.NET 设置，该设置控制保存和恢复 HttpSessionState 的方式。默认设置 (InProc) 指示 ASP.NET 会话状态的内容将会放入内存中，并且对于每个 ASP.NET 会话实例，会话状态的内容都是惟一的。

在使用这些设置时，每个 ASP.NET 会话都具有一个 ISession 实例，并且在整个 ASP.NET 会话生命周期内，ISession 实例都存储在 HttpSessionState 中。为了使 ISession 实例在会话超时或结束时能正确处置，必须将以下语句增加到 Global.asax Session\_End 方法中。

```
MapInfo.Engine.Session.Dispose(this.Session);
```

在 C# 中或

```
MapInfo.Engine.Session.Dispose(Me.Session)
```

在 VB.NET 中

执行此调用将会确保 ISession 实例被正确处置并且回收内存，以避免应用程序耗尽所有内存。

不必为其它任何配置使用此调用，这是因为在其它任何配置中 ISession 实例未存储在内存中。

## 使用 MapInfo.Session.Engine

本节描述了在何时以及如何使用 MapInfo.Session.Engine.UseCallContext web.config 设置。

### Session.Current 与 Response.Redirect 和 Server.Transfer 的问题

在 web 应用程序中使用 Response.Redirect 或 Server.Transfer 时，有可能会对将由 web 服务器接收的同一 ASP.NET 会话实例同时发出多个请求。较高的加载量可能会使部分请求在其它线程中执行。例如，BeginRequest 可能会执行不同于 IHttpHandler 执行的线程，这可能会产生无法解决的异常。这种风险在使用多个框架时甚至会更高。

WebSessionActivator 将 ISession 实例存储在特定于请求的存储中。默认情况下，此存储是通过使用特定于线程的存储内部实现的。如果在使用重定向/传输时给定指明的行为，可能会出现以下问题：

- 如果 IHttpHandler 在另一线程中执行，则它可能不具有对其特定于请求的 ISession 实例的访问权。会抛出以下应用程序异常：

session 会话不活动是因为未注册 WebSessionActivator，或试图在 IHttpHandler 的构造函数中访问 session。要注册 WebSessionActivator，请在 web.config 文件中加入以下项：  
<httpModules><add type="MapInfo.Engine.WebSessionActivator, MapInfo.CoreEngine, Version=<version>, Culture=neutral, PublicKeyToken=93e298a0f6b95eb1" name="WebSessionActivator" /></httpModules>。请用 MapInfo.CoreEngine.dll 程序集当前的版本号来替换 <version> 值。

- WebSessionActivator 无法将 ISession 实例存储在特定于请求的存储中，因为它正在另一请求执行的线程中执行。

已存在活动的 session。

要解决此问题，可以将存储 ISession 实例的存储类型从特定于线程切换为特定于调用上下文。特定于调用上下文的存储允许 ISession 实例在 ASP.NET 请求的生命周期内保持不变，而不考虑请求的每个部分是在哪个线程中执行的。要启用这个特定于调用上下文的存储功能，请将以下特定于应用程序的设置增加到 web.config 文件中：

```
<add key="MapInfo.Engine.Session.UseCallContext" value="true" />
```

使用此设置后，应不会再看到上述的问题。

UseCallContext 设置会影响性能，应用程序运行速度可能会变慢。

### 使用 WebService 客户端进行异步调用时的 Session.Current 问题

Visual Studio .NET 中的 web 服务客户端支持异步调用 web 服务方法。可以通过搜索 MSDN web 站点（网址为：<http://www.microsoft.com/china/msdn/default.asp>）获取有关使用 SoapHttpClientProtocol.BeginInvoke 方法的信息。

如果服务器是 ASP.NET web 服务实现 (asmx) 的，且为同一会话实例从客户端异步调用多个方法，则可能会发生以下问题：

1. 对于专门在服务器端启用 ASP.NET 会话状态的 web 服务方法（需要设置 cookie 容器以使它在客户端正常运行）：
  - a. 某些请求是在单独的会话状态下处理的，而不是使用同一状态以同步方式处理的。这意味着会话状态是经过折衷了。该问题是 MapXtreme 2004 独有的问题。目前尚无对此问题建议的解决方案。

- b. 对于那些确实需要共享同一会话状态的请求，它们的反序列化可能会在请求调用上下文外部进行。如果 `UseCallContext` 设置为 `true`，则会产生异常，这是因为无法达到会话（无可用的上下文）。如果 `UseCallContext` 为 `false`，则反序列化会在单独的 `ISession` 实例中进行，而请求执行的其余部分无法到达该实例，这样会丢失状态。

除了避免从客户端进行异步调用以外，目前尚无对这两个问题建议的解决办法或解决方案。

2. 对于任何 web 服务方法（不考虑是否已启用了 ASP.NET 会话状态），部分请求生命周期可能会在其它线程中执行，所产生的问题与上述的 `Response.Redirect/Server.Transfer` 问题类似。这种情况下，推荐将 `UseCallContext` 设置为 `true` 以解决此问题。

## Web 配置更改

### 升级 WMS/WFS 服务器

要进行升级，必须将 web 配置文件更新为使用新的 WMS 和 WFS 服务器。

1. 编辑 `web.config` 文件，为 WMS 和 WFS 服务器指定 MapXtreme 2004 Build 104。例如，当前的 `web.config` 文件（用于 Build 75）可能指定 `Version=1.0.0.75`。必须将它更改为使用 `Version 1.1.0.104`。更新后的 `web.config` 文件可能如下所示：

```
<httpHandlers>
  <add verb="GET,POST" path="*.ashx"
  type="MapInfo.Wfs.Server.HttpHandler, MapInfo.Wfs.Server,
Version=1.1.0.104, Culture=neutral, PublicKeyToken=5539cc02a8223bb3"/>
</httpHandlers>
<httpModules>
  <add type="MapInfo.Engine.WebSessionActivator, MapInfo.CoreEngine,
Version=1.1.0.104, Culture=neutral, PublicKeyToken=93e298a0f6b95eb1"
  name="WebSessionActivator" />
</httpModules>
```

2. 停止并重新启动 IIS
3. 使用“任务管理器”停止 `aspnet_wp.exe` 然后再重新启动它，在 Windows 2003 操作系统中此 `aspnet` 进程名为 `w3wp.exe`。

### 创建或重新编译 Web 应用程序

如果重新编译现有 web 应用程序（或创建新的 web 应用程序），则应忽略 `ExitProcess` 键或将其设置为 `false`。例如：

```
<add key="MapInfo.Engine.Session.ExitProcess" value="false" />
```

这和 Build 75 中的设置不同，在 Build 75 中必须将此键设置为 `true`。

要将 web 应用程序从 6.0 重新编译为 6.1：

1. 停止 IIS 并关闭 Visual Studio 的所有实例。
2. 在文本编辑器（如记事本）中打开 `.cs` 或 `.vb` 项目文件。
3. 将 `AssemblyFolderKey = "hk\mapinfo.net"` 更改为 `AssemblyFolderKey = "hk\mapinfo.net 6.1"`。如果没有显示任何键，则为 MapInfo 程序集增加一个键。在“项目”文件中可以使用两种方法制定程序集引用。可以找到与下列其中之一相类似的条目：

- <Reference  
Name = "MapInfo.CoreTypes"  
AssemblyName = "MapInfo.CoreTypes"  
HintPath = "..\..\..\Program Files\Common  
Files\MapInfo\MapXtreme\6.0\MapInfo.CoreTypes.dll"  
AssemblyFolderKey = "hklm\mapinfo.net"  
>
- <Reference  
Name = "MapInfo.CoreTypes"  
AssemblyName = "MapInfo.CoreTypes"  
HintPath = "..\..\..\Program Files\Common  
Files\MapInfo\MapXtreme\6.0\MapInfo.CoreTypes.dll"  
>

对该文件进行必要的更改之后，可能看起来类似如下：

- <Reference  
Name = "MapInfo.CoreTypes"  
AssemblyName = "MapInfo.CoreTypes"  
HintPath = "..\..\..\Program Files\Common  
Files\MapInfo\MapXtreme\6.1\MapInfo.CoreTypes.dll"  
AssemblyFolderKey = "hklm\mapinfo.net 6.1"  
>
4. 在文本编辑器中打开 WebForm1.aspx。将任何出现 1.0.0.75 的地方都更改为 1.1.0.104。
  5. 打开 web.config 并将任何出现 1.0.0.75 的地方都更改为 1.1.0.104。
  6. 重新启动 IIS。
  7. 在 Visual Studio 中重新打开并重新编译该项目。
  8. 使用“任务管理器”停止 aspnet\_wp.exe 然后再重新启动它，在 Windows 2003 操作系统中此 aspnet 进程名为 w3wp.exe。

## 重定向程序集版本

本节描述如何将应用程序配置为使用新版本的 MapInfo 程序集，而不必重新进行编译。重定向程序集版本意味着指示应用程序查找已引用程序集的升级版本。Microsoft 提供了另外一些用于完成此功能的机制。

有关更多详细信息，请参阅 <http://www.microsoft.com/china/msdn/default.asp> 并搜索“重定向程序集版本”以找到适当的主题。

### 重定向程序集版本而不更改现有应用程序

可使用多种方法重定向程序集版本。

#### 应用程序配置文件

推荐客户使用应用程序配置文件来实现程序集的重定向。

可以使用位于“控制面板”管理工具下的 Microsoft .NET Framework 1.1 Configuration 工具来配置应用程序的重定向设置。此工具将会创建和 / 或操控用于应用程序的应用程序配置文件。

应用程序配置文件会覆盖发布方策略文件中的设置。

安装完 MapXtreme 2004 发行版 6.1 之后，请遵循以下步骤重定向程序集版本。

1. 创建应用程序配置文件以将应用程序重定向为从 MapXtreme 6.1 加载新的程序集。

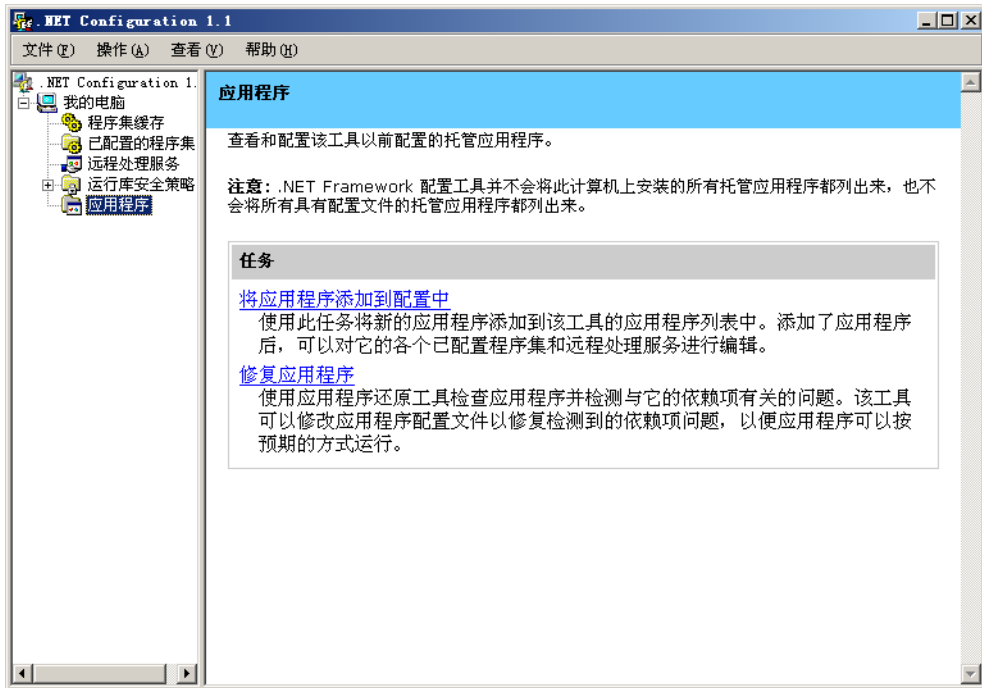
**注：** 配置文件必须与应用程序位于同一目录中，并且配置文件名必须以应用程序名来命名。例如，用于 *myApp.exe* 的配置文件必须命名为 *myApp.exe.config*。可以使用 Microsoft 提供的配置工具来实现此操作。

示例应用程序 .config 文件可能如下所示。新版本为 Build 104:

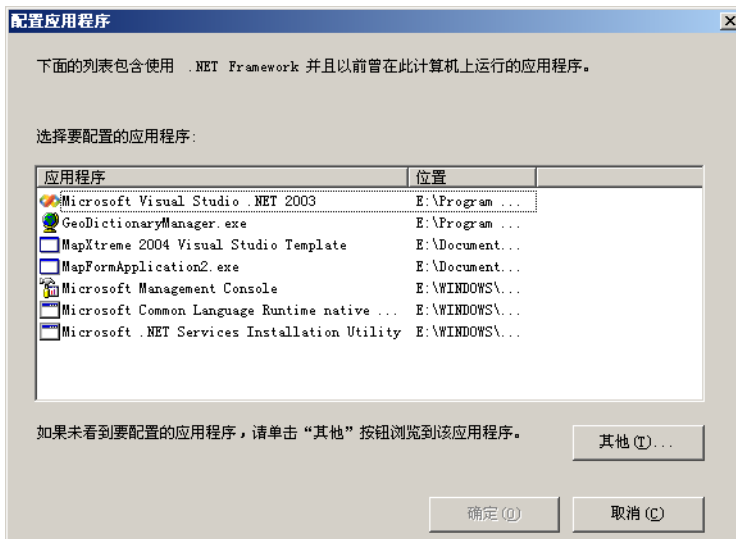
```
<?xml version="1.0"?>
<configuration>
  <runtime>
    <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
      <dependentAssembly>
        <assemblyIdentity name="MapInfo.CoreEngine"
          publicKeyToken="93e298a0f6b95eb1" />
        <bindingRedirect oldVersion="1.0.0.75" newVersion="1.1.0.104" />
      </dependentAssembly>
      <dependentAssembly>
        <assemblyIdentity name="MapInfo.CoreTypes"
          publicKeyToken="f548bcba69d4b8da" />
        <bindingRedirect oldVersion="1.0.0.75" newVersion="1.1.0.104" />
      </dependentAssembly>
    </assemblyBinding>
  </runtime>
</configuration>
```

2. 从“控制面板”选择**管理工具**。

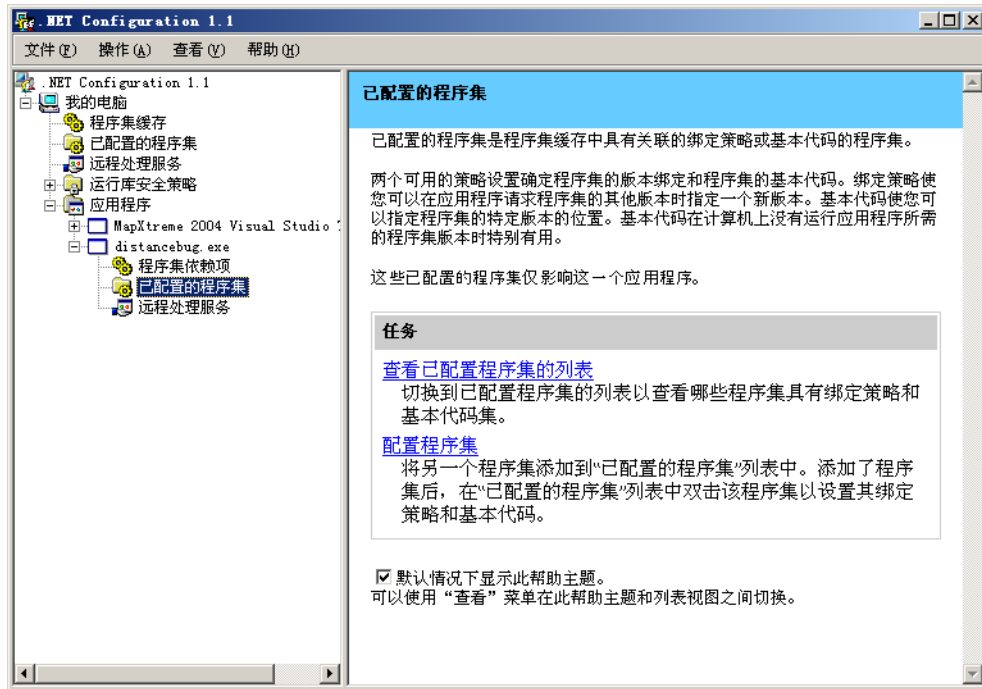
3. 选择 Microsoft .NET Framework 1.1 Configuration。在左侧框架中单击**应用程序**。



4. 选择**将应用程序增加到配置中**。



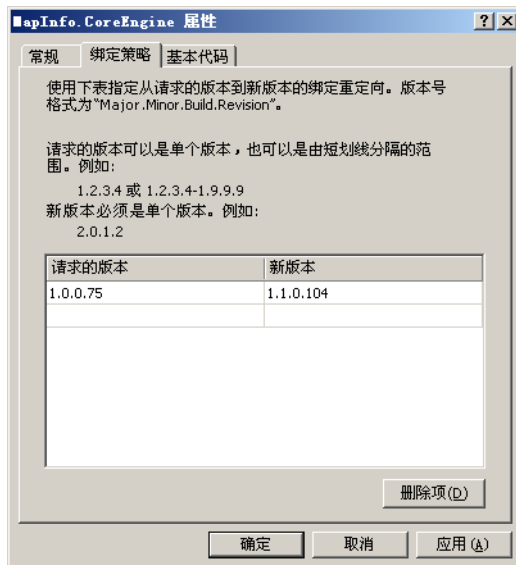
- 单击**其他 ...** 并浏览至应用程序的可执行名。该条目将会显示在左侧框架中的“应用程序”下面。在下面的示例中，应用程序命名为 `distancebug.exe`。单击 + 号展开应用程序。在左侧框架中单击**已配置的程序集**。应看到下面的对话框：



- 在任务列表中，选择**配置程序集**。还可以右键单击“已配置的程序集”，然后选择**增加**。



- 选择第一个单选按钮以从程序集列表中选择程序集，然后单击**选择程序集 ...** 选择程序集（在本示例中选择 `MapInfo.CoreEngine`），然后单击**完成**。现在应看到下面的对话框。选择**绑定策略**选项卡。



- 在 `MapInfo.CoreEngine` 属性对话框中，指定“请求的版本”和“新版本”。“请求的版本”是应用程序 `.exe` 实际参考的版本（如 `1.0.0.75`）。“新版本”是目前重定向应用程序的版本。将新版本指定为 `1.1.0.104` 可以将应用程序重定向到 MapXtreme 2004 发行版 6.1。然后单击**确定**。
- 同样，将要重定向的 `MapInfo.CoreTypes` 和其他程序集配置为新的版本。即重复**步骤 7**，不过要用 `MapInfo.CoreTypes`（或其他相应的程序集）来代替 `MapInfo.CoreEngine`。现在，如果浏览至应用程序所在的位置，则应找到名为 `myApp.exe.config` 或 `myApp.dll.config` 的文件（其中，`myApp` 是应用程序的名称）。

## 发布方策略文件

包含重定向设置的发布方策略文件可与程序集一起安装在 GAC 中。但是，它不是受支持的 MapInfo 配置。请参阅**第 9 页的应用程序配置文件**获取重定向程序集版本的推荐方法。

## 机器配置文件

可以使用位于“控制面板”管理工具下的 Microsoft .NET Framework 1.1 Configuration 工具来配置程序集的重定向设置。此工具将会操控位于下列位置中的机器配置文件：

`WINDIR\Microsoft.NET\Framework\v1.1.4322\Config`。

在此处指定重定向设置将会使所有引用程序集的应用程序使用已升级的版本。应小心使用这种重定向方法，因为机器配置文件会覆盖应用程序配置文件和发布方策略文件中的设置。

## MapXtreme 2004 中的新增内容

---

MapXtreme 2004 发行版 6.1 包含了以下增强功能。

- **并行的安装程序**
- **部署 Web 应用程序**
- **配置自定义光栅句柄**
- **自定义“自定义符号目录”**
- **自定义上下文菜单**
- **键盘快捷键编程**

注： 某些说明开头的数字为 MapInfo 事件跟踪编号。

### 并行的安装程序

并行的安装程序使您可以安装 MapXtreme SDK 的新版本，而不会对现有版本产生影响。原始版本和新安装的 SDK 可以同时并行运行。要实现此功能，并行的安装程序将会创建惟一的虚拟目录、单独的目录以及注册表键。

### 部署 Web 应用程序

《MapXtreme 2004 开发人员指南》概述了使用您自己的安装程序部署应用程序的步骤。在《MapXtreme 2004 开发人员指南》的第二章中，请参阅名为“用自己的安装程序进行部署”的主题以及概述特定步骤的子主题。最终的步骤是为 web 应用程序创建虚拟目录。此步骤不再是必需的步骤，这是因为现在会自动创建虚拟目录。

MapInfo 还提供了下面的白皮书，名为：How to Create, Package, and Deploy a MapXtreme 2004 Application。该白皮书可从 MapInfo web 站点上下载，网址为：

<http://www.mapinfo.com/support/documentation/manuals.cfm>

从 Publications 列表中，选择 MapInfo MapXtreme for Windows，然后选择 MapXtreme 2004 Deployment 白皮书。

### 配置自定义光栅句柄

可以配置 MapXtreme 2004 应用程序不使用 MapXtreme 2004 分发中包括的光栅句柄，而使用另一光栅句柄，或支持全新的光栅类型。还可以更改使用光栅句柄的优先顺序。

在 MapXtreme 2004 的默认安装中，所有光栅句柄都位于 `<program files>\Common Files\MapInfo\MapXtreme\6.1\RasterGridHandlers` 中。这也是文件 `mirasteru.dll` 所在的位置。该位置是要在应用程序中使用的其它任何光栅句柄的推荐安装位置。如果使用默认位置，则不再需要其它配置步骤。

如果要将自定义光栅句柄放置于非默认的位置，则必须为桌面应用程序指定自定义光栅句柄在应用程序配置文件中的位置，或为 web 应用程序指定自定义光栅句柄在 `web.config` 文件中的位置。要执行此操作，开发人员应在 `<ApplicationDataPaths>` 下定义一个 `<Path>` 或 `<SpecialPath>` 元素，然后将所选的光栅句柄复制到该文件夹中。

例如，要在非默认 `MyAppData` 目录中配置桌面应用程序自定义光栅句柄，则应使用下面的 `.config` 文件。

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <configSections>
    <section name="MapInfo.CoreEngine"
      type="MapInfo.Engine.ConfigSectionHandler, MapInfo.CoreEngine,
      Version=1.1.0.104, Culture=neutral,
      PublicKeyToken=93e298a0f6b95eb1" />
  </configSections>
  <MapInfo.CoreEngine>
    <ApplicationDataPaths>
      <SpecialPath>
        <Personal>MyAppData</Personal>
      </SpecialPath>
      <Path>c:\MyAppData</Path>
    </ApplicationDataPaths>
  </MapInfo.CoreEngine>
</configuration>
```

在这个示例中，**<Personal>** 标记是在用户的“我的文档”中定义的特殊位置。指得是“我的文档”中名为 **MyAppData** 的文件夹。在 **<ApplicationDataPaths>** 标记中，或者使用 **<SpecialPath>** 或者使用 **<Path>** 标记。两者都使用就意味着可将光栅句柄放入任一 **"MyAppData"** 文件夹中。虽然不会发生错误，但这可能不是您想实现的目的。由于这是非默认的配置，所以还必须将 **mirasteru.dll** 放在同一目录中。

如果应用程序数据存储相对于 .NET Framework 特殊系统文件夹的位置，则使用 **<SpecialPath>** 语法。例如，如果应用程序数据存储于“我的文档”目录下名为 **MyAppData** 的目录中，则配置文件中的条目可能为：

```
<Personal>MyAppData</Personal>
```

其中 **Personal** 是表示“我的文档”目录的 .NET Framework 枚举 **Environment.SpecialFolder** 的值。

此配置方法还可以用来更改那些可由多个光栅句柄管理的文件类型的首选光栅句柄。例如，JPEG 文件可由 **Halo** 或 **LEADTOOLS**（这两者都捆绑在 **MapXtreme 2004** 中）处理。通常，优先使用 **Halo**，这是因为 **Halo** 的 **\*.rhv** 文件扩展名在字母顺序上排在 **LEADTOOLS \*.rhx** 扩展名的前面。但是，由于 **MapXtreme 2004 v6.1** 首先查找在配置文件中定义的任何 **<ApplicationDataPaths>**，所以将会在找到指定的光栅句柄之后才会在默认的 **[CommonFiles]** 文件夹中查找句柄。因此，例如，可以将 **LEADTOOLS \*.RHX** 句柄复制到已定义的文件夹中，并将 **MapXtreme 2004** 配置为先查找并使用该句柄。

或者，**MapXtreme 2004** 开发人员可以重命名 **[CommonFiles]** 文件夹中的文件扩展名，以使想要使用的光栅句柄先出现在字母列表中。但是，这将会影响使用 **MapXtreme 2004** 开发的所有应用程序，并可能产生无法预料的副作用。另外，如果以这种方式更改文件扩展名，则在卸载 **MapXtreme 2004** 时将不会删除那些已重命名的光栅句柄。出于对上述这些原因的考虑，您可能想要使用配置方法来更改光栅句柄的位置和优先顺序。

## 自定义“自定义符号目录”

自定义符号可以位于 **CustSymb** 文件夹中，该文件夹可以与应用程序位于同一文件夹中，也可以将该文件夹放在其它位置。有关自定义文件位置的信息，请参阅《**MapXtreme 2004** 开发人员指南》附录 B 中的“应用程序数据文件的位置”。

如果想要查找另一 **CustSymb** 文件夹中的自定义位图符号，则在应用程序配置文件中 使用 `<Path>` 或 `<SpecialPath>` 元素。这与 [第 13 页的配置自定义光栅句柄](#) 中所描述的策略非常类似。下面的示例显示了 `<SpecialPath>` 标记：

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <configSections>
    <section name="MapInfo.CoreEngine"
      type="MapInfo.Engine.ConfigSectionHandler,
      MapInfo.CoreEngine, Version=1.1.0.104, Culture=neutral,
      PublicKeyToken=93e298a0f6b95eb1" />
  </configSections>
  <MapInfo.CoreEngine>
    <ApplicationDataPaths>
      <SpecialPath>
        <LocalApplicationData>Your Corporation\Your
Application<\LocalApplicationData>
          <\SpecialPath>
            </ApplicationDataPaths>
          </MapInfo.CoreEngine>
        </SpecialPath>
      </ApplicationDataPaths>
    </MapInfo.CoreEngine>
  </configuration>
```

此示例将会使 **MapXtreme 2004** 搜索类似如下文件夹中的自定义符号：`C:\Documents and Settings\username\Application Data\Your Corporation\Your Application\CustSymb`。请注意 `<SpecialPath>` 元素不应包含 **CustSymb** 文件夹名。如果该元素包含了文件夹名，则路径将会被解释为 `...\Your Application\CustMapXtreme 2004\CustSymb\CustSymb`。重复的 **CustSymb** 是不正确的。

如果应用程序数据存储相对于 **.NET Framework** 特殊系统文件夹的位置，则使用 `<SpecialPath>` 语法。例如，如果应用程序数据存储于“我的文档”目录下名为 **MyAppData** 的目录中，则配置文件中的条目可能为：

```
<Personal>MyAppData</Personal>
```

其中 **Personal** 是表示“我的文档”目录的 **.NET Framework** 枚举 `Environment.SpecialFolder` 的值。

## 自定义上下文菜单

**MapXtreme 2004** 开发人员可以创建自定义的 **LayerControl** 上下文菜单项，这些菜单项会在用户右键单击图层树中的节点时显示。要支持此功能，新的 `ContextMenuTargetObject` 属性返回用户右键单击的对象。

已提供了显示如何定义 **LayerControlEnhancer** 类的代码示例，此类允许用户将自定义项增加到 **LayerControl** 的上下文菜单中。该示例位于 `\Samples\Features\LayerControl` 目录的 **LayerControl** 示例应用程序中。

## 键盘快捷键编程

可以创建键盘快捷键以访问 **LayerControl** 工具栏上的任何内容。`PerformDown`、`PerformUp` 和 `PerformRemove` 方法提供了对“下移”、“上移”和“删除”按钮的编程访问权。

新的 `AddMenuMnemonic` 属性提供了对与显示“增加”菜单相关的键的编程访问权。用于 `AddMenuMnemonic` 属性的语法为：

```
public System.Windows.Forms.Keys AddMenuMnemonic {get; set;}
```

## 将自定义按钮增加到工具栏

MapXtreme 2004 将自定义工具按钮增加到工具栏的步骤现在已简单很多。为了支持此增强功能，已将 `ImageList` 属性增加到 `MapToolBar` 类。此属性已被覆盖以从设计时属性页面隐藏。现在在设计时可以使用 `MapToolBarButton` 的 `ToolId` 属性，以便可以为自定义工具设置值。

要将自定义按钮增加到工具栏，请遵循以下已简化的步骤：

1. 将 `MapToolBar` 增加到窗体。
2. 在 Visual Studio 属性窗口中，突出显示 `Buttons` 属性，然后按 ... 按钮以调用集合编辑器对话框
3. 单击**增加**按钮以增加新的 `MapToolBarButton`。
4. 将按钮的 `ButtonType` 属性设置为 `CustomTool`（下拉列表中的最后一项）。请注意，该按钮这时显示为空白，因为没有和自定义按钮关联的图像。
5. 将按钮的 `ToolId` 属性设置为增加到 `MapControl Tools` 集合中的自定义工具的名称。如果 `ToolId` 值和集合中的工具不匹配，那么当用户单击该按钮时就会抛出运行时异常。
6. 单击**确定**关闭按钮集合编辑器对话框。
7. 在窗体的构造函数中，在 `InitializeComponent` 调用之后，增加代码以便将自定义位图增加到 `MapToolBar` 的 `ImageList`。如何将自定义位图和应用程序关联取决于程序员。一种办法是在设计时将 `ImageList` 增加到窗体，用自定义图像填充，然后编写代码，在运行时将其图像传递到工具栏的 `ImageList`。该代码类似于：

```
// 将自定义工具按钮的位图增加到工具栏的图像列表
foreach (Image image in this.imageList1.Images) {
    this.mapToolBar1.ImageList.Images.Add(image);
}

// 将位图和自定义工具的按钮（列表中最后的图像）相关联
this.mapToolBarButtonBlueSelect.ImageIndex =
this.mapToolBar1.ImageList.Images.Count-1;
```

## MapXtreme 2004 发行版 6.1 中的错误更正

---

MapXtreme 2004 发行版 6.1 的错误更正包括了产品以下部分中存在的问题:

- [数据访问](#)
- [文档](#)
- [Selection 类](#)
- [图例](#)
- [光栅](#)
- [日文 LegacyText](#)
- [安装](#)
- [部署](#)
- [绘制地图](#)
- [持续性](#)
- [SDK](#)
- [用户接口](#)
- [Web 应用程序](#)
- [桌面应用程序](#)
- [常规](#)

注: 每一项说明开头的数字为 MapInfo 事件跟踪编号。

### 数据访问

#### TableInfoServer

31997 TableInfoServer 从 Oracle 增加表时不再忽略 where 条件。

#### Table.UpdateFeature()

32002 Table.UpdateFeature 已可以正确分析语句。该问题由列标记为只读的基础 ResultSet 表造成。

#### AddColumns

T11013 改进了向 TAB 文件增加包含大量行的列时使用 Table.AddColumns 的性能。现在可以更快地将一系列从大型的 MS Access、ADO.NET 或 RDB 表增加到本地 TAB 文件。

#### 视图表中的 OrderBy 子句

T11023 MapXtreme 2004 在使用 TableInfoView() 的查询中忽略 OrderBy 子句。不会抛出异常。此项工作设计为响应 RDBMS 数据库中的不同行为（例如，Oracle 允许而 SQL Server 不允许）。

#### 打开 Oracle 视图时出错

T11188 在 Workspace 管理器中打开 Oracle 视图时偶尔会产生错误。在本版本中已得到更正。

### **Catalog.Search 内存泄漏**

T11696 加载数据之后，Catalog.Search 将最终耗尽内存。此内存泄漏已得到更正。

### **MiScrollable 读取器**

T11750 当尝试使用当前属性时 MiScrollable 读取器不再抛出并出错。

### **InsertFeature 方法**

T11621 InsertFeature 方法现在可以正常运行。

### **Table.InsertFeatures 方法**

T11160 Table.InsertFeatures 方法现在可以正确地在表中插入图元。

### **Table.UpdateFeature**

32002 Table.UpdateFeature 方法现在可以正确更新表中的某一行。

### **Spatialware 加载器性能增强**

T11861 改进了表从 SpatialWare 4.81 向 MapXtreme 的加载性能。

### **FeatureProcessor 类**

T12279 在尝试缓冲生成的 IResultFeatureCollection 时，使用 SearchALL 返回所有记录会产生错误。在本版本中已得到更正。

### **内存管理**

T11696, T12007, T11645 改进了内存管理，使得多种方法不再泄漏内存。

### **连接 Access 和 MapInfo 表**

T11323 某些情况下，将 Access 表与 MapXtreme 2004 USA 表连接将不会产生正确的地图主题。在本版本中已得到更正。

### **TableInfoView**

T12076 某些 SQL 查询在通过 TableInfoView 发出时会产生不精确的结果。在本版本中已得到更正。

### **示例应用程序 Find 错误**

30578 Find 示例应用程序用于非索引表时会抛出异常。本版本中可生成适当的错误消息。

### **Info 工具控件**

T12451 使用 AddColumns 增加一列之后，Info 工具现在可以运行正常。

## 文档

### GridRead 构造函数示例代码

T13262 联机帮助中的 GridRead 类示例已更正为包括 Session.Current 调用。

### LegendControl 示例代码

T13242 联机帮助中的 LegendControl 示例代码已更正。

### TableServerInfo

31857 向 TableInfoServer.ConnectionString 属性描述中增加注解，以指示在用于 ODBC 的连接字符串中必须指定 "DLG=SQL\_DRIVER\_NOPROMPT"。

### TableInfoServer 创建 CacheParameter 类

30173 TableInfoServer 类自动创建 CacheParameter 类并将其类型设置为 On。以前没有明确地写入文档。

### SpatialSchemaPointRef 示例代码

32001 联机帮助中的 SpatialSchemaPointRef 示例代码已更正。

### TableInfoNative 类示例代码

T11830 联机帮助中的 TableInfoNative 类示例代码已更正，从而现在可以创建 .TAB 文件。

### 查询中的 Order By 子句

T11023 对于 TableInfoView 查询，忽略 Order By 子句并且不会生成错误消息。此为预期的行为，现在已写入文档。

### TableInfoView

T10899 使用 TableInfoView 创建基于图元子集的新图层，新图层和基础图层具有相同的 MBR。设计有此特性，并已相应地写入文档。

### Selection 类示例代码

T12120 联机帮助中的 Selection 类示例代码已更正。

### AllowOverlap 与 AllowDuplicates

T12473 如果处理标注时许可，则允许标注之间的复制和重叠。这和 MapXtreme 中的早先行为有所不同。此新行为与 MapXtreme 和 MapInfo Professional 相一致。

### TableInfoAscii 最多字符数

30955 在 ASCII 表中，可显示最多 253 个字符。附加字符被截断。设计有此特性，并在 TableInfoAscii 中写入文档。

## MapExport 类

T12229 联机帮助中的 MapExport 示例代码已更正，以指示必须克隆地图，使用 MapControl 时导出克隆的地图。

## FeatureProcessor 类

T11923 用于 FeatureProcessor 类的示例代码已更新，可以更好的说明如何使用 Combine 方法。

## Selection 类

### IFeatureCollections 无法增加到 Selection

Selection 为 MultiResultSetFeatureCollection，仅支持增加 IResultSetFeatureCollections。不支持增加其它类型的 IFeatureCollections。

结果，向 Selection 增加 IFeatureCollection 时显示的错误消息已更正为：“值必须为 MapInfo.Data.IResultSetFeatureCollection 类型”。

### Selection.Add 方法

31995 Selection.Add(IFeatureCollection) 方法已改正为允许将 IResultSetFeatureCollection 作为 IFeatureCollection（其基本接口）增加到 Selection。以下示例使用 Selection.Add 方法在 IResultSetFeatureCollection 中选择图元：

```
Dim pFeatureCollection As MapInfo.Data.IFeatureCollection
Dim pTable as MapInfo.Data.Table
Dim pRdr MIDataReader
Dim pFtr As Feature

' 使用表加载地图并检索表:
MapControl1.Map.Load(New MapInfo.Mapping.MapTableLoader("C:\Program
Files\MapInfo\MapXtreme\6.0\Samples\Data\USA.TAB"))
pTable = MapInfo.Engine.Session.Current.Catalog.GetTable("USA")

' 获取第一个图元:
pRdr = pTable.ExecuteReader()
pRdr.Read()
pFtr = pRdr.Current
pRdr.Close()

' 创建结果集图元集合:
pFeatureCollection =
MapInfo.Data.FeatureCollectionFactory.CreateResultSetFeatureCollection
(pTable, pTable.TableInfo.Columns)

' 将图元增加到集合:
pFeatureCollection.Add(pFtr)

' 将图元集合增加到 Selection 以选择图元:
MapInfo.Engine.Session.Current.Selections.DefaultSelection.Add
(pFeatureCollection)
```

## 图例

### 图例边框

T11038 现在可以在图例周围适当地绘制边框，不用考虑其垂直或水平方向。

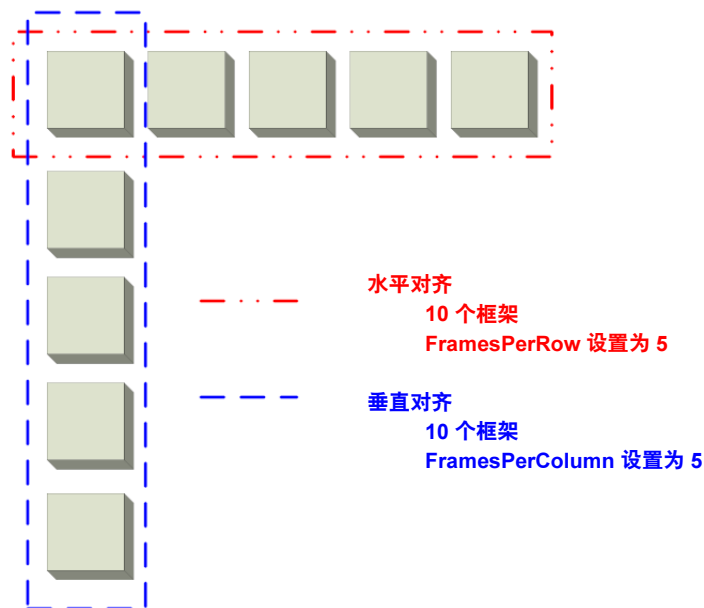
### 图例框架对齐属性更改

T11036 LegendFormat.FramesPerRow 和 LegendFormat.FramesPerColumn 属性的默认值已从 1 更改为 0，以与 FramesAlignment 属性的行为保持一致。FramesPerRow 和 FramesPerColumn 与 FrameAlignment 一起使用，可以水平对齐或垂直对齐框架。

例如，假设想要水平对齐包含 10 个框架的图例。水平 FrameAlignment 意味着要使用 FramesPerRow 值。如果 FramesPerRow 值为 5，结果将为两行，每行包含 5 个框架。

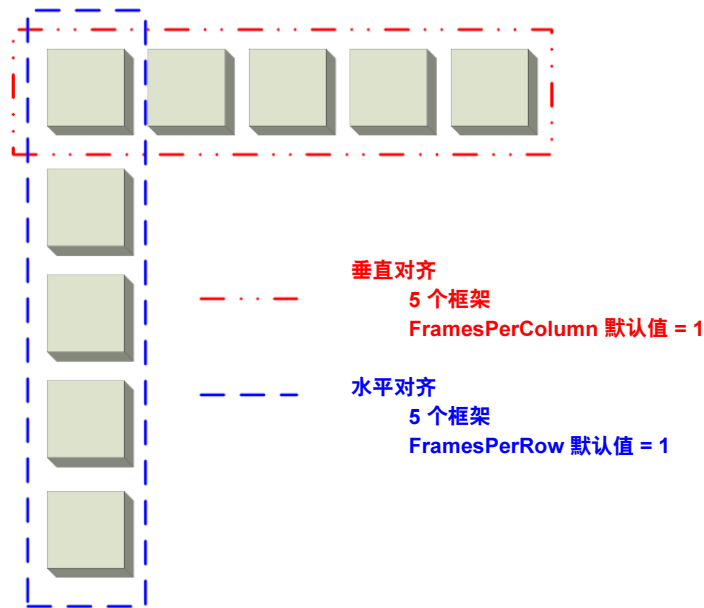
同样，要在图例中垂直对齐 10 个框架，将 FramesPerColumn 设置为 5，将会得到 5 行，每行 2 个框架（每列 5 个框架）。

以下图例显示 10 个框架垂直和水平对齐时的情形。



Build 75 中使用默认值为 1，根据其设计，使对齐看起来象是其自身的反向。默认值为 1 的 FramesPerRow 在图例中有多个框架时（每个框架都在其各自的行中）看起来更象是垂直对齐。请参阅以下图例。

同样，默认值为 1 的多框架 `FramesPerColumn` 看起来象是水平对齐（每个框架在其各自列中）。新的默认值为零，则只考虑 `FrameAlignment` 值，而不考虑图例中的框架数目。



## 光栅

T10973 Halo JPEG 光栅句柄现在为线程安全的。

## 日文 LegacyText

T10922 日文 LegacyText 对象现在可以正确地显示为非默认字体。

T11144 创建 LegacyText 对象时现在允许使用日文字体名称。

## 安装

T12914 用于 WMS/WFS 的 XML 方案现在包括在 MapXtreme 2004 CD 中，已正确安装。

## 部署

### Build 75 运行时

如果是升级到 MapXtreme 2004 发行版 6.1，可以与发行版 6.0、Build 75 一同安装。

## 和自己的安装程序一起打包

发行版 6.1 解决了与 MapXtreme 2004 一起使用自己的应用程序安装程序时出现的多个问题，如下所示：

T10905 如果使用自己的安装程序打包应用程序，并且应用程序使用 COM+ 对象池，则必须在其中包括 MapInfo.Web.MSM 以及 MapInfo.CoreEngine.MSM。这样会将 CoreEngine 注册为 COM+ 对象。

T10903 使用合并模块 (MSM) 所创建的应用程序安装程序现在可以在适当的位置（System32 文件夹）安装 Microsoft C Runtime DLL。

T10907 创建为使用 MSM 的应用程序安装程序现在可以正确地创建虚拟目录 (VDir) MapXtremeWebResources6\_1。

## 与 LayerControl 一起部署

T12671 与 LayerControl 一起部署 MapXtreme 2004 应用程序现在可以正常进行。以前可能会遇到无法解决的异常。

## 绘制地图

### Geoset 导入

T12443 Geoset 导入现在可以正常运行。与事件 T12473 中所描述的 AllowOverlap 和 AllowDuplicates 行为有关。

### Workspace 管理器中的修饰

T11702 在某些情况下，打开 .TAB 文件时会显示修饰，而打开 Workspace 时不显示修饰。在本版本中得到了更正，打开 workspace 时修饰也会如期显示。

### ScaleBarAdornment

T11703 在 Workspace 管理器中更改 Map 缩放时，ScaleBarAdornment 值可以正确更改。

### 修改了的标注行为

T11819 在 visible range 上启用“自动标注”时，会适当地显示修改了的标注（使用“选择”工具拖动）。

### WMF 导出

T12122 以 WMF 导出的图像现在能够以正确的文件格式导出。

### 通过图层控制打开 Geoset

31692 某些情况下，通过图层控制打开 geoset 可能会产生故障。在本版本中已得到更正。

### SeachWithinScreenRadius 与 SearchWithinScreenRect

T13037 使用 ResultSetCombineMode.Replace 时，SearchWithinScreenRadius 与 SearchWithinScreenRect 现在可以正确更新。

### 标注的对齐

T11590 直线的默认对齐现在为 `Alignment.TopCenter`。点的默认对齐现在为 `Alignment.CenterRight`。这与 MapInfo Professional 中的行为一致。

### 标注上的标注线

T11933 标注的标注线在应当隐藏时仍然显示。本版本中已得到更正，使标注线在 `CalloutLine.Use = False` 时不再可见。

### Geoset 缩放速度

T10985 Geoset 在 Workspace 管理器中打开的更快。

## 持续性

### Workspace 地图单元

T12097, T12043 MapXtreme 2004 现在可以使用设置为 "survey ft" 和 "sft" 的地图单元加载 workspace 文件。

### 保存在 Workspace 中的视图表

T11833 在 workspace 中保存时视图表可以适当地保存为 Native .tab 文件。

### 标签位置

T12654 当使用上偏移或下偏移时，Workspace 管理器现在可以适当地保存标注位置。

### LabelLayer 持续性

T11350 LabelLayer 可视性现在可以正常地保持在 Workspace 管理器中。

### 标注字体持续性

T12086 编辑后的字体现在可以正常地保持在 Workspace 管理器中。

### Legend.Format.FramesPerRow

T11274 当 `FrameAlignment` 设置为水平时，`Legend.Format.FramesPerRow` 现在可以正常地保存。

### 标注上的标注线

31907 标注上的自定义标注线现在可以保持在 Workspace 管理器中。

### GML 属性

T12891 GML 属性现在可以根据 W3C 规范写出。

## SDK

### 桌面许可证和对象池 Web 应用程序

T12909 以前，对象池 web 应用程序可以使用桌面许可证运行，运行时不需要水印。在本版本中已得到更正。

### Web 许可证问题

T12911 如果在 web 应用程序中启用对象池，如果仅有 web 运行时许可证，web 应用程序将无法运行并抛出异常。在本版本中已得到更正。

### Workspace 管理器崩溃

T13132 在某些情况下，Workspace 管理器在关闭所有表时会产生 `ArgumentNullException` 崩溃。在本版本中已得到更正。

### 条形图主题

T12719 创建条形图主题然后堆叠条形图主题时会导致崩溃。在本版本中已得到更正。

### 增加投影

T11546 对于 WFS，尝试加载包含 MapInfo Professional 7.8 中增加的新投影的表时会发生服务器异常。在本版本中已得到更正。

### 自定义菜单项

T13021 增加自定义菜单项（使某项增加到图层树）可能会造成 `LayerControl` 上下文菜单的短暂混乱。在本版本中已得到更正。

## 用户接口

### 信息工具图标

31693 选择之后，信息工具图标现在可以清楚地显示为激活状态。

### 通过图层控制修改主题样式

31969 在某些情况下，通过图层控制修改主题样式会造成一些无法解决的异常。在本版本中已得到更正。

### FeatureStyles 示例应用程序

31891 在 FeatureStyles 示例应用程序中，如果缩小主窗体使 `MapControl` 不再可视，将会产生 `System.ArgumentException`。在本版本中已得到更正。

### ModifyRangedThemeDlg

T11279 使用“增加主题图”向导创建 `RangedLabelTheme` 时，现在可以成功地调整范围数量。

## Workspace Manager

T11704 打开包括修饰的 workspace 之后，依次选择文件 > 新建 > 否（不保存当前地图）将会产生异常。在本版本中已得到更正。

## 主题图向导

31935 通过“地图”菜单启动“主题图”向导之后，增加有范围的标注主题图时可能会生成无法解决的异常。在本版本中已得到更正。

## 表达式对话框

T11609, 31864 “主题图”向导的表达式对话框现在可以正常关闭。

## 删除标记图层

31926 通过“主题图”向导创建标注主题图之后，从地图删除标注图层将产生 MapException 错误。在本版本中已得到更正。

## 标注表达式

29997 某些正确的标注表达式原来解释为无效表达式。正在已正常工作。

## AddAdornments

31643 以前，如果向已经包含了修饰的地图中再增加修饰，就要将新的修饰增加到底部而不是顶部。这样就造成了比例栏隐藏在标题后面。此问题在本版本中已得到更正。

## 拖动 / 移动修饰

31641 现在可以拖动并移动地图上的图例、标题以及比例栏修饰。

## Web 应用程序

### SelectionChanged 句柄

T11648 用于 PolygonSelectionTool、RadiusSelectionTool 和 RectangleSelectionTool 的 SelectionChanged 句柄现在可以在 web 应用程序中运行正常。

### State 示例应用程序

31587 从 web 访问的 State 示例应用程序现在可以运行正常。

### ASP.NET 应用程序

31273 VB ASP.NET 应用程序现在可以正常运行对服务器的回贴操作。

### Response.Redirect 和 Server.Transfer

T13463 某些情况下，Response.Redirect 和 Server.Transfer 将会抛出无法解决的异常。在本版本中已得到更正。该解决方案包括新的 Session.Dispose 方法，用于正确地处置 ISession 实例和新的 web.config 条目，强制 ISession 实例的特定于上下文的存储。关于如何管理 MapInfo.Engine.ISession 实例的详细解释，请参阅[第 3 页的管理 MapInfo 会话状态](#)。

### 对象池应用程序的内存泄漏

T11483 处置图例之后不会释放内存。在本版本中已得到更正。

## 桌面应用程序

### Add 工具

31799 增加工具现在可以在适当的位置以适当的缩放级别增加工具。

## 常规

### 许可证失效

A10015 安装 MapXtreme 2004 的丹麦版本时的许可证失效已得到更改。

### 安装程序失效

T10905 如果开发人员使用合并模块 (MSM) 发布 MapXtreme 2004 程序集，那么使用 COM+ 访问 MapXtreme 的应用程序将失效。

### AddWebForm 向导

T13025 MapXtreme AddWebForm 向导的路径已更正。此向导现在可以正常启动。

### Geoset Selectable 属性

T13249 MapXtreme Geoset Manager 中对 Selectable 和 Editable 属性所作的更改已加强。

### Workspace 管理器 TitleAdornment

T11713 现在，当文件从 Workspace 管理器加载时，TitleAdornment 可以正常显示。

### Geometry 对象的 Distance 方法

T12588 现在，当两个对象的几何体在不同的坐标系中时，Distance 方法可以正常工作。

### LoadMapWizard

31778 现在，根据所有的表是否可以成功打开，LoadMapWizard 返回值已更正。

### 日文运行时

T11536 Workspace 管理器现在可以在日文运行时中正常运行。

### 缩放级别

T11076 最大和最小缩放级别现在可以与 Geoset Manager 和 Workspace 管理器保持一致。

### 搜索结果

T13344 现在，当搜索操作检索到多个匹配时，这些匹配可以在列表框中正常显示。

## 已知问题

---

以下问题为发行版 6.1 发行说明中的已知问题。有些问题目前正在研究之中，有些问题则处于积极寻找解决方案的过程中。

- [数据访问](#)
- [桌面应用程序](#)
- [主题](#)
- [Web 应用程序](#)
- [Workspace Manager](#)
- [安装](#)
- [工具](#)
- [常规](#)

### 数据访问

- 24759 增加具有非常小的坐标系边界的 Oracle 图层会导致错误。
- 30165 对无效 dbtype 设置新的 MIPParameter 会抛出 NotSupportedException，现在已经得到更正。
- 30375 打开从 Oracle 数据库下载的表时（TAB 文件中未包含密码）不提示用户输入密码。为避免此问题，请在 tab 文件中增加密码。
- 30503 无法通过 TableInfoServer 增加 Access 表。不支持在 TableInfo 中未填写列信息就直接打开 \*.mdb 文件。为避免此问题，打开含有 tab 文件的 Access 表或在 TableInfo 对象中填入列信息。
- 30839 象 Ä 和 Ü 等特殊字符的 GROUP BY 不返回正确结果。
- 31278 未使用 tab 文件中的密码信息将链接的 ODBC 表加载到 MapControl 中会造成 web 页面超时。为避免此问题，请在 tab 文件中增加密码。
- 31529 当 MapTableLoader 加载 DBMS 表失败时，会抛出消息信息不完全的 TableException 错误。表名称没有包括在消息中。
- Web 控件不支持框架。

### 桌面应用程序

- 31268 在 windows 窗体上复制和粘贴桌面 MapControl 时会留下一些属性为空。以下属性会保留为空白：LeftButtonTool、MiddleButtonTool 和 RightButtonTool。如要改正，在属性窗口中手动重置控件的属性。
- 30800 样式对话框第一次启动会很慢。之后调用打开特定对话框启动会比较快。
- 30365 如果使用修饰键或鼠标滚轮，橡皮圈形状的工具会在地图上留下痕迹或橡皮圈图像的残迹。为避免此问题，在使用任何工具时请关闭鼠标滚轮支持。
- 29444 使用多边形选择工具或自定义多边形 / 折线工具时，在绘制完成时橡皮圈图像始终不会完全消失。如果重绘速度没有问题，则在 ToolUsed 事件之后调用 Map.Invalidate() 方法。
- 30310 如果使用平移或其它工具，鼠标滚轮平移会使地图绘制质量很差。如果在应用程序中禁用鼠标滚轮，则可以避免此类错误。也可以编写程序使地图定期重绘。
- 31955 只有当用户在 LayerControl 中以网格图层作为顶部图层时，GridInfoForm 示例应用程序才工作。要使应用程序正常工作，请将网格图层移到 LayerControl 的顶部图层。

## 主题

- 31915 未显示标注源时，CreateThemeWizard 允许用户选择标注主题图。这样操作之后，会接收到一个含义不明确的错误：“无法绑定到新的显示成员。参数名称：newDisplayMember。”
- 31555 如果使用阴影线线型，则缩放时饼图主题图中的饼图颜色将会更改。
- 31234 饼图主题图显示没有图元的某些行的饼图。饼图主题图在表中显示每一行的饼图，甚至为没有图元的行显示。
- 30036 如果创建包含负数的条形图，条形图不受饼块的边界框限制。
- 30175 向 Workspace 管理器的“修改范围主题对话框图例”选项卡中的图例增加范围之后，图例边框没有正确更新。
- 30977 独立值（标注）和有范围的（标注）紧凑图例没有符号，只是文本。
- 31479 “范围自然间隔”方法不会在 MapXtreme 2004 中生成象在 MapInfo Professional 中那样的结果。
- 31463 渲染时饼图和条形图主题图会随机更改 Z-ordering。

## Web 应用程序

- 31177 如果用于生成主题图的表在 LayerControl、LegendControl 或 MapControl 中设置为不可见的，主题图仍然会在 MapControl 中显示。
- 31502 从 LayerControl 设置时，组图层和标注组图层没有选项以在缩放范围和比例范围中显示。
- 31510 使用 LayerControl 在设计阶段设置时，图层的直线方向、显示节点及质心不在 web 应用程序中显示。
- 31263 在 windows 窗体上复制和粘贴 MapControl 时会留下一些属性为空。这样 MapAlias 和 WorkspaceFile 属性值仍不变，使某张地图上所作的更改无意间在另一张地图上显示。
- 31354 使用 web LayerControl 表示图层的图像可能和桌面 LayerControl 的显示有所不同。
- 31427 web LayerControl 不保持其状态。
- 31700 不使用地图增加表之后会不正确地保持 Workspace。
- 31893 光栅图层的样式覆盖不能通过 web LayerControl 访问。
- 31725 无法在设计阶段为 web 应用程序加载命名连接文件。
- 31743 当页面上没有地图时，默认 VB ASP.NET 模板抛出错误。此错误为“Microsoft JScript 运行时错误：‘未定义’为空或不是对象。”
- 31854 向 web 工具设置无效的 MapID 时，会错误地抛出 NullReferenceException；将会抛出 ArgumentException。
- 31845 在某些情况下，ASP.NET 应用程序渲染空地图并显示其自身状态有问题。
- 31933 Web 示例应用程序在它们的 MWS 文件中有硬编码的路径，如果 MapXtreme 2004 未安装在默认目录下，则其不会正常运行。要解决此限制，需要从 MapControl 清除地图，然后将与此项目关联的表和/或 workspace 重新加载到示例数据的同一位置。下面的示例应用程序使用以下数据文件：ThematicsWeb - World.tab、Tools - World.mws、FindWebSample - World.mws。
- 31496 在 LayerControl（Web 或桌面）中删除组图层不会在删除时显示警告或出错消息。
- 31654 “Web RadiusSelect 工具”未正确绘制。使用该工具绘制时应当可以看到的可视透明圆在选择时被截断。
- 31645 关闭组标注图层父节点的可视性，不会自动关闭子节点的可视性。这些属性在 LayerControl 中设置。

- 31925 如果显示器设置为 16 色（Windows 2000 默认值），Marquee Zoom 不显示。
- 31929, 31881 如果对加载在控件中的地图更改 MapControl 的 ID，则旧 MWS 文件留在硬盘上。同时，从加载了表的 webform 删除 MapControl 也不会删除 MWS 文件。该旧 MWS 文件可以手动删除。

## Workspace Manager

- 30802 对于使用 Workspace 管理器的光栅图层无法创建透明度。可以通过编程方式创建透明度或将该信息保存到 tab 文件。
- 30164 使用 Workspace 管理器时，Shift 和 Ctrl 键的使用相反。当选择多个对象时，请按住 Ctrl 键。取消选择独立的对象时，请按住 Shift 键。此行为与在 MapInfo Professional 中相反。
- 31183 当从 Workspace 管理器打印时，光栅图层的样式覆盖不在输出中显示。例如，半透明样式覆盖的光栅不打印任何透明度。
- 31673 Workspace 管理器在没有安装 OCI 或 IUS 客户端的机器上打开远程 OCI 或 IUS 表时不显示错误。执行该操作时，不会出现任何情况。在 web 应用程序中，从没有在设计阶段安装的客户端的源打开来自远程 OCI 或 IUS 源的图层，会在运行时造成空地图，没有出错指示。

## 安装

- 删除 MapXtreme 2004 之后，在 GAC（全局程序集缓存）中留有 MapInfo 程序集。发生这种情况有两种原因：

aspnet\_wp.exe 进程仍有一个打开的程序集引用，未检测到卸载该引用。要进行更改：

- a. 使用控制面板 -> 管理工具 -> Internet 信息服务来停止 web 服务。选择 web 服务器并停止服务。
- b. 请转到下文的“删除程序集”部分。

安装程序服务破坏了注册表。确定是否为以下情况：

- a. 打开注册表编辑器。（开始 -> 运行并输入 regedit。）
- b. Navigate to HKEY\_CLASSES\_ROOT\Installer\Assemblies\Global。
- c. 检查（默认）键类型并查看其是否为 REG\_MULTI\_SZ。
- d. 如果值为 REG\_MULTI\_SZ，请选择（默认）键，然后选择编辑 -> 删除。

该键将会与类型 REG\_SZ 重新显示。

- e. 请转到下文的“删除程序集”部分。

从 GAC 删除程序集：

- a. 从开始 -> 运行菜单，输入 assembly。
- b. 选择并删除任何 MapInfo.\* 程序集。
- c. 如果收到错误，为所有 MapXtreme 2004 引用检查 AddRemove 程序控制面板。

- 运行安装过程时会显示一个命令窗口，在此可以注册具有 COM+ 服务的 MapInfo.CoreEngine。显示有三种错误（返回的十六进制数在不同的系统上会有所不同）：  
Unhandled Exception: System.DllNotFoundException: Exception from HRESULT: 0x80131524. at \_\_DllMainCRTStartup@12(Void\* , UInt32 , Void\* )  
Unhandled Exception: System.DllNotFoundException: Exception from HRESULT: 0x80131524. at \_\_DllMainCRTStartup@12(Void\* , UInt32 , Void\* )  
Unhandled Exception: System.DllNotFoundException: Exception from HRESULT: 0x80131524. at \_\_DllMainCRTStartup@12(Void\* , UInt32 , Void\* )  
可以放心地忽略这些错误。
- 在 MapXtreme 2004 提供的示例数据中，表元数据中的以下表 coordsys 条目是不正确的：
  - US\_CNTY: Coordsys 元数据指经度 / 纬度（无基准面），而地图文件为经度 / 纬度 (NAD 27 for Continental US)。
  - 墨西哥: Coordsys 元数据指 Conformal Projection（北美），而地图文件为经度 / 纬度 (NAD 27 for Central America)。
  - 美国: Coordsys 元数据指 Conformal Projection（北美），而地图文件为经度 / 纬度 (WGS 84)。
- 29781 运行 .NET 服务安装工具 (regsvcs.exe) 时，安装程序在 COM+ 中注册 MapInfo 引擎时抛出“找不到系统 dll”异常错误。该错误由卸载和加载混合 dll 造成。此问题的更正由 Microsoft 负责。
- 31883 注册表中的序列号在执行了安装修复之后被删除。为避免此操作，用您的序列号替换注册表条目 HKEY\_LOCAL\_MACHINE\Software\MapInfo\MapXtreme\6.0\SerialNumber 键。
- 31701 当以某用户安装，然后以另一用户登录时，MapXtreme Toolbox 在 Visual Studio 中不可用。
- 31832 如果通过删除工具来修改 MapXtreme 2004 安装，修改之后删除程序会抛出错误。

## 工具

- 31855 工具事件返回的坐标系根据缩放级别偏移很长的距离。工具事件当前返回鼠标单击点的大概值，如果对齐模式是开状态，则返回的坐标系距离所选择的点有一长段距离。问题是，事件句柄自动将所有坐标系“对齐”其整数像素的等价值。这意味着缩放级别大时误差更大——在地图宽度较小时也是如此（即使缩放级别相同）。常用 Select 工具和 Add 工具与“对齐”可以正常地工作——但其 ToolUsed 事件报告用户正单击距离实际位置 5 英尺的位置。要解决此问题，从常用 Select 和 Add 工具而不从 Custom 工具派生自定义工具，并与所选择的或创建的实际对象一起工作而不使用 ToolUsed 返回的坐标系。
- 24604 常用工具 AddPolyline 和 AddPolygon 不能够一致地绘制。使用这些工具时无法估计所绘制对象的 Z-order。该问题的一个解决方法是将每个图元加为其自身图层。这样可以确保排序达到预期目标。
- 30205 选择某标注时（通过按住 Select 工具），旋转句柄不工作。
- 30813 MapTool 类的 InfoTipsEnabled 属性不代表每个工具的覆盖。在某特定对象上设置此属性不局限于对该对象的设置，而在全局基础上都有影响。
- 31656 使用 Point Selection 工具时会将选择集反向。（在区域重叠的情况下，例如可选择的海洋图层，选择集互相取消才会发生此情况。）
- 31405 具有状态（例如 EditText、AddPolyline 等）的工具当 MapControl 失去焦点时没有被清除。要更改此种情形，可以编写程序使地图定期重绘。
- 31567 ZoomIn 和 ZoomOut 工具缩放增量与 MapX 不同。注意此行为可以为任何期望的行为重载或自定义。

- 31569 使用 LayerControl 时，缩放工具使地图名称递增。例如，如果地图名称为 Map1，每一次缩放后地图名称就会更改为 Map2、Map3 等。
- 31799 使用 Add 工具向地图增加对象时，新的对象在应该出现的地方没有出现。相反，这些对象出现在 0,0 点。只要地图上发生视图变化，这一行为就会停止。要更正这一问题，在激活任何 Add 工具之后设置 ToolBar Click 事件中的 `Map.Zoom = Map.Zoom`。
- 31917 当选中“动态选择集工具”复选框时，对象在地图上显示为选中，甚至在取消选择集之后。
- 31945 AddText 工具活动时“对齐”模式不工作。
- 31953 当放大倍数很高时，在尝试使用 ZoomOut 工具时缩放级别有时会保持为 0。要离开这一状态，请使用 LayerControl 中的某些控件（例如“查看全部”）更改视图。

## 常规

- 31889 无论实际单位应该是什么，从 geoset 加载的第一个 Zoom 修饰符的单位始终为英里。
- 当在多线程模式中使用 LeadTool 的 JPEG 2000 和 TIFF，MapExport 会失败。在 Windows 2000 中 JPEG 2000 会失败。在所有平台上 TIFF CMYK 会失败。
- WMS 服务器导出的 GIF 文件无法由 MapInfo Professional 读取。MapXtreme 2004 中的 WMS 服务器生成 GIF 图像有效但是会造成 MapInfo Professional v7.5 失效。要解决此问题，请使用由服务器返回的其它图像格式。
- 25344 在此版本中不支持书写形状 (\*.shp) 文件。
- 25630 无法将非地球坐标系中的 Point 转换到 MGRS。该转换将会抛出异常或不做任何操作。在本版本中，该操作会产生不特定的后果。
- 25655 在经度 / 纬度中创建 Point 类型或将 Point 类型的 Data 属性设置到很高的值时，地图视图无法正确剪辑经度 / 纬度值。例如，如果创建经度 / 纬度 Point 或将 Point 的 Data 属性设置为不存在的值如 (999999999.0, 999999999.0)，则此值会被剪辑到 1000, 1000，而不是所设计的 -360, -90, 360, 90。
- 30827 样式库不是类型安全的。将不是样式的对象传递到样式库中会导致崩溃。这一点对于任何实现 IList 或 ICollection 的对象都是成立的。
- 27788 将关于 Font 符号样式的 Angle 属性设置为正值时，符号逆时针旋转。MapXtreme 2004 中的其它对象在 Angle 属性为正值时顺时针旋转。
- 29042 MapXtreme 2004 无法给位图符号绘制空符号，不过可以为线、填充、字体符号等绘制。如果从“符号样式控件”中的“定制符号”选择 N，则会显示一个不需要的符号。要避免这种情况，请使用不同的点样式并使一个为“空”。
- 29092 缓冲具有许多线段复杂的 MultiPolygon/Region 对象会抛出错误。可以通过减少缓冲的数据来避免此现象。
- 29250 应用程序在运行状态时，不要尝试在设计阶段修改 MapControl。如果一定要尝试，则会从 MapControl 抛出 SystemInvalidOperation 异常。
- 29391 MapXtreme 应用程序无法加载包含实时分配了 SpatialSchemaXY 的表的持续性 workspace 文件。为避免这种情况，将空间元数据保存到 tab 文件，然后重新创建 TableInfo 对象。
- 29824 “符号样式”对话框中的“应用符号颜色”复选框在选择新的扩展自定义符号设置时将不可用。此选项应当为不可用的，但是在这里却可用。
- 30528 SelfIntersects 在如图 8 和图 B 所示线性对象 (MultiCurve, Curve, LineString) 的某些情况下无法正常运行。
- 30413 图层不可视时，标注仍显示在图层上。为避免这种情况，关闭“标注修饰符”可视性。

- 30423 当在配置为日语地区的计算机上运行时，某些情况下无法从 tab 文件创建 ASCII 文件。
- 30456 96dpi, 501px X 300px, 8bpp 的 PNG 透明度无法正常工作。
- 30812 切换对齐时，对齐节点交叉线图元存在重绘问题。为避免这种情况，可以通过平移或刷新地图来强制地图重绘。
- 31199 WMS 客户端允许用户不指定任何边界而“获取”地图。必须将边界指定为 GetMap 请求的一部分。
- 31792 不要尝试向 MapControl 中重新加载已经加载好的 MWS 文件。这样会产生错误如“加载文件时出错: xxx.mws”此处 xxx 为被加载文件的文件名。
- 31223 当修改 LabelSource 对象中使用的表时，这些更改在 LabelSource 对象中没有反映。
- 31267 LabelSource 图层别名对于地图不是惟一的，但对于标注图层却是惟一的。要解决此限制，同时使用图层别名和 LabelSource 别名来确保其惟一性。
- 31372 “符号样式”自定义列表框的垂直滚动条不工作。
- 31342 在 Web MapControl 上平移时，图像顶部和左侧显示有边界。
- 31471 无法读取 WBMP 文件。
- 31689 LayerControl 在包含混合对象的缓存远程表上可能会不一致。远程数据源无法始终以预期顺序返回图元。为避免此问题，调用 LayerControl.SetTableImage 以显式指定图像。
- 31480 第一次设置样式覆盖时，不应用样式覆盖的颜色。
- 31481 GRC 和 GRD 光栅句柄是线程安全的。
- 31488 WFS 客户端无法理解 GML2。WFS 客户端只理解 Mapinfo-GML3。因为这样客户端就只与 MapInfo 服务器协同工作。
- 31811 当前代码页与数据代码页不相等时，WriteTabFile 为列名称写出无效 TAB 文件字符。
- 31803 如果找不到默认 coordsys, CoordSysFactory.LoadDefaultProjectionFile() 抛出 NullReferenceException。
- 31809 TableInfo.WriteTabFile() 会写入不可用的 TAB 文件。在某些情况下，此方法会覆盖现有的 TAB 文件，使其无效。仅使用此方法创建新的 TAB 文件，而不要更新或覆盖已有的文件。
- 31617 用于 LegacyText 对象的 Edit 句柄无法正常运行。
- 31668 ScaleBarAdornment 和 ScaleBarAdornmentControl 类会报告不同大小的比例栏。
- 31899 切换“动态选择集”之后编程更改 InsertionLayerFilter 时发生重绘问题。该问题为，所选择区域的 edit 句柄在地图刷新之前是看不到的。为避免此问题，如果不在应用程序中使用“动态选择集”则请不要将其启用。也可以限制 LayerFilter 对最终用户通过 LayerControlDlg 访问的更改。
- 31671 单击地图时，如果所选择的标注不在屏幕上，则不取消选择。
- 31726 GML 发布无法正常读取或写入具有多个几何体列的 FeatureCollections。
- 31702 通过 LoadMapWizard 在底部而不是顶部堆栈新图层来打开多个 geoset。要更正这一问题，在激活任何 Add 工具之后，设置 Map.Zoom = Map.Zoom。
- 31815 缩放图层更改之后，地图可能会错误地记住隐藏的选择集。
- 31791 Mapping.SearchInfoFactory 方法不验证参数。
- 31850 动画图层处于激活时，关闭临时表会导致其标注从显示中消失。重要的是要注意 LabelsAreEditable 属性设置为 True 与动画图层不兼容。要更改此行为请在应用程序中多次调用 Mapinfo.Mapping.Map.Invalidate()。
- 31920 标题修饰显示之后，无法修改其字体或调整大小。
- 31836 如果 workspace 包含有光栅图层的编辑好的标注，那么在加载 workspace 时会出错。

- 31846 CursorImageURL URL 类型在设计阶段不保存。
- 31890 对话框的标记和最小尺寸及示例应用程序窗体常常是错误的或不存在的。这样会造成对话框调整为无外观效果。
- 31898 不要在对话框中对大值不合理地使用科学记数法。可能会导致出错例如“对象引用未设置到对象实例”。
- 31936 FillStyle 对话框中的某些 FillStyle 属性设置禁用所有的对话框选取器。
- 31958 WFS 示例运行时生成 System.URLFormatException。要更正此问题，请更改 simplesample.cs 文件中的 SimpleSample 类的 URL 常数，指向有效的 WFS 服务器。