

MapXtreme® 2005

V6.5

开发者指南

本文档中的信息如有更改恕不另行通知，且不表示对供应商或其代表的任何部分做出承诺。未经 MapInfo Corporation (One Global View, Troy, New York 12180-8399) 书面许可，本文档的任何部分都不可以用任何形式或任何方式复制或传播（电子或传统形式，包括复印）。

© 2006 MapInfo Corporation. 保留所有权利。MapInfo、MapInfo 徽标和 MapXtreme 2005 是 MapInfo 公司和（或）其附属体的商标。

MapInfo 公司总部：

语音电话：(518) 285-6000

传真：(518) 285-6070

销售信息热线：(800) 327-8627

政府销售热线：(800) 619-2333

技术支持热线：(518) 285-7283

技术支持传真：(518) 285-6080

MapInfo 全球办事处联系信息位于：<http://www.mapinfo.com/contactus>。

本文中涉及的产品均为各自制造商的商标，并特此证明。具有商标的名称仅用于文字编辑，一切利益归商标所有者拥有，并不表示有意侵犯商标权利。

Adobe Acrobat ® 是 Adobe Systems Incorporated 在美国的注册商标。

ECW by ER mapper © 1998-2001。

LEAD Technologies, Inc. © 1996。

本文中涉及的产品均为各自制造商的商标，并特此证明。具有商标的名称仅用于文字编辑，一切利益归商标所有者拥有，并不表示有意侵犯商标权利。

2006 年 3 月

目录

第 1 章： MapXtreme 2005 简介	1
概述 MapXtreme 2005	2
主要功能	3
新增内容	4
MapXtreme 学习资源页面	4
最佳实践：规划应用程序	4
对象模型文档	5
Web 控件的体系结构	5
地理编码和路径规划客户端	5
打印支持	6
Shapefile 中的 M 和 Z 值	6
ECW 光栅句柄	6
瞬态图元支持	6
性能测试	7
示例应用程序和代码示例	7
IN/ANY/ALL 运算符和子查询	7
StyleFactory 类	7
ISession.Reload 方法	8
永久性	8
自定义位图符号 MSM	8
许可证文件工具	8
发生变化的内容	9
WinForms 的 LayerControl	9
MultiPolygon 行为中的 Point	9
搜索方法	10
对象处理	10
OCI 连接对话框	11
迁移到 MapXtreme 2005	11
学习使用 MapXtreme 2005	13
支持资源	13

第 2 章：入门指南.....	15
安装	16
系统要求.....	16
安装类型	17
开发 (SDK) 安装.....	17
部署 (运行时) 安装.....	17
SDK 和运行时许可证	18
试用许可证	18
SDK 许可证	18
运行时许可证	18
如何获得许可证?	19
安装之前	19
管理员特权	19
临时目录和其它资源的权限	19
首先安装 Visual Studio .NET 2003	19
.NET Framework SP1 安装	20
Windows 2000 Post-Service Pack 4 (SP4) COM+ 1.0 rollup package 27	20
其它安装特性	20
将 MapXtreme 2005 安装到系统中	20
升级到 MapXtreme 2005	25
更新现有的桌面应用程序	25
更新现有的 Web 应用程序	25
重定向程序集版本	26
在 Visual Studio .NET 中创建应用程序	27
地图应用程序	28
ASP.NET Web 应用程序	30
MapXtreme 2005 控件	32
构建 ASP.NET Web 应用程序但不使用模板	33
部署应用程序	34
用运行时安装程序部署	34
用自己的安装程序进行部署	35
部署访问数据的应用程序	38
已部署 Web 应用程序临时目录的权限	38
应用程序数据文件	38

第 3 章：地图绘制概念	39
地图绘制和 MapXtreme 2005	40
地图	40
表	41
图层	41
图元	41
标注和图例	42
主题	43
工具	43
工作空间	44
坐标系和投影	44
使用 MapXtreme 2005 进行地理编码	45
使用 MapXtreme 2005 进行路径规划	45
第 4 章：理解 MapXtreme 2005 体系结构	47
MapXtreme 2005 体系结构	48
对象模型概述	49
MapInfo.Data 命名空间	50
MapInfo.Data.Find 命名空间	50
MapInfo.Engine 命名空间	50
MapInfo.Geometry 命名空间	50
MapInfo.Mapping 命名空间	50
MapInfo.Mapping.Legends 命名空间	50
MapInfo.Mapping.Thematics 命名空间	51
MapInfo.Persistence 命名空间	51
MapInfo.Raster 命名空间	51
MapInfo.Styles 命名空间	51
MapInfo.WebControls 命名空间	51
MapInfo.Windows 命名空间	51
MapInfo.Tools 命名空间	52
MapInfo.Geocoding 命名空间	52
MapInfo.Routing 命名空间	52
应用程序体系结构	52
体系结构的描述	53
桌面应用程序体系结构	55

第 5 章：应用程序的规划57
规划桌面应用程序	58
将 MapXtreme 2005 添加到现有应用程序	58
示例应用程序和项目模板	58
最佳实践	59
规划 Web 应用程序	60
有多少用户将访问您的应用程序?	61
InProc 应用程序开发模型	62
对象池应用程序开发模型	63
保存和恢复状态：示例	66
对象池会话和状态	69
会话实例（预创建的和对象池的）	69
Microsoft COM+ 对象池	70
应用程序性能	72
CPU 和内存	72
常规性能测试	72
优化应用程序	74
IIS 服务器设置	75
通过有选择地保存对象来进行优化	77
应用程序示例	77
规划数据模型	79
MapXtreme 数据体系结构	79
选取正确的数据源	81
访问数据的方法	83
相关表	85
分析数据	90
改进数据访问性能	92
第 6 章：运用 MapXtreme 2005 Web 控件93
WebControls 命名空间概述	94
Web 控件的体系结构	94
关于 MapTools 体系结构	96
使用 Web 控件	97
什么是 Web 控件?	97
使用新的 Web 控件	98
创建自定义命令	99

创建自定义工具	99
在框架中使用 Web 控件	100
在表单元格中使用 MapControl	101
Web 控件本地化	101
管理控件	102
错误管理	102
状态管理	103
事件处理	106
从 2004 Web 控件迁移到 2005 Web 控件	106
加载数据	106
替换控件	107
状态和事件管理	107
更改和分发 Web 控件	107
第 7 章：使用桌面工具	109
MapInfo.Tools 命名空间概述	110
可用工具	111
View 工具	112
Select 工具	112
Add 工具	113
Custom 工具	113
Shape 工具	114
自定义工具	115
工具事件	116
教程：使用 Select 工具编辑 FeatureGeometry	118
重新定形图元	118
增加节点	120
通过编程方式重新定形和增加节点	120
第 8 章：用于桌面应用程序的对话框和控件	123
MapInfo.Windows.Controls 命名空间	124
要在应用程序中使用的关键控件	124
MapControl	124
MapToolBar	126
图层控件	128

MapInfo.Windows.Dialogs 命名空间.....	129
CreateThemeWizard	130
定制控件和对话框	135
第 9 章：运用核心 MapXtreme 2005 类.....	137
ISession 接口	138
会话管理.....	139
使用 Session.Dispose 方法	139
串行化和永久化	140
串行化.....	140
永久化.....	141
Selection 类	142
使用选择集的属性.....	143
突出显示和导出选择集	143
SelectionChangedEvent.....	143
Selection 和 Selections 类上的 ISerializable 接口	143
Selection 代码示例	144
选择另一个图元中的图元	144
检查选择集的表	145
从表中返回所有列.....	145
更改选择集后的地图视图	146
事件参数	146
异常	146
第 10 章：处理数据.....	147
MapInfo.Data 命名空间概述	148
表	149
表别名.....	149
列	149
Catalog	151
代码示例	152
定位打开的表	153
关闭表	154
打包表中的文件	154
侦听表和目录事件.....	155
表元数据 (TableInfo)	156
检查 TAB 文件元数据	157

创建新表	157
将表达式列增加到表	160
数据源	161
使表可制图	162
数据绑定	164
如何从 Oracle 表连接数据	165
使用 ADO.NET 数据提供方	166
源行	169
GeoDictionary	169
使用 GeoDictionary 的 AutoMatching	169
MapInfo ADO.NET 数据提供方	170
MConnection	170
MCommand	171
MIDataReader	173
MapInfo SQL	174
空间查询	175
合计查询	175
获取特定的记录（由键获取）	175
从特定的记录获取特定的值	176
Insert 语句	176
带有 Select 的 Insert 语句	176
Update 语句	177
Delete 语句	177
IN/ANY/ALL 运算符和子查询	177
表达式	178
创建表达式	178
Where 子句 - 布尔表达式	179
表达式中的函数	179
表达式示例	180
Features 和 Feature 集合	183
Feature	183
Feature 集合	184
搜索图元	185
Catalog 搜索方法	185
SearchInfo 和 SearchInfoFactory	187

第 11 章：从 DBMS 访问数据	193
访问远程空间数据	194
通过 .TAB 文件访问远程表	194
不使用 .TAB 文件访问远程表	195
使用 X/Y 列映射 DBMS 数据	195
Oracle 支持	195
几何体转换	196
多维几何体支持（大于两维）	197
SDO_Geometry Arc 和 Circle 转换	197
可视化不可转换的 Oracle 对象	197
质心支持	198
附加信息	198
Oracle Spatial 引用支持 (SRID)	198
OCI 连接对话框	199
DBMS 连接字符串格式	199
ODBC 连接字符串格式	199
Web 应用程序中的 ODBC 图层和对象池	200
Oracle Spatial 连接字符串格式	200
样本连接字符串	200
在服务器表查询中定义可制图的表	201
几何体列	201
键列	202
访问属性数据	203
性能问题	203
运用缓存	204
什么是缓存？	204
缓存如何工作	204
TableInfoServer 对象和 CacheSettings 属性	204
MapInfo MapCatalog	206
将空间数据加载到 DBMS	206
手动创建 MapInfo MapCatalog	206
将行添加到 Map Catalog	208
Symbol、Pen、Brush 子句语法	210
每记录样式	212
已知限制 / 要求	212
故障诊断	213

第 12 章：将地图绘制功能增加到应用程序	215
Mapping 命名空间简介	216
基本地图绘制类	216
MapExport	216
Map	217
MapFactory	218
MapLoader	218
MapViewList、MapView	218
MapControl	219
图层	219
FeatureLayer	219
Layers	220
MapLayer	220
UserDrawLayer	220
ObjectThemeLayer	220
GroupLayer	220
图层筛选器	221
IVisibilityConstraint	221
代码示例：动画图层	221
标注	223
LabelLayer	223
LabelSource	223
LabelModifier	224
ILabelSourceFilter	224
LabelProperties	224
生成标注	224
标注优先权	225
标注图层的可选择性	225
代码示例：创建 LabelLayer	226
修饰	227
图例	227
ScaleBar 修饰	228
标题修饰	228

图元样式修饰符	229
FeatureStyleModifier	229
FeatureStyleModifiers.	229
FeatureOverrideStyleModifier.	230
打印地图	230
第 13 章：使用主题和图例	231
主题概述	232
Mapping.Thematics 命名空间.	232
修饰符主题	232
对象主题	233
GraduatedSymbolTheme	234
何时使用分级符号主题	234
PieTheme	235
何时使用饼图主题	235
BarTheme	237
何时使用条形图主题	237
RangedTheme	238
何时使用范围主题	238
范围值的类型	238
RangedLabelTheme.	240
何时使用 RangedLabelTheme 类	240
IndividualValueTheme.	241
何时使用 IndividualValueTheme 类	241
使用自定义位图符号创建 IndividualValueTheme	242
IndividualValueLabelTheme	242
何时使用 IndividualValueLabelTheme 类	242
DotDensityTheme	243
何时使用 DotDensityTheme 类	243
双变量主题地图	244
图例概述	245
主题图例	245
制图图例	247
将图例格式化	248

第 14 章：查找位置	251
查找功能概述	252
查找过程	252
匹配地址号码	254
使用优化边界表匹配	254
查找结果	254
Data.Find 命名空间概述	255
Find	255
FindAddressRange	259
FindCloseMatch	260
FindResult	261
微调查找过程	263
编辑 Mapinfow.abb 文件	263
第 15 章：为地图设置样式	269
MapInfo.Styles 命名空间概述	270
StyleFactory	271
样式说明	272
AreaStyle	272
BitmapPointStyle	272
CompositeStyle	272
SimpleInterior	272
Font	273
FontPointStyle	273
GridStyle	273
RasterStyle	273
Hillshade	273
Inflection	273
SimpleLineStyle	274
BasePointStyle	274
BaseLineStyle	274
BaseInterior	274
StockStyles	274
TextStyle	274
SimpleVectorPointStyle	274

预定义的样式和 StyleRepository 类	275
StyleRepository 类	275
使用样式	276
样式和图层控制	276
创建定制位图样式	276
覆盖样式	277
FeatureOverrideStyleModifiers	277
第 16 章：空间对象和坐标系	279
MapInfo.Geometry 命名空间简介	280
几何体	280
Geometry 对象	281
FeatureGeometry 对象	282
Geometry 对象	287
将 FeatureGeometry 包括在地图中	289
坐标系	290
第 17 章：运用光栅和网格	293
MapInfo.Raster 命名空间概述	294
光栅图像	294
光栅类	295
光栅图像和坐标系	296
光栅图像的限制	296
代码示例：将光栅图像添加到地图	296
光栅句柄	297
配置自定义光栅句柄	298
网格图像	299
网格类	300
代码示例：将网格图像添加到地图	300
代码示例：从网格地图检索数据	301
第 18 章：WMS 和 WFS	303
Web 地图服务	304
设置 WMS 服务器	304
理解 WMS 服务器的特定操作	305
使用 MapXtreme 2005 WMS 客户端	306

使用 WMS 地图图像	308
使用 MapInfo Professional 访问 WMS 服务	309
Web 图元服务	314
设置 WFS 服务器	314
理解 WFS 服务器的特定操作	315
使用 MapXtreme 2005 WFS 客户端	316
使用 MapInfo Professional 访问 WFS 服务	318
第 19 章： Workspace 管理器	323
Workspace 管理器的特性	324
工作空间格式和内容	325
Workspace 管理器菜单命令	325
“文件” 菜单命令	325
“地图” 菜单命令	327
“工具” 菜单命令	330
图层控制	332
图层控件工具	333
图层树	333
图层控件选项卡	334
图层设置	336
主题设置	336
标记设置	337
第 20 章： 使用 GeoDictionary 管理器	341
使用 GeoDictionary 管理器	342
GeoDictionary 管理器中的变化	342
GeoDictionary 管理器的用户界面	342
运行 GeoDictionary 管理器	342
GeoDictionary 文件	346
示例 .dct 文件	346
附录 A： 术语表	349
术语	350

附录 B: MapInfo SQL 参考357
MapInfo SQL 语言概述	358
空间运算符	358
字符串常量	358
引用标识符	359
函数	359
IN/ANY/ALL 运算符和子查询	359
MapInfo SQL 命令	360
运算符	362
运算符优先级	364
函数列表	365
按字母顺序排列的函数列表	370
日期格式化字符串	406
使用日期格式化字符串的示例	407
数字格式化字符串	408
使用数字格式化字符串的示例	409
附录 C: 迁移到 MapXtreme 2005	411
比较 MapXtreme 2005 和 MapX 的对象模型	412
特定对象模型实现的不同之处	412
附录 D: 如何创建和部署 MapXtreme 2005 应用程序	421
创建应用程序	422
构建桌面应用程序	422
生成 Web 应用程序	429
将应用程序打包	431
桌面应用程序的打包	431
Web 应用程序打包	433
部署应用程序	436
部署桌面应用程序	436
部署 Web 应用程序	436
授权分布式应用程序	437
附录 E: 案例研究: 在 Web 应用程序中管理会话状态	439
案例研究: 管理状态	440
应用程序设置	441

附录 F: 定制 MapXtreme 2005	447
可定制的类	448
Engine.CustomProperties	448
搜索	448
FeatureStyleModifier 或 FeatureOverrideStyleModifier	449
UserDrawLayer	450
Windows.Controls	451
工具	453
样式	453
GmlFeatureCollection	454
WorkSpacePersistence 和 WorkSpaceLoader	454
MapXtreme 2005 其它可定制的方面	455
ADO.NET	455
应用程序数据文件的位置	455
查找缩写文件	457
附录 G: 理解 MapInfo 工作空间	459
什么是 MapInfo 工作空间?	460
工作空间的结构	461
Header 部分	461
Connection 部分	461
DataSource Definition 部分	462
Map Definition 部分	464
通过编程方式创建工作空间	466
附录 H: 从 MapXtreme 应用程序进行打印	467
概述	468
了解 MapXtreme 中的打印选项	469
打印大小	469
特殊透明光栅处理	469
特殊透明矢量处理	470
尽可能地以真彩色显示光栅	470
抖动方法	470
特殊多边形孔处理	471
缩放图案	471

直接打印到设备	471
使用增强元文件 (EMF) 打印	471
在应用程序中实现打印功能	472
常规打印技巧与方法	474
打印地图中的图例	474
已知打印问题的解决方案	478
平台独立问题	478
特定平台问题	479
附录 I: 样式查找	483
填充图案	484
了解索引编号结构	484
直线样式	494
矢量符号	495
MapInfo 箭头	495
MapInfo 制图	495
MapInfo 杂类	496
MapInfo 油气	496
MapInfo 遮罩	496
MapInfo 房地产	497
地图符号	497
MapInfo 符号	497
MapInfo 交通	498
MapInfo 天气	498
定制符号	499
附录 J: MapXtreme 2004 Web 控件	501
Web.UI.WebControls 命名空间概述	502
Web 控件	502
Web 控件类型	503
使用 Web 控件	504
MapControl	504
LegendControl	506
LayerControl	506
在用户控件中使用 MapControl	507
Web 工具体系结构	508
Info 工具	508

距离工具	509
创建自定义工具	509
客户端工具示例	515
附录 K: Web 地图服务器示例	517
WMS 配置文件示例	518
附录 L: 坐标系的元素	525
投影及其参数	526
投影	527
投影基准面	531
单位	539
坐标系原点	540
基准面转换	542
有关坐标系和投影的详细信息	542
定制基准面	544
定义定制基准面	544
附录 M: 定义 MapInfo Codespace	549
定义 MapInfo Codespace	550
附录 N: 示例数据的版权	557
索引	569

1

MapXtreme 2005 简介

欢迎开发人员使用 MapInfo 公司最新提供的 .NET 编程技术。为了支持 Microsoft 公司的 Windows .NET 框架，MapXtreme 2005 打造了单一的对象模型，可用于开发或扩展桌面、传统客户机/服务器环境或 Web 方式地图绘制应用程序。

对于那些已经认识到数据可视化和地图绘制有助于制定更好的商业决策、管理资产以及更有效地运作的公司，MapXtreme 2005 是一个功能非常强大的应用程序开发工具。MapXtreme 2005 设计用于需要将地点分析或定义放入桌面、客户机 / 服务器或基于 Web 的产品的公司。MapXtreme 2005 是一个功能强大的分析工具包，可以用来制定关键的商业决策，例如销售办事处的最佳地点、如何最有效的运输产品以及如何管理和保护资产。开发人员可以使用 MapXtreme 2005 来缩短开发时间并提高产品的性能、可靠性和安全性。

本章内容：

- ◆ 概述 *MapXtreme 2005* 2
- ◆ 新增内容 4
- ◆ 发生变化的内容 9
- ◆ 迁移到 *MapXtreme 2005* 11
- ◆ 学习使用 *MapXtreme 2005* 13

概述 MapXtreme 2005

MapXtreme 2005 是 MapInfo Corporation 的主要 Windows 软件开发工具包，具有 .NET 开发经验的开发人员使用该工具可以创建功能强大的位置增强型桌面和客户机 / 服务器应用程序。

开发人员可在这个 SDK 中使用熟悉的 .NET 编程语言开发应用程序，在桌面和 web 部署之间共享和重用代码，使用标准协议访问大量数据源中的数据以及更多其它功能。

这些均可通过 MapXtreme 2005 的对象模型实现，该对象模型是在 Microsoft 的 .NET Framework 上开发的完全可管理代码 API。Framework 的 Common Language Runtime (CLR) 提供了实现简化开发的基础。

MapXtreme 2005 包括以下组件和功能：

- **产品框架：**MapXtreme 2005 对象模型与 .NET 框架 1.1 兼容。有关详细信息，请参阅第 48 页第 4 章中的 *MapXtreme 2005 体系结构*。因为 MapXtreme 基于 Microsoft 的 .NET 框架，所以我们利用 .NET 框架的诸多功能，如 ADO.NET。
- **开发环境工具：**Visual Studio.NET 中提供的大量模板、控件、示例代码和工具有助于开发 Windows 窗体和 ASP.NET 应用程序。您可以利用对象模型将这些组件进行扩展，从而获得更高级的功能。MapXtreme 2004 包括了两个数据管理实用程序，Geodictionary 管理器用于管理将在应用程序中使用的表，而 Workspace 管理器用于管理工作空间以方便用户使用和移植。请参阅第 6 章：*WebControls 命名空间概述* 和第 129 页第 8 章中的 *MapInfo.Windows.Dialogs 命名空间*。
- **强大的地图绘制和分析能力：**地图创建和显示、数据访问、主题图地图绘制、光栅和网格处理、对象处理和表示等等。
- **规模可伸缩的基础结构：**Session 对象池和缓存功能可使 web 应用程序获得出色的性能。通过将信息保存为 MapXtreme 的基于 XML 的工作空间格式来维护会话和用户信息。请参阅第 9 章：*运用核心 MapXtreme 2005 类*。
- **运行时部署：**MapXtreme 2005 使用 Windows Installer 技术（即合并模块），开发人员可以使用该技术来安装或重新发布在已部署的应用程序中使用的运行时组件。请参阅第 34 页第 2 章中的 *部署应用程序*。
- **丰富的文档：**产品文档已作为组件完全集成在 Visual Studio.NET 开发环境中。“MapXtreme 资源”页面将帮助您连接到全部产品资源。可以通过“开始”菜单访问该页面。

主要功能

MapXtreme 2005 可以帮助您方便有效地构建 Windows 窗体或 ASP.NET Web 应用程序。不论是设计应用程序的基本绘制地图功能还是增加基本地图绘制功能以支持现有的应用程序，所使用的框架和工具都是相同的。以下是 MapXtreme 功能的概述：

如果您是 MapInfo 公司地图绘制产品的新用户，请务必参阅[第 3 章：地图绘制概念](#)以获取基础知识的帮助信息。

对于升级到 MapXtreme 2005 的开发人员，请参阅此列表后面的章节以了解本产品中的新增内容和变化。另请参阅[第 11 页中的迁移到 MapXtreme 2005](#)以获取新旧功能对应信息。

功能	目的
表、图层、图元	MapXtreme 2005 中的地图由地理图元（例如点位置、边界和街道网络）组成。图元信息存储在表中并在地图中显示为图层。
数据访问	MapXtreme 2005 支持来自多种数据源的数据，包括空间和非空间 RDBMS、MS Access、dBase 和 ASCII 以及自带的本地 MapInfo Table (.TAB)。所有数据操作都通过 MapInfo.Data 命名空间执行。数据操作包括添加和删除表，插入、更新和删除来自各种数据源的记录。 .NET Dataset Provider 支持：任何 ADO.NET 数据集提供方可被视作 MapInfo.Data 中的 Table，这样就可以使用外部不可制图的数据。
Web 服务	MapXtreme 2005 提供了客户端和 API 用于访问一些常用的 web 服务：地理编码、路径规划、WMS 和 WFS。
选择和搜索	该公共地图绘制操作通过使用属性或空间查询，可以找到与条件相匹配的数据。
主题地图绘制	对数据进行分析是最常使用的一种方法，可以通过主题地图显示可视的关系和基础数据。MapXtreme 2005 支持创建和使用六种主题：范围、单值、分级符号、点密度、饼图和条形图。
标注	MapXtreme 2005 提供了成熟的标注功能，不仅可以使用名称或其它信息（列数据或表达式）对图元进行标注，还可以在标注本身创建范围和单值主题图，用来表示信息而不必仅依赖于文本。
地图样式	标注是其中一种地图样式，可以在地图上以任意方式对地图样式进行控制。样式还指地图图元的颜色、图案、字体、直线样式和符号，MapXtreme 2005 中的许多地方（包括图元、修饰即地图标题、对话框和文本）都使用这些样式。

功能	目的
地理处理和分析	该功能是指从现有图元中产生新的图元，例如将邮政编码边界进行组合从而创建销售区域。该功能还指使用图元的位置坐标了解更多与其它图元关系的信息。例如，在点周围半径 5 英里范围内创建缓冲区以找出落在该缓冲区内的其它点。
投影和坐标系	在二维地图上可以有多种方式来表示地点。了解数据的坐标系使 MapXtreme 2005 可以正确地排列图元，从而获得精确的显示和度量。MapXtreme 2005 支持多种投影和坐标系，并为创建自己的投影和坐标系提供了信息。

新增内容

MapXtreme 学习资源页面

学习资源与 MapXtreme 一起安装，基于浏览器的工具会将 MapXtreme 2005 中所有文档和学习资源集中到同一位置。您可以在安装过程结束时运行学习资源，或通过 MapXtreme 程序组下的“开始”菜单随时运行。

通过学习资源，可以获得对整个产品的简单印象、浏览“如何实现”代码片断表、到示例应用程序的链接，以及连接到 MapInfo 网站上的关键 MapXtreme 位置等。可从“学习资源”访问本《开发人员指南》的 PDF 版本。

务必将有关这个新工具的反馈发送给我们。告诉我们您喜欢或不喜欢的内容以及查找重要信息时的感受。请访问“反馈”页面上的链接。

最佳实践：规划应用程序

新的 MapXtreme 2005 文档包括在构建应用程序前必须考虑的关键因素。信息包括可伸缩性、对象池、状态管理和性能。请参阅第 5 章：应用程序的规划。此外，还提供了使用一个运输 web 示例应用程序的案例研究，用于说明这些最佳实践的概念。请参阅附录 E：案例研究：在 Web 应用程序中管理会话状态。

对象模型文档

MapXtreme 2005 对象模型的概述，全新的彩色文档，显示主要接口和类以及彼此间的关系。该信息对新用户尤为有用，它提供了需要了解的有关对象模型中命名空间的术语和要点，用户可从中获得详细信息。可通过学习资源 > 文档 > 对象模型文档下的“学习资源”浏览器访问产品模型概述的 PDF 版本。要直接访问该 PDF（及本《开发人员指南》），请转到 ..\MapInfo\MapXtreme\6.5\Documentation\PDF。

Web 控件的体系结构

为了更容易使用，我们已重新设计了 MapXtreme 2005 中的 Web 控件。另外，现在所有的 Web 控件均可在框架中运行。

增加了新的 NavigationTool 和 ZoomBarTool。所有 Web 控件都提供了源码，这样就可以对其进行定制，以适应应用程序业务需要。此外也可以分布修改的 Web 控件。源码见 ..\MapInfo\MapXtreme\6.5\Samples\Web。

其它重要更改包括在错误管理方面的改进和对状态管理和事件处理的更改。

有关使用和管理 Web 控件以及从 MapXtreme 2004 迁移到 Web 控件的详细信息，请参阅第 6 章：运用 *MapXtreme 2005 Web 控件*。

地理编码和路径规划客户端

MapXtreme 2005 的地理编码和路径规划客户端具有新的强化功能，允许用户创建更强健的文本。现在，地理编码客户端支持 MapInfo 世界地理编码程序，可以提供覆盖许多国家的城市、邮政编码和街道级别数据。已添加附加地理编码约束和地址级别选项（如地址范围），生成更加用户化、更加精确的候选地理编码。这些约束中包括新增的 CASS 认证，通过该过程可将邮寄地址表标准化，以满足 U.S. Postal Service® (USPS) 对大批邮寄折扣的要求。

现在，路径规划客户端包括计算驾驶时间和驾驶距离等值线的功能，还能够使用瞬态更新重新计算路径，取消或更改道路类型、点或线段的优先权。现在已支持矩阵路径规划，通过它可以查找任意数目起点和终点之间的最短或最快路径。此项功能通常用作分析工具，根据在合理时间长度内救护车、公共安全或消防人员可救助财产/个人的概率确定保险财产或个人的危险性。为支持运输系统类型路径规划，我们已添加了基于时间的路径规划，通过它可以指定开始、停止和结束时间。现在，路径规划客户端包括大量参数选择，通过它可以定制路径分析、驾驶指示和路径几何体。

打印支持

MapXtreme 2005 的新打印支持功能允许用户通过基于 MapXtreme 的应用程序、Workspace 管理器或打印 API 生成高质量的地图和图例打印输出。基于 Microsoft .NET 1.1 Framework 的 MapInfo.Printing 对象模型从 System.Drawing.PrintDocument 类继承。地图和图例可以直接打印到设备或首先生成 Enhanced Metafile (EMF)。此外，可在“打印预览”中查看地图和图例。可以通过 MapPrinting.PrinterSettings (LegendPrinting.PrinterSettings) 和 MapPrinting.PageSettings (LegendPrinting.PageSettings) 并使用 MapInfo.Printing 命名空间更改打印机和页面设置。

当前没有内置的 GUI 支持。MapPrinting.Print、MapPrinting.PrintPreview 和 MapPrinting.PageSetup 使用默认的 .NET 打印对话框。更改为 MapPrintMethod 和 LegendPrintMethod，则 MapPrintSize、Map/Legend.DrawingAttribute 通过 MapPrintDocument 和 LegendPrintDocument API 完成。可控制的绘制属性包括抖动方法（半色调或误差扩散）、缩放位图填充图案、24 位彩色光栅打印以及对具有孔、透明矢量和光栅图层的多边形的特殊处理。注：MapXtreme 不支持半透明光栅图像的打印。

Workspace 管理器已更新为可利用新的打印 API。它使用默认设置，包括打印方法 (EMF)、打印大小（适合于页面）和绘制属性（全部为 true）。

有关 MapXtreme 打印支持的详细信息，请参阅[附录 H：从 MapXtreme 应用程序进行打印](#)。

Shapefile 中的 M 和 Z 值

MapXtreme 2005 现在可读取包含 M 和 Z 值的 ESRI Shapefile。这些值不能使用。

ECW 光栅句柄

MapXtreme 2005 在使用 ECW 光栅句柄时支持多线程。句柄的文件名已更改。现在的名称为 ecw.rhy 且位于 C:\Program Files\Common Files\MapInfo\MapXtreme\6.5\RasterGridHandlers。（先前文件名为 ecw.rhl。）光栅句柄的扩展名指示 MapXtreme 2005 使用该文件的顺序。请参阅[第 17 章：光栅句柄](#)。

瞬态图元支持

现在对象模型中支持瞬态 FeatureGeometry 对象。MapInfo.Geometry.FeatureGeometry.IsTransient 可用于确定 FeatureGeometry 是否为瞬态的 Boolean 属性。

通过某些数据访问方法获得的 FeatureGeometry 对象可能为瞬态。它们是短时间存在于可能回收的内存中的对象，用于后续的获取操作。使用瞬态 Geometry 对象可以提高性能。如果需要超越对象的短暂生存期，则应该生成对象副本。

新 Transient 图元功能的结果是更新 SearchResultProcessor 类使其支持此项功能。现在该基类返回瞬态图元（您可为该基类自定义搜索结果的后处理程序）。不过，请注意返回的任何瞬态图元只在当前图元上有效。一旦移至下一图元，则先前图元无效。

性能测试

MapXtreme 2005 已执行了大量测试，已确定在应用程序中使用池对象时的性能效果。此外还包括了多层应用程序，这些应用程序访问远程服务并分配池对象以获得最高性能。详细讨论请参阅第 74 页第 5 章中的优化应用程序。

示例应用程序和代码示例

MapXtreme 2005 向产品中添加了更多的示例应用程序和代码片断，以帮助您了解并且最有效地利用对象模型。找到所有示例的最佳方式是使用“程序”菜单下的“学习资源”浏览器。在浏览器的“学习资源”部分单击“示例应用程序”链接。

在可导航的浏览器中会显示代码片断的“如何实现 ...”，用户可通过“学习资源”浏览器的“文档”页面打开该功能。这些片断是基本构建块，用户可在其基础上构建桌面或 web 应用程序。

IN/ANY/ALL 运算符和子查询

下面是与数据访问相关的新的和增强的功能。

- 现在提供带有文字列表和子查询的 IN、ANY 和 ALL 运算符。例如：

```
Select ... From T Where X IN (2, 6, 12)Select ... From T Where X IN  
(Select X From U Where B > 50)
```

- 现在提供相关联的子查询，其中子查询引用来自外部查询的列（或表达式）。例如：

```
Select ... From T Where X IN (Select X From U Where B = T.A)
```

- 空间 OR 列表已进行优化。例如：

```
{Select ... From T Where Obj Within @Var1 OR Obj Within @Var2 OR  
Obj Within @Var3}
```

StyleFactory 类

MapInfo.Styles.StyleFactory 中包含了一个新的类，它允许用户将画笔、画刷和符号子句从 MapBasic 字符串转换到 CompositeStyle。

I Session.Reload 方法

I.Session.Reload 是一种新方法，可以清除 Session 的状态并重新加载先前在应用程序配置文件中配置的工作空间。 I.Session.Reload 替换 I.Session.Reset。

要重置会话的状态而无需重新加载预装的 workspace，请使用 I.Session.Clear 方法。

永久性

名为 CenterAndScale 的新枚举值已增加到 MapInfo.Persistence.AreaOfInterestPreference。该值更倾向于表示应如何在工作空间中表示每个地图的重点区域。 ScaleAndZoom 值已很少使用。

自定义位图符号 MSM

MapXtreme 2005 包括合并模块，使得可以更加容易地在部署项目中包括自定义符号。只需在项目中包括 MapInfoCustSymb.msm 即可。这会将符号安装到 Program Files\Common Files\MapInfo\MapXtreme\6.5\CustSymb 文件夹中。要为部署定位其他位置的自定义符号，请参阅第 455 页附录 F 中的应用程序数据文件的位置。

许可证文件工具

MapXtreme 将许可证工具 MILicenseNodeID.exe 和该工具的库安装到 Program Files\Common Files\MapInfo\MapXtreme\6.5\License Tool 文件夹中。当您需要永久使用许可证用于开发和部署 MapXtreme 应用程序时，请运行此工具。将生成的节点 ID 文本文件提交给 MapInfo Corporation 以申请许可证。

如果正部署 web 应用程序或正使用软件防复制版本的 MapXtreme 2005 开发和部署 web 或桌面应用程序，则必须在项目中提供该工具。节点 ID 工具包括在合并模块 MapInfoMXTConfig_6.5.msm 中。“核心引擎”合并模块需要该 msm，因此许可证工具目前将随您的应用程序一同部署。请参阅第 437 页附录 D 中的授权分布式应用程序。

发生变化的内容

WinForms 的 LayerControl

MapXtreme 2005 增强了 LayerControl 的多项功能。

桌面 LayerControl 支持新的拖放功能。现在，可以将 Style Override 节点拖放到另一图层，以达到将其复制到该图层的目的。图层必须为可比较类型（即，无法用拖放的光栅类型图层覆盖矢量图层）。

现在，“图层控制”中显示的“工具提示”可以显示有关每个图层基础的信息，而不是存在于所有地图图层的一个一般字符串。

以前，在 LayerControl 中右键单击树中某一节点时实际上未选中该节点。（事实上，表面上看是“临时选中”，却不是真正选中。）。这样可能会误导用户。现在，右键单击图层树中某一节点时，会选中该节点并将后续操作应用到该所选图层。这才是预期行为并同时提高产品的可用性。

MultiPolygon 行为中的 Point

MapXtreme 2005 在具有重叠 Polygon 的 MultiPolygon 中提供了优化的 Point 选择集。以前，Contains 查询会失败，原因是由于应用程序将重叠的 Polygon 视为一个合计对象。现在，当 MapXtreme 2005 确定 Point 在任何 Polygon 之内时，会满足 MultiPolygon 的 Contains 查询并结束处理过程。

这种情况同样出现在 MultiPolygon 选择集。以前，如果尝试在重叠区域内选择 MultiPolygon，则应用程序会将 Point 选择集视为位于 MultiPolygon 外部并且不选择。现在，MapXtreme 2005 将该 Point 视为位于 MultiPolygon 之内并正确选择。

这些更改涉及到更多关于 MapXtreme 在 Polygon 测试中如何执行 Point 的问题。对于由单一外部边界 (Ring) 和任何数目内部边界组成的 Polygon，在算法上没有任何更改。但是对于 MultiPolygon，如果 Point 位于 Polygon 的重叠区域，则先前版本的 MapXtreme 会产生错误结果。为纠正该问题，现在 MapXtreme 2005 将 MultiPolygon 视为一系列 Polygon（单一外部 Ring 边界和其中所有内部 Ring 边界）。它独立检查每个 Polygon，以确定 Point 是否位于 Polygon 之内。如果 Point 位于任何 Polygon 之内，则认为它位于 MultiPolygon 之内。

由于 MapXtreme 2005 可在了解到 Point 位于 Polygon 之内时结束处理过程，因此也可以提高性能。

搜索方法

SearchWithinDistance 方法

现在，SearchInfoFactory.SearchWithinDistance 使用 Geometry.Distance() 方法（而不是在给定距离内缓冲 Geometry），然后搜索缓冲区内的行。

以前，MapXtreme 2005 检查对象是否位于缓冲区内。因此，对象必须完全位于距离之内。（即，如果对象上最远的点大于距离值，则不检索该对象。）这与询问对象是否与缓冲区相交类似，胜于询问是否在给定距离内。

现在，MapXtreme 2005 提供了更加精确的 SearchWithinDistance 结果，而不再使用缓冲方法的近似值。

对象处理

SelfIntersections

对于 MapXtreme 2005，SelfIntersects 属性和 SelfIntersection 方法现在在 Geometry 类中定义。以前，这些方法只适用于 LineString、Curve、MultiCurve、Ring、Polygon 和 MultiPolygon 类。

现在，SelfIntersection 返回 FeatureGeometry 而不是 MultiPoint。由于该方法现在返回 MultiPoint、MultiPolygon 或包含 MultiPoint 和 MultiPolygon 的 FeatureGeometryCollection，因此此项更改十分必要。

Geometry 编辑器的新方法

Geometry API 中增加了新的 Get 方法，用于替换由 Geometry.GeometryEditor 和相关 Geometry 类的属性提供的功能。

新的 Geometry.GetGeometryEditor() 返回对实现 IGeometryEdit 接口的类的引用。

旧属性将产生编译错误并抛出 System.NotSupportedException。将有消息指示应该使用哪种新方法替换相应属性。

对象编辑接口

Edit 接口增加了新方法以获得基础对象。例如，有了这些新模块，就能够知道可编辑 Curve 是否为 MultiCurve 的一部分，做法是调用：

```
curveEdit.Curve.IsGeometryPart
```

Curve 是 ICurveEdit 上的新属性，返回基础 Curve 对象。

OCI 连接对话框

MapInfo.Data.DBMSConnectionCollection 类现在通过订阅 ConnectionFailedEvent 事件句柄支持 ConnectionFailed 事件。激发后，该事件会显示 OCILoginDlg 以使用户可以更改登录信息并再次重试到 Oracle 数据库的连接。该句柄特定于 MapXtreme 2005 OCI 工具包。还可通过 MapInfo.Windows.Dialogs 命名空间中的 LoadMapWizard 类获得该事件。

迁移到 MapXtreme 2005

下表是 MapX 和 MapXtreme 先前发行版的功能与本发行版 MapXtreme 2005 中对等功能的对照。使用产品的任何新体系结构时，注意对等的功能可能不精确。使用该列表中右边一列的 MapXtreme 2005 主题可以了解进一步的详细信息，这些信息位于本《开发人员指南》和联机帮助以及对象模型的其它章节中。MapX 对象模型和 MapXtreme 2005 中等价功能的详细列表位于第 411 页附录 C 中的 *迁移到 MapXtreme 2005*。

MapX5.0 /MapXtreme3.0	MapXtreme 2004/MapXtreme 2005
Map 对象	Map 类：保存 Layers 集合。 MapControl：查看窗体上的地图的方法。 <i>MapInfo.Mapping</i> 命名空间
MapXBroker	Session 类：所有基于 MapXtreme 2005 的应用程序的起点。 相关主题：MICommand, Catalog, Pooling。 <i>MapInfo.Engine</i> 命名空间
以图层为中心的模型	以表为中心的模型 相关主题：表元数据 (TableInfo 类)、Feature 类、Column 类 (MI_Geometry、MI_Style 和 MI_Key) 和 MapInfo ADO.NET 数据提供方。 <i>MapInfo.Data</i> 命名空间
数据集和数据绑定	使用 Table.AddColumn() 方法将临时列添加到 Table 中。 相关主题：Geodictionary 管理器 <i>MapInfo.Data</i> 命名空间
Geosets GeoSet Manager	工作空间 (.MWS)：新的 XML 格式。GeoSet 仍可读取。 相关主题：Workspace 管理器。 <i>MapInfo.Persistence</i> 命名空间

MapX.0 /MapXtreme3.0	MapXtreme 2004/MapXtreme 2005
注释	修饰：Legend、Title 和 Scalebar，或单一地图中其它一些类似用户定义对象。 <i>MapInfo.Mapping</i> 命名空间
主题地图绘制	与主题地图类型相同。主题不再是图层。 相关主题：ModifierTheme（分级符号、饼图和条形图主题）和对象主题（范围、单值、点密度主题）。 <i>MapInfo.Mapping.Thematic</i> 命名空间
图元图层和图元集合	FeatureGeometry：全部几何体现在都是对象。其中包括 point、multipoint、curve、multicurve、polygon、multipolygon、ring。 几何体包括矩形、有圆角的矩形、椭圆、旧式的弧线和旧式的文本。 相关主题：CoordSysFactory（已注册的坐标系）和对象处理（请参阅下面的 FeatureProcessor） <i>MapInfo.Geometry</i> 命名空间
FeatureFactory	FeatureProcessor 类：对象处理 Buffer、Combine、Intersection 和 ConvexHull。 相关主题：FeatureGeometry 类中的 Difference（先前的 Erase）。 <i>MapInfo.Geometry</i> 命名空间
工具	新工具：SelectRegion：可以指定给鼠标按键并配置鼠标滚轴进行缩放。 <i>MapInfo.Tools</i> 命名空间
光栅和网格图像	新的表结构、RasterImageInfo 和 GridImageInfo。 相关主题：可控制的样式：亮度、对比度、颜色 / 灰度、半透明度和单色透明度。 <i>MapInfo.Raster</i> 命名空间
标注对象和标注集合	LabelLayer：允许对标注和图层进行单独排序。LabelSource：来自用于进行标注的数据源的信息。 <i>MapInfo.Mapping</i> 命名空间
Selection 对象	Selection 类：连接到表的图元列表的多图元集合。 <i>MapInfo.Engine</i> 命名空间
Spatial Server 连接	MI ADO.NET 数据提供方，MapInfo SQL 语言。 <i>MapInfo.Data</i> 命名空间

MapX5.0 /MapXtreme3.0	MapXtreme 2004/MapXtreme 2005
Style 对象	<p>Style 类：新的对象模型。Style 现在成为对象，不再是其它对象的属性，信息存储在类 MI_Style 中。</p> <p>相关主题：FeatureStyleModifiers 和 FeatureOverrideStyleModifiers 位于 Mapping 命名空间中。</p> <p><i>MapInfo.Styles</i> 命名空间</p>

学习使用 MapXtreme 2005

MapXtreme 2005 为入门和经验丰富的 .NET 开发人员提供了大量知识。本节描述了多种支持方式，使用户可以迅速进入角色，将对开发日程表的影响降至最低。

支持资源

MapInfo 现在继续转向 .NET 环境，承诺支持新的 MapInfo 开发人员和原来的长期客户。我们提供了多种工具和以下资源，可以帮助您进行转换或在 Visual Studio .Net 环境中迅速展开工作。

MapXtreme 2005 学习资源页面

安装 MapXtreme 2005 后可以通过“开始”菜单访问，“学习资源”页面集中了关于 MapXtreme 2005 的各类信息，其中包括最佳开发实践、代码示例、到 MapInfo 网站上所有文档和在线资源的链接等。访问该页面能够使您逐步熟悉 MapXtreme 2005 并帮助您继续开发适应业务需要的地图绘制应用程序。

文档和帮助系统

《MapXtreme 2005 开发人员指南》概述了 MapXtreme 2005 开发环境和命名空间。Visual Studio .NET 集成的帮助系统提供了更多特定的 API 级别的信息。用户需要使用这些工具开发与 MapInfo 强大的地图绘制组件集成的 Windows 桌面和基于 Web 的应用程序。请将对本文档的意见发送至 documentation@mapinfo.com。

注： 上述电子邮件地址不适用于特定于软件的问题，以及有关包含在本文档中的主题的问题。请将这种问题发送至“技术支持”（见下文）。

如果您刚开始使用 MapInfo 地图绘制产品，则一定要阅读第 3 章：地图绘制概念。

技术支持

MapInfo Corporation 为 MapInfo 软件产品的用户提供了强大的技术支持。我们的“技术支持”部门为已注册的 MapInfo 软件用户提供技术上的帮助 – 因此不需要精通我们产品的一方面面就可以获得所需的结果。有关技术支持的信息请访问 MapInfo Web 网站

www.mapinfo.com。

2

入门指南

本章提供了首次安装、配置和部署 MapXtreme 2005 应用程序需要的所有信息。

本章内容：

◆ 安装	16
◆ 安装类型	17
◆ <i>SDK 和运行时许可证</i>	18
◆ 安装之前	19
◆ 将 <i>MapXtreme 2005</i> 安装到系统中	20
◆ 升级到 <i>MapXtreme 2005</i>	25
◆ 在 <i>Visual Studio .NET</i> 中创建应用程序	27
◆ 构建 <i>ASP.NET Web</i> 应用程序但不使用模板	33
◆ 部署应用程序	34

安装

以下章节说明了 MapXtreme 2005 的系统要求和安装步骤。

系统要求

经 MapInfo 公司测试，以下操作系统支持 MapXtreme 2005：

操作系统*	<ul style="list-style-type: none">Windows 2000 Professional SP 4Windows 2000 Server SP 4Windows 2000 Advanced Server SP 4Windows XP Professional SP 2Windows 2003 Server Standard Edition SP1Windows 2003 Web Edition SP1Windows 2003 Server Enterprise Edition SP1
开发框架支持	<ul style="list-style-type: none">Microsoft .NET Framework 1.1 和 SP1Visual Studio .NET 2003 (C# 和 VB)VisualBasic.NET 2003

* 仅适用于 32 位系统

其它软件要求包括：

- MDAC 2.7 或更新版本 (MDAC 2.8 位于 MapXtreme 2005 产品光盘。)
- IIS 5.0 或 IIS 6.0

最低系统要求

内存	Windows Server 2003: 256 MB 内存 Windows Server 2003 Web: 256 MB 内存 Windows XP: 256 MB 内存 Windows 2000: 256 MB 内存 Windows 2000 Server: 256 MB 内存 Windows 2000 Advanced Server: 256 MB 内存
处理器	600-MHz Pentium III 级处理器
显卡	支持至少 256 色的显卡

支持的数据库包括：

- IUS 9.3
- IUS 9.4
- Microsoft Access 97
- Microsoft Access 2000
- Oracle 9i
- Oracle 9ir2
- Oracle 10G
- Microsoft SQL Server 2000
- MapInfo SpatialWare® 4.8.1 for Microsoft SQL Server 2000

安装类型

MapXtreme 2005 提供了两种安装类型：一种是开发 (SDK)，一种是部署（运行时）。每种类型都可以从产品光盘浏览器中选择。

开发 (SDK) 安装

“开发安装”在计算机上安装 MapXtreme 2005 软件开发工具包 (SDK)。选择此选项，用于开发桌面和 web 应用程序。安装后，该 SDK 自动集成到 Microsoft Visual Studio .NET IDE 并与 .NET Framework 1.1 共同使用。SDK 提供了用于简化开发工作的 C# 和 VB 应用程序模板。

有关如何安装 SDK 的指示，请参阅第 20 页中的将 *MapXtreme 2005* 安装到系统中。也可从安装对话框上的“帮助”按钮获得指示。

部署（运行时）安装

“部署安装”选项安装“位置运行时环境”，它放置 MXTRuntimeNCP.exe（或软件防复制保护版本的 MapXtreme 2005 的 MXTRuntimeSCP.exe）中。有关安装运行时安装程序的指示，请参阅第 34 页中的用运行时安装程序部署。

SDK 和运行时许可证

所有 MapXtreme 2005 SDK 或运行时安装都需要许可证文件 (.LIC)。某些许可证在安装中提供，另外一些则必须从 MapInfo Corporation 获得。以下主题描述了许可证类型和如何从 MapInfo 获得许可证。

试用许可证

试用许可证 (MapXtremeTrial.lic) 在 SDK 和 MapXtreme 2005 的评估版本中提供。该许可证在安装之日起 60 天后过期。使用试用许可证开发的应用程序会在地图上显示水印。

SDK 许可证

开发人员可通过 SDK 许可证创建桌面和 web 应用程序。

对于安装没有复制保护的软件开发工具包 (NCP SDK) 的用户，您将自动获得永久 SDK 许可证 (MapXtremeSDK.LIC) 以及试用许可证。有了永久许可证后只要愿意，您就可以一直使用 SDK。不再需要其他 SDK 许可证。

对于安装有复制保护 SDK (SCP SDK) 的用户，只能收到试用许可证。要永久使用 SDK 且不希望在每张地图上都出现水印，您必须从 MapInfo Corporation 获得永久的 SDK 许可证。
请参阅第 19 页中的如何获得许可证？。

运行时许可证

所有运行时桌面和 web 安装都需要运行时许可证。

- 对于使用 NCP 版 MapXtreme 2005 构建的桌面运行时部署，该许可证可在多个安装中使用。
- 对于使用 SCP 版 MapXtreme 2005 构建的桌面运行时部署，每个安装都需要一个许可证。
- 对于使用 NCP 或 SCP 版 MapXtreme 2005 构建的 web 运行时部署，每个安装都需要一个许可证。

有关在分布中打包许可证的详细信息，请参阅第 437 页附录 D 中的授权分布式应用程序。

如何获得许可证？

要获得永久使用许可证，请运行 MILicenseNodeID.exe 以生成惟一的机器特定硬件 ID，此 ID 包含在名为 MILicenseNodeID.txt 的文本文件中。MapInfo Corporation 使用此信息为您的特定安装生成自定义许可证（SCP SDK、SCP Desktop 运行时或任何 Web 运行时）。评估版用户需要购买产品后才能收到永久许可证。

有关获得许可证的完整详细信息，请参阅产品包装盒中或安装在以下位置的“软件复制保护声明”：C:\Program Files\MapInfo\MapXtreme\6.x\Documentation\PDF\

SoftwareCopyProtectionNotice_NCP.pdf。请根据声明所提供的您所在地区的相应电子邮件或传真号联系 MapInfo Corporation（或其代表）。

收到永久许可证后，请在首次使用已部署应用程序前将该许可证复制到 C:\Program Files\Common Files\MapInfo\MapXtreme\6.x 文件夹中。

安装之前

下文是安装 MapXtreme 2005 之前需要了解的内容。

管理员特权

要安装 MapXtreme 2005，您必须是计算机的管理员或当前用户是管理员组的成员。这既适用于 SDK 安装也适用于 Runtime 安装。

临时目录和其它资源的权限

对于部署的 web 应用程序，如果使用集成安全系统，则请确保访问该站点的所有用户都具有访问临时目录及任何其它资源的权限。MapXtreme 2005 在 ASP.NET 处理空间中执行，该处理执行从 IIS 进程传入的安全标记。必须将权限授予任何可能登录并访问该目录的用户。如果使用匿名访问，则必须将访问临时目录的权限授予 IUSR_LocalMachineName 系统帐户。MapXtreme 2005 会从当前的 TEMP 环境设置中获得临时目录。

首先安装 Visual Studio .NET 2003

建议在安装 MapXtreme 2005 之前先安装 Visual Studio .NET 2003。如果尚未安装 Visual Studio.NET 2003 或 Microsoft .NET Framework，则安装程序将提示安装 Microsoft .NET Framework，并在安装 MapXtreme 2005 之前先安装 Microsoft .NET Framework。

注： 如果尚未安装 Visual Studio .NET 2003，则不会安装某些样本模板。

.NET Framework SP1 安装

MapXtreme 2005 需要具有 Service Pack 1 (SP1) 的 .NET Framework 1.1。MapXtreme 安装程序将进行检测，如果没有找到服务包，安装程序将弹出“是/否”对话框提示您进行安装。确保单击“是”。如果单击“否”，则安装将失败。

Windows 2000 Post-Service Pack 4 (SP4) COM+ 1.0 rollup package 27

如果系统为 Windows 2000，且要在基于 Web 的应用程序中使用 MapXtreme 2005，则必须获得 Microsoft Windows 2000 Post-Service Pack 4 (SP4) COM+ 1.0 rollup package 27。有关详细信息请参阅 Microsoft Knowledge Base Article - 822618，位于

<http://support.microsoft.com/?id=822618>。

其它安装特性

MapXtreme 2005 光盘提供了用户可以遵循的联机安装指示。也可以在安装过程中的安装对话框上通过“帮助”按钮访问这些指示。

MapXtreme 2005 提供了大量与各个世界位置有关的样本数据。要安装这些数据，请选择“从光盘浏览器安装样本数据”。通过选择“定制”选项，可以控制要安装的数据量。

“完全”选项（默认）将在 ..\MapInfo\MapXtreme\6.5\Samples\Data 下安装大约 300 MB 世界数据集。

注： 无需运行该数据安装程序就可以使用 MapXtreme 2005 包含的示例应用程序。为实现该目的，会将基本示例数据自动安装到 \Data 文件夹。

MapXtreme 2005 光盘浏览器还提供了到 PDF 版《开发人员指南》的链接。

将 MapXtreme 2005 安装到系统中

要安装 MapXtreme 2005：

1. 将 MapXtreme 2005 光盘插入光盘驱动器。
2. 在光盘浏览器主页面，单击“安装”。“安装说明”页面显示。
3. 选择“开发安装”以安装 SDK，或选择“部署安装”以安装运行时版本。

注： “部署”安装需要运行时许可证。请参阅第 18 页中的 *SDK 和运行时许可证*。

4. 选择“安装 SDK”或“安装运行时”。在“欢迎”对话框中，单击“下一步”继续。对于部署安装，则跳到步骤 10。

注： 也可以查看安装指示并从该页面安装样本数据。

5. 在“安装程序欢迎”对话框中，阅读面板中的信息并单击**下一步**以继续。



6. 选择接受“许可证协议”。单击**下一步**。“用户信息”对话框显示。



7. 在“用户信息”对话框中，将用户名和单位输入适当的字段。单击**下一步**。“安装类型”对话框显示。



8. 在“安装类型”对话框，选择**完整安装或自定义**。如果要选择安装的特性或不安装到默认位置 (C:\Program Files\MapInfo\MapXtreme\6.x)，则选择**自定义**。单击**下一步**。如果选择**完整安装**则继续**步骤 10**。



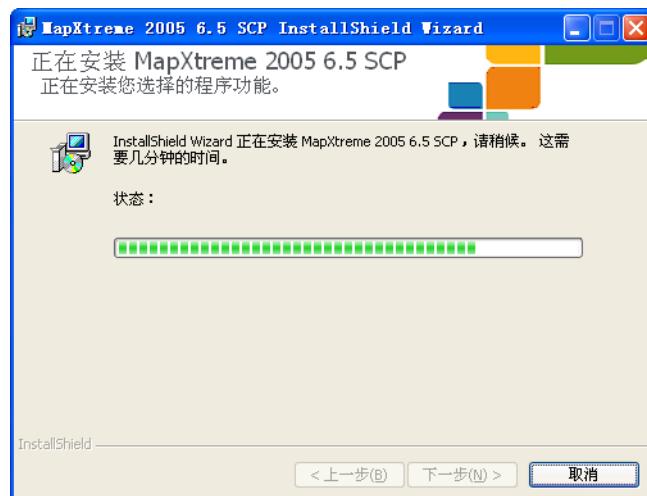
9. 如果在上一步中选择了“自定义”，则选择要安装的组件或单击**更改**按钮来指定新的安装路径。单击**下一步**。



10. 在“已做好安装程序的准备”对话框中，单击**安装**。



11. “正在安装 MapXtreme 2005”对话框启动。



12. 如果不想查看 MapXtreme 2005 学习资源页面，请在“InstallShield Wizard 完成”对话框中取消选中**启动学习资源**复选框，然后单击**完成**退出软件安装程序。

可以随时通过 Windows “开始” 菜单访问 MapXtreme 2005 学习资源页面。



升级到 MapXtreme 2005

MapXtreme 2005 的升级将安装到自身目录下。它不会覆盖先前版本。这样就可以同时安装有不同发行版本的产品。

并存安装需要您将原许可证文件复制到最新安装的位置

C:\Program Files\Common Files\MapInfo\MapXtreme\6.5 (以 v6.5 为例)。

注意，使用并存安装时，必须在打开不同类型项目时关闭 Visual Studio。对于 web 应用程序，您需要重启 IIS (在控制台窗口中或从“开始 > 运行”菜单选项运行 iisreset.exe 命令)。

更新现有的桌面应用程序

使用先前发行版本 MapXtreme (MapXtreme 2004 v6.0、6.1 或 6.2) 创建的桌面应用程序可重新编译。控件可能需要重新增加到窗体中。

或者，可以将应用程序重定向为使用新的程序集，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。请参阅[第 26 页中的重定向程序集版本](#)。

更新现有的 Web 应用程序

对于使用先前发行版本 MapXtreme (MapXtreme 2004 v6.0、6.1 或 6.2) 创建的 web 应用程序，可以将应用程序重定向为使用新的程序集，而不必重新进行编译。请参阅[第 26 页中的重定向程序集版本](#)。

此外，可以将项目版本号从旧的版本号更改为当前版本。使用 4.0.0.xxx 的格式，其中 xxx 为新发行版本的内部版本号。要确定任何已安装版本的内部版本号，请选择开始 > 运行，然后输入 **assembly**。

要重新编译 web 应用程序以使用 MapXtreme 2005 的最新发行版本：

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 = "hklm\mapinfo.net" 更改为 AssemblyFolderKey = "hklm\mapinfo.net 6.5"，其中 6.5 为当前发行版本。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\6.5\MapInfo.CoreTypes.dll"
AssemblyFolderKey = "hklm\mapinfo.net"
/>>
- <Reference
Name = "MapInfo.CoreTypes"
AssemblyName = "MapInfo.CoreTypes"
HintPath = "..\..\..\Program Files\Common
Files\MapInfo\MapXtreme\6.5\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\6.5\MapInfo.CoreTypes.dll"
AssemblyFolderKey = "hklm\mapinfo.net 6.5"
/>>

4. 在文本编辑器中打开 WebForm1.aspx。将任何版本引用更改为 4.0.0.xxx，其中 350 为当前发行版本的内部版本号。
5. 打开 web.config 并更改对 4.0.0.xxx 的任何引用。
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

本节描述了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告知应用程序查找已引用程序集的升级版本。Microsoft 为完成这项功能提供了多种不同机制。

有关更多详细信息，请参阅 <http://www.msdn.com> 并搜索 "Redirecting Assembly Versions" 以定位相应主题。

重定向程序集版本而不更改现有应用程序

有多种用于重定向程序集版本的方法。

应用程序配置文件

推荐用户使用应用程序配置文件来实现程序集的重定向。

注： 配置文件必须与应用程序位于同一目录并以应用程序命名。例如，*myApp.exe* 的配置文件必须命名为 *myApp.exe.config*。可以使用由 Microsoft 提供的配置工具实现此功能。

可以使用控制面板管理工具下的 Microsoft .NET Framework 1.1 配置工具配置应用程序的重定向设置。该工具将创建和 / 或操纵应用程序的应用程序配置文件。

应用程序配置文件将重载发布程序策略文件中的设置。

发布程序策略文件

包含重定向设置的发布程序策略文件可与程序集一起安装在 GAC 中。不过不支持 MapInfo 配置。

计算机配置文件

可以使用控制面板管理工具下的 Microsoft .NET Framework 1.1 配置工具配置程序集的重定向设置。该工具将操纵计算机配置文件，该文件位于：

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

在此处指定重定向设置将使所有引用程序集的应用程序使用升级版本。由于计算机配置文件将重载应用程序配置文件和发布程序策略文件的设置，因此应小心使用这个重定向方法。

在 Visual Studio .NET 中创建应用程序

使用 MapXtreme 2005，很容易就可以将地图添加到应用程序。MapInfo 提供了 Visual Basic.NET 和 Visual C# 项目模板，允许创建简单的地图绘制应用程序而无需编写任何代码。

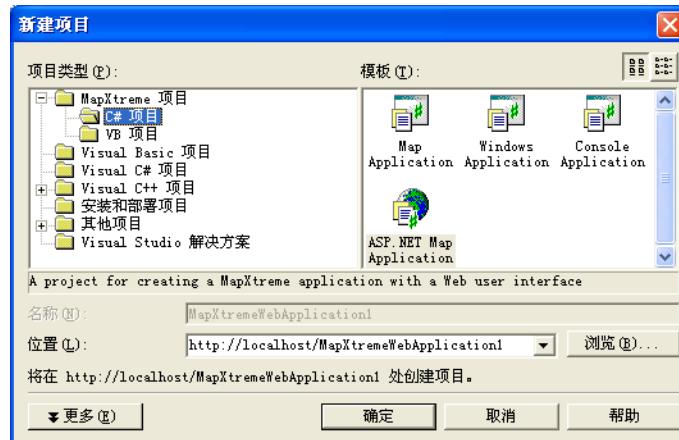
此外，还提供了可供研究的示例应用程序和 web 应用程序，用于实验并适应自身情况。有关按步骤说明如何使用这些示例应用程序的教程，请参阅附录 D：如何创建和部署 *MapXtreme 2005* 应用程序。

以下步骤概述了生成简单桌面地图绘制应用程序的步骤。有关创建 web 应用程序的步骤，请参阅第 30 页中的 *ASP.NET Web 应用程序*。

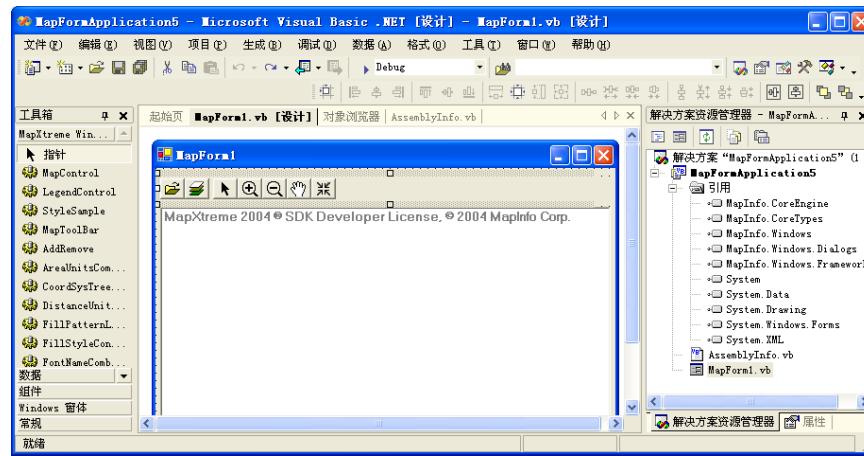
地图应用程序

注：该示例是特定于 Visual Basic.NET 的示例。要创建 Visual C# 地图应用程序，在以下步骤中用 Visual C# 替代 Visual Basic。

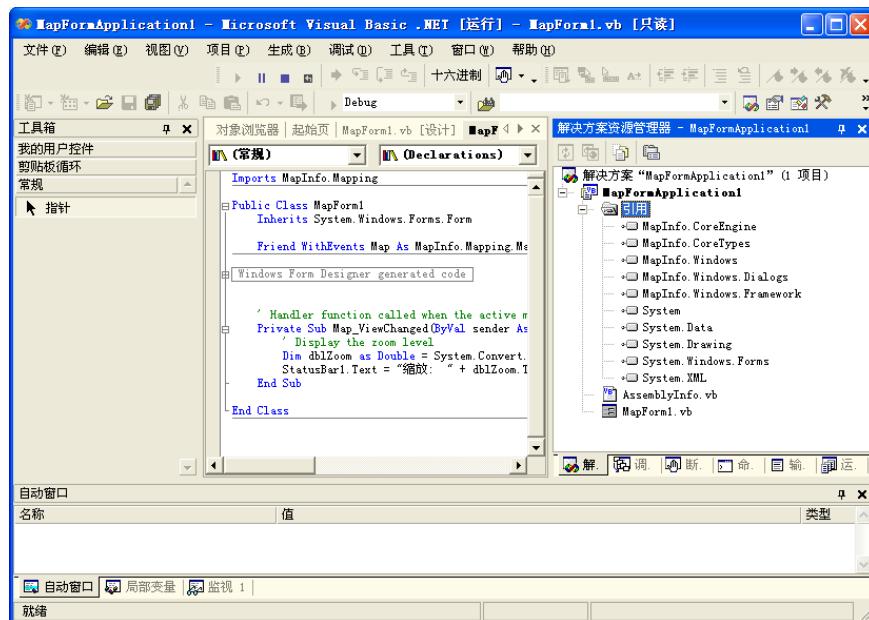
1. 从 Visual Studio .NET **文件**菜单，指向**新建**然后单击**项目**。“新建项目”对话框显示。
2. 在“新建项目”对话框的“项目类型”框架中（位于**MAPXTREME 项目**文件夹下），单击**VB 项目**。
3. 在“新建项目”对话框的“模板”框架中选择**MAP APPLICATION**。



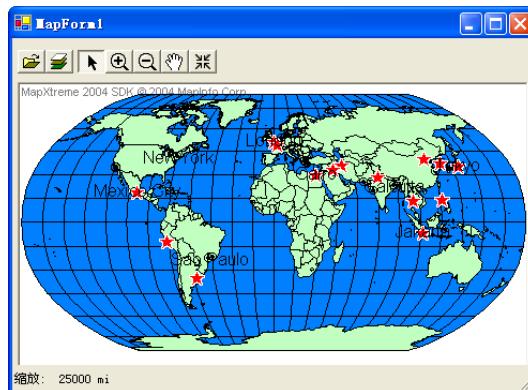
4. 选择适当的名称和位置，然后单击**确定**。MapXtreme 2005 创建应用程序并显示 MapForm1.vb [设计]。



5. 在**视图**菜单中，单击**代码**检查编写用于创建应用程序的代码。



6. 在**调试**菜单，单击**启动**运行应用程序。
7. 单击**打开表**图标并加载数据。示例数据的默认位置是 C:\Program Files\MapInfo\MapXtreme\6.x\Samples\Data。



8. 可以像使用任何其它 MapInfo 地图绘制应用程序一样，使用工具栏的控件来操控地图。

ASP.NET Web 应用程序

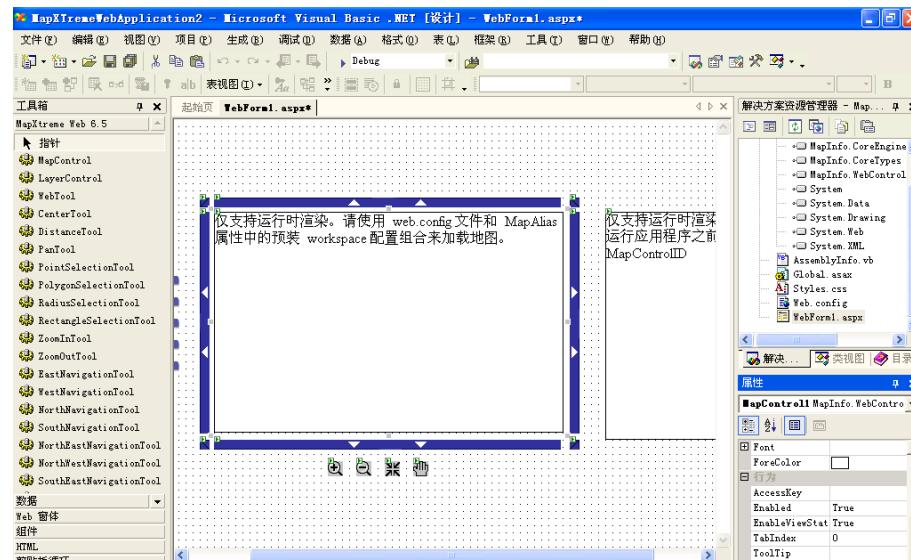
以下步骤概述了生成简单的 ASP.NET Web 应用程序的步骤。有关编制 web 应用程序时的种种事项及其详细讨论，请参阅第 60 页第 5 章中的规划 Web 应用程序。

注：该示例是特定于 Visual Basic.NET 的示例。要创建 Visual C# ASP.NET 地图应用程序，在以下步骤中用 Visual C# 替代 Visual Basic。

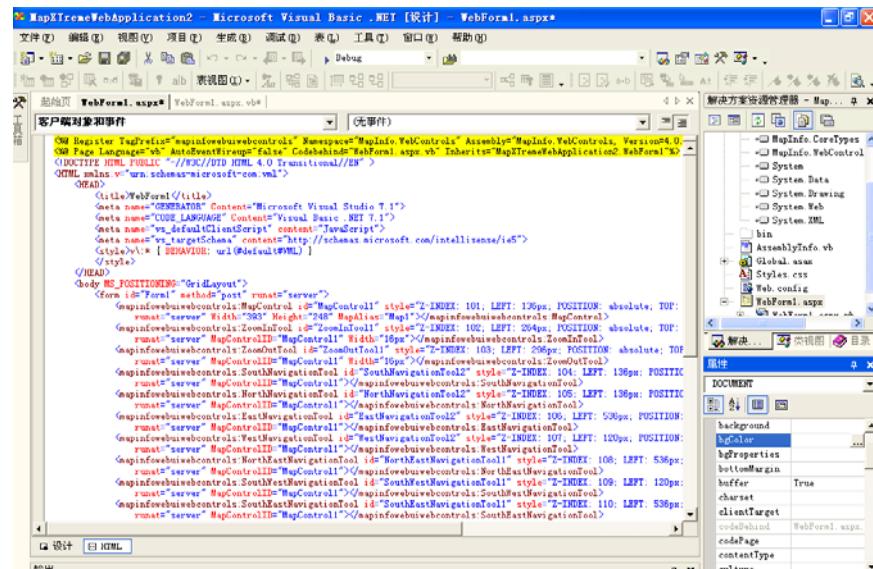
1. 从 Visual Studio .NET **文件**菜单，指向**新建**然后单击**项目**。“新建项目”对话框显示。
2. 在“新建项目”对话框的“项目类型”框架中（位于**MAPXTREME 项目**文件夹下），**单击 VB 项目**。
3. 在“新建项目”对话框的“模板”框架中选择**ASP.NET 地图应用程序**。



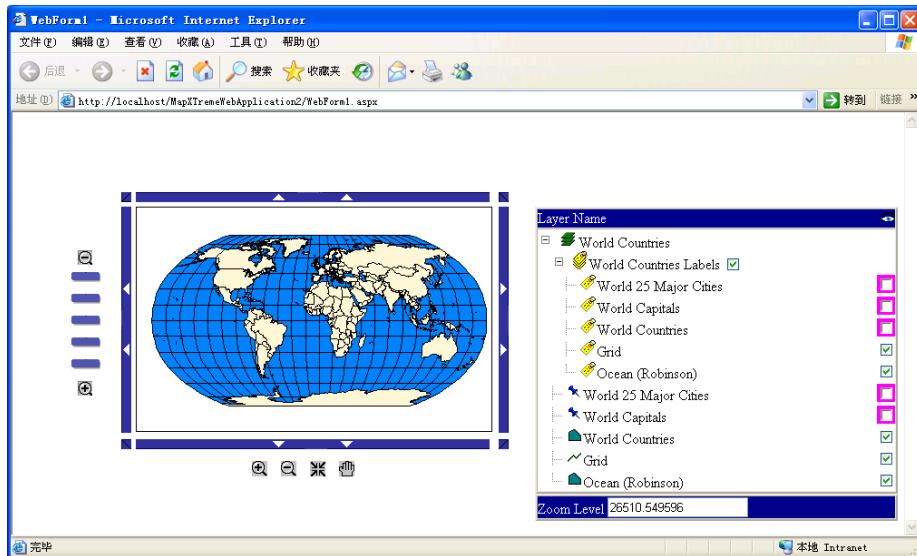
4. 选择适当的位置，然后单击**确定**。MapXtreme 2005 创建应用程序并显示 MapForm1.vb [设计]。



5. 单击**HTML**，在 WebForm1.aspx 的底部查看应用程序的 HTML。



6. 在调试菜单，单击启动运行应用程序。



7. 可以像使用任何其它 MapInfo 地图绘制应用程序一样，使用工具栏的控件来操控地图。

MapXtreme 2005 控件

使用其中一个模板创建基本应用程序后，就可以使用各种在 Toolbox 中提供的 MapXtreme 2005 控件扩展。对于使用 Windows 窗体构建的简单地图应用程序，可以使用 MapXtreme 2005 Windows 控件。任何 Visual Studio .NET Toolbox 的 MapXtreme 2005 Windows 控件选项卡中的控件都可以添加到窗体。

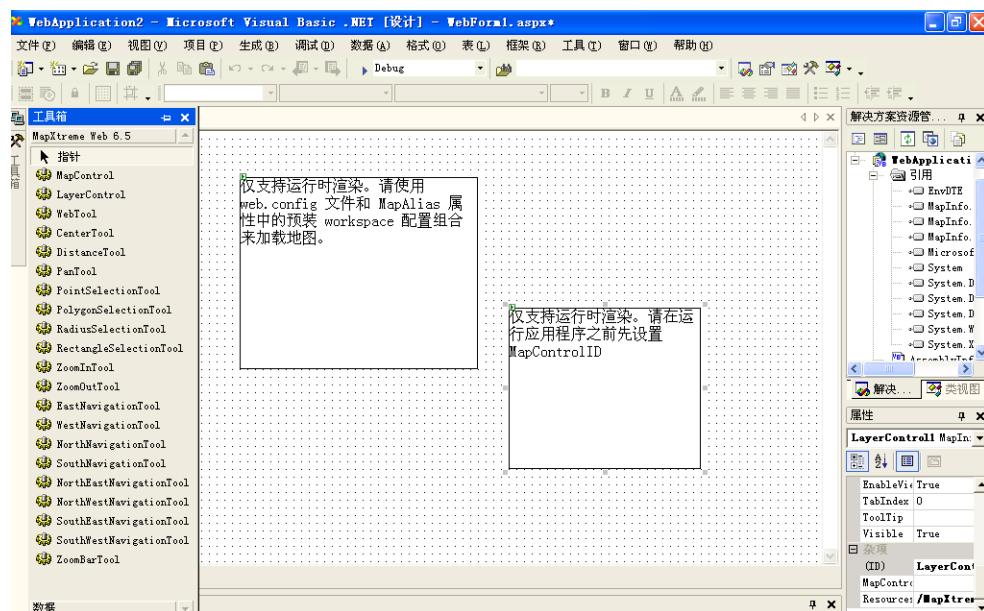
对于 MapXtreme ASP.NET Web 应用程序，可以使用 MapXtreme 2005 Web 控件。任何 Toolbox 的 MapXtreme Web 控件选项卡中的控件都可以添加到窗体。有关详细信息，请参阅第 94 页第 6 章中的 *WebControls 命名空间概述*。

构建 ASP.NET Web 应用程序但不使用模板

不使用 MapXtreme 2005 模板也可以构建 ASP.NET 地图应用程序。例如，可以按以下步骤创建 Visual Basic Web 应用程序：

1. 选择“文件”>“新建”>Visual Studio .NET 菜单的“项目”。 “新建项目”对话框显示。
2. 在 Visual Basic 项目文件夹的“新建项目”对话框上，选择 ASP.NET Web 应用程序并单击**确定**。
3. 从 Toolbox 中的 MapXtreme Web 控件组选择控件并将其拖到窗体。该操作将三个 MapXtreme 2005 程序集作为引用添加到项目中：MapInfo.CoreEngine、MapInfo.CoreTypes 和 MapInfo.WebControls。

该操作也会更新 Web.config 文件，以便像 httpModules 的元素一样添加程序集信息。



部署应用程序

有两种主要策略用于在希望存放应用程序的服务器上安装 MapXtreme 2005 组件：1) 使用包括的运行时安装程序；2) 创建自己的安装程序并添加适当的合并模块 (MSM)。

MapXtreme 2005 SDK 包含 NCP 版本的运行时安装程序 MXTRunNCP.exe。SCP 版本的 SDK 包含 MXTRunSCP.exe。

用运行时安装程序部署

将运行时安装程序 (MXTRunNCP.exe 或 MXTRunSCP.exe) 用作定制安装过程的一部分。这会安装 MapXtreme 2005 程序集并创建需要的注册项。它还包括必需的 .NET Framework v1.1。Web 和 Desktop 应用程序都包括运行时程序集和文件。此外，需要将从 MapInfo Corporation 或经销商获得的运行时许可证文件安装到路径：

C:\Program Files\Common Files\MapInfo\MapXtreme\6.x。请参阅第 18 页中的 *SDK 和运行时许可证*。

对于基于 Web 的应用程序，使用运行时安装程序是较好的策略，这种情况也会创建所需的虚拟目录。所有运行时组件都安装在它们的默认位置。没有用户配置的选项。使用运行时安装程序的另一个好处是可以从 MapInfo 申请补丁程序而无需重新生成安装程序。

使用运行时安装程序的一个缺点是会安装不必要的文件（取决于不同的应用程序）。此外，如果要安装多个启用 MapXtreme 的应用程序，则需要在使用运行时安装程序时维护自己的引用计数。通过直接用自己的安装程序使用 MSM，引用可以自动维护。

使用运行时安装程序部署应用程序的步骤

1. 从 MapXtreme 2005 光盘中选择“安装”。“安装”选项页面显示。
2. 选择“部署安装”并选择“安装运行时环境”。安装向导显示。
3. 遵循提示继续安装。

注： 要运行安装程序，您在计算机上必须具有管理员权限。要安装到 Web 服务器上还需要 IIS 权限。

4. 按照您的需要部署 Windows 或 Web 应用程序（例如，在 Visual Studio 中创建部署项目并添加应用程序）。
5. 将运行时许可证文件安装在路径 C:\Program Files\Common Files\MapInfo\MapXtreme\6.x 中。

如果想从命令行执行运行时安装程序，则遵循以下指示。可执行文件位于光盘的 \Install\InstallRuntime 文件夹中和 C:\Program Files\MapInfo\MapXtreme\6.x\Redistributables 文件夹中。

- 要用最简练的 UI 执行运行时安装程序，请按照以下示例执行运行时安装程序：
`MXTRunNCP.exe /v"/qb"`
- 要自动执行运行时安装程序，请按照以下示例执行运行时安装程序：
`MXTRunNCP.exe /v"/qn"`

如果使用的 MapXtreme 2005 是软件防复制保护的版本，则运行时可执行文件是 **MXTRunSCP.exe**。

用自己的安装程序进行部署

使用 Windows Installer 技术 (MSI) 创建自己的安装程序并包括 MapXtreme 合并模块。

合并模块 (MSM 文件) 是一个独立的程序包，包含了安装组件必需的所有文件、资源、注册表项和安装逻辑。合并模块无法单独安装，它们必须置入 MSI 文件中。

如果要精确调控安装的组件或想创建自己基于 MSI 的安装程序，则可以使用该策略。

使用 MSM 的一个缺点是如果 MapInfo 对产品提供了更新，那么您必须创建自己的补丁程序或更新的安装程序。

使用 MSM 的好处是可以控制要安装的组件且引用计数可以自动维护。特殊版本的程序集仅被复制到全局程序集缓存 (GAC) 中一次，且使用这些程序集为每个应用程序维护引用计数。如果随后删除一个应用程序，则引用计数（按 1 递减）将把程序集保留在 GAC 中。使用这些程序集的最后一个应用程序被删除后，程序集本身将被删除。

有多种开发人员工具可以帮助您创建 MSI 安装程序。例如 InstallShield Developer (InstallShield Corporation)、Visual Studio .NET (Microsoft) 和 Wise for Windows Installer (Wise Solutions)。有关 Windows 安装程序的详细信息，请参阅

http://msdn.microsoft.com/library/en-us/msi/setup/windows_installer_start_page.asp。

MapXtreme 2005 合并模块

以下是 MapXtreme 2005 的合并模块。必要时将这些模块包括在安装程序 MSI 中。有关使用合并模块的详细信息，请参阅第 431 页附录 D 中的将应用程序打包。

合并模块的名称	包括的程序集	目的	何时需要
定制符号 MapInfoCustSymb.msm	无	包含定制符号	应用程序要使用常用位图符号时需要
台式 MapInfoDesktop_6.5.msm	MapInfo.Windows MapInfo.Windows.Dialogs MapInfo.Windows.Framework	包含具有 .NET 控件的程序集（用于 C# 和 VB .NET 桌面应用程序）	应用程序使用“桌面”工具时需要
字体 MIFonts_6.1.msm	无	包含 MapInfo 字体，如 Symbols、Cartographic、Real Estate、Arrows、Miscellaneous、Oil&Gas、Transportation、Weather 和 Shields。	应用程序要使用常用 TrueType® 字体时需要
地图化 MapInfoCoreEngine_6.5.msm MapinfoMXTConfig_6.5.msm MapInfoCoreEngineIntl.msm * MapInfoCoreResJPN_6.5.msm † MapInfoCoreResCHN_6.5.msm ‡	MapInfo.CoreEngine MapInfo.CoreTypes MapInfo.WMS.Client MapInfo.Windows.Printing	提供核心地图绘制功能。它们也为这些程序集安装常用配置和默认引用文件。	必需

合并模块的名称	包括的程序集	目的	何时需要
Web 控件 MapInfoWeb_6.5.msm	MapInfo.Web MapInfo.WebControls	安装 .NET 程序集 MapInfo.Web，大量“web 资源”将用在基于 web 的应用程序中。 它还启动 ASPNetState 服务，并为“web 资源”创建虚拟目录（如果已安装 IIS）。	任何应用程序使用常用 web 控件时需要
Web 服务客户端 MapInfoServices_6.5.msm	MapInfo.Services	包含用于地理编码和路径规划的程序集	应用程序需要地理编码和路径规划时需要
WFS MapInfoWFS_6.5.msm	MapInfo.WFS.Server	包含 WFS 所需的程序集。	应用程序使用 Web 图元服务用于数据变换时需要
WMS MapInfoWMS_6.5.msm	MapInfo.WMS.Server	包含 WMS 所需的程序集。	应用程序使用 Web 地图服务用于检索数字图像时需要。

* 使用 MapXtreme SCP 构建部署应用程序时包括。

使用 MapXtreme JPN 构建部署应用程序时包括。

** 使用 MapXtreme CHN 构建部署应用程序时包括。

用自己的安装程序部署应用程序的步骤

以下是使用自己的安装程序安装 MapXtreme 2005 时必须执行的任务：

1. 包括所需的 MSM。

默认情况下，Visual Studio .NET 和 InstallShield 在 C:\Program Files\Common Files\Merge Modules\Program Files\Common Files\Merge Modules 中查找要包括的合并模块。这样就可以快速生成安装程序，不必重新配置开发环境。

2. 包括运行时许可证文件。

指向文件夹 C:\Program Files\Common Files\MapInfo\MapXtreme\6.x\WebResources 的虚拟目录。

部署访问数据的应用程序

任何包括数据访问的 Visual Basic 或 Visual C# 应用程序都对 Microsoft Data Access Components (MDAC) 2.7 版或更新版本具有依赖关系。安装应用程序之前 MDAC 必须安装在目标计算机上，否则应用程序会失败。MDAC 2.8 位于 MapXtreme 2005 产品光盘。

已部署 Web 应用程序临时目录的权限

对于部署的 web 应用程序，如果使用集成安全系统，则请确保访问该站点的所有用户都具有访问临时目录及任何其它资源的权限。MapXtreme 2005 在 ASP.NET 处理空间中执行，该处理执行从 IIS 进程传入的安全标记。必须将权限授予任何可能登录并访问该目录的用户。如果使用匿名访问，则必须将访问临时目录的权限授予 IUSR_LocalMachineName 系统帐户。MapXtreme 2005 会从当前的 TEMP 环境设置中获得临时目录。

应用程序数据文件

应用程序数据文件是应用程序使用的不可执行的文件。MapXtreme 2005 安装和使用以下应用程序数据文件。有关定制这些文件位置的详细信息请参阅第 455 页附录 F 中的应用程序数据文件的位置：

文件类型	文件名
缩写文件	MAPINFO.W.ABB
画笔文件	MAPINFO.W.PEN
投影文件	MapInfoCoordinateSystemSet.xml
矢量符号文件	MapInfo.fnt
定制符号目录	CustSymb
Nadcon 文件	*.las、*.los
jgd2000 文件	jgd2000.*

默认情况下，MapXtreme 应用程序在以下目录寻找数据文件：

- C:\Program Files\Common Files\MapInfo\MapXtreme\6.0 - 这是 MapXtreme 2005 安装程序存放这些文件的目录。
- 用户的应用程序存放的目录。对于 Windows 应用程序，这是存放 .exe 文件的目录。对于 web 应用程序，这是存放 Web.config 文件的目录。
- MapInfo.CoreEngine 程序集文件夹 - 该目录是应用程序寻找数据文件的最后位置。

3

地图绘制概念

创建地图绘制应用程序之前，理解基本的地图绘制概念及其如何在 MapXtreme 2005 中实现这些概念会对您有所帮助。本章讨论了学习 MapXtreme 2005 时将遇到的常用概念。

在本《开发人员指南》的最后，我们提供了有关地图绘制和编程的术语表，会对您有所帮助。

本章内容：

- ◆ 地图绘制和 *MapXtreme 2005* 40
- ◆ 使用 *MapXtreme 2005* 进行地理编码 45
- ◆ 使用 *MapXtreme 2005* 进行路径规划 45

地图绘制和 MapXtreme 2005

地图绘制应用程序的核心元素是地图。本章概述了使用 MapXtreme 2005 构建应用程序时可能遇到的最重要的地图绘制术语。此外还介绍了 MapXtreme 2005 对象模型中对应的命名空间，方便您快速获得需要的技术信息。这些主题包括：

- 地图
- 表
- 图层
- 图元
- 标注和图例
- 主题
- 工具
- 工作空间
- 坐标系和投影

地图

地图显示地图图元之间的空间关系，例如城市边界、客户位置或输电线路。地图可以让您从视觉上确定图元的方向及它们所表示的内容。除了图元外，地图上的元素可包括标注、标题、图例和主题。主题是根据某项涉及图元的操作和地图上的信息创建的。

地图包含在 MapControl 中。MapControl 还提供了查看地图的基本工具（平移、放大、缩小、居中）。

可通过多种方法来创建地图：

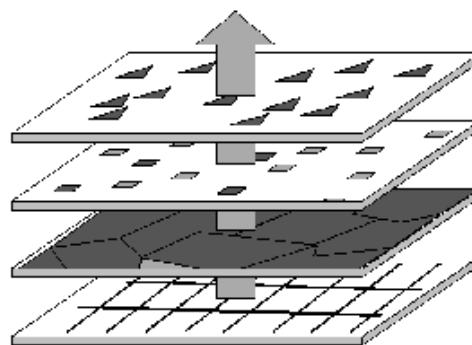
- 使用 MapXtreme 2005 Workspace 管理器 构建和保存地图工作空间。（请参阅第 324 页中的 *Workspace 管理器的特性*）。
- 请使用 MapXtreme 2005 模板，该模板提供了可拖放到 Visual Studio .NET 窗体上的 MapControl（对于桌面应用程序，请参阅第 124 页第 8 章中的 *MapControl*，对于 web 应用程序，请参阅第 94 页第 6 章中的 *WebControls 命名空间概述*）。
- 使用 MapXtreme 2005 对象模型通过编程方式在应用程序中构建地图绘制（请参阅第 218 页中的 *MapFactory* 和程序员参考（联机帮助）中的 MapInfo.Mapping 命名空间）。

表

表包含要显示在地图上的数据。表保存描述图元信息的行和列，这些图元包括它们的几何体、样式和属性。MapXtreme 2005 支持各种数据源中的表，包括本地表 (MapInfo .TAB)、关系数据库管理系统 (RDBMS)、dBase、MS Access、ASCII 文件和 ESRI ShapeFile。特性表包括光栅、网格、无缝、视图、WMS 和 ADO.NET。可通过 TableInfo 类查看表的类型。可以通过 Data 命名空间中的 Catalog 打开或关闭表。请参阅[第 10 章：处理数据](#)。

图层

地图由图层组成。图层包含地图图元，例如，邮政编码边界、学校或街道网络。了解图层的顺序，这点非常重要。最底部的图层是第一个绘制的，而最顶部的图层是最后绘制的。应将其图元会遮蔽其它图层图元的图层分别放置到较低的位置。例如，边界区域图层应放置到点图层下面。



MapXtreme 2005 中的图层可以表示除图元外的内容。图层可以是光栅或网格图像、无缝地图（连接的地图），包含标签或用户绘制的图元，还可以包含对象主题，例如饼图主题。可以通过简单的定位将图层分组，并简化其图元的动画。主要接口是 IMapLayer。有关详细信息请参阅[第 220 页中的 Layers](#)。

图元

图元由它们的几何体、样式、数据源、键和属性描述。通常，图元为表中的行。支持的几何体包括包含给定区域的闭合对象 (Polygons、MultiPolygons、Rings、Rectangle、RoundedRectangles 和 Ellipses)；表示数据某一位置的点对象 (Points、MultiPoints)；以及包含给定距离的直线对象 (Curves、MultiCurves 和 LegacyArcs)。

计算机化地图其中一项主要用途就是获取关于图元的信息。在 MapXtreme 2005 中，可通过多种方法将图元返回到 FeatureCollections，例如，使用构架从空白创建，使用选择集工具或方法选择，或通过搜索这些满足特定标准集合的工具或方法的 Catalog 选择。

Feature 类位于 MapInfo.Data 命名空间。

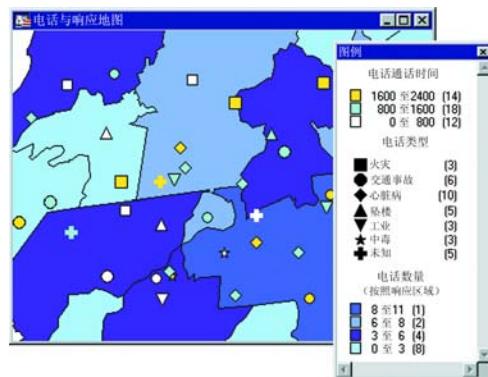
标注和图例

如果地图中没有描述显示内容的元素，则这种地图用处不大。地图需要标注和图例等文本。上述提及的标注属于名为 LabelLayer 的图层类型。该类型允许控制标注的各个方面，例如标注可视性、标注位置、标注样式和标注内容。使用标注的 MapXtreme 2005 类包括 LabelSource、LabelProperties 和 LabelModifiers。请参阅 [第 216 页中的 Mapping 命名空间简介](#)。

其它文本元素也可以用在地图中帮助传递正确的消息。图例是以编码方式描述图元的制图元素。例如，图例可以描述边界（例如学校地区）、直线（例如电力线路网络）或点（公司办事处位置）。图例也包含标题，用于描述地图表示的全部内容。

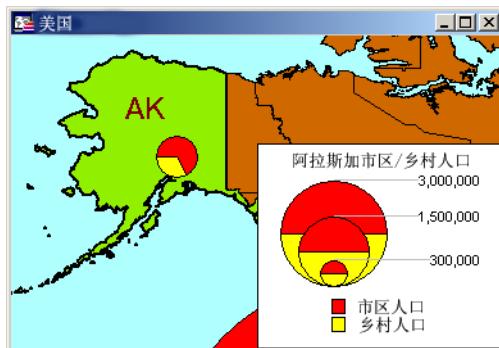
MapXtreme 2005 中，图例连同地图标题和比例栏都是 Adornments 类的一部分。

Adornments 位于 MapInfo.Mapping 命名空间中。



主题

计算机上的电子地图不仅可以直观地显示地图图元之间的空间关系，而且还可以分析与图元关联的基础数据，从而了解到与看到的内容有关的详细信息。通常的分析技术是创建基于图元图层的主题，该图元图层的数据以特定的方式排列。例如，范围主题显示的颜色区中，每个颜色表示了地图上满足相同标准的图元。分级符号主题可用于表示人口的分布，例如最大的符号表示最多的人口。



此外也可以为标注创建主题。例如，使用范围标注主题表示城市之间的相对人口数量。最大的标注表示有最多人口的城市。

MapInfo.Mapping.Thematics 命名空间包含的类以 Feature 图层上的样式重载和 Object 主题实现来主题。对对象主题添加新的图层时修饰符主题更改样式。所有主题都实现 ITheme 接口。

工具

多数地图绘制应用程序提供工具栏按钮（工具）的分类，用于一般绘制任务（例如在地图上绘制直线）和导航任务（例如放大）。MapXtreme 2005 提供大量地图绘制工具，此外，还可以创建自己的定制工具。

这些工具分为桌面工具和 web 工具，每个工具的 API 都包含在其自己的命名空间中（MapInfo.Tools 用于桌面工具，MapInfo.WebControls 用于 web 工具）。

有关 MapXtreme 2005 中桌面工具的详细信息请参阅第 111 页中的可用工具。有关 web 工具的详细信息请参阅第 6 章：运用 MapXtreme 2005 Web 控件。

工作空间

严格来说这不是地图绘制概念，将工作空间包括在这里，是因为工作空间可以使得处理所有地图绘制元素更为容易。MapXtreme 2005 支持使用 .MWS 扩展名且基于 XML 的工作空间格式。其中全是地图的设置。工作空间的格式在 [附录 G：理解 MapInfo 工作空间](#) 中说明。MapXtreme 2005 提供了名为 *Workspace* 管理器的实用程序，帮助构造工作空间和保存工作空间方便以后使用。请参阅 [第 19 章：Workspace 管理器](#)。

坐标系和投影

坐标系和投影是两个重要的地图绘制概念，应该对它们有基本的理解。投影是指如何将地图显示在图纸地图或计算机屏幕等平面上，而坐标系说明的是如何将地图图元在空间上安排。开发应用程序，尤其是开发空间精度和准确度很重要的应用程序时，坐标系和投影都需要重点考虑。

将球面的对象显示在平面上时，投影是减少失真发生的一种方法。主要需要权衡两个方面：保持图元的面积相等和保持图元的真实形状。有多种不同类型的投影，每种投影都可用于减少给定区域上的失真量。某些投影保持形状，而其它投影保持面积、距离或方向的精确度。

坐标系是一组参数，告知如何解释对象的位置座标。这些参数之一是投影。坐标可以有两种类型：球面坐标或笛卡儿坐标。球面坐标描述在 Earth 曲面上的位置，而笛卡儿坐标以二维空间来描述平面位置。二者均用 x 和 y 坐标表示。计算图元的距离或面积时就会出现差异，表示街道或河流等真实 Earth 位置时用球面坐标，而表示大脑解剖或棋盘等相对位置时用笛卡儿坐标。

开发应用程序时需要重点考虑地图要使用的坐标系。包含距离和面积计算的分析操作（例如缓冲、路径规划和查询）使用坐标系和投影来得出正确的结果。

坐标系和投影类是 Geometry 命名空间的一部分。有关详细信息请参阅 [第 16 章：空间对象和坐标系](#)。

使用 MapXtreme 2005 进行地理编码

以上讨论的地图都使用提供附加信息的数据，该附加信息不显示在地图上。例如，商店位置的表不仅包括将商店放置在正确地图位置的地理坐标，还可以包含与位置有关的数据，例如商店营业时间、客户服务电话号码和经理姓名。这让应用程序可以分析和得出那些会丢失在表的行和列中的信息。

通常，定制数据的表与参考图层一起包括在地图上，例如街道、城镇边界和河流等表示真实区域环境的图元。通常购买这些参考图层后就可以显示在地图上。MapInfo Corporation 销售大量世界各地的参考数据。

不过像您的商店位置等定制数据还不能显示在地图上。表必须包含地理坐标，这样地图绘制引擎才能知道在何处绘制对象。将坐标分配到数据的过程称为地理编码。任何包含地理信息（例如地址或邮政编码）的数据表都可以地理编码。在包含相同位置的已地理编码的表上，该过程包括匹配定制的表。如果地址匹配完成，则地理编码表的坐标将分配到定制的数据。此外，定制的数据可以在地图上查看。

通常，地理编码是创建地图过程中的前期步骤。作为地图绘制应用程序的开发人员，您需要考虑要在地图上显示的数据类型及其地理编码的需要。

MapXtreme 2005 框架提供了使用地理编码客户端的类，该客户端可以访问 MapInfo Corporation 的服务器地理编码产品。有关地理编码的详细信息请参阅联机帮助中的 MapInfo.Geocoding 命名空间（可通过 Visual Studio .NET 访问）。（中文版不包括此功能。）

使用 MapXtreme 2005 进行路径规划

MapXtreme 2005 开发人员可以使用的另一个组件是路径规划。驾驶导航型应用程序和那些与运输或布置电缆的规划路径有关的应用程序均需要使用路径规划。通常的目标是用最短距离或最短旅行时间来定位路径。

本 MapXtreme 发行版提供了四种路径规划类型：点到点路径规划、多点路径规划、矩阵路径规划和等值线路径规划。每种类型都提供了大量选项，用于按需创建适当的路径规划网络。

像地理编码所提供的功能一样，MapXtreme 2005 允许开发人员在他们的应用程序中使用预先构建的路径规划客户端，这些应用程序可以与 MapInfo Corporation 的路径规划服务器产品交互。请参阅 MapXtreme 2005 联机帮助中的 MapInfo.Routing 命名空间。（中文版不包括此功能。）

4

理解 MapXtreme 2005 体系结构

本章着重介绍 MapXtreme 2005 体系结构的设计，方便您根据开发需求做出精明的选择。理解产品的体系结构有助于创建出有效使用产品特性和功能的应用程序。

本章内容：

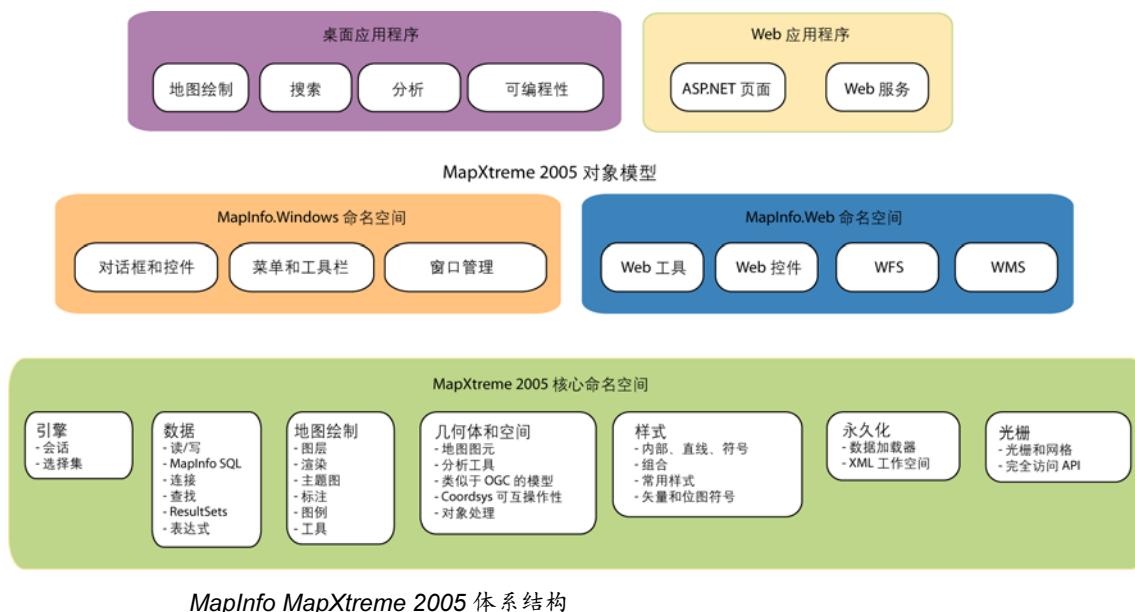
- ◆ *MapXtreme 2005 体系结构* 48
- ◆ 对象模型概述 49
- ◆ *Web 应用程序体系结构* 53
- ◆ 桌面应用程序体系结构 55

MapXtreme 2005 体系结构

MapXtreme 2005 的构建基础是 Microsoft .NET 框架，MapXtreme 2005 使用了 Microsoft .NET 框架包括的功能。MapInfo 的这种创新让您可以并适应在 .NET 框架上开发应用程序。我们还努力结合本公司的 Windows 产品的功能和易用性，使这些产品成为一套综合性的对象模型。要为可以预测到的未来技术开发基于 Windows 的产品，那么对象模型就是 MapInfo 合作伙伴、客户以及 MapInfo 自身的基础。

使用类似的代码，您可以开发既能在桌面又能在 Web 上部署的应用程序。如果开发桌面应用程序，则可以调整应用程序，只需要更改很少的代码即可用于后续的 web 部署。

下图说明了 MapXtreme 2005 体系结构。MapInfoCoreEngine.dll 程序集和 MapInfo.CoreTypes.dll 程序集包含了大多数核心地图绘制和数据访问功能。Core Engine 的上面是 MapInfo.Windows 和 MapInfo.Web 命名空间，这些命名空间包含了控件、工具和其它特定于每种部署环境的功能。任何从 MapXtreme 2005 对象模型开发的应用程序都在 MapInfo.Windows 或 MapInfo.Web 命名空间之上构建。



对象模型概述

MapXtreme 2005 对象模型由大量命名空间组成。.NET 命名空间是分类系统，用于区分具有相同名称的其它对象和特定的类、方法和属性。通过使用命名空间，.NET 开发人员可以避免对象名称与其方法和属性名称发生冲突。

MapXtreme 2005 包含主要接口和类的彩色框图，并通过命名空间分隔显示它们如何彼此相关。可以通过“学习资源”页面查看 PDF 版本的框图，该页面可通过 Windows “开始”菜单访问。特殊情况下，可通过以下方式访问“学习资源”页面：开始 > 程序 > MapInfo > MapXtreme 2005 6.5 > 学习资源

下表包含了在 MapXtreme 2005 对象中实现的多个命名空间。每个命名空间的概述包括在后续章节中。本手册包含了与命名空间及其使用有关的详细信息，以下每个概述说明都包含了对本手册某些部分的引用。

- MapInfo.Data
- MapInfo.Data.Find
- MapInfo.Engine
- MapInfo.Geometry
- MapInfo.Mapping
- MapInfo.Persistence
- MapInfo.Raster
- MapInfo.Styles
- MapInfo.Tools
- MapInfo.Windows.Controls
- MapInfo.WebControls
- MapInfo.Geocoding
- MapInfo.Routing

在 MapXtreme 2005 的“程序员参考”中，完整的对象模型由命名空间组织并集成在 Visual Studio .NET 中。

如果已经使用了 MapX 或非 .NET 版本的 MapXtreme (MapXtreme for Windows v3)，请务必查看附录 C：迁移到 *MapXtreme 2005* 比较两个产品的对象模型。

MapInfo.Data 命名空间

MapInfo.Data 命名空间包含了实现 MapInfo 数据提供方的类和接口。对象模型具有多个类用于访问数据。可以根据存储的数据格式而使用特定的类来访问数据。此外，我们现在实现了 ADO.NET，用来访问以其它类不包含的格式包含的任何数据。有关 MapInfo.Data 命名空间的详细信息，请参阅第 10 章：处理数据和第 11 章：从 DBMS 访问数据。另请参阅第 5 章：规划数据模型。

MapInfo.Data.Find 命名空间

MapInfo.Data.Find 命名空间包含了用于搜索数据的类。该命名空间通过指定可以绘制地图的表和执行搜索的列（必须带有索引）来简化对象的搜索。有关 MapInfo.Data.Find 命名空间的详细信息，请参阅第 14 章：查找位置。

MapInfo.Engine 命名空间

MapInfo.Engine 命名空间包含了所有直接与核心功能有关的类，该核心功能驱动基于 MapXtreme 2005 的所有应用程序。其中包括核心 Session 类，该类是所有 MapXtreme 2005 应用程序的起点。有关 MapInfo.Engine 命名空间的详细信息，请参阅第 9 章：运用核心 MapXtreme 2005 类。

MapInfo.Geometry 命名空间

MapInfo.Geometry 命名空间是一种可扩展的层次结构，基于 OGC (Open GIS Consortium) 标准、坐标系可互操作性和对象处理。MapInfo.Geometry 命名空间包含了用于创建和编辑 Geometry 对象的类、接口和枚举。有关 Geometry 命名空间的详细信息请参阅第 16 章：空间对象和坐标系。

MapInfo.Mapping 命名空间

MapInfo.Mapping 命名空间包含了用于创建、显示和导出地图、图层、修饰符和标注的类、接口和枚举。有关 MapInfo.Mapping 命名空间的详细信息，请参阅第 12 章：将地图绘制功能增加到应用程序。

MapInfo.Mapping.Legends 命名空间

MapInfo.Mapping.Legends 命名空间包含用于创建并显示 Cartographic 和 Thematic Legends 的类、接口和枚举。有关详细信息请参阅第 227 页第 12 章中的图例和第 13 章：使用主题和图例。

MapInfo.Mapping.Thematics 命名空间

MapInfo.Mapping.Thematics 命名空间包含了作为图层样式和图层本身实现主题的类。对象主题添加新的图层时，主题可应用到 Modifier 主题来更改样式。所有主题都实现 ITheme 接口。有关 MapInfo.Mapping.Thematics 命名空间的详细信息，请参阅第 13 章：使用主题和图例。

MapInfo.Persistence 命名空间

MapInfo.Persistence 命名空间包含了支持基于 XML 工作空间读写的类，用于启用对地图绘制工作空间的保存和检索。请参阅附录 G：理解 MapInfo 工作空间。

MapInfo.Raster 命名空间

MapInfo.Raster 命名空间公开了 MapInfo C/C++ Raster 和 Grid API 的全部功能。使用 MapInfo.Raster.RasterRead 可以打开光栅图像用于查询。使用 MapInfo.Raster.GridRead 可以打开网格图像用于查询。使用 MapInfo.Raster.HillshadeWrite 可以将 Hillshading 添加到现有的网格。相关类包括 MapInfo.Raster.RasterInfo 和 MapInfo.Raster.GridInfo。有关 Raster 命名空间的详细信息请参阅第 17 章：运用光栅和网格。

MapInfo.Styles 命名空间

MapInfo.Styles 命名空间强调了新的 Styles 对象模型。Style 类是所有样式的基类。有关 Styles 命名空间的详细信息请参阅第 15 章：为地图设置样式。

MapInfo.WebControls 命名空间

MapInfo.WebControls 命名空间为使用 MapXtreme ASP.NET 应用程序的 Visual Studio .NET 模板提供支持。对于该命名空间，具有 MapControl 和 LayerControl 设计阶段扩展及 web 工具。有关 MapInfo.WebControls 命名空间的详细信息，请参阅第 6 章：运用 MapXtreme 2005 Web 控件。

MapInfo.Windows 命名空间

MapInfo.Windows 命名空间包含了实现各种窗口控件及其必需组件的类，用于在 Windows 应用程序中开发窗体。Windows.Dialogs 命名空间包含了实现 Windows 应用程序中使用的各种对话框和对话框组件的类。有关 MapInfo.Windows 命名空间的详细信息，请参阅第 8 章：用于桌面应用程序的对话框和控件。

MapInfo.Tools 命名空间

MapInfo.Tools 命名空间包含了用于创建和实现在桌面地图应用程序中使用的多种工具的类。有关 MapInfo.Tools 命名空间的详细信息，请参阅第 7 章：使用桌面工具。

MapInfo.Geocoding 命名空间

MapInfo.Geocoding 命名空间包含了定义 MapXtreme 2005 客户端用于地理编码的类、接口和枚举。支持使用 MapInfo 地理编码服务器或 MapInfo Location Utility 服务的地理编码（中文版中没有此功能）。要执行地理编码，则必须有用于运行地理编码服务器或 Location Utility 服务的 URL。地理编码服务器和 Location Utility 服务的接口是类似的，因为它们对于地理编码请求、约束、响应、结果代码、输入和候选地址都使用相同的类。

MapInfo.Routing 命名空间

MapInfo.Routing 命名空间包含的类、接口和枚举用于让 .NET 客户端进行 Routing。

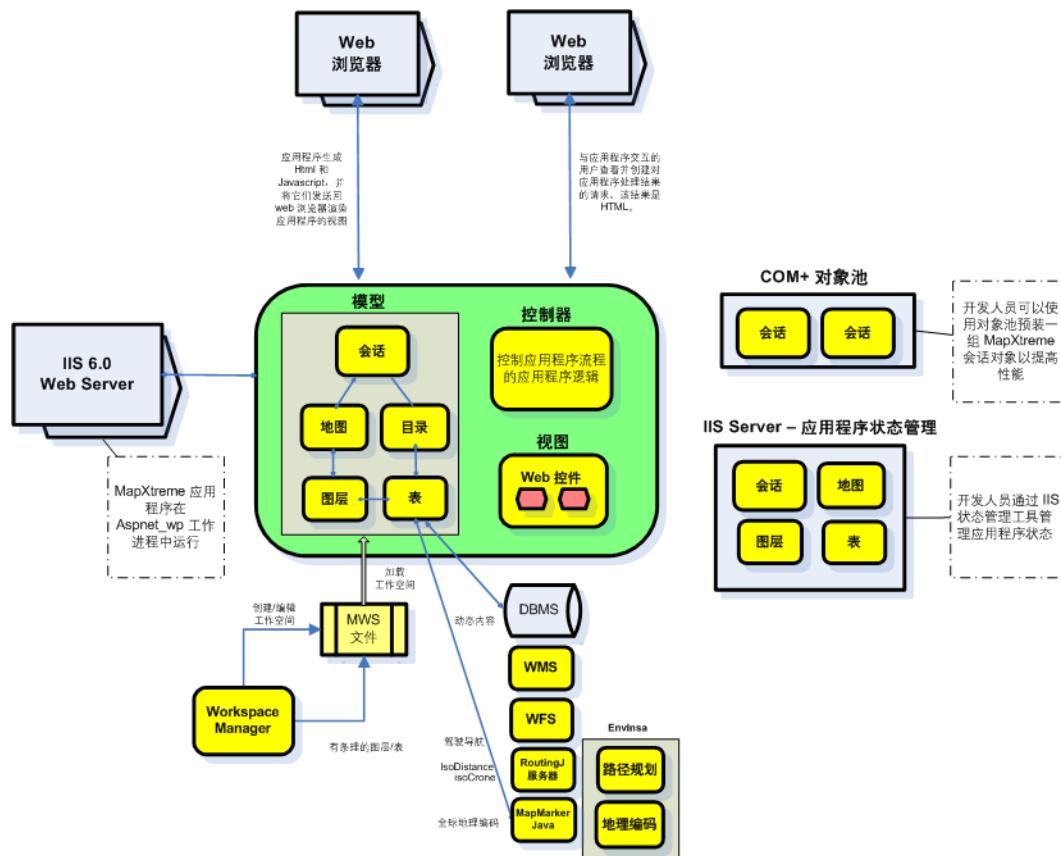
MapInfo.Routing 命名空间包含了支持点对点、多点、矩阵和等值线路径规划的类，以实现最短时间或最短距离。它与其它 MapInfo 路径规划服务器产品进行交互。此外还具有避免特定点的功能。路由结果可以包括能在地图上显示的详细指示和（或）路由几何体。

应用程序体系结构

在了解 MapXtreme 2005 命名空间概况后，下一步就要考虑计划构建的应用程序的体系结构。

使用 MapXtreme 2005，可以同时构建 web 应用程序和桌面应用程序。以下部分说明了 web 和桌面应用程序可能具有的体系结构。设计基于模型 - 视图 - 控制器范例，该范例将应用程序的数据模型、用户接口和控制逻辑分隔到三个独立的组件中。这样就可以在对其它组件产生最小影响的情况下修改某一组件。[第 5 章：应用程序的规划](#)提供了关于主要设计元素的重要附加信息以及在计划 MapXtreme 2005 应用程序时要考虑的内容。

Web 应用程序体系结构



体系结构的描述

上述 Web 应用程序考虑到以下组件和功能：

- MapXtreme 2005 Web 应用程序
- Microsoft .NET 基础结构
- 地图构建工具
- 数据访问

MapXtreme 2005 Web 应用程序

典型的 MapXtreme 2005 web 应用程序包含视图（表示层）、模型（与数据源和应用程序内部数据模型进行交互）和控制器（控制应用程序流程的业务逻辑）。

MapXtreme 2005 提供了集成到 Microsoft Visual Studio .NET 的 web 模板，帮助您创建初始的 web 应用程序。构建 web 应用程序的相关教程，请参阅[附录 D：生成 Web 应用程序](#)。

通过将 MapXtreme web 控件拖放到 Visual Studio .NET web 窗体来构建视图。通过使用 MapInfo.Session 和 MapInfo.Data 命名空间下的对象，构建内部数据结构并与外部数据源、基本地图和动态内容进行交互。使用控制器代码将视图和数据联系在一起并向用户提供活动顺序，以便有效地利用应用程序来解决业务需求或问题。

Microsoft .NET 基础结构

MapXtreme 2005 在 Microsoft .NET framework 1.1 下运行。使用 MapXtreme 2005 构建的应用程序以 ASP.NET 应用程序的形式在 IIS server 6.0 的工作进程中运行。

Microsoft ASP.NET 框架为高性能企业应用程序的开发人员提供了 COM+ 对象池，以便能够预装如工作空间等对象。MapXtreme 的对象模型在该框架下能够非常有效地运行。该框架还提供了应用程序状态管理工具（如 StateServer 和 SQL Server）及自动和手动状态管理控件。[第 5 章：规划 Web 应用程序](#)显示了关于这些主题的重要信息。

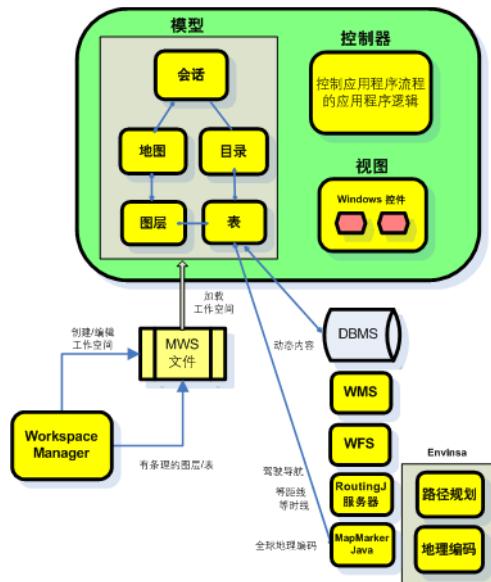
地图构建工具

使用 MapXtreme Workspace 管理器创建应用程序的基本地图。通过这些工具可以管理每个地图图层并控制其缩放级别、标注、样式、主题图和修饰，以提供对应用程序的精确表示。信息保存到基于 XML 的工作空间，以方便日后检索。请参阅[第 19 章：Workspace 管理器](#)。

数据访问

该体系结构的一个要素是访问动态数据内容的能力。MapInfo.Data 命名空间中的对象提供了这种能力。数据内容可来自大量数据源，如 WMS、WFS、远程数据库管理系统、GPS 中的实时种子或 MapInfo 路径规划服务中的驾驶导航。要充分利用各种数据，可以同时使用来自不同数据源的数据。请参阅[第 79 页第 5 章中的规划数据模型](#)。

桌面应用程序体系结构



桌面应用程序的体系结构与其模型 - 视图 - 控制器设计中 web 应用程序的体系结构类似。

独立的组件用于表示层、应用程序模型和业务逻辑。使用 Workspace 管理器构建所需的基本地图。使用 Windows 控件和对话框赋予应用程序丰富的用户体验。此外也可以同时使用来自各种数据源的动态数据内容，控制应用程序的流程和逻辑。

有关详细信息，请参阅第 58 页第 5 章中的规划桌面应用程序。

5

应用程序的规划

在规划桌面或 web 应用程序时需要确定关键应用程序设计方案，本章是对此过程的最佳实践。本章包括以下主题：

- ◆ 规划桌面应用程序 58
- ◆ 规划 Web 应用程序 60
- ◆ 应用程序性能 72
- ◆ 规划数据模型 79

规划桌面应用程序

MapXtreme 2005 是一种基于 .NET 的对象模型，向任何基于 .NET 的开发工作提供地图功能。其中包括许多使用 .NET 框架开发应用程序的新技术。在本节中，我们将集中讨论 WinForm 和旧的基于 COM 的应用程序。

将 MapXtreme 2005 添加到现有应用程序

如果具有现有的 WinForm 或其它基于 .NET 的应用程序框架，则可以简单地将 MapXtreme 2005 类集成到应用程序中。按照以下章节提供的“最佳实践”指导进行。MapXtreme 2005 对象模型是一个完全符合 .NET 的对象模型，因此您可以将 .NET 框架的功能与其它对象模型交互。以应用程序框架定义的模式简单地开发应用程序，并引入 MapXtreme 2005 对象。

.NET 并不意味着 COM 的结束。许多开发厂家对 COM 对象模型做了很大的努力，投入很大不能够轻易放弃。那么开发具有 COM 对象的应用程序的最佳方法是什么呢？

最可能的情况是，新的 .NET 代码将必须与现有的 COM 代码可互操作（而不是相对立的情况），因此我们将集中讨论此种情况。.NET 客户端通过运行时可调用包装程序 (RCW) 访问 COM 服务器。RCW 将 COM 对象打包，并将其与 .NET 普通语言运行时 (CLR) 环境进行调节，使 COM 对象显示到 .NET 客户端，就象它是本地 .NET 对象，并使 .NET 客户端显示到 COM 对象，就象它是一个标准的 COM 客户端。

例如，您可以直接从 Visual Studio 将引用创建到这些 COM 对象，并简单地与它们进行交互，就象对 .NET 类一样。这样就可以在与旧 COM 对象进行通信时创建 .NET 应用程序，并将结果传递给 MapXtreme 2005 .NET 对象。将正在开发的 .NET 应用程序作为应用程序框架的协调段。这样就可以使用 .NET 框架中最新的技术，并保留在旧的程序库中现有的投入。因为 COM 对象中的所有数据已经转换到 .NET 环境，因此可以将这些数据传递到其它受管理的类，而不会产生域的问题。

示例应用程序和项目模板

MapXtreme 2005 集成到 Visual Studio .NET 中，并附带多种示例应用程序和项目模板。示例应用程序从不同方面展示如何在 .NET 框架和 Windows 窗体应用程序中使用对象模型。这些模板是应用程序开发的基本起点。

这些项目模板是 Visual Studio 的一部分，可以用于创建启动项目，并演示通用的应用程序框架。从“新建项目”菜单项开始，并选择 MapXtreme 2005 项目 > 地图应用程序图标来创建一个基本的单文本界面 (SDI) 应用程序。示例桌面应用程序包含一个项目来演示基本的多文本界面 (MDI) 应用程序，以及如何最好地将 MapXtreme 2005 集成到此模型。

.NET Windows 窗体开发平台是其方法和功能的基础。我们建议使用可以提供更完整的 UI 开发平台的第三方工具。

MapXtreme 2005 附带的 Desktop 示例应用程序设计用于以最小开销显示特定任务。使用这些作为学习工具，而不是作为产品应用程序的起点，因为这些并不是设计为应用程序开发实例。

最佳实践

包括在 Visual Studio “新建项目” 对话框中的 MapApplication 模板仅用于快速原型开发。使用该模板作为快速演示或概念证明的起点。Windows 应用程序是一个基本模板，因为启动时只引用 MapXtreme 程序集。

WinForm 和 .NET 框架包括对基于 MDI 和 SDI 应用程序的基本支持。WinForm 不提供可与 Microsoft 基础类 (MFC) 相似的应用程序框架。不过可以执行 MDI 和 SDI，缺少的是文档视图、数据更改及其它基于 MFC 的用户界面概念。

WinForm 还具有一些 UI 元素，这些元素不如真实的应用程序所需要的那样稳定。我们建议使用可以提供更完整的 UI 开发平台的第三方工具。

MapControl

MapControl 是与桌面上地图交互的主要控件。运行时，开发人员可以加载地图，并使用图层对话框修改图层列表。此设计时功能对于在原型应用程序上的快速显示地图非常有用。对于实际应用程序，您可能想要使用作为应用程序启动部分的加载方法。使用设计器中的右鼠标上下文菜单，用“清除地图”选项将工作空间重新设置为初始状态。

.NET 开发模型一个重要的方面是业务逻辑与表示的分离。MapXtreme 在其控件中使用此范例，使用户可以打开一个对话框修改对象的副本。以此方法，对话框不直接编辑运行时对象，可以在其它进程或机器中创建新的对象，并将串行化后的版本传送回去用于您的应用程序。

另请参阅 [第 124 页第 8 章中的要在应用程序中使用的关键控件](#)。

常用对话框

MapXtreme 附带许多常用对话框和控件，处理操纵 MapXtreme 对象的多项任务。Workspace 管理器中使用的对话框和控件都可以在应用程序中使用。这些对话框只是低级控件的容器。这些控件设计用于在特殊情况下与多数 MapXtreme 对象一起使用。

常用对话框使用包含 MapXtreme 的控件创建基本的 UI 组件。这些控件置于对话框中，用于创建特定的 UI 组件。同样地，您可以使用基本控件设计自己的对话框。要创建对话框，只需创建对话框类，然后将控件拖放到表面。

也可以通过可视化继承来自定义常用对话框。可以从我们的对话框中派生类，然后通过重载方法和属性自定义某些行为。其它对话框（如“图层控件”）都可定制，因此可以更改默认行为，以便隐藏不想让用户访问的控件、更改图标的外观或删除选项卡。

基本上可以控制 MapXtreme 包含的所有 UI 组件。这样就可以创建自己的自定义界面。使用如下所述的任一种方法都可以创建一个自定义的应用程序，只显示需要的内容（而不是我们定义的所有内容）。

另请参阅第 129 页第 8 章中的 *MapInfo.Windows.Dialogs* 命名空间。

MapTool 对象模型

MapTool 是用作与地图交互的对象。这些工具实现您需要地图具有的基本行为。有一些基于视图的工具（缩放和平移），以及在特定图层创建几何体的工具，还有生成以地图某区域为基础的选择集的工具。每种工具都是可自定义的，做法是使用事件来捕获预处理和后处理事件。

与工具相关联的方法的属性在工具上，而不是在图层上。例如，图层的可选择性是在工具上指定，而不是在图层上指定。以此方法，可以使用一个工具在某图层上选择对象，而使用另一个工具从另一图层上选择。这提供了更多在工具中使用的自由，而且您还可以仿效图层上的属性。“图层控制”在地图的整个工具集中操作，这时选择集或其它属性通过其 UI 更改。

此设计还允许您创建插入特定图层的工具。例如，城市规划者可能想要应用程序将探井仅插入水图层中，而将树插入蔬菜图层中。您可以创建一个自定义工具，选择时仅在适当的图层插入指定的符号或几何体。对常用工具也起作用，因此绘制多边形将始终在指定的图层上进行。还有，常用“图层控制”操纵工具集，使其象是插入在单独的图层上。

另请参阅第 110 页第 7 章中的 *MapInfo.Tools* 命名空间概述。

规划 Web 应用程序

创建 web 应用程序需要了解您的用户，特别是您想要以此应用程序服务的用户数量。在实际编写代码之前，web 应用程序开发的许多方面都需要做出设计选择。最重要的是要了解应用程序将如何使用，然后再去设计该应用程序。

在使用 MapXtreme 2005 设计分布式应用程序时必须考虑许多因素。有些因素是 MapXtreme 设计决定，其它因素是 Microsoft 技术设计模式。本章将让您了解一些必须做出的决定，并展示如何做出符合应用程序类型的正确选择。

其中一些决定是：

- 有多少用户正准备使用此应用程序？
- 采用有对象池模型还是无对象池模型？
- 应用程序如何处理状态：自动还是手动？
- 应用程序的初始状态是什么？
- 随用户而更改的详细情况是什么？
- 如何从应用程序获得优化性能？

我们将探讨做出每一个决定的过程的各个方面是怎样影响整个体系结构的，以及如何做出最佳的选择。体系结构决定正确后才能达到最佳的开发效果。体系结构决定错误将导致开发失败，必须重新开始。

有多少用户将访问您的应用程序？

开发基于 Web 的应用程序之前，您可以回答的最重要的问题是：预期有多少用户会访问这个站点？

此回答将会大大地影响应用程序的整个体系结构。如果选择了应用程序的基本原则，后来再更改到其它模式，那么基本上必须重写应用程序。因此一开始就做出正确的选择是极其重要的。

用户数目已知

一个明显的模式是如果您选择创建的应用程序的用户数目是在控制范围内的。通常是内联网站点，具有明确的用户群。在此种情况下，您可以有足够的硬件为确定数目的用户维持正常的响应时间。以此模型开发的益处是您可以使用 Visual Studio 中“新建项目”菜单中的 MapXtreme 项目模板。这些模板将创建基于 WebForms 的应用程序，在其中您可以象开发桌面应用程序一样开发应用程序。这是一个非常简单的编程模型，但是在可伸缩性方面有局限。我们将此类型的应用程序称为 **InProc** 模型。

用户数目未知

其它开发模型是当您不知道要访问站点的用户的数目时，您需要开发具有可伸缩性的应用程序。以后可能需要添加更多的服务器来处理附加负荷，因此应用程序必须了解如何保存用户当前状态，以及当下一个请求到达服务器时，何时及如何应用。这可能会发生在服务器群内相同或不同的服务器上。这种可伸缩性导致对系统如何存储状态、访问数据和响应多个请求而需要作出的许多选项和选择。因此，规划应用程序并创建一个强健的、支持分布式应用程序的体系结构是非常重要的。我们将此类型的应用程序称为有对象池模型。

InProc 应用程序开发模型

对于 InProc 模型，开发通常和桌面应用程序一样。这意味着每个用户耗用其自身进程空间，并且资源没有共享。对于开发目的来说这是一个简单的模型，但其伸缩性很差。要完成此类应用程序，可以简单地使用 Visual Studio 中包括的 MapXtreme 项目模板，并使用没有任何配置更改的 MapXtreme 对象模型。请参阅第 30 页第 2 章中的 *ASP.NET Web* 应用程序。

默认情况下，MapXtreme 2005 对象模型为每个用户创建一个 MapXtreme 会话。称为 InProc 会话。MapXtreme 会话是 DataAccess 引擎、MapFactory、CoorsSys Factory 和其它 MapXtreme 对象的保持者。这意味着应用程序开发人员不必生成全局对象来支持 MapXtreme 各部分。

Web 应用程序的 Web.config 通过设置来控制应用程序模型。对于 InProc 模型，Web.config 文件设置为：

```
<!--Use this setting to turn Session pooling on/off (true/false)-->
<add key="MapInfo.Engine.Session.Pooled" value="false" />
<!--Use this setting to save Session state automatically
(HttpSessionState) or manually (Manual)-->
<add key="MapInfo.Engine.Session.State" value="HttpSessionState" /
>
<sessionState mode="InProc"
stateConnectionString="tcpip=127.0.0.1:42424"
sqlConnectionString="data source=127.0.0.1;userid=sa;password="
cookieless="false" timeout="20" />
```

在此情况下，MapInfo.Engine.Session 对象没有被集中到对象池，Session 对象的状态不会自动存储在用户 Http 会话（键值 = HttpSessionState）中。Pooled 如果设置为 false，State 设置会因为 Session 对象用于指定用户而被忽略。

如果未指定 sessionState，则 InProc 为默认模式。其它可以设置的 sessionState 模式是基于服务器的 StateServer 或 SQLServer 模式。

Session 实例的生命周期取决于 Session State 配置。如果使用基于服务器的 Session State 且 Web.config 文件中的 ASP.NET sessionState 元素设置为 InProc，则在 ASP.NET 会话的生命周期内 Session 实例在内存中高速缓存。使用 InProc 时，状态永久性保存，设置保持在内存中直至应用程序结束，此时状态被写入 XML。

如果使用任何其它类型的 Session State 配置（例如，基于客户端或基于服务器的 sessionState 元素设置为 StateServer 或 SQLServer），则为每个 ASP.NET 请求创建和处置 Session 实例。有关如何配置 Session State 的详细信息请参阅第 66 页中的 *保存和恢复状态：示例*。

一旦决定要按照此选项的 InProc 会话，都将提供给每个访问 web 站点的用户一个 Session 对象的副本及其所包含的所有内容。很显然，如果用户的数目增加，则内存占用区也会增大。

应用程序也将加载一个背景地图，该背景地图与 MapInfo 工作空间一起自动加载。对于 InProc 模型，这意味着创建了新的 Session，并且当用户首次访问该站点时工作空间被加载。然后将控件传递到应用程序代码。您的代码必须只根据此指定给用户的背景的更改来考虑。请参阅第 64 页中的 [背景地图](#) 以了解关于加载背景地图及 Web.config 文件中的相关设置的详细信息。

对象池应用程序开发模型

“有对象池”模型需要应用程序处理资源的共享并记录状态。此方式中的选项也提供更多的选择，从而可以做出更多的决定。

对象池帮助减少并发 MapInfo.Engine.ISession 实例的数目，从而减少从零创建每个对象的耗用。需要对象时就从对象池中取出。如果不再需要对象则将对象放回对象池，等待下一个请求。MapXtreme 2005 中的对象池意味着可创建多个 Session 对象且该对象可用于 web 请求。

MapXtreme 2005 附带多个演示“有对象池”模型的示例 web 应用程序。ThematicsWeb 示例是有关状态管理案例研究的主题（请参阅[附录 E: 案例研究：在 Web 应用程序中管理会话状态](#)和[附录 D: 生成 Web 应用程序](#)中的教程）。

对象池帮助您用以下方法优化服务器的资源：

- 通过并行运行多个 ASP.NET 请求改进 web 应用程序的整个响应时间。
- 通过减少并发 Session 实例的数量来节约资源。
- 通过减少同时运行的请求（减少线程上下文切换）数量来最大化 CPU 的利用率。
请求要占用大量 CPU 时（例如 Map 图像导出）这就显得尤其有用。通常推荐的池大小为每个 CPU 1-2 个会话实例。

以下是要了解的关于规划 web 应用程序的一些重要术语。这些因素和决定点都是在使用“有对象池”模型开发应用程序所必须考虑的。

- **背景地图：**地图的该部分不会随用户的更改而更改。这可以是 StreetPro 数据，所有用户的都始终相同。
- **起始状态：**这是应用程序的起点，在此处决定根据用户是否到达该站点您需要执行什么操作。因为我们将使用池对象，实际上有四种状态要捕获：
 - 首次来到该站点的用户，以及新的对象池会话。
 - 首次来到该站点的用户，以及旧的对象池会话。
 - 再次访问的用户，以及新的对象池会话。
 - 再次访问的用户，旧的对象池会话。
- **池对象：**池对象可以处于两种状态之一。刚刚创建且加载有背景，或者是旧的，并在放回到池中之前其它应用程序使用过。

- **用户状态：**应用程序的此部分随每个用户而更改。这可以是一个选择集，用户 id 特定的表查询，或主题图地图。单个用户特定的应用程序的任何方面都在此术语中表示。

背景地图

开发应用程序时，必须判断哪些信息对于所有用户都是相同的。这包括图层、标注、标题和主题图。它们称为背景地图。创建背景地图的第一步是使用 MapXtreme Workspace 管理器创建一个工作空间，然后保存该工作空间用于加载到应用程序的 web.config 文件中。请参阅第 19 章：*Workspace 管理器*。

web.config 文件包括许多 MapXtreme 设置。我们感兴趣的是：

```
<configuration>
  <appSettings>
    <add key="MapInfo.Engine.Session.Workspace"
      value="C:\MIDATA\EvalData\WorldDetail\World_Detail.mws" />
  </appSettings>
</configuration>
```

此标记指示 MapXtreme Session 只要创建 Session 新实例即加载此工作空间。

当对象池 Session 返回到对象池，并被另一客户端请求时，此池对象不清除或重新加载该工作空间。这样可以增强性能，因为打开文件和创建主题图是一个非常费时的操作。

起始状态

优化的“起始状态”根据您正在写入的应用程序类型以及用户访问站点的方式。

基于 InProc 模型的应用程序不必占用不同的起始状态，因为每个用户以默认背景工作空间开始，然后基于它们的应用程序会话添加。每个用户的状态位于它们的 Session 对象副本中。

对于基于“有对象池”模型的应用程序，使用 ASP.NET 和 WebForm 可以将确定用户当前状态的逻辑放入 Page_Load 方法。使用以下设置激活 Web.Config 文件中的池对象：

```
<!--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="HttpSessionState" /
>
<sessionState mode="StateServer" stateConnectionString=
"tcpip=127.0.0.1:42424" sqlConnectionString="data
source=127.0.0.1;userid=sa;password=" cookieless="false" timeout="20"
/>
```

这两种设置可以组合为以下的选项：

- Pooled 为 true, State 为 HttpSessionState。
- Pooled 为 true, State 为 Manual。

池对象

设置 Pooled 为 True, State 为 HttpSessionState

此情况意味着将使用池对象，而每个用户的状态都将自动保存到 web.config（StateServer 或 SQLServer）中定义的会话机制。应用程序的程序员不必用编程方式保存当前用户的任何状态信息，例如图层、主题图、打开表或当前缩放。将保存 Session 对象的整个状态。

与 InProc 模型相似，此选项保存系统资源，因为并不是每个用户都有一个 Session，而所有用户将共享对象池 Session 对象。缺陷是当前 Session 的所有状态都将串行化为当前定义的状态机制。通常情况下，Session 对象在工厂中检索所有的对象，并将它们串行化为该状态。这包括所有的地图和它们所包括的图层，所有打开的表定义和任何加载的投影。Session 对象不了解单个用户的状态，而是存储所有可用的信息。这包括在不同用户之间不会更改的图层，即背景地图。这会是一个花费时间的过程，可能会造成此类应用程序比 InProc 模型执行的要慢。有关串行化的信息请参阅[第 140 页第 9 章中的串行化](#)。

设置 Pooled 为 True, State 为 Manual

此配置意味着 Session 对象将是对象池，依靠应用程序的程序员来保存每个用户的状态需要。Session 对象不会将其任何状态存储为当前定义的状态机制。

此配置提供性能和可伸缩性的最好效果。在此模型中，您使用的是池对象，Session 不保存任何状态，而池对象返回到“不干净”的 COM+ 池。通过这些可以认识到池对象对当前用户执行的背景工作空间作出了修改，需要您清除该状态，将其正确设置以备下一个用户使用。可以选取在 web 应用程序请求开始时清除，或在结束时清除。

用户状态

应用程序从池中检索时必须考虑用户的当前状态和池对象的状态。在此种情况下您必须决定：

1. 用 Page_Unload 方法将池对象清除后返回到池中？
2. 只是将池对象保持原样，而用 Page_Load 方法重新设置任何状态？

正确的选择不一定很明显，因为各个用户的状态都各有不同，可能导致会话对象未清除就放回到池中。其它应用程序可能会通过等待 Page_Load 清除或检查 Session 是否需要清除而节省时间。

保存和恢复状态：示例

本节讲述 Web Form 应用程序的一个示例，该应用程序的作用是检查状态是否实际已经恢复。基本模板是：

```
Private Sub Page_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    Me.RestoreState()
End Sub

Private Sub Page_UnLoad(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Unload
    Me.SaveState()
End Sub
```

只要请求作为 Page Load 到达服务器，代码将恢复状态。只要页面卸载事件发生，该应用程序将保存允许用户更改的任何状态。对于基本的 web 地图绘制模板，这通常意味着保存当前缩放和中心，以便用户可以在应用程序服务当前的请求之前回到以前的状态。

在以下的示例代码中，应用程序创建 TitleAdornment 来保存当前 SessionID，形象地证明当前用户的状态正在更改。如果用户已经访问过站点以及如果这是一个清除后的池对象，会返回两种方法。

确定用户是否到过站点的方法是简单地检查任何已保存变量的 Session 状态。如果缺少一个则用户尚未访问过站点。不干净的池对象的检查只是简单地查看地图的 Adornment 集，查看是否添加有内容。如果有修饰，则这是一个不干净的池对象，否则对象是清除过的。

```
Private Function IsUsersFirstTime() As Boolean
    Return Me.Session.Item("Center") Is Nothing
End Function

Private Function IsDirtySession() As Boolean
    ' add a method to check for an item you pragmatically modified
    Dim ret As Boolean = False
    Dim map1 As Map = MapInfo.Engine.Session.Current.MapFactory(0)

    If Not map1 Is Nothing Then
        ret = map1.Adornments.Count > 0
    End If
    Return ret
End Function
```

因为在应用程序中有“图层控制”，所以“图层”集合可以被修改。因此需要将此添加到 SaveState 方法中的状态。

```
Private Sub SaveState()
    Dim map1 As Map = MapInfo.Engine.Session.Current.MapFactory(0)

    If (Not map1 Is Nothing) Then
```

```
    Me.Session.Item("Zoom") = map1.Zoom  
    Me.Session.Item("Center") = map1.Center  
    Me.Session.Item("Layers") = map1.Layers  
End If  
End Sub
```

恢复状态需要大量编码，因为我们必须考虑起始状态条件。基于当前应用程序所发生的情况以及用户的访问，可以确定需要做的操作。基本上这是判断要恢复哪些内容，以及是否需要创建 TitleAdornment。

```
Private Sub RestoreState()  
    Dim map1 As Map = MapInfo.Engine.Session.Current.MapFactory(0)  
  
    If (IsUsersFirstTime()) Then  
        If (IsDirtySession()) Then ' Set the Session back to the default  
  
            map1.SetView(MapInfo.Engine.Session.Current.Catalog.GetTable("Ocean"))  
            Dim adorn As MapInfo.Mapping.TitleAdornment =  
                CType(map1.Adornments.Item(0), MapInfo.Mapping.TitleAdornment)  
                adorn.Title = "Welcome " + Me.Session.SessionID.ToString()  
                ' We also need a way to set all the layers back to the  
                ' default. This code has to be  
                ' aware of the controls the user has access to and  
                ' be able to get back to a default state.  
        Else  
            Dim ta As New MapInfo.Mapping.TitleAdornment(New  
System.Drawing.Size(100, 50), MapControl1.Map)  
            ta.Title = "Welcome " + Me.Session.SessionID.ToString()  
            MapControl1.Map.Adornments.Append(ta)  
        End If  
    Else  
        ' User is coming back  
        If (Not IsDirtySession()) Then ' Did they get a clean pooled object  
            Dim ta As New MapInfo.Mapping.TitleAdornment(New  
System.Drawing.Size(100, 50), MapControl1.Map)  
            ta.Title = "Welcome " + Me.Session.SessionID.ToString()  
            MapControl1.Map.Adornments.Append(ta)  
        End If  
        Dim adorn As MapInfo.Mapping.TitleAdornment =  
            CType(map1.Adornments.Item(0), MapInfo.Mapping.TitleAdornment)  
            adorn.Title = "Welcome " + Me.Session.SessionID.ToString()  
            ' now grab settings out of the current users Session  
            map1.Zoom = CType(Me.Session.Item("Zoom"), Distance)  
            map1.Center = CType(Me.Session.Item("Center"), DPoint)  
    End If  
End Sub
```

Restore 正确地处理所有的 Beginning 状态条件，以后也不会失败，因为该应用程序不处理已有用户获取新池对象的情况。

请注意 SaveState 方法存储地图的图层集，但是 RestoreState 方法不参考图层集。某些 MapXtreme 对象比较复杂，需要帮助程序类来实际进行 Serialization 的工作。请参阅 [第 140 页第 9 章中的串行化](#)。

以下类可以放入 Session，并且当用户的 Session 激活时这些类将反串行化其自身。不必恢复它们的状态。

反串行化其自身的类

AreaStyle	IndividualValueLabelTheme	PieTheme
BarTheme	IndividualValueTheme	RangedLabelTheme
BitmapSymbol	LabelLayer	RangedTheme
CartographicLegendFrame	LabelModifier	RasterStyle
Catalog	LabelSource	ResultSetFeatureCollection
CompositeStyle	LabelSources	ScaleBarAdornment
CoordSysFactory	Layers	Selection
DotDensityTheme	Legend	Selections
FeatureCollection	LegendFrames	Session
FeatureLayer	Legends	Table
FeatureOverrideStyleModifier	LineStyle	TextStyle
FeatureStyleModifier	Map	ThemeLegendFrame
FillStyle	MapFactory	TitleAdornment
FontStyle	MIGmlFeatureCollectionModifiers	UserDrawLayer
FontSymbol	MultiFeatureCollection	VectorSymbol
GraduatedSymbolTheme	MultiResultSetFeatureCollection	RasterStyle
GridStyle	NamedConnectionInfoCollection	
GroupLayer	ObjectThemeLayer	

有关如何在 web 应用程序中运用示例状态管理，我们提供一个基于 MapXtreme 2005 附带的 ThematicsWeb 示例应用程序的案例研究。请参阅第 440 页附录 E 中的案例研究：管理状态。

对象池会话和状态

Session 是所有基于 MapXtreme 2005 的应用程序的起点。它管理 MapXtreme 2005 应用程序所需资源的初始化。Session 还提供对其他重要对象的访问如 Data.Catalog、MapFactory、CoordSysFactory、Selections 及其。要访问会话，请调用 Session.Current() 方法。进程中的每个线程都有一个 Session 对象。每个线程只能有一个会话且会话不可以被多个线程共享。

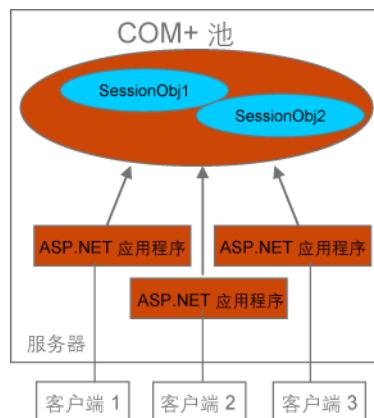
Web 应用程序可以是每个用户都有一个会话对象，或多个用户有一个对象池会话。Session 的一些特征包括：

- Session 实例可以是预创建的和对象池的。
- 使用通过 .NET 框架可用的 Microsoft COM+ 对象池。
- 可配置（池大小、超时、初始工作空间等）。
- 会话可以默认返回到“不干净”的池。

在下一节中逐个讨论。

会话实例（预创建的和对象池的）

Session 对象可以是对象池的，以便特定 web 应用程序的所有用户能够共享特定数目的对象池会话对象。如下图所示。



三个客户端的每一个都访问两个池对象中的一个。这意味着返回到某个用户的池对象可能（非常有可能）曾经被其他用户使用过。这意味着您必须做出一些状态管理决定。您可以使用自动状态管理，使 MapXtreme 管理当前用户状态，或者可以使用手动状态管理，控制必须为每个用户保存和恢复的更改。请参阅第 65 页中的池对象。

这些池对象用初始状态创建，初始状态在 MapXtreme 应用程序开发中指背景地图。这包括不随用户而更改的图层、标注、标题和任何主题图。使用 Web.config 可以指定要加载的初始工作空间，只要创建 Session 对象（池或非池），此工作空间将被加载。这提供给应用程序一个已知起点。

Microsoft COM+ 对象池

Session 对象用系统上的 COM+ 服务注册。此系统处理任何已注册池对象的配置和激活。运行时安装程序具有注册和创建 MapInfo Session 对象默认设置的逻辑。默认情况下，Session 对象用两个池对象创建，并且 60 秒会超时。

配置池大小

MapXtreme Session 可以使用系统配置方法对适当部分进行配置。这些设置在“控制面板”的系统对话框中和应用程序的 .NET 配置文件中都有。Web 应用程序在使用池对象时必须管理这些设置。

一个重要的设置是 COM+ 系统所创建的池对象数目，用于服务运行的应用程序。此设置可使用**控制面板 -> 管理工具 ->** 和“组件服务”来访问。浏览**组件服务 -> 计算机 -> 我的电脑 -> COM+ 应用程序 -> MAPINFO.COREENGINE -> 组件**。在此处您将会找到 MapInfo.Engine.Session+PooledSession 对象。右键单击该图标并选择“属性”。

该对话框可用于设置各种不同的属性，包括每个应用程序的池对象数目和会话创建的激活超时。池对象数目的正确设置直接影响应用程序的性能。要正确设置池对象的数目，需要了解应用程序及它是如何访问数据的。请参阅第 74 页中的优化应用程序中关于如何确定池对象的优化数目的讨论和详细说明。

可以直接影响应用程序性能的另一方面，是应用程序 Web.config 文件中定义的启动工作空间。此工作空间定义在创建会话对象时将创建哪些地图、图层和表。在 COM+ 池线程中及 web 应用程序空间之外会发生此种情况。应用程序启动时，会向应用程序池请求一个会话对象。COM+ 池会创建在控制面板“组件服务”对话框中指定数目的池对象。该会话对象读取 Web.config 文件，在应用程序从池中得到实例之前加载指定的工作空间。因此，如果加载工作空间花费的时间比“组件服务”对话框中指定的超时时间还要长，将会接收到 COM+ 激活异常。因此非常重要的一点就是要了解工作空间中要加载的内容，以及完全加载需要多长时间。

如果指定“最小池大小”为 2，则 COM+ 池将为池对象创建首次请求的两个副本。当 COM+ 池发现超时等待池对象时，它将创建更多对象最多达到最大设置。如果工作空间非常复杂，需要花费大量时间来加载，则在应用程序激活状态下，后续对象的激活可能会造成超时错误。Web 应用程序池可以通过设置来控制对象池应用程序的再次激活，包括回收对象池类的设置。因此在某些点，应用程序和池对象可能会回收，这将造成其它工作空间加载。必须判断哪些地图、表和图层是重要的，要作为应用程序的背景，并通过 Web.config 文件的工作空间加载。不过大的工作空间需要时间来加载，需要注意其产生的影响。要测试加载时间，只需简单地加载服务器上 Workspace 管理器中的工作空间，来确定实际的加载时间和调节相应的超时设置。

初始工作空间在加载不随用户而更改的地图、图层和表时非常有用，但是必须注意加载复杂工作空间的影响。

将会话对象返回到池

在应用程序中使用对象池 Session 对象和手动状态管理时，请求页面完成之后，该会话停用，Session 对象以当前状态返回到池。这意味着第一个用户如果获取世界的默认地图并缩放至欧洲，此池对象的下一个请求将得到一个当前以欧洲为中心而进行缩放的地图。该应用程序负责保持每个用户的状态，并设计为处理当前用户的状态，就象处理一个完全干净的（新创建的池对象）或不干净的会话（之前用户使用过的）一样。

做法是检查当前会话是否在初始工作空间中。如果是，则得知其已经被其他用户修改。有关如何执行该操作的示例请参阅[第 66 页中的保存和恢复状态：示例](#)。

如果使用的是池会话和自动状态，则不必考虑是干净的会话还是不干净的会话。整个会话将为每个用户保存并在用户访问 web 站点时恢复。但是使用此方法要考虑到性能。如果工作空间很复杂，有许多不会更改的图层和主题图，则保存和恢复状态会很花费时间。仅保存随用户而更改的状态会更有效。如果有 50 个图层，而其中 45 个对不同用户始终是相同的，则将其共用的图层保存到会话状态效率会很低而且也不必要。

应用程序的体系结构的理念应当是包含应用程序数据的共享部分和用户的特定部分。它们应当是隔离的，而且应用程序必须可以管理这些用户特定图层。

MapXtreme 会话的可配置性很高，设计为可以与 Microsoft COM+ 和 web 技术一起使用。要了解如何最好地使用此会话，需要了解 Microsoft web 传输体系结构。有多种途径可以获得更多有关上述主题的信息。以下是一些有关 web 体系结构信息的链接：

IIS 6.0	<i>Internet 信息服务 和 Internet 通信</i> http://www.microsoft.com/technet/prodtechnol/windowsserver2003/library/W2K3InternetMgmt/5b36c13b-c72e-4488-8bbe-7e4228911c38.mspx
COM+ 对象池	<i>COM+ 对象池概念</i> http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cossdk/html/74a45220-449a-4d89-a979-a206e5e3d3ad.asp
IIS 会话状态	<i>选择维护和存储 ASP.NET 会话状态 (IIS 6.0) 的方法</i> http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/Library/IIS/ccca60f3-8175-4005-8b68-d3748aea8623.mspx

应用程序性能

性能始终是最难以预估的方面。每个应用程序都有特定细微差别，可以更改整个性能。本节解释一些服务器设置如何影响性能，以及如何对不同类型的基于 web 的应用程序应用这些设置。

请注意下面的章节是参考 IIS 6.0，在基于 Windows 2003 Server 的系统上运行。提供了最好的性能选项。本节不描述 XP 系统上 IIS 5.0 的性能问题。

本节中描述的测试在 MapInfo 性能实验室进行。

CPU 和内存

CPU 和内存对任何服务器而言都是最重要的性能，两者如果能够更大都将改进性能。不过硬件设置通常都有限制，所以需要有好的启发式设置来确定所需要的最重要的硬件。

理想情况下，当设计一个系统时，应当首先考察用户的数目和应用程序需要的空间，然后查看所需要购买的硬件。

常规性能测试

测试摘要

本节描述的测试是性能的基本起点。可以查看相对 MapInfo StreetPro 数据集的原始渲染速度。在后面的示例中，我们在添加应用程序逻辑之前将这些 StreetPro 渲染作为背景数目。

本测试中所使用的系统是 Windows 2003 Server，具有双 3.2GHz 超线程处理器和 2048 MB RAM。本测试中使用的数据由 MapInfo StreetPro US 版本 5.3（MapInfo .tab 格式）组成。测试中生成的地图图像大小为 200 x 200 像素。

为了提供系统时间来启动和优化其自身，在结果集合测试之前先进行预热测试。要模拟加载，会生成许多并发连接，所有这些因素使地图的模拟请求超过几个小时。并发连接接收到完整的地图后，会立即请求另一个地图。

请注意并发连接与并发用户并不相同。并发连接代表模拟请求，可以实际代表成千个用户，因为不是所有的用户会同时请求地图。还要特别注意地图中演示的数据量以及地图的大小和密度对系统的性能而言非常重要。这些属性可以正面也可以负面影响系统性能。

本测试中，每个并发连接随机选择缩放级 5、10、15、20、50 和 100 英里。因此，地图内容并不相同，尽管生成地图的实际坐标系通常是最重要因素。

结果摘要

这些测试运行中，并发连接（随机选择稀疏和详尽地图）每秒生成 6 个地图。MapInfo 性能实验室中进行的测试案例也指明处理器越快则生成的地图越快（即，双处理器速度将使每秒生成的地图数量翻倍）。

吞吐量 - MapXtreme 2005 可伸缩性

并发连接	每秒钟事务处理	CPU 使用百分比	使用的 RAM
10	6.62	95.8	395.9
20	6.68	98.7	405.2
30	6.71	98.7	414.6
40	6.68	98.8	420.6
50	6.68	98.7	427.7
60	6.65	98.8	431.6
70	6.67	98.8	435.6
80	6.69	98.8	447.7
90	6.64	98.8	456.2
100	6.63	97.5	462.9

优化应用程序

有对象池和无对象池会话和状态

本节讨论池对象的重要性原因和使用它们的原因。请参阅第 69 页中的对象池会话和状态中关于如何使用 Session 和 State 的描述。

将池对象作为应用程序的共享资源来使用。这会有助于限制应用程序的内存要求。

在无对象池会话情况下，如果有 100 名用户必须同时访问服务器，则每个用户有相同的工作空间，除了图层和表之外。每个用户的会话对象可以占用 50 MB（作为示例），而用户的可变图层占用其中的 1 MB。在此情况下，将需要内存总量的 5,000 MB。

用户数目	*	会话内存	=	需要内存总量
100	*	50 MB	=	5,000 MB

作为比较，在 Pooled Session 中两个池对象在 100 名用户中共享（相同条件下，每个用户必须保存一个单一图层和表），内存总量将下降到 200 MB。

池内存	+	100 用户图层	=	需要内存总量
100 MB	+	100 MB	=	200 MB

可以发现，在第一种情况下每个用户的 Session 都是不可测量的。但是第二种情况中可以仅存储每个用户的需要。这样在用户数目增加时管理系统资源更加容易。如果用户数目从来没有超过十个，则不值得费力去管理状态信息。但是如果要向更多的用户开放此应用程序，可能需要重写该程序才能使用。

要了解如何设定应用程序中的池对象数目，必须了解应用程序的性质及其访问数据的方式。我们建议根据从本地 TAB 文件访问本地数据，每个 CPU 分配两个对象池。

第一种情况（无对象池会话）下，直接访问数据，CPU 完全利用（运行程序时使用率为 90% 或更高）。测试时每个 CPU 分配两个进程来限制上下文切换的数量。当进程大量占用 CPU 时，多个池对象意味着操作系统必须在它们之间进行切换，这将降低性能。进行性能测试时发现，如果进程不等待数据时，每个 CPU 使用两个池对象会提供最好的性能。

在有对象池情况下访问远程数据时，CPU 使用率在 50% 至 70% 之间。这意味着当数据被远程服务器捕获时，进程可能处于等待状态。在此种情况下，可以增加池对象的数目，以此来确定 CPU 在 web 服务器上的使用率是否增加。通常情况下这会提供更好的性能，因为在线程之间的上下文切换不会影响到处理数据，可以用附加的循环来完成此任务。如果正接受数据访问的服务器可以处理多个请求的附加负载，那么也将提高性能。

调节应用程序的最好方法是观察 Web 服务器和数据库服务器 CPU 使用率，并调节服务器上的池对象，使这些数字达到最大。如果数据库服务器具有专用的 CPU，则增加池对象将很有可能没有效果。如果数据库服务器没有超负荷工作，则增加池对象数目并观察是否可以获得更好的性能。此方法需要一些时间，要注意与 CPU 使用率有关的服务器设置。

第三种测试情况包括访问 WMS 服务器，在其中所有的数据访问和渲染都是在隔离于 IIS 系统的服务器上进行的。在此种情况下，增加池对象的数目肯定会提升性能。因为只有两个线程可以随时同时从 IIS 服务器访问 WMS 服务器，CPU 使用率在 30% 至 40% 之间。大多数时间，在等待 WMS 服务器返回图像。Web 应用程序将大大地减慢。

当使用桌面客户端直接访问 WMS 服务器时，会快速返回。这意味着 WMS 服务器的响应要快于 IIS 服务器在两个池对象之间的上下文切换。就是说 WMS 服务器的时间都花在等待上。通过增加池对象的数目（我们的示例中有 5-10 个池对象），可以增加服务器的吞吐量。必须调节应用程序，因为我们执行的惟一操作是访问 WMS 服务器。

可以看出一个方案适合所有池对象调节是不太容易的。通过了解应用程序和数据访问方法，以及测量 CPU 使用率，可以获得最好的结果。

IIS 服务器设置

IIS 有许多设置用于微调应用程序。以下 MSDN 文章详细讨论了 IIS 体系结构：

[IIS 6.0 文档 \(IIS 6.0\)](http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/Library/IIS/848968f3-baa0-46f9-b1e6-ef81dd09b015.mspx) <http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/Library/IIS/848968f3-baa0-46f9-b1e6-ef81dd09b015.mspx>

特别请查看《IIS 6.0 部署指南》，在附录 A 会看到多种配置设置的详细解释。

用于查找性能技巧的一个网址是：

[性能调节 \(IIS 6.0\)](http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/Library/IIS/71490aae-f444-443c-8b2a-520c2961408e.mspx) <http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/Library/IIS/71490aae-f444-443c-8b2a-520c2961408e.mspx>

有许多在线资源讲述关于 IIS 6.0 的信息。以下章节描述特用于 MapXtreme 应用程序的 IIS 设置。

应用程序池

Microsoft 将应用程序池定义为“一组由工作进程服务的一个或多个 URL”。IIS 6 中的应用程序池可用于：

- 隔离独立 web 应用程序以便某个应用程序不会带给其它应用程序不好的影响。
- 将 URL 分组以在相同的 ID 标记下运行。
- 控制组的性能和健康状况。

可以使用 IIS 管理器创建“应用程序池”。创建之后可以开始将“虚拟”目录分配给特定池并将它们配置到特定应用程序。查看池的常规设置时会看到以下分类：

- **回收：**管理工作进程的回收。在工作进程隔离模式中，可以将 IIS 配置为在应用程序池中定期重启工作进程，允许您精确管理错误的工作进程。这确保这些池中指定的应用程序仍然是健康的，系统资源可以恢复。
- **性能：**配置 Internet 信息服务 (IIS) 关闭闲置工作进程的方式、等待进程的请求数目、CPU 监控，及 Web 集群中的进程数目。
- **健康状况：**配置工作进程健康状况监视。可以设置工作进程通信限制，确定如果大量进程快速失败时的操作过程，并设置启动和关闭时间。
- **标识：**应用程序池的标识是在应用程序池工作进程下运行的帐户的名称。

回收用于管理员配置回收此池的工作进程的条件。通常，可以在设定的时间之后回收进程或在指定的时间或根据内存的使用回收请求。可以独立地配置每个进程或根据多个条件配置。

我们的测试显示回收很大程度上是取决于站点的流量和系统观察值。回收并不意味着服务的中断，当前用户将完成正常的回收，然后进程会关闭并启动新的进程。回收设置最好根据系统的观察值和使用来设置。可能需要根据较慢的流量时间或根据大量占用内存的应用程序所需的内存来回收。对于长时间运行的进程而言回收是必要的，应用程序池使回收很容易就可以完成。

性能用于管理员设置选项来关闭不活动进程、限制排队请求的数目、监视 CPU 使用并进行操作，并为单一池创建多个工作进程。每一项都有其对运行系统的影响。

我们的测试显示当服务 WMS 请求时，等待 WMS 服务器响应会造成基于默认设置的线程池的饱和。当测试系统压力时，您会注意到响应时间很慢，并且服务器 CPU 没有完全利用。MapInfo 性能试验室测试显示当运行压力测试时 CPU 使用率为 35%。这意味着 CPU 大多数情况下在等待事件的发生。

如果增加对象池的数目，会释放更多的线程来用于服务这些请求。这会提供更好的性能数目，但是感知性能会更差。即，如果打开 web 浏览器并查看相同的页面，将会有很长的延迟。在此情况下，基于一个工作进程和其可用线程排队将产生延迟。增加工作进程的数目之后，纯性能降低，但感知性能会好一些。独立的 web 请求不必在长队列中等待，可以跳到较短的队列。响应会更快，但是因为工作进程中的上下文切换，在性能方面会有一个小的权衡。

健康状况使得管理员可以对系统做自我诊断。设置可以用于检查运行进程以查看是否仍然激活。如果是在定义的绑定之外，还可以检查失败和回收的特定数目。也可以设置时间限制，即启动和关闭所需花费的时间。

标识用于管理员设置此应用程序池在哪个用户 ID 标记下运行。如果某些运行的应用程序需要不同的凭据而不是其它情况（典型案例），则会非常有帮助。

通常情况下，可以通过更改“性能”部分的设置来获得最好的性能。请记住，这些设置是特定应用程序的，改进某个应用程序性能的设置可能会影响其它应用程序的性能。

通过有选择地保存对象来进行优化

MapXtreme 2005 通过 ASP.NET 框架自动处理保存或恢复应用程序的状态，此时设置为 HttpSessionState。这种情况下，整个 Session 对象被串行化。从优化服务器的性能角度来讲，这不是一种理想的方法，因为这会保存和恢复所有的内容（即，Catalog、Maps/Layers、Selections 等）。保存整个 Session 对象时允许快速实现，但包括了许多不必要的耗用。

如果将 State 设置为 Manual，则 MapXtreme 2005 不自动保存或恢复任何状态。这允许用户有选择地选择 Session 内哪些对象真正需要保存和恢复，因为用户可能更改这些对象。可以通过将要保存和恢复的对象放置到 ASP.NET Session 对象中，完成不同对象的选择集。例如，以下代码片段保存和恢复 Map 中所有 Layers 的状态：

```
// Save the map's layers collection  
Session["Layers"] = map.Layers;
```

注： ASP.NET 框架自动恢复应用程序的状态。到调用 Page_Load() 方法时，所有串行化的对象已反串行化。

检查应用程序以确定哪些对象需要串行化。有关哪些对象需要串行化的决定取决于应用程序允许用户更改的对象。例如，如果允许用户更改图层上的属性，创建或删除图层或主题，则应用程序需要串行化整个 Map.Layers 集合。如果应用程序仅允许用户平移和缩放，则只有地图的中心和缩放值需要串行化。最简单的方法是串行化整个 Map 对象。不过这样会保存 Map 上的每个属性和所有 Map 的 Layers。如果应用程序仅更改地图的中心和缩放，则这样做就有点过于复杂。处理这种情况的一种方式是串行化地图的中心和缩放，然后在 Page_Load() 方法中重新应用这些值。该方法比保存整个 Map 对象更有效率，因为这样串行化的信息远远少于后者。

应用程序示例

WMS 客户端性能

一个有趣的案例是从应用程序绘制 WMS 图层。在此情况下，绘制在独立的服务器上完成，而 Web 应用程序服务器等待来自 WMS 服务器的响应。示例应用如下所述。

WMSAsUserDrawLayer: 此应用程序使用实现为 UserDrawLayer 的类直接与 WMS 客户端程序集通信，并将返回的 WMS 服务器图像使用 GDI+ 直接绘制到地图并绕过 Data Access 引擎。

WMSTABLAYER: 此应用程序使用标准 TAB 图层访问方法。

测试	池对象数目	用户数目	反应时间	吞吐量 (Silk)	吞吐量 (实际)	传输总量	% CPU
WMSAsUserDrawLayer	1-2	1	0.15	1.85	7.4	6666	7%
WMSAsUserDrawLayer	1-2	10-100	3.68	3.75	15	13540	18%
WMSAsUserDrawLayer	5-10	10-100	2.38	5.79	23.16	20911	40%
WMSAsUserDrawLayer	15-20	10-100	2.38	5.77	23.08	20841	30%
WMSTABLAYER	1-2	1	0.22	1.14	4.56	4117	8%
WMSTABLAYER	1-2	10-100	2.91	4.73	18.92	17078	35%
WMSTABLAYER	5-10	10-100	1.53	8.98	35.92	32049	95%
WMSTABLAYER	15-20	10-100	1.64	8.38	33.52	30266	99%

有趣的是当池对象数目由 1-2 变为 5-10 时 WMSTABLAYER 行中的性能增强。当应用程序用 1-2 池对象运行时，CPU 的使用率会非常低。当池对象数目增加时，OS 可以生成更多的工作线程。效果为使许多线程等待 WMS 服务器响应。这使 CPU 可以在不同的时间进行字段请求，而不会阻断在一或两个线程上，从而使服务器的吞吐量更大。

有趣的是，当 WMSTABLAYER 应用程序的对象池增加到 15-20 时，会生产出更少的地图。这是因为 CPU 已经达到极限值，而线程之间切换的附加耗用实际上已经阻碍了运行进程。

该表对重要的事实进行图解：当服务器正运行 MapXtreme 应用程序时，CPU 不会达到其极限值，它帮助添加更多的池对象，以便操作系统可以创建更多的工作进程来处理请求。但是，有一个很好的平衡可以为任何特定的应用程序优化这些数字。应用程序程序员的责任是了解应用程序的行为，并调节服务器设置以匹配特定的应用程序逻辑。

规划数据模型

将数据放入应用程序可能是一项艰巨的任务，特别是当资源多种多样时。MapXtreme 有各种方法来通过 MapXtreme 数据库系统访问和分析数据。

设计一个考虑周全的数据访问模型与应用程序设计的其它元素同样重要。本节描述：

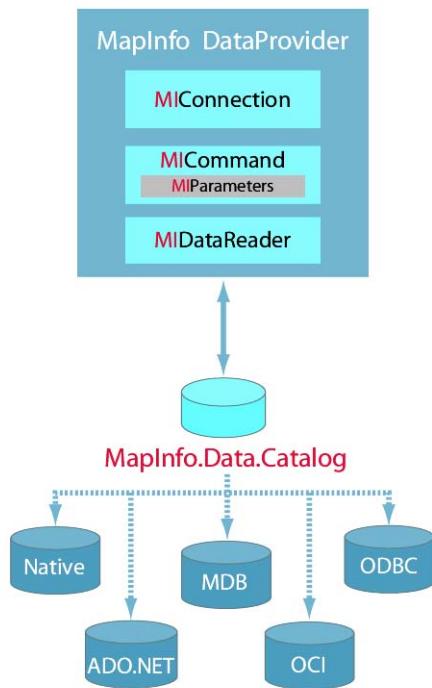
- MapXtreme 数据体系结构
- 选取正确的数据源
- 访问数据的方法
- 相关表

MapXtreme 数据体系结构

MapXtreme 数据访问由 Catalog 驱动，Catalog 是会话中当前所有打开表的库。使用 Catalog 打开、创建新的或参考打开的表。然后使用 MapXtreme ADO.NET 接口来查询这些表，并返回记录集。

表是 MapXtreme 使用的基础数据容器。每个表是指数据库中特定的行集合。表可以是可制图的并有一个空间组件，或是不可制图的，仅包含数据列。MapXtreme Catalog 可以打开两种类型，并在查询或连接中使用。连接可以用于将可制图组件添加到您自己的数据，并在地图和空间分析中使用这些连接的表。添加空间组件稍后讨论。

MapXtreme Catalog 通过可以访问 Catalog 的 MapInfo ADO.NET 数据提供方公开。访问表和结果集通过此接口控制。下图显示 ADO.NET 接口和 MapXtreme Catalog 及内部数据源之间的关系。



ADO.NET 图层是进入 MapXtreme Catalog 的接口。Catalog 在会话中，具有一个包含所有当前打开表的列表。Catalog 还有一个 SQL 引擎，可以用于选择、插入、更新和删除表中的表和数据。

SQL 引擎还允许 Catalog 中定义的任意表之间的连接。这意味着可以连接 Catalog 中的任何 DataSource（从本地连接到 SQLServer，或从 SQLServer 连接到 Oracle）。当组织来自不同的数据源时，这就成为一个有力的工具。这也将增加在数据访问设计模型时可以进行的选择。在以后的章节中我们将讨论不同情况下的最好方法。

MIConnection 类表示到 Catalog 的连接。使用此类来创建 MICommand 类。MICommand 提供了必要的接口来在 MapInfo 数据提供方上执行 SQL 命令。MICommand 创建 MIDataReader 和 MIScrollableReader 实例，分别通过 ExecuteReader 和 ExecuteScrollableReader 方法来获取数据。MIDataReader 提供了从 MapInfo 数据提供方读取行的前向数据流的方法。还有一个 MIScrollableReader 可以前向和反向记录读取以及重新读取整个记录集。请参阅第 170 页第 10 章中的 *MapInfo ADO.NET 数据提供方*。

MapXtreme Catalog 管理和联合表的集合。用户将其作为一个集合来访问，但它可能由多个表组成。每个表使用内部数据提供方体系结构访问它代表的数据。访问远程数据库管理系统的表将使用 DBMS 本地语法来执行查询或编辑，代表表和内部数据提供方为特定数据库来处理语法的转换。

选取正确的数据源

MapXtreme 2005 附带了多种数据源。选取正确的数据源可以使应用程序性能有很大的不同。在某些情况下您没有选择，例如本地 MapInfo 文件 (.TAB)，而在其它情况下可能会有多种选择。大部分情况下，您使用的是支持的数据提供方；而在某些情况下，使用这些提供方却无法访问数据。如果是这种情况，可以使用 ADO.NET 数据源。此数据源可以用于 Catalog 使用 ADO.NET DataProvider 接口来检索数据。

此接口是一个功能强大的工具，可以访问非标准格式的数据。例如您可以创建自己的数据提供方，访问 LDAP 数据库上的数据，这种提供方可能需要多种查询，并使用 ADO.NET 接口将值返回到 Catalog。Catalog 然后会向数据提供方询问数据行，并象其它表一样来处理。然后使用 Catalog 来查询此表或将其连接到任何其它表。用此方法，可以将外部数据连接到任何可制图的表，来生成空间查询或主题图。

如果数据位于 .TAB 文件中，可以简单地打开它们，使用 MapXtreme 查询工具提供的任何一种工具（例如查询对象和 ADO.NET 接口）来访问该数据。打开表之后，以后就可以通过 Catalog 使用，并将在 Catalog 中保持打开，直至通过 Table.Close 或 Catalog.Close 将其关闭。

支持的数据源

下表描述了 MapXtreme 2005 附带的数据源。

类	TableType	描述
TableInfoNative	本地	MapInfo 本地表。表可以具有关联的 MAP 文件，文件内包含几何体和样式信息。
TableInfodBase	dBase	表将数据存储在 dBase 文件。表可以具有关联的 MAP 文件，文件内包含几何体信息和样式信息。
TableInfoMSAccess	Access	表将数据存储在 Microsoft Access 数据库表中。表可以具有关联的 MAP 文件，文件内包含几何体信息和样式信息。
TableInfoAscii	ASCII	表将数据存储在分隔的 ASCII 文本文件。表可以具有关联的 MAP 文件，文件内包含几何体信息和样式信息。
TableInfoServer	服务器	基于远程 DBMS 查询的表具有特定于服务器表的属性和行为。例如，服务器表需要连接字符串和查询。

类	TableType	描述
TableInfoSeamless	Seamless	Seamless 表用相邻地理组合两个或多个基表。该表显示为单个地图图层。
TableInfoRaster	光栅	包含光栅图像的表。
TableInfoGrid	网格	包含网格图像的表。
TableInfoWms	Wms	包含 WMS 服务器生成的图像的表。
TableInfoShapefile	Shapefile	ESRI Shapefile 表。
TableInfoMemTable	MemTable	内存中的临时表。表可以包含几何体和样式列。
TableInfoView	视图	具有命名 SQL Select 语句的表。出现在选择列表中的表达式形成了表的列和记录，where 子句组成表成员的条件。
TableInfoResultSet	ResultSet	目前 ResultSet 表专用于 IResultSetFeatureCollections。
TableInfoAdoNet	AdoNet	由导入到 Catalog 中的客户端数据组成，并公开为表。客户端数据可以使用多种形式的 ADO.NET 对象提供。只有不可绘制地图的部分才能提供到 AdoNet 表，不过可以将 SpatialSchema 应用到表定义，使得该表可以绘制地图。

每个数据源都有特定的性能特性。本地表提供最佳的访问和地图绘制时间。数据存储在本地系统中，为当前操作而优化。其它基于文件的表类型性能很好，这取决于当前硬件和文件大小。有关详细信息，请参阅[第 72 页中的应用程序性能](#)。

数据源的数据打开之后，它就代表着 Catalog 中的表。现在可以对这些表进行查询，以返回记录集或连接表。这是一个功能强大的工具，您现在可以提取一个 SQLServer 表并将其连接到 Oracle 表。SQL 过程和连接这些表的工作在 Catalog 中执行。结果记录集可以作为单一光标访问，而对独立数据提供方的后续请求由 Catalog 分配到相应数据源。您要做的就是在结果集中滚动查找。

通常有两种访问数据的基本方法，使用列出的指定数据源或 ADO.NET 数据源。访问数据的最好方法是使用特定到数据存储位置的 TableInfo 类直接打开。ADO.NET 数据源 (TableInfoAdoNet) 用于访问数据，这些数据不被内部支持，但是具有 ADO.NET 提供方。

访问数据的方法

MapXtreme 2005 提供了多种方法将数据加载到 Catalog 中：

1. 直接访问数据源
2. 通过 ADO.NET 数据提供方访问
3. 来自第三方 web 服务的 XML/GML

访问数据的最好方法是使用特定到数据存储位置的 TableInfo 类直接打开（请参阅[第 81 页中的支持的数据源](#)）。

使用第二种方法 (TableInfoAdoNet) 访问不被内部支持但有 ADO.NET 提供方的数据。

第三种方法使开发人员可以将数据集成到可能与返回 XML 或 GML 的 HTTP 服务交互的 Catalog。

直接访问数据源

MapXtreme 2005 支持本地访问存储在 RDBMS 服务器（如 SQL 服务器和 Oracle）中的数据。可以使用 TableInfoServer 类来定义连接串和 SQL 语句在远程表上执行。MapXtreme 内部使用 ODBC 或 OCI 访问远程数据库。使用此类型的数据提供方时，Catalog 可以确定基于远程服务器表的空间特征，并将其正确显示。请参阅开发人员参考帮助系统中的远程表和地图目录。

要打开访问 RDBMS 数据的表，使用以下程序代码：

```
public static void TableInfoServer(MIConnection connection) {  
    // Note: Do not specify any columns.  
    // These are determined dynamically from the query.  
    TableInfoServer ti = new TableInfoServer("States");  
    ti.ConnectionString = "SRVR=ontario;UID=mapx;PWD=xxxx";  
    ti.Query = "Select * From States";  
    ti.Toolkit = ServerToolkit.Oci;  
    ti.CacheSettings.CacheType = CacheOption.Off; // On is the default  
    Table tbl = connection.Catalog.OpenTable(ti);  
}
```

这将打开到服务器的连接，查询表的元数据，创建适当的表定义，具有远程服务器上定义的任何空间特征。这是使用远程数据的最好执行方法。在内部，MapXtreme 仅可以访问执行当前操作的数据。地图绘制过程中，MapXtreme 会构造一个查询，此查询仅返回几何列，而不返回数据列。这会最小化网络通讯。如果高速缓存开启，那么只有第一次访问会有这个问题，因为所有后续请求将来自该缓存。

通过 ADO.NET 数据提供方访问

第二种远程数据访问方法是使用 ADO.NET 数据提供方。这需要数据检索的 ADO.NET 类定义。只有不可制图的表可以提供为 AdoNet 表。不可制图的表是那些不包含关于数据几何体信息的表。但是，从 ADO.NET 提供方检索的表可以通过将 SpatialSchema 应用到表定义而成为可制图的表。用此种方法，无论何时用户请求数据，MapXtreme DataAccess 引擎均将调用 ADO.NET 数据提供方。这会是一种较慢的访问数据的方法。但是，当与高速缓存共同使用时就会很好地执行。

下一节是关于如何定义数据提供方访问 Excel 电子表的示例。

Excel 电子表

MapXtreme 没有直接访问 Excel 数据的常用数据源。需要定义 Excel 的 ADO.NET 提供方，以将数据加载到 Catalog。下面是在 MapXtreme 中打开 Excel 表的示例：

```
public static void TableInfoAdoNet2(MIConnection connection) {
    TableInfoAdoNet ti = new TableInfoAdoNet("EuropeanCities");
    OleDbConnection _conn = new
    OleDbConnection(
        @"Provider=Microsoft.Jet.OLEDB.4.0;Data Source=C:\Data.xls");
    string selectQuery =
        "SELECT City, Country, Continent, Pop_1994 FROM EuropeCities";
    OleDbCommand _OleDbCommand = new OleDbCommand(selectQuery);
    _OleDbCommand.Connection = _conn;

    selectQuery = selectQuery +
        " where City = @City AND Country = @Country";
    OleDbCommand _OleDbKeyCommand = new OleDbCommand(selectQuery);
    _OleDbKeyCommand.Parameters.Add("@City", OleDbType.Char);
    _OleDbKeyCommand.Parameters.Add("@Country", OleDbType.Char);
    _OleDbKeyCommand.Connection = _conn;

    // The MapInfo Table will Open/Close the connection as necessary.
    // If this is expensive,
    // the application could open the connection before opening the
    // table and closing the
    // connection after the table is closed.
    ti.SequentialCommand = _OleDbCommand;
    ti.FetchByKeyCommand = _OleDbKeyCommand;

    // Tell the table which column(s) constitute a key -
    // for this table it is a compound
    // key consisting of values from the City and County columns.
    StringCollection sc = new StringCollection();
    sc.Add("City");
    sc.Add("Country");
```

```
ti.KeyColumns = sc;
ti.KeyType = KeyType.Explicit;
// Ask the Catalog to open the table.
Table tbl = connection.Catalog.OpenTable(ti);

// Now the MICommand object may be used to select data from
// the table (by the name EuropeanCities
// since that is the alias we assigned to it). The data in
// this table may be joined with any other
// table and it may be used as source data in a call to
// AddColumns to populate temporary
// columns with data from this table.
}
```

请注意，该代码创建了一个 OleDb ADO.NET 提供方和命令对象，并将它们通过 TableInfoADONet 类传递到 Catalog。一旦打开，所有的数据请求将从 Catalog 汇集到这些提供方。这种情况下性能就受到限制，因为数据必须从一个系统转换到其它系统。

注：有两种基于 ADO.NET 的表：上面示例中定义的基于命令的形式，以及基于 DataTable 的形式（内存中的表）。详细信息请参考 MapXtreme 2005 开发人员参考帮助系统中的 TableInfoAdoNet 类。

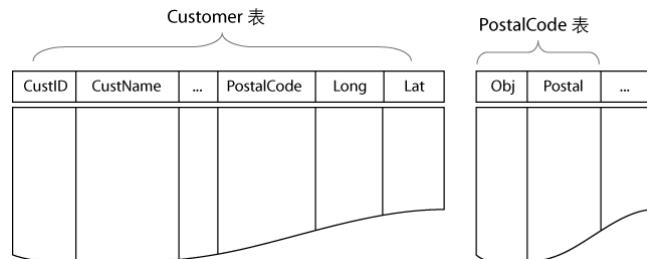
第三方 Web 服务中的数据

MapXtreme 2005 可以将 Web 服务 XML 或 GML 输出集成到 Catalog 中以便用于 MapXtreme 桌面或 web 应用程序中。数据可以通过 MapXtreme 2005 API 进入 Catalog 并转换到 MultiPolygon、LineString、Point 或其它 Geometry。MapXtreme 然后将 Geometry 转换为 FeatureCollection，并依次将其保存到 memTable 或本地 TAB 格式。

如果想要使数据在 MapXtreme 中可用但不必用于地图显示，则可以使用此方法。

相关表

表一旦在 Catalog 中打开，必须将数据转换为要求的格式。该键用于确保该数据从数据源用适当且可行的方法来检索。本节讲述了多种方法从不同的数据源与数据关联。本节中我们将使用下表来说明该方法：



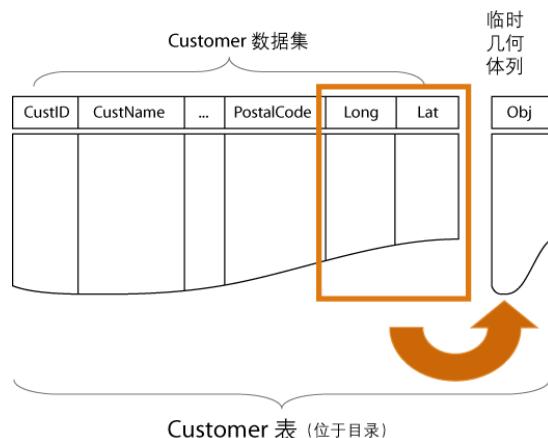
Customer 表是一个简单的表，包含代表数据点纬度值和经度值的数据列和数字列。第二个表是可制图的 EuPostal 表，包含代表邮政编码界线的数据列和几何体列。此几何体列可以是任何几何体类型。

该 Customer 表包含位置信息，但不包含几何体列。此表不会添加到地图，除非它关联到包含几何体列的表。

可能还需要在 Customer 表中分析数据。例如，可以通过其位置信息分组或求和，以便可以作为带底纹地图显示在邮政编码界线之上。以下章节描述获取非空间数据及添加空间组件的多种方法。请参阅 MapXtreme 2005 开发人员参考帮助系统获取这些方法的代码示例。

XY 空间构架

第一个示例演示如何向 Customer 表添加空间构架，以便可以在地图上显示这些点。空间构架是一种应用于表来增强其空间功能的服务。在不可制图表的情况下，构架必须有一个属性列，代表 X 值和 Y 值（例如，经度 / 纬度坐标系）或者是可以用于参考可制图表中的记录。当表打开时，不可制图表可以使用空间构架服务来同步 GeometryColumn。



Customer 数据集分配给基于所使用的 TableInfo 结构的“数据提供方”。然后空间构架分配给 TableInfo 类，然后该表打开。在内部，数据集是简单值的表，但是 XYSpatialSchema 创建了一个临时列代表基于纬度列和经度列的几何体。当使用 Catalog 访问此表时，将会看到一个 Geometry 列，并可以在空间查询中使用此列或用于制图。空间选择集在第 185 页第 10 章中的搜索图元中讨论。

XY 空间构架具有以下特征：

- Geometry 列可编辑。
- 自动编辑 Geometry 会更改经度和纬度值。
- XYSpatialSchema 可以应用于任何数据源。

- 可以在表中定义每个点的样式。
- 可以将表信息存储为 TAB 文件，并象任何其它表一样打开。

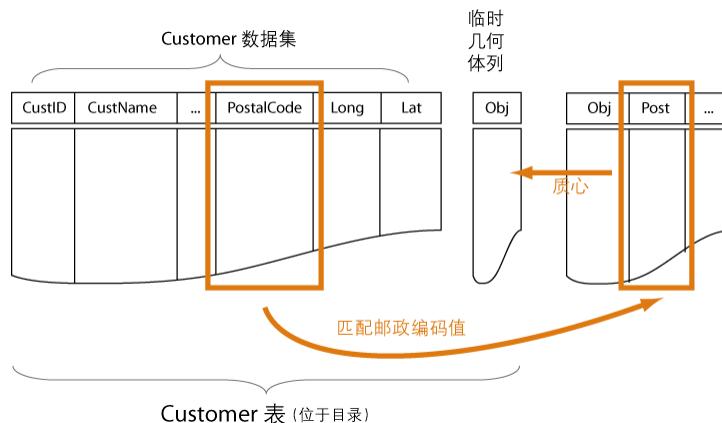
此空间构架可以用于传统的没有 MapCatalog 的服务器 XY 数据。（使用 MapCatalog 可以提供 RDBMS 上的更好性能，因为有更多的工作在服务器上完成请参阅第 206 页第 11 章中的 *MapInfo MapCatalog*。）

详细信息和代码示例，请参阅开发人员参考帮助系统中的 `MapInfo.Data.SpatialSchemaXY` 类。

PointRef 空间构架

PointRef 空间构架实际上是两个表之间的连接，一个包含数据，另一个包含连接列和对象列。结果是包含几何体列的表。此几何体列具有以下特征：

- 该数据表可以匹配几何体表中的多个记录。发生此种情况时，相似行会合集到 MultiPoint 几何体中。
- 该几何体是其它表中的几何体的质心。



连接在 EuPostal 表的 Post 列上执行。几何体的质心为 Customer 表上的临时列。Customer 表现在可以添加到地图上，以便您可以执行空间查询、创建主题图并执行其它地图分析。

PointRef 空间构架具有以下特征：

- 临时 Geometry 列为只读。
- 任何对 EuPostal 表中的值所作的编辑自动更改 Customer 表中的 Geometry 值。
- PointRef 空间构架可以应用于任何数据源。
- 可以在表中定义每个点的样式。
- 可以将表信息存储为 TAB 文件，并象任何其它表一样打开。

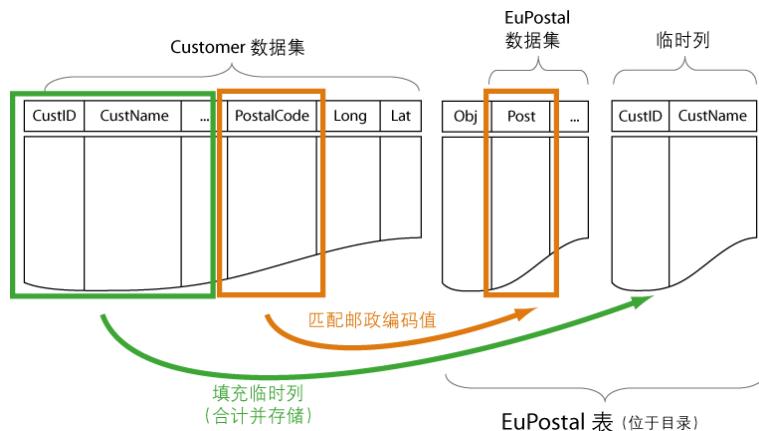
详细信息和代码示例，请参阅开发人员参考帮助系统中的 MapInfo.Data.SpatialSchemaPointRef 类。

添加列

关联表的另一种方法是表上的 AddColumns 方法。此方法在基于 COM 的 MapX 产品中使用，在 MapXtreme 2005 中作为一种向那些转向 .NET 框架的客户提供的便利。

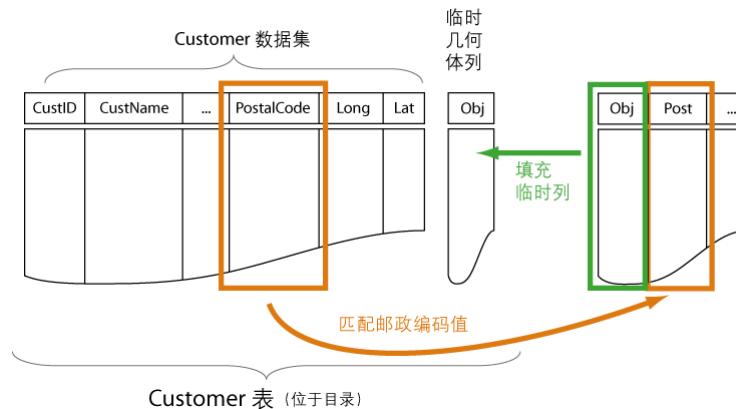
注：以下表类型不支持 Table.AddColumn：Server、View、Seamless、AdoNet、ResultSet 或 Drilldown。

在此种情况下，一个表中的列将添加到另一个表中。



在此种情况下，Customer 表中的列添加到 EuPostal 表中。现在用户信息可用于在 EuPostal 表中的分析或添加主题图底纹。

或者可以使用相反的策略：即，将 EuPostal 列添加到 Customer 表：

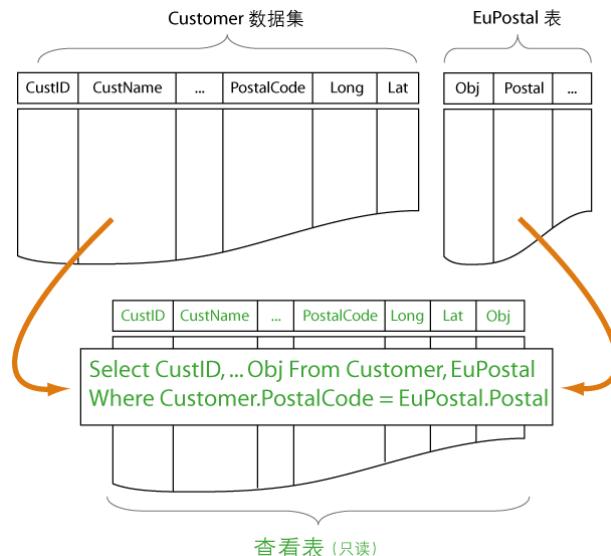


在此种情况下，EuPostal 表中的列添加为 Customer 表中的临时列。现在 Customer 表包含 Geometry 列，可以制图、在空间查询中使用或者以不同的主题加上底纹。使用此方法复制源数据并存储为临时列。将这些添加的列存储到磁盘上的临时文件中。使用 AddColumns 方法可以使桌面应用程序的性能改进，连接可以一次确定，而不是每次后续访问都要确定（如在视图中）。

详细信息和代码示例，请参阅开发人员参考帮助系统中的 MapInfo.Data.Table.AddColumns 类。

视图

视图是关联各表中信息的另一种方法，基于命名选择语句。Catalog 可用于创建基于任何表定义的视图。



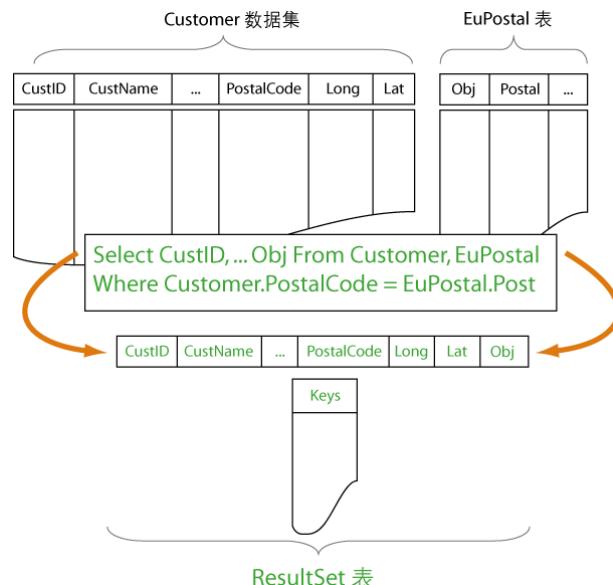
以上示例显示在 Catalog 中基于 SQL 语句创建的结果视图。此新表具有以下特征：

- 不复制数据
- 访问视图始终访问其基本表（Customer 和 EuPostal）
- 视图是具有名称的 SQL Select 语句（别名）
- 视图中的成员条件是活动的
- 异常：合计缓存数据的视图。

详细信息和代码示例，请参阅开发人员参考帮助系统中的 MapInfo.Data.TableInfoView 类。

结果集

结果集表与使用选择语句和具有相关名称（别名）定义的视图表相似。结果集具有记录的固定成员关系，该成员关系基于在创建结果集时对 where 子句（如果有）的评估。任何对结果集中数据的访问都反应源表中的数据。而对源数据的更改将不会引起结果集基于原始 where 子句添加或删除记录。结果集内部管理一组键。



通常结果集是轻量级和临时的。结果集的一些特征是：

- 不复制数据
- 访问结果集始终访问其基本表（Customer 和 EuPostal）
- 结果集是键的排序列表，列定义的集合和名称
- ResultSet 中的成员关系是固定的
- 异常：合计缓存数据的视图。

详细信息请参阅开发人员参考帮助系统中的 `MapInfo.Data.TableInfoResultSet` 类。

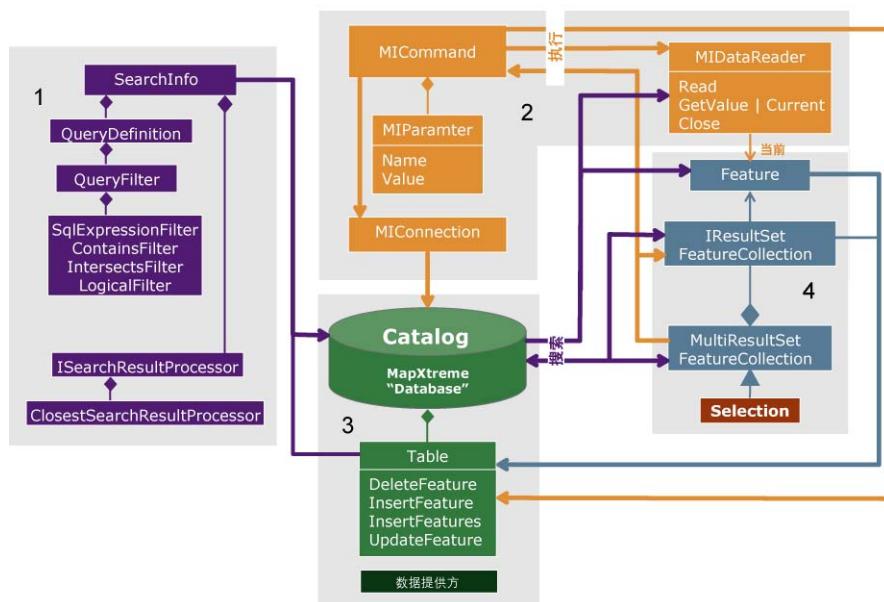
分析数据

数据构架设计之后信息在 Catalog 的表中可用，这时将需要分析数据。Catalog 具有一个 SQL 处理器，不仅可以进行以上连接，而且可以分析和合计数据。

下图显示基于 OGC 对象的查询接口和基于 ADO.NET SQL 的接口之间的关系。

第 4 组显示 OGC 查询接口。使用这些对象构造查询。该接口可以用于创建查询以过滤列和行，并添加空间和非空间条件。查询通过 Search 方法进行交互（而不是查询对象），然后返回数据读取器和结果集。如果更愿意使用面向对象编程技术而用 SQL 语法比较少，则使用这些对象。

ADO.NET 接口（如第 2 组中显示）使用定义的 ADO.NET 模型，允许通过 MapInfo SQL 语句访问。ADO.NET 接口使用 SQL 语法与 Catalog 交互。在此实例中，需要生成 SQL 状态，并将其分配到 MICommand 对象。这些对象使用 Execute 命令返回数据读取器或结果集。



MapXtreme 2005 数据模型

基于查询的 OGC 和基于命令的 ADO.NET 方法使用 Catalog（第 3 组）来组织数据源作为对象或 SQL 查询的响应。基于对象的查询 API 将生成 SQL，并将此传递给 Catalog 进行处理。在一些实例中，您可以手动生成更有效的 SQL，但是对象可以很好地进行定义并且接口限制交互的方式，因此最好使用 SQL。如果您愿意使用 SQL 语言，那么使用 ADO.NET 方法会更习惯。但是如果我没有使用 SQL 的经验，那么基于查询的 OGC 对象也可以很好地工作。

MapInfo SQL 语法在 MapXtreme 附带的 SQL 参考中定义，安装 MapXtreme 2005 后集成到 VS.NET 中。该语言基于 SQL3，具有为空间分析而定义的特殊 MapInfo 运算符。这些运算符以 MI_prefix 开头。该参考位于第 357 页附录 B 中的 *MapInfo SQL 参考*。安装 MapXtreme 2005 之后，Visual Studio .NET 中还有可用的 CHM 格式。

访问数据返回数据读取器或结果集的方法。数据读取器允许以顺序方式访问，不存储数据的副本。除非在缓存数据源的情况下，否则均从数据源检索数据。结果集是键的集合。这些键可以用于访问回原始表，并不创建该数据的副本。

MemTable 也可以用于将数据从各种源中存储到表中。此种类型的表以内存数组与临时磁盘存储方式存储数据。添加数据时，MemTable 制作一个数据的副本，且没有键或指针指向原始表。这些对于地图和容器的临时图层来讲是非常有用的，可以返回进程值如地理编码或路由结果。MemTable 访问和地图渲染性能与本地表相同。

当需要访问定义的一组行和需要从源获取数据时，结果集会是一个非常有用的工具。如果源数据在会话过程中可以更改，如果数据源支持同步访问，则此方法可用于查看结果。因为 MemTable 是数据的副本，它们是数据行的静态集，将不反应用于原始数据源的更改。

改进数据访问性能

性能始终是任何应用程序访问数据的一个重要方面。下面列出了用于改进性能的方法：

- 只请求需要的数据（尤其是从 RDBMS）。这就限制了通过连接发送的数据数量。
- 如果需要排序的列表，仅将表排序。此过程需要花费时间来读取整个表才能构建顺序。如果列没有索引，排序还会更慢。
- 如果需要随机访问表，则仅做滚动。此外还生成索引来加速访问和记住顺序。数据读取器直接访问数据，而不需要读取附加数据。
- 使用一致坐标系进行 Join 和 Search 操作。从而不需要每一次访问时都转换几何体。
- 使用带索引的列来进行 Join / Filter / Sort / Aggregate 等操作。
- 实际检查几何体相交之前，使用 CentroidWithin、ContainCentroid 和 EnvelopesIntersect。这些测试非常快速，大多数情况下可以从列表中不费力地删除大量几何体。
- 当执行多查询和/或编辑时，使用 BeginAccess/EndAccess（特别是对基于文件的表）。
- 尝试避免在 where 子句中调用 Area 和 Buffer，因为每次创建一个新光标时必须执行此操作。
- 定义视图或结果集时，尝试避免在 Select 列表中调用 Area 和 Buffer，原因同上。
- 对于管理键的中间结果或操作使用结果集。这些是很轻量的操作，可以提供快速直接访问原始数据。

6

运用 MapXtreme 2005 Web 控件

本章讨论与 ASP.NET 应用程序配合使用的 MapXtreme 2005 Web 控件和工具。

本章内容：

- ◆ *WebControls* 命名空间概述 94
- ◆ *Web* 控件的体系结构 94
- ◆ 使用 *Web* 控件 97
- ◆ 管理控件 102
- ◆ 从 2004 *Web* 控件迁移到 2005 *Web* 控件 106
- ◆ 更改和分发 *Web* 控件 107

WebControls 命名空间概述

MapInfo.WebControls 命名空间包括用户界面工具和 Web 控件，用于设计和使用由 MapXtreme 2005 构建的 Web 窗体应用程序。用户可通过使用 Visual Studio 方便地在 Web 窗体上拖放 Web 控件。这些控件和工具提供了包含各种功能的服务器和客户端行为，包括：地图显示、导航、图元选择以及图层管理。

MapXtreme 2005 Web 控件允许用户利用快速应用程序开发 (RAD) 的内建功能。这些 Web 控件提供了常规用户界面组件，这些组件在包含 MapXtreme 的绝大多数应用程序中均可使用。这些 Web 控件还提供了学习参考，说明如何使用 MapXtreme 功能构建可视的 Web 应用程序。

Web 控件的体系结构

新的 MapXtreme 2005 Web 控件体系结构不同于现有的 MapXtreme 2004 体系结构。新的控件遵循 ASP.NET 模型来创建 Web 应用程序。不同之处如下所示：

MapXtreme 2004 Web 控件

- 每次使用工具时都会使用名为 Page_load 和 Page_unload 的回送页面。
- MapControl 会在设计阶段和运行时加载工作空间。
- 状态管理在 Page_load 和 Page_Unload 中完成。
- 错误处理由 ASP.NET 执行。由于每次都会重新渲染页面，所以 ASP.NET 将会捕获异常并重绘错误页面。
- 用于 selection 和 distance 工具控件的事件句柄在 Page 类中处理。

MapXtreme 2005 Web 控件

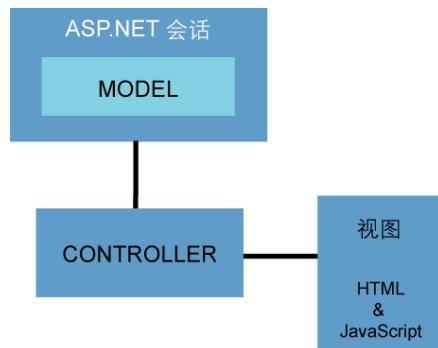
- 第一次渲染页面时会调用 Page_load 和 Page_unload。每次使用工具时都不会出现回送页面。
- 使用预装的工作空间。在设计阶段，可以将 MapControl MapAlias 属性设置为在预装的工作空间中定义的地图的别名。这样就会在运行时将相应的地图加载到 MapControl。

- StateManager 在应用程序中作为类实现，并且会将实例放入 ASP.NET 会话中。SaveState 和 RestoreState 方法从该对象调用。每次使用工具时都会调用 SaveState 和 RestoreState 方法，并且若 ASP.NET 会话包括 StateManager 类实例，则仅更新 LayerControl。若状态管理使用 Manual 模式，则 StateManager 类实例必须位于该会话中。（在 web.config 文件中将 MapInfo.Engine.Session.State 设置为 Manual。）
- 您可以控制执行错误处理的方式。错误处理在 global_asax.cs/.vb 文件的 application_error 事件句柄中执行。
- 由于在使用地图工具时未重新加载 Page，则无法使用事件句柄（该句柄在 MapXtreme 2004 中实现）。对于 MapXtreme 2005 Web 控件，必须编写客户端 Javascript 命令以将请求发送到服务器并获取和处理该响应。还必须编写自己的服务器端命令类以执行该工具的服务器端处理。在设计阶段，必须将工具的 ClientCommand 属性设置为 Javascript 中的函数名并将 Command 属性设置为服务器端命令类名。

在初始化期间渲染页面时会渲染所有控件。完成初始化后，使用工具时仅渲染地图图像。例如，绘制缩放工具矩形，然后在收集数据之后使用 **img.src = url** 组成 URL 并仅更新地图图像。当该 URL 用在源中时，会将请求发送到服务器。在服务器端会分析请求并执行操作。图像将被导出，并将该图像作为返回客户端的数据流写入到响应。通过此方法，可以可视化编写更加吸引人的 Web 应用程序。

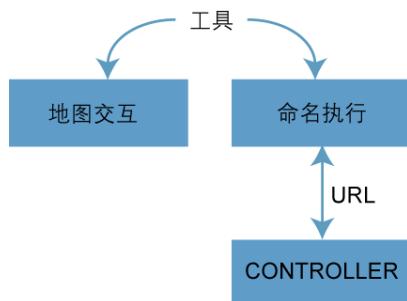
一般体系结构包含以下三个主要功能：视图、控制和模型。

- 视图** - 视图是用 C# 或 VB 代码在 Web 控件中完成的 HTML 和 Javascript 渲染。
- 控制** - 视图使用命令和数据与控制进行通信以执行各种操作。然后，控制发送响应。
- 模型** - 模型是接口特定于产品的实现。



关于 MapTools 体系结构

MapXtreme 工具会渲染 JavaScript 和 IMG 标记以指向活动和不活动状态的图像。当使用 MapController 更改地图时，大多数工作都通过 JavaScript 执行。该工具设计为两个部分：地图交互和命令执行。



客户端地图交互

使用 JavaScript 继承模型创建交互类。这些类可由任意用户接口使用以在任意 HTML 元素上执行交互。用于交互的构造函数是：Interaction(elementID, onComplete)，该构造函数具有地图 IMG 标记的 ID 以及回调函数 onComplete，交互完成时会调用此函数。例如，交互类可以在给定的元素上绘制橡皮圈几何体并收集数据。结束时会调用 onComplete 函数。

客户端命令执行

工具具有负责执行特定任务的命令对象。还有 Command 基类以及派生的和实现的特定命令（例如，MapCommand 或 PanCommand）。交互完成后，工具会在命令上调用 Execute 方法执行该任务。还可以创建可由任意用户接口执行的命令，不必通过 MapXtreme 工具。

Command 类的构造函数是：Command(name, interaction)，其中命令的 name 是请求 URL 中所包含的服务器端命令类名，并且会将该类名发送到服务器以便服务器知道哪个服务器命令类必须处理此客户端请求。interaction 是交互对象，用来访问要包括在 URL 中的数据。此交互对象可以是空。

例如，MapCommand 的任务是组成 URL 并使用适当命令将其发送到 MapController，然后更新地图。MapCommand 会假设交互对象中的元素是地图图像，然后将图像的源更新为新的 URL。

XmlHttpCommand 的任务是使用 XmlHttp 机制将请求发送到句柄并获取返回的响应，而不执行回送。

MapController 用于执行操作，然后将图像传输回视图。MapControls 的模型用于调用方法并执行操作。

服务器端命令的体系结构

MapControlModel 具有可更改或扩展的命令体系结构。MapControlModel 是命令的集合。调用 InvokeCommand 时，将会获取带有给定名称的命令并调用其 Execute 方法。可更新该集合、可更改现有命令或可增加新的命令。Execute 会先调用 RestoreState，然后调用 Process，最后调用 SaveState。

使用 Web 控件

MapXtreme 2005 包括了帮助开发人员使用 MapXtreme API 的 Web 控件。可以使用这些 Web 控件来构建新的应用程序或增强现有的应用程序。使用 MapXtreme 2005 Web 控件的好处是：

- 这些控件提供了内置功能帮助您轻松快速地将基于位置的功能增加到商业应用程序中。
- 只需更改少量编码即可增加功能。若为工具编写您自己的命令类，则必须将命令增加到 MapControlModel 后才能使用该命令。
- 提供了允许修改和扩展 Web 控件以满足应用程序业务需要的源代码。

什么是 Web 控件？

安装 MapXtreme 后，会将 Web 控件和工具部署和集成到现有的系统配置中。可在 MapXtreme 2005 中使用以下 Web 控件：

- **MapControl** - MapControl 允许用户显示 Map 对象的实例。通过使用 MapAlias 属性，可从 MapFactory 获得该地图。通过导出地图图像来绘制地图，并在 IMG HTML 标记中引用该图像。如果 MapAlias 属性未指定或者无效，则从 MapFactory 中选择第一个地图。强烈建议您在设计阶段显式设置 MapControl MapAlias 属性。
- **LayerControl** - LayerControl 允许用户以树视图结构显示地图图元图层和标注图层。该控件可以打开或关闭特定图层的可视性并显示只读的当前缩放值。可视性更改仅在使用 LayerControl 的应用程序处于活动状态时永久有效。要更改图层的缩放值，请使用其中一个缩放工具。
- **LegendControl Sample Web Application** - LegendControl 包括在示例 Web 应用程序中。该示例说明了如何基于 Web 控件体系结构创建并使用自定义的 LegendControl。该示例允许用户创建主题图并（通过 Javascript）将请求发送到服务器来显示图例，而不必刷新页面。该示例应用程序中所包括的工具用来在 LegendControl 中执行标准操作，如平移和缩放、创建或删除 IndividualValueTheme 以及显示或隐藏主题图图例。

可在 MapXtreme 2005 中使用以下工具：

- **CenterTool** - CenterTool 允许用户通过单击地图来重定地图的中心。
- **DistanceTool** - DistanceTool 允许用户通过单击地图来获取两点或多点之间的距离。
- **NavigationTools** - NavigationTools 允许用户按固定方向平移地图：North、South、East、West、NorthEast、NorthWest、SouthEast 和 SouthWest。
- **PanTool** - PanTool 允许用户通过按任意方向拖动在窗口中重新定位地图。
- **PointSelectionTool** - PointSelectionTool 允许用户在单击地图时可以选择图元（离该点最近的图元）。
- **PolygonSelectionTool** - PolygonSelectionTool 允许用户选择其中心位于多边形内的所有图元。使用鼠标单击多边形节点的表示，在地图上绘制多边形。双击会封闭 / 结束该多边形。
- **RadiusSelectionTool** - RadiusSelectionTool 允许用户选择其中心位于半径内的所有图元。使用鼠标单击圆中心和边界的表示，在地图上绘制半径。
- **RectangleSelectionTool** - RectangleSelectionTool 允许用户选择其中心位于矩形内的所有图元。使用鼠标单击对角的表示，在地图上绘制矩形。
- **ZoomBarTool** - ZoomBarTool 允许用户将地图缩放至一系列预设置的级别。
- **ZoomInTool** - ZoomInTool 允许用户绘制表示在地图上视图要放大到的矩形。
- **ZoomOutTool** - ZoomOutTool 允许用户绘制表示在地图上视图要缩小到的矩形。
- **InfoTool Sample Web Application** - InfoTool 包括在示例 Web 应用程序中。该示例说明了如何基于 Web 控件体系结构创建并使用自定义的 InfoTool。该示例允许用户通过在屏幕上选择图元并通过使用 javascript 将请求发送到服务器（不必刷新整个页面）来检索和显示图元的信息。

使用新的 Web 控件

要使用 MapXtreme 2005 Web 控件，在创建应用程序时必须使用以下步骤（随 MapXtreme 一起安装的所有示例中都包含这些步骤）：

1. 安装完成后，使用 MapXtreme 模板创建模板应用程序或创建新的 Web 应用程序。
2. 在 Visual Studio 中，所有控件和工具都可从 MapXtreme Web 6.5 工具箱拖放。
3. 请确保将 web.config 文件设置为预加载工作空间。这是加载工作空间的唯一方法。若应用程序中需要多个地图，则这些地图可位于一个或多个工作空间中，每个地图都具有唯一的地图别名。可通过更改地图别名来改变地图。还可以在开发环节在该点更改对象池和状态设置。
若想在 web.config 文件中预装多个工作空间，请在每个工作空间的完整路径之间使用分号。
4. 请确保设置 MapControl 的 MapAlias 属性。若未设置该属性，则将仅从该会话中选取第一个地图。

5. 对于工具、LayerControl 和其它自定义的控件，必须将 MapControlID 设置为指向 MapControl。若未执行此操作，则客户端将会出错。
6. 若仍未实现 StateManager 类，则会在应用程序中实现该类。出于效率目的，请确保只保存和恢复更改的信息。创建 StateManager 的实例并将其放入 ASP.NET 会话中。若正在使用其中一个 MapXtreme 模板创建应用程序，则该模板的所有概要代码都会存在。

创建自定义命令

通过使用以下步骤，可以更改现有命令的行为或增加新的自定义命令：

1. 从 MapBaseCommand 派生命令类并在构造函数中命名此命令类。
2. 重载 Process() 方法并在那里执行处理。
3. 在 webform 的 Page_Load 方法中，将命令增加到该模型中的命令集合。使用 MapControlModel.SetDefaultModelInSession 静态方法获取模型的实例。然后为新命令，通过由 model.Commands.Add(new yourcommand) 增加命令实例来增加命令。要更改现有命令，可以从集合中删除该命令并增加您自己的命令。

创建自定义工具

可以通过以下方法创建您自己的 webcontrol：

1. 创建从 WebTool 派生的“自定义工具”类。
2. 在构造函数中，设置命令名、交互和光标 URL。
3. 在 aspx 窗体中，象其它控件那样手动插入该自定义工具。还要注册指定程序集和该控件所位于的命名空间的标记前缀。

在设计阶段或在构造函数中，Webcontrol 的 ClientCommand 属性必须设置为现有内置客户端命令名（例如 MapCommand），或在 Javascript 文件中编写的客户端命令名。Command 属性必须设置为现有内置服务器端命令名或编写的自定义命名类名。ClientInteraction 必须设置为现有内置交互或在 Javascript 中编写的交互名。

若编写您自己的 Javascript 处理该工具的客户端行为，则必须手动在 .aspx 页面中注册此 javascript 文件。

通常，不必为自定义工具编写新类。可以使用现有地图工具或名为 "WebTool" 的工具，该工具设计用于创建自定义工具。只需：

- 编写服务器端命令类（若现有的服务器端命令类不适用）处理该工具的服务器进程。
- 编写 Javascript 客户端命令（若现有的 Javascript 客户端命令不适用）处理该工具的客户端行为。

- 如果需要，则编写客户端交互。
- 然后，仅将该工具或 WebTool 拖动到 webform、设置其 Command、ClientCommand、ClientInteraction、InActive/ActiveImageUrl、CursorImageUrl 以及自定义工具的任何其它属性。

请参阅 web 示例应用程序：Tools、InfoToolWeb、LegendControlWeb，用于创建自定义服务器端命令类和客户端命令 Javascript 的真实示例。

若要增加现有工具不具有的额外属性，则必须编写从 WebTool 派生的新的自定义工具类。

在框架中使用 Web 控件

MapXtreme 2005 Web 控件目前在框架中运行。有关详细的指示，Visual Studio .NET 文档包含了有关如何创建框架、框架集以及将页面分配给框架的信息。

在框架中使用 Web 控件时，请记住框架指向 aspx 页面并且框架集包含一个或多个框架。利用以下方案：带有 MapControl 的页面以及带有工具或取决于 MapControl 的 LayerControl 的其它页面。所有页面组成了框架集。

给定上述的方案后，将会应用以下规则：

- 必须为附属的控件手动将 MapControlID 输入到属性中。若同一页面上的 MapControl 与附属的控件具有相同的 ID，则工具和 LayerControl 将选取那个控件。
- 由于页面是按特定顺序绘制的，因此该会话不是新的会话。要确保 StateManager 已正确设置，可以实现下列内容（请注意在恢复状态之前，如果该会话不存在则会增加会话）：

VB 代码示例

```
if (Session.IsNewSession) Then
    ' Set initial state of the map
    Dim sm As StateManager = New AppStateManager()
    MapInfo.Engine.Session.Current.Reload(True)
    StateManager.PutStateManagerInSession(sm)
End If

if (StateManager.GetStateManagerFromSession() = Nothing) Then
    Dim sm As StateManager = New AppStateManager()
    StateManager.PutStateManagerInSession(sm)
End If
' Now Restore State
StateManager.GetStateManagerFromSession().RestoreState()
```

在表单元格中使用 MapControl

由于 HTML 行为，一旦将元素拖离另一元素，该元素就会调整为基本大小（主要是 0）。若将 MapControl 放入表单元格，则元素在 HTML 中没有绝对宽度和高度时可能会有问题。将元素拖离单元格时，该单元格会折叠起来，因此单元格大小变为 0。

要解决该问题，请通过设置 SPAN 标记的大小来显式设置 MapControl 的大小。下面的示例说明如何在 HTML 中设置大小：

```
<table style="Z-INDEX: 101; LEFT: 32px; POSITION: absolute; TOP: 64px"  
borderColor="#ff00ff" border="1">  
  <TR bordercolor="#ff3366">  
    <TD bordercolor="#0066ff">  
      <ccl:mapcontrol id="Mapcontrol2" runat="server" Width="300px"  
      Height="300px"></ccl:mapcontrol>  
    </TD>  
  </TR>  
  <TR>  
    <TD>  
      <ccl:pantool id="Pantool2" runat="server"  
      MapControlID="MapControl2"></ccl:pantool>  
      <ccl:zoomintool id="ZoomInTool1" runat="server"  
      MapControlID="Mapcontrol2"></ccl:zoomintool>  
      <ccl:zoomouttool id="ZoomOutTool1" runat="server"  
      MapControlID="Mapcontrol2"></ccl:zoomouttool>  
    </TD>  
  </TR>  
</table>
```

Web 控件本地化

了解 Web 控件如何处理本地化资源，这点非常重要。本节描述了使用 Web 控件本地化的基内容，并且描述了本地化的某些方案。使用 MapXtreme Web 控件时，遵循以下常规步骤非常重要：

- 将所有字符串移至 resx 资源文件。这是将要翻译的文件。请确保源代码中没有硬编码。
- 使用该 resx 文件为 Web 控件生成卫星程序集。
- 根据系统的 locale，将程序集安装到文件系统上的适当位置，此时将会使用该卫星程序集的资源。

在 MapXtreme Web 控件中，已设计以下内容：

- strings.resx 资源文件包含字符串资源。该文件是嵌入式资源，且位于程序集中。在找不到用于特定 locale 的资源的情况下，将会使用该文件中的资源（英语）。

- 对于 MapXtreme 的本地化版本，卫星程序集 MapInfo.WebControls.resources.dll 安装在 GAC 中。
- 源文件 Resources.cs 包含从资源获取字符串的代码的实现。GetString 方法用来从资源获取字符串。

可以考虑使用以下方案：

- 若只为一个语言安装应用程序，则可以直接使用已翻译的 resx 文件，而不是使用卫星程序集。使用卫星程序集是个不错的方法，它简化了增加和删除语言的过程，而不必重新构建代码。
- 可以使用 resgen.exe 和 al.exe（程序集链接器）来创建卫星程序集。使用 resgen.exe 将 resx 文件转换成资源文件。al.exe 用来创建卫星程序集。
- 可以将卫星程序集放入 GAC 或将程序集复制到可执行的位置。要复制程序集，请使用与语言名相同的名称创建文件夹并将卫星程序集复制到新文件夹中。例如，对于日文卫星程序集，创建 ja-JP 文件夹。

管理控件

使用 Web 控件的重要部分就是管理。本节说明了如何执行错误管理、状态管理以及事件处理。有关使用 Web 控件创建应用程序的更多信息，请参阅第 60 页中的规划 Web 应用程序。

错误管理

在 MapXtreme 2004 Web 控件中，所有操作都会回送页面。因此，服务器端异常由 ASP.NET 捕获。若异常未正确处理，则会显示带有错误描述的错误页面。在 MapXtreme 2005 Web 控件中，始终更新地图图像。因此，若服务器端的异常没有进行错误处理，将会使地图图像显示为错误图标，并且不会出现异常信号。

可通过多个方法处理 Web 控件中的错误处理，且错误处理完全特定于应用程序。因此，本节仅说明处理错误的其中一种方法。由于客户端所期望的响应是图像，因此可以使用详细消息捕获说明并将响应与图像以及错误消息一同发送回去。因此，MapControl 现在将会包含错误消息。

下面的示例使用 Global.asax 文件并为 application_error 编写新的事件句柄。请注意，位图大小是硬编码的，应与 MapControl 的大小匹配以正确查看消息。注，可以增加代码以在消息中放入更多信息。

VB 代码示例

```
Sub Application_Error(ByVal Sender As Object, ByVal e As EventArgs)
    Dim ex As Exception = Server.GetLastError()
    If Not( ex Is Nothing) And ex.Message.Length > 0 Then
        Dim b As Bitmap = New Bitmap(300, 300)
        Dim g As Graphics = Graphics.FromImage(b)
        Dim msg As String = ex.Message
        If Not ex.InnerException Is Nothing Then
            msg+= ex.InnerException.Message
        End If
        g.DrawString(msg, New Font("Tahoma", 10), New _
            SolidBrush(Color.Yellow), New RectangleF(0, 0, 300, 300), _
            StringFormat.GenericDefault)
        Dim contentType As String = String.Format("text/HTML")
        If Not contentType Is Nothing Then
            HttpContext.Current.Response.ContentType = contentType
        End If
        b.Save(HttpContext.Current.Response.OutputStream, _
            ImageFormat.Gif)
        HttpContext.Current.Response.End()
        End If
    End Sub
```

有关错误管理的更多信息，请参阅 MSDN 站点上关于使用 ASP.NET 处理错误的内容。

状态管理

新的 MapXtreme 2005 Web 控件的状态体系结构不同于 MapXtreme 2004 Web 控件，原因是不执行回送。因此，每次使用工具时都不会调用 Page_Load 和 Page_Unload 方法。但是，若应用程序显式执行回送，则可能会调用这些方法。

StateManager 是包含 SaveState 和 RestoreState 方法的抽象类。在处理前后工具会分别调用 RestoreState 和 SaveState。由于状态管理是特定于应用程序的，因此您的责任是在应用程序的具体类中实现这些方法。这将允许用户控制恢复和保存哪些内容以及如何恢复和保存这些内容。通过 ASP.NET 进程，会自动恢复实现 ISerializable 接口的对象。这些对象只需保存。

要实现状态管理，请在应用程序中执行下面的步骤（代码部分可根据应用程序进行更改）：

1. 在应用程序中实现 StateManager 抽象类的 RestoreState 和 SaveState 方法：

VB 代码示例

```
<Serializable> _
Public Class AppStateManager
    Inherits StateManager
    Public Sub New()
    End Sub
```

```
Public Overrides Sub RestoreState()
    ' Important Note: The objects which implement ISerializable get
    restored automatically by ASP.NET serialization.
    ' However if InProc method
    ' is used instead of StateServer, you must save AND restore
    state by yourself.
    End Sub

Public Overrides Sub SaveState()
    Dim key As String = HttpContext.Current.Session.SessionID + _
        "_MISession"
    ' PERFORMANCE WARNING: This is not efficient way of saving state,
    depending upon the application you should decide what needs
    ' saving and restoring. This is just example of saving entire
    session.
    'Save the entire session object in ASP.NET session.
    HttpContext.Current.Session(key) = MapInfo.Engine.Session.Current
End Sub
End Class
```

C# 代码示例

```
[Serializable]
public class AppStateManager: StateManager {
    public AppStateManager() {}

    public override void RestoreState()
    {
        // Important Note: The objects which implement ISerializable
        get restored automatically by ASP.NET serialization.
        // However if InProc method
        // is used instead of StateServer, you must save AND restore
        state by yourself.
    }

    public override void SaveState() {
        string key = HttpContext.Current.Session.SessionID + _
            "_MISession";
        // PERFORMANCE WARNING: This is not efficient way of saving
        state, depending upon the application you should decide what needs
        // saving and restoring. This is just example of saving entire
        session.
        //Save the entire session object in ASP.NET session.
        HttpContext.Current.Session[key] = MapInfo.Engine.Session.Current;
    }
}
```

2. 在 Statemanager 中使用其中一个静态方法创建实例，然后将其放入 ASP.NET 会话。

VB 代码示例

```
Private Sub Page_Load(ByVal sender As System.Object, ByVal e As _
    System.EventArgs) Handles MyBase.Load
    If (StateManager.GetStateManagerFromSession() = Nothing) Then
        Dim sm As StateManager = New AppStateManager()
        StateManager.PutStateManagerInSession(sm)
    End If
    'Do Initial State
```

C# 代码示例

```
private void Page_Load(object sender, System.EventArgs e) {
    if (StateManager.GetStateManagerFromSession() == null) {
        StateManager sm = new AppStateManager();
        StateManager.PutStateManagerInSession(sm);
    }
    // Do Initial State
```

3. 在 Page_Load 和 Page_Unload 方法中调用 RestoreState 和 SaveState 方法。首次创建页面且应用程序没有执行回送时会调用这些方法。

用于恢复状态的 VB 代码示例

```
Private Sub Page_Load(ByVal sender As System.Object, ByVal e As _
    System.EventArgs) Handles MyBase.Load
    .....
    StateManager.GetStateManagerFromSession().RestoreState()
    .....
End Sub
```

用于恢复状态的 C# 代码示例

```
private void Page_Load(object sender, System.EventArgs e) {
    .....
    StateManager.GetStateManagerFromSession().RestoreState();
    .....
}
```

用于保存状态的 VB 代码示例

```
Private Sub Page_Load(ByVal sender As System.Object, ByVal e As _
    System.EventArgs) Handles MyBase.Load
    .....
    StateManager.GetStateManagerFromSession().SaveState()
End Sub
```

用于保存状态的 C# 代码示例

```
private void Page_Load(object sender, System.EventArgs e) {
    .....
    StateManager.GetStateManagerFromSession().SaveState();
}
```

事件处理

在MapXtreme 2004 Web控件中，具有同时用于Distance工具和Selection工具的事件。这些事件的事件句柄在Page类中编写。事件处理的方法将用于MapXtreme 2004 Web控件，原因是使用工具后，eventhandlers中的执行代码会执行回送。在MapXtreme 2005 Web控件中，事件处理的方法不可能执行，原因是不执行回送。

将eventhandlers视为命令的后处理。要在执行命令后激发处理，则可以在应用程序中创建自定义命令。该命令将替换现有命令或覆盖现有命令。下面的代码显示重载处理方法的示例。

VB代码示例

```
Public Overrides Sub Process()
    ....Do your preprocssing here...
    base.Process();
    ....Do your postprocessing here....
End Sub
```

从2004 Web控件迁移到2005 Web控件

将MapXtreme 2004 Web控件迁移到新的MapXtreme 2005 Web控件的过程不是自动过程。每个应用程序和迁移过程都不相同，并且取决于应用程序与旧的Web控件的功能和设计的紧密联系程度。

建议分阶段进行迁移过程。在某些情况下，可能需要重新调整代码结构。迁移Web应用程序时需要考虑下列过程：加载数据、替换控件以及维护状态和事件管理。

- 加载数据（始终使用预装的工作空间）
- 替换控件
- 状态和事件管理

加载数据

在新的MapControl中，加载数据的惟一方法就是使用预装的工作空间。在该工作空间中MapControl会指向MapAlias并尝试显示地图，而不加载地图。预装的工作空间可在web.config文件中指定。

在MapControl用来显示多个地图图像的情况下，设置该数据以便所有地图位于同一位置，并更改MapControl的MapAlias以选择地图。

替换控件

可通过各种方式来替换控件，如通过从窗体中删除旧的 Web 控件并将新的 Web 控件拖放到窗体上，或通过创建新的 webforms、拖放新的控件，然后将旧窗体的功能增加到新的窗体。请确保为 MapControl 设置适当的 MapAlias 并为所有附属工具和 LayerControl 设置适当的 MapControlID。

状态和事件管理

在大多数情况下，状态管理的老方法是在 Page_Load 中恢复状态，在 Page_Unload 中保存状态。必须将该代码移至新的类，该类从 StateManager 派生到 RestoreState 和 SaveState 方法。若使用自动状态或 HttpSessionState，则不必实现 StateManager。但是，若手动处理状态，则必须实现 StateManager 并将其放入 ASP.NET 会话中。

在为工具（如缩放或导航工具）编写 eventhandlers 的情况下，必须创建自定义命令并将该处理移至在 **事件处理** 部分所显示的 Process 方法。

在 MapPage 中使用 Server.Transfer 或 Response.Redirect 的情况下，该会话不再是新的。若 StateManager 先前不存在，则必须将其放入 ASP.NET 会话中。

更改和分发 Web 控件

使用 MapXtreme 2005 Web 控件，具有更改 Web 控件源代码以及分发这些控件的选项。Web 控件的源代码安装在 <MX2005_HOME>\Samples\Web\WebControls 目录中。更改现有 Web 控件时，请考虑以下关于程序集的重要因素。

- 程序集名称是 MapInfo.WebControls.dll，安装在全局程序集缓存中。该程序集具有特定的版本号，由模板和示例应用程序使用。
- 程序集中的控件安装在 Visual Studio .NET 的工具箱中。
- 程序集具有对 MapInfo.CoreEngine.dll 和 MapInfo.CoreTypes.dll 的引用。
- 资源（如图像和脚本）安装在 <MX2005_HOME>\Samples\Web\WebControls 子目录中。
- 以下文件由 Web 控件使用：
 - 所有由工具使用的 gif 文件。（*.gif）。旧的 MapXtreme 2004 Web 控件也会共享这些文件。
 - Interaction.js、Command.js、LayerControl2.js 和 Tool.js 脚本。
 - 所有由工具使用的光标文件（*.cur）与旧的 MapXtreme 2004 Web 控件共享。

要更改和分发新的程序集，请考虑以下指南：

- 将源复制到另一目录，以保留原始内容。
- 从全局程序集缓存和Visual Studio .NET工具箱中删除原始程序集。
- 使用sn - k MapInfo.WebControls.snk创建强制命名键文件(.snk)并将该键文件复制到主项目文件夹(与项目文件位于同一级别)。
- 更改AssemblyInfo.cs或AssemblyInfo.vb文件修改稿版本号以满足需要。
- 在Visual Studio .NET中打开项目，进行更改并构建项目。新的程序集应位于bin\Release目录中，因此可以分发程序集的发行版。
- 对任何脚本或图像进行更改。
- 在全局程序集缓存和Visual Studio .NET工具箱中注册新的程序集。
- 现在可以使用更改的程序集编写应用程序了。在Visual Studio .NET中注册时，可以从工具箱拖放更改的控件。
- 在Web应用程序的安装程序中，确保新的程序集安装在全局程序集缓存中。只要Web应用程序指向此版本的程序集，都将会使用更改的控件。

请考虑以下方案：

- 可以将Web控件项目和资源权限插入Web应用程序解决方案中。在这种情况下，更改资源(脚本和图像)的URL以从项目的根目录开始。这将防止创建虚拟目录。
- 不必使用全局程序集缓存和强制命名的程序集。可以将程序集的Copy属性设置为true，并使程序集位于应用程序的bin文件夹中。

7

使用桌面工具

MapInfo.Tools 命名空间包含了所有允许在 MapXtreme 2005 中创建工具的类。

本章内容：

- ◆ *MapInfo.Tools* 命名空间概述 110
- ◆ 可用工具 111
- ◆ 自定义工具 115
- ◆ 工具事件 116
- ◆ 教程：使用 *Select* 工具编辑 *FeatureGeometry* 118

MapInfo.Tools 命名空间概述

MapInfo.Tools 命名空间包含了所有允许为 MapXtreme 2005 应用程序创建基本和可定制工具的类。可用工具可细分为五个不同的类别：View 工具、Select 工具、Add 工具、Base 工具和 Custom 工具。

常用 SelectMapTool 工具包括 SelectPointMapTool、SelectPolygonMapTool、SelectRectMapTool、SelectRadiusMapTool 和 SelectRegionMapTool。对于每个 Select 工具，都有可编辑和可选择的图层和节点属性。

常用的 AddMapTool 工具包括 AddPointMapTool、AddLineMapTool、AddPolylineMapTool、AddPolygonMapTool、AddCircleMapTool、AddEllipseMapTool、AddRectangleMapTool 和 AddTextMapTool。对于每一种 Add 工具，它们都有 Style 和 Insertion Layer 属性。

Custom 工具基类包括 Point、Line、Polyline、Polygon、Circle、Ellipse 和 Rectangle。它们都可用来创建 Custom 工具。

可以使用事件代码自定义这些常用工具的新实例，或编写从常用工具派生的您自己的工具类，但这样会覆盖特定的方法。请参阅[第 115 页中的自定义工具](#)。

可将工具指定给特定鼠标按键（鼠标左键、鼠标右键或鼠标中键）。以下字符串工具名称用于适当的鼠标按键属性：“Arrow”、“ZoomIn”、“ZoomOut”、“Center”、“Pan”、“SelectPoint”、“SelectRect”、“SelectRadius”、“SelectPolygon”、“SelectRegion”、“AddPoint”、“AddLine”、“AddPolyline”、“AddPolygon”、“AddRectangle”、“AddCircle”、“AddEllipse”、“AddText” 和 “Label”。

例如，下面的代码会将鼠标左键设置为 ZoomIn 工具：

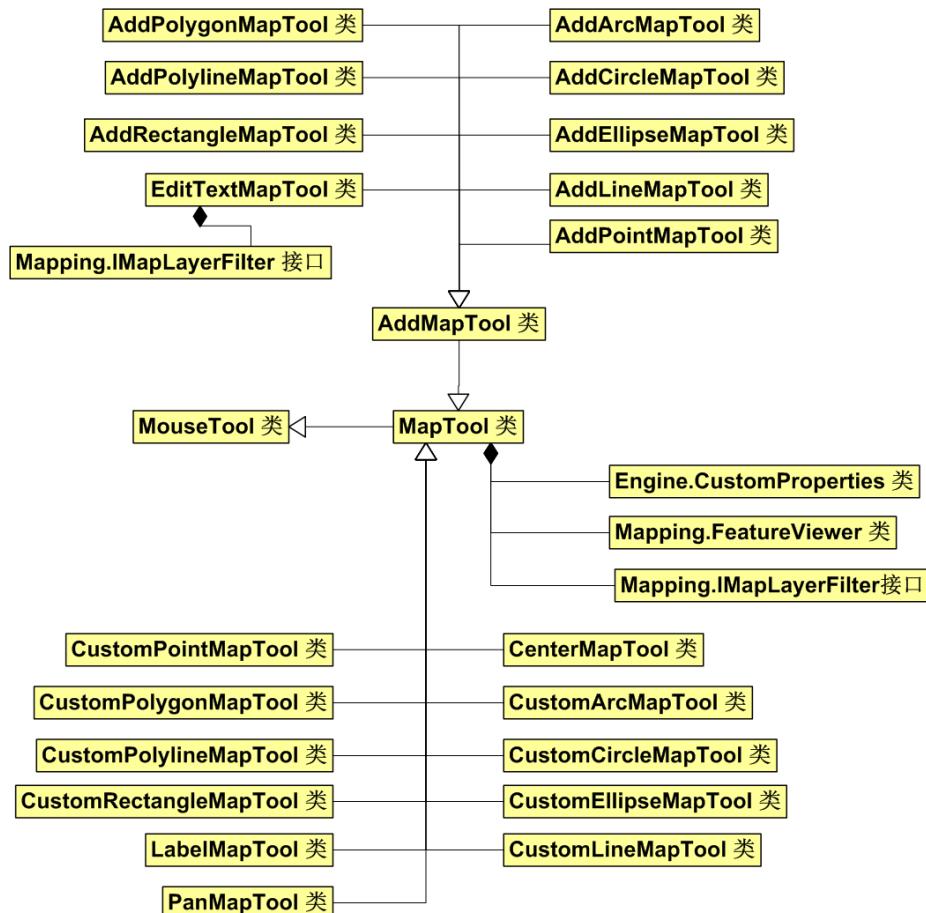
```
mapControl1.Tools.LeftButtonTool = "ZoomIn"
```

MapXtreme 2005 在其 Windows 窗体模板和 Visual Studio .NET 设计器工具箱中提供了立即可用的工具。只需简单地将工具拖放到窗体上并按需设置属性即可。

要获得通过编程方式使用桌面工具的示例，请参阅 MapXtreme 2005 安装目录
..\\MapInfo\\MapXtreme\\6.5\\Samples\\Desktop\\Features\\DesktopTools 下的示例应用程序。

可用工具

下面的图表显示了 MapTool 类的 UML 表示。



工具基本上都是从 `MapTool` 类派生的。Select 和 Add 工具分别从 `SelectMapTool` 和 `AddMapTool` 类派生。

该命名空间中所包括的 Tools 可分为四个类别：View 工具、Select 工具、Add 工具和 Custom 工具。

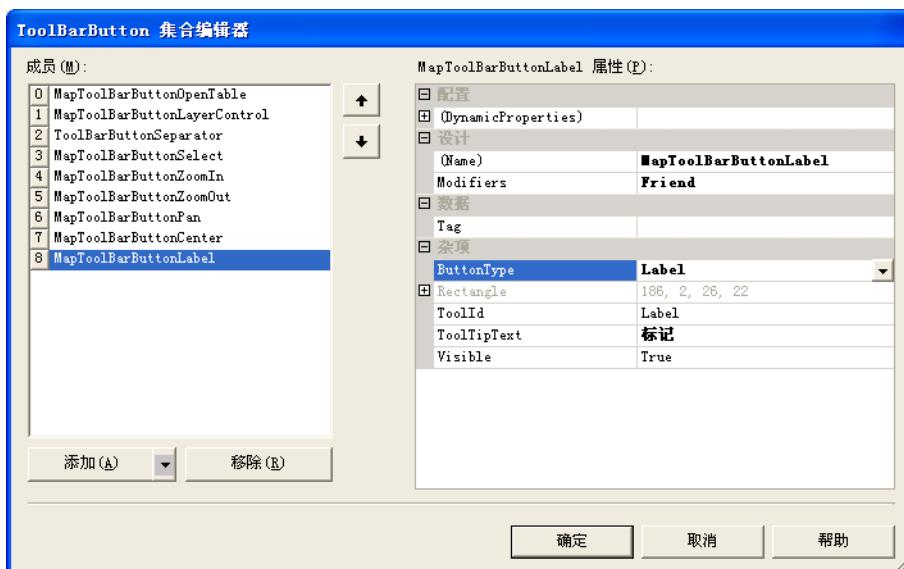
在 Add 中，Custom 和 Select 工具类别是用于绘制或选择以下这些几何体对象的工具： Ellipse、Arc、Circle、Rectangle、Polygon、Point、Line 和 Polyline。

所有工具都具有与它们实现三个接口相关联的属性： `IMapToolProperties`、
`ISelectMapToolProperties` 和 `IAddMapToolProperties`。

View 工具

View 工具（ZoomMapTool、CenterMapTool 和 PanMapTool）将地图的视图直接更改为正在使用的工具。这些工具是默认工具集合的一部分。若使用 MapXtreme 2005 模板构建桌面应用程序，则这些工具将出现在默认工具栏上。

LabelMapTool 可以在地图上标注图元。该工具未包括在默认工具集合内。要将该工具增加到集合中，单击 MapToolBar 的 Buttons 属性下面的“集合 ...”按钮。



Select 工具

Select 工具系列（SelectMapTool、SelectPointMapTool、SelectPolygonMapTool、SelectRadiusMapTool、SelectRectMapTool 和 SelectRegionMapTool）选择位于由这些工具定形的几何对象之内的点和区域。例如，SelectRectMapTool 会选择位于矩形内的对象。SelectMapTool 是派生其它 select 工具的抽象基类。

注：MapXtreme 2005 不支持对象与折线的交集，因此无 select polyline 工具。

SelectPointMapTool 可用于旋转对象。

Select 节点模式允许将独立点从对象中移动、添加或删除。请参阅第 118 页中的教程：使用 *Select 工具编辑 FeatureGeometry* 获取教程。

使用 SelectRectMapTool、SelectRadiusMapTool 和 SelectPolygonMapTool，动态 Selection 显示当鼠标移动时何种对象可以在选择集内。

Selection 地图工具具有设置各种选项的属性，包括哪些图层是可选的。这些默认的工具属性可在 MapTools 集合上设置，该集合位于 MapInfo.Windows.Controls.MapControl.Tools。每个工具也可以设置自己的重载属性并指定使用 MapTools 集合上的默认值或自己的重载值。

要设置可选图层的默认列表，请创建 IMapLayerFilter，指定哪些图层是可选的。然后在 MapTools 上为 SelectableLayerFilter 设置默认属性。

例如，要设置在所有矢量图层上使用的可选图层的默认列表：

```
// select from all vector layers
IMapLayerFilter selectableLayerFilter =
    MapLayerFilterFactory.FilterAnd(MapLayerFilterFactory.FilterVisibleLayers(true), MapLayerFilterFactory.FilterByLayerType(LayerType.Normal));

mapControl1.Tools.SelectMapToolProperties.SelectableLayerFilter =
    selectableLayerFilter;
```

有关 Select 工具属性的更多信息，请参阅 MapXtreme 2005 开发人员参考帮助中的 MapInfo.Tools.MapTool.ISelectMapToolProperties。

Add 工具

Add 工具系列（AddArcMapTool、AddCircleMapTool、AddEllipseMapTool、AddLineMapTool、AddPointMapTool、AddPolygonMapTool、AddPolylineMapTool 和 AddRectangleMapTool）全部基于 AddMapTool 类。Add 工具可以在地图上绘制特定的几何对象。要将对象增加到地图，那么地图中必须具有需要激活的插入图层，创建对象后，对象会驻留在该图层中。特定对象的大小和形状取决于所使用的工具，并可在工具使用过程中通过修饰符键（Shift 和 Ctrl）进行约束。

有关 Add 工具属性的更多信息，请参阅 MapXtreme 2005 开发人员参考帮助中的 MapInfo.Tools.MapTool.IAddMapToolProperties。

Custom 工具

Custom 工具（CustomArcMapTool、CustomCircleMapTool、CustomEllipseMapTool、CustomLineMapTool、CustomPointMapTool、CustomPolygonMapTool、CustomPolylineMapTool 和 CustomRectangleMapTool）是非常基本的工具，它们仅用于激发事件。可以使用这些类设计特定工具的定制行为。例如，每次使用这些工具时，都可以用 CustomEllipseMapTool 绘制具有红色轮廓的绿色 Ellipse。还可以使用各种工具事件（请参阅 [第 116 页中的工具事件](#)）指定对这些工具所生成的特定事件作出响应的特定行为。

自定义工具未包括在默认的工具集合内。若要将自定义工具分配给鼠标按键，则可能要提供标识该工具的任意字符串。

有关自定义工具的详细信息请参阅 [第 115 页中的自定义工具](#)。

Shape 工具

MapXtreme 2005 提供了一组工具，用于在地图上绘制几何图元。大多数工具都要通过“单击和拖放”使用。在地图上单击并拖放到另一位置时，会显示一个橡皮圈图像，它显示正在绘制的对象的当前大小和形状。这些工具都使用 ESC 键取消当前操作（如果适用的话）。

以下是 MapXtreme 2005 支持的 shape 工具类型。

Line 工具

该组中的工具包括 AddLineMapTool 和 CustomLineMapTool。这些工具绘制 Line。通过单击并从起点拖动到终点来激活该工具。松开鼠标按键将创建 Line。如果在拖动过程中按下 Shift 键，则 Line 的角度将被约束为 45 度的倍数。如果在拖动过程中按下 Ctrl 键，则 Line 的长度和高度加倍。在松开鼠标按键之前按下 Esc 键，将取消该操作。

Polyline 工具

该组中的工具包括 AddPolylineMapTool 和 CustomPolylineMapTool。该组中的工具绘制具有多个段的 Line。通过单击一个点然后再单击后续点可以激活该工具。完成单击点之后，再次单击最后一个点，则完成 Polyline 绘制。如果在拖动过程中按下 Shift 键，则 Line 的角度将被约束为 45 度的倍数。Ctrl 键不会影响该工具。在完成线段之前按下 Esc 键，将取消该操作。

Ellipse 和 Arc 工具

该组包括的工具有：AddEllipseMapTool、CustomEllipseMapTool 和 CustomArcMapTool。Ellipse 工具会绘制 Ellipse。CustomArcMapTool 仅会绘制圆弧；但不会将圆弧插入图层。

通过单击并将边缘一端上的点拖动到边缘另一端上的点来激活 Ellipse 工具。松开鼠标按键将创建 Ellipse。如果在拖动过程中按下 Shift 键，则 Ellipse 被约束在半径为常量的圆中。如果在拖动过程中按下 Ctrl 键，则使用中心内的起点绘制 Ellipse。在松开鼠标按键之前按下 Esc 键，将取消该操作。请注意 Ellipse 轴始终与坐标系对齐。

MapXtreme 2005 提供了轻量级的 CustomArcMapTool，该工具在地图上绘制参考圆弧，但不会提供 LegacyArc FeatureGeometry 或将其插入图层中。要创建带有上述行为的 arc 工具，必须创建您自己的实现方法。例如，若想要 CustomArcMapTool 与 AddMapTool 具有相同的操作，则创建从 CustomArcMapTool 派生的新 arc 工具类并实现 IAddMapToolProperties 接口。

要查看说明该过程的 C# 和 VB 示例代码，请转到 MapInfo Knowledgebase 中 *MapXtreme 2004 > 工具*。可从“程序”菜单上 MapXtreme 2005 学习资源浏览器或 MapInfo 网站的 [支持和培训](#) 访问该 Knowledgebase。

Circle 工具

该组中的工具包括 AddCircleMapTool 和 CustomCircleMapTool。Circle 工具绘制圆形。这与按下 Shift 键时使用 Ellipse 工具具有同样的功能。通过单击并从 Circle 中央的中心点拖动到边缘上的点来激活该工具。拖动时，用户可看到 Circle 的大小在该中心点周围进行扩展。松开鼠标按键将创建 Circle。如果在拖动过程中按下 Ctrl 键，则以两个边缘点绘制 Circle。在松开鼠标按键之前按下 Esc 键，将取消该操作。

Rectangle 工具

该组中的工具包括 AddRectangleMapTool 和 CustomRectangleMapTool。这些工具绘制矩形。通过单击并从一个顶点拖动到相对的顶点来激活该工具。松开鼠标按键将创建 Rectangle。注意，绘制的对象始终轴对齐。如果在拖动过程中按下 Shift 键，则 Rectangle 被约束为正方形。如果在拖动过程中按下 Ctrl 键，则使用中心内的起点绘制 Rectangle。在松开鼠标按键之前按下 Esc 键，将取消该操作。

自定义工具

可通过两种方式定制工具：通过创建现有工具的子类或使用 Custom Tools 组中的其中一个工具来自定义工具。在 MapXtreme 2005 中提供的任何 Tool 类都可通过创建子类自定义，而子类是从需要的特定 Tool 类派生的。下面的示例就是从 AddLineTool 类派生新的工具类。用户可能想将该工具更改为特定行为（例如，绘制直线时，始终发出蜂鸣声）、特定外观（例如，始终以红色绘制直线）或者特定功能（例如，始终将绘制直线的角度约束为添加 90 度）。

另一个示例是基于同一个工具创建两个不同的工具。可以使用 AddPointTool 划分两个不同工具的子类，这两个工具使用不同的符号以指示地图上两种类型的点。将其中一个工具指定给鼠标左键，将另一个工具指定给鼠标右键。

以下代码示例说明了 AddPolygonTool 的定制。

VB 示例：

```
Dim insertionlayerfilter As IMapLayerFilter
Dim style As MapInfo.Styles.CompositeStyle
Dim addmaptoolproperties As MapInfo.Tools.AddMapToolProperties
Dim maptool As MapInfo.Tools.MapTool

insertionlayerfilter =
    MapLayerFilterFactory.FilterByLayerType(LayerType.Normal)
style = New MapInfo.Styles.CompositeStyle
```

```
addmaptoolproperties = New _  
MapInfo.Tools.AddMapToolProperties(MapLayerFilterFactory.Filter_ _  
ForTools(MapControl1.Map, insertionlayerfilter, _  
MapLayerFilterFactory.FilterVisibleLayers(True), _  
"CustomPolygonAddMapToolProperties", Nothing), style)  
  
maptool = New MapInfo.Tools.AddPointMapTool(MapControl1.Viewer, _  
MapControl1.Handle.ToInt32(), MapControl1.Tools, New  
MapInfo.Tools.MouseToolProperties(Cursors.Default, Cursors.Default, _  
Cursors.Default), MapControl1.Tools.MapToolProperties, _  
addmaptoolproperties)
```

MapInfo KnowledgeBase 包含附加的代码示例，用于自定义桌面工具。要获得如何创建桌面 InfoTool 示例，请转到 [MapXtreme 2004 > 工具](#)。

此外，MapXtreme 2005 提供了桌面工具示例应用程序，在该应用程序中可以按需分析和实现桌面工具。请参阅 MapXtreme 2005 安装目录下的

..\\MapInfo\\MapXtreme\\6.5\\Samples\\Desktop\\Features\\DesktopTools\\cs。

工具事件

MapInfo.Tools 命名空间支持带有信息的 Tool 事件并具有取消工具操作的能力。Select 工具事件列出选定的或取消选定的项。可在使用特定工具的每个阶段激发事件。在代码中，可以在工具使用周期中的某些时刻设置陷阱。在地图绘制应用程序中定制工具时，将代码添加至特定事件就提供了极大的灵活性。

命名空间中的事件如下所示：

FeatureAddingEventArgs

在添加工具将要绘制对象时激发该事件。

FeatureAddedEventArgs

在添加工具已将对象添加到表和地图时，激发该事件。

FeatureSelectingEventArgs

在选择工具将要更改选择集时激发该事件。

FeatureSelectedEventArgs

在选择工具更改选择集时激发该事件。

FeatureChangingEventArgs

在选择工具将要更改图元时激发该事件。使用该事件进行检查以了解将要进行的更改是否有效。

FeatureChangedEventArgs

在选择工具更改图元之后激发该事件。

NodeChangedEventArgs

在选择工具改变选择集中的节点时激发该事件。

NodeChangingEventArgs

在选择工具将要更改选定图元的节点时激发该事件。使用该事件进行检查以了解将要进行的更改是否有效。

ToolActivatedEventArgs

在激活鼠标工具时激发该事件。

ToolActivatingEventArgs

将要激活鼠标工具时激发该事件。

ToolEndingEventArgs

在鼠标工具将要结束时激发该事件。此时可以开始一个新操作。

ToolUsedEventArgs

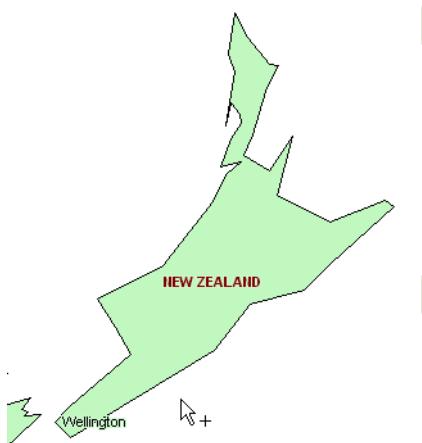
该事件在鼠标工具正在使用时激发。可以使用该事件为鼠标单击连续点的开始，中间和结束处设置标记。

教程：使用 Select 工具编辑 FeatureGeometry

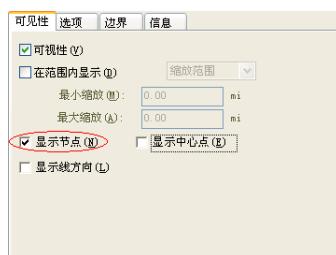
下面的教程说明如何通过重新定形节点并将节点增加到 FeatureGeometry 来编辑图元。本教程说明了 MapControl、LayerControl 和 Select 工具的使用。另请参阅第 124 页中的 *MapControl* 和第 128 页中的图层控件。

重新定形图元

1. 将地图加载到 MapControl 并放大想要修改的 Feature。



2. 在 LayerControl 中选择对象的图层。
3. 在 LayerControl 的“可见性”选项卡上，选择“显示节点”复选框。



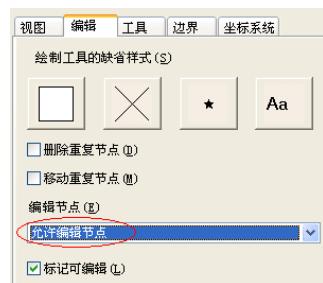
4. 在“选项”选项卡上，选择“可选”和“可编辑”复选框。



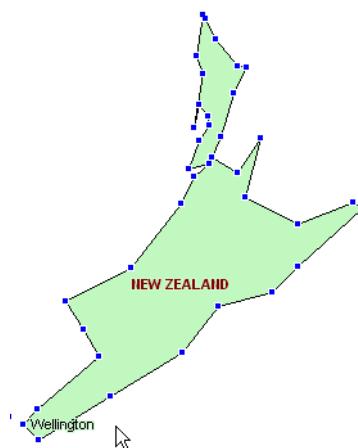
5. 在图层控制树视图中选择地图的根节点。



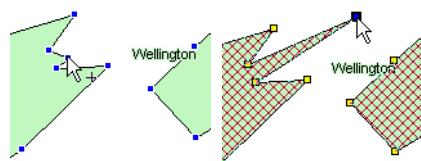
6. 在“编辑”选项卡上，从“编辑”模式组合框选择“允许编辑节点”。



7. 单击“确定”接受更改。此时地图显示将会更改并显示该对象的节点。



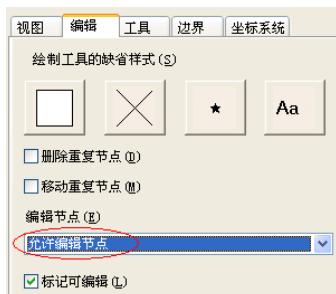
8. 使用 Select Item 工具，选择想要修改的多边形，然后单击并拖动节点以更改其位置。
使用 Shift 键增加节点，使用 Ctrl 键从组中删除节点。



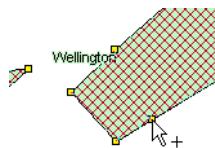
增加节点

要将节点增加到 Feature Geometry：

1. 再次打开 LayerControl 并选择地图的根节点。
2. 在“编辑”选项卡上，从“编辑”模式组合框选择“允许增加节点”。



3. 单击“确定”接受更改。
4. 在地图上，选择想要修改的多边形。
5. 使用 Select Item 工具，按住 Ctrl 键并单击想要增加节点的多边形的边缘。此时将会显示新节点。



通过编程方式重新定形和增加节点

此教程说明了如何通过编程方式重新定形和增加节点。提供有 C# 和 VB 代码。

1. 将名为 "btnEditNodeTool" 的按钮增加到 Visual Studio .NET 设计器的主窗体中。
2. 双击该按钮以在代码页中打开该按钮的句柄。增加适当的代码示例：

C# 示例：

```
private void btnEditNodeTool_Click(object sender, System.EventArgs e)
{
    mapControl1.Tools.LeftButtonTool = "Select";
    mapControl1.Tools.SelectMapToolProperties.EditMode =
    MapInfo.ToolsEditMode.Nodes;
}
```

VB 示例：

```
Private Sub btnEditNodeTool_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    MapControl1.Tools.LeftButtonTool = "Select"
    MapControl1.Tools.SelectMapToolProperties.EditMode =
    MapInfo.Tools.EditMode.Nodes
End Sub
```

3. 运行应用程序。
4. 在 LayerControl 中选择对象的图层。
5. 在 LayerControl 的“可见性”选项卡上，选择“显示节点”复选框。



6. 在“选项”选项卡上，选择“可选”和“可编辑”复选框。



7. 单击“确定”接受更改。
8. 现在，单击已增加到窗体的新按钮。此时，光标将会更改为 Select 箭头。
9. 选择要修改的对象，然后单击并拖动节点（与上一示例步骤 8 中所述）。

8

用于桌面应用程序的对话框和控件

本章包括了为 MapXtreme 2005 应用程序创建图形用户界面时可使用的各种控件和对话框的说明。

本章内容：

- ◆ *MapInfo.Windows.Controls* 命名空间 124
- ◆ 要在应用程序中使用的关键控件 124
- ◆ *MapInfo.Windows.Dialogs* 命名空间 129
- ◆ 定制控件和对话框 135

MapInfo.Windows.Controls 命名空间

MapInfo.Windows.Controls 命名空间包括实现桌面应用程序中控件的类。使用 Windows 控件十分简单，可以将需要的控件放到窗体上并且可以使用各种属性对它们进行配置以满足您的规范。您将发现在此命名空间中的控件类似于 MapInfo.WebControls 命名空间中的控件。但是，每个命名空间中的控件不能彼此替换。此命名空间中的控件特定用在桌面应用程序中。有关基于 Web 的应用程序的控件的信息，请参阅第 6 章：运用 *MapXtreme 2005 Web 控件*。

注： 某些控件不能在设计阶段显示。将它们拖放到窗体时，这种控件仅显示为矩形。这些控件在运行时会正确显示。

MapXtreme 2005 包括的桌面控件可分为两类：标准控件和地图控件。标准控件包括：按钮、视图工具、标注工具和各种各样的方框等。它们在很多情况下与 System.Windows.Forms 命名空间中的类十分类似，或者它们就是从 System.Windows.Forms 命名空间中的类继承而来的。

地图控件特定于 MapInfo 的地图绘制实现。其中包括设置或修改对象样式、标注、图层、坐标系、主题以及其它相关功能方面的控件。

要在应用程序中使用的关键控件

虽然包括在 MapXtreme 2005 中的所有控件都非常有用，但有几个控件比其它控件在大多数地图绘制应用程序中更为重要。以下章节将讨论这些关键控件。其中的每个控件都假定项目中包括了适当的引用。（如果已经为 Visual Studio 中使用 MapXtreme 2005 安装的地图绘制模板创建项目，则将自动包括这些引用。）

注： 如果未从我们的模板之一创建项目，则请确保将引用添加到相应组件（例如：MapInfo.CoreEngine、MapInfo.Windows、MapInfo.Windows.Dialogs 和 MapInfo.Windows.Framework）。另请参阅第 36 页第 2 章中的 *MapXtreme 2005 合并模块*。

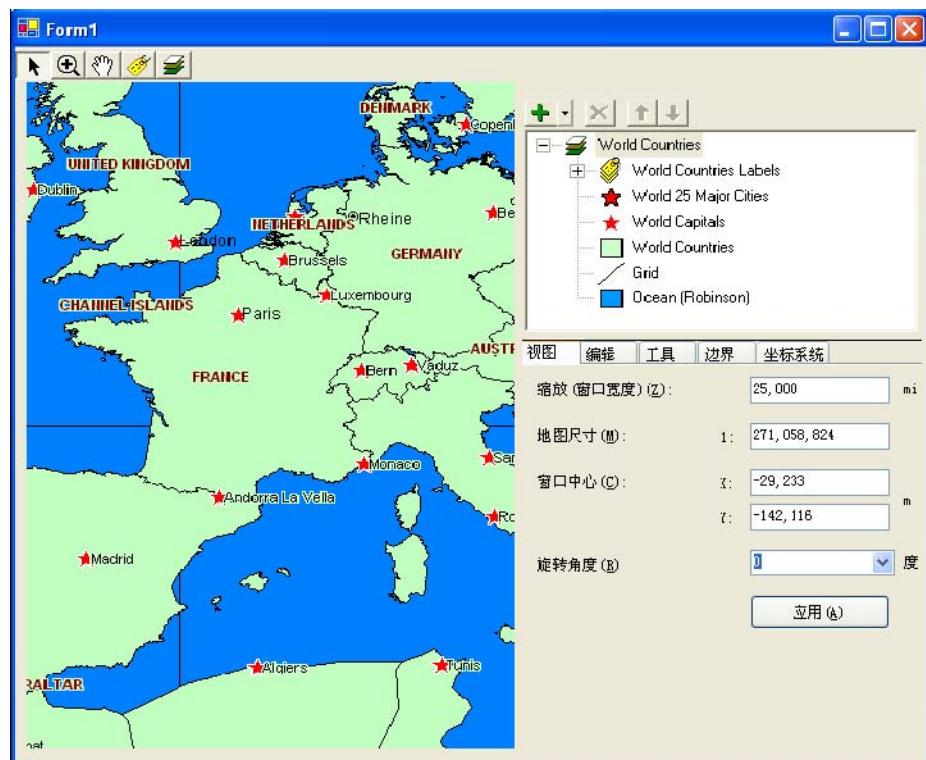
MapControl

每个显示地图的应用程序都需要 MapControl。要将 MapControl 添加到窗体上，只需将它从 Visual Studio 的工具箱中拖放到窗体上即可。MapControl 位于窗体之后，可对其执行多种操作，让用户可以更好地使用地图。

在窗体上选择 MapControl 时可通过“属性”窗口执行对 MapControl 的修改。在设计阶段右键单击 MapControl，显示具有以下选项的上下文菜单：**加载地图**、**清除地图**、**图层控制**和**创建主题**。**加载地图**打开一个标准文件选取器，开发人员可从中选择要预加载到 MapControl 中的地图。**清除地图**在将 MapControl 添加到窗体或设计过程中的某个位置时清除任何已在控件中的地图。**图层控制**启动允许定制地图的“图层控件”对话框（请参阅第 128 页中的图层控件）。**创建主题**启动允许在地图上创建主题的 CreateThemeWizard（请参阅第 130 页中的 CreateThemeWizard）。

MapControl 还具有多个与已加载的地图相关的属性，可以预设置已加载的地图，为特定应用程序进一步定制在运行时显示的地图。可以修改的属性包括：缩放级别、坐标系、地图中心和运行时期间三个鼠标按键的设置。可将鼠标按键设置为 Zoom、Pan、Draw geometries、Select 或其它任何几个地图工具。请参阅第 110 页中的 *MapInfo.Tools* 命名空间概述。

还可以设置设计阶段工具以在 Visual Studio .NET 中工作时进一步操控地图。设计阶段功能包括 Zoom in、Zoom Out、Select、Pan、Center 和默认的箭头。上下文菜单中的“清除地图”从 MapControl 地图中删除所有图层并关闭相应的表（如果它们未在另一个 MapControl 中使用的话）。



显示 MapControl、LayerControl 和 MapToolBar 的窗体

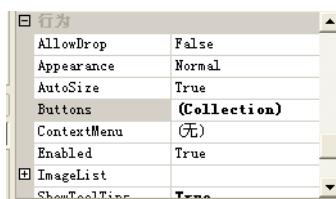
MapToolBar

MapToolBar 将多个地图工具控件（例如，放大和打开表）组合为单个控件。

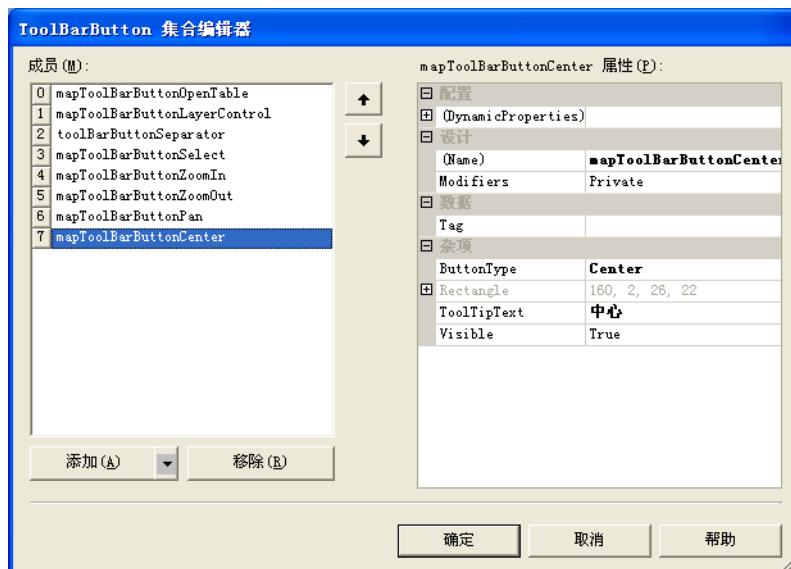
如果使用 MapXtreme 2005 随附的模板创建地图应用程序，则该应用程序将包括 MapToolBar。或者，可以通过将 MapToolBar 从 Visual Studio 工具箱的拖动将它添加到窗体。默认情况下，MapToolBar 包括下列按钮：OpenTable、LayerControl、Select、ZoomIn、ZoomOut、Pan 和 Center。

还可以通过使用下列步骤将按钮添加到 MapToolBar：

1. 在“属性”窗口中通过单击 Buttons 属性旁边的省略号 (...), 打开 ToolBarButton 集合编辑器。



显示 ToolBarButton 集合编辑器。



2. 单击**添加**按钮。

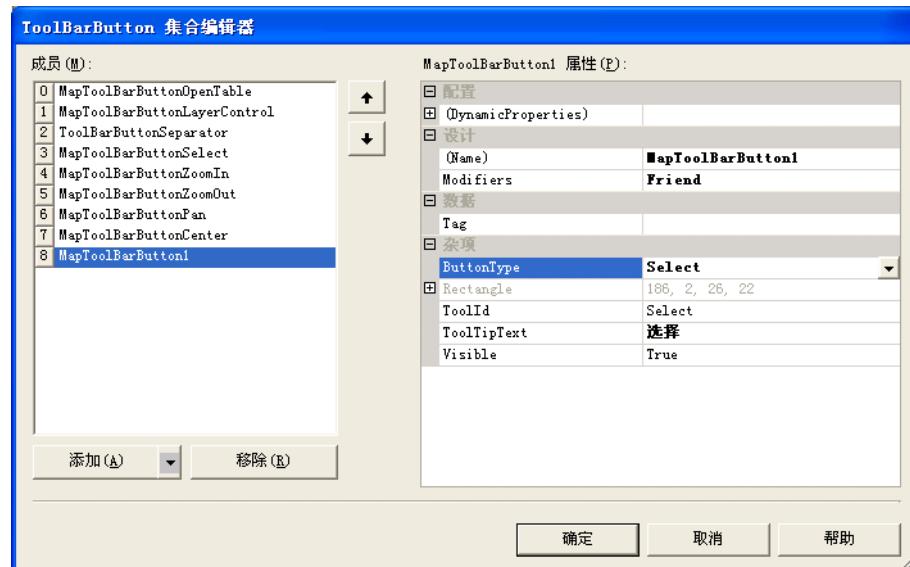
将在列表中最后一个按钮的后面创建新的 MapToolBarButton。

注：如果想要添加定制的工具，则需要单击“添加”按钮旁的箭头并选择 ToolBarButton。需要为新按钮编写定制的处理程序。

3. 命名新按钮。

4. 从右侧下拉菜单选择 ButtonType。

例如，如果添加 Select 工具按钮，则从列表中选择 Select。



5. 单击**确定**。

现在创建了 MapToolBarButton 并将其添加到按钮栏。新添加的工具在默认方式下工作。

将自定义按钮添加到 ToolBar

该程序假定已创建自定义工具并将其添加到 MapControl.Tools 集合。添加自定义按钮包括：将按钮的 ToolId 分配到自定义工具的名称和将自定义位图添加到 MapToolBar 的 ImageList。

有关如何创建自定义桌面工具的详细信息，请参阅第 115 页中的自定义工具。

有关 MapXtreme 2005 包含的自定义位图符号的详细信息，请参阅第 499 页中的定制符号。

要将自定义按钮添加到工具栏：

1. 将 MapToolBar 添加到窗体。
2. 在 Visual Studio 属性窗口中，突出显示“按钮”属性并按下 ... 按钮，调用 ToolBarButton 集合编辑器对话框。
3. 单击**添加**按钮，添加新的 MapToolBarButton。
4. 将按钮的 ButtonType 属性设置为 CustomTool（下拉列表中最后一项）。注意，按钮将显示为空白，因为暂时没有与该自定义按钮关联的图像。
5. 将按钮的 ToolId 属性设置为自定义工具的名称，该工具已添加到 MapControl 的工具集合。如果 ToolId 值与集合中任意工具都不匹配，则在用户单击该按钮时抛出运行时异常。
6. 单击**确定**，关闭 ToolBarButton 集合编辑器对话框。
7. 调用 InitializeComponent 之后，在窗体的构造函数中将要添加自定义位图的节点添加到 MapToolBar 的 ImageList。自定义位图与应用程序的关联方式由程序员决定。一种选择是在设计阶段将 ImageList 添加到窗体，用自定义图像填充，然后在运行时写入代码以便将其图像传输到工具栏的 ImageList。代码将如下所示：

```
// Add custom tool button's bitmap to toolbar's image list
foreach (Image image in this.imageList1.Images) {
    this.mapToolBar1.ImageList.Images.Add(image);
}

// Associate the bitmap with the custom tool's button
(last image in the list)
this.mapToolBarButtonBlueSelect.ImageIndex =
this.mapToolBar1.ImageList.Images.Count-1;
```

图层控件

“图层控件”对话框显示构成当前地图的所有图层以及图层属性的状态。这些属性是：可视、可编辑、可选择和自动标注。每个复选框列上面的图标表示这些属性类型。当将光标移到属性图标上面时，图标上面将显示“工具提示”以帮助您熟悉每个图标。使用复选框可以轻松地更改一个或多个图层的属性。

“图层控件”还具有可用于更改 Display 和 Label 设置；修改任何已创建的主题地图以及重新排序、添加或删除图层的选项。

可以通过在“图层控件图层”列表中拖放图层来重新排序 LayerControl 中的图层。

注： 将图层放在“标注图层”上时会将新的标注集添加到“标注图层”。例如，如果尝试将图层正好移动到“标注图层”上方的位置，就会偶然发生这种情况。

提示： 如果想要将图层放在“标注图层”之外，则在放下图层之前按下 Shift 键。按下并松开 Shift 键时，光标会发生更改以指示要将图层放在目标图层的上面还是目标图层的内部。

“图层控制”将“图层控件”对话框的所有功能放在一个窗体上。在 Form_Load() 方法中需要一行代码将“图层控件”链接到 MapControl 的地图。

```
layerControl1.Map = mapControl1.Map;
```

要更好地了解“图层控件”的复杂性和功能，请运行名为 Workspace 管理器的 MapXtreme 2005 实用程序。该工具包括“图层控件”的运用示例。请参阅第 19 章：*Workspace 管理器*。可通过 MapInfo\MapXtreme 6.x 下的“程序”菜单访问 Workspace 管理器。

定制上下文菜单

可为“图层控件”创建自定义上下文菜单项，这些项目在用户右键单击图层树中的节点时显示。使用 ContextMenuTargetObject 属性返回用户右键单击的对象。

提供的代码示例说明了如何定义 LayerControlEnhancer 类，该类允许用户将自定义项目添加到“图层控件”的上下文菜单中。可在位于

..\MapInfo\MapXtreme\6.x\Samples\Desktop\Features\LayerControl 目录下的 LayerControl 示例应用程序中找到该示例。

快捷键编程

可以创建用于访问 LayerControl 工具栏上全部内容的快捷键。PerformDown、PerformUp 和 PerformRemove 方法提供了对“下移”、“上移”和“删除”按钮的编程式访问。

AddMenuMnemonic 属性提供了显示“添加”菜单的键的编程式访问。AddMenuMnemonic 属性的语法为：

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

MapInfo.Windows.Dialogs 命名空间

MapInfo.Windows.Dialogs 命名空间包括创建具有特定功能的对话框的类。与控件相反，对话框在设计阶段不是可视的，并且只能在代码中创建和配置。可以使用 MapInfo.Windows.Dialogs 命名空间中的类创建您自己定制的对话框，然后通过调用 System.Windows.Forms.Form.ShowDialog() 方法来利用它们。

要在应用程序中使用特定对话框，请按下面的办法将 MapInfo.Windows.Dialogs 命名空间添加到代码：

使用 MapInfo.Windows.Dialogs

```
private void DoLayerControl()
{
    LayerControlDlg layerControl = new LayerControlDlg();
    layerControl.Map = mapControl1.Map;
    layerControl.ShowDialog(this);
}
```

上面的代码在调用 DoLayerControl() 方法时显示 LayerControl 对话框。

常用对话框

MapXtreme 2005 包含许多常用对话框和控件，用于处理操纵 MapXtreme 对象的任务。Workspace 管理器中使用的所有对话框和控件都可在您的应用程序中使用。这些对话框只是低级控件的容器。这些控件设计用于在特殊情况下与多数 MapXtreme 对象一起使用。

常用对话框使用包含 MapXtreme 的控件创建基本的 UI 组件。这些控件置于对话框中，用于创建特定的 UI 组件。同样地，您可以使用基本控件设计自己的对话框。要创建对话框，只需创建对话框类，然后将控件拖放到表面。

也可以通过可视化继承来自定义常用对话框。可以从我们的对话框中派生类，然后通过重载方法和属性自定义某些行为。其它对话框（如“图层控件”）都可定制，因此可以更改默认行为，以便隐藏不想让用户访问的控件、更改图标的外观或删除选项卡。

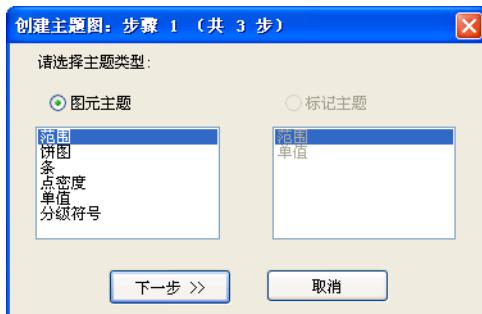
基本上可以控制 MapXtreme 包含的所有 UI 组件。这样就可以创建自己的自定义界面。使用上述任何方法都可以创建自定义应用程序，该应用程序只公开所需内容，而不是 MapXtreme 2005 定义的所有内容。

CreateThemeWizard

CreateThemeWizard 类是一种可添加到应用程序的简洁类。该类可以启动向导（一组对话框）以指导用户完成创建新主题的过程。创建主题有三个基本步骤：1) 选择主题类型；2) 选择用于主题的表和列；3) 修改主题属性（样式、范围数量等）。向导将所有这些设置集成到连续的对话框，以使 MapXtreme 应用程序的最终用户可以很简单地创建主题。

使用 CreateThemeWizard

根据用户指定的选择，CreateThemeWizard 显示几种不同的对话框。通过第一个对话框可以选择主题类型。



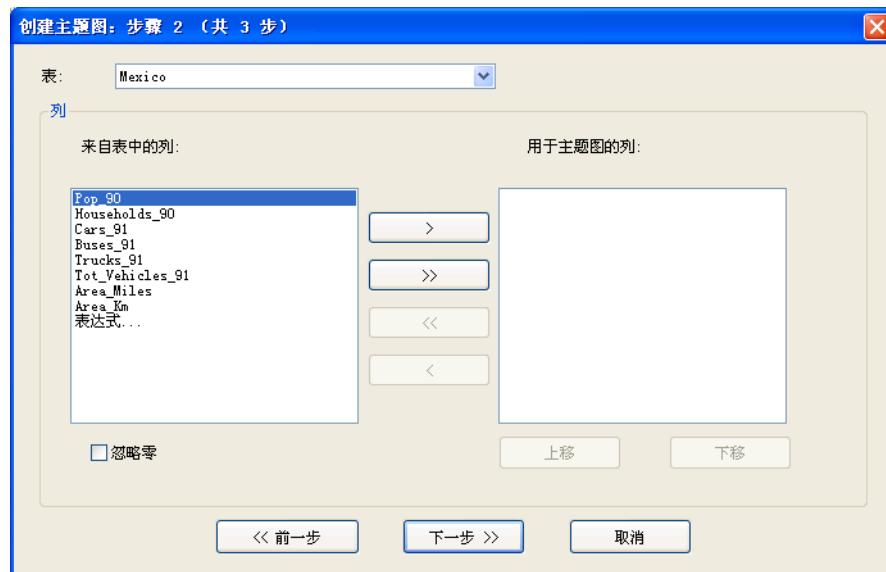
创建主题图：步骤 1 (共 3 步) 对话框

如果用户选择单变量主题类型（“范围”、“点密度”、“单值”或“分级符号”），则显示的第二个对话框允许选择表和单一列。



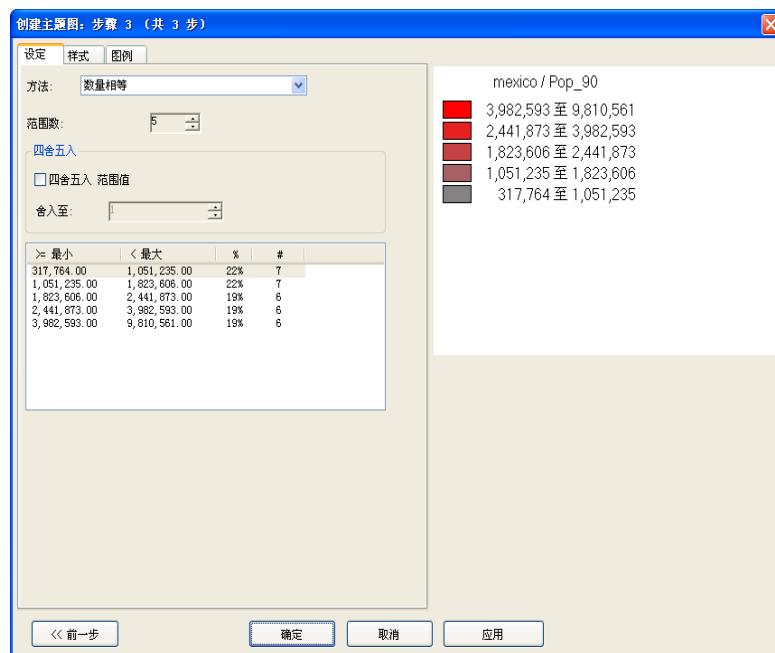
创建主题图：步骤 2 (共 3 步) 对话框 (单一列)

如果用户选择多变量主题（饼图或条形图），则显示的第二个对话框允许选择从中创建主题的多列数据。



创建主题图：步骤 2（共 3 步）对话框（多列）

以下对话框特定于在步骤1中选择的主题类型。下图显示了特定于范围主题的步骤3对话框。



创建主题图：步骤3（共3步）对话框（范围主题）

根据应用程序中的控件，在设计阶段和运行时，可从多个不同地方访问 CreateThemeWizard。在设计阶段，可从位于 MapControl 属性窗口底部的菜单访问 CreateTheme 向导；还可以通过单击一个图层，然后选择[添加主题 ...](#)从 LayerControl 访问 CreateThemeWizard。

使用 CreateThemeWizard 进行开发

要通过编程方式添加 CreateThemeWizard，请执行以下示例代码。

```
using MapInfo.Windows.Dialogs;  
  
CreateThemeWizard themeWizard = new CreateThemeWizard(mapcontrol1.Map,  
this);
```

方法

创建向导后，可从 CreateThematicWizard 类调用 CreateTheme() 方法来创建主题。可添加可选字符串参数以设置主题的别名。在 ObjectTheme (Bar、Pie 和 GraduatedSymbol) 情况下，可以使用别名访问图层集合中的主题，或者在 FeatureStyleModifier (Ranged、RangedLabel、IndividualValue、IndividualValueLabel 和 Dot Density) 情况下，可以使用别名访问修饰符集合。可以访问主题以进行修改或删除。

```
ITheme theme = themeWizard.CreateTheme("theme1");
```

如果已经了解要作为主题列出的表或要用于标注主题的标注源，则可以使用方法 CreateFeatureTheme() (用于表) 或 CreateLabelTheme() (用于标注)。

属性

以下是一些属性，对获取维护有关主题创建的信息非常有用。WizardResult 包括允许检查用户如何退出向导的枚举 WizardStepResult。可以选择 WizardStepResult.Done 和 WizardStepResult.Cancel。它们对适当地更新应用程序中的控件或菜单非常有用。

以下示例说明了如何使用。

```
if (createThemeWizard.WizardResult == WizardStepResult.Done
{
    // Update the menus
    mnuRemoveTheme.Enabled = true;
    mnuModifyTheme.Enabled = true;
}
```

SelectedLabelSource、SelectedLayer 和 SelectedThemeType 是 CreateThemeWizard 类上可用于找出主题已应用了哪些图层或标注源的属性。这在想要修改主题或需要访问对象时非常有用。

要修改主题，首先要确定主题的类型，然后启动适当类型的修改对话框。用于修改主题的对话框类包括 ModifyBarThemeDlg、ModifyDotDensityThemeDlg、ModifyGradSymbolThemeDlg、ModifyIndValueThemeDlg、ModifyPieThemeDlg 和 ModifyRangedThemeDlg。

MapXtreme 2005 包含 ThemeDialogs 示例应用程序，可以分析其中的实现并根据需要对它进行自定义。请参阅 MapXtreme 2005 安装下的 ..\MapInfo\MapXtreme\6.x\Samples\Desktop\Features\ThemeDialogs。

定制控件和对话框

MapXtreme 2005 在 `MapInfo.Windows.Dialogs` 命名空间中随附了多种对话框类。其中的每个对话框类都具有特定于每个对话框类的定制功能。要修改特定对话框，只需将值指定给特定于每个对话框的不同属性。因为将值指定给属性的情况非常多，所以无法列出全部的定制设置。请参阅联机参考以获取关于每个对话框的详细信息。

定制对话框的示例如下所示：`LineStyleDlg` 类创建 Line Style 对话框。此对话框可以使用稀疏样式模式（打开时未选择任何内容）创建或使用某些选项是可视的但未激活的模式（灰色）创建。

由设置属性值提供的定制局限于要修改的特定对话对象。通过在设计阶段为新窗体添加一些控件，可基于已建立对话框的设计创建新的对话框。例如，要创建修改后的 `LineStyle` 对话框，可以将 `LineStyle` 控件放在新窗体上，然后为那个窗体添加其它控件，并使它们对已修改的 `LineStyle Dialog` 框生效。

MapInfo 的知识库提供了定制桌面控件的示例。例如，要创建自定义 `LayerControl`，请参阅 *MapXtreme 2004 > Selections* 下的相关主题。

运用核心 MapXtreme 2005 类

MapInfo.Engine 命名空间包含了所有直接与核心功能有关的接口和类，该核心功能驱动基于 MapXtreme 2005 的所有应用程序。其中包括核心 ISession 接口，该类是所有 MapXtreme 2005 应用程序的起点。该命名空间中的类包括 Session、Selections 和 SearchPath。命名空间中的其它类型支持 Collections、Resources 和 CustomProperties 的类、委托、结构体和枚举。

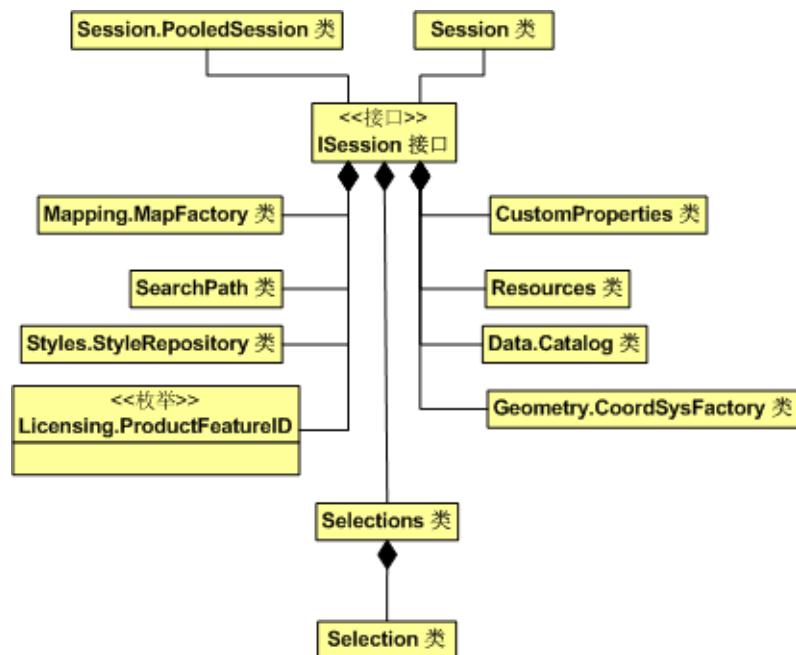
本章内容：

- ◆ *ISession* 接口 138
- ◆ 串行化和永久化 140
- ◆ *Selection* 类 142
- ◆ *Selection* 代码示例 144
- ◆ 事件参数 146
- ◆ 异常 146

ISession 接口

ISession 接口是所有基于 MapXtreme 2005 的应用程序的起点。它管理 MapXtreme 2005 应用程序所需资源的初始化。

ISession 的实例保存 MapXtreme 对象模型的组件，如 DataAccess 引擎、MapFactory、CoordSysFactory，这样桌面或 web 应用程序才能够运行。下图对实现 ISession 接口的类进行了说明。



在 ASP.NET 应用程序中，每个客户端请求都具有其自身的 ISession 实例。该实例驻留在调用上下文中，在客户端请求的整个生命周期中都可用。

在单一线程的桌面应用程序上，只有一个实例。在多线程桌面应用程序中，每个线程都有一个实例。

MapInfo.Engine.Session 类提供了对 ISession 对象的访问。要获得当前的 ISession 实例，请访问 MapInfo.Engine.Session.Current 属性。

会话管理

会话管理是在设计应用程序时需要了解的关键主题。桌面应用程序的会话管理简单易懂（每个用户都具有其自身的 ISession 实例）时，而 web 应用程序则需要考虑或许会有未知数目的用户使用您的应用程序。必须了解如何处理每个用户的状态，以便将正确的信息和可视化显示返回给正确的用户。

MapXtreme 2005 提供了用于构建 web 应用程序的模板，帮助您正确管理状态。有关该主题的讨论请参阅[第 5 章：应用程序的规划](#)。显示有关状态管理、对象池、性能和数据访问的关键决定，有助于在开始编码前的项目设计阶段做出周密的决定。

使用 Session.Dispose 方法

MapInfo.Engine.Session 类包含两个预装的 Dispose 方法。您的选择将取决于要构建的应用程序类型。

Session.Dispose()

Session.Dispose() 会处置 ISession 实例，该实例可通过 Session.Current 属性访问。该方法仅用于多线程桌面应用程序。

不要将该方法用于 web 应用程序或单线程桌面应用程序。对于 web 应用程序，ISession 由 WebSessionActivator 管理。

对于单线程桌面应用程序，会在应用程序关闭或使用 MapXtreme 2005 的 AppDomain 被卸载时自动调用 Dispose。

Session.Dispose(HttpContext) 方法

在使用默认会话状态设置的 web 应用程序中使用 Session.Dispose(HttpContext)，因为此时 ISession 存储在内存中。不要在任何其它配置中调用该方法，因为在任何其它配置中 ISession 实例都不存储在内存中。

状态设置通过下列键表示在应用程序项目的 web.config 文件中：

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

第一个设置是特定应用程序的设置，它控制用于保存和恢复 MapInfo.Engine.ISession 实例状态的机制。该实例可通过 MapInfo.Engine.Session.Current 属性访问。HttpSessionState 设置表示该会话通过 ASP.NET 会话状态进行保存和恢复。该状态通过当前的 HttpContext 公开，属于 HttpSessionState 类型。

第二个设置是 ASP.NET 设置，它控制 HttpSessionState 的保存和恢复方式。InProc 表示 ASP.NET 会话的状态被放置在内存中并且对于每个 ASP.NET ISession 实例都是惟一的。这是默认设置。

使用这些设置时，每个 ASP.NET 会话都有一个 ISession 实例，该实例在 ASP.NET 会话的整个生命周期都存储在 HttpSessionState。要在会话暂停或结束时正确处置 ISession 实例，则必须将下列语句添加到 Global.asax 源码文件下的 Session_End 方法中。

C# 示例：

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

VB 示例：

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

执行该调用可以确保正确处置 ISession 实例且回收占用的内存。

串行化和永久化

Map 绑定到 ISession 对象。无法从一个 ISession 获取 Map 对象，而在另一个 ISession 对象中使用 Map 对象。如要克隆整个 ISession 对象，可使用两个选项 – 通过使用 Serialization 或通过使用 Persistence。

串行化

串行化是将对象转换为数据流的过程，从而将对象保留在服务器上。该过程是在 MapXtreme 2005 web 应用程序中维护对象的基本部分。如果对象没有维护，则服务器需要为每个 web 请求重建对象（例如地图）。对象被请求时，首先反串行化对象（或从数据流重建），然后修改对象。因为该算法不生成对象的副本（与其它串行化算法相同），所以反串行化的对象仅创建一次。

串行化通过嵌入 Microsoft.NET Framework 中的格式化程序来执行。Framework 中包括两个不同的格式化程序，一个是用于二进制对象的 BinaryFormatter，另一个是用于 SOAP 对象的 SOAPFormatter（SOAP 是轻型协议，用于在诸如 web 的分布式环境中交换结构化信息）。相对来说，SOAPFormatter 比 BinaryFormatter 要快。SOAPFormatter 用于某些基本的数据类型（Int、Byte、Decimal 等），而 BinaryFormatter 用于复杂的对象。有关 SOAPFormatter 和 BinaryFormatter 的详细信息请参阅 Microsoft MSDN 文档。

要将对象传递到其中一个格式化程序，请使用 GetObjectData() 方法。要反串行化对象（从数据流恢复对象），请使用 SetObjectData() 方法。

任一支持 `ISerializeable` 接口的对象都将被自动恢复或反串行化。ASP.NET 框架在反串行化 `HttpApplication.BeginRequest` 之后自动将 `context.Session[]` 反串行化。`MapInfo.Engine.Session` 在 `HttpApplication.BeginRequest` 处理程序中设置，这样就可将对象反串行化为 `MapInfo.Engine.Session`。

有关 MapXtreme 2005 中其它串行化类的详细信息，请参阅[第 68 页第 5 章中的反串行化其自身的类](#)。

串行化 / 反串行化 Session 对象

以下实例说明了如何将 Session 对象串行化 / 反串行化。

```
// Create a MemoryStream to serialize into
MemoryStream stream = new MemoryStream();
// Serialize the Session object
BinaryFormatter formatter = new BinaryFormatter();
formatter.Serialize(stream, Session.Current);
stream.Position = 0;
// Make changes to the Session object to make sure the
// deserialization works correctly
...
// Recreate the Session object from the stream
// Note: this will replace the current Session object with the
//       contents of the stream
formatter = new BinaryFormatter();
formatter.Deserialize(stream);
```

传递到 `formatter.Serialize` 方法的数据流参数可以是从 `System.IO.Stream` 派生的任何内容。

该串行化功能设计用于与 MapXtreme 2005 本发行版中的 State 功能一起使用。如将该信息保存到磁盘，则以后尝试重载该信息时（用本产品的其它版本）不保证能正常工作。

永久化

MapXtreme 2005 中的永久化是将应用程序对象存储为 `MapInfo Workspace (.MWS)` 文件（XML 格式）的过程。永久化和反永久化 `ISession` 对象可通过使用 `WorkSpacePersistence` 和 `WorkSpaceLoader` 类来完成。这两个类将写入和读出 `.MWS` 文件。应用程序可写出工作空间（`Session` 的副本），并将工作空间文件的内容应用到新的 `Session` 对象，从而创建 `Session` 的克隆。

VB 示例：

```
Public Shared Sub MapInfo_Persistence_WorkSpacePersistenceSave()
    ' Create a named connection point to "D:\data\version2"
    Dim info As NamedConnectionInfo = New NamedConnectionInfo("file",
        ConnectionMethod.FilePath, "D:\data\version2")
    Session.Current.Catalog.NamedConnections.Add("MyDataFolder", _
```

```
info)

' Create a map
Dim map As Map =
Session.Current.MapFactory.CreateEmptyMap("MyMap", __
    "MyMapAlias", New Size(400, 400))
Dim table As Table =
Session.Current.Catalog.OpenTable("MyDataFolder",
    "myTableAlias", "Seamless\Lines - NYALBA\SeamCapDist.TAB")
map.Layers.Add(New FeatureLayer(table))

' Save the Session to a workspace file
Dim w As WorkSpacePersistence = New WorkSpacePersistence
w.Save("c:\workspace\mySeamless.mws")
End Sub
```

注意，添加工作空间（使用 `WorkSpaceLoader` 类）是个累积的过程。要确保 `Session` 仅包含新工作空间文件的内容，请首先在 `ISession` 对象上调用 `Reload()` 方法。该方法清除会话状态并在 `web.config` 文件中重新加载预定义的工作空间。

有关永久性和 XML 构架的详细信息，请参阅[附录 G：理解 MapInfo 工作空间](#)。

Selection 类

`Selection` 是保存图元列表的 `IResultSetFeatureCollection` 对象的集合。这些图元是表中行的子集。这些图元可以是属性边界、街道网络、手机信号发射塔位置或自然图元（例如河流）。显示在 `Map` 上时这些图元通常用特殊的突出显示绘制。对于 `Selection` 中的给定表，只可以有一个 `IResultSetFeatureCollection`。

`ISession` 中可以有多个 `Selection`。`Selections` 集合包含了应用程序中所有的选择集。始终至少有一个选择集，名为 `DefaultSelection`。每个 `Selection` 必须具有一个名称和唯一的别名。默认情况下，地图选择集工具修改 `Selection`（如果使用）。可以分别设置每个工具来使用任何特定的 `Selection`。

`MapXtreme 2005` 中的选择集不是副本；对于 `Selection` 中的给定表它是对 `IResultSetFeatureCollection` 的引用。如果关闭正处理的表后尝试修改 `Selection`，则对 `IResultSetFeatureCollection` 的引用将无效，会导致异常。

图元可以通过使用工具或搜索方法来选择。有关构建 Windows 窗体应用程序时可以使用的不同选择集工具的讨论，请参阅[第 112 页中的 Select 工具](#)。有关选择集 Web 控件和工具的信息，请参阅[第 6 章：运用 MapXtreme 2005 Web 控件](#)。

图元也可以通过源于 `MapInfo.Data.Catalog` 类的搜索方法来选择，其中 `MapInfo.Data.Catalog` 类返回 `IResultSetFeatureCollection` 集合。`Selection` 对象可传入用于填充或更改 `Selection` 的搜索中。

图元也可以通过源于 `Data.MICommand` 类的 `ExecuteFeatureCollection` 方法来选择。这种情况下，会对 `MapInfo` 数据提供方执行 SQL 命令。

有关图元、表、Catalog 和 `MICommand` 的详细信息，请参阅[第 10 章：处理数据](#)。

使用选择集的属性

`Selection` 类的属性用于设置（必需的）名称和别名、将选择集设置为 `Visible` 或 `Editable`，或者用于获得选择集的 `Style`。要确定 `Selection` 是否显示其突出显示以指示其是否被选中或可用于编辑，请分别使用 `Visible` 或 `Editable` 属性。

如果 `Editable` 属性设置为 `true`，则正处理的表也必须是可编辑的。

`Style` 属性指示 `Selection` 的样式并返回 `Selection` 组合样式的副本。因为返回的是副本，所以其不可修改。有关 `Style` 的讨论请参阅[第 15 章：为地图设置样式](#)。

突出显示和导出选择集

选择集通常用特殊的突出显示在地图上绘制，以此将这些图元与周围未选中的图元区分。突出显示的部分由 `Mapping.FeatureViewer.DrawSelections` 属性控制。该属性设置为 `true` 时，如果这些图元是在可见的图层上绘制，则这些图元用突出显示的选择集绘制。

同样，`MapExport.ExportSelection` 属性可用于控制选择集是否绘制到导出图像中。

SelectionChangedEventArgs

委派方法被附加到 `SelectionChangedEventArgs`，以便接收该选择集已更改的通知。例如，如果添加记录则激发 `SelectionChangedEventArgs`。

Selection 和 Selections 类上的 ISerializable 接口

我们实现了 `Selection` 和 `Selections` 类上的 `ISerializable` 接口。以下代码示例说明了如何将 `Selections` 对象串行化或反串行化：

```
// Create a MemoryStream to serialize into  
MemoryStream stream = new MemoryStream();  
  
// Serialize the Selections object  
BinaryFormatter formatter = new BinaryFormatter();  
formatter.Serialize(stream, Session.Current.Selections);
```

```
stream.Position = 0;

// Make changes to the Session's Selections object to make sure the
// deserialization works correctly.
...

// Recreate the Selections object from the stream.
// Note: this replaces the current Session's Selections object with
// the contents of the stream
formatter = new BinaryFormatter();
formatter.Deserialize(stream);
```

Selection 代码示例

以下是常用选择集操作的代码示例。请在 MapXtreme 2004 > Selections 下的 MapInfo Knowledgebase 中查找这个及其它示例。可以通过“程序”菜单上的“MapXtreme 2005 学习资源”浏览器或“支持和培训”下的 MapInfo 网站访问 Knowledgebase。其它代码示例包括在 MapXtreme 2005 开发者参考的多个主题中。

选择另一个图元中的图元

使用 MapXtreme 2005 的常用搜索技术是在另一个图元中查找图元。通过该操作可以查找邮政编码边界内的所有客户或某个地区内正在构建的所有高速公路。请参照以下示例。参数 f 代表 MapInfo.Data.Feature。

VB 示例：

```
Dim si As MapInfo.Data.SearchInfo =
    MapInfo.Data.SearchInfoFactory.SearchWithinFeature(f, _
        MapInfo.Data.ContainsType.Centroid)
Dim irfc As MapInfo.Data.IResultSetFeatureCollection =
    MapInfo.Engine.Session.Current.Catalog.Search("USCty_8k", si)

MapInfo.Engine.Session.Current.Selections.DefaultSelection.Clear()
MapInfo.Engine.Session.Current.Selections.DefaultSelection.Add(irfc)
irfc.Close()
```

检查选择集的表

请参照以下代码示例学习如何获得表中的选择集计数。

VB 示例：

```
Public Shared Sub MapInfo_Engine_Selection2()
    Dim session As ISession = MapInfo.Engine.Session.Current
    Dim tableUsa As Table = session.Catalog("usa")

    ' Get fc for selection on usa.
    Dim fc As IResultSetFeatureCollection =
        session.Selections.DefaultSelection(tableUsa)
    Dim nCount As Integer = 0
    If Not fc Is Nothing Then
        nCount = fc.Count
    End If
End Sub
```

也可以使用 MapInfo SQL 查询和 ADO.NET 数据提供方执行选择集操作。请参阅第 10 章：[处理数据](#)。

从表中返回所有列

以下示例说明了如何从选择集中返回所有列：

VB 示例：

```
Dim Connection As MIConnection = New MIConnection()
Connection.Open()
Dim lyr As FeatureLayer = mapControl1.Map.Layers("usa") as
FeatureLayer
Dim ti As MapInfo.Data.Table =
    MapInfo.Engine.Session.Current.Catalog.GetTable("usa")
Dim si As MapInfo.Data.SearchInfo =
MapInfo.Data.SearchInfoFactory.SearchAll()
si.QueryDefinition.SetColumns("*")
Dim irfc As MapInfo.Data.IResultSetFeatureCollection =
    MapInfo.Engine.Session.Current.Catalog.Search(ti.Alias, si)
Dim l As MapInfo.Data.Feature
For Each l In irfc
    Dim column As MapInfo.Data.Column
    For Each column In l.Columns
        MessageBox.Show(
            String.Format("{0}:{1}",
            column.ToString().ToUpper(), l(column.ToString()).ToString()))
    Next
Next
```

更改选择集后的地图视图

以下示例说明了如何更改缩放以显示选择集中的所有图元。

VB 示例：

```
Me.MapControl1.Map.Bounds =  
    MapInfo.Engine.Session.Current.Selections.DefaultSelection.Envelope.Bo  
unds
```

事件参数

MapInfo.Engine 命名空间包含了各种为事件提供数据的事件参数类。有关详细信息请参阅联机帮助。某些事件参数类包括：

- CollectionCancelableEventArgs - 为可以被取消的集合事件提供数据。
- CollectionEventArgs - 提供集合事件的数据。
- NodeSelectionChangedEventArgs - 节点选择集更改时激发这些事件参数。
- SelectionChangedEventArgs - 其它对象可将委派附加到该事件以在选择集更改时获得通知。

异常

Engine 命名空间包含了各种异常类。有关详细信息请参阅联机帮助。某些异常类包括：

- ResourceNotFoundException - 在 Resource 表中未找到请求的对象时抛出此类异常。
- ResourceTypeMismatchException - 当从 Resource 中读取的对象不是预期类型时抛出异常。
- TimeoutException - 等待池中的 ISession 成为可用时，如果 Current 超时则抛出异常。

10

处理数据

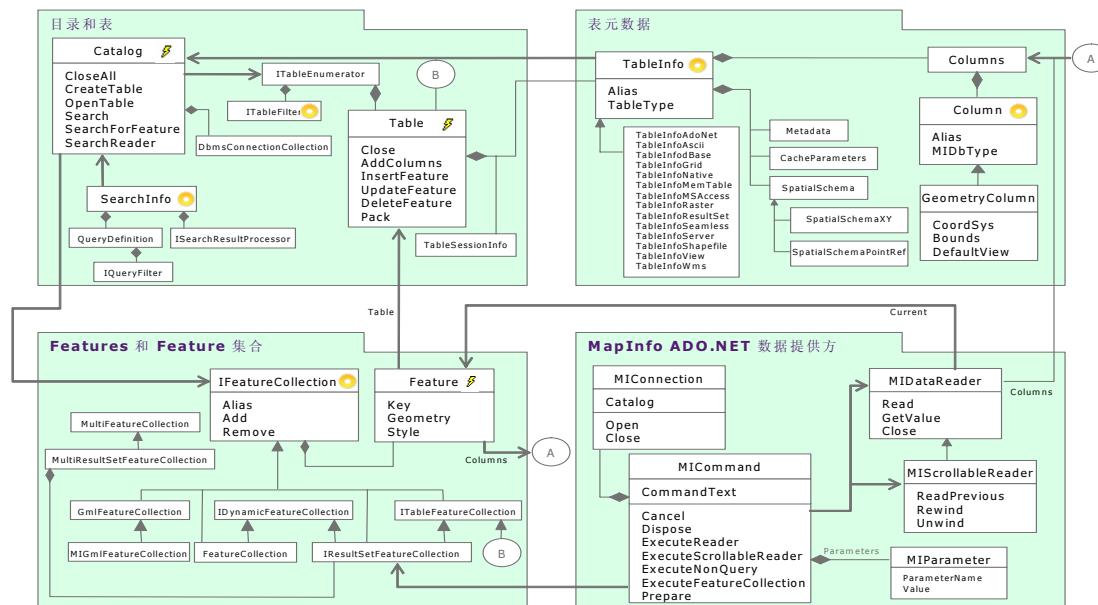
MapInfo.Data 命名空间包含了提供多种方式访问 MapXtreme 2005 应用程序数据的类和接口。

本章内容：

- ◆ *MapInfo.Data* 命名空间概述 148
- ◆ 表 149
- ◆ *Catalog* 151
- ◆ 表元数据 (*TableInfo*) 156
- ◆ 数据源 161
- ◆ 数据绑定 164
- ◆ *MapInfo ADO.NET* 数据提供方 170
- ◆ *MapInfo SQL* 174
- ◆ 表达式 178
- ◆ *Features* 和 *Feature* 集合 183
- ◆ 搜索图元 185

MapInfo.Data 命名空间概述

MapInfo.Data 命名空间包含了提供多种方式访问 MapXtreme 2005 应用程序数据的类和接口。在该命名空间内可以使用具有 MapInfo SQL 语言的 MapInfo ADO.NET 数据提供方，用于数据库和表的标准查询。Feature 对象模型是另一种访问数据的方法，该方法使用对象来代替 SQL。Catalog 是数据访问的起点，包含了以各种方式管理表（打开、关闭、创建）和搜索数据的方法。



MapXtreme 2005 数据访问模型

表

Table 类是所有数据访问的基本单元。Table、Column 和所有 TAB 文件元数据信息可以从 MapInfo Table 访问。表可以是可制图（包含 FeatureGeometry 类型的列）或不可制图的表。表不在地图中显示时也可以打开和访问。

表别名

打开表时，这些表可以分配名称（或别名），为引用该表而打开表时使用该名称（或别名）。例如，在 SQL 语句中表可以引用别名。从 TAB 文件打开的表如果未指定别名，则分配默认的别名。默认的别名基于 TAB 文件的名称。该属性是可选的且可以设置为空。不过，最好分配别名。

列

Column 对象标识了表、图元或图元集合中一列的属性，并标识了列名称（别名）、数据类型、宽度（对于字符串和小数列）以及列的其它属性。

支持的数据类型包括：

数据类型	描述
Int	提供 32 位带符号整数。这里映射到 .NET Framework 数据类型 Int32。
SmallInt	提供 16 位带符号整数。这里映射到 .NET Framework 数据类型 Int16。
Double	从 -1.79E +308 到 1.79E +308 范围内的浮点数。这里映射到 Double。
dBaseDecimal	提供从内部视为与 Double 相同的浮点数。dBaseDecimal 保持在表中时具有固定的精度和比例。从名称上就可以看出，这是从 dBase 文件格式派生的旧式数据类型。这里映射到 Double。
Boolean	提供布尔值。这里映射到 Boolean。
String	提供变长，用空终止的 UNICODE 字符串值。这里映射到 String。
Date	提供日期值。该对象模型目前不支持 Time。尽管类型 MIDbType.Date 的字段返回（或）接受 DateTime 值，但时间会被忽略。这里映射到 .NET Framework 数据类型 DateTime。
FeatureGeometry	提供 FeatureGeometry。请参阅 第 16 章：空间对象和坐标系。
Binary	提供二进制数据的数组。这里映射到 Byte 值的 Array。

数据类型	描述
Key	提供表中的键。这是 Table 上 Key 伪列的数据类型。
CoordSys	提供坐标系。该类型存在的惟一目的是将坐标系对象绑定到函数的 MICommand，该函数需要坐标系的规范。
Style	提供来自表样式列的样式。这是存储在 Table 上样式列中 AllStyle 对象的数据类型。
Raster	提供来自表的光栅列的 RasterInfo。这是存储在 Table 上光栅列中 RasterInfo 对象的数据类型。
Grid	来自表的网格列的 GridInfo。这是存储在 Table 上网格列中 GridInfo 对象的数据类型。
Wms	提供来自表的 Wms 列的 WmsClient。这是存储在 Table 上 Wms 列中 WmsClient 对象的数据类型。

以下章节向您提供了 MI_Key、MI_Geometry 和 MI_Style 列的信息。

MI_Key

所有的表都有一个伪列，名称为 MI_Key，返回 Key 的实例。MI_Key 伪列的概念与 MapInfo Professional 和 MapBasic 中的 rowid 伪列类似。与 rowed 不同，该列不是数字列。Key 实例可以与字符串文字互相转换。

MI_Geometry

表、图元或图元集合中的 Geometry 列对象包含 FeatureGeometry 对象并指定如列的坐标系、其所包含的所有几何体对象的整个边界等属性。

大多数表类型的几何体列都被赋予了名称 "Obj"。为了兼容较早版本的 MapX 和 MapInfo Professional，别名 "Obj" 将解析到表中的第一个 GeometryColumn。此外，别名 "MI_Geometry" 也可以用于任何表，指代 "Obj" 所指的列。

MI_Style

具有 Geometry 列的表也包含具有名称 "MI_Style" 的列，或者如果找不到，则源于具有 MIDbType.Style 类型的第一列。该列用于保存 Geometry 对象的样式信息，如多边形的直线宽度和点的符号大小。该列无法独立更新。Style 和 Geometry 列必须同时更新。

在打开 MapInfo 本机格式 (.TAB) 的表时自动创建 MI_Style 列。对于所有其它表类型，该列必须专门创建。如果使用 MapInfo.Data.ColumnFactory.CreateStyleColumn，则创建具有 "MI_Style" 名称（别名）和 MIDbType.Style 数据类型的列。

将表插入数据库时，必须在插入语句中包括 MI_Style 列。请参阅以下代码示例：

```
Dim tab As Table =  
    MapInfo.Engine.Session.Current.Catalog.GetTable("MapView")  
Dim ti As TableInfo =  
    TableInfoFactory.CreateTemp("Test", (CType(tab.TableInfo.Columns("Obj"))  
        ' -  
        MapInfo.Data.GeomeTryColumn)).CoordSys)  
Dim tabTemp As Table =  
    MapInfo.Engine.Session.Current.Catalog.CreateTable(ti)  
  
Dim conn As MIConnection = New MIConnection()  
conn.Open()  
Dim comm As MICommand = conn.CreateCommand()  
comm.CommandText = "Insert Into " + tabTemp.Alias +  
    " (Obj, MI_Style) SELECT MI_Point(MI_X(Obj), MI_Y(Obj), '" +  
    ((MapInfo.Data.GeomeTryColumn)tab.TableInfo.Columns("Obj")).CoordSys.S  
    rsString + "'), MI_Style" + " FROM " + tab.Alias + " WHERE msaname =  
    'Minneapolis-St.  
    Paul, MN-WI' AND Not Obj = Null"  
MessageBox.Show(comm.CommandText)  
Dim numChanged As Integer = comm.ExecuteNonQuery()  
  
mapControl1.Map.Layers.Add(New FeatureLayer(tabTemp))  
mapControl1.Map.SetView(mapControl1.Map.Layers("Test") as  
FeatureLayer)
```

Catalog

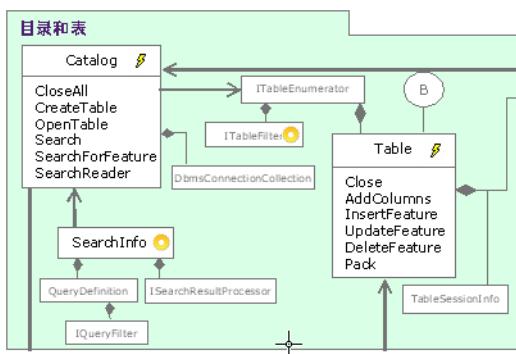
Catalog 实际上是 MapXtreme 2005 数据访问模型的管理器。Catalog 类保存目前在会话中打开的表的列表。表也可从 Catalog 打开、创建和关闭。Catalog 可以视为在其中保存所有打开的表的单一数据库，与其实际数据源无关。

Catalog 在初始化时不包含任何表。表打开时，别名（或名称）被分配到表或由调用方提供表（或名称）。别名用来标识查询和其它操作中的表。

Catalog 提供了工具来创建新建表的定义，以及通过当前打开的表进行枚举。

Catalog 还包含了搜索方法，可用于访问打开的表中的数据。

每个 MapXtreme 2005 Session 只能管理一个 Catalog。



代码示例

以下示例说明了如何访问 Session 对象的 Catalog，打开多个表并通过 Catalog 中所有可编辑的表进行枚举。

VB 示例：

```

Public Shared Sub MapInfo_Data_Catalog()
    ' Catalog is accessible off the Session object
    Dim catalog As Catalog = Session.Current.Catalog

    ' Open several tables
    Dim table As Table = catalog.OpenTable("States.tab")
    table.SessionInfo.ReadOnly = True ' Make states ReadOnly
    table = catalog.OpenTable("world.tab")
    table = catalog.OpenTable("worldcap.tab", "World Capitals")

    ' Enumerate the catalog directly - includes All tables
    Dim t As Table
    For Each t In catalog
        Console.Out.WriteLine("Table : {0}", t.Alias)
    Next
    Console.Out.WriteLine()

    ' Now enumerate through only tables that are editable (not
    ReadOnly)
    Dim tEnum As ITableEnumerator =
    catalog.EnumerateTables(TableFilterFactory.FilterEditableTables())
    While tEnum.MoveNext()
        Console.Out.WriteLine("Table: {0}", tEnum.Current.Alias)
    End While
  
```

```
Session.Current.Catalog.CloseAll()  
End Sub
```

有关 Catalog 的详细信息，另请参阅第 79 页第 5 章中的规划数据模型。

定位打开的表

要定位打开的表，必须枚举目录。该操作可以通过使用以下章节中的方法实现。

Catalog.GetTable

GetTable 方法返回由 TableAlias 参数引用的 Table 对象。这必须是已经打开的表。如果未找到这种表（或表已随后关闭）则方法返回空。

目录索引

Catalog.Item 属性可用作通过其 Alias 定位 Table 的索引。这在功能上与使用 Catalog.GetTable 方法等价，不过该索引生成的代码可读性更好。Alias 必须指定已打开的表。

TableEnumerators

表枚举符可以通过各种重载的 EnumerateTables 方法来获取。表枚举符可以与筛选器一同创建。筛选器确定了枚举中实际包括了哪些表，而枚举符只是简单地提供了枚举的机制。用户可以创建自己的表筛选器用在 TableEnumerator 中。也可以通过实现 ITableEnumerator 接口来创建自己的表枚举符。

代码示例

Catalog 可通过多种方法来枚举。最简单的机制是按以下方法直接枚举目录：

```
Public Shared Sub MapInfo_Data_Catalog2()  
  
    Dim tbl As Table  
    For Each tbl In Session.Current.Catalog  
        System.Console.WriteLine("Table: " + tbl.Alias)  
    Next  
  
End Sub
```

另一个选项是按以下方法使用表枚举符 (TableEnumerator):

```
Public Shared Sub MapInfo_Data_Catalog3(ByVal catalog As Catalog)
    Dim te As ITableEnumerator =
        catalog.EnumerateTables(TableFilterFactory.FilterEditableTables())

    While te.MoveNext()
        Dim tbl As Table = te.Current

    End While
End Sub
```

关闭表

有两种方法可以关闭表。Catalog.CloseAll 关闭所有打开的表，而 Catalog.CloseTable 关闭某一个打开的表。

打报表中的文件

Table.Pack 方法打报表中的数据。打报表时，激发表的 TablePacked 事件。事件的参数指示表的键是否因打包而更改（删除已删除的记录时发生）。如果 PackType 包括 RemoveDeletedRecords 且如果表的中间实际有已删除的记录，则仅更改键。如果表中仅删除的记录不在表的结尾，则不更改任何键且事件不指示键已更改。

PackType 枚举提供了以下选项。

- PackGeometry - 指示几何体对象被打包。打包对象操作会尝试删除尽可能多的未使用空间。完全打包的 RTree（用于从空间访问几何体对象的空间索引）由于引起许多不必要的读取会降低性能。要在磁盘空间和处理速度之间取得平衡，打包几何体对象可以继续在 RTree 中留下一些未使用的空间。此外还要注意：打包的 RTree 会导致在插入和更新操作时出现轻微的性能降低，因为 RTree 更有可能需要扩展。
- RebuildGeometry - 重建几何体对象删除了从一系列插入、更新和（或）删除操作生成的未使用空间。与打包几何体对象不同，该选项有意在 RTree 索引中留下未使用的空间以改善以后插入和更新操作的性能。
- PackIndex - 非空间的索引作为 B*trees 维护。这些结构不总是具有填存的内部或分支节点。默认情况下，有意允许在索引的位置放置插入和更新操作，而无需索引的大量调整。由于插入、更新或删除操作的出现，未使用的空间会剧增。打包索引操作完全打包了每个内部和分支节点（除可能的“最后”节点）。该选项尽可能减少了索引所用的磁盘空间，还改善了索引的读取性能。在完全打包的索引上插入和更新操作的性能会降低。

- RebuildIndex - 重建索引不会像 PackIndex 选项一样完全打包内部和分支节点。相反，重建索引创建具有未使用空间的索引，该未使用空间有意放置在索引中用于在磁盘空间、读取性能和修改性能之间取得平衡。多次修改操作后，索引会包含大量的未使用空间。该选项重新获得未使用的空间。
- RemoveDeletedRecords - 包括 MapInfo Native 和 dBase 数据源在内的某些数据源当它们被删除时并不物理删除记录。要物理删除已删除的记录，必须指定此选项打包表。记录号通常用作这些数据源类型的记录键。从表删除已删除的记录会使得键成为无效，因为这些键会因打包而更改。
- CompactDb - 如果表的数据源是 Microsoft Access (Access 的 TableType)，则通过使用 Pack 方法并指定该选项，也可以压缩包含表数据的 MDB 文件。
- All - 这是与 PackGeometry | PackIndex | RemoveDeletedRecords 等价的简洁选项。

倾听表和目录事件

Table 公开了应用程序可以订阅的多个事件。它们是：

- RowInsertedEvent - 将新行添加到表时发生。
- RowUpdatedEvent - 更新表中现有的行时发生。
- RowDeletedEvent - 删除表中的行时发生。
- TablePackedEvent - 打包表时发生。
- TableCloseRequestEvent - 已经要求关闭表时发生。
- TableIsClosingEvent - 关闭表时发生。
- TableClosedEvent - 关闭表时发生。

Catalog 也公开以下事件。

- TableOpenedEvent - 打开表时发生。
- TableCreatedEvent - 创建新表时发生。
- TableIsClosingEvent - 关闭表时发生。

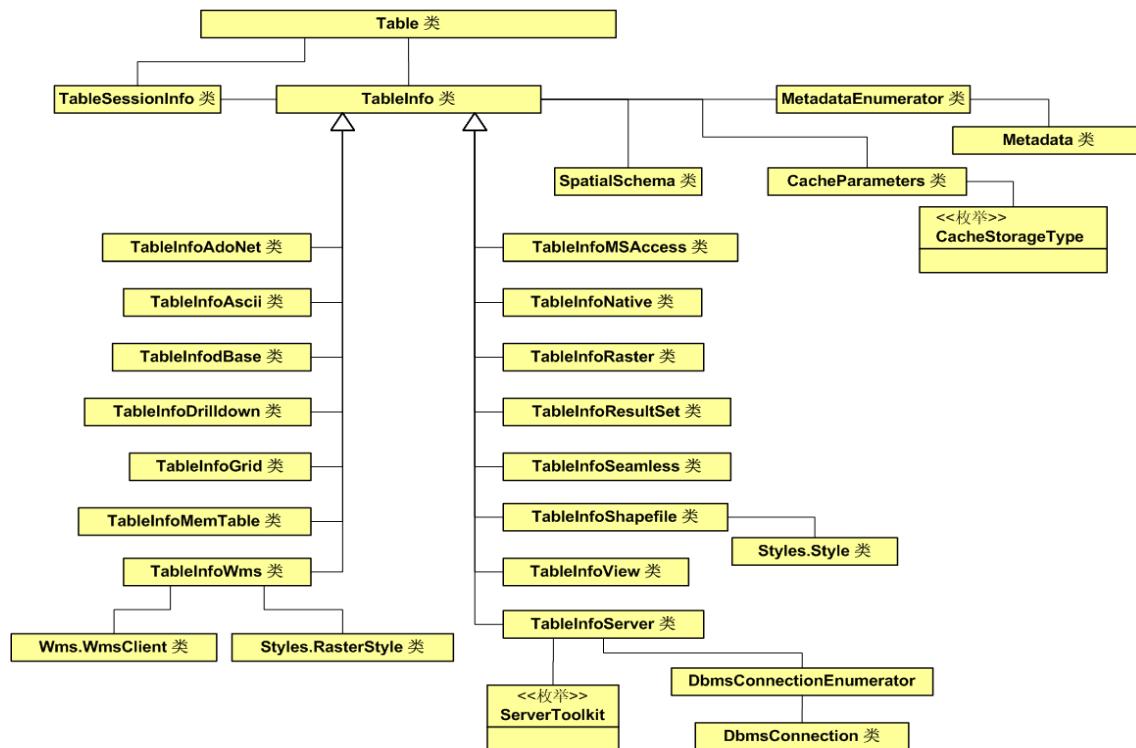
表元数据 (TableInfo)

Data 命名空间中的 TableInfo 类包含与现有表有关的信息或元数据。每个 MapInfo 表具有与表有关的元数据或信息。该信息包括：

- 列 - 编号、名称、数据类型等。
- 数据源的说明和路径名。
- 客户端元数据（TAB 文件中 begin_metadata/end_metadata 标记之间的信息）。

TableInfo 也用于定义新表或打开现有的表。例如，数据源的 .TAB 不可用时，可以使用 TableInfo 对象通过 Catalog 打开。

下图显示了 TableInfo 类及其相关类的结构。



TableInfo 类也可用于从 RDBMS 访问数据。有关详细信息，请参阅[第 11 章：从 DBMS 访问数据](#)。

检查 TAB 文件元数据

TAB 文件元数据可以访问和编辑。可从 Table 获得 TableInfo 类来获得与表结构有关的信息。以下代码说明了如何获得打开的表的元数据。该代码还说明了几何体列如何用于确定表的坐标系和边界。

VB 示例：

```
Public Shared Sub MapInfo_Data_TableInfo2()
    ' Get the metadata for an open table
    Dim ti As TableInfo = Session.Current.Catalog("states").TableInfo

    ' Print out some information to the console
    Console.Out.WriteLine("Table Alias={0}, Datasource={1}, _"
        Description={2}, Type={3}", _
        ti.Alias, ti.DataSourceName, ti.Description, ti.TableType)

    ' Print out some information about each column
    Dim col As Column
    For Each col In ti.Columns
        Console.Out.WriteLine("Column {0} Type={1} Width={2}", _
            col.Alias, col.DataType, col.Width)
        ' If the column is a geometry column, print csy and bounds.
        If col.DataType = MIDbType.FeatureGeometry Then
            Dim geocol As GeometryColumn = col
            Dim csys As MapInfo.GeomeTry.CoordSys = geocol.CoordSys
            Console.Out.WriteLine("CSys : {0}", csys.MapBasicString)
            Dim dr As MapInfo.GeomeTry.DRect = geocol.Bounds
            Console.Out.WriteLine("Bounds=({0},{1}),({2},{3})", dr.x1, _
                dr.y1, dr.x2, dr.y2)
        End If
    Next
End Sub
```

创建新表

以下章节说明了如何创建永久本地表、临时本地表和临时 MemTable。

创建新的永久本地表

MapInfo 本地表的 Table.TableInfo 属性返回 TableInfoNative 的实例。本地表为 MapInfo .TAB 文件。该类可用于访问特定于本地表类型的属性。可以创建该类的新实例，并使用该实例构造新表。

请注意 ColumnFactory 类的使用。提供该类的目的是帮助您了解对于不同的数据类型哪些参数是必须的。例如，几何体列需要坐标系。

VB 示例：

```
Public Shared Sub MapInfo_Data_TableInfoNative()
    Dim ti As TableInfoNative = New TableInfoNative("NewTable")
    ti.TablePath = "c:\data\Capitals.TAB"
    ti.Columns.Add(ColumnFactory.CreateIndexedStringColumn("Capital", _
        25))
    ti.Columns.Add(ColumnFactory.CreateStringColumn("Country", 30))
    ti.Columns.Add(ColumnFactory.CreateDoubleColumn("Pop_Grw_Rt"))

    ' Make the table mappable
    ti.Columns.Add(ColumnFactory.CreateStyleColumn())
    Dim Robinson As CoordSys = _
        Session.Current.CoordSysFactory.CreateFromPrjString("12, _ 
        62, 7, 0")

    ti.Columns.Add(ColumnFactory.CreateFeatureGeometryColumn(Robinson))
    ' Note we do not need to (nor should we) add a column of type Key.
    ' Every table automatically contains a column named "MI_Key".
    Dim table As Table = Session.Current.Catalog.CreateTable(ti)
End Sub
```

创建临时本地表

VB 示例：

```
Public Shared Sub MapInfo_Data_TableInfo3(ByVal conn As _
    MIConnection)
    Dim ti As TableInfoNative = New TableInfoNative("NewTable")
    ti.Temporary = True
    Dim col As Column

    col = New Column
    col.Alias = "FString30"
    col.DataType = MIDbType.String
    col.Indexed = True
    col.Width = 30
    ti.Columns.Add(col)

    col = New Column
    col.Alias = "FInt32"
    col.DataType = MIDbType.Int
    col.Indexed = True
    ti.Columns.Add(col)

    col = New Column
    col.Alias = "FInt16"
    col.DataType = MIDbType.SmallInt
    col.Indexed = True
```

```
ti.Columns.Add(col)

col = New Column
col.Alias = "FDouble"
col.DataType = MIDbType.Double
ti.Columns.Add(col)

col = New Column
col.Alias = "FDateTime"
col.DataType = MIDbType.Date
ti.Columns.Add(col)

col = New Column
col.Alias = "FBoolean"
col.DataType = MIDbType.Boolean
ti.Columns.Add(col)

' Note we do not need to (nor should we) add a column of type Key.
' Every table automatically contains a column named "MI_Key".
Dim miTable As Table = conn.Catalog.CreateTable(ti)
End Sub
```

创建临时 MemTable

内存表的 Table.TableInfo 属性返回 TableInfoMemTable 的实例。该类可用于访问特定于内存表类型的属性。可以创建该类的新实例，并使用该实例构造新表。

可以通过这种方式将窗体（如 XML 或 GML）中的数据加载到 Catalog。有关将数据加载到 MapXtreme 2005 的详细信息，请参阅[第 83 页第 5 章中的访问数据的方法](#)。

VB 示例：

```
Public Shared Sub MapInfo_Data_TableInfoMemTable()
    Dim ti As TableInfoMemTable = New TableInfoMemTable("NewTable")
    ' Note: The TablePath property does not apply - it can be set but it
    ' is meaningless.

    ti.Columns.Add(ColumnFactory.CreateIndexedStringColumn("Capital", _
        25))
    ti.Columns.Add(ColumnFactory.CreateStringColumn("Country", 30))
    ti.Columns.Add(ColumnFactory.CreateDoubleColumn("Pop_Grw_Rt"))

    ' Make the table mappable
    ti.Columns.Add(ColumnFactory.CreateStyleColumn())
    Dim Robinson As CoordSys =
        Session.Current.CoordSysFactory.CreateFromPrjString("12, 62, _
        7, 0")
    ti.Columns.Add(ColumnFactory.CreateFeatureGeometryColumn_
        (Robinson))
```

```
' Note we do not need to (nor should we) add a column of type Key.  
' Every table automatically contains a column named "MI_Key".  
    Dim table As Table = Session.Current.Catalog.CreateTable(ti)  
End Sub
```

将表达式列增加到表

使用 MapInfo.Data.Table.AddColumn 方法向表中添加表达方式列。这种形式的 AddColumns 根据由函数、运算符、重复值和表上其它列组成的表达式，使用 Columns 对象创建临时列。columns 参数中 Column 的所有实例必须具有指定的表达式字符串。

注： Table.AddColumn 不支持以下表类型： Server、View、Seamless、AdoNet、ResultSet 或 Drilldown。MapXtreme 2005 检查表并在遇到上述表类型时抛出异常。

VB 示例：

```
Public Shared Sub MapInfo_Data_TableAddColumns(ByVal miTable As Table)  
    Dim NewCols As Columns = New Columns  
    NewCols.Add(New Column("PopDensity1990", "Pop_1990 / _  
        MI_Area(Obj, 'sq mi', 'Spherical')"))  
    NewCols.Add(New Column("PopDensity2000", "Pop_2000 / _  
        MI_Area(Obj, 'sq mi', 'Spherical')"))  
    miTable.AddColumn(NewCols)  
End Sub
```

表达式字符串“Pop_1990 / MI_Area(Obj, 'sq mi', 'Spherical')”表示放置在临时列中的派生信息。该段代码表示“每个人口记录除以平方英里为单位的面积得出人口密度”。表达式中的 SQL 函数 MI_Area() 将从记录的几何体派生面积。

有关创建表达式的详细信息请参阅第 178 页中的表达式。

数据源

下表列出了 MapXtreme 2005 支持的数据源。

类	TableType	描述
TableInfoNative	Native	MapInfo 本地表。表可以具有关联的 MAP 文件，文件内包含几何体和样式信息。
TableInfodBBase	dBase	数据存储在 dBase 文件中。表可以具有关联的 MAP 文件，文件内包含几何体和样式信息。
TableInfoMSAccess	Access	Microsoft Access 数据库表。
TableInfoAscii	Ascii	数据存储在 ASCII 文本文件中。最大字符串长度为 255 个字符（最多包括两个引号）。
TableInfoServer	Server	基于 RDBMS 服务器连接和查询字符串的表（可能使用 SpatialWare 或 Oracle Spatial）。SpatialWare 4.8 for SQL Server、SpatialWare 4.5.1 for Informix、Oracle Spatial 817、9iR1、9iR2、9iR2Locator、XY 和 MICode。
TableInfoShapefile	Shapefile	ESRI Shapefile 表。
TableInfoMemTable	MemTable	临时表，数据存储在内存中，几何体存储在临时 MAP 文件中。
TableInfoView	View	基于 SQL Select 语句（MapBasic 或 MapInfo SQL - 这与 Server 表中使用的服务器端 SQL 不相同）的视图。出现在选择列表中的表达式形成了表的列和记录，where 子句组成表成员的条件。
TableInfoSeamless	Seamless	无缝表。
TableInfoRaster	Raster	包含光栅图像的表。
TableInfoGrid	Grid	包含网格图像的表。
TableInfoWMS	WMS	包含源于 WMS 服务器的图像的表。
TableInfoAdoNet	ADONET	该类型的表由导入到 DataAccess 引擎并公开为 Table 的客户端数据组成。客户端数据可以使用多种形式的 ADO.NET 对象提供。
TableInfoResultSet	ResultSet	目前 ResultSet 表专用于 IResultSetFeatureCollections。

有关“数据源”的详细信息，请参阅[第 81 页第 5 章中的选取正确的数据源](#)。

使表可制图

数据的表可制图（包含 GeometryColumn）或不可制图（无空间属性数据）。可制图表可作为地图中的图层添加到 MapXtreme 2005 应用程序。在为不可制图表（如客户数据）创建 GeometryColumn 时，不可制图表也可制图。MapXtreme 2005 提供了用于完成这项功能的空间构架。

空间构架是服务，可以应用到表来增强表的空间能力。有如下两种类型的空间构架：XY 和 PointRef。具有表示为 X 和 Y 值（如经度和纬度）的属性列的不可制图表使用 SpatialSchemaXY，具有可用于引用可制图表中记录的属性列的表则使用 SpatialSchemaPointRef。

XY 空间构架

XY 空间构架由表使用，根据每个记录的 X 和 Y 数据值构造 Point 几何体。XY 空间构架可应用到除 Seamless、Views 和 ResultSet 之外任意数据源的表。

XY 空间构架可以应用于不可制图的表，使其可制图。方法是指定包含 X 和 Y 坐标值的列以及这些值所表示的坐标系。表打开时，它包含 GeometryColumn 并可以作为图层添加到 Map 中。几何体列 (MI_Geometry) 可以修改，它可以修改表中的 X 和 Y 值。这样就可以使用 Map 中的工具来移动几何体。X 和 Y 列被标记为只读。

VB 示例：

```
Public Shared Sub MapInfo_Data_SpatialSchemaXY()
    Dim ti As TableInfo =
        TableInfo.CreateFromFile("c:\data\customers.TAB")
        ' a non-mappable table
    Dim xy As SpatialSchemaXY = New SpatialSchemaXY
    xy.XColumn = "Xcoord"
    xy.YColumn = "Ycoord"
    xy.NullPoint = "0.0, 0.0"
        ' Any customer at 0,0 means we don't know their location.
    xy.StyleType = StyleType.None
    xy.CoordSys =
        Session.Current.CoordSysFactory.CreateLongLat(DatumID.WGS84)
    ti.SpatialSchema = xy
        ' Now set the spatial schema information before
        ' opening the table.
    Dim table As Table = Session.Current.Catalog.OpenTable(ti)
End Sub
```

该表应该具有 FeatureGeometry 列和 Style 列 (MI_Geometry 和 MI_Style)。原始的 TAB 文件还未修改，如果重新打开该文件（或在另一个会话中打开）则不可制图。使用 TableInfo.WriteTabFile 重写带有新元数据属性的 TAB 文件。如果 TAB 文件被重写，则每次打开表时会自动重新生成并使用空间构架。

```
Public Shared Sub MapInfo_Data_TableInfoNative2(ByVal ti As  
TableInfoNative)  
    ti.WriteTabFile()  
End Sub
```

PointRef 空间构架

该构架使用表数据中的查找值，通过将查找值匹配到可制图的表来创建 Point 几何体对象。

PointRef 空间构架可以应用于不可绘制地图的表，使其可绘制地图。方法是使用表中某列 (MatchColumn) 的值来查找另一个表（引用表）中的 Geometry。

如果表不可制图，但具有可以与可制图表的值关联的数据值，则该表可以使用 Point Reference 空间构架访问，如以下示例所示。

VB 示例：

```
Public Shared Sub MapInfo_Data_SpatialSchemaPointRef(ByVal _  
map As Map)  
  
    ' a non-mappable table  
    Dim ti As TableInfo =  
        TableInfo.CreateFromFile("c:\data\customers.TAB")  
    Dim pr As SpatialSchemaPointRef = New SpatialSchemaPointRef  
        pr.CoordSys = map.GetDisplayCoordSys()  
        pr.StyleType = StyleType.None  
        pr.RefTable = "us_zips"  
  
    ' the column in RefTable which will match the MatchColumn in my data  
    pr.RefColumn = "zipcode"  
  
    ' a column in the Customer table  
    pr.MatchColumn = "zip"  
    pr.RefTableLocation = "c:\data\us_zips.tab"  
  
    ' Now set the spatial schema information before opening the table.  
    ti.SpatialSchema = pr  
    Dim table As Table = Session.Current.Catalog.OpenTable(ti)  
End Sub
```

该表应该具有 FeatureGeometry 列和 Style 列 (MI_Geometry 和 MI_Style)。原始的 TAB 文件还未修改，如果重新打开该文件（或在另一个会话中打开）则不可制图。使用 TableInfo.WriteTabFile 重写带有新元数据属性的 TAB 文件。如果 TAB 文件被重写，则每次打开表时会自动重新生成并使用空间构架。

```
Public Shared Sub MapInfo_Data_TableInfoNative2(ByVal ti As  
TableInfoNative)  
    ti.WriteTabFile()  
End Sub
```

有关如何关联可制图和不可制图表的详细信息，请参阅第 85 页第 5 章中的相关表。

数据绑定

数据绑定是将数据从数据源带到 MapXtreme 2005 中的过程。通过使用 TableInfoAdoNet 作为 Table 打开 ADO.NET DataTable，实现将外部的（ADO.NET 和其它旧的数据源）数据绑定到 MapInfo.Data.Tables。表然后可与另一个表连接、与自身连接或使用 Table.AddColumn 将列绑定到第二个表。

要将表与其自身连接，请参照以下示例：

```
Select ... From T as A, T as B Where A.X = B.Y
```

如果应用程序具有存储在 DataTable 中的数据，或具有可以通过 ADO.NET 数据提供方访问的数据，则这些数据可以提交到 Catalog 并视为 MapInfo 表。如果数据不能通过其它表类型之一来访问，就需要使用这种功能。

例如，如果数据存储在 dBase 文件、Microsoft Access 表，或可以通过 ODBC 或 Oracle 的 OCI 接口访问，则推荐使用这些表类型来访问数据。如果数据不能通过其中一个表类型来访问，但可以加载到 DataTable 中或通过某些 ADO.NET Data Provider（实现 Command、Parameter 和 DataReader 对象类型）可以访问，则这些数据仍然可以通过 Catalog 来访问。

应用程序可能需要数据可以像 MapInfo 表一样使用，以便可以执行查询将数据与其它 MapInfo 表数据连接。对于 Catalog 可能也需如此，以便在对 Table.AddColumn 的调用中可以将 Catalog 用作源数据。有关数据访问和相关表的详细信息，请参阅[第 79 页第 5 章中的规划数据模型](#)。

如何从 Oracle 表连接数据

VB 示例：

```
public Shared Sub MapInfo_Data_TableAddColumns5(ByVal map As Map)
    Dim Connection As MapInfo.Data.MIConnection = New _
        MapInfo.Data.MIConnection
    Connection.Open()

    'Add the USA table to the map
    map.Load(New _
        MapInfo.Mapping.MapTableLoader("C:\\maps\\usa.TAB"))

    Dim lyr As MapInfo.Mapping.FeatureLayer = map.Layers("usa")

    ' Open the table from Oracle

    Dim ti As TableInfoServer = New TableInfoServer("StateCapXY",_
        "SRVR=tempest;UID=tn;PWD=tn", "Select * from usa_caps", _
        MapInfo.Data.ServerToolkit.Oci)
    Dim StateCapXY As Table = Connection.Catalog.OpenTable(ti)

    ' Add the Oracle columns to the USA table
    Dim states As Table = Connection.Catalog.GetTable("usa")
    states.AddColumn(Nothing, MapInfo.Data.BindType.Static, _
        StateCapXY, "state", MapInfo.Data.Operator.Equal, "state")

    'Create a ranged theme on the USA layer using a field
    'from the Oracle table
    Dim thm As MapInfo.Mapping.Thematics.RangedTheme = New _
        MapInfo.Mapping.Thematics.RangedTheme(lyr, _
        "pop_1990", "popusa", 4, _
        MapInfo.Mapping.Thematics.DistributionMethod.EqualCount_
        PerRange)
    lyr.Modifiers.Insert(0, thm)

    'Create a legend to appear on the map
    Dim legend As MapInfo.Mapping.Legends.Legend = _
        map.Legends.CreateLegend(New Size(5, 5))
    legend.Border = True
    Dim frame As MapInfo.Mapping.Legends.ThemeLegendFrame = _
        MapInfo.Mapping.Legends.LegendFrameFactory.CreateTheme_
        LegendFrame("Pop", "pop", thm)
    legend.Frames.Append(frame)
    frame.Title = "pop"
    map.Adornments.Append(legend)
End Sub
```

要获得该示例的 C# 版本或查看电子版示例，请访问位于“程序”菜单的“MapXtreme 2005 学习资源”浏览器。通过浏览器主页，查看“学习资源 > 文档链接”。单击第一个主题“代码片断如何实现”，打开代码示例的帮助文件。该示例位于“数据访问”下。

使用 ADO.NET 数据提供方

具有无法直接访问的特定 TableInfo 数据源的数据可以使用 TableInfoAdoNet。ADO.NET 表可以是以下两种形式之一：DataTable（内存数据，允许读写访问）；或 IDbCommand（动态数据，只读）。

访问 DataTable 中的数据

使用 DataTable 时，Catalog 实际保存到对 DataTable 的引用上，其中 DataTable 提供到对 Catalog.OpenTable 的调用（使用 TableInfoAdoNet 类）。通过发布 Insert、Update 和（或）Delete 命令，使用 MapInfo ADO.NET Data Provider 可以编辑 DataTables。应用程序也可以继续直接访问 DataTable。不过要注意，Catalog 具有到表的引用时表的结构不应该更改。还需注意，MapInfo Data Provider 之外的其它数据发生更改时（例如不使用 MICommand 发布 Insert、Update 或 Delete 命令时）不会激发插入、更新或删除表的事件。

DataTable 几乎包含了 Catalog 定义表的全部信息。不过对于字符串列，Catalog 需要将一个长度分配到该字段。构造临时索引、用于合计的临时表等时将使用该长度。对于这种类型的操作，一定要保证获得的字符串长度正确。 DataColumn 具有 MaxLength 属性，该属性设置用于指示列可以保存的最大字符串长度。如果不设置，则该值默认为 -1，这种情况下使用的长度值为 254。检查 MaxLength 属性之前，目录查看 DataColumn 是否具有在其 ExtendedProperties 集合中定义的属性，名称为 "StringWidth"。如果找到该属性，则将该属性的值用作列的宽度。

以下示例说明了如何创建 MapInfo Table，该表的数据存储在 DataTable 中。

VB 示例：

```
Public Shared Sub MapInfo_Data_TableInfoAdoNet(ByVal connection As _
MIConnection)
    ' Create a new DataTable.
    Dim dt As DataTable = New DataTable("CityData")
    Dim dc As DataColumn
    dc = dt.Columns.Add("City", Type.GetType("string"))
    dc.MaxLength = 30
    dc = dt.Columns.Add("Country", Type.GetType("string"))
    dc.MaxLength = 30
    dc = dt.Columns.Add("Continent", Type.GetType("string"))
    dc.MaxLength = 30
    dc = dt.Columns.Add("Population", Type.GetType("string"))
```

```
' Populate the DataTable...
dt.Rows.Add(New Object() {"Jim", "0001", -73.77, 42.67, 314.15})
dt.Rows.Add(New Object() {"Tom", "0002", -73.87, 42.82, 628.3})

' Now open a MapInfo Table which accesses this DataTable
Dim ti As TableInfoAdoNet = New TableInfoAdoNet("Cities")
ti.ReadOnly = False
ti.DataTable = dt
Dim table As Table = connection.Catalog.OpenTable(ti)
End Sub
```

使用 IDbCommand 访问数据

第二种形式的 ADO.NET 表基于 ADO.NET 中连接的对象类型：Connection、Command 和 DataReader。以这种方式构造的 MapInfo Tables 是只读的。通过传递到已配置的 Catalog 和 IDbCommand 对象，创建这些类型的表来返回构成表的所有数据。如果初始化时创建表（通过调用 Catalog.OpenTable），则在 IDbCommand 上调用 ExecuteReader。生成的数据读取器用于确定列及其数据类型。所有后续光标请求（不是检索特定记录的光标 - 名为键获取）也调用 ExecuteReader 来获取满足光标的 data。注意，这种操作效率很低。实际情况中最好使用其它的表类型之一来访问数据。

因为基于命令形式的 ADO.NET 表设计用于一般接口，无需任何与这些接口的特殊实现有关的特定信息，所以该表也不假定 IDbCommand.CommandText 是任何形式的标准 SQL。实际上，它根本不可以为 SQL。该表类型不访问、分析或修改 CommandText。这意味着该表类型不具有相应的机制来了解结果中的哪些列设计惟一、非空的键值。对于这种类型的表，需要告知表哪些列构成键。这可以通过将 KeyType 指定为 Explicit 并设置 KeyColumns 属性来实现。

MapInfo Data Provider 内有很多操作需要通过键（也称为键获取）检索特定的记录。Select 语句带有格式为 MI_Key = '5' 的 where 子句，这是一个需要检索记录的简单示例，该记录的 MI_Key 列可以通过字符串文字 '5' 表示。映射选择集、标注和在 MIScrollableReader 中滚动（这种情况下读取器可以通过键值的列表滚动）时常用键检索。MapInfo 表取决于通过键值有效获取记录的能力。就是因为基于命令形式的 ADO.NET 表不读取、分析或修改定义表（“Sequential” IDbCommand）的 IDbCommand 对象的 CommandText，所以它不能修改 IDbCommand 对象来获取特定的记录。因此，要达到该目的必须提供第二个 IDbCommand 对象。“FetchByKey” IDbCommand 对象必须满足以下要求：

- 在该命令对象上调用 ExecuteReader 时，生成的数据读取器必须具有与连续命令对象相同的列，且顺序也相同。

- FetchByKeyCommand 必须包含 Parameters 集合，且为键的每个成员包含一个参数。例如，如果 TableInfo.KeyColumns 指定了由 "city" 和 "state" 列组成的键，则 FetchByKeyCommand 必须包含两个参数对象。为第一个参数对象分配的值表示在 TableInfo.KeyColumns 集合中指定的第一列（例如，值 "city"），为第二个参数对象分配的值表示在 TableInfo.KeyColumns 集合中指定的第二列（例如，值 "state"）等等。在 FetchByKeyCommand 上调用 ExecuteReader 时，读取器必须返回表示指定键的记录。

以下示例说明了如何创建 MapInfo Table，该表通过连接命令对象的 ADO.NET 访问数据。

VB 示例：

```
Public Shared Sub MapInfo_Data_TableInfoAdoNet2(ByVal connection _  
    As MIConnection)  
    Dim ti As TableInfoAdoNet = New TableInfoAdoNet("EuropeanCities")  
    Dim _conn As OleDbConnection = New _  
        OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0;Data _  
            Source=C:\Data\EuropeCities.mdb")  
    Dim selectQuery As String = "SELECT City, Country, Continent, _  
        Pop_1994 FROM EuropeCities"  
    Dim _OleDbCommand As OleDbCommand = New OleDbCommand(selectQuery)  
    _OleDbCommand.Connection = _conn  
  
    selectQuery = selectQuery + " where City = @City AND _  
        Country = @Country"  
    Dim _OleDbKeyCommand As OleDbCommand = New _  
        OleDbCommand(selectQuery)  
    _OleDbKeyCommand.Parameters.Add("@City", OleDbType.Char)  
    _OleDbKeyCommand.Parameters.Add("@Country", OleDbType.Char)  
    _OleDbKeyCommand.Connection = _conn  
  
    ' The MapInfo Table will Open/Close the connection as necessary.  
    ' If this is expensive the application could open the connection  
    ' before opening the table and closing the connection after the  
    ' table is closed.  
    ti.SequentialCommand = _OleDbCommand  
    ti.FetchByKeyCommand = _OleDbKeyCommand  
  
    ' Tell the table which column(s) constitute a key - for this table  
    ' it is a compound key consisting of values from the City and County  
    ' columns.  
    Dim sc As StringCollection = New StringCollection  
    sc.Add("City")  
    sc.Add("Country")  
    ti.KeyColumns = sc  
    ti.KeyType = KeyType.Explicit  
  
    ' Ask the Catalog to open the table.
```

```
Dim tbl As Table = connection.Catalog.OpenTable(ti)

' Now the MICommand object may be used to select data from the table
' (by the name EuropeanCities since that is the alias we assigned to
' it). The data in thistable may be joined with any other table and
' it may be used as source data in a call to AddColumns to populate
' temporary columns with data from this table.
End Sub
```

源行

源行表示 AddColumns 所涉及的表中记录之间的匹配。SourceRows 类是 SourceRows 的集合，标识了来自数据源的记录，对添加到表中的目的地临时列的每个记录进行合计。

可以用 AddColumns 方法来向 Table 中添加临时列。从 bind 表中的关联记录到另一个数据源的记录的临时列的值可以自动计算。SourceRows 包含了源数据和绑定数据（及优化表，若使用）之间记录的映射。

注： Table.AddColumns 不支持以下表类型：Server、View、Seamless、AdoNet、ResultSet 或 Drilldown。

GeoDictionary

GeoDictionary 维护可以将地图实体匹配到某些信息的信息。GeoDictionaries 类是 GeoDictionary 对象的集合。MapInfo.Data.GeoDictionary 命名空间通过成为 GeoDictionary 文件的可编程表示为数据自动绑定提供支持。GeoDictionary 文件包含与表有关的信息（仅 TAB 文件）。GeoDictionary 用于自动确定应该将应用程序绑定到的表。GeoDictionary 保留在文件中（通常为 GeoDict.DCT）并使用 GeoDictionary 管理器实用程序维护（请参阅 第 20 章：使用 *GeoDictionary 管理器*）。

使用 GeoDictionary 的 AutoMatching

Data.GeoDictionary 命名空间中的 MatchResolver.AutoMatch 方法初始化 AutoMatching 进程。它不调用 AddColumns，即不执行绑定。执行自动绑定需要对 BindColumn 的后续调用或对 AutoMatchAndBind 的直接调用。

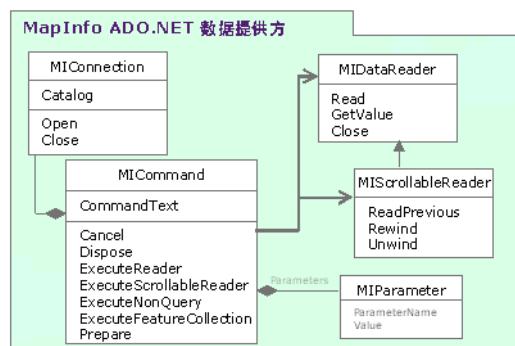
Automatching 可能会遇到模棱两可的情况。这些情况包括：

- 在用户数据中检测到多个源列
- 检测到匹配源列的多个表 / 图层
- 匹配的表 / 图层有多个 geosets/ 工作空间

在解决模棱两可的匹配过程中 GeoDictionary 是与 MatchResolver 对象通信。它提供了匹配算法。基类选择第一个或匹配百分比最高的项目。该类未封闭，客户端应用程序可以从中派生出其自己的类并重载其行为。

MapInfo ADO.NET 数据提供方

MapXtreme 2005 提供了发布 SQL 命令的机制，该命令使用 ADO.NET 从表返回记录集。MapInfo ADO.NET 数据提供方是访问 .NET 应用程序中数据的首选机制。关于另一机制，请参阅第 183 页中的 *Features* 和 *Feature* 集合。



以下章节介绍了通过 MapInfo ADO.NET 数据提供方访问数据的主要接口和类。

- **MIConnection**
- **MICmd**
- **MIDataReader**
- **MapInfo SQL 语言**

MIConnection

MIConnection 表示到 Catalog 的连接。连接提供了发布命令和获取结果的起点。虽然大多数数据提供方连接都允许用户对已有的表（或构架对象）立即开始发布查询或其它命令，但 MapInfo Data Provider 在开始时并没有表。表要在打开或创建之后才能访问。打开后，可以将名称（别名）与表关联，该名称在查询引擎中解析标识符时使用。

在 MapInfo Data Provider 中，连接是不进入连接池的，创建新的连接也不需要连接字符串。

Session 类创建并初始化 Catalog，它可以通过 Session.Current.Catalog 属性来访问。
MICnection.Open 方法用 Session.Current.Catalog 属性来获取对 Catalog 的引用，
MICnection.Close 方法将对 Catalog 的内部引用设置为空。

VB 示例：

```
Public Shared Sub MapInfo_Data_MICnection()
    Dim connection As MICnection = New MICnection
    Dim command As MICommand = connection.CreateCommand()
    command.CommandText = "Select * From States Where Pop > 1000000"

    connection.Open()
    Dim reader As MIDataReader = command.ExecuteReader()
    Dim i As Integer, n As Integer = reader.FieldCount
    For i = 0 To n - 1 Step i + 1
        Console.Out.WriteLine("{0}\t", reader.GetName(i))
    Next
    Console.Out.WriteLine()
    While reader.Read()
        For i = 0 To n - 1 Step i + 1
            Dim o As Object = reader.GetValue(i)
            If o Is DBNull.Value Then
                Console.WriteLine("null\t")
            Else
                Console.WriteLine("{0}\t", o.ToString())
            End If
        Next
        Console.Out.WriteLine()
    End While
    reader.Close()
    command.Dispose()
    connection.Close()
End Sub
```

MICommand

MICommand 提供了必要的接口来根据 MapInfo Data Provider 执行 SQL 命令。MICommand 创建 MIDataReader 和 MIScrollableReader 实例，分别通过 ExecuteReader 和 ExecuteScrollableReader 方法来获取数据。

支持的命令

MICommand 识别的命令为：

Select

```
SELECT < select_list >
    FROM { < table_source > } [ ,...n ]
    [ WHERE < search_condition > ]
    [ GROUP BY expression [ ,...n ] ]
    [ ORDER BY {expression | column_position [ ASC | DESC ] } [ ,...n ]]

< select_list > ::=*
{
    *
    | { table_name | table_alias }.*
    | { expression } [ [ AS ] column_alias ]
} [ ,...n ]

< table_source > ::=
table_name [ [ AS ] table_alias ]
```

Insert

```
INSERT [INTO] { table_name } [ ( column_list ) ]
{ VALUES ({expression | NULL}[, ...n]) | query_specification }
```

Update

```
UPDATE { table_name }
SET {{ column_name }} = { expression | NULL } [, ...n]
[WHERE < search_condition > ]
```

Delete

```
DELETE [FROM] { table_name } [ WHERE < search_condition > ]

< search_condition > ::=
{ [ NOT ] < predicate > | ( < search_condition > ) }
[ { AND | OR } [ NOT ] { < predicate > |
( < search_condition > ) } [ ,...n ] ]

< predicate > ::=
{
    expression [ { = | < > | != | > | >= | < | <= } expression ]
    | string_expression [ NOT ] LIKE string_expression [ ESCAPE
        'escape_character' ]
    | expression [ NOT ] BETWEEN expression AND expression
    | expression IS [ NOT ] NULL
}
```

expression

是列名、伪列、列的别名、常数、函数，或是用运算符连接的以上项目的任意组合。列名和伪列可以是以表名或表别名后接点 (".") 字符为前缀。

group_by_expression

是选择列表的列或选择列表表达式的匹配副本的参考，别名 - 基于 1 的数字，指示列的位置，或 *n* 是表示列的数字的 *coln* 位置。

order_by_expression

是选择列表的列或选择列表表达式的匹配副本的参考，别名 - 基于 1 的数字，指示列的位置，或 *n* 是表示列的数字的 *coln* 位置。

有关表达式，使用表达式的情况以及创建表达式的方式，请参阅[第 178 页中的表达式](#)。

ExecuteFeatureCollection

MICommand 类中的 ExecuteFeatureCollection 方法是 MapInfo ADO.NET 数据提供方和 Feature 对象模型之间的桥梁。该方法在数据源连接上执行命令文本（SQL 语句）并生成 IResultSetFeatureCollection。有关 Feature 模型的讨论请参阅[第 183 页中的 Features 和 Feature 集合](#)。

MIDataReader

对从执行 SQL Select 语句返回的数据，MIDataReader 提供了只前进和只读的访问。要创建 MIDataReader，必须调用 MICommand 对象的 ExecuteReader 方法，而不是直接使用构造函数。

MapInfo Data Provider 允许在同一个连接中使用多个 MIDataReader 实例。但是，如果被访问的 Table 驻留于 Microsoft SQL Server 数据库，则一次只能打开一个 MIDataReader。

MIDataReader 关闭后只有 IsClosed 和 RecordsAffected 属性可以调用。尽管存在 MIDataReader 时任何时候都可以访问 RecordsAffected 属性，但为了保证返回准确的值应始终调用 Close 再返回 RecordsAffected 的值。

使用 MIDataReader 时必须显式调用 Close 方法。

通过 IEnumator 或 IFeatureEnumerator 接口来访问 DataReader 时，如果 MoveNext() 返回 false 则会自动调用 Close()。DataReader 中只能使用一个枚举符。

注： 为了得到最佳性能，MIDataReader 会试图避免创建不必要的对象或复制不必要的数据。这样，对 GetValue 等方法的多重调用可能会返回对同一对象的引用。修改 GetValue 等方法所返回对象的基础值时要小心。

可滚动的数据读取器

MIScrollableReader 从 MIDataReader 派生并提供前向和反向读取。MIScrollableReader 可用的某些选项包括：

- ReadPrevious
- Rewind
- Unwind
- ReadTop
- ReadBottom
- AtTop / AtBottom

注： 创建 MIScrollableReader 比创建 MIDataReader 更消耗资源。

MapInfo SQL

MapInfo SQL 语言允许您添加对 MapXtreme 2005 应用程序强大的分析处理。MapXtreme 2005 通过用于访问数据（特别是 *MICommand* 对象）的 MapInfo ADO.NET Data Provider 公开 SQL 处理过程。此外，表达式也用于标注、主题、图例、AddColumns、图元搜索和选择集处理。

MapInfo SQL 根据 SQL-3 规范标准化。例如，您会发现：

- 字符串常数包括在单引号内
- 标识符可以包括在双引号内
- Select 与 Selection 没有关系

有关 MapXtreme 2005 支持的 MapInfo SQL 语言的完整参考，请参阅 [附录 B: MapInfo SQL 参考](#)。此外，您也可以直接从 Visual Studio .NET 的“帮助”系统中查看。

下面是 MapInfo SQL 语言中可用查询的范围示例。这些示例能够指导您如何添加对应用程序强大的分析处理。

空间查询

VB 示例：

```
Public Shared Sub MapInfo_Data_MICommand(ByVal command As _
    MICommand, ByVal nyGeom As FeatureGeometry)
    command.CommandText = "SELECT * FROM States WHERE Obj _ 
        within @newyork"
    command.Parameters.Add("@newyork", nyGeom)
    Dim reader As MIDataReader = command.ExecuteReader()

    ' Do something here...
    reader.Close()

    ' or... to get a FeatureCollection
    Dim fc As IFeatureCollection =
    command.ExecuteFeatureCollection()
End Sub
```

合计查询

VB 示例：

```
Public Shared Sub MapInfo_Data_MICommand2(ByVal command As _
    MICommand)
    command.CommandText = "SELECT State, Avg(Pop), Sum(Pop), " + _
        "MI_AggregateUnion(Obj) As UnionGeom " + _
        "MI_AggregateEnvelope(Obj) As TotalMBR " + _
        "FROM Cities Group By State"
    Dim reader As MIDataReader = command.ExecuteReader()

    ' Do something here...
    reader.Close()
End Sub
```

获取特定的记录（由键获取）

VB 示例：

```
Public Shared Sub MapInfo_Data_MICommand3(ByVal command As _
    MICommand)
    command.CommandText = "SELECT * FROM States WHERE MI_Key = '5'"
    Dim reader As MIDataReader = command.ExecuteReader()
    If reader.Read() Then
        Dim theGeom As FeatureGeometry = reader.GetFeatureGeometry(0)
        Dim theStateCode As String = reader.GetString(1)
    End If
    reader.Close()
End Sub
```

从特定的记录获取特定的值

VB 示例：

```
Public Shared Sub MapInfo_Data_MICommand4(ByVal command As _
    MICommand)
    command.CommandText = "SELECT Obj FROM States " + "WHERE _
        MI_Key = '5'"
    Dim theGeom As FeatureGeometry = command.ExecuteScalar()
End Sub
```

Insert 语句

VB 示例：

```
Public Shared Sub MapInfo_Data_MICommand5(ByVal command As _
    MICommand, ByVal nyGeom As FeatureGeometry)
    command.CommandText = "Insert Into States (Obj, State, -
        State_Name) " + "Values (@stobj, 'NY', 'New York')"
    command.Parameters.Add("@stobj", nyGeom)
    Dim NumberOfRecordsAffected As Integer = command.ExecuteNonQuery()
End Sub
```

带有 Select 的 Insert 语句

VB 示例：

```
Public Shared Sub MapInfo_Data_MICommand6(ByVal command As _
    MICommand, ByVal floodGeom As FeatureGeometry)
    command.CommandText =
        "Insert Into TempCounties (Obj, County, State, Pop, PopDensity, -
            Area) " + "Select Obj, County, State, Pop, Pop / MI_Area(Obj, -
            'sq mi', 'Spherical'), " + " MI_Area(Obj, 'sq mi', -
            'Spherical') " + " From Counties Where Obj intersects -
            @floodRegion"

    command.Parameters.Add("@floodRegion", floodGeom)
    Dim NumberOfRecordsAffected As Integer = command.ExecuteNonQuery()
End Sub
```

Update 语句

VB 示例：

```
Public Shared Sub MapInfo_Data_MICommand7(ByVal command As _
    MICommand)
    command.CommandText = "Update States Set Population = "
    Population * 1.05" + " Where Population / MI_Area(Obj, _
    'sq mi', 'Spherical') > 2000"
    Dim NumberOfRecordsAffected As Integer = command.ExecuteNonQuery()
End Sub
```

Delete 语句

VB 示例：

```
Public Shared Sub MapInfo_Data_MICommand8(ByVal command As _
    MICommand)
    command.CommandText = "Delete From States Where MI_Key = @varkey"
    ' Could be a MapInfo.Data.Key object too.
    command.Parameters.Add("@varkey", "4")
    Dim NumberOfRecordsAffected As Integer = command.ExecuteNonQuery()
    command.Parameters("@varkey").Value = "5"
    NumberOfRecordsAffected = command.ExecuteNonQuery()
End Sub
```

IN/ANY/ALL 运算符和子查询

下面是与数据访问相关的新的和增强的功能。

- 现在提供带有文字列表和子查询的 IN、 ANY 和 ALL 运算符。例如：

```
Select ... From T Where X IN (2, 6, 12)Select ... From T Where X IN
(Select X From U Where B > 50)
```

- 现在提供相关联的子查询，其中子查询引用来自外部查询的列（或表达式）。例如：

```
Select ... From T Where X IN (Select X From U Where B = T.A)
```

- 空间 OR 列表已进行优化。例如：

```
{Select ... From T Where Obj Within @Var1 OR Obj Within @Var2 OR
Obj Within @Var3}
```

表达式

表达式是用于描述和格式化数据的语句。例如，会有“a median income of more than \$50,000”或“female percent of population”这样的英文表达式。

表达式使用列名、常数（即特定数据值）和作用在列和常数上的函数和运算符形成。运算符和函数以 MapInfo SQL 语言定义，这是为 MapXtreme 2005 和其它 MapInfo .NET 支持的产品新开发的功能。有关详细信息，请参阅本指南中的[附录 B: MapInfo SQL 参考](#)，或通过 Visual Studio.NET 中集成的“帮助”联机查看详细信息（查找“动态帮助”目录窗格中的“MapInfo SQL 参考”）。

使用表达式可以生成大部分数据。通过使用表达式，您可以：

- 仅显示感兴趣的列和行。
- 根据现有列的内容通过计算新值派生新的列。
- 合计数据来处理分类汇总而不是整个表。
- 将两个或多个表的数据组合到一个结果表中。

要用的数据集包括的对象和信息要比项目所需的对象和信息多。很多情况下，运用完整数据产品的子集更为容易。例如，如果要通过人口普查记录跟踪某个郡的犯罪统计，则不必处理整个州的人口普查记录。可以使用表达式仅提取该郡的人口普查记录。

在整个 MapXtreme 2005 中，表达式用于以下情况：

- SQL 语句 (select, insert, update, delete, group by, order by)
- 将表达式作为参数使用的 SQL 函数（例如 MI_Area() 中的几何体参数是返回几何体对象的表达式。）
- 添加列 (MapInfo.Data.Table.AddColumn 根据表达式创建临时列。)
- 图元搜索 (SearchInfo 和 SearchInfoFactory)
- 主题 (FeatureStyleModifier)
- 标注 (LabelModifier)
- InfoTips (FeatureLayer、MapTools)
- 表达式对话框

创建表达式

最简单的表达式由常数组成，例如“2”（数字示例）或“Friday”（文本示例）。

其它简单表达式由列名称组成，例如：

POP_2000
STATE

在 select 语句中请求特定的多个列时，例如这些列一起称为表达式列表。

```
Select colA, colB, colC from Table1, Table2  
Select colA/2, ColB/ColC from Table1
```

也可以编写对数据执行数学运算的表达式。

例如，RENT + UTILITIES 是将两列相加在一起的表达式。该表达式可以用在 SQL 语句中查找每月总费用少于 \$800 的公寓。

Where 子句 - 布尔表达式

布尔表达式是搜索条件，生成 True 或 False。例如，表达式

```
2 < 5
```

是布尔表达式，因为结果是 True。

所有包含关系运算符（例如小于号 <）的表达式是布尔表达式。运算符 AND、OR 和 NOT 是布尔运算符。布尔表达式也称为比较表达式、条件表达式和关系表达式。

```
POP_2000 > 500000  
POP_2000 <= POP_1990  
PROVINCE <> 'Ontario'  
County = 'Columbia' AND VALUE >= 250000
```

MapInfo SQL 中支持的运算符在第 362 页附录 B 中的运算符中定义。有关查询代码示例请参阅第 174 页中的 *MapInfo SQL*。

布尔表达式用于 SQL 语句的“where 子句”中。where 子句是控制返回的行（生成 True 的行）的表达式。

例如，该语句中的布尔表达式后跟 WHERE。只有落在 France 边界内位于 Europe 表中的对象才返回 True。

```
"SELECT * FROM Europe WHERE MI_Geometry within @France";
```

表达式中的函数

MapXtreme 2005 中的函数用于创建更加复杂的表达式来检索满足特定标准的数据。例如，MapInfo SQL 支持处理字符串、日期和数字的大量常用数据库函数。

MapInfo SQL 中最强大的函数是那些利用绘制地图数据的空间性质的函数。这些地理函数用于创建新的几何体，度量面积和长度，返回空间信息，验证多个几何体之间的空间关系等。支持的函数在 MapInfo SQL 参考中定义，始于第 365 页附录 B 中的函数列表。

查看边界表的面积（例如学校地区）就是在表达式中使用函数的一个示例。使用函数 `MI_Area()` 返回表中每个记录的面积。

其它在表达式中使用函数的示例可以在以下的“表达式示例”一节中找到。

表达式示例

以下章节着重说明了在 MapXtreme 2005 中使用表达式的各种情况。

SQL 语句示例

该示例将从位于 Germany 且人口超过 1 百万的 Eurcity_1K 表选择所有记录。

```
Select * from Eurcity_1K WHERE (MI_Geometry MI_Within @Germany) AND  
Tot_Pop > 1000000
```

MapInfo SQL 函数示例

以下表达式使用 MapInfo SQL 函数查找缓冲区内的图元。

```
Obj CentroidWithin MI_Buffer(Obj, 5, 'km', 'Spherical', 24)
```

该表达式使用 MapInfo SQL 特殊关键字，为名为‘Obj’的地理对象所保留。该关键字说明了诸如坐标系和边界的对象的几何体。该关键字与以前版本的 MapX 和 MapInfo Professional 兼容。且等价于列名称 `MI_Geometry`。

注意，`km` 和 `Spherical` 包括在单引号内。在 MapInfo SQL 中，字符串文字必须包括在单引号内，而标识符（例如列名称、表名称、别名等）应该包括在双引号中。只有分析的逻辑不能正确分析标识符时，标识符才需要加引号。这种情况包括名称中有空格或有其它特殊字符的标识符。

要查找位于缓冲区外的图元，可以使用以下表达式：

```
NOT Obj CentroidWithin MI_Buffer(Obj, 5, 'km', 'Spherical', 24)
```

添加列示例

使用 `AddColumns` 方法将临时列添加到表时，提供的列包含了定义如何计算列值的表达式。如果多个源记录要匹配到添加列的表中的单一记录，则表达式可以包含聚合函数。

以下示例使用表示人口密度的表达式 "`Pop_1990 / MI_Area(Obj, 'sq mi', 'Spherical')`"。这些表达式前是它们新的列名称。`PopDensity1990` 和 `PopDensity2000`。

VB 示例：

```
Public Shared Sub MapInfo_Data_TableAddColumns(ByVal miTable _  
    As Table)  
    Dim NewCols As Columns = New Columns  
    NewCols.Add(New Column("PopDensity1990", "Pop_1990 / _  
        MI_Area(Obj, 'sq mi', 'Spherical')"))  
    NewCols.Add(New Column("PopDensity2000", "Pop_2000 / _  
        MI_Area(Obj, 'sq mi', 'Spherical')"))  
    miTable.AddColumn(NewCols)  
End Sub
```

图元搜索示例

以下示例使用布尔表达式 SearchWhere("State='FL'")，该表达式执行时将为包含 FL 的每行返回值 1。

VB 示例：

```
Public Shared Sub MapInfo_Data_SearchInfo(ByVal catalog As Catalog)  
    Dim fFlorida As Feature = catalog.SearchForFeature("usa",  
        MapInfo.Data.SearchInfoFactory.SearchWhere("State='FL'"))  
    Dim si As SearchInfo =  
        MapInfo.Data.SearchInfoFactory.SearchWithinGeometry(fFlorida._  
            Geometry, ContainsType.Centroid)  
    Dim fc As IResultSetFeatureCollection =  
        MapInfo.Engine.Session.Current.Catalog.Search("uscty_1k", si)  
  
    ' Set the map view to show search results  
  
    MapInfo.Engine.Session.Current.MapFactory(0).SetView(fc.Envelope)  
    ' Set the view of the first map.  
  
    ' Add results to selection.  
    MapInfo.Engine.Session.Current.Selections.DefaultSelection.Add(fc)  
End Sub
```

主题图表达式示例

饼图主题是根据主题表达式的数字值绘制饼图的对象主题。主题的表达式由三列组成。

"Pop_Native", "Pop_Asian", "Pop_Other"。

VB 示例：

```
Public Shared Sub MapInfo_Mapping_Thematics_PieTheme(ByVal _
    map As Map)
    ' Load a map based on one table
    map.Load(New MapTableLoader("world.tab"))
    Dim lyr As FeatureLayer = CType(map.Layers("world"), FeatureLayer)

    ' Create a new pie theme
    Dim pieTheme As MapInfo.Mapping.Thematics.PieTheme = New
    MapInfo.Mapping.Thematics.PieTheme(map, lyr.Table, "Pop_Native", _
        "Pop_Asian", "Pop_Other")

    ' Create an object theme layer based on that pie theme
    Dim thmLayer As ObjectThemeLayer = New ObjectThemeLayer("World _"
        "Pop Growth Rate", Nothing, pieTheme)

    'Add object theme to the map's layer collection.
    map.Layers.Add(thmLayer)
End Sub
```

标注表达式示例

表达式是标注修饰符的属性，该标注修饰符用于修改用于生成标注的默认属性。

VB 示例：

```
Public Shared Sub MapInfo_Mapping_OverrideLabelModifier(ByVal _
    modifier As OverrideLabelModifier)
    modifier.Name = "'Capital Name: ' + Capital"
End Sub
```

InfoTips 表达式示例

InfoTips 是工具悬停在 Feature 上时显示的文本项。表达式可以与标注所用方式类似的方式用于生成 InfoTips 的文本。使用 MapTool.SetInfoTipExpression 静态帮助程序函数来设置 InfoTip。该函数负责创建 InfoTip 哈希表并将图层条目添加到哈希表（如果哈希表还不存在）。可以设置每个工具启用或禁用 InfoTips。

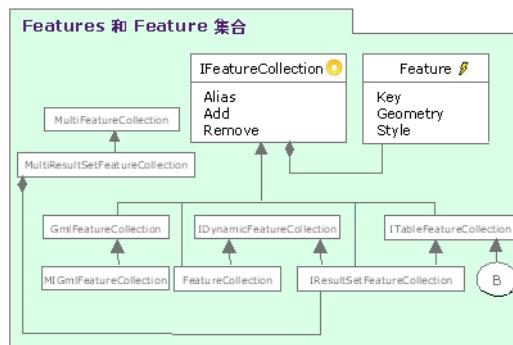
以下示例将生成两行文本标注来显示 Table 别名和光标悬停的对象质心的 X 或 Y 坐标。

VB 示例：

```
Public Shared Sub MapInfo_Mapping_HowDoICreateExprForInfoTip(ByVal
mapControl1 As MapControl)
    MapTool.SetInfoTipExpression(mapControl1.Tools.MapToolProperties,
        CType(mapControl1.Map.Layers(0), FeatureLayer), "@TableAlias + "
        & char(13) + 'Centroid X:' + MI_CentroidX(obj) + ' Y:' +
        MI_CentroidY(obj))
End Sub
```

Features 和 Feature 集合

MapXtreme 2005 中的 Feature 类对象模型提供了不基于 SQL 的方法来访问和操控数据。本节涵盖了 Feature 类和 IFeatureCollection 接口。处理图元时的主要任务是可以使用查询定义对象搜索图元。



Feature

图元由它们的几何体、样式、数据源、键和属性描述。通常，图元为表中的行。图元的几何体是 FeatureGeometry 对象。FeatureGeometries 可以包含给定的区域 (MultiPolygon)，位置 (Points、MultiPoints) 和距离 (MultiCurves、LegacyArcs)。从 FeatureGeometry 派生和用于地图图元的其它 Geometry 类是 FeatureGeometryCollection 和 LegacyText。（矩形、有圆角的矩形和椭圆对象也从 FeatureGeometry 派生，但主要用于外观显示。）

计算机化地图其中一项主要用途就是获取关于图元的信息。在 MapXtreme 2005 中，可通过多种方法将图元返回到 FeatureCollections，例如，使用构架从空白创建，使用选择集工具或方法选择，或通过搜索这些满足特定标准集合的工具或方法的 Catalog 选择。

可以使用 Load 方法强制 Load。Feature 保存回表之前，对 Feature 所做的更改不反映在底层表（如果有的话）中。使用 Update 方法、UpdateFeature 或 InsertFeature 可以完成该操作。此外，可以使用 DiscardEdits 方法在保存编辑之前丢弃任何对 Feature 对象所做的编辑。

Feature 具有描述 Feature 属性的构架。Columns 属性描述该构架。

从表检索图元

Table 是 Feature 集合的一种类型。因此，表内的 Features 可以直接枚举。例如：

VB 示例：

```
Dim ftr As Feature  
For Each ftr In table  
    ...
```

表的默认图元枚举符内部使用具有以下命令的 MIDataReader：

```
command.CommandText = "Select MI_Key, * From \"\" + table.Alias + "\"";
```

要检索表中图元的子集，请使用某一种 Catalog.Search 方法或
MICommand.ExecuteFeatureCollection 方法。

修改表中的图元

要修改表中的图元，请使用以下方法之一。

- Feature.Update
- Table.UpdateFeature
- Table.InsertFeature

Feature 集合

Feature 集合是一组 Feature 对象。集合中的所有 Features 共享同一 Schema（列）。Feature 集合包含了某个构架，该构架是所有其成员图元实例的构架。某些 Feature 集合拥有它们的 Features，而其它 Feature 集合维护对 Features 的引用。

搜索图元

MapInfo 地图绘制应用程序最普通的任务之一就是搜索满足某些标准的图元。有了感兴趣的图元之后，就可以执行更多分析，例如主题绘制。在 MapXtreme 2005 中，搜索图元有很多方式：使用工具，使用 Catalog 搜索方法，或使用 SQL 和 MapInfo ADO.NET 数据提供方。

以下代码示例说明了搜索同一内容的两种方式，该例中的城市为 New York。

```
// Using SQL
command.CommandText = "Select Obj From States Where state = 'NY';
FeatureGeometry nyGeom = command.ExecuteScalar() as FeatureGeometry;
command.CommandText =
    "SELECT * FROM Cities WHERE Obj within @newyork";
command.Parameters.Add("@newyork", nyGeom);
MIDataReader reader = command.ExecuteReader();
// or... to get a FeatureCollection
IFeatureCollection fc = command.ExecuteFeatureCollection();

// Using Features
Feature fNY = catalog.SearchForFeature("States",
    SearchInfoFactory.SearchWhere("state='NY'"));
SearchInfo si = SearchInfoFactory.SearchWithinFeature(fNY,
    ContainsFilter.ContainsType.Centroid);
IDynamicFeatureCollection dfc =
    catalog.Search("Cities", si) as IDynamicFeatureCollection;
Console.Out.WriteLine(
    "There are {0} cities whose centroid is within NewYork."
    dfc.Count);
```

有关 SQL 搜索的完整讨论请参阅第 170 页中的 *MapInfo ADO.NET* 数据提供方。以下章节着重介绍使用 Catalog 和 SearchInfo 进行搜索。

Catalog 搜索方法

Catalog 具有大量搜索方法成员函数。重载的 Search 方法可用于在一个或多个表上搜索。这些方法包括了不同的参数，使得每个搜索惟一。例如，基本的 Search (Table, SearchInfo) 搜索给定的表并返回 FeatureCollection。Search (ITableEnumerator, SearchInfo) 方法在多个表上搜索并返回 MultiResultSetFeatureCollection。

SearchForFeature 方法从结果返回第一个 Feature。SearchReader 方法与结果一起返回 MIDataReader 光标。

代码示例：SearchForFeature

以下示例说明了如何使用 Catalog.SearchForFeature 和 Catalog.SearchWithinGeometry。它在 uscty_1k 表中查找 Florida 内的所有城市。假定 "usa" 和 "uscty_1k" 表均打开，此外还有一张地图。

VB 示例：

```
Public Shared Sub MapInfo_Data_SearchInfo(ByVal catalog As Catalog)
    Dim fFlorida As Feature = _
        catalog.SearchForFeature("usa", MapInfo.Data._.
        SearchInfoFactory.SearchWhere_("State='FL'"))
    Dim si As SearchInfo = _
        MapInfo.Data.SearchInfoFactory.SearchWithinGeomeTry(fFlorida._.
        Geometry, ContainsType.Centroid)
    Dim fc As IResultSetFeatureCollection = _
        MapInfo.Engine.Session.Current.Catalog.Search("uscty_1k", si)

    ' Set the map view to show search results
    MapInfo.Engine.Session.Current.MapFactory(0).SetView(fc.Envelope)
    ' Set the view of the first map.

    ' Add results to selection.
    MapInfo.Engine.Session.Current.Selections.DefaultSelection.Add(fc)
End Sub
```

SearchInfo 和 SearchInfoFactory

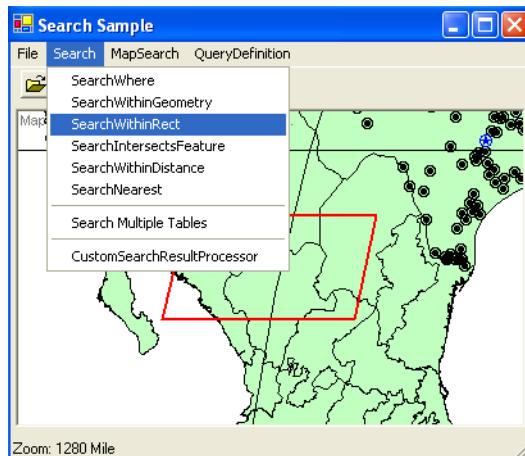
MapInfo.Data.SearchInfo 类定义搜索中所用的查询并处理搜索结果必需的任何后续处理。

SearchInfoFactory 创建 SearchInfo 对象。SearchInfoFactory 包含了大量搜索方法，允许使用对搜索位置的空间引用或使用在屏幕上绘制的几何体进行搜索。

下表说明了 SearchInfoFactory 搜索方法。

SearchInfoFactory 方法	行为
SearchAll	返回所有行。
SearchNearest	返回带有表几何体的行，该表几何体最接近给定的搜索点。
SearchWhere	返回给定的 where 子句指定的行。
SearchWithinDistance	返回表几何体包含在搜索点、矩形或几何体的距离范围内的行。该方法使用 Geometry.Distance 方法确定对象是否在搜索区域内。先前的 SearchWithinDistance 对距离进行缓冲并在缓冲区内搜索，因此结果不够精确。
SearchWithinFeature	返回表几何体包含在搜索图元几何体内的行。
SearchWithinGeometry	返回表几何体包含在搜索几何体内的行。
SearchWithinRect	返回表几何体与给定矩形相交的行。
SearchIntersectsFeature	返回表几何体与搜索图元的几何体相交的行。
SearchIntersectsGeometry	返回表几何体与搜索几何体相交的行。
SearchWithinScreenRadius	在表几何体与屏幕圆圈相交的位置，创建返回行的 SearchInfo。
SearchWithinScreenRect	返回表几何体与给定屏幕矩形相交的行。

MapXtreme 2005 包含了 Search 示例应用程序，可以运行和了解每种搜索类型。下图显示了带有 SearchInfoFactory 方法的 Search 菜单，该方法使用空间引用。Map Search 菜单具有根据绘制的屏幕几何体对象（圆或矩形）进行搜索的方法。Query Definition 菜单突出使用各种作用在 SQL 语句上的筛选器。示例位于 ..\MapInfo\MapXtreme\6.x\Samples\Desktop\Features\Search 文件夹中。



代码示例

本节包括了属于 SearchInfoFactory 方法的多个代码示例。

SearchNearest

使用 SearchNearest 方法后，代码模拟 Select Point 工具行为，通过鼠标单击选择最顶层的项目，并将它们添加到选择集。

VB 示例：

```
Public Shared Sub MapInfo_Mapping_SearchInfoFactory(ByVal _  
    mapControl1 As MapControl)  
    ' Get a point from mouse click. Hard coded value use in sample.  
    Dim pt As System.Drawing.Point = New System.Drawing.Point(100, _  
        100)  
    ' Assumes there is a MapControl with a map in it.  
    Dim map As Map = mapControl1.Map  
    Dim session As ISession = MapInfo.Engine.Session.Current  
    Dim si As SearchInfo = _  
        MapInfo.Mapping.SearchInfoFactory.SearchNearest(map, _  
            pt, 3) ' 3 pixel tolerance radius  
    si.QueryDefinition.Columns = Nothing  
    ' fetch all columns instead of just default  
    ' Customize to stop at topmost layer where something is found  
    CType(si.SearchResultProcessor, ClosestSearchResultProcessor)._  
        Options = ClosestSearchOptions.StopAtFirstMatch  
    ' Puts results of search directly into default selection  
    ' Searches all tables in map in order from top to bottom.  
    MapInfo.Engine.Session.Current.Catalog.Search(map._  
        Layers.GetTableEnumerator(), si, _  
        session.Selections.DefaultSelection, _  
        ResultSetCombineMode.Replace)  
End Sub
```

SearchIntersectsFeature

VB 示例：

```
Public Shared Sub _  
    MapInfo_Data_SearchInfoFactorySearchIntersectsGeomeTry(ByVal _  
        map As Map)  
    Dim ti As Table = _  
        MapInfo.Engine.Session.Current.Catalog.GetTable("usa")  
    Dim lParks As MapInfo.Mapping.FeatureLayer = _  
        CType(map.Layers("USA"), MapInfo.Mapping.FeatureLayer)  
    Dim g As MapInfo.GeomeTry.FeatureGeomeTry = New _  
        MapInfo.GeomeTry.Point(map.GetDisplayCoordSys(), -98, 34)  
    Dim si As SearchInfo = _  
        MapInfo.Data.SearchInfoFactory.SearchIntersects_  
        Geometry(g, MapInfo.Data.IntersectType.GeomeTry)  
    Dim fc As IResultSetFeatureCollection = _  
        Session.Current.Catalog.Search("usa", si)  
    map.SetView (fc.Envelope)  
End Sub
```

SearchWithinScreenRadius

该示例说明了如何使用屏幕绘制的圆来搜索图元。

```
// find nearest city to center of map
private void menuItemSearchNearest_Click
    (object sender, System.EventArgs e)
{
    try
    {
        Cursor.Current = Cursors.WaitCursor;
        // to compare to SearchWithinScreenRadius, we are
        // calculating the search distance the same way it does
        System.Drawing.Rectangle rect=mapControl1.Bounds;
        System.Drawing.Point pt =
            new System.Drawing.Point(rect.Left, rect.Top);
        pt.X += rect.Width/2;
        pt.Y += rect.Height/2;

        DPoint dpt1 = new DPoint();
        // convert center point to map coords (could use map.Center)
        _map.DisplayTransform.FromDisplay(pt, out dpt1);
        Distance d =
            MapInfo.Mapping.SearchInfoFactory.ScreenToMapDistance
            (_map, 3);

        SearchInfo si =
            MapInfo.Data.SearchInfoFactory.SearchNearest
            (dpt1, _map.GetDisplayCoordSys(), d);
        IResultSetFeatureCollection fc =
            _catalog.Search("uscty_1k", si);

        MapInfo.Geometry.Point p =
            new MapInfo.Geometry.Point
            (_map.GetDisplayCoordSys(), dpt1);
        FeatureGeometry buffer =
            p.Buffer(d.Value, d.Unit, 20, DistanceType.Spherical);
        ShowSearchGeometry(buffer);

        SelectFeatureCollection(fc);
    }
    finally
    {
        Cursor.Current = Cursors.Default;
    }
}
```

代码示例：如何查找包含点的图元

您也可能希望通过相反类型的搜索查找包含点的图元。例如，在运输规划中，可能要了解客户所在区域。

VB 示例：

```
Dim g As MapInfo.Geometry.FeatureGeometry = New _
    MapInfo.Data.SearchInfoFactory.SearchIntersectsGeometry(g, _
    MapInfo.Data.IntersectType.Geometry)
Dim irfc As MapInfo.Data.IResultSetFeatureCollection =
    MapInfo.Engine.Session.Current.Catalog.Search("british_"
    "columbia", si)
Me.MapControl1.Map.SetView(irfc.Envelope)
```

该示例及其它示例都位于 MapInfo Knowledgebase (*MapXtreme 2004 > Selections*)。

从 DBMS 访问数据

MapXtreme 2005 提供了空间服务器访问。这是一项强大的功能，允许开发人员连接到存储在空间服务器（例如，在 Microsoft SQL Server、Informix 或 Oracle Spatial 数据库上运行的 MapInfo SpatialWare）中的实时数据。空间服务器允许公司将其地图数据存放在企业数据库中，以便集中管理并获得更高的安全性。SpatialWare 这样的空间服务器为公司的空间数据提供了高级查询处理和提高的性能。

本章内容：

- ◆ 访问远程空间数据 194
- ◆ 通过 .TAB 文件访问远程表 194
- ◆ 不使用 .TAB 文件访问远程表 195
- ◆ 使用 X/Y 列映射 DBMS 数据 195
- ◆ Oracle 支持 195
- ◆ OCI 连接对话框 199
- ◆ DBMS 连接字符串格式 199
- ◆ 在服务器表查询中定义可制图的表 201
- ◆ 访问属性数据 203
- ◆ 性能问题 203
- ◆ 运用缓存 204
- ◆ MapInfo MapCatalog 206
- ◆ 将行添加到 Map Catalog 208
- ◆ 每记录样式 212
- ◆ 故障诊断 213

访问远程空间数据

可以使用具有不同 DBMS 服务器的 MapXtreme 2005 访问数据。这些服务器包括：

- MS Access 和 SQL Server2000。
- Informix IDS 9.3。
- 空间服务器，例如 SpatialWare 4.6 for Microsoft SQL Server 2000 和 4.8.1 for Microsoft SQL Server 2000 以及 SpatialWare 4.5.1 for Informix IDS 9.3。
- MapInfo Geocoding DataBlades for Informix。
- Oracle Spatial（8.1.7、9ir1 和 9ir2）。
- Oracle 9ir2 Locator。

可以使用 MapInfo.Data 命名空间中的 TableInfoServer 类，从 DBMS 中的数据添加表。

有关添加空间数据的详细信息包括在以下章节中。一旦决定了如何处理数据，请遵循相应章节中的指示。

MapXtreme 2005 仅支持打开 SpatialWare 4.5（或更高版本）表。

通过 .TAB 文件访问远程表

MapXtreme 2005 应用程序可以“实时”访问 DBMS 数据，或可以打开 MapInfo Professional 链接的表。但是，链接的表是只读的，不能通过应用程序刷新。表中数据实际上是来自远程数据库的，不会反映位于本地链接的版本中的数据。

可以创建 .TAB 文件以提供对远程数据的访问。要使用 MapInfo Professional 生成 .TAB 文档，请选择“文件”->“打开 DBMS 表”。

.TAB 文件是文本文件，可以使用任何文本编辑器创建 .tab 文件。创建 .tab 文件以后，访问该文件的方式与通过 Catalog 对象或通过 Workspace 管理器以编程方式访问任何其它 MapInfo .TAB 文件的方式相同。

不使用 .TAB 文件访问远程表

应用程序不需要 .tab 文件就可以访问远程数据。以下代码示例说明了该过程：

VB 示例：

```
Public Shared Sub MapInfo_Data_TableInfoServer(ByVal connection As _
    MIConnection)
    ' Note: Do not specify any columns. These are determined
    ' dynamically from the query

    Dim ti As TableInfoServer = New TableInfoServer("Provinces")
    ti.ConnectionString = "SRVR=ontario;UID=mapx;PWD=mapx"
    ti.Query = "Select * From Provinces"
    ti.Toolkit = ServerToolkit.Oci

    ti.CacheSettings.CacheType = CacheOption.Off ' On is the default
    Dim tbl As Table = connection.Catalog.OpenTable(ti)
End Sub
```

使用 X/Y 列映射 DBMS 数据

可以访问来自 DBMS 表的数据，该 DBMS 表具有 X/Y 坐标。需要创建 Map Catalog，将表注册为 SpatialType 4.0 并指定两个列名作为坐标。这两个列应该在该表上被编入索引。通过 ODBC 连接至 DBMS 并在查询中指定新的列作为 “OBJ” 或 “MI_FEATUREGEOMETRY”。

Oracle 支持

MapXtreme 2005 支持以下 Oracle 数据库系统：

- Oracle 9i
- Oracle 9iR2
- Oracle 10G

要从 MapXtreme 2005 应用程序连接到 Oracle 数据库，必须安装 Oracle OCI 连接客户端并授予适当的权限。请参阅 Oracle 文档以获取详细信息。

几何体转换

下表显示从 MapXtreme 2005 对象到 Oracle Spatial (SDO_Geometry) 的转换。

从 MapInfo	到 Oracle
NULL 几何体	NULL
Point	1 POINT
MultiCurve (IsLegacyLine = true)	2 LINESTRING 几何体包含一行字符串
Polygon	3 POLYGON 几何体包含一个或多个多边形。
FeatureGeometryCollection	4 Collection 几何体是元素的异种集合。
MultiPoint	5 MULTIPOINT
MultiCurve	6 MULTILINESTRING 几何体具有多行字符串。
MultiPolygon	7 MULTIPOLYGON 几何体具有多个或多边形。
Ellipse	NULL
LegacyArc	NULL
Rectangle	NULL
LegacyText	NULL
RoundedRectangle	NULL
PieTheme, BarTheme	NULL

下表描述从 Oracle (GTYPES) 到 MapInfo Spatial 对象的转换。

从 Oracle GTYPES	到 MapInfo
0 UNKNOWN_GEOMETRY (Spatial 忽略此几何体。)	
1 POINT 几何体包含一个点。	Point
2 LINESTRING 几何体包含一行字符串。	MultiCurve
3 POLYGON 几何体包含一个多边形。	MultiPolygon
4 Collection 几何体是元素的异种集合。	FeatureGeometryCollection
5 MULTIPPOINT 几何体具有多个点。	MultiPoint
6 MULTILINESTRING 几何体具有多行字符串。	MultiCurve
7 MULTIPOLYGON 几何体具有多个多边形 (多外部边界)。	MultiPolygon

多维几何体支持 (大于二维)

MapXtreme 2005 应用程序可以读取作为 SDO_GEOMETRY 列的图层的数据和地图，这些列包含以两个以上的维定义的几何体。不过，只有前两个维用作 x 轴和 y 轴。

SDO_GEOMETRY Arc 和 Circle 转换

可将圆和圆弧解析为具有 25 段 / 360 度圆的 MultiCurves。

可视化不可转换的 Oracle 对象

MapXtreme 2005 应用程序不能转换的 Oracle 空间对象在 SDO_Spatial 点的位置生成带有默认样式（黑色的星型标记）的 Point 对象，或生成纵坐标数组中的第一个 SDO_Spatial 纵坐标。这样就可以在不可转换的对象所属的正确图形区域中可视化这些对象。不可转换对象的示例是：用户定义的对象 GTypes 0,4,5，或包含无法识别的 GTypes、ETypes 或解释的无效 sdo_geometries。使用 sdo_validate_geometry()，第二个类也会失败。

质心支持

MapXtreme 2005 应用程序使用 SDO_POINT 作为多边形的质心值。质心功能用于定位标注，此外还影响对象的工具选择。如果点位于区域内，则 Oracle

SDO_GEOMETRY.SDO_POINT_TYPE 字段（如果非 NULL 的话）将被解释为图元质心。如果点位于区域外，则始终计算它的质心。

注： 当前在 MapXtreme 2005 中没有设置区域图元质心的方法或工具，但可以读取并使用存储的质心。

附加信息

下面的 sdo_geometry 值不总是可转换的，或不总是来自 MapInfo 对象格式而没有数据丢失。如果进行更新的话，某些几何体详细信息可能会丢失。

- GTYPE 0 - 用户定义的 / 未知的几何体
- GTYPE 4 - 元素的异种集合

如果非 NULL，则这些类型被解释（可能时）为单一点 SDO_POINTTYPE 值。它们“抓取”将被解释为 NULL 几何体的有序数组的第一个点。

注： 包括圆或圆弧组件的任何其它 SDO_GEOMETRY 值会转换为包含 25 行/360 度的行字符串。

Oracle Spatial 引用支持 (SRID)

可使用空间引用系统定义 Oracle SDO_GEOMETRY 列。通过在 USER_SDO_GEOM_METADATA 中提供 Oracle SRID，还通过在存储的 SDO_GEOMETRY 值中指定 SRID 来执行此操作。如果表包含具有指定的 SRID 的 Oracle Spatial 列，则 MapXtreme 2005 应用程序可以查询并正确解释数据。MapInfo_MapCatalog 必须包含与数据的 SRID 中指示的字符串相同的 MapInfo Professional CoordSys 字符串，因为在当前用于解释和更新数据的 MapInfo_MapCatalog 中就是 Coordsys。

如果 Spatial 列未包含 SRID 值（该值为 NULL），则 MapXtreme 2005 应用程序还能够通过在 MapCatalog 中定义的 MapInfo Professional Coordsys 来解释数据。

在将使用纬度 / 经度坐标系 (Geodetic Data) 的表加载到 Oracle Spatial 时，验证所有几何体坐标是否位于经度 (-180,180) 和纬度 (-90, 90) 之间非常重要。超出该范围的 Geodetic 数据坐标在 Oracle Spatial 中不受支持，并且可能引起问题。可以在加载前使用 MapInfo Professional 检查数据；或可以在将数据加载到 Oracle Spatial 后，使用 Oracle Spatial SDO_Geom.VALIDATE_LAYER() 函数检查数据。

OCI 连接对话框

MapInfo.Data.DBMSConnectionCollection 类现在通过订阅 ConnectionFailedEvent 事件句柄支持 ConnectionFailed 事件。激发后，该事件会显示 OCILoginDlg 以使用户可以更改登录信息并再次重试到 Oracle 数据库的连接。该句柄特定于 MapXtreme 2005 OCI 工具包。还可通过 MapInfo.Windows.Dialogs 命名空间中的 LoadMapWizard 类获得该事件。

DBMS 连接字符串格式

ODBC 连接字符串格式

ODBC 连接字符串的格式由多个使用分号 (;) 分隔的子句定义。每个子句都具有格式：
Key=Value。重要的键在以下列出。

关键字	描述
DLG=	控制连接对话框显示的数字： 0 - 禁用连接对话框。 1 - 显示连接对话框。 2 - 仅在需要时显示连接对话框（即，不是所有需要的信息都已提供时）[默认]。 注： 使用对象池的 ASP.NET 应用程序必须在用于 ODBC 的连接字符串中具有 DLG=0 子句。
DSN=	指定 ODBC 数据源名称。 注意： 如果使用 DSN= 语法键，则指定的名称必须与用户系统上使用的数据源名称匹配。注意：不同的用户可能使用不同的名称来引用相同的数据源。如果无法预先了解要使用的数据源名称，则使用 DRIVER= 语法键，而不使用 DSN= 语法键。
DRIVER=	指定已安装驱动程序的正确驱动程序名称。用来替代 DSN= 语法键。 示例： DRIVER={SQL Server} 注： Informix 2.80.0861 不支持 DRIVER=。也不能应用于 Oracle Spatial。
UID=	必要时指定数据源需要的 UserId。
PWD=	必要时指定数据源的用户口令。对于要匹配的两个字符串，口令不必位于连接字符串中。如果两个表位于相同的数据库，则连接字符串相同。

注： 连接属性 / 参数不必按顺序，且可以使用对话框从现有的连接池获得连接以避免冗余的连接。在 API 的先前版本中，如果每次使用对话框连接到相同的数据库，或如果没有按照文档中的顺序对连接字符串中的连接关键字进行排序，则连接不能共享且会获得多个连接。

Web 应用程序中的 ODBC 图层和对象池

通过 ODBC 将远程图层添加到使用对象池的 ASP.NET 应用程序时，确保在连接字符串中具有 DLG=0 子句。这将避免显示可能超时的不必要的用户和口令对话框。这适用于 TAB 文件和工作空间。下面的连接字符串示例显示了突出显示的 DLG=0 子句。

```
<ConnectionString>DRIVER={SQL Server};DATABASE=Devel;Server=Paladin;  
UID=devel;PWD=devel;QuotedID=Yes;Trusted_Connection=No;  
Network=DBMSSOCN;Address=PALADIN,1433;DLG=0</ConnectionString>
```

在使用 TableInfoServer 且对象池已打开时，要访问 SQL Server，则在连接字符串中指定 "DLG=SQL_DRIVER_NOPROMPT" 以避免抛出 MapInfo.Data.TableException: 不能打开表。

Oracle Spatial 连接字符串格式

这些是 Oracle Spatial 关键字。该字符串由多个使用分号 (;) 分隔的子句定义。每个子句都具有格式：Key=Value。重要的键在下表中列出。

关键字	描述
SRVR=	反映在 Oracle Net8 EasyConfig 实用程序中设置的服务器的服务名称。Oracle 连接需要该值，但该值对 ODBC 连接不适用。
UID=	必要时指定数据源需要的 UserId。
PWD=	必要时指定数据源的用户口令。

样本连接字符串

此处是用于 Oracle Spatial、IUS/UDO 和 SpatialWare ODBC 驱动程序的样本连接字符串。

Oracle Spatial 连接字符串：

```
UID=george;PWD=password;SRVR=OracleSpatial9i
```

IUS/UDO 连接字符串：

```
DRIVER={INFORMIX 3.81 32 BIT};UID=informix;PWD=secret;DATABASE=sw;  
HOST=HostName;SERVER=ServerName;SERVICE=sqlexec;PROTOCOL=onsoctcp;
```

SpatialWare for SQL Server 字符串：

```
DRIVER={SQL SERVER};  
SERVER=ServerName;VID=Troll;PWD=secret;Database=GEOGETOWN
```

在服务器表查询中定义可制图的表

为服务器表指定的查询定义来自 DBMS 的数据的结果集，表示正在添加表中的数据。可以将相当复杂的查询公式化以执行强大的服务器端分析，用于在 MapXtreme 2005 中定义可制图的表。MapXtreme 2005 应用程序在内部使用该查询以访问数据。

MapXtreme 2005 基于访问地图中数据的查询生成几个内部查询以及基于选择 / 键的查询。从中选择几何体列的表必须注册到服务器上的 MapInfo MapCatalog 中。MapXtreme 2005 需要这个操作以获取某些关于几何体列的元数据，例如，坐标系、空间类型和默认样式。

要根据查询顺序定义可制图的表，则该查询必须包含几何体列和键列。有时候，对于有关小数据集更复杂的查询（其中空间索引或空间谓词将导致该查询失败），可以指定 TableInfoServer.MbrSearch=false 以启用要映射的结果。

几何体列

如果未指定 MapXtreme 2005 应用程序可以识别的几何体列，则将打开该表，但不能将它添加到地图中（该表是不可绘制的）。MapXtreme 2005 通过在 MapCatalog 中查找表的几何体列并检查该列的结果集数据类型来确定几何体列。通常可以通过伪列名称“OBJECT”来引用几何体列，或者可以使用几何体的特定列名来引用它。需要使用这种形式来引用 X/Y 可制图的图层的几何体列。可以通过任何服务器支持的几何体函数 / 表达式来指定几何体列。

示例

```
Select OBJECT from rdbsdata  
Select sw_geometry from rdbsdata  
select sw_member, ST_Buffer(geometry, 66.0, 0.1) from rdbsdata  
    // a geometry function  
Select st_geometry(st_point(72.5, 42.5.) from rdbsdata  
    // a geometry constructor
```

Spatialware 函数示例：

```
select sw_member, ST_Buffer(sw_geometry, 66.0, 0.1) from rdbsdata  
select ST_Overlap(flood100.sw_geometry, lake.sw_geometry) from  
flood100, lake where ST_Overlaps(flood100.sw_geometry,  
lake.sw_geometry)
```

Oracle sdo_buffer 示例：

```
Select mi_prinx, mdsys.sdo_geom.sdo_buffer(geoloc, (select diminfo
from sdo_geom_metadata where table_name = 'ALINE'), 20) from aline
where prinx = 1
```

Oracle 构造函数示例：

```
Select 1 "mi_prinx",
mdsys.sdo_geometry(3,null,null,mdsys.sdo_elem_info_array(1,3,3),
mdsys.sdo_coordinate_array(-79.919909,40.553465,-71.060457,45.363657))
from dual
```

键列

键列必须在查询中返回以使它作为表打开。即，启用您的 MapXtreme 2005 应用程序标识结果集中的每一行以对图层执行添加底纹、选择和标注操作。

在绝大多数情况下，不需要在查询中指定键列。

为了惟一地引用结果集中的行，MapXtreme 2005 应用程序可以查找并决定要使用的最佳键列，然后可将它们添加到查询中（如果该查询不存在键列）。在绝大多数情况下，这是主键 / 惟一的索引。

对 Oracle Spatial 表，可以使用 MI_PRINX。

对于某些查询，MapXtreme 2005 应用程序可能无法标识该键。在有关视图或同义词的查询中会出现这种情况。视图或同义词必须出现在 MapInfo MapCatalog 中。在绝大多数情况下，它们还必须使用基础 Spatial 索引系统按需进行注册。由于 MapXtreme 2005 不能决定视图和同义词上面的键，所以必须提供一种机制来允许应用程序开发人员 / 查询编写者标识位于结果表中的键列。该键列必须是单一列并且在结果集中必须是与其它列不同的值。要标识将用作键列的那个列，可以指定列别名 prinx 或 mi_prinx（例如，`select custid mi_prinx, ccustname, object from mycust`）。

示例

```
Select customer_id mi_prinx, obj, from customer_view
```

列别名“mi_prinx”用于标识和使用作为该表的键列的 `customer_id` 列。另外，可以在 `create view` 语句中为需要的键列指定别名，从而自动为有关视图的任何查询标识键列。

示例

```
Create view customer_view as select customer_id mi_prinx, geloc from
customer
```

通常，如果 `prinx` 的列名或列别名，或 `mi_prinx` 在结果集中找到，则该列用作这个表的键列。这使应用程序 / 查询编写者可以指定需要的键列。

访问属性数据

要使用所有可用数据列，则指定一个查询，例如 `Select * From tablename`。不需要指定 `*`（星号），可以特别指定想要使用的列。为了获取最佳性能，对您的查询进行了限制，使它仅检索需要的列。

在添加 DBMS 表时，出于性能原因，应该仅指定想要在应用程序中使用的查询中的列。这些列是空间列、键列（如果未指定，则将自动添加它们）以及想要标注或从其创建主题的列。可以为任何可制图的表使用伪列“OBJ”以引用包含空间数据的列。使用具有 X/Y 列的表上的 MapMarker MDIGEOADDRESS 列的表需要此操作。

可以使用任何服务器端表达式 / 函数来指定列。还要避免从实际应用程序的表中选择 `*`。

下面的代码示例使用 `TableInfoServer` 定义服务器表并使用此定义将图层添加到地图。现在可以根据表中的列标注或放置主题。

VB 示例：

```
Public Shared Sub MapInfo_Data_TableInfoServer(ByVal connection As _
    MConnection)
    ' Note: Do not specify any columns. These are determined
    ' dynamically from the query

    Dim ti As TableInfoServer = New TableInfoServer("Provinces")
    ti.ConnectionString = "SRVR=ontario;UID=mapx;PWD=mapx"
    ti.Query = "Select * From Provinces"
    ti.Toolkit = ServerToolkit.Oci

    ti.CacheSettings.CacheType = CacheOption.Off ' On is the default
    Dim tbl As Table = connection.Catalog.OpenTable(ti)
End Sub
```

性能问题

与数据库服务器建立连接可能需要花费几秒钟。这是一次性开销，在第一次打开表时发生。

地图显示速度取决于从服务器检索了多少数据。在某些情况下，从服务器显示地图明显比从本地文件显示地图要慢。显示速度还取决于 `MapXtreme 2005` 应用程序是否已缓存将要显示的地图图元。

运用缓存

了解如何在 MapXtreme 2005 中运用 Cache 来提高应用程序的性能。下面的章节说明了什么是缓存，如何在 MapXtreme 2005 对象模型中使用缓存，以及 TableInfoServer 对象的 CacheSettings 属性。

什么是缓存？

根据本地文件位置，应用程序可以从远程数据库访问 MapXtreme 2005 图元。为了避免每次对地图进行动作时都要从数据库中读取这些记录，MapXtreme 2005 可以把这些记录临时存储在缓存中。这可以限制应用程序和远程数据库之间的调用次数。可对服务器表中的记录（即绘图、主题、标注等）进行缓存以提高应用程序性能。在读取服务器表数据读取并将其绘制到 Map 窗口时，可在内部对其进行缓存。所有后续重绘都从缓存中读取，而不是进入服务器数据库获取同样的数据。缓存能够显著提高重绘性能。

开发人员可以使用若干设置来定制缓存的使用。通过为 TableInfoServer 对象的 CacheSettings 属性指定值来添加服务器表并且该服务器表是默认的 ON 时，可以启用缓存。

缓存如何工作

对于每个缓存的记录，每个属性数据值都存储在内存中，每个图元对象都存储在磁盘上的临时 Rtree 文件中。对于记录很多和（或）记录很大（例如，属性数据每个记录的字节数）的表，进行缓存可能会使用很大的内存。如果应用程序试图缓存过多数据，则可能需要使用太多虚拟内存，这会降低性能。应用程序应有选择地决定如何利用缓存。MapXtreme 2005 提供了各种用于控制缓存的机制。

TableInfoServer 对象和 CacheSettings 属性

将表添加到地图时，默认启用缓存，但可能需要使用 TableInfo 对象的 CacheSettings 属性进行进一步配置。该属性有四个可能的值：ON、OFF、ALL 和 USER，对于 TableInfoServer，默认为 ON，对于其它 TableInfo 对象，默认为 OFF。

参数	描述
OFF	值 'Off' 表示表将根本不使用缓存。所有数据操作将直接进入数据库服务器。
ON	<p>缓存被启用，并且表自动执行基于地图视图（中心或缩放）的缓存。用户还可以通过缓存约束对象来控制缓存。</p> <p>缓存的表以最能改善标准地图操作的方式来维护记录缓存。缓存维护至少包含显示在每个 Map（该表位于其中且可见）当前窗口中的所有记录。一旦初始地图窗口已经缓存，平移和缩放操作（完全位于缓存的初始扩展内）就访问缓存的记录且不必查询数据库。如果平移或缩放操作位于缓存区域外，则表将新的地图窗口 MBR（视图）添加到缓存，然后从数据库服务器获得缺少的记录并将这些记录添加到缓存。开始时不丢弃旧的地图视图，相反还维护先前地图视图的内部历史记录。要避免缓存增长过大，可以在表的缓存上设置一些控制来决定何时丢弃旧的缓冲视图（地图窗口 MBR 区域）。这些控件是 CacheParameters 对象的属性，可在最初打开表时进行设置。这允许开发人员做出一些限制，例如缓存所用内存或磁盘空间的最大容量，在历史记录中维护先前地图窗口视图的最大数目，在缓存中维护记录的最大数目，和（或）缓存历史记录中允许保留的旧地图窗口视图的最长时间。可以单独或组合使用这些限制以适应应用程序所需提供的最优缓存管理。</p> <p>ON 是 TableInfoServer 的 CacheParameter 设置的默认设置。对于其它 TableInfo 数据源，默认设置为 OFF。例如，默认情况下不缓存 TAB 文件。</p>
USER	<p>LayerInfo CACHE 参数的值 USER 表示应用程序创建了缓存，但只有应用程序开发人员指定的那些记录放入缓存中。这些机制可用于指定要将哪些记录放入缓存，记录类型包括 BoundConstraint、FeaturesConstraint 和 AllFeaturesConstraint 对象。约束一词意味着这些对象正在约束要包括特定记录的缓存。BoundsConstraint 对象可用于将全部记录放入 MBR 图元与 MBR 约束相交的缓存中。</p> <p>FeaturesConstraint 对象可用于将特定记录添加至缓存。例如，如果正在执行一个分析，它涉及多个步骤和（或）对图元的 Feature 和 RowValues 的读取，可能在从 Layer.Search、Layer.SearchWithinDistance 等返回的一组图元上进行，那么在分析期间将这些记录放入本地缓存并在分析完成时删除就非常有用。FeaturesConstraint 提供了这个功能。如果应用程序将要执行可能击中每个记录的密集分析操作，则可能需要这个功能临时缓存用于图层的整个数据集。通过使用 AllFeaturesConstraint 可以实现此功能。这些缓存约束对象还可在将缓存设置为 ON 时使用。在这种情况下，它们可以将记录添加到缓存，但该缓存对先前地图窗口视图的缓存历史无效。这些缓存约束还可在将缓存设置为 OFF 或 ALL（缓存不起作用）时使用。</p> <p>注： 该约束对象对非服务器表无效。</p>
ALL	整个表被缓存。使用该选项，表的数据只需从服务器检索一次，以后就可以从本地进行访问。要刷新缓存中的数据，请对该表使用 Refresh 方法。

如果试图缓存过多数据或过多表，则会强制使用虚拟内存，这样可能会降低性能。

MapInfo MapCatalog

为了显示地图上的数据，MapXtreme 2005 应用程序需要访问特殊表，也称为 MapInfo Map Catalog。每个数据库必须创建一个目录才能将数据库中的任何表作为 MapXtreme 2005 应用程序中的地图图层进行查看。Map Catalog 必须在每个想要从数据库进行访问的可制图的表中包括关于空间列的信息。MapInfo EasyLoader 实用程序在将表上载到数据库时自动在 MapInfo Map Catalog 中插入适当的行。

应用程序可以使用服务器上已存在的 Map Catalog。（各种 MapInfo 客户端应用程序可共享同一个目录）。如果服务器上没有 Map Catalog，则需要创建一个。MapXtreme 2005 支持将各个图元的样式信息存储在远程数据库中。

将空间数据加载到 DBMS

如果空间数据是 MapInfo 表形式的数据，则可以将那些数据导入到 DBMS。

要将数据添加到 SQL Server SpatialWare、Informix IDS SpatialWare 和 Oracle Spatial，请使用 MapInfo EasyLoader，它与 MapInfo Professional 一起发布，可从以下网址获得：

<http://www.mapinfo.com>。如果服务器上没有任何地图目录，则 EasyLoader 实用程序将在上载表时自动创建地图目录。

手动创建 MapInfo MapCatalog

如果您不是 MapInfo Professional 或 EasyLoader 用户，则需要按如下所述手动创建 MapCatalog（或由数据库管理员创建 MapCatalog）。每个服务器 / 数据库只能创建一次 MapCatalog。

1. 在可制图的表所属的特定数据库中创建用户 MAPINFO。
2. 在该数据库中创建表 MAPINFO_MAPCATALOG。Create Table 语句需要等价于下列 SQL Create Table 语句：

```
Create Table MAPINFO_MAPCATALOG (
  SPATIALTYPE Float,
  TABLENAME Char(32),
  OWNERNAME Char(32),
  SPATIALCOLUMN Char(32),
  DB_X_LL Float,
  DB_Y_LL Float,
  DB_X_UR Float,
  DB_Y_UR Float,
  COORDINATESYSTEM Char(254),
  SYMBOL Char(254),
  XCOLUMNNAME Char(32),
  YCOLUMNNAME Char(32),
  RENDITIONTYPE INTEGER,
```

```
RENDITIONCOLUMN CHAR(32),  
RENDITIONTABLE CHAR(32)  
NUMBER_ROWS INTEGER  
)
```

注： 表的结构与这个语句完全相同十分重要。可以进行的惟一替换是支持 varchar 或文本数据类型的数据库，这些数据类型可以替换为 Char 数据类型。

3. 在 TABLENAME 和 OWNERNAME 上创建惟一的索引，因此对于每个所有者只有一个表可以是可制图的。

```
create unique index mapcat_i1  
on mapinfo.mapinfo_catalog (OwnerName,TableName)
```

4. 对于 MAPINFO_MAPCATALOG 的 Grant Select、Update、Insert 和 Delete 特权。这允许用户使表成为可制图的。

```
grant select, insert, update, delete on mapinfo.mapinfo_mapcatalog  
to public
```

将行添加到 Map Catalog

对于每个想要从应用程序进行访问的空间表，需要将一行添加到 MAPINFO_MAPCATALOG 表。如果没有使用 MapInfo Professional 来管理 Map Catalog，则必须手动将行添加到 MAPINFO_MAPCATALOG 表。下表说明了每个列的语法和含义：

列名	要指定的值	示例
SPATIALTYPE	MapInfo 空间对象格式 1: 在 X/Y 列索引的 Point 图层 具有 micode (已串行化的 quadtree 键) 4: X/Y 列中的 Point 图层 5.x: SpatialWare for Oracle 6.x: Ingres SOL - 不受支持 7.x: Sybase SOL - 不受支持 8.x: Oracle SDO 版本 2 - 不受支持 9.x: MapInfo Geocoding DataBlade SpatialWare 点模块 10.x: MapInfo Geocoding DataBlade XY 模块 11.x: SpatialWare IDS/UDO datablade 13.x: Oracle Spatial 14.x: SpatialWare for Microsoft SQL Server 空间对象类型 x.0: 仅允许点 x.1: 仅允许直线 x.2: 仅允许区域 x.3: 所有受支持的类型 注： 该列说明了用来存储和索引数据的“空间对象格式”以及在该列中受支持和不受支持的“空间对象”类型。小数点左边的数字是“空间对象格式”。小数点右边的数字表示了可存储在该列的“空间对象类型”的类型。	5.0 = Oracle 5.1 5.2 5.3
TABLENAME	表的名称。	STATES
OWNERNAME	表的所有者名称。	BOB

列名	要指定的值	示例
SPATIALCOLUMN	包含空间图元的列的名称（如果有）： SW_Geometry (使用 SpatialWare Type/IDS/UDO 可制图) NO_COLUMN (使用 X-Y 可制图) MI_SQL_MICODE (使用 MI Code 可制图) 或 IDS/UDO 的名称，或 ST_SPATIAL 数据类型 Oracle 列。 Oracle SDO_Geometry 列的名称。	SW_Geometry
DB_X_LL	图层的最小边界矩形的左下角的 X 坐标，使用 由 COORDINATESYSTEM 指示的单位（详情如 下）。	-360
DB_Y_LL	左下边界上的 Y 值。	-90
DB_X_UR	右上边界的 X 值。	360
DB_Y_UR	右上边界的 Y 值。	90
COORDINATESYSTEM	表示 MapInfo CoordSys 子句（但一开始不具有 关键字 CoordSys）的字符串，该字符串指定 地图投影、坐标单位等。对于简单经度 / 纬度地 图，指定 Earth Projection 1, 0。	Earth Projection 1, 0
SYMBOL	MapInfo Symbol 子句（如果图层仅包含点）， 或 Symbol 子句，它后跟指示线性图元样式的 Pen 子句，而该 Pen 子句后跟指示区域边界样式的 其它 Pen 子句，最后又后跟 Brush 子句。	Symbol(35,0,12) Pen(1,2,0) Pen(1,2,0) Brush(2,255,255)
XCOLUMNNAME	对于 X/Y 可制图的表，指定包含 X 坐标的列的 名称。如果没有这样的列（即，如果该表使用 单一空间列而不使用 X-Y 列对），则指定 NO_COLUMN 或保留为空。	NO_COLUMN
YCOLUMNNAME	对于 X/Y 可制图的表，指定包含 Y 坐标的列的 名称，或指定 NO_COLUMN	NO_COLUMN
RENDITIONTYPE	这指示对象样式信息的应用方式。 0 - 指示所有对象将具有在要应用的 Map.Catalog 的符号字段中指定的样式。 1 - 指示该表已生成包含字符串的独立列，该字 符列表示该表中每个对象的样式信息（即，每 个对象都可以具有自己的样式）。	0 或 1

列名	要指定的值	示例
RENDITIONCOLUMN	如果 RENDITIONTYPE 为 1，则该字段存储包含样式信息的空间表（由 TABLENAME 标识）中的列名称。自动将该列添加到任何针对表的查询，并在更新该对象时对该列进行维护（更新）。用户不应在查询中指定此列，因为在相交或更新语句时可能会引起问题。将此列包括在 select 子句（不包括通配符“*”）中的查询可以通过 Dataset 对象访问值。在样式列中具有 NULL 值的行将具有样式，该样式来自要应用到该对象的 MapCatalog 的 SYMBOL 字段。	MI_SYMBOL
RENDITIONTABLE	当前未使用，当保留以供将来使用，该列必须存在以便 MapXtreme 2005 正确标识和应用记录级样式。	NULL

Symbol、Pen、Brush 子句语法

如果手动创建 MAPINFO_MAPCATALOG 表以提供对远程空间数据库的支持，则将需要指定符号样式，以及可能的直线和填充样式。

指定点样式

使用 Symbol 子句来指定点样式。有如下三种类型的 Symbol 子句：一种用于指定 MapInfo 3.0 样式符号，一种用于指定 TrueType 字体符号，一种用于指定位图符号。

Symbol 语法	示例
Symbol(shape, color, size) 或 Symbol(shape,color,size,font,fontstyle,rotation) 或 Symbol(bitmapname,color,size,customstyle)	Symbol (35, 0, 12) Symbol (64, 255, 12, "MapInfo Weather", 17, 0) Symbol ("sign.bmp", 255, 18, 0)

指定直线样式

使用 Pen 子句来指定直线样式。在 Map Catalog 中，可能需要指定两个 pen 子句：一个用于指定线性图元的外观，另一个用于指定区域边界的外观。

Pen 语法	示例
Pen(thickness, pattern, color)	Pen(1, 2, 0)

指定填充样式

使用 Brush 子句来指定闭合图元（区域）的样式。

Brush 语法	示例
Brush(pattern,color,backgroundcolor)	Brush(2, 255, 65535)

每记录样式

每记录样式支持将图元带到空间数据库实现，该实现已长期在 MapInfo TAB 文件中可用。特别是，它允许单一表中的每个几何体具有自己的样式。例如，Oracle Spatial 中的单个“公共机构”表可以具有学校、市政大厅、图书馆和警察局，并且每个点类型将使用自己的符号表示（即学校符号用于所有学校）。类似的，SpatialWare SQL Server 中的单一道路表可能具有不同的道路类型，如显示为单像素黑色直线的街道、显示为双像素红色直线的辅道以及显示为红色平行直线的十字路口。每记录样式可以根据下表，通过 MapCatalog 中的项为表激活：

列	数据类型	描述
RENDITIONTYPE	INTEGER	表的每记录转换类型的数值指示器。有效值为： 0: 对于指示的表，没有每记录样式生效。使用该表的默认样式前将要读取 / 更新对象。 1: 指示的表使用每记录样式。包含该样式的表中的列记录在 RENDITIONCOLUMN 中，并且它包含 MapBasic 字符串（例如，当前用在 Map Catalog 的 SYMBOL 列中的同一格式）。
RENDITIONCOLUMN	VARCHAR(32)	该列包含位于指示的表中的列名，该表保存对象样式值（用于转换类型 1）。
RENDITIONTABLE	VARCHAR(32)	目前不使用这个列，但是，在应用程序正确检测到新图元之前它必须存在。已经计划对功能进一步增强：可在独立的表中管理样式并使由 RenditionColumn 标识的该列中的值包含可用于查找样式的键，该样式位于由此属性标识的表中。该功能尚未实现。

注： 如果没有这些列，则对象样式将继续按先前样式工作（例如，将表的默认样式应用到全部对象）。

已知限制 / 要求

LegacyText 对象具有它们自己用来显示样式的方式，与使用 MI_Style 列分开。因此，任何形式的文本对象都需要单独对待，它们彼此之间互不相同。嵌入用于任何文本对象的样式并将 NULL 值插入到类型列中。

故障诊断

如果在使用 SpatialWare 或 Oracle 应用程序时遇到问题，则使用下表来帮助分析和解决问题。

问题说明	可能原因	解决方案
表是不可匹配的。	尝试对 SpatialWare 图层进行数据绑定。	SpatialWare 图层目前不支持 AddColumns。
使用指定的索引没有找到任何对象。	针对不存在的表执行查询。	检查表名是否正确并且大小写正确。还有，该表需要是可制图的。
	在空间查询的结果中未包含任何空间对象。	使用 EasyLoader Upload 实用程序使该表成为可制图的表。
	针对非空间表执行查询。	检查查询以获取可能的语法错误。还要确保查询结果包括在 MapInfo_MapCatalog 的空间列中指定的字段。
地图显示为具有不正确的缩放级别。例如，地图可能缩的太小，不能标识任何地理位置。	DBMS 图层的 MBR 由 MapInfo_MapCatalog 表确定。该表以不同于所需输出的缩放级别在 MapCatalog 结果中进行扩展。	使用 MapInfo Professional MBX 工具 MISETMBR.MBX 编辑 MapInfo_MapCatalog 中的扩展 (DB_X_LL、DB_X_UR、DB_Y_LL 和 DB_Y_UR)。

12

将地图绘制功能增加到 应用程序

使用 MapInfo.Mapping 命名空间可以将地图绘制功能添加到应用程序。本章解释了如何使用 Mapping 命名空间增强地图绘制应用程序。

本章内容：

- ◆ *Mapping* 命名空间简介 216
- ◆ 基本地图绘制类 216
- ◆ 图层 219
- ◆ 代码示例：动画图层 221
- ◆ 标注 223
- ◆ 修饰 227
- ◆ 图元样式修饰符 229
- ◆ 打印地图 230

Mapping 命名空间简介

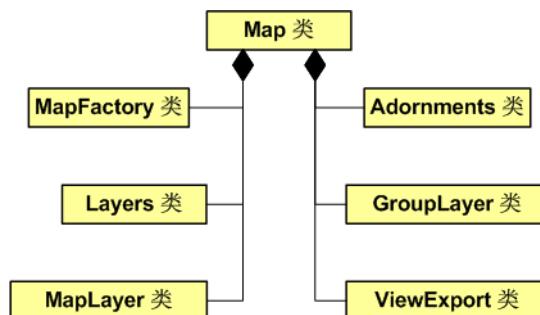
MapInfo.Mapping 命名空间包含了用于创建、显示和导出地图、图层、修饰符、主题、图例和标注的类、接口和枚举。

Map 类是桌面应用程序中地图的顶层对象。每个 Map 对象正好包含一个地图。每个 Map 具有一个 Adornment 集合和一个 Layer 集合。开发 web 应用程序时，Map 对象附加到 MapExport 对象，以便将该地图图像导出到位图或数据流。

如第 41 页第 3 章中的图层中所述，从根本上讲，地图是一组互相层叠的图层。使用 Mapping 命名空间类，应用程序可按照需要设计用于操控这些图层。

基本地图绘制类

本节讨论了如何在 MapInfo.Mapping 命名空间中使用基本地图类。下图说明了 Map 层次的 UML 表示。



MapExport

MapExport 类用于将 Map 导出到图像。MapExport 支持多种不同的图像格式：BMP（默认）、WBMP、WMF、EMF、GIF、J2K、JPG、PNG、PSD、TIFF、TIFFCymk 和 TIFFLzw。该类的属性指定了图像的各个方面，例如 BorderPen、ExportSize、Format 等等。

使用 MapExport 的示例

如果正使用 MapControl，则必须在导出地图前将其克隆，具体如下示例所示：

VB 示例：

```
Public Shared Sub MapInfo_Mapping_MapExport(ByVal mapControl1 As  
MapControl)  
'must clone since map is coming from mapcontrol and is linked to it via  
the HWND  
Dim NewMap As Map = CType(mapControl1.Map.Clone(), Map)  
Dim exportObject As MapExport = New MapExport(NewMap)  
    exportObject.ExportSize = New MapInfo.Mapping.ExportSize(2931,  
4104)  
    exportObject.Format = ExportFormat.Gif  
    exportObject.Export("C:\Temp\ExportImage.gif")  
End Sub
```

Map

Map 类包含了可以在地图中放入的任何内容。对于桌面应用程序，Map 对象放置在 MapControl 对象中；对于 web 应用程序，Map 对象附加到 MapExport 对象。

Map 具有以下属性：

- Bounds
- Center
- Zoom
- Scale
- Size
- Rotation
- DisplayTransform
- DisplayCoordSys

每个 Map 都具有 Layers 集合，用于保存组成地图的所有图层（请参阅第 220 页中的 [Layers](#)），还具有一个包含所有地图修饰的 Adornment 集合。Adornments 包括 Legends、Titles 和 Scalebars（请参阅第 245 页第 13 章中的图例概述）。

MapFactory

MapFactory 类包含了用于从 Geosets、工作空间和表的列表创建地图的对象。MapFactory 的功能也类似于在特殊会话中创建的所有地图的集合容器，而且 MapFactory 可以跟踪特殊会话中创建的所有地图的集合。

使用 MapFactory 的示例

该示例使用 MapFactory 的 CreateEmptyMap 方法创建空地图，地图大小为 300 乘 300 像素。

```
' Create a new map
Dim map As Map = Session.Current.MapFactory.CreateEmptyMap(New
Size(300, 300))
```

MapLoader

MapLoader 类提供了一种机制，用于从 Geoset 文件、XML Workspace 文件或 TAB 文件加载地图的图层。对于每种要加载的地图类型，都有可用于加载地图的 MapLoader 的子类。这些子类包括 MapGeosetLoader、MapWorkspaceLoader 和 MapTableLoader。

VB 示例：

```
Public Shared Sub MapInfo_Mapping_MapLoad()
    ' Create an empty map
    Dim map As Map =
        MapInfo.Engine.Session.Current.MapFactory.CreateEmptyMap(New_
        Size(400, 300))

    ' Create a maploader. Make sure that Session.Current.TableSearchPath
    points to the folder with the tables in it
    Dim tl As MapTableLoader = New MapTableLoader("ocean.tab", "usa.tab",
        "mexico.tab", "us_hiway.tab")

    ' Load tables into a map
    map.Load(tl)
End Sub
```

MapViewList、MapView

这些类包含的对象有助于维护 Map 的上一个和下一个视图列表。可以使用 MapViewList 类作为一种简便方法来浏览地图视图的堆栈，使用提供每个地图视图细节（Name、Center 和 Zoom）的 MapView 类显示这些视图。

在 ..\MapInfo\MapXtreme\6.x\Samples\Desktop\Features 中，MapXtreme 2005 提供了 MapViewList 的 VB 示例。

MapControl

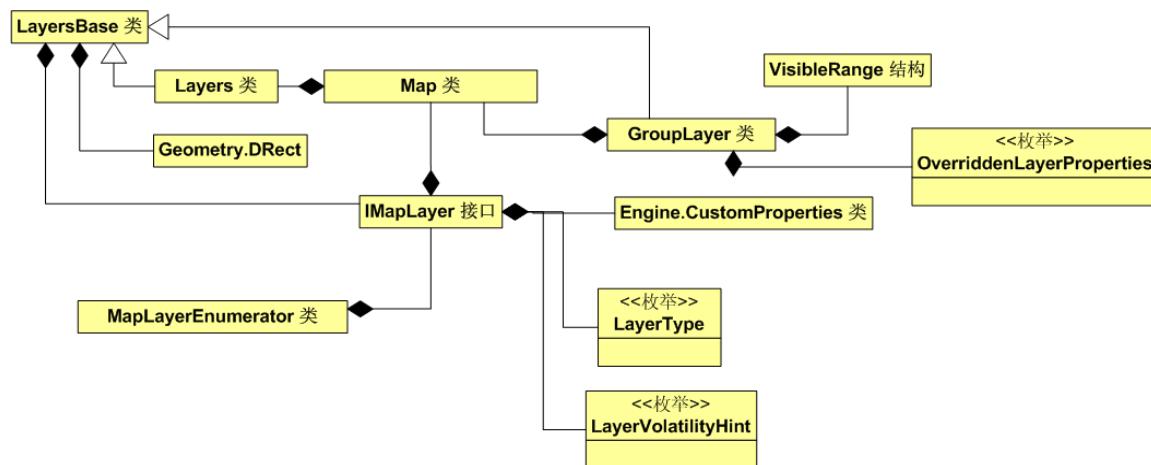
MapControl 类包含了允许用户在屏幕上查看地图的对象。将 MapControl 添加到 Windows Form 来查看地图。MapControl 还保存了 MapTools 集合。MapControl 的版本也存在于 web 应用程序。

用于桌面应用程序的 MapControl 类位于 MapInfo.Windows.Controls 命名空间中。请参阅[第 219 页第 12 章中的 MapControl](#)。

用于 web 应用程序的 MapControl 类位于 MapInfo.WebControls 命名空间中。请参阅[第 94 页第 6 章中的 Web 控件的体系结构](#)。

图层

下一节讨论了 Layers 对象和类。下图是 Layers 层次的 UML 表示。



FeatureLayer

FeatureLayer 是从 Table 显示 Features 的图层。例如表示世界国家的区域对象的图层是 FeatureLayer。FeatureLayers 可以是本地的 .TAB 数据、远程 RDB、无缝或光栅数据。

Layers

每个地图包含了由 Layers 类表示的 FeatureLayers 的集合。集合中的顺序是绘制图层的顺序。集合类的方法包括 Add、Insert、Move 和 Delete。Layers 集合从 LayersBase 类派生。通过图层枚举的最好方式是使用 Layer 筛选器。

要更改图层，图层首先需要是可编辑的。通过更改其 LayerControl 中的设置或通过编程方式更改其EditMode 属性，可以使图层成为可编辑的。图层可以编辑后，该图层中的图元就可以移动、调整大小或删除。

注： 对特殊图层所做的任何编辑会立即生效，因此选择可编辑图层中的图元时要小心操作。

要实现筛选器，请使用 MapLayerFilterFactory 类。该类允许创建源于一组常用筛选器的筛选器，例如图层类型或所有可见的图层。要创建自己的筛选器，请编写实现 IMapLayerFilter 接口的类。

MapLayer

MapLayer 类是任何图层的基类。该类实现 IMapLayer 接口。属性包括 Enabled、VisibleZoomRange、Name 和 Alias。使用该类访问一般的图层属性。

UserDrawLayer

UserDrawLayer 是抽象类，允许重载绘制方法绘制自己的图层。它提供了用于将自定义对象增加到 MapControl 顶部的有效方法，即使用 Windows 绘制方法，而不是使用地图坐标创建新图元并将其实际增加到地图中。

ObjectThemeLayer

ObjectThemeLayer 类包含了三种不同主题中的一个主题（饼图、条形图和分级符号）。该图层的行为与其它图层类似，且可以分组，还具有可视性集合等等。

GroupLayer

这是表示一组图层的类，该组图层可以一起移动和开/关。GroupLayer 是由 IMapLayer 组成的 LayersBase 集合。该对象也支持 AnimationLayers 功能。

如果图层是在 VolatilityHint 等于 Animate 的组中，则只有该组内的那些图层需要重绘。如果 Layer 中的 VolatilityHint 等于 CacheIfPossible 或 Normal，则需要重新绘制所有图层。

图层筛选器

IMapLayersFilter 和 IMapLayersFilterFactory 接口提供了对图层枚举的支持。

IVisibilityConstraint

IVisibilityConstraint 是确定特殊的对象是否可见的接口。该接口由大量类型实现，包括所有 Layer 类型、LabelModifiers、FeatureStyleModifiers 和 Themes。

代码示例：动画图层

以下 *MapInfo Knowledgebase* VB 示例说明了如何设置图层的动画。

```
Private Sub btnInitializeObjects_Click(ByVal sender As System.Object,
 ByVal As_
 System.EventArgs) Handles btnInitializeObjects.Click
 Dim Cat As Catalog = MapInfo.Engine.Session.Current.Catalog

 'Create Temp layer
 Dim tblInfoTemp As New TableInfoMemTable("Animation")
 Dim tblTemp As Table = Cat.GetTable("Animation")
 If IsNothing(tblTemp) = False Then 'Table exists close it
 Cat.CloseTable("Animation")
 End If
 tblInfoTemp.Columns.Add(ColumnFactory.CreateFeatureGeometryColumn(Map_
 Control1.Map.GetDisplayCoordSys()))
 tblInfoTemp.Columns.Add(ColumnFactory.CreateStyleColumn())
 tblInfoTemp.Columns.Add(ColumnFactory.CreateStringColumn("Name", 40))
 tblInfoTemp.Columns.Add(ColumnFactory.CreateStringColumn("Dept", 15))
 tblInfoTemp.Columns.Add(ColumnFactory.CreateIntColumn("Level"))

 tblTemp = Cat.CreateTable(tblInfoTemp)

 Dim lyr As New FeatureLayer(tblTemp)
 MapControl1.Map.Layers.Add(lyr)

 'Create Points
 Dim pt As FeatureGeometry = New Point(lyr.CoordSys, New DPoint(-76,
 42))
 Dim cs As New CompositeStyle(New SimpleVectorPointStyle(37,
 System.Drawing.Color.Red, 10))
 Dim ftr As New Feature(tblTemp.TableInfo.Columns)
 ftr.Geometry = pt
 ftr.Style = cs
 ftr.Item("Name") = "Kelly"
```

```
ftr.Item("Dept") = "Sales"
ftr.Item("Level") = 3
tblTemp.InsertFeature(ftr)

Dim pt2 As FeatureGeometry = New Point(lyr.CoordSys, New DPoint(-119,
34))
Dim cs2 As New CompositeStyle(New SimpleVectorPointStyle(44,
System.Drawing.Color.Purple, 10))
Dim ftr2 As New Feature(tblTemp.TableInfo.Columns)
ftr2.Geometry = pt2
ftr2.Style = cs2
ftr2.Item("Name") = "Greg"
ftr2.Item("Dept") = "Marketing"
ftr2.Item("Level") = 2
tblTemp.InsertFeature(ftr2)
End Sub

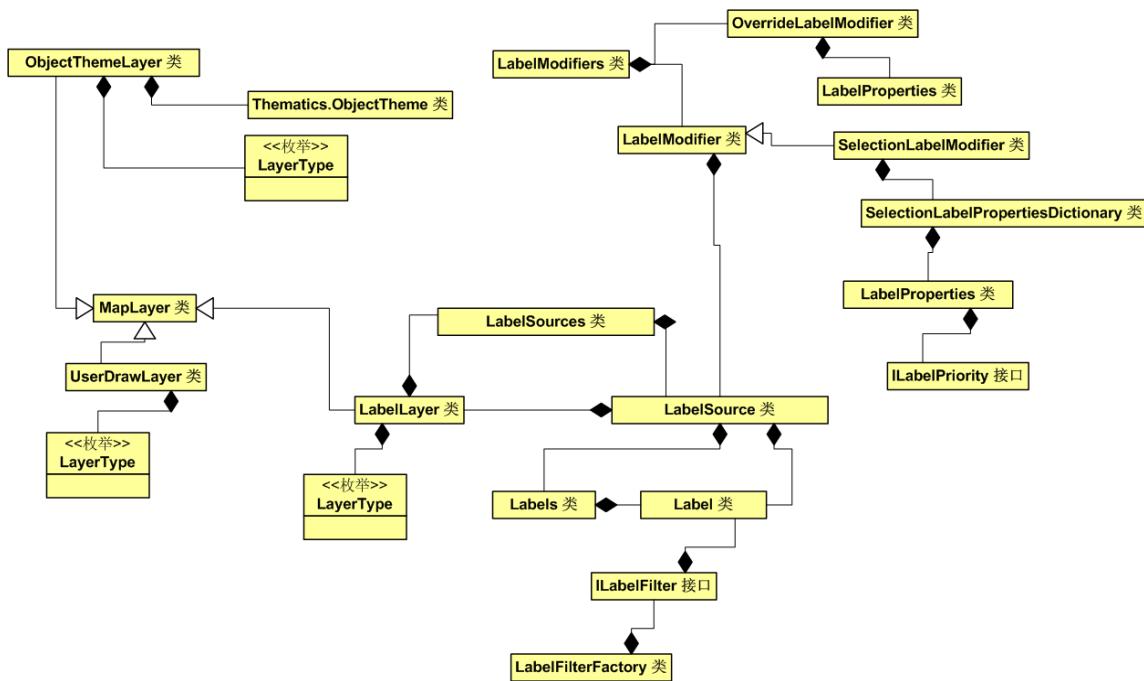
Private Sub Timer1_Tick(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Timer1.Tick
Dim cat As Catalog = MapInfo.Engine.Session.Current.Catalog
Dim tbl As Table = cat.GetTable("Animation")
If IsNothing(tbl) = False Then
'Update the position of the points
Dim si As SearchInfo =
MapInfo.Data.SearchInfoFactory.SearchWhere("Name = _
'Kelly'")
Dim ftr As Feature = cat.SearchForFeature(tbl, si)
Dim si2 As SearchInfo =
MapInfo.Data.SearchInfoFactory.SearchWhere("Name = _
'Greg'")
Dim ftr2 As Feature = cat.SearchForFeature(tbl, si2)

If TimeOfDay.Now.Second Mod 4 = 0 Then
ftr.Geometry.GeometryEditor.OffsetByXY(-5, -25, DistanceUnit.Mile, _
DistanceType.Spherical)
ftr2.Geometry.GeometryEditor.OffsetByXY(0, 25, DistanceUnit.Mile, _
DistanceType.Spherical)
Else
ftr.Geometry.GeometryEditor.OffsetByXY(-10, 0, DistanceUnit.Mile, _
DistanceType.Spherical)
ftr2.Geometry.GeometryEditor.OffsetByXY(10, 5, DistanceUnit.Mile, _
DistanceType.Spherical)
End If
ftr.Geometry.EditingComplete()
ftr2.Geometry.EditingComplete()
ftr.Update()
ftr2.Update()

End If
End Sub
```

标注

下一节讨论了 Labels 对象和类。下图是 Labels 层次的 UML 表示。



LabelLayer

LabelLayer 类允许分开对标注和图层进行排序。LabelLayer 是一种 MapLayer，且其行为也与 MapLayer 相同。这类似于 MapLayer 支持将 LabelLayer 放置在可以放置 MapLayer 的任何位置，允许相对于地图中的其它图层定位 Labels。每个 LabelLayer 由 LabelSources 组成且用作这些对象的集合。

LabelSource

根据数据源和定义如何标注该源的规则，LabelSource 类将图形化地理层显示为标注。要绘制 LabelSource，则将其添加到地图上的 LabelLayer。LabelLayer 提供了在 Map 内的位置，还提供了管理与其它 LabelSources 交互的规则。要使用 LabelSource 类，请指定要获得数据的位置的表 (MITable)，同时指定定义标注文本的表达式和其它相应的默认标注属性。

LabelModifier

LabelModifier 类用于修改用于生成标注的默认属性。LabelLayer 为其 Sources 集合中的每个 LabelSource 生成标注时，首先使用 DefaultLabelProperties 生成每个标注。然后使用 Modifiers 集合中的每个可见 LabelModifier。

ILabelSourceFilter

该接口允许在筛选特定规则的 LabelLayer 中通过 LabelSource 对象的集合进行枚举。此外也可以实现该接口来定义自己的筛选规则。

LabelProperties

该类具有诸如样式、位置等标注属性信息。该类支持稀疏标注属性的表示。使用 LabelModifier 仅修改标注属性的一部分时这会非常有用。该类也允许设置标注的优先权和改善的位置。

生成标注

LabelLayer 类在地图绘制或调用 LabelLayer.Refresh 方法时生成标注。需要考虑每个可视标注源。地图中可以有多个标注图层。

要为当前地图视图中源表的每个图元生成标注，LabelLayer 类执行以下操作：

1. 将 DefaultLabelProperties 属性用作属性集的开始属性，用于生成标注。
2. 调用 Modifiers 集合（如果有）中每个可视标注修饰符的 Modify 方法。这样每个修饰符都可以更改用于生成标注的标注属性。
3. 执行可视性约束检查，以决定是否保存标注。
 - a. 通过可视性约束与当前地图缩放 / 比例的比较，检查标注可视性。
 - b. 如果标注可视且在该标注上不允许重叠和重复，则使用其它现有标注检查重叠和重复。如发现有任何重叠或重复，则使用 [标注优先权](#) 决定保存哪个标注。
4. 如果可视性约束检查成功，则将标注增加到生成标注的集合。

注意，标注生成规则仅在每个标注图层中适用，不适用于整个地图。例如，如果将所有标注源（包含在地图的标注图层中）的 AllowOverlap 属性都设置为 False，则一个标注图层中的标注仍将与另一个标注图层中的标注重叠，因为标注的生成是在两个标注图层之间独立进行的。

如果地图还未根据当前地图视图绘制生成标注，请使用 LabelLayer.Refresh 方法。

访问生成的标注

可以通过 LabelSource.Labels 属性访问生成的标注。该集合表示位于当前地图视图的标注。它们不表示地图中所有标注。集合中项目的更改与地图视图的更改同步。

标注优先权

标注优先权决定在 AllowOverlap 或 AllowDuplicates 被设置为 False 时同一 LabelLayer 中生成哪些标注。注：正如生成标注中所述，每个标注图层都彼此相对独立，因此，其它图层中不同的重叠或重复设置不在此处显示。

标注与同一图层中另一个标注重叠，或为另一个标注的副本时，将比较两个标注的优先权以确定保存哪个标注。首先，该过程比较每个标注（或标注交互源）的优先权。将保存具有较高优先权的标注。

MapXtreme 2005 提供了两个用于控制标注显示的优先权等级：主优先权和次优先权。这允许将优先权分成组和子组。例如，可以让人口较多的大城市标注具有的优先权高于人口较少的小城镇。不过，也可以增加一个用于提高小城镇主优先权的修饰符，使小城镇的优先权有机会高于大城市。

Major 优先权设置为空（Visual Basic 中的 Nothing）时，使用的值是标注在 LabelLayer 中 LabelSource 位置的反向值。索引位置越高，优先权越低。例如，如果 LabelSource 是位于索引 3 且索引有 10 个 LabelSources（从 0 到 9 的索引），则 Major 优先权将为 6（基于总数的反向索引）。

Minor 优先权设置为空（Visual Basic 中的 Nothing）时，使用的值是基于被标注的 Feature 的 Key。使用键的数字表示的反向值，相对于 Table 中的行数。例如，如果 Table 中有 10 行，对于行 ID 为 7 的图元的标注，次优先权默认为 3 ($10 - 7 = 3$)。如果键不为数值，则次优先权默认为 0。

Major 优先权相等时，具有较高 Minor 优先权的标注保存。

要创建优先权，请使用生成数字值的表达式。例如，在 C# 中生成数字值的表达式，可以是数字类型或表达式的字段，例如字段值中首字母的 ASC 值：

```
"(1/Asc(Country))*1e6"
```

标注图层的可选择性

要控制标注图层的可选择性，请通过编程方式或通过作为可编辑复选框的 Workspace 管理器的标注来使用 SelectMapToolProperties.LabelsAreEditable 属性。

LayerHelper.SetSelectable 方法不影响特定图层类型（包括标注、WMS/WFS、光栅和组图层）的可选择性。

代码示例：创建 LabelLayer

以下示例说明了如何使用与 Labels 关联的类。

VB 示例：

```
' Open usa table using the data catalog
Dim table As Table = Session.Current.Catalog.OpenTable("usa.tab")

' Create a new map
Dim map As Map = Session.Current.MapFactory.CreateEmptyMap(New _
    Size(300, 300))

' Create a new feature layer that references the table and add it to
the map
Dim featureLayer As New FeatureLayer(table)
map.Layers.Add(featureLayer)

' Create a new label layer and add it to the map.
' Note that if you call MapInfo.Mapping.LayersBase.Add" method instead
of
' MapInfo.Mapping.LayersBase.Insert method it will automatically
position the
' label layer before the feature layer

Dim labelLayer As New MapInfo.Mapping.LabelLayer()
map.Layers.Insert(0, labelLayer)

' Create a new label source that references the table
Dim source As New MapInfo.Mapping.LabelSource(table)

' Change its caption expression to be a specific column from the table
' called "State_Name"
source.DefaultLabelProperties.Caption = "State_Name"

' Append the label source to the label layer so that it shows on the
map
labelLayer.Sources.Append(source)
```

修饰

Adornments 类是地图修饰的无序集合。修饰是 Legend、Title、Scalebar 或某些其它用户定义的对象。每个地图包含一个 Adornments 集合。每个修饰属于一个 AdornmentControl。

要创建自己的修饰，请从 IAdornment 接口和 AdornmentControl 抽象类派生一个类。

图例

图例用于与主题一起使用。有关主题的详细信息请参阅第 13 章：使用主题和图例。图例是使用 LegendFactory 类创建的。Legend 由一个或多个图例框架组成。每个框架不是 ThemeLegendFrame 就是 CartographicLegendFrame。通过从创建的 Legend 对象使用 LegendFrameFactory 类，可以创建两种 LegendFrames。对于图例及其框架的操控和定制，请使用 MapInfo.Mapping.Legends 命名空间中的类。该命名空间中的类包括：CartographicLegendFrame、ThemeLegendFrame、LegendFormat、LegendControl 及其它。

注意，Legend 的大小可以通过 Legend.Size 属性设置，但 LegendFrames 的大小无法设置。LegendFrame 的大小由它所包含的数据量控制。

VB 示例：

```
Private Sub Page_Load(ByVal sender As Object, ByVal e As _  
System.EventArgs)  
    LegendControl1.Map = MapControl1.Map  
    If Not IsPostBack Then  
        Dim normalLyr() As MapInfo.Mapping.LayerType = New _  
MapInfo.Mapping.LayerType(1) {}  
        normalLyr(0) = MapInfo.Mapping.LayerType.Normal  
        Dim filter As MapInfo.Mapping.IMapLayerFilter = _  
MapInfo.Mapping.MapLayerFilterFactory.FilterByLayerType(normalLyr)  
        Dim frame As MapInfo.Mapping.Legends.LegendFrame  
        Dim NewLegend As MapInfo.Mapping.Legends.Legend = _  
MapControl1.Map.Legends.CreateLegend(New System.Drawing.Size(5, 5))  
        NewLegend.Format.FrameAlignment = _  
MapInfo.Mapping.Legends.LegendFrameAlignment.Horizontal  
        Dim ftrLayer As MapInfo.Mapping.FeatureLayer  
        For Each ftrLayer In _  
MapControl1.Map.Layers.GetMapLayerEnumerator(filter)  
            frame = _  
MapInfo.Mapping.Legends.LegendFrameFactory.CreateCartographic_ _  
LegendFrame(ftrLayer)  
            NewLegend.Frames.Append(frame)  
        Next  
        LegendControl1.Legend = NewLegend  
    Else  
        LegendControl1.Legend = MapControl1.Map.Legends(0)  
    End If  
End Sub
```

ScaleBar 修饰

比例栏是用于度量地图距离的线性对象，它使用地图单位（如千米或英尺）。使用 ScaleBarAdornment 类创建 ScaleBarAdornmentControl，并将控件增加到地图。如果试图将 ScaleBarAdornment 增加到地图本身，则显示 ScaleBarAdornment，但不具有 ScaleBarAdornmentControl，ScaleBar 不会链接到地图本身。

VB 示例：

```
Public Shared Sub MapInfo_Mapping_ScaleBarAdornment(ByVal mapControl1 As _  
MapControl)  
    ' Create a scalebar  
    Dim sba As ScaleBarAdornment = New  
    ScaleBarAdornment(mapControl1.Map)  
  
    ' Position the scalebar at the lower right corner of map  
    Dim x As Integer = mapControl1.Map.Size.Width - sba.Size.Width  
    Dim y As Integer = mapControl1.Map.Size.Height - sba.Size.Height  
    sba.Location = New System.Drawing.Point(x, y)  
  
    ' Add the control to the map  
    Dim sbac As ScaleBarAdornmentControl = New  
    ScaleBarAdornmentControl(sba, _  
    mapControl1.Map)  
    mapControl1.AddAdornment(sba, sbac)  
End Sub
```

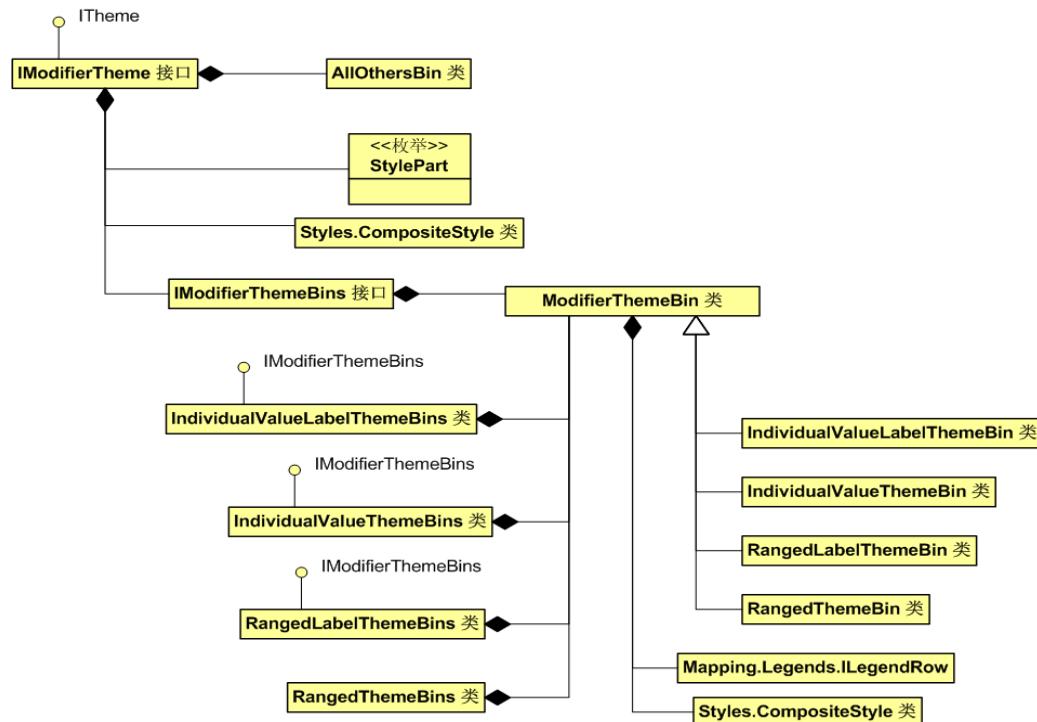
标题修饰

标题修饰是绘制在地图上的文本对象，表示地图标题或提供用于阐明地图上其它信息的文本。在 MapXtreme 2005 中，使用 TitleAdornment 类创建标题并将其增加到地图。

```
Public Shared Sub MapInfo_Mapping_TitleAdornment(ByVal mapControl1 As  
MapControl)  
    ' Create a Titlebar  
    Dim ta As New MapInfo.Mapping.TitleAdornment(New  
    System.Drawing.Size(100, 50), mapControl1.Map)  
    ta.Title = "This is a watermark"  
    mapControl1.Map.Adornments.Append(ta)  
End Sub
```

图元样式修饰符

绘制 Feature 的样式之前，图元样式修饰符允许用一种方式更改或修改 Feature 的样式。这些类使用 CompositeStyle 对象的稀疏属性方面仅修改您感兴趣的样式部分。Range、Individual Value 和 Dot Density 主题是样式修饰符。有关主题的详细信息请参阅 [第 13 章：使用主题和图例](#)。下图说明了 Modifier 和主题层次的 UML 表示。



FeatureStyleModifier

这是所有修饰符必须派生的抽象基类。IndividualValue、Ranged 和 DotDensity 主题全是 FeatureStyleModifier 对象。可以创建从 FeatureStyleModifier 派生的自己的类，还可以重载 `Modify()` 方法。

FeatureStyleModifiers

`FeatureStyleModifiers` 类是包含在每个 `FeatureLayer` 内的 `FeatureStyleModifier` 对象的有序集合。绘制图元的几何体之前按顺序调用集合中的每个修饰符。

FeatureOverrideStyleModifier

FeatureOverrideStyleModifier 是简单的 FeatureStyleModifier。该类提供了 Layer 级别上的样式重载功能。FeatureOverrideStyleModifier 具有组合样式且实现 IVisibilityConstraint。这类类似于 MapX 和 MapInfo Professional 中的功能。

打印地图

应用程序创建后，您也许想要用户能够打印生成的地图。提供的 MapInfo.Printing 命名空间可用于简化应用程序的功能（包括打印）。该命名空间中的类根据 .NET Framework 类生成并用于打印，因此，需要的构造类似于任何其它 Windows 应用程序。除了常规打印外，我们还提供了多项附加功能，用于优化所创建地图的打印版本。

有关从 MapXtreme 应用程序打印的详细信息，请参阅[附录 H: 从 *MapXtreme* 应用程序进行打印](#)。

使用主题和图例

MapXtreme 2005 提供了多个用于将主题和图例添加到地图的选项。以下章节说明了可用的不同类型主题和图例并解释了如何使用它们。

本章内容：

- ◆ 主题概述 232
- ◆ *GraduatedSymbolTheme* 234
- ◆ *PieTheme* 235
- ◆ *BarTheme* 237
- ◆ *RangedTheme* 238
- ◆ *RangedLabelTheme* 240
- ◆ *IndividualValueTheme* 241
- ◆ *IndividualValueLabelTheme* 242
- ◆ *DotDensityTheme* 243
- ◆ 图例概述 245

主题概述

主题地图绘制用于显示数据的趋势，而这种趋势在表格数据中难于表现。主题通常使用某块或某几块数据。可以使用来自数据源（例如，本地 MapInfo 表）的数据将地图以不同的主题加上底纹。例如，可以根据每个州的平均温度为“美国”地图加上底纹。在看见红色时，就可以知道此地天气炎热（高温），但是在看见蓝色时，就知道此地天气寒冷（低温）。

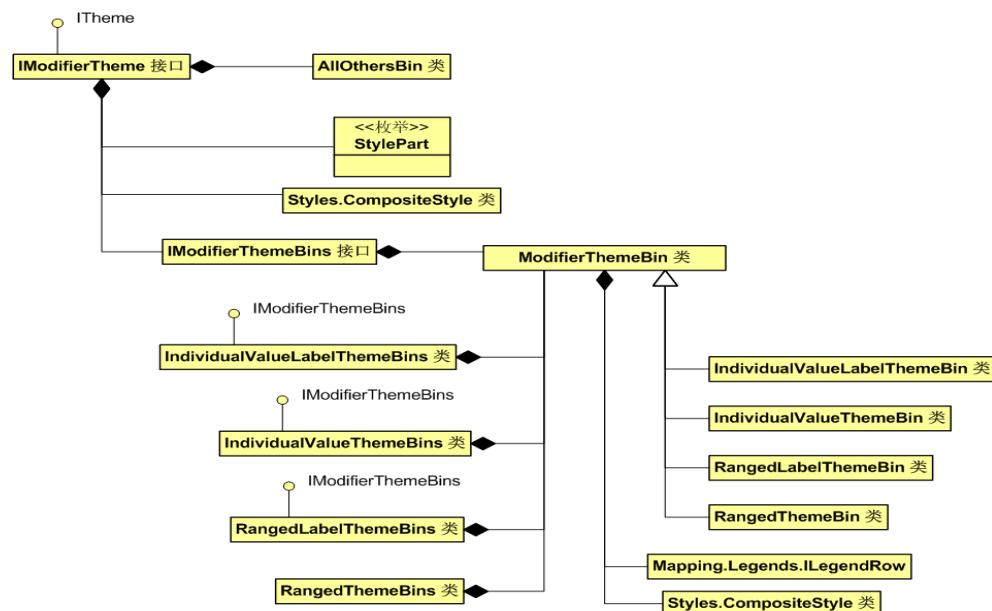
主题使用颜色、填充图案或符号底纹来表示数据。主题地图显示数据有多种用法。可以通过将这些颜色、图案或符号指定给地图对象（根据数据中的特定值）来创建不同的主题地图。

Mapping.Thematics 命名空间

MapInfo.Mapping.Thematics 命名空间包含的类以 Feature 图层上的样式重载和 Object 主题实现来主题。对象主题添加新的图层时修饰符主题更改样式。所有主题都实现 ITheme 接口。

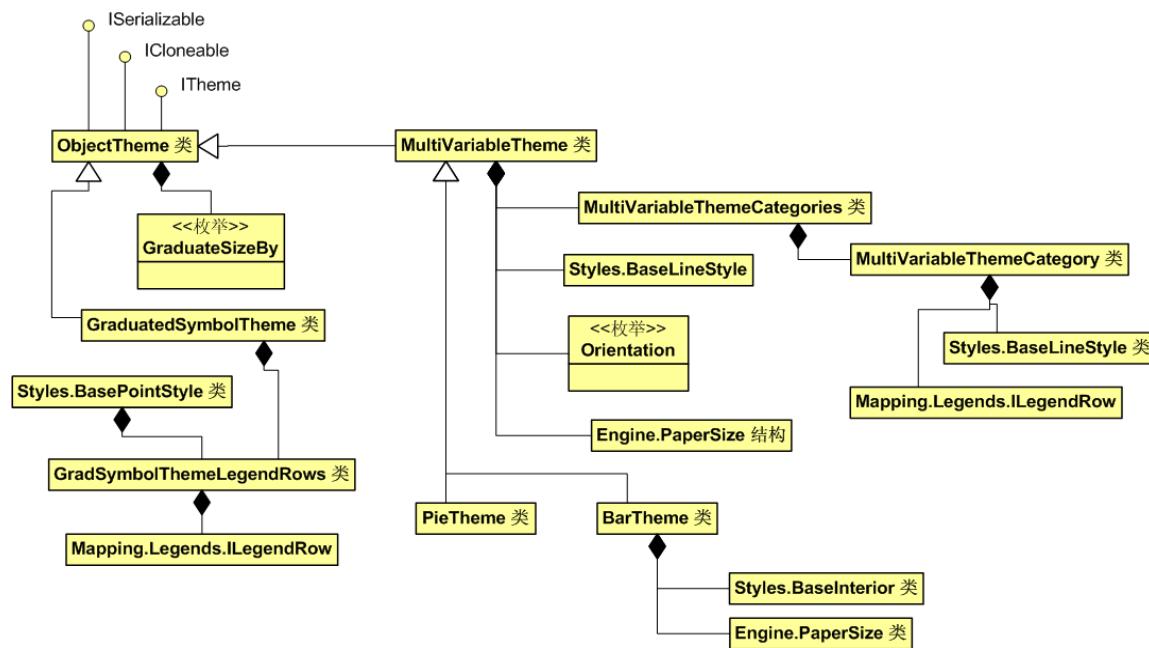
修饰符主题

图元修饰符主题的示例是范围、单值和点密度主题地图。它们修改图层中的现有图元。下面的 UML 图表概述了修饰符主题层次。



对象主题

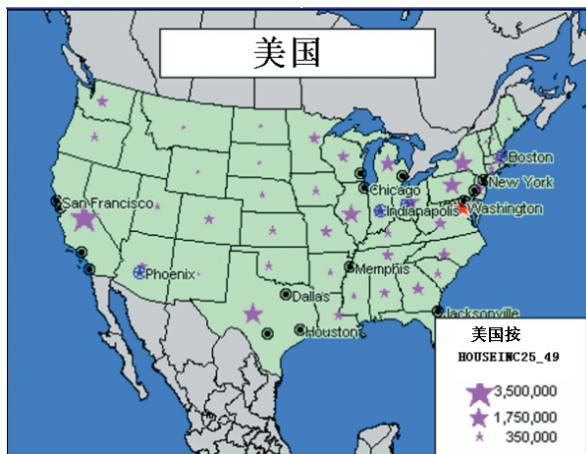
对象主题包括分级符号、饼图和条形图。这些主题创建表示数据值的对象。下面的 UML 图概述了对象主题层次。



GraduatedSymbolTheme

分级符号主题是包含点图元的对象主题，其符号大小基于主题表达式的数字值。

例如，用户可以用分级符号来显示某一地区人口特定阶段的家庭收入。



分级符号主题地图的示例。

何时使用分级符号主题

分级符号地图仅使用数值数据。如果正在处理的是餐厅表，则没必要根据每个餐厅所提供的烹调类型来创建分级符号。但是，如果想要按城市显示家庭收入的分布情况，则适合使用分级符号。

VB 示例：

```
Public Shared Sub MapInfo_Mapping_Thematics_GraduatedSymbolTheme(ByVal  
map As Map)  
  
    ' Load a map based on one table  
    map.Load(New MapTableLoader("world.tab"))  
    Dim lyr As FeatureLayer = CType(map.Layers("world"), FeatureLayer)  
  
    ' Create a new graduated symbol theme  
    Dim gradTheme As GraduatedSymbolTheme = New  
        GraduatedSymbolTheme(lyr.Table, "Pop_Native")  
  
    ' Create an object theme layer based on that graduated symbol theme  
    Dim thmLayer As ObjectThemeLayer = New ObjectThemeLayer("World Pop  
Growth Rate", Nothing, gradTheme)  
  
    'Add object theme to the map's layer collection.  
    map.Layers.Add(thmLayer)  
  
    ' Adjust how we graduate the size.  
    gradTheme.GraduateSizeBy = GraduateSizeBy.Constant  
    thmLayer.RebuildTheme()  
End Sub
```

PieTheme

饼状主题是包含饼状图的对象主题，其中饼块表示每个数据值。在饼图中可以比较单个饼图中的饼块，或检查所有饼图的特定饼块。饼图还能够比较整体中的各个部分。

何时使用饼图主题

饼图在分析人口统计学数据时特别有用。例如，对于“美国”人口统计学信息的数据集，数据集显示多个主要人口统计组的人口。通过饼图，可以显示每个人口统计组的人口，并查看组成饼图的每个饼块。这样就可以查看人口统计组在每一个州或整个“美国”的分布情况。还可以通过查看某一个人口统计组，了解不同州内人口的相对差异。

VB 示例：

```
Public Shared Sub MapInfo_Mapping_Thematics_PieTheme(ByVal map As Map)
    ' Load a map based on one table
    map.Load(New MapTableLoader("world.tab"))
    Dim lyr As FeatureLayer = CType(map.Layers("world"), FeatureLayer)

    ' Create a new pie theme
    Dim pieTheme As MapInfo.Mapping.Thematics.PieTheme = New
    MapInfo.Mapping.Thematics.PieTheme(map, lyr.Table, "Pop_Native",
    "Pop_Asian", "Pop_Other")

    ' Create an object theme layer based on that pie theme
    Dim thmLayer As ObjectThemeLayer = New ObjectThemeLayer("World Pop
    Growth Rate", Nothing, pieTheme)

    ' Add object theme to the map's layer collection.
    map.Layers.Add(thmLayer)

    ' DataValueAtSize is calculated automatically if not specified in
    the
    ' pie's constructor. But, you can adjust it. If you do so here,
    you
    ' have to rebuild the theme. You can adjust it before creating the
    ' object theme layer, and that way the pies won't need to be built
    ' twice.

    pieTheme.DataValueAtSize /= 2
    pieTheme.GraduateSizeBy = GraduateSizeBy.Constant
    thmLayer.RebuildTheme()
End Sub
```

BarTheme

条形主题是包含条形图的对象主题，其中条形表示每个数据值。条形图为了对象质心的每个地图对象（图元）生成，使您能够通过对条形高度的比较来分析特定图表中的多个主题变量。

何时使用条形图主题

条形主题可用于在地图的所有图元中检查同一变量。例如，对于包含女性和男性人口的美国州界表。通过条形图，可以创建主题地图，它为每个州都显示一个由两个条形组成的表：一个条形表示女性人口，另一个则表示男性人口。可以比较每个州的人口差异，或者检查多个州并比较与其它州的人口差异。

VB 示例：

```
Public Shared Sub MapInfo_Mapping_Thematics_BarTheme(ByVal map As Map)
    ' Load a map based on one table.
    map.Load(New MapTableLoader("world.tab"))
    Dim lyr As FeatureLayer = CType(map.Layers("world"), FeatureLayer)

    ' Create a new bar theme.
    Dim barTheme As MapInfo.Mapping.Thematics.BarTheme = New
    MapInfo.Mapping.Thematics.BarTheme(map, lyr.Table, "Pop_Native",
    "Pop_Asian", "Pop_Other")

    ' Create an object theme layer based on that bar theme.
    Dim thmLayer As ObjectThemeLayer = New ObjectThemeLayer("World
    Pop", Nothing, barTheme)

    ' Add object theme to the map's layer collection.
    map.Layers.Add(thmLayer)

    ' Stack the bars and graduate by a constant amount.
    barTheme.Stacked = True
    barTheme.GraduateSizeBy = GraduateSizeBy.Constant
    thmLayer.RebuildTheme()
End Sub
```

RangedTheme

范围主题会根据特定标准将已归类数据显示为范围（容器）。在 MapXtreme 2005 中，范围主题会修改现有的图层，以反映该项标准。不会创建新图层，原因是范围主题已在 MapX 和 MapXtreme 的先前版本中创建。在创建范围主题地图时，MapXtreme 2005 将所有数据集行集中在范围内，并为每行分配对象的颜色、样式或其对应范围的直线。



范围主题地图示例。

何时使用范围主题

例如，在有某区域人口统计数据时，可以使用范围主题。例如，可将亚洲的农村男性人口集中在容器内，并加上底色，以便指示存在于该区域内的人口范围。

数据集中所有记录都被分配到某一范围，然后根据该范围使用样式进行绘制。例如，如果颜色的范围是从黄色到绿色，则可将人口最高国家的底色设置为黄色，人口最低的则为绿色，其它范围选择由黄到绿之间任意颜色。地图显示时，这些颜色会明确指示出人口最高和最低国家的位置。

当区域的大小与数据值的大小不直接相关时，也可以使用范围。

范围值的类型

MapXtreme 2005 可以使用五种分布方法自动创建范围：

- 数量相等
- 范围相等
- 标准方差

- 自然间隔
- 分位数
- 定制

数量相等

“数量相等”在每个范围内具有相同数目的记录。如果想使用“数量相等”将 100 条记录分组到 4 个范围中，MapXtreme 2005 对这些范围进行计算，根据设置的舍入因子，为每个范围近似填充 25 条记录。

使用“数量相等”（或任何其它范围方法）时，注意任何可能影响主题地图的极大数据值（在统计信息中，这些值作为外部值引用）非常重要。

范围相等

“范围相等”按相等大小的区域划分记录。例如，在表中有一个字段，数据值取值范围从 1 至 100。想要创建具有四个相等大小范围的主题地图。MapX 产品划分了四块区域：1-25、26-50、51-75 和 76-100。切记，根据数据分布，MapXtreme 2005 可以创建不带任何数据记录的范围。

标准方差

使用“标准方差”创建范围时，中间范围按平均值进行间隔，并且中间范围上面和下面的范围按该平均值上面或下面的标准方差进行间隔。

自然间隔

自然间隔是显示非均匀分布数据的一种方式。它根据这样一种算法来创建范围：使用每个范围的平均值在所有范围中平均分布数据。使用“自然间隔”分布数据，可使每个范围的平均值尽可能接近该范围内的每个范围值。这可以确保各个范围可以很好地被它们的平均值表示，并且每个范围内的数据值彼此都非常接近。

分位数

分位数是显示非均匀分布数据的另外一种方式。分位数使用两个变量表达式。例如，使用 Quantile 分布方法显示人口识字率。

定制范围

如果以上分布方法都不能满足需要，则可以使用方法 `DistributionMethod.CustomRanges` 创建定制范围。请参阅 MapXtreme 2005 开发人员参考帮助中 `MapInfo.Thematics.RangedTheme.Recompute` 方法的代码示例。

VB 示例：

```
Public Shared Sub MapInfo_Mapping_Thematics_RangedTheme(ByVal map As Map)
    ' Create a ranged theme.
    Dim lyr As FeatureLayer = CType(map.Layers(0), FeatureLayer)
    Dim theme As MapInfo.Mapping.Thematics.RangedTheme = New
    MapInfo.Mapping.Thematics.RangedTheme(lyr, "Pop_1990/Area(obj)",
    "PopDensity", 5, DistributionMethod.EqualCountPerRange)

    ' Add the ranged theme to the layer.
    lyr.Modifiers.Append(theme)
End Sub
```

RangedLabelTheme

该类创建范围主题，在其中标注使用范围样式绘制。有关范围主题更加详细的讨论，请参阅第 238 页中的 *RangedTheme*。

何时使用 RangedLabelTheme 类

用标注来传递要标注的信息时，可使用范围标注主题。例如，可在标注城市或城镇人口时使用有范围的标注主题。人口较多的城市的标注字体会比人口较少的城镇大。

VB 示例：

```
Public Shared Sub MapInfo_Mapping_Thematics_RangedLabelTheme(ByVal
labelSource As MapInfo.Mapping.LabelSource, ByVal columnExpr As
String, ByVal themealias As String)
    ' Create new ranged label theme based on the label source of a
    ' LabelLayer already in the map. It will use 5 bins of equal range.
    Dim rangedLabelTheme As RangedLabelTheme = New
    RangedLabelTheme(labelSource.Table, columnExpr, themealias, 5,
    DistributionMethod.EqualCountPerRange)

    ' Add the label modifier to the label layer.
    Dim labelModifier As MapInfo.Mapping.LabelModifier =
    CType(rangedLabelTheme, MapInfo.Mapping.LabelModifier)
    labelSource.Modifiers.Insert(0, labelModifier)
End Sub
```

IndividualValueTheme

单值主题是修饰符主题，用来显示点、直线或由包含在数据集特定字段中的单值加上底纹的边界。可在单值地图中使用数字值和名义值。MapXtreme 2005 为每个惟一值指定自己的独特样式。

例如，使用 IndividualValue 主题图显示各块土地的分区类别。每个区域（商用、住宅、工业）将显示为不同的颜色。每块符合分区类别的土地都将加上相应底色。

何时使用 IndividualValueTheme 类

如果通过名义数据在点、直线或边界加上底纹，则只能通过单值执行。名义数据可以是非数字数据（例如，名称、提供的烹饪类型或销售的汽车品牌）或其数字不表示测量的数字数据。例如，名义数据可以是包含 ID 数值的列。

数据被视为数值数据，可用于范围地图和单值地图。

VB 示例：

```
Public Shared Sub MapInfo_Mapping_Thematics_IndividualValueTheme(ByVal  
map As Map)  
  
    ' Load a map based on one table  
    map.Load(New MapTableLoader("World.tab"))  
    Dim fLyr As FeatureLayer = CType(map.Layers("world"), FeatureLayer)  
  
    ' Create an individual value theme  
    Dim thm As IndividualValueTheme = New IndividualValueTheme(fLyr,  
    "Country", "World Pop")  
  
    ' Add the theme to the FeatureStyleModifiers list  
    fLyr.Modifiers.Append(thm)  
End Sub
```

使用自定义位图符号创建 IndividualValueTheme

以下示例说明了如何使用自定义位图符号创建 IndividualValueTheme。可用位图符号表包括在第 499 页附录 I 中的定制符号中。

```
MapInfo.Data.MIConnection conn = new MapInfo.Data.MIConnection();
conn.Open();

MapInfo.Data.Table ti=conn.Catalog.GetTable("usa_caps");
MapInfo.Mapping.FeatureLayer fl=mapControl1.Map.Layers["usa_caps"] as
FeatureLayer ;
MapInfo.Mapping.Thematics.IndividualValueTheme iv=new
MapInfo.Mapping.Thematics.IndividualValueTheme(fl,"state","state");
MapInfo.Styles.BitmapPointStyle bitmappointstyle = new
MapInfo.Styles.BitmapPointStyle("AMBU1-32.BMP",
MapInfo.Styles.BitmapStyles.All ,System.Drawing.Color.Red , 30);
MapInfo.Styles.CompositeStyle cs = new
MapInfo.Styles.CompositeStyle(null, null, null, bitmappointstyle);
MapInfo.Mapping.Thematics.ModifierThemeBin mtb= iv.Bins[0];
mtb.Style.ApplyStyle(cs);
bitmappointstyle = new MapInfo.Styles.BitmapPointStyle("BADG1-32.BMP",
MapInfo.Styles.BitmapStyles.All ,System.Drawing.Color.Red , 30);
cs=new MapInfo.Styles.CompositeStyle(null, null, null,
bitmappointstyle);
mtb= iv.Bins[1];
mtb.Style.ApplyStyle(cs);
fl.Modifiers.Append (iv);

conn.Close();
```

IndividualValueLabelTheme

该类创建在图层的标注上进行操作的单值主题。有关单值主题更加详细的讨论，请参阅第 241 页中的 *IndividualValueTheme*。

何时使用 IndividualValueLabelTheme 类

因为单值标注主题带有范围的标注主题，那么在使用这些标注来传递要标注的信息时，它们也非常有用。例如，利用街道数据时，可以使用单值标注主题用不同字体来标注不同类型的道路。在这种情况下，高速公路将使用与乡间小道不同类型的标注进行表示。

VB 示例：

```
Public Shared Sub  
    MapInfo_Mapping_Thematics_IndividualValueLabelTheme(ByVal labelSource  
        As MapInfo.Mapping.LabelSource, ByVal columnExpr As String, ByVal  
        themeAlias As String)  
        ' Create new individual value label theme  
        Dim theme As IndividualValueLabelTheme = New  
        IndividualValueLabelTheme(labelSource.Table, columnExpr, themeAlias)  
  
        ' Add the label modifier to the label layer.  
        Dim labelModifier As MapInfo.Mapping.LabelModifier = CType(theme,  
            MapInfo.Mapping.LabelModifier)  
        labelSource.Modifiers.Insert(0, labelModifier)  
    End Sub
```

DotDensityTheme

点密度主题是绘制区域填充图案的样式修饰符，它所使用的点基于主题表达式的数字值。

点密度地图用点来表示与边界或区域关联的数据值。区域中点的总数表示该区域的数据值。如果一个郡有 10,000 个老年人，每个点表示 100 个老年人，则在郡边界中应该有 100 个点。

何时使用 DotDensityTheme 类

点密度主题可用于显示原始数据，每个点表示较大量：人口、快餐店数目、某汽水品牌的批发商数目，等等。

例如，对于郡边界的人口明细表，则可以使用点密度主题来显示每个郡边界上的人口密度。可通过两个属性控制点密度地图。可以指定一个点的值。例如，要通过点密度主题表示纽约 Rensselaer County 中 20,000 个高校生，可以指定一个点表示 200 个学生。在该郡加上底纹时，地图会为该郡绘制 100 个点。

VB 示例：

```
Public Shared Sub MapInfo_Mapping_Thematics_DotDensityTheme(ByVal map
As Map)

    ' Load a map based on one table
    map.Load(New MapTableLoader("World.tab"))

    ' Create a dot density theme.
    ' Add it as a modifier.
    Dim lyr As FeatureLayer = map.Layers("World")
    Dim thm As MapInfo.Mapping.Thematics.DotDensityTheme = New
    MapInfo.Mapping.Thematics.DotDensityTheme(lyr, "Pop_1994", "World
    Pop", System.Drawing.Color.Red, DotDensitySize.Large)
End Sub
```

双变量主题地图

双变量主题地图绘制使用点或直线对象来表示两个主题变量。例如，星型标记可以表示一个变量，如青少年的数目，而使用蓝色填充的星型标记表示他们的年度购买量。

要在 MapX 中创建双变量地图，请创建两个主题地图，并将其中一个地图放在另一个地图上面，从而对象显示两个变量。

地图和变量的类型

适用于双变量地图绘制的主题地图类型只有范围地图和单值地图。可以根据数据，在双变量地图的组合之间进行选择：

- 两个范围地图
- 一个范围地图和一个单值地图

如果具有非数字变量，则其中一个地图必须是单值地图。使用两个非数字变量不能创建双变量地图。

显示属性

如果要使用一种符号显示两种变量，那么为每个变量选择不同的符号属性就很重要。例如，不能为两个变量选择同一种颜色，因为其中一种颜色将覆盖另一种颜色。从下列组合进行选择：

- 颜色和符号类型
- 颜色和大小
- 颜色和符号类型

符号类型只应用于名义或非数值数据，因为符号类型和数量之间没有内在关联。

VB 代码示例：双变量主题地图

```
Public Shared Sub  
    MapInfo_Mapping_Thematics_RangedThemeConstructor(ByVal lyr As  
        FeatureLayer)  
        Dim thm As MapInfo.Mapping.Thematics.RangedTheme = New  
        MapInfo.Mapping.Thematics.RangedTheme(lyr, "Literacy", "Pop_1994",  
        "Literacy Quantile by Pop", 4)  
        lyr.Modifiers.Append(thm)  
    End Sub
```

图例概述

MapInfo.Mapping.Legends 命名空间包含用于创建并显示主题和制图图例的类、接口和枚举。Legends 是 Thematic 或 Cartographic LegendFrames 的集合。每个框架都包含 LegendRow 的集合，每个 LegendRow 都具有文本和样式属性。

主题图例

主题图例提供用于主题的颜色、符号和样式的键。该键解释显示的颜色、符号和样式。



何时使用主题图例

具有包含主题的地图时，主题图例就非常有用。在显示降雨量的天气地图中，可用各种绿色底纹表示降雨。主题图例的重要性在于它可从视觉进行阐述，深绿色底纹表示最高降雨量，浅绿色底纹则表示较低降雨量。

代码示例

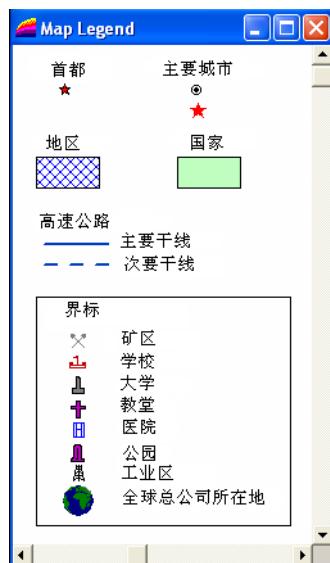
下面的代码是 ThemeLegends 样本的一部分，说明如何在表上创建范围主题，将主题作为 FeatureStyleModifier（如何修改几何体类型以反映主题中的容器）进行添加，创建图例和图例框架并将它作为修饰进行添加。完整的代码和 Visual Studio 解决方案位于 Samples\Desktop\Features\ThemeLegend 文件夹中。

VB 代码示例

```
Private Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
    LegendControl1.Map = MapControl1.Map
    If Not IsPostBack Then
        Dim normalLyr() As MapInfo.Mapping.LayerType = New MapInfo.Mapping.LayerType(1) {}
        normalLyr(0) = MapInfo.Mapping.LayerType.Normal
        Dim filter As MapInfo.Mapping.IMapLayerFilter =
MapInfo.Mapping.MapLayerFilterFactory.FilterByLayerType(normalLyr)
        Dim frame As MapInfo.Mapping.Legends.LegendFrame
        Dim NewLegend As MapInfo.Mapping.Legends.Legend =
MapControl1.Map.Legends.CreateLegend(New System.Drawing.Size(5, 5))
        NewLegend.Format.FrameAlignment =
MapInfo.Mapping.Legends.LegendFrameAlignment.Horizontal
        Dim ftrLayer As MapInfo.Mapping.FeatureLayer
        For Each ftrLayer In
MapControl1.Map.Layers.GetMapLayerEnumerator(filter)
            frame =
MapInfo.Mapping.Legends.LegendFrameFactory.CreateCartographicLegendFra
me(ftrLayer)
            NewLegend.Frames.Append(frame)
        Next
        LegendControl1.Legend = NewLegend
    Else
        LegendControl1.Legend = MapControl1.Map.Legends(0)
    End If
End Sub
```

制图图例

制图图例类用于读写制图图例元数据。图例通过元数据中的文本和样式标识地图上的每个制图图元。



何时使用制图图例

当地图包含表示地图上项目的对象时，制图图例非常有用。例如，具有路标的地图就需要制图图例。医院、学校、教堂和机场都由不同的符号表示。制图图例为表示在地图上的不同类型的路标提供可视性说明。

VB 示例：

```
Public Shared Sub MapInfo_Mapping_Legends_LegendFrameFactoryCreateCartographicLegendFrame()
    Dim Connection As MapInfo.Data.MIConnection = New MapInfo.Data.MIConnection
    Connection.Open()
    Dim versionNumber As String = String.Format("{0}.{1}",
        MapInfo.Engine.ProductInfo.MajorVersion,
        MapInfo.Engine.ProductInfo.MinorVersion)
    MapInfo.Engine.Session.Current.Catalog.OpenTable(("C:\Program
        Files\MapInfo\MapXtreme\" & versionNumber & "\Data\US\USA_CAPS.TAB"))

    Dim tab As Table = Connection.Catalog.GetTable("USA_CAPS")

    Dim lyr As MapInfo.Mapping.FeatureLayer = New MapInfo.Mapping.FeatureLayer(tab)

    Dim size As System.Drawing.Size = New System.Drawing.Size(400, 400)
    Dim legendfactory As MapInfo.Mapping.Legends.LegendFactory =
        MapControl1.Map.Legends
    Dim legend As MapInfo.Mapping.Legends.Legend =
        legendfactory.CreateLegend(size)

    Dim frame As MapInfo.Mapping.Legends.LegendFrame =
        MapInfo.Mapping.Legends.LegendFrameFactory.CreateCartographicLegendFrame(lyr, "State", "State",
            MapInfo.Mapping.Legends.RowTextSourceType.Expression)
    legend.Frames.Append(frame)
    LegendControl1.Legend = legend
End Sub
```

将图例格式化

MapInfo.Mapping.Legends.LegendFormat 类包含控制图例框架绘制方式的属性。可以控制显示属性，例如对齐、每行或每列中图例框架的数量、两个框架之间的间隔，以及是否应自动调整图例大小和框架位置。

LegendFormat.FrameAlignment 属性与 FramesPerRow 一起使用可用于水平对齐，FramesPerColumn 则用于垂直对齐。

例如，如果图例包含 10 个框架并将 FrameAlignment 和 FramesPerRow 分别设置为 Horizontal 和 5，则图例将框架显示为两行，每行显示五个图例。如果每行有 10 个框架，则 10 个框架显示在单一行中，宽度则为 10 个框架的宽度。

类似情况发生在垂直对齐时。如果有 10 个框架并将 FrameAlignment 和 FramesPerColumn 分别设置为 Vertical 和 5，则将显示 5 行，每行 2 个框架（5 行 2 列）。框架垂直向上对齐为每列 5 个框架。将 FramesPerColumn 设置为 10 时，图例将包含 10 行，每行 1 个框架（每列 10 个框架）。

FramesPerRow 和 FramesPerColumn 的默认设置为 0。使用的值是由 LegendFrameRows.Count 或 LegendFrameColumns.Count 属性指示的行或列中的当前框架数目。

14

查找位置

MapInfo.Data.Find 命名空间包括了在使用地址、十字路口或街道名称搜索地图图元时要使用的类。

本章内容：

- ◆ 查找功能概述 252
- ◆ *Data.Find* 命名空间概述 255
- ◆ 微调查找过程 263

查找功能概述

想通过地址、十字路口或街道名称定位地图图元时可以使用查找。要使用选择工具或查询来查找位置上的图元，请使用 Data 命名空间中的 Search 类（请参阅第 185 页中的搜索图元）。

一次成功的 Find 操作可能会出现完全匹配、一个或多个近似匹配或者没有匹配（匹配失败）。该操作可由多种属性和后备选项进行控制，有关内容将在第 255 页中的 *Data.Find* 命名空间概述中进行讨论。下节描述了 MapXtreme 2005 如何进行匹配。对 Find 过程了解的越多，对成功进行查找的操作控制就越多。

查找过程

MapXtreme 2005 通过将地址、十字路口或位置名称与图元表中的信息进行匹配来定位地图图元。例如，如果打开了 Washington D.C. 街道的表，就可以查找 Washington, D.C. 中的 1600 Pennsylvania Avenue。

查找地图图元也与上述方法类似。例如，如果有一张可制图的路标表，上面包含有“白宫”和它的地理参考（可制图的）位置，就可以找到“白宫”的位置。在按位置名称进行查找时不必提供它的地址。

必须提供构成十字路口的两个街道名称才能查找该十字路口。

MapXtreme 2005 尝试进行完全匹配，在该匹配中结果和输入的地址、位置名称或交点逐字匹配。如果无法进行完全匹配，MapXtreme 2005 会尝试查找基于其匹配规则和设置的模糊匹配。如果无法查找到模糊匹配，则返回匹配失败。注意，匹配不区分大小写。大写和小写字符之间可以成功匹配。

街道地址通常由街道号码、名称以及表示街道前缀（如，North）和后缀（如，Road）的缩写组成。地址可以采用各种形式并且可以包括附加的信息，例如，公寓号码或路径。另外，输入的地址可能缺少某些重要部分，像街道类型。MapXtreme 2005 会检查地址的每个部分并对其应用特定规则以查找匹配。

下面的章节描述了 MapXtreme 2005 如何处理包含街道名称、街道缩写、地址号码、优化表以及结果的特定信息和条件。

与街道名称匹配

与街道名称匹配是最直接明了的特性，它根据搜索表中的信息对地址的字符进行评估。例如，如果街道名称 LaSalle 位于表中，则 MapXtreme 2005 为它返回完全匹配 LaSalle。但是，如果将该地址拼写为 La Salle 或 LaSal，则返回近似匹配。

与街道缩写匹配

地址记录中的街道缩写千差万别。有时，完全缺少街道的组成部分。然而，在很多情况下，MapXtreme 2005 仍可以获得完全匹配，即使地址与搜索表稍有不同。MapXtreme 2005 用称为 MapInfo.abb 的标准地址缩写和替换列表来查找适合的匹配。该列表包括了街道前缀和后缀缩写的标准拼写，如，ST 表示 Street，Av 表示 Avenue。不过必须设置属性，告诉 MapXtreme 2005 使用该缩写文件，而且它还是添加获得完全匹配的机会或比不使用该文件找到更多近似匹配的好方法。

下表说明了地址中的各种变化以及根据缩写文件它是否可以获得完全匹配。第一列包含想要查找的街道名称。第二列包含来自源表的相应街道名称。第三列说明没有获得匹配的原因。第四列指示特定问题是否可以通过使用缩写等价文件进行更正。该表假设地址位于表中的单一列。而且，街道号码也位于同一列中，我们在此处没有指示街道号码，因为它们的处理方式与街道名称的处理方式不相同。

要查找的地址	搜索表地址	备注	是否可以使用缩写文件对其进行更正以获得完全匹配
LaSalle Street	LaSalle St	“Street”与“St”不匹配。	是
LaSalle Ave	LaSalle Av	“Ave”与“Av”不匹配。	是
LaSalle Ave	LaSalle St	“Ave”与“St”不匹配。	否
LaSalle	LaSalle St	目标缺少“St”。	否。如果缺少街道缩写，那么 MapXtreme 2005 无法判断它原来的样子。
LaSalle St North	LaSalle St	搜索表中不存在“North”。	否
LaSalle St North	LaSalle St N	“North”与 N 不匹配。	是
LaSalle St Apt 3	LaSalle St	源中没有与之匹配的公寓号码。	是。忽略该公寓号码。
Tenth St	10th St	“Tenth”与“10th”不匹配。	是
10th Av	Tenth Av	“10th”与“Tenth”不匹配。	是
Saint John's Lane	St John's Lane	“Saint”与“St”不匹配。	是

如果发现已经重复多次由于缩写而不能获得匹配的情况，那么可以：

- 编辑地址使它更符合缩写文件，或者
- 使用文本编辑器编辑缩写文件以添加您的特定缩写。mapinfow.abb 位于 C:\Program Files\Common Files\MapInfo\MapXtreme\6.x 中。有关编辑缩写文件的详细信息，请参阅 [第 263 页中的微调查找过程](#)。

匹配地址号码

MapXtreme 2005 在地址号码位于街道名称前面（类似“北美”地址）时或地址号码跟在街道名称后面（一般是“欧洲”地址）时可以进行匹配。默认情况下，MapXtreme 2005 假设地址号码位于街道名称的前面。当地址号码跟在街道名称后面时，必须对属性进行设置。

MapXtreme 2005 将比较输入地址号码与地址号码范围。通常，该表包括了该范围覆盖的那部分相应街道的街道地址号码范围。地址范围可精确匹配到街道的一侧，因为地址范围往往是街道的一侧是奇数，另一侧是偶数。

在 MapXtreme 2005 不能精确匹配地址范围的情况下，您可能想要在最接近的范围内获得能被视为匹配的近似匹配。仅为非常精确的查找使用严格的完全匹配要求。大多数情况下，没有必要使用这样的精确级别。近似匹配通常是可以接受的。

使用优化边界表匹配

MapXtreme 2005 还可以在一个可能存在多个可能匹配的表中查找地址。要避免找到错误的地址，可以指定优化表和列以将匹配集中到较小的区域内。

这个方法非常有用，例如，搜索覆盖了整个郡的街道的表并且要查找 Main St 时。很可能在这个郡里不止一个城镇具有 Main St。通过提供城镇边界的优化表，可以指定仅查找“城镇 A”中的 Main St。

可以按需要使用任何类型的优化边界表，例如，邮政编码边界或人口调查区域。此外，还可以指定进行 Find 的第二个优化边界。

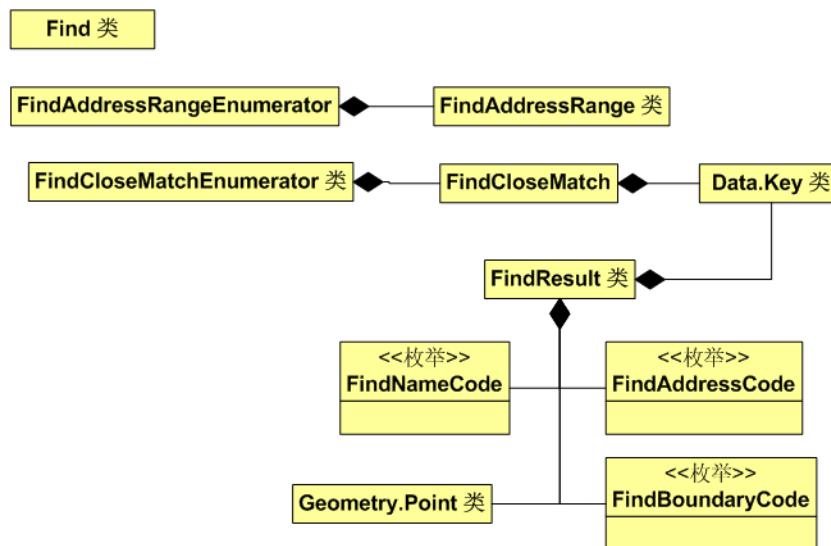
查找结果

MapXtreme 2005 返回完全匹配、一个或多个近似匹配或者未找到匹配。查找返回的结果取决于很多因素，包括输入数据的质量和为该查找操作设置的条件。查找还将返回说明地址的每个部分如何匹配（或未找到匹配）的信息。

Data.Find 命名空间概述

MapInfo.Data.Find 命名空间包含了使您可以在可制图的表中定位地图图元、街道地址或十字路口的类。Find 类的属性和方法用于设置 Find。使用 FindAddressRange、FindCloseMatch 和 FindResult 类返回 Find 结果。

下面的 UML 图表说明了 Find 命名空间。



Find

Find 对象用于在给定的可制图的表中定位地图对象、街道地址或十字路口。Find 对象对表进行搜索以获取匹配并在 FindResult 对象中返回结果。

要在 MapXtreme 2005 中使用 Find，必须具有：

- 可制图的表（即包含几何体对象的表）
- 编入索引的列，在该列执行搜索
- 要搜索的项，例如位置名称、街道地址或十字路口
- （可选）优化表，在该表中缩小搜索范围以获得更具体的位置。

Find 类提供了若干控制搜索操作的属性。例如，可以限制要返回的近似匹配的数目 (CloseMatchesMax) 或指示希望使用缩写文件来添加匹配的可能性 (UseAbbreviations)。

Find 类提供了四种搜索方法：其中两种方法使用 / 不使用优化边界搜索地址或图元，另外两种方法使用 / 不使用优化边界搜索十字路口。

属性	描述
AddressNumberAfterStreet	指定地址号码是否位于街道名称的后面（例如“Smith Street 107”）。
ChooseAlternateBoundary	指定是否与优化区域中找到的记录匹配，而不是与指定的优化区域中的记录匹配。
ChooseClosestAddressRange	指定地址号码不匹配时是否使用类中最接近的可用地址号码。
ChooseClosestObject	指定找不到完全匹配时是否查找最接近的对象。
CloseMatchesMax	指定如果找不到完全匹配则要返回多少个接近的匹配。
InsetDistance	正值表示到直线结尾的距离，用于调节地址位置的放置。
InsetPercentage	表示在放置地址的位置直线长度的百分比。
InsetUnit	表示 Inset 所用的距离单位。
OffsetDistance	表示从街道将地址位置的放置往后偏移的距离。
OffsetUnit	表示 Offset 所用的距离单位。
UseAbbreviations	指定从缩写文件替换的缩写是否用于查找匹配（例如，将“Smith Street”替换为“Smith St”）。
UseCloseMatches	指定如果找不到完全匹配是否返回“N”个接近的匹配。
UseInsetAsPercentage	指定 Inset 被用作百分比还是距离。

方法	描述
Search	在可制图的表中搜索命名的位置并返回 FindResult 对象。
SearchIntersection	在可制图的街道表中搜索给定的交点，返回 FindResult 对象。

Find.Dispose 方法

释放 Find 对象保存的非管理资源。在使用完 Find 对象后需要调用此方法。

代码示例

下面的代码示例基于与 MapXtreme 2005 一起提供的 Find 示例应用程序。该示例应用程序位于 C:\Program Files\MapInfo\MapXtreme\6.x\Samples\\Common\Features\Find 目录。

```
// Here we declare the tables and columns to be used in our
// Find operation.
Table _searchTable;
Table _refiningTable;
Column _searchColumn;
Column _refiningColumn;
FindResult _result;

// Here we declare our Find object.
Find find = null;

//Here we define the table that contains the locations we want to find.
_searchTable = Session.Current.Catalog.OpenTable("US_CNTY.TAB");
if (_searchTable == null)
    return false;

// Here we define the table that will narrow our search if our search
// // table has duplicate named locations it might find.
_refiningTable = Session.Current.Catalog.OpenTable("USA.TAB");

// Here we define the column in the search table that contains the
// // location we want to find. In this case we will be looking for
// particular county from a table of all counties in the United States.
Columns columns = _searchTable.TableInfo.Columns;
_searchColumn = columns["County"];
if (_searchColumn == null)
    return false;

// Here we define the column in the refine table that contains the
// // location we want to use to narrow our search in the
// cases where we find non-unique county names. In this case duplicate
// // counties will be further determined by the state
// they are in. This refining column guarantees a unique result.
columns = _refiningTable.TableInfo.Columns;
_refiningColumn = columns["State"];

// Here we create our find object. Here we are defining it to find
// // counties in a table of US counties and if we find
// // duplicate county names, return the one unique to a particular state.
find = new Find(_searchTable, searchColumn, _refiningTable,
    refiningColumn);

// Here we set some properties of the Find object. These properties
// define what to do if the result of our search is an inexact match.
// Here we say to return up to six close matches if no exact matches.
// Close matches refer to the name and are helpful in the case of
// misspellings, e.g., searching for Albny, NY would return Albany, NY.
find.UseCloseMatches = true;
find.CloseMatchesMax =6;
```

```
// Here we perform the search on our Find object. The Search method
// returns a SearchResult object. This object contains properties that
// describe the results of the Search operation. If the operation was
// successful you could use the SearchResult.FoundKey property to get a
// key to the row in your search table that is the result of your
// search operation. Or you can uses as we do here the the FoundPoint
// property to get the point location of your successful match.
if ( _refiningTable != null && _refiningTable.IsOpen )
{
    _result = find.Search("Albany", "NY");
}
else
{
    _result = find.Search("Albany");
}
if (( _result.ExactMatch)
    && (_result.NameresultCode.Equals(FindNameCode.ExactMatch)
        && (_result.FoundPoint != null)))
{
    // We succeeded, display the point. (This function is in the Find
    // sample application.)
    showPointOnSearchTableMap(_result.FoundPoint.X,
        _result.FoundPoint.Y);
}
else
{
    // We failed to get an exact match but have close matches, enumerate
    // through them.
    string labelSearchResult.Text = _result.NameresultCode.ToString();
    if ( find.UseCloseMatches )
    {
        listBoxSearchResult.Visible = true;
        FindCloseMatchEnumerator enumerator =
            _result.GetCloseMatchEnumerator();
        while ( enumerator.MoveNext() )
        {
            // This is in the Find sample application
            listBoxSearchResult.Items.Add(enumerator.Current.Name);
        }
    }
}
find.Dispose();
```

FindAddressRange

FindAddressRange 对象表示从 Find.Search 方法返回的地址范围项。在未找到街道地址、地址号码不在给定街道的最小 / 最大地址范围内或未指定地址号码时，FindAddressRange 对象作为 FindResult 对象的一部分返回。

代码示例

```
public void GetAddressRangesOnStreetTable()
{
    Table _table;
    _table =
Session.Current.Catalog.OpenTable("North_Greenbush.tab");

    Find _find = new Find(_table,_table.TableInfo.Columns[1]);
    FindResult _findResult= _find.Search("Meadow Dr");
    If ((!_findResult.ExactMatch) && (_findResult.NameresultCode
== FindNameCode.ExactMatch) && (findResult.AddressresultCode
== FindAddressCode.AddressNumNotSpecified))
    {
        FindAddressRangeEnumerator _enum =
            _findResult.GetAddressRangeEnumerator();
        FindAddressRange _findAddressRange;
        int _iIndex = 0;

        while (_enum.MoveNext())
        {
            _findAddressRange = _enum.Current;
            Console.WriteLine("_findAddressRange.BeginRange");
            Console.WriteLine("_findAddressRange.EndRange");

            _iIndex++;
        }

        if(_table != null)
        {
            _table.Close();
            _table = null;
        }
    }
    find.Dispose();
}
```

FindCloseMatch

FindCloseMatch 对象表示从 Find Search 方法返回的近似匹配项。该对象作为 FindResult 对象的一部分返回。近似匹配项是一个返回匹配，它与请求的搜索名称近似匹配。

要使用此功能，必须先设置 Find 对象的 UseCloseMatches 和 CloseMatchesMax 属性才能执行搜索。例如，如果尝试搜索“Washington Street”并将 UseCloseMatches 设置为 true，那么得到的近似匹配为“Washington Ave”。

代码示例

```
public void CloseMatchesOnStreetTable()
{
    Table _table;
    _table = Session.Current.Catalog.OpenTable("Rensselaer.tab");

    Find _find = new Find(_table, _table.TableInfo.Columns[1]);
    _find.UseCloseMatches = true;
    _find.CloseMatchesMax = 5;

    FindResult _findResult = _find.Search("70 Washington");

    if ((!_findResult.ExactMatch) && (_findResult.NameresultCode ==
        FindNameCode.ExactMatchNotFound))
    {
        FindCloseMatchEnumerator _enum =
            _findResult.GetCloseMatchEnumerator();
        FindCloseMatch _findCloseMatch;
        int _iIndex = 0;

        while (_enum.MoveNext())
        {
            _findCloseMatch = _enum.Current;
            Console.WriteLine(_findCloseMatch.Name);
            _iIndex++;
        }
    }

    if (_table != null)
    {
        _table.Close();
        _table = null;
    }
    find.Dispose();
}
```

FindResult

FindResult 类以属性形式返回 Find.Search 方法中的信息，这些属性说明要进行哪种类型的匹配，如下表所示。如果成功，FoundKey 属性包含要定位的对象的 Key。如果成功，FoundPoint 属性包含已定位对象的点。

属性	描述
AddressOutOfRange	指定传入的地址是否超出范围。
AddressresultCode	标识地址部分搜索的结果代码并返回 FindAddressCode 枚举。
BoundaryresultCode	标识优化边界部分搜索的结果代码并返回 FindBoundaryCode 枚举。
ExactMatch	指定是否找到完全匹配。
FoundKey	指定已定位对象的 Key。
FoundPoint	指定已定位对象的 Point。
IntersectionNotFound	指定是否找不到交集。
MultipleMatches	指定是否找到多个匹配。
NameresultCode	标识要搜索的名称的结果代码并返回 FindNameCode 枚举。
resultCode	如果 Find 生成了完全匹配，则值为 1。如果 Find 生成了近似匹配，则值大于 1。如果 Find 未找到匹配的地址，则结果为负值。
UseSubstitution	指定是否从缩写文件使用替换。

FindAddressCode 枚举

标识地址部分搜索的结果代码并且该代码由 FindResult.AddressresultCode 属性返回。

注： 该结果代码只应在搜索街道或十字路口时使用。

成员名称	描述
ExactMatch	找到完全匹配。
SideOfStreetUndetermined	未决定街道的那一侧。
WithinMinMax	地址号码在最小（或）最大范围内。
NotWithinMinMax	地址号码不在最小（或）最大范围内。

成员名称	描述
AddressNumNotSpecified	地址号码未指定。
StreetsDoNotIntersect	街道不相交。
NoMapObjectForRowMatched	匹配的行不具有地图对象。

FindBoundaryCode 枚举

标识优化边界部分搜索的结果代码并且该代码由 FindResult.BoundaryresultCode 属性返回。改进边界用于区分具有相同名称的多个图元。

注： 该结果代码只应在区域用于优化搜索时使用。

成员名称	说明
ExactMatch	找到完全匹配。
FoundInOneOtherRegion	仅在指定区域之外的某一个区域中找到该名称。
FoundInMoreThanOneRegion	在除指定区域外的多个区域中找到该名称。
NoRegionSpecifiedOneMatch	未指定优化的区域，且找到一个匹配。
NoRegionSpecifiedMultipleMatches	未指定区域，且找到多个匹配。
MultipleMatchesFound	在指定区域中多次找到该名称。

FindNameCode 枚举

标识正搜索的名称的结果代码并且该代码由 FindResult.NameresultCode 属性返回。

成员名称	描述
ExactMatch	找到完全匹配。
SubstitutionUsed	来自使用的缩写文件的替换。
ExactMatchNotFound	未找到完全匹配。
NoObjectNameSpecified	未指定对象名称；找不到匹配。
CloseMatch	找到接近的匹配。

微调查找过程

在本章的开始我们就已经指出，对 Find 的工作原理了解的越多，就越能更好地使用它的属性和输入信息，从而提高成功匹配的机会。本节提供了一些帮助，具有用于优化 Find 期间返回结果的策略。

编辑 Mapinfo.abb 文件

Mapinfo.abb file 是随 MapXtreme 2005 一起交付的缩写和替换文件。如果输入地址中的缩写可以在缩写文件中找到，则可以使用该文件来提高 Find 机会。有关内容包括在与街道缩写匹配（第 253 页）一节中。本节包括了可以包含在此文件中的附加信息类型。

可以在任何文本编辑器或字处理程序中编辑 Mapinfo.abb。打开该文件并根据需要添加关键字。下面是该文件的标准版本：

```
!Version 3.0
FIRST      1ST
SECOND     2ND
THIRD      3RD
FOURTH     4TH
FIFTH      5TH
SIXTH      6TH
SEVENTH    7TH
EIGHTH     8TH
NINTH      9TH
TENTH      10TH
NORTH      N
SOUTH      S
EAST       E
WEST       W
ALLEY      AL
AVENUE     AV
AVE        AV
BOULEVARD BLVD
BRIDGE     BR
CIRCLE     CIR
COURT      CT
DRIVE      DR
EXTENSION EXT
HIGHWAY    HWY
INTERSTATE I
LANE       LN
MOUNT      MT
PARK       PK
PARKWAY    PKWY
PLACE      PL
```

```
PLAZA      PLZ
POINT      PT
RAILROAD   RR
ROAD       RD
ROUTE      RT
SAINT      ST
SQUARE     SQ
STREET     ST
STR        ST
TERRACE    TER
!EOLNOSPACE

'
;
#
!EOLSPACE
FLOOR
SUITE
"P.O. BOX"
!NOSPACE

.
\""
\!
\\
!SPACE
"STATE HIGHWAY"STHWY"
"N ST" NORTH ST"
"S ST" SOUTH ST"
"E ST" EAST ST"
"W ST" WEST ST"
"N AV" NORTH AV"
"S AV" SOUTH AV"
"E AV" EAST AV"
"W AV" WEST AV"
```

可以对此文件进行添加以适应多方面的问题。最重要的是可以进行几种不同类型的添加。MapXtreme 2005 识别四种类型的替换项并对这些类型分别进行解释。每种类型前面都有关关键字，用来在缩写文件中对其进行标识。

空格分隔的简单替换	!SPACE
简单截断	!EOLNOSPACE
以空格分隔的截断	!EOLSPACE
简单替换	!NOSPACE

为了使 MapXtreme 2005 知道如何在缩写文件中解释一行或一组行，必须将指示适当解释策略的关键字放在行首。

当缩写文件中的所有项使用默认解释时，则没有必要将任何关键字放在行首。当缩写文件的开头没有任何关键字时，MapXtreme 2005 将把初始项视为需要默认解释。但是，一旦添加了其它类型的替换对，则必须添加关键字。

空格分隔的替换

默认值是以空格分隔的简单替换。这意味着：MapXtreme 2005 将目标地址中以空格分隔的标记与地址文件中的行进行比较。以空格分隔的标记是字符前后带有空格的字符串。例如，MapXtreme 2005 将把“*Ave*”与“*Park Ave*”中的“*Av*”进行匹配，但是不会将“*Avery Blvd*”与“*Avry Blvd*”进行匹配（尽管两个街道名称都包含字符串“*Ave*”）。字符串“*Ave*”只在“*Park Ave*”中由空格定界，而未在“*Avery Blvd.*”中以空格定界。在“*Avery Blvd*”中，“*Ave*”后跟“*r*”而不是空格。

缩写文件中的所有项都将接收这个默认解释。可以添加要接收同样处理的其它项。例如，可能想要添加“*WK WALK*”对，以使 MapXtreme 2005 知道将目标地址中的“*WK*”当作“*WALK*”进行解释。同样地，可能想要添加类似如下的对：“*WAY WY*”。

使用关键字“*!SPACE*”来指示以空格分隔的简单替换。将对跟在“*!SPACE*”后面的项进行默认解释（这使可以按其它顺序安排缩写文件的内容）。在 MapXtreme 2005 遇到另一个关键字时，它将切换到指示的解释策略。

简单截断

在简单截断中，MapXtreme 2005 查找地址中的项并将它以及后跟的内容完全忽略。这些项不一定要用空格分隔。这个策略对于处理类似如下的地址非常有用：

123 Appian Way, Mail Stop 829

7305 Van Zandt # 23

在第一种情况下，想要 MapXtreme 2005 忽略逗号以及它后跟的内容。在第二种情况下，想要 MapXtreme 2005 忽略#号以及后跟的内容。要处理这样的情况，请将下面的行添加到缩写文件中：

```
!EOLNOSPACE  
,  
#
```

“*!EOLNOSPACE*”是指示下面的项将被视为简单截断的关键字。之后，我们得到了一个具有逗号的行和一个具有#号的行。不论 MapXtreme 2005 何时在地址中遇到逗号或#号，都将忽略它以及后跟的内容。上面的示例变为：

123 Appian Way

7305 Van Zandt

以空格分隔的截断

在以空格分隔的截断中，MapXtreme 2005 查找那些以空格分隔的项并将它们以及后跟的内容删除。例如：

73 Appian Way Suite 829

3033 Van Zandt Room 202

要处理这样的情况，请将下面的行添加到缩写文件中：

```
!EOLSPACE  
SUITE  
ROOM"
```

“!EOLSPACE” 是指示下面的项将被视为简单截断的关键字。之后，我们得到了一个具有“Suite”的行和一个具有“ROOM”的行。无论 MapXtreme 2005 何时遇到这些标记，它都将截断地址。上面的示例变为：

73 Appian Way

3033 Van Zandt

简单替换

MapXtreme 2005 使用简单替换从地址中删除项，否则不执行任何操作。使用它来处理：

433 Van-Rensselaer

91 St Albans'

目标是删除连字符和单引号。将下面的项添加到缩写文件中：

```
!NOSPACE  
-  
,
```

“!NOSPACE” 是调用简单替换的关键字，并且下面行中的连字符和单引号都是要删除的标记。上面的示例变为：

369 VanRensselaer

91 St Albans

合法的空格

可能会有想为含有空格的搜索字符串指示替换的情况。可以在这些情况中使用双引号。将双引号放在：

- 行首；或者
- 搜索的字符串和替换之间；或者
- 行尾。

例如，可能想要使用“STHWY”替换“State Highway”。要执行此操作，请使用下面一行：

```
"State Highway"STHWY"
```

这对细微的问题提供了解决方案，街道名称与缩写文件中的项匹配。例如，“North St”和“Park Av”都具有与缩写文件中的项相匹配的初始字符串。因此，MapXtreme 2005 将使用“N”替换“North”以得出“N St”并使用“Pk”替换“Park”以得出“Pk Av”。可以将下面的行添加到“缩写”文件中以调整这些替换：

```
"N ST"North ST"  
"PK AV"PARK AV"
```

注意：这些行必须位于使用“N”替换“North”和使用“PK”替换“Park”的项的后面。如果这些行位于上述项的前面，则不起作用。因此：

```
...  
...  
NORTH      N  
...  
...  
PARK        PK  
..  
...  
..."N ST" "NORTH ST"  
..."PK AV" "PARK AV"  
...  
...
```

在 MapXtreme 2005 遇到 NORTH N 时，它将把 NORTH ST 变为 N ST。在遇到“N ST”“NORTH ST”，那么它将把 N ST 变为 NORTH ST。遇到 PARK AV 时也进行类似的处理。

特殊字符

MapXtreme 2005 使用惊叹号(!)、双引号(") 和反斜杠(\) 作为特殊字符。这些字符告诉 MapXtreme 2005 如何处理跟在它们后面的字符串，但通常它们自己不被视为替换字符串中的字符。惊叹号告诉 MapXtreme 2005 不应将字符串解释为缩写。双引号告诉 MapXtreme 2005 字符串中的空格是合法的。而反斜杠则告诉 MapXtreme 2005 将特殊字符视为普通字符。

当想要在一行中将任何字符用作简单字符时，就在它们的前面加上反斜杠。因此：

```
\!
\"
\\
```

在缩写文件中添加行

通过增加新行，可以将新的项增加到该文件中。除了考虑到对其它行的影响，希望使用一个替换对以外，添加新行的顺序不是很重要。一行中第一个项和第二个项之间的空格数也不重要。

错误的地址范围

地址包含源表以外的数字范围时，MapXtreme 2005 无法与之匹配。这样的地址可能落入范围值之间的间隙或超出范围界限。要处理该问题：

1. 将 Find.ChooseClosestAddressRange 属性设置为 true，然后执行搜索。
2. 通过查看 FindAddressCode 枚举（由 FindResult.AddressOutOfRange 属性返回）解析失败的匹配。

该地址可能用于在生成源表后增加的街道分段。在这种情况下，可能要对源表进行编辑，使其反映该街道地址的整个范围。

不准确的城镇名

MapXtreme 2005 在 Find 操作中的最后一步是决定将匹配的街道名称放置到哪个地区。如果在设置查找过程时已正确指定，则 MapXtreme 2005 只执行该步骤。通常将城镇或城市名作为改进边界。由于人们通常不使用 Census Bureau 分配到各地址的城镇名，因此就产生了一些问题。由于几乎美国所有的电子地图都基于 Census Bureau 地图，因此就产生一些问题。

例如，地址 “50 Wolf Rd., Albany, NY” 实际位于 Colonie 的小镇内。因此，在目标地址中该城镇名对应的地址无法与源文件中正确的城镇相匹配。

解决该问题的一个办法是使用 ChooseAlternateBoundary 属性。启用该选项时，MapXtreme 2005 会尝试将地址与其所在边界相匹配（假定该地址只位于一个边界内）。地址位于多个边界之内时，Find 操作失败。

处理该问题的另一个方法是将 ZIP Code 而不是城镇或城市名作为改进边界。

15

为地图设置样式

MapXtreme 2005 中的样式不仅影响地图图元的外观，而且还影响了地图绘制应用程序的许多组件。样式可用于具有很多可控属性的标注、文本、主题、图例、选择集和演示文稿，所以实际上可以设计出喜欢的任何样式。

本章讨论了 MapXtreme 2005 框架方面的样式，特别讨论了 `MapInfo.Styles` 命名空间。

本章内容：

- ◆ *MapInfo.Styles* 命名空间概述 270
- ◆ 样式说明 272
- ◆ 预定义的样式和 *StyleRepository* 类 275
- ◆ 使用样式 276
- ◆ 覆盖样式 277

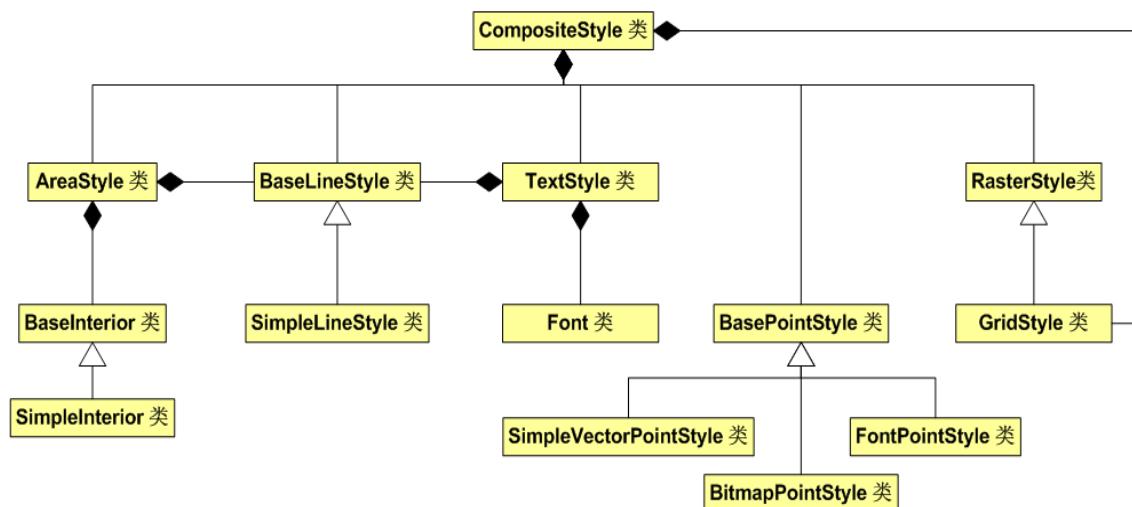
MapInfo.Styles 命名空间概述

MapInfo.Styles 命名空间强调了 MapXtreme 2005 的新 Styles 对象模型。Style 类是所有样式的基类。从 Style 派生的类包括 AreaStyle、BaseLineStyle、BaseInterior、BasePointStyle、CompositeStyle、Font、RasterStyle 和 TextStyle。从 BaseLineStyle 派生的是 SimpleLineStyle 类。从 BaseInterior 派生的是 SimpleInterior 类。从 BasePointStyle 派生的是 BitmapPointStyle、FontPointStyle 和 SimpleVectorPointStyle 类。GridStyle 从 RasterStyle 派生以包括网格特定样式设置。此外，提供了 StockStyles 类用于创建公共样式类型。

用户无法实例化抽象样式类 Style、BaseLineStyle、BaseInterior 或 BasePointStyle。您必须创建特定的样式，如 SimpleLineStyle，或将其创建为 CompositeStyle。

Style 自己就是一个对象，不再存储于其它对象中了。每个包含几何体列的表还包括 Style 列（别名为 MI_Style），该列包含了数据类型 Style。对于样式重载，Feature 类提供了 FeatureStyleModifier 和 FeatureOverrideStyleModifiers。Style 对象模型还生成一些可用的集合类（样式库）以保存用于样式对话框控件的样式。

所有样式类支持以稀疏方式应用的能力。请参阅 FeatureOverrideStyleModifiers 一节。



Style 可在 MapXtreme 2005 的多个区域使用，包括表示地理图元的直线、内部填充和点样式。Style 还整合了标注、文本、布局、主题、覆盖、图例和选择集。Style 属性的范围从标准颜色填充、直线宽度和磅值，到背景效果、位图点样式和旋转角度。实际上，可以想象出来的任意样式属性都可以应用于应用程序。此外，还可以更改全局样式或每个图元的样式，覆盖当前显示样式或进行永久性更改。

MapXtreme 2005 包含了很多让您入门的样本样式。其中包括 170 多个内部填充图案，大约 120 个直线样式线型和 70 个位图点样式图像。可以在任何可以创建位图的应用程序中创建位图图像，例如 MS Paint 或 Paint Shop Pro。尽管事实上对图像没有大小限制，不过 MapXtreme 2005 显示图像的能力取决于可用的内存。图像不一定必须是方形，而且还可以具有最多 24 位颜色深度。要确保图像以所需的高度和宽度显示，请选择 BitmapPointStyles 的“按实际大小显示”选项。图像创建后将其放置在 CustSymb 目录中。定制符号位于 C:\Program Files\Common Files\MapInfo\MapXtreme\6.x\CustSymb 文件夹。

此外，MapXtreme 2005 包含了允许快速将这些符号添加到窗体或 web 应用程序的样式控件和对话框。请参阅 Samples 文件夹下的 ChangeStyles 和 FeatureStyles 样本应用程序，该文件夹还包含了本章中讨论的所有样式类。有关窗体样式控件和对话框的详细信息请参阅第 129 页第 8 章中的 *MapInfo.Windows.Dialogs* 命名空间。有关 web 控件的详细信息请参阅第 6 章：运用 *MapXtreme 2005 Web* 控件。

有关支持的样式元素的可视化表示，请参阅附录 I：样式查找。

StyleFactory

StyleFactory 类可从 MapInfo.Style 中获取，而 MapInfo.Style 包含用于从各种样式参数的类型生成 MapInfo Style 对象的方法。例如，MapInfo.Style.FromMBstring 将 MapBasic 样式子句作为字符串输入并将其返回为 CompositeStyle。有关更多信息和代码示例，请参阅联机开发人员参考。

样式说明

AreaStyle

AreaStyle 类包含了绘图区域所用的样式属性。区域的绘制是通过 BaseLineStyle 和 BaseInterior。

BitmapPointStyle

MapInfo BitmapPointStyle 类包含了使用定制位图 () 绘图点的样式属性。使用该类确定点的位置。BitmapPointStyle 是支持的三种类型点样式之一，另外两种分别是 FontPointStyle 和 SimpleVectorPointStyle。

BitmapPointStyle 具有 ShowWhiteBackground 属性；如果设置为 false，则位图中的白像素为透明。默认情况下，ShowWhiteBackground 被设置为 false。例如，您或许想使用公司标识来表示全球办事处的位置，却不想遮盖即时区域内其它地图图元。其它设置可以控制 BitmapPointStyles 如何显示。有关更多详细信息请参阅联机帮助中的 BitmapStyles 枚举。

MapXtreme 2005 包含可指导您入门的大量位图点样式。它们位于 CustSymb 目录中。此外也可将自己的位图图像添加到该目录。允许包含图像的最大数目是 32,767。

CompositeStyle

CompositeStyle 类包含了所有样式类型的集合，用于默认样式、修饰符样式和图层覆盖样式。CompositeStyle 也可以用于描述 Collection 对象类型的样式。包含的样式类型是 AreaStyle、BaseLineStyle 派生类、TextStyle、BasePointStyle 派生类、RasterStyle 和 GridStyle。CompositeStyle 可以用任一或所有这些类型构造，但是必须至少包含一个样式类型。

例如，开发人员可以创建一个样式覆盖 (FeatureOverrideStyleModifier) 来改变图层中所有图元的外观。由于单一图层可包含点、线和区域，所以在生成样式覆盖时或许要指定点、线和区域样式。可以指定某一 CompositeStyle 对象中需要的所有样式类型，然后传递到 FeatureOverrideStyleModifier 构造函数。

SimpleInterior

MapInfo SimpleInterior 类包含了用于填充区域内部的样式属性。SimpleInterior 属性包括图案、前景和背景颜色、背景透明度。SimpleInterior 默认为白色实心内部图案。

Font

MapInfo Font 类包含了绘图文本所用的样式属性。字体属性包括黑体、斜体、下划线、删除线、阴影、光晕、全部大写、两倍行距、大小、前景和背景颜色。用户还可以更改字形（例如，Arial、Times New Roman）和字体大小。注意，轮廓字体属性已删除。要创建轮廓，请使用具有黑色背景颜色的光晕。

FontPointStyle

FontPointStyle 类包含了样式属性，主要使用 MapInfo.Styles.Font 类用于绘制点。用户可以定制点大小、字体颜色、旋转角度及其它字体属性。允许的最大点大小为 240 点。

GridStyle

GridStyle 是帮助程序类，包含与网格有关的显示样式信息（例如，颜色变化和空单元格颜色/透明度）。网格是连续色阶的地图，表示内插的数据值。有关网格的详细信息请参阅[第 17 章：运用光栅和网格](#)。

RasterStyle

RasterStyle 是帮助程序类，包含光栅图像的显示样式信息（包括亮度、对比度、灰度、透明度（颜色，开 / 关）以及半透明度）。有关光栅图像的详细信息请参阅[第 17 章：运用光栅和网格](#)。

Hillshade

Hillshade 是用于存储网格上 hill 底纹参数的帮助程序类。Hill shading，也称作浮雕底纹，可被添加到网格地图，以显示地图上的光源。这就为网格地图提供了更加广义的定义，对提升地图效果尤为有用。Hill shade 属性包括光源的水平和垂直角度，及垂直比例因子。有关网格的详细信息请参阅[第 17 章：运用光栅和网格](#)。

Inflection

Inflection 用于保存将颜色与值关联的单一变化点。网格具有表示其颜色变化的数组。网格地图是显示分级颜色在区域内更改的地图。变化产生了一个颜色与下一颜色的混合。有关网格的详细信息请参阅[第 17 章：运用光栅和网格](#)。

SimpleLineStyle

SimpleLineStyle 类包含了基于 MapBasic Pen 子句用于绘制折线的样式属性。该类用于地图图元，例如街道、电缆线路和区域周围的边界。描述 SimpleLineStyles 的属性包括线型、宽度（以像素或点为单位）和颜色。SimpleLineStyle 的默认设置是 1 像素宽的黑实线。SimpleLineStyle 的单位是像素（默认）或点。

MapInfo.Styles 命名空间中的 LineWidth 类是用于定义直线样式宽度和单位的帮助程序类。

BasePointStyle

这是所有 MapInfo 点样式的抽象基类。该类不能被实例化。SimpleVectorPointStyle、BitmapPointStyle 和 FontPointStyle 从该类派生。

BaseLineStyle

这是所有 MapInfo 直线样式的抽象基类。该类不能被实例化。SimpleLineStyle 从该类派生。

BaseInterior

这是所有 MapInfo 内部样式的抽象基类。该类不能被实例化。SimpleInterior 从该类派生。

StockStyles

该类包含了创建各种默认样式对象的静态方法，这些默认样式对象包括黑色、蓝色、红色和白色内部图案，黑色、蓝色和红色直线，空心内部方案和直线，默认字体和点样式。

```
SimpleLineStyle redLine = StockStyles.RedLineStyle( );
```

TextStyle

TextStyle 类包含用于绘制文本的样式属性。还包含标注线的 MapInfo.Styles.Font 类和 BaseLineStyle 派生类。BaseLineStyle 是可选的（TextStyle 可以包含也可以不包含一个这种线型）。

SimpleVectorPointStyle

该类包含使用 MapInfo 3.0 兼容专有字体用于绘制点的样式属性（MapInfo.fnt 随 MapXtreme 2005 一起交付）。SimpleVectorPointStyle 属性包括了要为点绘制的实际符号的颜色、点大小和形状码。标准集包括符号 31 至 67。

注：另一个符号字体集称为 MapInfo Symbol，是使用 FontPointStyle 类显示的 TrueType 字体集。

预定义的样式和 StyleRepository 类

MapXtreme 2005 包含了大量位图图像，这些图像中含有大量可用作位图点样式的主题。此外还有 170 多种填充图案和直线样式可以使用。这些内容均通过应用程序来安装，且可以通过样式对话框（例如 LineStyleDlg）或各种 StyleRepository 类来访问。

有关支持的样式元素的可视化表示，请参阅附录 *I*: 样式查找。

StyleRepository 类

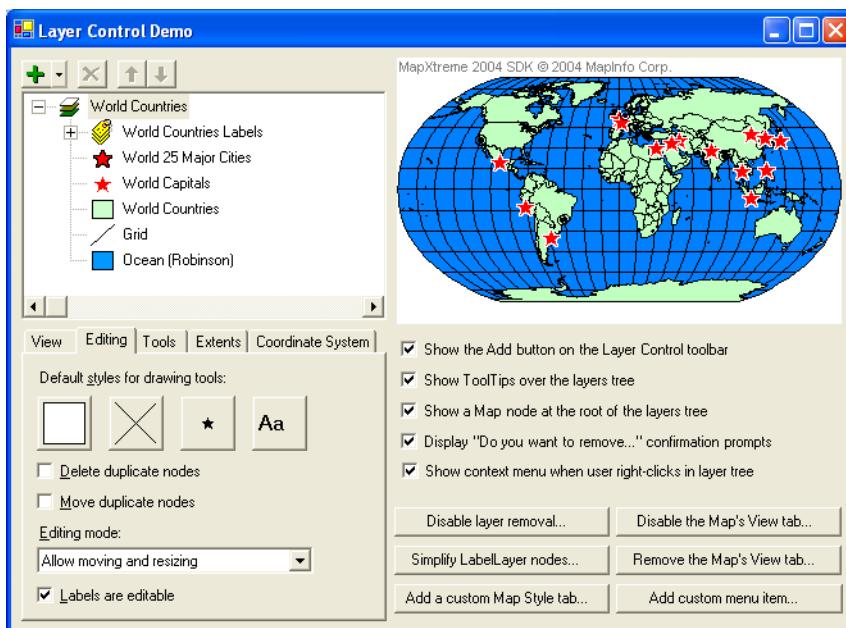
StyleRepository 类包含样式集合类（VectorSymbolRepository、BitmapSymbolRepository、LineStyleRepository、InteriorStyleRepository），这些类允许用户重复当前所有样式，并通过特定文件或目录中的新样式重新加载集合。StyleRepository 类还包含一个库（TrueTypeFontInfoRepository），它表示关于系统中已安装的 TrueType 字体的信息。

VectorSymbolRepository 表示一组源于 MapInfo 3.0 兼容符号集的符号。BitmapSymbolRepository 表示目前在 CustSymb 目录中的图像集合。LineStyleRepository 表示目前直线样式可以使用的线型集合。此外，InteriorStyleRepository 表示可用的内部图案集合。

使用样式

样式和图层控制

可以在设计阶段或运行时使用 LayerControl 修改和覆盖样式。有关 MapInfo.Windows.Controls.LayerControl 对象模型的介绍，请参阅样本应用程序 LayerControl。该代码示例直接在窗体上使用 LayerControl 对象，未使用相关的 LayerControlDlg 对话框进行说明，而使用独立的类。因为 LayerControlDlg 类公开 LayerControl 属性，所以该代码示例中说明的每个操作也可以应用到 LayerControlDlg 对象。



有关 LayerControl 的讨论请参阅第 129 页第 8 章中的 *MapInfo.Windows.Dialogs* 命名空间和第 19 章：*Workspace* 管理器。

创建定制位图样式

可以在任何可以创建位图的应用程序中创建位图图像，例如 MS Paint 或 Paint Shop Pro。尽管事实上对图像没有大小限制，不过 MapXtreme 2005 显示图像的能力取决于可用的内存。图像不一定必须是方形，而且还可以具有最多 24 位颜色深度。要确保图像以所要的高度和宽度显示，请选择 BitmapPointStyles 的“按实际大小显示”选项。图像创建后将其放置在 CustSymb 目录中。

覆盖样式

通过将新样式保存到表，可以永久更改图元的样式。通过用另一个样式覆盖当前样式，也可以更改图元的样式。这不会永久更改样式，而是只更改当前显示的样式。例如，范围主题覆盖给其加底纹的区域对象的样式。有关详细信息，请参阅第 231 页第 13 章中的[使用主题和图例](#)。

也可以覆盖标注的样式。本节介绍了图元的主要样式覆盖类。有关本章中图元和标注的详细信息，请参阅第 219 页中的[图层](#)和第 223 页中的[标注](#)中的 Mapping 命名空间。

FeatureOverrideStyleModifiers

该类实现了 FeatureStyleModifier，从而提供覆盖图元的样式功能。它的 Style 属性是组合样式对象，用于指定哪部分的图元样式要覆盖。

代码示例：FeatureOverrideStyleModifier

以下示例说明了如何使用 FeatureOverrideStyleModifier 和图层 FeatureStyleModifiers 来更改地图内各种图元的样式。

在 \Sample\Features\ChangeStyles.vbproj 的以下代码片断中，我们要用一个黄色符号来覆盖世界首都图层，但点大小不变。

VB 示例：

```
'Get the layer we want
Dim _lyr As FeatureLayer = Me.mapControll.Map.Layers("worldcap")

'Create a sparse point style
Dim vs As MapInfo.Styles.SimpleVectorPointStyle = New _
SimpleVectorPointStyle

'Just change the color and code and attributes flag to indicate that
vs.Code = 55
vs.PointSize = 25
vs.Color = System.Drawing.Color.Red
' vs.Attributes = StyleAttributes.PointAttributes.Color |
' StyleAttributes.PointAttributes.VectorCode;

' And apply to the layer
Dim fsm As FeatureOverrideStyleModifier = New _
FeatureOverrideStyleModifier(Nothing, New _
MapInfo.Styles.CompositeStyle(vs))
_lyr.Modifiers.Append(fsm)
Me.mapControll.Map.Zoom = New MapInfo.Geometry.Distance(6250, _
MapInfo.Geometry.DistanceUnit.Mile)
End Sub
```

16

空间对象和坐标系

本章介绍了 MapXtreme 2005 几何体命名空间并提供编写应用程序以创建和操控几何体对象的说明和示例。

本章内容

- ◆ *MapInfo.Geometry* 命名空间简介 280
- ◆ 几何体 280
- ◆ 将 *FeatureGeometry* 包括在地图中 289
- ◆ 坐标系 290

MapInfo.Geometry 命名空间简介

MapInfo.Geometry 命名空间用于创建和操控几何体对象以及这些几何体对象使用的坐标系。Geometry 对象在地图中用于表示某一点（例如城市，用 point 对象表示）、边界线（例如国家边界，通过 MultiCurve 对象表示）和区域（例如国家或邮政编码地区，通过 MultiPolygon 对象表示）。

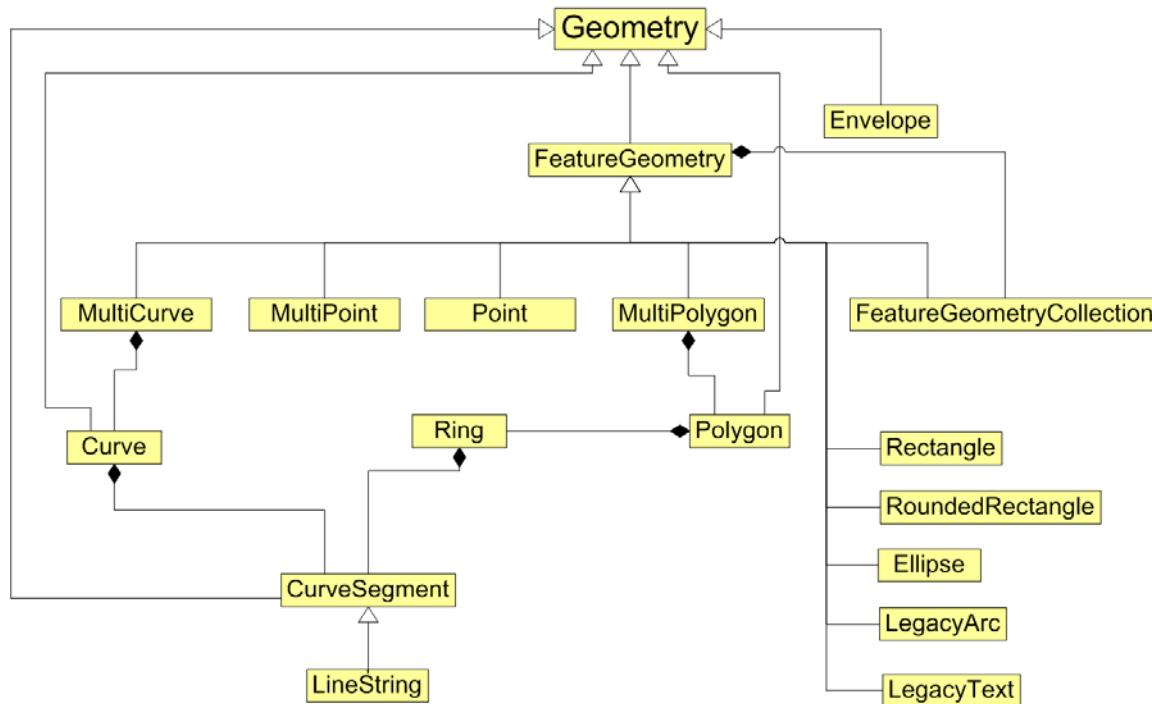
MapInfo.Geometry 命名空间中的类、接口和枚举定义了若干类型，用于表示用于显示地图上地理图元的几何体和坐标系。接口允许创建和编辑几何体对象。诸如 Buffer、Combine、Difference 和 Intersection 这样的方法可以在单一对象或成对的对象上提供对象处理。

几何体

Geometry 类允许创建、编辑和对几何体对象的其它操控。从 Geometry 类继承和表示 Geometry 对象类型的类包括 Point、MultiPoint、Polygon、MultiPolygon、Curve、CurveSegment、LineString 和 Ring。以下旧式类也从 Geometry 类继承：Rectangle、RoundedRectangle、Ellipse、LegacyArc 和 LegacyText。

Geometry 类表示 MapInfo Geometry 对象模型的最高级别。这是抽象类且不能被实例化。所有从该类派生的类包含与其坐标系有关的信息。所有类都能够制作其自身的副本，并可以比较自身与其它 Geometry 对象是否相等。

下图说明了新 Geometry 模型的 UML 表示。



Geometry 对象

所有 MapXtreme 2005 中的几何体对象都用无法更改的特定坐标系创建。如果需要改变对象的坐标系，则可以在新的坐标系中生成该对象的副本。

所有 Geometry 对象包含了用于检索到编辑器的接口的方法，该编辑器将对象置入编辑模式。一旦编辑完成，需要调用 `EditingComplete()` 方法来表示对象的编辑已完成。

`EditingComplete()` 方法被调用时，Geometry 包含的对象顺序重新排序且到对象的所有引用被丢弃，要再次访问这些对象时所有引用需要重新建立。

例如，用户创建 **MultiPolygon**，然后编辑 **MultiPolygon**。如果用户无意间将内部环的节点移动到其包含的 **Polygon** 外部，则 **Polygon** 不再有效。`EditComplete` 被调用时，改组 **MultiPolygon** 内包含的所有对象来修复该问题。

MapXtreme 2005 对象模型中的几何体对象在以下章节中说明。

FeatureGeometry 对象

FeatureGeometry 类是专为包含可以放置到表中且可以是 Features 和 FeatureCollections 一部分的类而设计。要在地图中显示的某些内容必须位于表中。FeatureGeometry 对象通过定义包括在表中。如果对象是 Geometry 的子类而不是 FeatureGeometry 的子类，则该对象无法保存到表或作为 Feature 或 FeatureCollection 的一部分包括。如果要尝试这种操作，则抛出异常或程序不能编译。像 FeatureGeometry 类等 Geometry 类是抽象类，无法被实例化。

Point

Point 从 FeatureGeometry 类派生且表示地图上的某个点。Point 可以包括在 MultiPoint 集合内，然后作为一个整体操作。

使用以下代码示例对创建 Point 建模：

```
using MapInfo.Geometry;
using Mapinfo.Design.Windows;

CoordSys longLatNad83;
CoordSysFactory coordSysFactory = new CoordSysFactory();
longLatNad83 = coordSysFactory.CreateLongLat
    (MapInfo.Geometry.DatumID.NAD83);
DPoint point = new DPoint(0.0, 0.0);
Point pointGeometry = new Point(LongLatNad83, point);
```

MultiPoint

MultiPoint 包含了无序和不连续的 Point 集且可用于在多个点上执行多个操作。

使用以下代码示例对创建 MultiPoint 对象建模：

```
using MapInfo.Geometry;

CoordSys longLatNad83;
CoordSysFactory coordSysFactory = new CoordSysFactory();
longLatNad83=coordSysFactory.CreateLongLat
    (MapInfo.Geometry.DatumID.NAD83);
MultiPoint multiPointGeometry = new MultiPoint
    (longLatNad83, pointArray);
```

其中 pointArray 是 DPoints 的数组。

MultiCurve

MultiCurve 类从 FeatureGeometry 类派生并包含一组可能断开的 Curve。这些 Curve 可以用多种方式交互，也可以连接或断开，还可以彼此相交或重叠。

尽管 Geometry Object Model 支持每个 Curve 包含多个 CurveSegment，但当前版本的 MapInfo 引擎对于 FeatureGeometry 的一部分（即，MultiCurve）的每个 Curve 仅限于包含一个 CurveSegment。该限制从当前的 TAB 文件格式派生，对于本版本的 MapInfo，这些限制的绝大部分保持不变。因此，该限制仅涉及 FeatureGeometry 对象。

构造 MultiCurve 时，构造函数获取一个或多个 Curve，每个 Curve 都可能包含多个 CurveSegment，则改变包含在构造的 MultiCurve 中的实际 Curve 使得每个 Curve 始终仅包含一个 CurveSegment。目前，现有 CurveSegment 的类型只有 LineString。包含多个 LineString CurveSegment 的 Curve 将 LineString 组合以形成一个大的 LineString。

通过调用 EditingComplete() 表示的编辑完成时，添加到 MultiCurve 中且包含多个 CurveSegments 的任何 Curve 按照与上述类似的方式改变，从而生成包含单一 CurveSegments 的 Curves。MultiCurves 中包含的 Curves 始终仅包含一个 CurveSegment 的限制会像引入新类型的 CurveSegments（例如 EllipticalArcs、CircularArcs 和 Splines）一样从以后版本的 MapInfo 中去除，且 TAB 文件格式也会改变。此外，在构造期间且编辑完成时，会从 MultiCurve 自动删除任何空的 Curve。

与编辑完成时传递到构造函数的 Curve 数组相比较，MultiCurve 中 Curve 的顺序可以在构造期间改变。因此，编辑完成后（即，调用 EditingComplete() 后），空 Curve 的删除，当前实现中的限制以及编辑之前和期间对包含在 MultiCurve 中 Curve 的任何引用不再有效。如果引用这些对象，则抛出 ObjectDisposedException。编辑完成后，部分 FeatureGeometry 应该重新获取以获得有效的引用。

Line 对象由存在于 MapInfo TAB 文件中的两个点组成并成为 MultiCurve FeatureGeometry 对象。通过使用 MultiCurve 的 IsLegacyLine 属性，可以作为两点 Lines 来检测这些对象。

```
using MapInfo.Geometry;

MultiCurve multiCurve =
    <... FeatureGeometry obtained from TAB file...>;
if(multiCurve.IsLegacyLine){
    ...
}
```

创建了一行代码，使用 MultiCurve 类中的 CreateLine 方法。以下代码是使用该方法的示例：

```
using MapInfo.Geometry;

MultiCurve multiCurve = new MultiCurve(longLatNad83);
multiCurve = MapInfo.Geometry.MultiCurve.CreateLine
    (longLatNad83, dpoints[0], dpoints[1]);
```

创建和编辑 MultiCurve 的示例

```
using MapInfo.Geometry;

// construct a MultiCurve
MultiCurve multiCurve = new MultiCurve(longLatNad83);
// get a reference to one of the Curves
Curve curve1 = multiCurve[0];
// put MultiCurve into Edit Mode
IMultiCurveEdit multiCurveEditor = multiCurve.MultiCurveEditor();
// add new empty curve
Curve newCurve = multiCurveEditor.CreateCurve(true);
multiCurve.EditingComplete();
// at this point, newCurve is removed from multiCurve, because it is
// empty. curve1 is also no longer a valid reference
// It still exists in the Curve and can be retrieved again.
```

LineStrings

LineString 是以线性方式连接的连续点的有方向集合。LineString 中任意两个连续点都由直线连接。LineString 可以是 Curve 或 Ring 的一部分，也可以作为独立的 Geometry 存在。作为 Curve 或 Ring 一部分的 LineString 继承其容器的坐标系。独立的 LineStrings 可以无内容。如果 LineString 包含在未处于编辑模式的 Curve 或 Ring 中，则 LineString 不可以无内容，且必须至少包含两个点。

使用以下代码示例建模 LineString 的创建和编辑。

```
using MapInfo.Geometry;

longLatNad83 = coordSysFactory.CreateLongLat
    (MapInfo.Geometry.DatumID.NAD83);
LineString lineString = new LineString(longLatNad83, dPoints);
ILineStringEdit edit = lineString.LineStringEditor;
edit.Append(new DPoint(-75, 43));
lineString.EditingComplete();
```

dPoints 是先前已经声明的 Points 数组。

Rectangle

Rectangle Geometry 包含 2 个点，用于表示 Rectangle 的左下角和右上角。其它 2 个点是隐含的。Rectangles 始终为轴对齐，始终显示为矩形形状（与坐标系无关），且不投影。也不包含由坐标系表示的任意弯曲。

使用以下代码示例对创建 Rectangle 建模：

```
using MapInfo.Geometry;

CoordSys longLatNad83;
longLatNad83 = coordSysFactory.CreateLongLat
    (MapInfo.Geometry.DatumID.NAD83);
DRect rect = new DRect(-40.0, 60, -35.0, 65.0);
coordSysFactory = new CoordSysFactory();
longLatNad83 = coordSysFactory.CreateLongLat
    (MapInfo.Geometry.DatumID.NAD83);
Rectangle rectangle = new Rectangle(longLatNad83, rect);
```

RoundedRectangle

Rounded Rectangle 行为与 Rectangle 非常类似，不过在仅显示图元的阶段，这些矩形的角显示为圆形。这些角显示为四分之一圆且圆的半径由 CornerRadius 参数控制。

因为 RoundedRectangle 对象像矩形对象一样由 2 个点定义，且始终显示为轴对齐且未投影，所以主要将其设计用于外观显示。如果多项操作可以内部使用 Rectangle 对象（例如 Combine），则 Rectangle 的 MultiPolygon 副本用于这些操作。生成的 MultiPolygon 包含 5 个点（第一个点与最后一个点相等）且受坐标系影响。在某些实例中，转换的 Rectangle 可以不再显示为矩形。使用 CreateMultiPolygon 方法将 RoundedRectangle 转换为 FeatureGeometry 对象。

使用以下代码示例对创建 RoundedRectangle 建模：

```
using MapInfo.Geometry;

DRect rect = new DRect(-40.0, 60, -35.0, 65.0);
double radius = 20;
DistanceUnit unit = DistanceUnit.Mile;
DistanceType type = DistanceType.Spherical;
coordSysFactory = new CoordSysFactory();
longLatNad83 = coordSysFactory.CreateLongLat
    (MapInfo.Geometry.DatumID.NAD83);
RoundedRectangle roundedRectangle = new RoundedRectangle
    (longLatNad83, rect, radius, unit, type);
```

Ellipse

Ellipse 包括在由 DRect 定义的轴对齐矩形中。DRect 由 2 个点定义，即，矩形的两个对角，其中隐含矩形的其它 2 个角。与坐标系无关，Ellipse 显示为不投影，还可以显示为可以由坐标系表示的任何倾斜。

因为 Ellipse 对象由 2 个点定义，且始终显示为轴对齐且未投影，所以主要将其设计用于外观显示。如果多项操作可以内部使用 Ellipse 对象，则 Ellipse 的 MultiPolygon 副本用于这些操作。生成的 MultiPolygon 会受到坐标系影响，且在某些情况下可能显示的不再是完美的椭圆。

使用以下代码示例对创建 Ellipse 建模：

```
using MapInfo.Geometry;
using MapInfo.Design.Windows;

CoordSys longLatNad83;
coordSysFactory = new CoordSysFactory();
longLatNad83 = coordSysFactory.CreateLongLat
    (MapInfo.Geometry.DatumID.NAD83);
DPoint center = new DPoint(-20.0, 20.0);
double xRadius = 100;
double yRadius = 135;
DistanceUnit unit = DistanceUnit.Mile;
Ellipse ellipse = new Ellipse(longLatNad83, center, xRadius, yRadius,
    unit, DistanceType.Spherical);
```

LegacyArc

LegacyArc 对象是 Ellipse 的一部分且通过 DRect 定义开始角度和结束角度。构造 Ellipse，使其包括在由 DRect 定义的矩形中。与所用的坐标系无关，包括 Ellipse 的矩形是轴对齐的，且始终显示为矩形。角度以度为单位度量，沿正 X 轴角度为零，而逆时针方向的角度为正。角度仅按十分之一度的分辨率存储，且角度值的范围为 0.0 到 360.0。

因为 LegacyArc 对象由 2 个点（构成 DRect）和角度定义，且始终显示为轴对齐，所以主要将其设计用于外观显示。如果多项操作可以内部使用 LegacyArc 对象，则 LegacyArc 的 MultiCurve 副本用于这些操作。这有时会导致意外结果。

使用以下代码示例对创建 LegacyArc 建模：

```
using MapInfo.Geometry;

CoordSys longLatNad83;
coordSysFactory = new CoordSysFactory();
longLatNad83 = coordSysFactory.CreateLongLat
    (MapInfo.Geometry.DatumID.NAD83);
DRect rect = new DRect(-40.0, 60, -35.0, 65.0);
LegacyArc arc = new LegacyArc(longLatNad83, rect, 30.0, 60.0);
```

LegacyText

LegacyText 对象与 MapInfo Professional 的文本对象等价。如果给定的数据库不支持 Text，则使用这种格式时 LegacyText 对象会丢失。LegacyText 对象用指定的左下角标记点放置在地理尺寸的矩形内。文本的点大小由矩形中最合适放置的内容决定。

LegacyText 对象不完全符合 Geometry 模型。Geometry FeatureGeometry 类上可用的多个方法（例如 Combine）对 LegacyText 不起作用，且会抛出 NotSupportedException。在 Geometry 列的 MapInfo 本地 TAB 文件中不存在 Text 对象。LegacyText 类提供了访问这些对象的方法。有关 LegacyText 对象的特定行为请参阅联机参考。

Geometry 对象

不是 FeatureGeometry 对象的 Geometry 对象需要被转换为地图上显示的对应 FeatureGeometry 对象。大多数 FeatureGeometry 类包含了获取适当 Geometry 对象并创建新 FeatureGeometry 对象的构造函数：

```
using MapInfo.Geometry;

Curve curve = new Curve(csys, lineString);
MultiCurve multiCurve = new MultiCurve(curve.CoordSys, curve);
```

以上代码使用 CoordSys (csys) 和 LineString (lineString) 代码之外定义的参数创建 Curve。然后使用 Curve 的 CoordSys 属性及 Curve 本身创建新的 MultiCurve。

在以上示例中，与从对象创建的所有 FeatureGeometries 一样，因为引用无法共享，所以会创建原始对象的副本。

Curve

Curve 类从 CurveSegmentList 类继承，表示相邻的线性 Geometry。Curve 包含必须保持相邻的 CurveSegment 的集合。该类包括在允许以后扩展的模型中且是 OGC 标准的一部分。

使用以下代码示例对创建 Curve 建模：

```
using MapInfo.Geometry;

DPoint[] points = new DPoint[4];

points[0]= new DPoint(-88.135215,43.998892);
points[1]= new DPoint(-104.875119,43.998892);
points[2]= new DPoint(-120.242895.47.048364);
points[3]= new DPoint(-89.135215 46.998892);

LineString lineString = new LineString(csys, points);
Curve curve = new Curve(csys, lineString);
```

CurveSegment

目前，CurveSegment 只可以是 LineString。该类设计用于以后扩展类似的产品，从而包括 Spline、CircularArc 和 EllipticalArc CurveSegment。Curve 和 Ring 由 CurveSegment 组成。

Ring

Ring 是必须保持相邻且封闭的 CurveSegment 集合。

使用以下代码示例对创建 Ring 建模：

```
using MapInfo.Geometry;

dPoints = new DPoint[102];
dPoints[0] = new DPoint(-109.171279, 49.214879);
dPoints[1] = new DPoint(-109.169283, 49.241794);
...
dPoints[101] = new DPoint(-109.171279, 49.214879);
Ring newRing = new Ring(longLatNad83, CurveSegmentType.Linear,
dPoints);
```

Polygon

Polygon 是 Ring 组成的对象。多边形必须至少有一个 Ring，用于定义 Polygon 的外部边界。其它 Ring 可以包括在该环内部，然后在其中定义 Polygon 中的孔。一旦 Ring 放置在另一个 Ring 的内部，则该对象成为 MultiPolygon。

使用以下代码示例对创建 Polygon 建模：

```
using MapInfo.Geometry;

DPoint[][] points = new DPoint[1][];
points[0] = polyPointArrays[0];
Polygon polygon = new Polygon
    (longLatNad83, CurveSegmentType.Linear, polyPointArrays[0]);
```

将 FeatureGeometry 包括在地图中

几何体创建后，需要将几何体添加到地图中，然后才能显示、选择、标注该几何体，或在该几何体上执行任何与地图有关的其它操作。

```
Public Shared Sub MapInfo_Mapping_HowDoICreateFeatureAddToMap(ByVal  
mapControl1 As MapControl, ByVal connection As MConnection, ByVal x  
As Double, ByVal y As Double)  
    Dim map As Map = mapControl1.Map  
  
    'uses wldcty25 as a template  
    Dim table As Table =  
        MapInfo.Engine.Session.Current.Catalog.GetTable("wldcty25")  
  
    ' create a temp table and add a featurelayer for it  
    Dim coordSys As CoordSys = map.GetDisplayCoordSys()  
    Dim tableInfo As TableInfoMemTable = New  
        TableInfoMemTable("temp")  
        tableInfo.Temporary = True  
  
    ' add Geometry column  
    Dim column As Column  
  
    ' specify coordsys for object column  
    column = New GeometryColumn(coordSys)  
    column.Alias = "MI_Geometry"  
    column.DataType = MIDbType.FeatureGeometry  
    tableInfo.Columns.Add(column)  
  
    ' add style column  
    column = New Column  
    column.Alias = "MI_Style"  
    column.DataType = MIDbType.Style  
    tableInfo.Columns.Add(column)  
  
    Dim pointTable As Table =  
        Session.Current.Catalog.CreateTable(tableInfo)  
  
    ' Set the location and display style of the point  
    Dim Geometry As FeatureGeometry = New  
        MapInfo.Geometry.Point(coordSys, x, y)  
        Dim vStyle As SimpleVectorPointStyle = New  
            SimpleVectorPointStyle(37, Color.Red, 14)  
            Dim cStyle As CompositeStyle = New  
                MapInfo.Styles.CompositeStyle(vStyle)
```

```
'Update the table with the location and style of the new
feature
    Dim cmd As MICommand = connection.CreateCommand()
    cmd.Parameters.Add("Geometry",
MIDbType.FeatureGeometry)
    cmd.Parameters.Add("style", MIDbType.Style)
    cmd.CommandText = "Insert Into temp
(MI_Geometry,MI_Style) values (Geometry,style)"
    cmd.Prepare()
    cmd.Parameters(0).Value = Geometry
    cmd.Parameters(1).Value = cStyle
    Dim nchanged As Integer = cmd.ExecuteNonQuery()
    cmd.Dispose()

    'add the table to the map
    map.Layers.Add(New
MapInfo.Mapping.FeatureLayer(pointTable))
End Sub
```

坐标系

坐标系描述了包含特定对象或对象集合的域。坐标系允许以特定的方式描绘对象或描述的对象。CoordSys 类包含了允许创建、操控和编辑坐标系的方法、属性和接口。

如要创建 Geometries，则在创建对象时指定的特定坐标系中创建。对象无法更改创建 Geometries 的坐标系。它们只可以复制到另一个坐标系。

CoordSys 类简化了对坐标系的创建和操控。

Coordsys 类使用投影文件的 XML 版本 (C:\Program Files\Common Files\MapInfo\MapXtreme\6.5\MapInfoCoordinateSystemSet.xml)。CoordSysFactory 对象包含所有注册的坐标系。可以将一个或多个 XML 文件加载到 CoordSysFactory 对象并通过编程方式将坐标系注册到该对象。该类也允许用户将 CoordSysFactory 保存为新的 XML 格式。

CoordSysFactory 对象包含注册的坐标系。通过加载一个或多个 XML 投影文件或使用 RegisterCoordSys 或 RegisterCoordSysInfo 方法，可以注册 CoordSys 定义。从工厂或从代码 - 代码空间 (EPSG, SRID)、PRJ 字符串、MapBasic 字符串和其它 Factory 创建方法来创建 CoordSys 对象。在 CoordSys 类中也有 Military Grid Reference System 转换方法。

以下代码示例说明了创建 CoordSys 对象的多种不同方式：使用 MapInfo codespace；通过 EPSG；作为 PRJ 字符串的经度 / 纬度；从 MapBasic 字符串；通过 SRID。

VB 示例：

```
Public Shared Sub MapInfoGeometryCreateCoordSys()
    Dim factory As CoordSysFactory = Session.Current.CoordSysFactory

    ' create CoordSys objects from srsName
    Dim csysWGS84 As CoordSys = factory.CreateCoordSys("EPSG:4326")
    Dim csysNAD83 As CoordSys =
        factory.CreateCoordSys("mapinfo:coordsys 1,74")
    Dim csysNAD27 As CoordSys = factory.CreateCoordSys("SRID:8260")

    ' create CoordSys objects from code/codeSpace
    csysWGS84 = factory.CreateCoordSys("4326", CodeSpace.Epsg)
    csysNAD83 = factory.CreateCoordSys("coordsys 1,74", _
        CodeSpace.MapInfo)
    csysNAD27 = factory.CreateCoordSys("8260", CodeSpace.Srid)

    ' create CoordSys objects from user-defined parameters
    Dim dat As Datum = factory.CreateDatum(DatumID.WGS84)
    csysWGS84 = factory.CreateCoordSys(CoordSysType.LongLat, dat, 0, _
        0, 0, 0, 0, 0, 0, nothing)
    dat = factory.CreateDatum(DatumID.NAD83)
    csysNAD83 = factory.CreateCoordSys(CoordSysType.LongLat, dat, 0, _
        0, 0, 0, 0, 0, 0, nothing)
    dat = factory.CreateDatum(DatumID.NAD27ContinentalUS)
    csysNAD27 = factory.CreateCoordSys(CoordSysType.LongLat, dat, 0, _
        0, 0, 0, 0, 0, 0, nothing)

    ' create Long/Lat coordinate system
    csysWGS84 = factory.CreateLongLat(DatumID.WGS84)
    csysNAD83 = factory.CreateLongLat(DatumID.NAD83)
    csysNAD27 = factory.CreateLongLat(DatumID.NAD27ContinentalUS)

    ' create from MapBasic string
    Dim csysRGF93 As CoordSys =
        factory.CreateFromMapBasicString("CoordSys Earth Projection _"
            & "3, 33, ""m"", 3, 46.5, 44, 49, 700000, 6600000")

    ' create from PRJ string
    csysNAD83 = factory.CreateFromPrjString("1, 74")
End Sub
```

下一示例说明了如何将 Geometry 对象从一个坐标系转换到另一个坐标系。

VB 示例：

```
Public Shared Sub MapInfoGeomeTryCoordSys(ByRef coordSys As _
    CoordSys, ByRef points() As DPoint, ByRef alternateCoordSys As _
    CoordSys
    ' All Geometry constructors require a CoordSys parameter
    ' Note that the points array is assumed to be in coordSys
    Dim lineString As LineString = New _
    LineString(coordSys, points)

    ' The Geometry has a reference to the CoordSys used during
    ' construction. Unlike the coordinate data represented by the
    ' points array, the CoordSys' is not copied
    If ReferenceEquals(coordSys, lineString.CoordSys) Then
        Console.WriteLine("Geometry objects hold a reference to _
            the CoordSys used during construction")
    End If

    ' if you want to convert the object to another coordinate
    ' system, you need to make a new copy using one of the copy methods
    If Not coordSys.Equals(alternateCoordSys) Then
        Dim newGeometry as MapInfo.Geometry.Geometry = _
            lineString.Copy(alternateCoordSys)
    End If
End Sub
```

运用光栅和网格

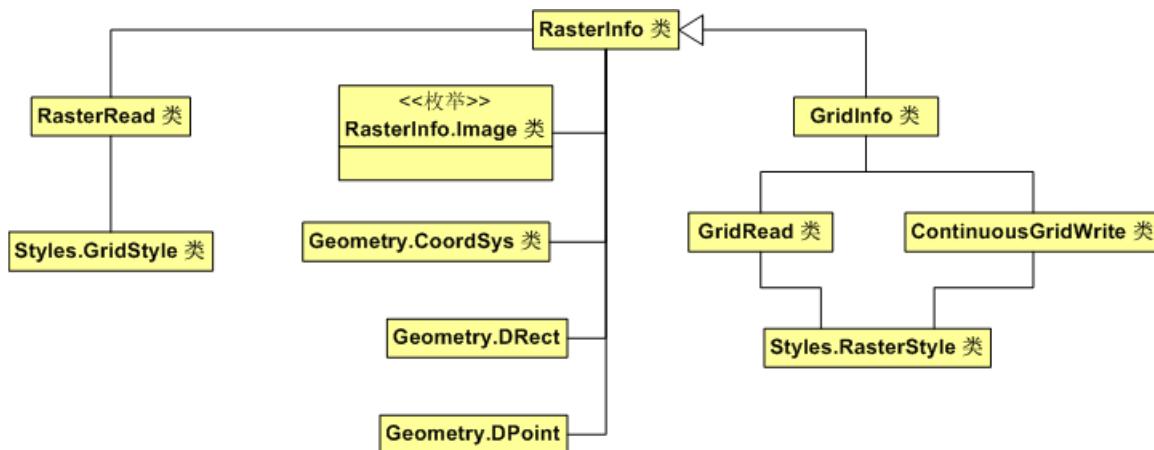
MapInfo.Raster 命名空间包含的类用于对 MapXtreme 2005 中光栅和网格图像的使用和显示进行控制。光栅是由像素组成的计算机图形，由像素渲染整个图像。很多卫星图像渲染为光栅图像。网格图像是主题地图，显示了表示内插值信息的连续色阶。

本章内容：

- ◆ *MapInfo.Raster* 命名空间概述 294
- ◆ 光栅图像 294
- ◆ 光栅句柄 297
- ◆ 配置自定义光栅句柄 298
- ◆ 网格图像 299

MapInfo.Raster 命名空间概述

MapInfo.Raster 命名空间公开了 MapInfo C/C++ Raster 和 Grid Engine API 的全部功能。光栅图像是位图，提供了地图的有用背景和参考图层。网格图像是一种主题，显示了多个图像之间的连续色阶。颜色的过渡表示了对基础数据的解释。网格图像是将数据与它们关联的光栅。常见的示例为剖面地图。光栅不包含任何基础数据。



光栅图像

光栅图像为地图生成了非常好的背景。例如，反映真实世界详细信息（例如建筑物、厂房、植被）的航空照片就非常适合做地图的基本图层。扫描的纸式地图是光栅图像的另一个示例。将光栅图像用作基本图层并重叠诸如街道网络、表示客户的点位置和邮政边界等矢量数据来创建有用且在视觉上吸引人的地图。

与矢量数据一起使用的光栅图像必须注册，以便图像上已知的图形点与矢量数据上相同的图元一致。此外，要显示在 MapXtreme 2005 中的公司徽标和其它美工图形即使不是真实的地理参考数据，也必须注册到地面上的某些位置。目前与注册文件一起提供了很多可用的光栅图像。例如 GeoTIFF、ADRG、ASRP、CADRG 和 CIB。要注册光栅图像，可以将其加载到 MapInfo Professional 中并在其中注册。注册信息存储在 .TAB 文件。

以下是 MapXtreme 2005 中支持的光栅图像格式：

- TIFF 和 GeoTIFF (*.tif)
- MrSID (*.sid)
- ECW (*.ecw)
- Spot (*.bil)
- JPEG (*.jpg)
- JPEG2000 (*.jp2, *.j2K)
- PCX (*.pcx)
- GIF (*.gif)
- Windows Bitmap (*.bmp)
- PNG (*.png)
- Photoshop (*.psd)
- Targa (*.tga)
- Windows Metafile (*.wmf)
- Windows Enhance Metafile (*.emf)
- Vertical Mapper Continuous Grid (*.grd)
- Vertical Mapper Classified Grid (*.grc)
- ADRG - ARC Digitized Raster Graphics (*.gen)
- ASRP - ARC Standard Raster Product (多种文件扩展名)
- CADRG - Compressed ARC Digitized Raster Graphics (*.gen)
- CIB - Controlled Image Base (多种文件扩展名)
- NITF - National Imagery Transmission Format (*.ntf)

如果安装了定制光栅句柄，则系统可以支持其它的光栅格式。

光栅类

光栅图像的主要类是 MapInfo.Raster.RasterInfo 和 RasterRead。样式信息通过 MapInfo.Style.RasterStyle 处理。

RasterInfo 提供了与图像（以像素为单位）的高度和宽度有关的信息，还提供了光栅格式、颜色深度和注册信息。请参阅所提供的 RasterInfo 示例应用程序，该应用程序位于 .\MapInfo\MapXtreme\6.5\Samples\Desktop\Features\RasterInfo。

RasterRead 是为要渲染的图像按顺序读取光栅图像和样式信息的类。

RasterStyle 与光栅的外观有关。可以控制亮度和对比度，用灰度显示彩色图像，设置透明度和半透明度。MapXtreme 2005 支持每个图像一种颜色透明。这意味着图像中该颜色所在的每个位置都是不可见的，且允许显示图像下边的图层。半透明度是整个图像上透明的程度。如果需要显示光栅图像之下的图层，则设置较高的半透明度值（100% 是透明）。

Raster 列是只读的，因此无法永久更改它们的样式。但是，可以通过编程方式设置并获取“图像”的属性，例如亮度、灰度以及半透明度。请参阅联机的开发人员参考中的 RasterStyle 类。

光栅图像和坐标系

将光栅、网格或 WMS 图像显示为地图图层时，MapXtreme 2005 自动设置所有矢量地图图层的旋转和投影，以便使其与光栅图像的旋转和投影匹配。

如果地图包括多个光栅、网格或 WMS 图像图层，则 MapXtreme 2005 在由可见性最好的光栅图像指定的投影中自动显示地图。如果具有不同投影的不同图像成为可见性最好的图像，则坐标系可以随地图视图的更改而更改（由于缩放或平移）。这种情况下，不可以更改地图的显示坐标系。

光栅图像的限制

- 不可以选择光栅图层的任何图元。
- 不可以搜索光栅图层的图元。

代码示例：将光栅图像添加到地图

将光栅图像添加到地图的方法和将任何其它图层添加到地图一样。

C# 示例：

```
Table MyTable = Session.Current.Catalog.OpenTable("MyRaster.tab");
FeatureLayer MyLayer = new FeatureLayer(MyTable);
MyMap.Layers.Add(MyLayer);
```

VB 示例：

```
Dim MyTable As Table =
Session.Current.Catalog.OpenTable("MyRaster.tab")
Dim MyLayer As FeatureLayer = New FeatureLayer(MyTable)
MyMap.Layers.Add(MyLayer)
```

光栅句柄

MapXtreme 2005 具有大量不同的库，可以使用其中一个库来加载光栅图像。通过 MapXtreme 2005 加载光栅图像时，光栅图像搜索这些 DLL 并检查给定的文件是否可以通过该 DLL 读取。DLL/Raster 格式匹配完成之后，MapXtreme 2005 就知道哪个 DLL 处理文件的格式。Format 句柄称为“xxxxxxxx.RHx”。名称的基本部分取决于格式。扩展名始终以 RH 开头，但可以用任意字母 (A-Z) 结尾。搜索格式句柄时，MapXtreme 2005 按字母顺序搜索格式，始于 RHA，止于 RHZ。该过程允许 MapXtreme 2005 区分使用句柄的优先次序。例如，检查任何其它格式之前应该检查 SPOT 文件，因为这些文件只是原始数据，可能与其它格式混淆。SPOT 句柄的扩展名是 RHD。Halo 格式句柄称为 RHV。LEADTOOLS 格式句柄称为 RHX。

MapXtreme 2005 包括了 LEAD Technologies, Inc. 提供的 LEADTOOLS Win32 Pro 和 Media Cybernetics 提供的 HALO Imaging 库。在 MapXtreme 2005 中引用图像时 LEADTOOLS 将整个光栅图像加载到内存中。这意味着图像的加载时间变长，但平移和缩放的速度变快。HALO 仅把需要显示的内容加载到内存中，因此这种方式下图像加载速度较快，但平移和缩放的速度较慢。默认情况下，首先尝试 HALO。也有一些两种库都不能支持的格式，这时就需要它们自己独立的库。MapXtreme 2005 内包括的光栅句柄捆绑在 MapInfo.CoreEngine.dll 内，因此开发人员不可以访问或修改这些光栅句柄。因此，检查句柄的顺序不可以改变。例如，如果检查可以同时处理 Halo 和 LEADTOOLS 的 JPEG，则始终先检查 Halo，这是因为其扩展名 *.rhv 在 LEADTOOLS 的扩展名 *.rnx 之前。

在多线程应用程序（例如 ASP.NET 应用程序）中有几种光栅格式不被支持。这些格式如下所示：

读取	导出
Vertical Mapper GRD、GRC	LeadTools 的 JPEG 2000（仅针对 Win2k）
Halo 的 JPEG（注意，LeadTools JPEG 句柄是线程安全的）	LeadTools 的 TIFF CMYK
Halo TIFF 带 jpeg 压缩 注：LeadTools TIFF 句柄是线程安全的	

配置自定义光栅句柄

可以配置 MapXtreme 2005 应用程序以使用其它光栅句柄（这些光栅句柄不同于 MapXtreme 2005 分发中所包含的光栅句柄）或支持全新的光栅类型。还可以更改光栅句柄所用的优先级。

在 MapXtreme 2005 的默认安装中，所有光栅句柄均放置在

`<program files>\Common Files\MapInfo\MapXtreme\6.x\RasterGridHandlers`。这也是文件 `mirasteru.dll` 的位置。这是应用程序中所用的任何其它光栅句柄的推荐安装位置。若使用默认位置，则不需要任何其它的配置步骤。

若要将自定义光栅句柄放置到非默认位置，则必须在桌面应用程序的应用程序配置文件中或 web 应用程序的 `web.config` 文件中指定自定义光栅句柄的位置。要执行该操作，请在 `<ApplicationDataPaths>` 下面定义 `<Path>` 或 `<SpecialPath>` 元素，然后将所选光栅句柄复制到那个文件夹。

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

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
    <configSections>
        <section name="MapInfo.CoreEngine"
            type="MapInfo.Engine.ConfigSectionHandler,
MapInfo.CoreEngine,
            Version=4.0.0.350, Culture=neutral,
            PublicKeyToken=93e298a0f6b95eb1" />
    </configSections>
    <MapInfo.CoreEngine>
        <ApplicationDataPaths>
            <SpecialPath>
                <Personal>MyAppData</Personal>
            </SpecialPath>
            <Path>c:\MyAppData</Path>
        </ApplicationDataPaths>
    </MapInfo.CoreEngine>
</configuration>
```

在该示例中，`<Personal>` 标记是在用户的“我的文档”中定义的特殊位置。这将会引用“我的文档”中名为 `MyAppData` 的文件夹。在 `<ApplicationDataPaths>` 标记中，使用 `<SpecialPath>` 或 `<Path>` 标记。同时使用这两个标记意味着光栅句柄可能会放置在任一 `"MyAppData"` 文件夹中。虽然这种情况不能算是错误，但也可能不是您想要实现的操作。由于这是非默认的配置，还必须将 `mirasteru.dll` 放置到同一目录。

如果应用程序数据存储在相对于 .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在“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。

该配置方法还可以用来更改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（这两者都已捆绑在 MapXtreme 2005 中）处理。通常，Halo 具有优先权，因为它的 *.rhv 文件扩展名按字母排序位于 LEADTOOLS *.rnx 扩展名的前面。但是，由于 MapXtreme 2005 会先查看在配置文件中定义的任何 <ApplicationDataPaths>，因此，它会先定位指定的光栅句柄，然后再查找默认 [CommonFiles] 文件夹中的句柄。因此，例如，可能会将 LEADTOOLS *.RHX 句柄复制到定义的文件夹中并配置 MapXtreme 2005 首先查找并使用那个句柄。

或者，可以重命名 [CommonFiles] 文件夹中的文件扩展名，以便所需的光栅句柄先出现在字母列表中。但是，这将会影响所有使用 MapXtreme 2005 开发的应用程序并可能产生意外的影响。此外，若按此方式更改了文件扩展名，则在卸载 MapXtreme 2005 时将不会删除这些重命名的光栅句柄。出于上述这些原因，可能想要使用配置方法来更改光栅句柄的位置和优先级。

网格图像

如前所述，网格图像显示了区域之间数据值的内插值。网格图像从数据文件中创建，该数据文件中的数据以平均间距的点度量。整个地图区域转换为网格，其中每个网格单元表示一个值。尽管数据集合点需要均匀间隔，不过网格值不必以生成网格的内插值替换。MapInfo Professional 通过使用网格句柄的内插值来创建网格。

除色阶外，网格图像还可以显示山脉或辅助底纹。辅助底纹允许网格表面根据虚拟光源加上底纹。每个网格单元的亮度与照射在表面上的光线一致，且亮度可以根据对光源的方向来调整。这非常适合于剖面网格地图，在这种地图中可以相对于光线的方向考虑表面倾斜和方向。最大亮度分配在太阳光线与表面垂直的点上。随着斜面转动远离光源，亮度值降低。支持的网格格式包括：

- MapInfo Grid (*.mig)
- USGS DEM (*.dem)
- GTOPO30 (*.dem)

- DTED (*.dt0, *.dt1, *.dt2)
- Vertical Mapper Continuous Grids^{1 2}(*.grd)
如果安装了定制网格句柄，则系统可以支持其它的网格格式。

网格类

网格图像的主要类是 MapInfo.Raster.GridInfo、GridRead、ContinuousGridWrite 和 HillShadeWrite。GridInfo 和 GridRead 类似于 RasterInfo 和 RasterRead，这种情况下可以获得与网格文件有关的信息。使用 ContinuousGridWrite 为数据点的图层创建网格。HillShadeWrite 允许将辅助底纹添加到或改变为网格。网格的很多属性从 RasterInfo 继承，包括图像类、坐标系、光栅控制点和最小边界矩形 (MBR)。

代码示例：将网格图像添加到地图

将网格图像添加到地图的方法和将任何其它图层添加到地图一样。

C# 示例：

```
Table MyTable = Session.Current.Catalog.OpenTable("MyGrid.tab");
FeatureLayer MyLayer = new FeatureLayer(MyTable);
MyMap.Layers.Add(MyLayer);
```

VB 示例：

```
Dim MyTable As Table =
Session.Current.Catalog.OpenTable("MyGrid.tab")
Dim MyLayer As FeatureLayer = New FeatureLayer(MyTable)
MyMap.Layers.Add(MyLayer)
```

-
1. 可以显示为网格或光栅。.TAB 文件确定图像应使用网格句柄还是光栅句柄来绘制。
 2. 不是多线程安全的。

代码示例：从网格地图检索数据

以下示例说明了如何打开并读取网格文件的信息。

VB 示例：

```
Public Shared Sub MapInfo_Raster_GridRead(ByVal strGridFilename _  
As String)  
    Dim strHillshadeFilename As String = _  
        MapInfo.Raster.GridRead.DefaultHillshadeFilename(strGridFilename)  
    Dim session As ISession  
  
    session = MapInfo.Engine.Session.Current  
    Dim gridread As GridRead = New GridRead(strGridFilename, _  
        strHillshadeFilename)  
    Console.WriteLine(gridread)  
End Sub
```

以下示例说明了如何打开并读取网格文件的单元格值。

VB 示例：

```
Public Shared Sub MapInfo_Raster_GridReadStartRead(ByVal _  
gridread As GridRead, ByVal strGridFilename As String)  
    Dim strHillshadeFilename As String = _  
        gridread.DefaultHillshadeFilename(strGridFilename)  
  
    If gridread.StartRead() Then  
        Dim x As Integer = 0      ' TODO - set to a pixel column value  
        Dim y As Integer = 0      ' TODO - set to a pixel row value  
        Dim bIsNull As Boolean  
        Dim dValue As Double  
        If gridread.GetValue(x, y, bIsNull, dValue) Then  
            If bIsNull Then  
                ' read a null cell  
                Console.WriteLine("{0,20}", "NULL")  
            Else  
                ' read a non-null cell, with value == dValue  
                Console.WriteLine("{0,20}", dValue)  
            End If  
        End If  
        gridread.EndRead()  
    End If  
End Sub
```


18

WMS 和 WFS

MapXtreme 2005 提供在服务器上配置 Web 地图服务 (WMS) 和 Web 图元服务 (WFS) 的功能，并使用户可以在多个厂商出品的地图绘制应用程序中使用任意一种服务。

本章内容：

- ◆ *Web 地图服务* 304
- ◆ *Web 图元服务* 314

Web 地图服务

MapXtreme 2005 Web 地图服务 (WMS) 使客户端可以请求，服务器可以传递空间参考地图图像，这些操作可以通过因特网或专有的内联网完成。MapXtreme 2005 提供的工具可以部署 WMS Server 供他人查询，同时支持 WMS 客户端功能与应用程序的结合，实现从 WMS Server 请求图像。WMS 服务器和客户端实现是基于 1.1.1 OpenGIS® Web Map Service 实现规范，该规范的文档位于 <http://www.opengis.org/docs/01-068r2.pdf>。WMS 客户端是基于 WMS 1.0.0、1.1.0 和 1.1.1 OGC 规范构建，并在声明支持这些规范和某些次要修订版（例如，1.0.5）的服务器上进行了成功测试。

基本的 WMS 将保存到图层中的几何引用信息进行分类并提供显示这些图层的预定义样式。符合 OGC 规范的 WMS 还可以支持某些图像格式的透明像素定义。透明像素使用户可以将 WMS 图像用作光栅覆盖而不是单独地作为地图的背景图层。可用数据的数量和质量由 WMS 服务器决定。

本节涵盖了以下信息：

- [设置 WMS 服务器](#) - 提供设置 WMS 服务器所需的步骤。
- [理解 WMS 服务器的特定操作](#) - WMS 服务器或服务所执行操作的概述。
- [使用 MapXtreme 2005 WMS 客户端](#) - MapXtreme 2005 WMS 客户端（用于创建应用程序）的概述。
- [使用 WMS 地图图像](#) - 提供如何使用 WMS 响应和如何操纵或定制 WMS 地图图像的示例。
- [使用 MapInfo Professional 访问 WMS 服务](#) - 提供将 MapInfo Professional 用作 WMS 客户端所需的步骤。

设置 WMS 服务器

可在应用程序中使用 WMS 客户端功能或发送 HTTP 请求之前，必须访问 WMS 服务器。有两种方式可供选择：设置自己的 WMS 服务器或使用现有的 WMS 服务器。要设置自己的 WMS 服务器，则必须配置与 IIS 连接的服务器并创建必要的 XML 文件以提供管理 Web 地图服务所需的数据连接。在本节中假定用户了解 WMS 和 MapXtreme 2005 工作空间构架的基本知识。工作空间方案的信息位于 [附录 G: 理解 MapInfo 工作空间](#) 中。如果用户只对运行 WMS 客户端以访问其它服务的数据感兴趣，请参阅第 306 页中的 [使用 MapXtreme 2005 WMS 客户端](#)。不必创建 WMS 服务器以使用 WMS 客户端实现。

WMS 服务器在 Microsoft Internet 信息服务 (IIS) 内运行。必须配置主机来使用正确版本的 ASP.NET (v 1.1.4322)。然后要运行服务，必须先修改配置文件以包括特定的 MapXtreme 2005 WMS 信息。此外，需要创建有效的配置文件，该文件包含了与要存放的数据有关的信息。

注: 创建 XML 配置文件时, 它必须通过 WMS 构架文件验证 (MXP_WMSConfiguration_1_0.xsd), 否则用户尝试运行 WMS 服务器时将出现错误。

对于在设置和配置 WMS 服务器时所需的步骤, 教程中提供了相关描述, 包括用于设置 WMS 服务器、定制 WMS 服务、使用数据配置服务器和测试服务器实例的所有必需步骤。

理解 WMS 服务器的特定操作

WMS 服务器特定操作有三种, 它们提供了服务器实现的基础。

- **GetCapabilities** - 首次查找地图服务器时, 需要查找正在使用的可用图层名称、样式、空间信息及该服务器提供的其它信息, 以便生成对服务器参数内图像的请求。该操作在因特网或内联网上检索服务级元数据, 包括服务器名称、图层名称、数据摘要和可接受的请求参数。
- **GetMap** - 使用或建立 Web 地图服务器的主要原因是获得或服务于地图。使用 GetMap 请求从 WMS 服务器获得图像。该操作在因特网或内联网上检索附属于请求的图层、样式、边界、空间参考系 (检索的数据的坐标系) 的空间几何信息, 包括请求所需数据的地图图像。也可以指定图像格式 (JPEG、GIF 等) 和样式。

调用 GetMap 时, WMS Client 可以指定:

- 图层
- 样式
- 边界框 (要绘制地图的区域)
- 坐标参考系 (地图的数字表示, 也称为空间参考系)
- 输出格式
- 输出大小 (高度和宽度)
- 背景透明度和颜色
- **GetFeatureInfo** - 成功完成 GetMap 请求后, 用户可能要了解有关地图中所包括图元的详细信息。GetFeatureInfo 操作返回关于地图中图层的信息和每个图层的可查询属性。该操作是可选的。如果使用 GetFeatureInfo, 则必须在发送 GetMap 请求后发布, 因为该地图上点的 X/Y 像素坐标是 GetFeatureInfo 请求的必需输入。

图层和样式

渲染的图像可由分层图层组成 - 可服务的图像, 由其它图层集合中的图层组成。由于 WMS 渲染图层, 因此修改图层视图 (如样式和坐标系) 的特征也可包括在提供的图层中。这样就可以定制图像, 因为每个图像都可以在正访问这些图像的客户端中作为独立的图层。

使用 MapXtreme 2005 WMS 客户端

用户可以通过编程方式使用 MapXtreme 2005 附带的 WMS 客户端，访问符合 OGC 标准的 Web 地图服务的数据。可以将 WMS 客户端添加到应用程序，并通过指定 WMS 服务器的 URL 和需要返回的图层运行 WMS 客户端。WMS 地图请求必须包含以下元素：

1. 应该用于渲染图像的图层。
2. 每个图层应该渲染的样式。
3. 应该用于渲染的坐标系。
4. 地图应该表示的地理区域或边界。
5. 使用的图像格式。图像可以具有指定的背景颜色或具有透明背景颜色。透明度可用时，用于在地图中覆盖图像。不透明图像只可用于作为地图的最底部图层。

在应用程序中为 WMS 提供 URL 并调用一种 MapInfo.Wms.WmsClientUtilities 中支持的方法以连接到服务器：GetCapabilities、GetMap 或 GetFeatureInfo。

GetCapabilities

检索服务级元数据。GetCapabilities 方法有三个参数：url、version 和 userDefinedParameters。**url** 指定其功能正在返回的 WMS 服务器。**version** 可以是请求的单一 WMS 版本号或 WMS 版本号的数组。返回首次成功的版本号尝试。**userDefinedParameters** 可以指定要传送到服务器的名称 / 值对。

VB 方法

```
Overloads Public Shared Function GetCapabilities( _
    ByVal url As String, _
    ByVal versions As String(), _
    ByVal userDefinedParameters As NameValueCollection _
) As ICapabilities
```

GetMap

检索具有定义完善的地理空间和维参数的地图图像。GetMap 方法有五个参数：url、version、exceptionType、userDefinedParameters 和 mapReq。**url** 指定其功能正在返回的 WMS 服务器。**version** 可以是请求的单一 WMS 版本号或 WMS 版本号的数组。返回首次成功的版本号尝试。**exceptionType** 可以指定 WMS 服务器报告异常的格式。**userDefinedParameters** 可以指定要传送到服务器的名称 / 值对。**mapReq** 定义使用 WmsMapRequest 类的地图图像的详细信息。使用 WmsMapRequest 和 WmsMapDescription 类可以指定多个属性，用于定义和定制从服务器返回的地图图像：

- **BGColor** - 地图图像的十六进制红绿蓝背景颜色值（如，0xFFFFFFF）。
- **MimeType** - 要返回的地图图像的格式（如，image/png）。

- StyleList - 各请求图层中渲染样式的逗号分隔列表。每个请求图层都需要一个样式。如果未请求任何样式，则属性必须存在但没有样式值。如果请求多个图层，则所列样式应与图层的顺序相同。可以在列表中将样式保留为空并允许使用为图层定义的默认样式。
- Transparency - 确定地图是否将具有透明背景的布尔值。
- Bounds - 使用 DRect 结构确定要在指定坐标系中返回的地图区域。
- Height - 地图图像的高度（以像素为单位）。
- LayerList - 要在地图图像中包括的图层的逗号分隔列表。
- Srs - 地图中使用的坐标参考系（地图投影的数字表示，也称为空间参考系）。如未指定，则根据图层默认样式中的定义使用默认的坐标参考系。
- Width - 地图图像的宽度（以像素为单位）。

VB 方法

```
Overloads Public Shared Function GetMap( _
    ByVal url As String, _
    ByVal version As String, _
    ByVal mapReq As WmsMapRequest, _
    ByVal exceptionType As String, _
    ByVal userDefinedParameters As NameValueCollection _
) As Byte()
```

GetFeatureInfo

检索在请求的地图上有关特定图元的信息。GetFeatureInfo 方法有五个参数：url、version、exceptionType、userDefinedParameters 和 fiReq。url 指定其功能正在返回的 WMS 服务器。version 可以是请求的单一 WMS 版本号或 WMS 版本号的数组。返回首次成功的版本号尝试。exceptionType 可以指定 WMS 服务器报告异常的格式。userDefinedParameters 可以指定要传送到服务器的名称/值对。fiReq 定义使用 WmsFeatureInfoRequest 类的地图图像和图元搜索的详细信息。使用 WmsFeatureInfoRequest 和 WmsMapDescription 类可以指定多个属性，用于定义和定制图元搜索：

- FeatureCount - 要返回信息的图元的最大数目。默认情况下，只返回一个图元。
- LayerList (WmsFeatureInfoRequest) - 用于搜索图元的图层的逗号分隔列表。
- MimeType - 要返回的地图图像的格式（如，image/png）。
- QueryPoint - 在其中执行图元搜索的地图上的点，以屏幕 X 和 Y 坐标（地图图像左上角的像素单位）表示。
- Bounds - 使用 DRect 结构确定要在指定坐标系中返回的地图区域。
- Height - 地图图像的高度（以像素为单位）。
- LayerList (WmsMapDescription) - 要在地图图像中包括的图层的逗号分隔列表。

- Srs - 地图中使用的坐标参考系（地图投影的数字表示，也称为空间参考系）。如未指定，则根据图层默认样式中的定义使用默认的坐标参考系。
- Width - 地图图像的宽度（以像素为单位）。

VB 方法

```
Overloads Public Shared Function GetFeatureInfo( _
    ByVal url As String, _
    ByVal version As String, _
    ByVal fiReq As WmsFeatureInfoRequest, _
    ByVal exceptionType As String, _
    ByVal userDefinedParameters As NameValueCollection _
) As Byte()
```

使用 WMS 地图图像

由政府、公司和个人组织提供的可用空间几何信息量正不断增长，所以用户可以检索它们以增强地图的精确性和完整性。使用陆地用途和水文用途数据的公司可以从 U.S. Census Bureau 或本地数据提供商获得更多的高程和人口信息。将交通图案数据和存储位置信息组合在一起，可以提供建立附加存储位置的洞察力或优化市场和产品放置成本。作为开发人员，可以通过扩展 WMS 代码来定制包括特定旅馆、名胜或休假目的地的路径请求。用户需要并且能够想到的功能都可以实现。

此外，由于 WMS 信息是作为其自身的光栅图像返回，所以可以使用样式和透明度设置进行操控。该特性使用户可以定制要检索的数据的用户视图。请参阅第 15 章：为地图设置样式。

创建 tab 文件

使用 MapXtreme 客户端功能检索 WMS 图像后，用户可以按需要创建图像的 .tab 文件以便 MapXtreme 可以管理图像的显示。请记住，检索的是静态图像，具有严格的边界。如果需要在应用程序中平移或缩放该光栅图像，.tab 文件使用户可以使用 MapXtreme 2005 中的地图绘制和数据功能操控数据和显示。

从 WMS 光栅图像创建 .tab 文件的方法有两种。可以在 MapInfo Professional 中打开图像并将其保存为 .tab 文件，也可以使用 TableInfo.WMS 通过编程方式创建 .tab 文件。

以下是从检索的 WMS 光栅图像创建 .tab 文件的代码示例。

```
// build the capabilities
ICapabilities capabilities = WmsClientUtilities.GetCapabilities
(url, "1.1.1");

// create the WMS client
WmsClient wmsClient = new WmsClient(capabilities);
wmsClient.AddLayer("WORLD");
wmsClient.Srs = "EPSG:4326";
```

```
wmsClient.BGColor = Color.Blue;
wmsClient.MimeType = "image/gif";

// create the table info
TableInfoWms wmsTableInfo= new TableInfoWms
("MyWmsTable", wmsClient);

// create the table
Table wmsTable = Session.Current.Catalog.OpenTable(wmsTableInfo);

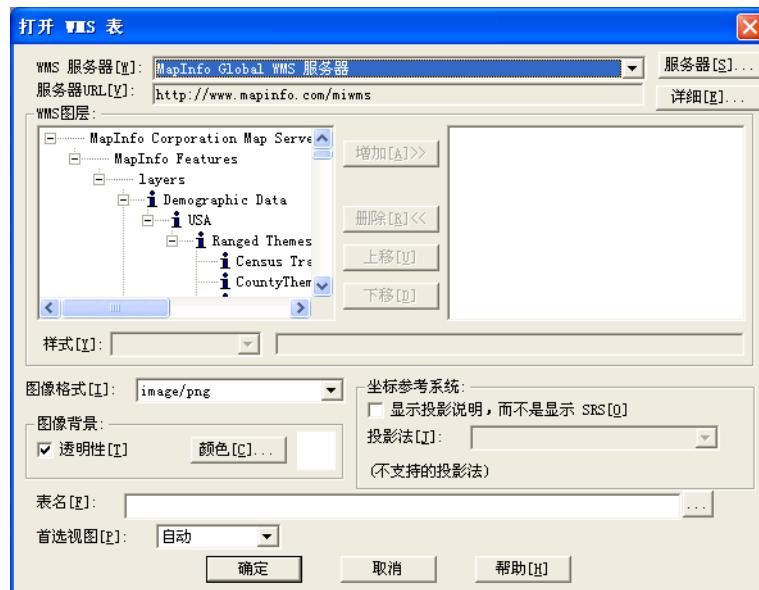
// creates a FeatureLayer from the table entry
FeatureLayer featLyr = new FeatureLayer(wmsTable);
```

注: WMS 的 MapXtreme 2005 实现限制了返回图像的大小, 宽和高都限制为 2000 像素。

使用 MapInfo Professional 访问 WMS 服务

MapInfo Professional 可用作 WMS 客户端, 用于查看或创建由 WMS 服务器提供的图形文件。以下步骤显示了如何将 MapInfo Professional 用作 WMS 客户端并访问 WMS 服务器:

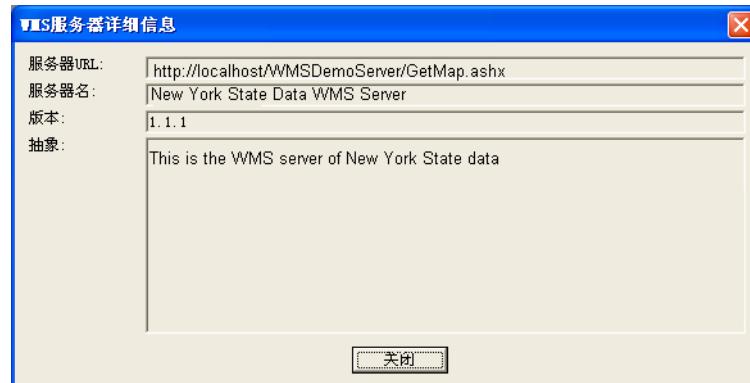
1. 启动 Mapinfo Professional (7.5 或更新版本), 然后使用**文件 > 打开 WEB 服务 > 打开 WMS** 命令。此时显示**打开 WMS 表**对话框。显示整个列表可能需要几秒钟时间, 因为应用程序需要从 web (这种情况下为 MapInfo Global WMS 服务器) 下载默认 WMS 服务器的详细信息。



2. 单击**服务器**按钮编辑可用服务器列表。如果已安装 WMS 服务器，则可以在此处添加服务器。单击**增加**按钮，输入自己的 WMS 服务地址。



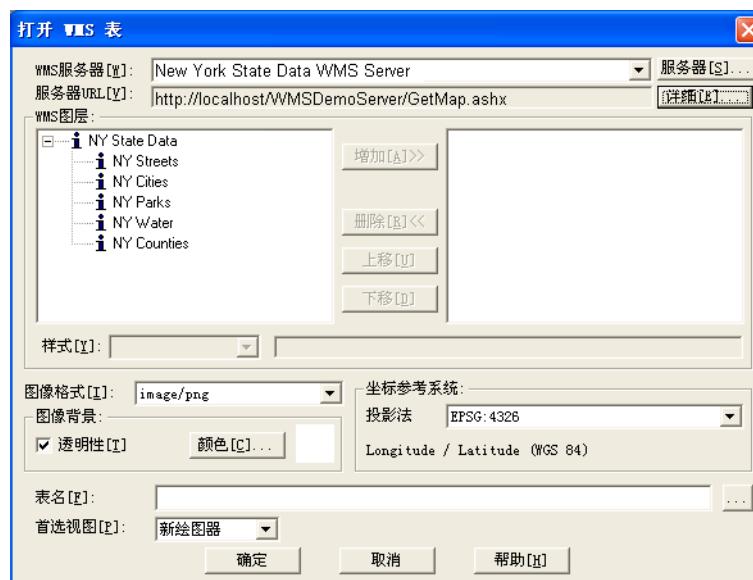
3. 要确保已正确输入 URL 地址，请单击**测试 URL**按钮。如果测试成功，则出现的对话框应与以下对话框类似。注意，“服务器名”和“抽象”与配置文件中提供的名称匹配。



4. 单击**关闭**按钮，返回**WMS 服务器信息**对话框。单击**显示说明**按钮，获得**描述**字段的值。然后单击**确定**。

注：如果想要自己填充**描述**字段，请键入任何所需内容，而无需单击**显示说明**按钮。

此时将返回**WMS 服务器列表**对话框。从可用服务器列表中选择刚刚创建的 WMS 服务器。完成该操作后，会在对话框左侧面板显示在 WMSConfig.xml 文件中指定的图层列表。



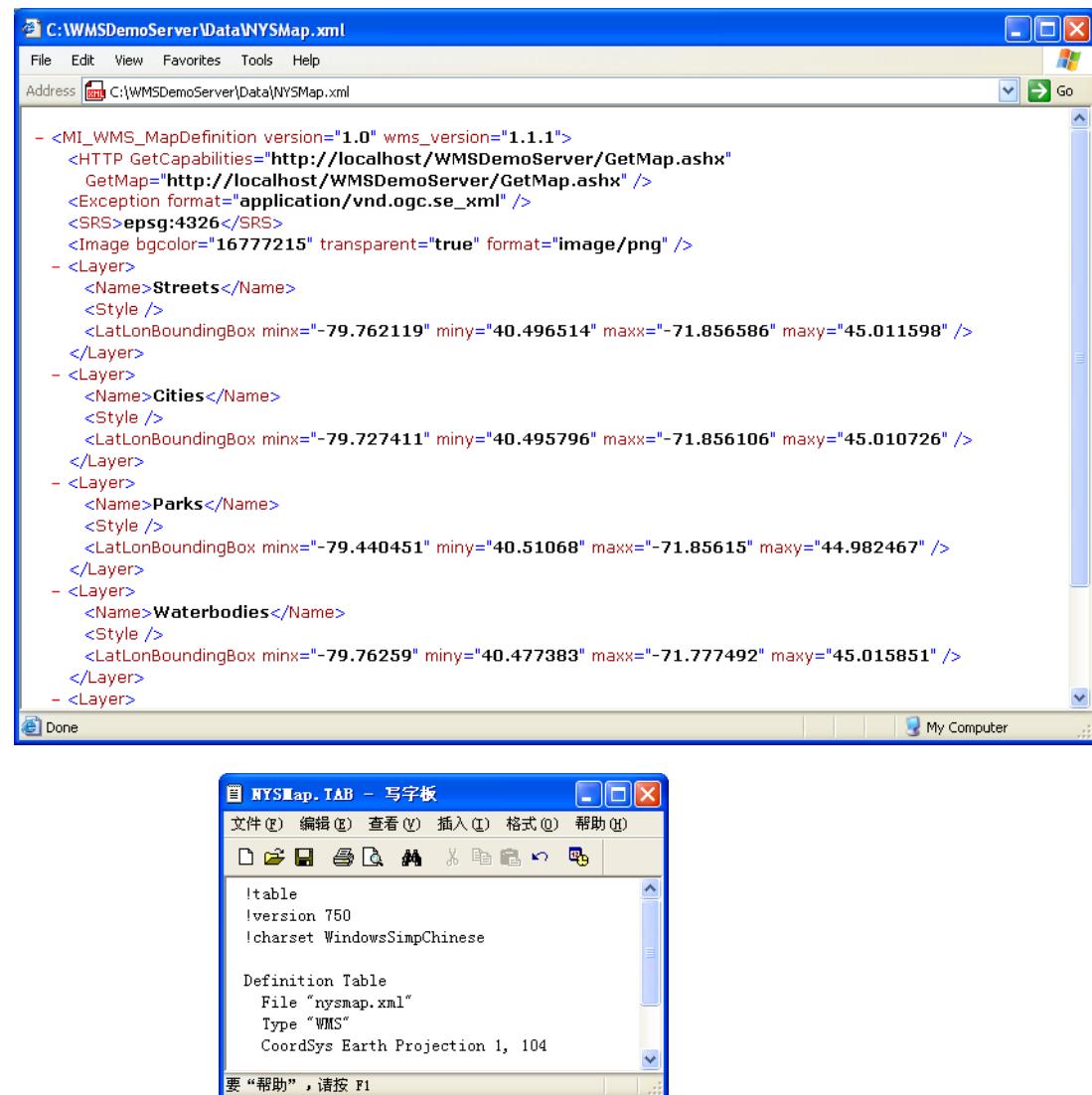
5. 突出显示要包括的图层并单击**增加**按钮。注意，选择每个项目时都会在列表底部显示完整的图层别名。与使用文件名或其它缩写（如 QLD_OC）相比，使用描述名更加简单。可将图层显示为 gif、jpeg、png 或 tiff 图像。

6. 最后, 键入输出 TAB 文件的名称或通过省略号按钮 (...) 浏览选择, 然后单击**确定**按钮。如果所有设置正确, 则会出现一个地图窗口, 显示 WMS 服务器提供的内容(例如, 不是真正的 TAB 文件)。



此时, MapInfo Professional 显示指定为单一图像的图层组合, 该组合在 Map 窗口中作为单一图层。可以多次使用**打开 WMS 表**对话框创建一叠分层图像, 然后创建独立的表。

每次从 WMS 服务器打开图层时，都会在本地保存。本地版本的文件同时包括 TAB 文件和 XML 文件（如下所示）。这些文件包含用于连接和显示该 WMS 图层的所有信息。如果地图内容相同，则无需再次使用**打开 WMS 表**对话框。



Web 图元服务

MapXtreme 2005 提供了 Web 图元服务 (WFS) 实现，以在因特网或内联网上返回空间几何 GML2 (地理标记语言)、GML3 数据和 HTTP GET、HTTP POST 请求。MapXtreme 2005 中的 WFS 根据 1.0.0 OpenGIS® Web 图元服务实现规范开发。该文档位于 <http://www.opengis.org/docs/02-058.pdf>。

WFS 规范包含必需和可选的实现。必需部分称为 WFS Basic，是只读服务。可选部分称为 WFS Transaction，允许添加、修改和删除图元类型中的图元。目前，MapXtreme 2005 中的 WFS 仅实现 WFS Basic。

本节涵盖了以下信息：

- [设置 WFS 服务器](#) - 提供设置 WFS 服务器所需的步骤。
- [理解 WFS 服务器的特定操作](#) - WFS 服务器所执行操作或服务的概述。
- [使用 MapXtreme 2005 WFS 客户端](#) - MapXtreme 2005 WFS 客户端（用于创建应用程序）的概述。
- [使用 MapInfo Professional 访问 WFS 服务](#) - 提供将 MapInfo Professional 用作 WFS 客户端所需的步骤。

设置 WFS 服务器

可在应用程序中使用 WFS 客户端功能或发送 HTTP 请求之前，必须访问 WFS 服务器。有两种方式可供选择：设置自己的 WFS 服务器或使用现有的 WFS 服务器。设置 WFS 服务器的指示非常类似于 WMS 服务器的指示。要设置自己的 WFS 服务器，则必须配置与 IIS 连接的服务器并创建必要的 XML 文件以提供管理 Web 图元服务所需的数据连接。在本节中假定用户了解 WFS 和 MapXtreme 2005 工作空间构架的基本知识。工作空间方案的信息位于[附录 G: 理解 MapInfo 工作空间](#)中。如果用户只对运行 WFS 客户端以访问其它服务的数据感兴趣，请参阅[第 306 页中的使用 MapXtreme 2005 WMS 客户端](#)。不必创建 WFS 服务器以使用 WFS 客户端实现。

WFS 服务器在 Microsoft Internet 信息服务 (IIS) 内运行。必须配置主机来使用正确版本的 ASP.NET (v 1.1.4322)。然后要运行服务，必须先修改配置文件以包括特定的 MapXtreme 2005 WFS 信息。

对于在设置和配置 WFS 服务器时所需的步骤，教程中提供了相关描述，包括用于设置 WFS 服务器、定制 WFS 服务、使用数据配置服务器和测试服务器实例的所有必需步骤。

理解 WFS 服务器的特定操作

WFS 服务器特定操作有三种，它们提供了服务器实现的基础：GetCapabilities、DescribeFeatureType 和 GetFeature。

- **GetCapabilities** - 发送 Web 图元请求时，可以访问 WFS 服务的多个实例，根据可用图元数据的不同，每个实例可能支持不同的地理区域和功能类型。将请求发送到 Web 图元服务实例之前，有必要浏览可用的功能和数据。Web 图元服务能够返回该服务级元数据。该元数据是服务的信息内容和可接受请求参数的说明。GetCapabilities 通过 HTTP GET 和 HTTP POST 支持。VendorSpecificCapabilities 元素为可选，不会返回。

MapXtreme 2005 支持以下空间运算符：

- BBox
- Equals
- Disjoint
- Intersects
- Within
- Contains

MapXtreme 2005 支持以下非空间运算符：

- 逻辑运算符：AND、OR 和 NOT
- PropertyIsEqualTo
- PropertyIsGreater Than
- PropertyIsGreaterThanOrEqualTo
- PropertyIsLessThan
- PropertyIsLessThanOrEqualTo
- PropertyIsNotEqualTo

MapXtreme 2005 支持以下简单的算术运算符：

- Add
- Div
- Mul
- Sub

- **DescribeFeatureType** - 找到 GetCapabilities 请求中的可用图元后，请使用 DescribeFeatureType 请求获得关于一个或多个图元的详细信息。此类请求定义要如何在输入上对图元实例进行编码（描述）以及如何在输出上生成图元实例。结果以 XML 构架文档返回，该文档表示特定图元类型上所有查询的格式。在 MapXtreme 2005 中，图元类型为 FeatureLayer。Schema 的格式不用 WFS 规范定义，而是完全取决于实现。

DescribeFeatureType 通过 HTTP GET 和 HTTP POST 支持。返回的构架将与 WFS 1.0.0 规范中描述的示例具有相同形式。根据规范里的规定，只要构架为 GML2，就可以采用任何形式。在方案采用的形式中，图层中每一列都可以成为元素。

MapXtreme 2005 不返回以下列类型：

- MIDbType.Binary
- MIDbType.CoordSys
- MIDbType.Grid
- MIDbType.Key
- MIDbType.Raster
- MIDbType.Style
- **GetFeature** - GetFeature 请求允许 WFS 服务器查询一个或多个图元类型并返回满足查询的图元列表。此外，客户端能够指定要获取的图元属性，也能够从空间和非空间上约束查询。该服务可以根据 DescribeFeatureType 请求中返回的描述返回图元的搜索结果 (Content)。通过筛选器还可以选择一组图元用于分析，例如在某一区域或新住宅开发区（自定义多边形）内所有的水图元。GetFeature 通过 HTTP GET 和 HTTP POST 支持。它可以读取 GML2 几何体并返回 GML2 或 GML3 几何体。在 HTTP GET 请求中不支持筛选器。

使用 MapXtreme 2005 WFS 客户端

用户可以通过编程方式使用 MapXtreme 2005 附带的 WFS 客户端，访问符合 OGC 标准的 Web 图元服务的数据。MapXtreme 2005 Web 图元服务 (WFS) 客户端在 API 级提供与任何符合 WFS 1.0.0 标准的服务器交互的能力。WfsClient 和相关类位于 MapInfo.Wfs.Client 命名空间并使用 OGC 规范的 WFS Basic 部分。使用 WfsClient 类，可以在任何符合 WFS 1.0.0 标准的服务器上调用以下方法：GetCapabilities、DescribeFeatureType 和 GetFeature。

GetCapabilities

WFS 客户端可以通过 HTTP GET 或 HTTP POST 调用 GetCapabilities 方法。对该请求的响应为 MapInfo.Wfs.Client.WfsCapabilities 对象或 ServiceExceptionReport。WfsCapabilities 对象直接匹配 WFS 1.0.0 XML 构架中的 WFS_Capabilities 元素。使用 Wfs_Capabilities 对象，可以了解到关于特定 WFS 1.0.0 服务器的详细信息。

GetCapabilities 方法有两个参数：RequestMethod 和 url。url 指定要返回其功能的 WFS 服务器。RequestMethod 用于通过 HTTP GET 或 POST 与 WFS 服务器通信。

VB 方法

```
Public Shared Function GetCapabilities( _  
    ByVal requestMethod As RequestMethod, _  
    ByVal url As String _  
) As WfsCapabilities
```

DescribeFeatureType

WFS 客户端可以通过 HTTP GET 或 HTTP POST 调用 DescribeFeatureType 方法。响应为 System.Xml.Schema.XmlSchema 对象或 ServiceExceptionReport。DescribeFeatureType 方法有三个参数: url、typeNames 和 namespaces。url 指定用于描述图元的 WFS 服务器。typeNames 用于定义要描述的图元列表 (如机场)。通过指定为空或无内容的列表, WFS 服务器将返回对所有图元类型的描述。namespaces 用于列出图元类型的命名空间及其前缀 (如 miwfs:World)。

VB 方法

```
Overloads Public Shared Function DescribeFeatureType( _
    ByVal url As String, _
    ByVal typeNames As IList, _
    ByVal namespaces As Hashtable _)
    ) As XmlSchema
```

GetFeature

WFS 客户端可以通过 HTTP GET 或 HTTP POST 调用 GetFeature 方法。该方法使用户可以对一个或多个图元类型执行空间和非空间 (标量) 查询。WfsClient 类的 GetFeature 方法被重载, 每个方法 (BBox、filters、featureIds 或 queries) 在其中定义不同的查询方式并返回图元。这些方法的通用属性包括: url、typeNames、propertyNames、outputFormat 和 maxFeatures。url 指定用于查询图元的 WFS 服务器。typeNames 用于定义要查询的图元列表 (如机场)。通过指定为空或无内容的列表, WFS 服务器将查询所有图元类型 (局限于请求中的空间范围或过滤)。propertyNames 用于定义要返回的图元属性的列表。例如, World 图元可能包含 Country、Continent 和 Population 属性 (列)。在该实例中, 可将属性名称指定为 miwfs:Country,miwfs:Continent,miwfs:Pop_1994。注意, 以 GetCapabilities 响应返回时每个属性都必须包括命名空间前缀。outputFormat 用于定义来自 WFS 服务器的响应格式。MapXtreme 2005 WFS 服务器既支持 GML2 输出也支持 GML3 输出。不过, 对于 outputFormat, WFS 客户端只支持 GML3。因此, 默认为 GML3。如果指定 GML2, 则客户端返回异常, 除非读取器已注册并可将 GML2 转换为 MapXtreme 2005 的图元模型。因为有多种 GML2 形式, 所以只有通过实现 IWfsReader 并用 IWfsReaderFactory 注册该实现, WfsClient 才可以与任意 WFS 服务器通信。详细信息请参阅 [第 318 页中的分析 WFS 响应](#)。maxFeatures 用于指定从查询返回的图元最大数目。值 -1 用于返回具有指定图元类型的所有图元。

GetFeature 与 BBox VB 方法

```
Overloads Public Shared Function GetFeature( _
    ByVal url As String, _
    ByVal typeNames As IList, _
    ByVal BBox As DRect, _
    ByVal propertyNames As IList, _
    ByVal outputFormat As String, _
    ByVal maxFeatures As Integer _
) As MultiFeatureCollection
```

使用 WFS 客户端 GetFeature 筛选器

可以使用 IFilter 接口将筛选器应用到 GetFeature 请求。筛选器可以是空间的或非空间的（标量）。WFS 客户端完全支持 HTTP GET 和 HTTP POST 请求中的空间查询，例如 Bbox（引用边界框）、Within 和 Intersects。标量筛选器可以进行对于图元类型特定属性的查询。WFS 客户端完全支持 HTTP GET 和 HTTP POST 请求中的标量查询，例如 PropertyIsEqualTo、PropertyIsGreater Than 和其它查询。也可以使用组合筛选器，例如 AND 和 OR。请参阅 WFS 构架和 WFS 规范 (<http://www.opengis.org/docs/02-058.pdf>)，也可以参阅 Filter Encoding 规范 (<http://www.opengis.org/docs/02-059.pdf>) 以获得详细信息。

注： HTTP 协议仅强制到 2048 个字符的 URL 长度。使用筛选器可以很容易就创建出大于 2048 个字符的 URL。这将在 GetFeature 请求中导致异常。

分析 WFS 响应

分析响应未在 WFS 规范中指定，该规范也没有描述 GetFeature 请求结果的准确格式。WFS 规范只规定请求至少必须符合 GML2 标准。由于有很多 GML2 的版本，为了使 WFS 客户端可以正确转换 WFS 服务器的 GML2 响应，必须创建分析器以将 GML2 GetFeature 请求转换为 MapXtreme 2005 MapInfo.Data.MultiFeatureCollection。这可以通过实现 IWfsReader 接口并使用 MapInfo.Wfs.Client.WfsReaderFactory 类注册该实现的方法来完成。注册必须出现在每个 URL 的基础上，也就是说，如果用户需要和两个具有相同 GetFeature 响应的 WFS 服务器交互，则对于每个服务器 URL 特定的 IWfsReader 实现必须注册两次。无需实现整个接口。可以扩展 MapInfo.Wfs.Client.WfsReader 类以重载 ReadGetFeature(XmlReader) 方法。与不兼容的 WFS 服务器交互时，任何其它方法都可以被重载。

使用 MapInfo Professional 访问 WFS 服务

MapInfo Professional 可以用作 WFS 客户端，查看从 WFS 服务器转换的 TAB 文件的 GML。以下步骤显示了如何将 MapInfo Professional 用作 WFS 客户端并访问 WFS 服务器：

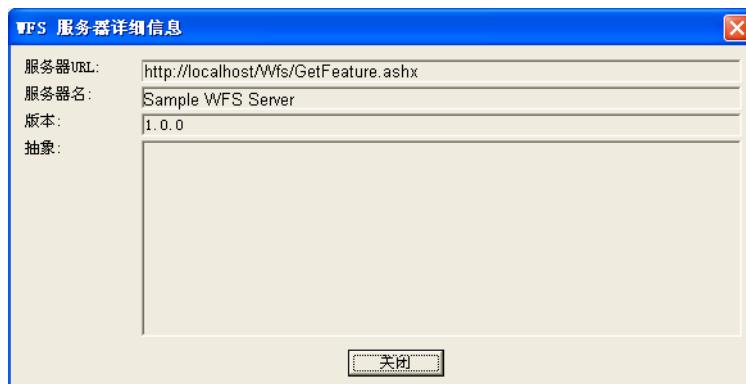
1. 使用文件 > 打开 WEB 服务 > 打开 WFS... 菜单命令。

此时显示 **WFS 服务器属性** 对话框。

2. 单击**服务器 ...**显示**WFS 服务器列表**对话框。
3. 单击**下一步 ...**。
此时显示**WFS 服务器信息**对话框。
4. 键入 WFS 服务器的 URL 并单击**测试 URL**。请参阅下图的 URL 示例。

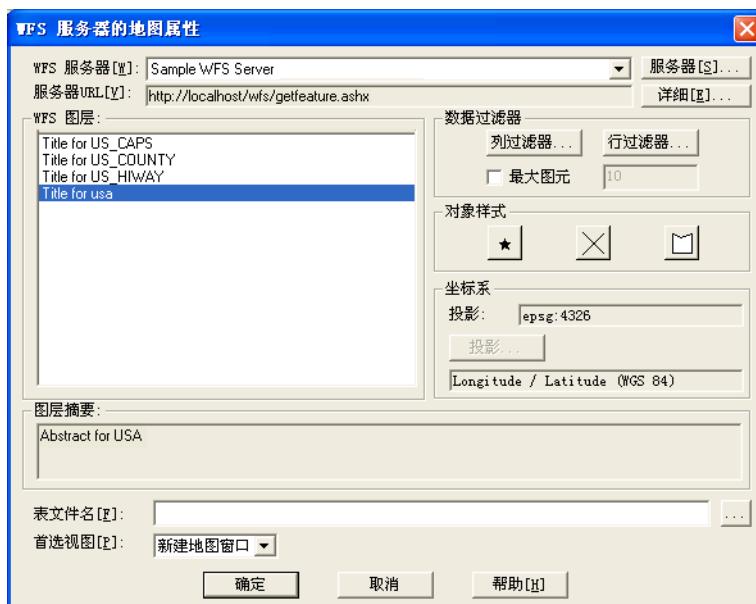


如果连接成功，则显示包含该服务器信息的**WFS 服务器详细信息**对话框。



5. 单击**关闭**返回 WFS 服务器信息对话框，然后单击**显示说明**，用服务器信息填充**描述**字段。单击**确定**关闭对话框返回**WFS 服务器的地图属性**对话框。

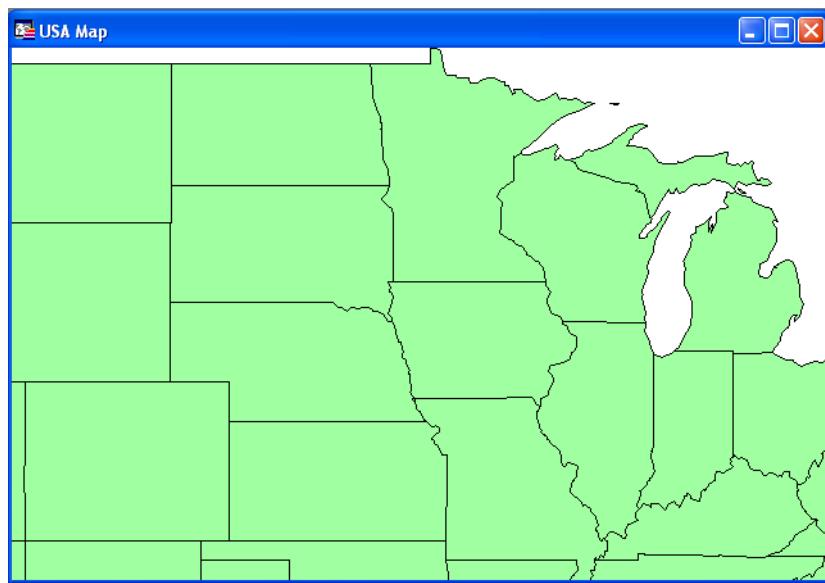
注：如果想要创建自定义描述，请在**定义**字段中键入自定义文本，然后单击**确定**，而无需单击**显示说明**按钮。



此时，**WFS 服务器的地图属性**对话框应列出服务器配置中所有的表（如果已根据教程设置了 WFS 服务器，则列出的信息位于 WFSConfig.xml 文件）。

6. 突出显示选择的图层并单击**确定**。

MapXtreme WFS 服务器将该图层同时作为空间几何 GML 2 和 GML 3 属性数据。MapInfo Professional 从服务器接收该数据并将其转换为标准的可绘制地图的 TAB。



注: 使用 WFS (Web 图元服务器) 部署数据时, 请务必遵守相关的数据许可证。就是 WFS 的性质简化了对空间数据的“命名用户”样式访问。这意味着任何反映该级别数据访问的数据许可证均须在可以部署数据之前建立。在 MapInfo 数据产品中, 这意味着任何反映该级别数据访问的数据许可证都必须在可以部署数据之前建立。

WMS (Web 地图服务器) 允许用户使用不同的 MapInfo 数据许可证。使用 WMS 部署的 MapInfo 数据可接受“MapInfo 服务器”许可证, 而不是“命名用户”许可证。

Workspace 管理器

Exponare 2.3 包含的 Workspace 管理器实用程序让用户可以创建和管理新的基于 XML 的工作空间文件 (.MWS 格式)，这类似于 MapX 产品中允许用户创建和管理 geoset 文件的 Geoset Manager。Workspace 管理器简便的用户界面使得用户更易于组装组成地图的表，应用大量附加的设置并作为工作空间保存地图。完成的工作空间还可用于自己的应用程序，或用于打印或文件输出。

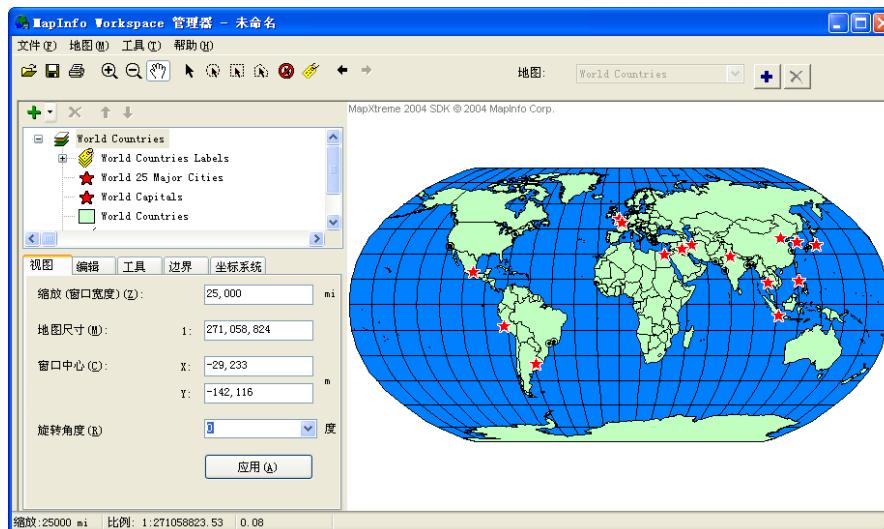
GeoDictionary 管理器

本章内容 ...

- ◆ *Workspace* 管理器的特性 324
- ◆ 工作空间格式和内容 325
- ◆ *Workspace* 管理器菜单命令 325
- ◆ 图层控制 332

Workspace 管理器的特性

Workspace 管理器允许用户控制大部分（但不是全部）可以存储在工作空间文件中的设置。例如，工作空间可以包含有关制图图例和修饰的信息；不过，Workspace 管理器未提供任何创建制图图例或修饰的选项。要创建制图图例或修饰，请使用 API。



通过 Workspace 管理器，用户可以：

- 加载 XML 工作空间、表、geosets 和 MapInfo 工作空间 (.mws, 不是 .wor 文件)。
- 将工作空间保存为 .MWS。
- 控制作为工作空间的一部分打开的表。
- 使用命名连接管理器创建和加载命名的连接。
- 添加 / 删除地图和查看一个或多个地图。
- 使用内建的图层控件设置地图和图层可视性、图层和标记样式以及主题的属性。
- 添加、删除或改变定制标记。
- 创建允许用户将图层组织到逻辑组的组图层，以便用户通过单击就可以显示和隐藏整个组。
- 查看具有多个视图的下一个和前一个地图视图。
- 使用导航和手动标记放置的地图工具，并使用选择集工具来验证图层可选择设置是否正确。
- 预览和打印地图。
- 从最近文件列表快速打开最近使用过的工作空间。

工作空间格式和内容

工作空间文件是 XML 文档（扩展名是 .MWS），包含了对所有地图、表、图层和设置（这些元素构成工作空间）的说明和元数据的位置。因为工作空间文件是 XML 文档，所以工作空间是可移植的，这意味着用户可以与不同计算机、不同网络上的其它用户跨平台共享工作空间。

在 MapXtreme 2005 中，XML 工作空间的可移植性通过使用命名连接实现。命名连接让用户可以根据自己的环境定义其它驱动器、路径和数据库连接信息，以便可以使用其它用户创建的工作空间。工作空间在打开时解析所有的路径和连接字符串。可以直接在 Workspace 管理器（**文件 > 管理命名连接**）中设置命名连接。有关 XML 工作空间结构的详细信息请参阅附录 G：理解 *MapInfo* 工作空间。

Workspace 管理器菜单命令

本节解释了 Workspace 管理器中可用的每个菜单命令。

“文件”菜单命令

“文件”菜单中的命令提供了所有标准的“文件”菜单功能（例如打开、保存和打印文件），还提供了一些 Workspace 管理器特有的功能。每个命令的信息如下所述。

新建

创建新的空地图，用户可以使用图层控件的“添加”工具或“文件”菜单的“打开表”命令将表添加到该地图。如果对当前工作空间做出更改，则创建新的工作空间之前将提问是否要保存更改。

打开工作空间

打开现有的工作空间 如果对当前工作空间做出更改，则打开现有的工作空间之前将提问是否要保存更改。

打开表

“打开表”命令允许用户打开一个或多个表并将这些表添加到地图。

保存

将地图保存为工作空间。

另存为

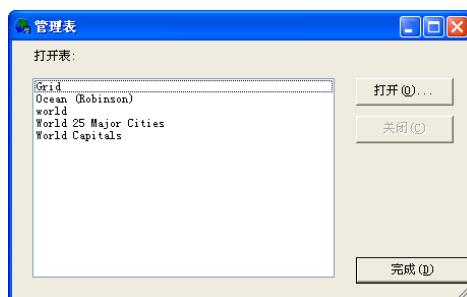
用新的文件名保存工作空间的副本。

关闭所有表

“关闭所有表”命令关闭所有打开的表。

管理表

“管理表”命令显示列出组成地图的表的对话框，还可以使用户打开地图中可能包括的其它表。单击**打开**以显示“打开”对话框并打开表。打开的表添加到“管理表”对话框中打开表的列表。然后可以使用“添加”工具将表添加到地图，“添加”工具位于 Workspace 管理器图层控件窗口上。



要关闭表，首先单击列表中的表以突出显示要关闭的表，然后**关闭**按钮激活，最后单击**关闭**以关闭表。引用关闭的表的图层从地图删除。

管理命名连接

命名连接使用别名说明了到数据源的连接。可以创建以下类型的命名连接：FilePath、DatabaseSource、ODBC 或 Oracle OCI。指定连接字符串或文件路径后，可以将命名连接保存为 XML 用于以后检索。可以设置默认的连接，以便运行 Workspace 管理器时到数据源的连接可用。命名连接也保存到工作空间。



页面设置

“页面设置”允许用户指定打印地图的纸张大小、方向和页边距。也可以使用该选项访问特定于打印机的设置。

打印

“打印”命令允许用户将地图打印到纸张或作为文件输出。在“打印”对话框中，可以指定要使用的打印机，如果作业是多个页面则可以指定打印的页面范围，此外还可以指定打印的份数。打印机属性允许用户设置布局和特定于正使用的打印机的其它选项。**打印到文件**复选框允许用户打印输出到文件。

打印预览

使用“打印预览”命令可以在打印之前查看输出的外观如何。

最近使用的工作空间

“最近使用的工作空间”显示了最近打开的工作空间的列表。

“地图”菜单命令

使用“地图”菜单中的命令可以添加和删除地图，操控地图的视图，还可以创建主题地图。也可以在弹出菜单中使用视图命令。在 Workspace 管理器地图窗口中右键单击显示菜单。

添加新地图

“添加新地图”命令允许用户使用目前打开的表创建新的地图窗口。

选择地图

“选择地图”命令允许用户选择在 Workspace 管理器中要查看哪个地图。

删除当前地图

从 Workspace 管理器视图移除当前地图并从工作空间删除当前地图。

改变视图

“改变视图”命令允许用户更改地图的当前视图（也就是说，窗口中目前显示的地图区域）。可以将地图的缩放和比例更改为自己的设置。也可以设置地图窗口的中心或更改旋转角度。

“改变视图”对话框允许用户选择缩放宽度的单位（英里、千米等）和中心 X/Y 坐标的单位（米、度等）。“改变视图”对话框中选择的任何单位也都用于图层控件。例如，如果要将图层控件中的所有距离以千米显示而不以英里显示，则显示“改变视图”对话框并从显示在“缩放”字段旁边的单位列表选择千米。



查看整个图层

使用“查看整个图层”命令查看整个图层或地图中的所有图层。“查看整个图层”对话框显示构成地图的图层的列表。从列表选择想要的图层或选择**所有图层**，以便所有的图层完全显示在视图中，然后单击**确定**。地图重绘以显示整个地图。

前一视图

使用“前一视图”命令返回地图的前一个视图。

下一视图

使用“前一视图”后可以使用“下一视图”命令。使用“前一视图”命令之前使用“下一视图”命令重新显示屏幕上地图的视图。

“前一视图”和“下一视图”命令可以一起使用来在地图的两个视图之间向前和向后切换。这些命令可以像工具栏的工具一样使用。

保持比例 / 缩放

使用“保持比例”和“保持缩放”命令使得缩放和（或）比例与用户更改地图的大小和形状相同。

重绘

使用“重绘”命令以重绘地图。

视图选择集

使用“视图选择集”命令放大或缩小一个或多个选中的对象。

创建主题

通过**地图 > 创建主题**菜单可以创建图元主题和标记主题。图元主题包括范围、单值、点密度、分级符号、饼图和条形图。标记主题包括有范围和单值。

如果地图包括至少一组标记（显示在标记图层中），则用户可以创建标记主题。标记主题根据表中的数据将不同的标记样式（不同颜色等）分配到每个标记。例如，使用标记主题显示其它标记位置上的突出部分。有范围标记主题组根据类似的数据值（例如人口）标记。处于某些人口范围内的城市使用一种样式标记，而处于其它范围的城市用另一种样式标记，通常不太突出的样式指示了不必使用人口值标记的城市大小。

“创建主题”命令打开“创建主题”向导，该向导可以让用户更易于一步一步地创建主题。

要创建主题：

1. 选择**地图 > 创建主题**。

创建主题：步骤 1/3 对话框显示。

2. 选择图元主题或标记主题，并选择要创建的主题的类型。

3. 单击**下一步**。

创建主题：步骤 2/3 对话框显示。

4. 选择要加底纹的表。

5. 选择要使用的数据。从包含数据的表选择列或选择**表达式**以使用表达式从表派生使用的数据。

6. 单击**下一步**。

创建主题：步骤 3/3 对话框显示。在这一步中，可以定制主题类型设置、样式和图例。

7. 单击**应用**以应用定制的设置。
8. 完成后单击**确定**。

“工具”菜单命令

Workspace 管理器的“工具”菜单提供了通过菜单命令对地图工具的访问。这些工具允许用户以各种方式放大和缩小地图，更改地图的位置并选择地图对象。在 Workspace 管理器工具栏上也可以使用这些相同的工具。下文解释了每种工具。

放大

使用“放大”工具放大地图的区域视图。要放大地图：

1. 选择**工具 > 放大**来激活工具。
光标更改为带有加号的放大镜。
2. 单击地图。
地图在放大的区域视图上重绘，并以单击的点为视图的中心。

缩小

使用“缩小”工具缩小地图的区域视图。要缩小地图：

1. 选择**工具 > 缩小**来激活工具。
光标更改为带有减号的放大镜。
2. 单击地图。
地图在缩小的区域视图上重绘，并以单击的点为视图的中心。

平移

使用“平移”工具重定位地图，而不更改缩放级别。例如，可以重定向地图的视图使得某些国家或城市位于中心。要平移地图：

1. 选择**工具 > 平移**来激活工具。
光标更改为手形图标。
2. 单击地图并按下鼠标按钮将地图拖到想要的位置。
地图重绘以反映新的位置。

选择

使用“选择”工具一次选择一个对象或选择通常位于相同区域的所有对象。

要使用“选择”工具选择对象：

1. 选择**工具 > 选择**来激活工具。

光标更改为箭头。

2. 在地图上单击要选择的对象。

选中的对象突出显示。

半径选择

使用“半径选择”工具选择所有位于给定半径内的对象。例如，如果有捐血人的表和捐血点的表，则使用“半径选择”工具就可以创建捐血人的临时列表，这些捐血人居住在距离每个捐血点方圆 1.5 英里内。

注意，“半径选择”工具选择所有质心位于圆内的对象。对象不必完全以圆为边界。要选择半径内的对象：

1. 选择**工具 > 半径选择**来激活工具。

光标移动到地图上时成为手形。

2. 单击地图上的某个位置，用作半径搜索的中心点。例如，如果要搜索消防站方圆 2 英里内的所有消防栓，则单击消防站并将其用作中心点。

3. 按下鼠标按钮并将鼠标拖离中心点。

Workspace 管理器围绕该点绘制圆并在 StatusBar（屏幕的左下角）报告圆的半径。

4. 达到想要的半径后释放鼠标按钮。

Workspace 管理器突出显示所有位于圆内的地图对象。

矩形选择

使用“矩形选择”工具选择矩形内的对象。通过单击和拖放使用“矩形选择”工具，创建虚线矩形，或围绕要选择的对象创建选取框。

注意，“矩形选择”工具选择所有质心位于矩形内的对象。对象不必完全以矩形为边界。要选择矩形内的对象：

1. 选择**工具 > 矩形选择**来激活工具。光标移动到地图上时成为手形。

2. 单击地图上处于选取框区域外部的某个位置。

3. 按下鼠标按钮并拖动鼠标围绕要选择的点形成虚线矩形。

4. 达到想要的矩形大小后释放鼠标按钮。

Workspace 管理器突出显示所有位于矩形内的地图对象。

多边形选择

“多边形选择”工具选择位于地图上绘制的多边形内的地图对象。

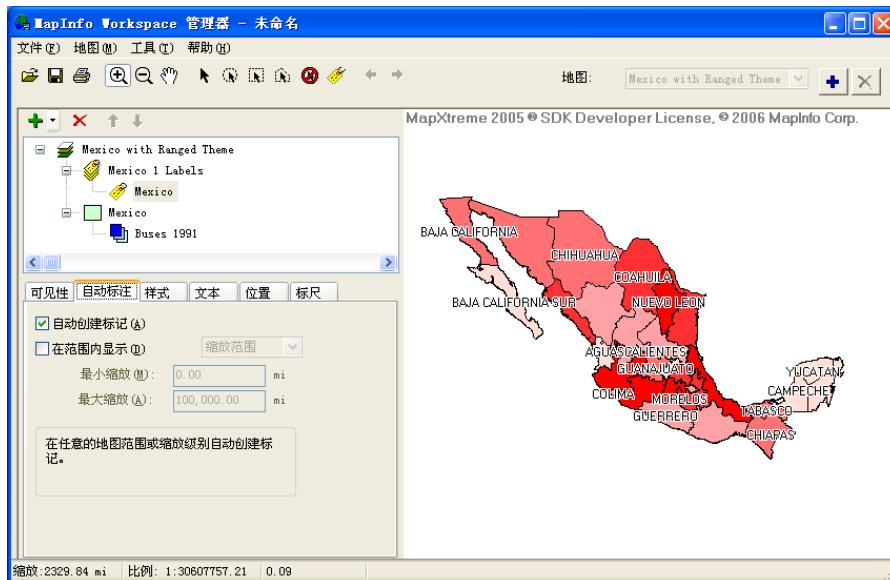
要使用“多边形选择”工具选择对象：

1. 选择工具 > 多边形选择来激活工具。
光标移动到地图上时成为手指。
2. 单击要放置多边形的第一个终点的地图位置。沿任意方向在地图上移动光标。
Workspace 管理器从用户单击光标的点处绘制直线。
3. 单击创建另一个终点。继续移动光标，然后在达到想要的多边形边数时单击。
4. 要封闭多边形，请在尽可能接近第一次单击的地方做最后一次单击。
Workspace 管理器封闭多边形并选择多边形内的对象。

图层控制

Workspace 管理器应用程序窗口分为两个主要部分。图层控件窗口和命令位于左侧，而地图窗口位于右侧。图层控件窗口显示打开的地图及其所有附属的图层。

Workspace 管理器的图层控件特性使得用户可以组装地图的图层，并将设置应用到管理图层或地图如何显示的独立图层或整个地图。



图层控件工具

图层控件窗口顶部的工具使得用户易于从图层控件窗口中添加、移动和删除图层。

- “添加”工具允许用户打开表，并将组图层和标记图层插入地图中。
- “删除所选项目”工具从地图删除选中的图层。
- “上移”和“下移”工具栏按钮允许用户在图层列表中上移和下移图层，更改图层在图层列表中显示的顺序。

图层树

图层控件显示了说明地图和地图中所有图层的树，还有诸如主题的其它地图元素。图层树允许用户执行以下操作：

更改图层顺序

要更改图层顺序，可以选择图层并单击“上移”或“下移”工具栏按钮。此外，可以向上或向下拖动图层以更改图层在列表中的位置。

有两种与拖放操作有关的特殊情况：

- 要将图层移动到组图层中，请将图层拖动到组上。
- 要将新的标记源添加到标记图层，请将图层拖动到标记图层上。

如果不想将图层移动到组中（相反，如果想重定位图层使得该图层位于组图层上），则完成拖放操作之前按下 SHIFT 键。类似的，如果不想将新的标记源添加到标记图层，则按下 SHIFT 键。

显示上下文菜单

图层树中的每个项目具有一个上下文菜单。要显示图层的上下文菜单，请右键单击图层或按下 SHIFT-F10 显示目前所选图层的菜单。

上下文菜单的项目取决于指定图层的类型。特别要注意以下部分：

要将样式覆盖添加到图层，请显示上下文菜单并选择“添加样式覆盖”。注意，每个图层具有多个样式覆盖，每个图层还具有不同的缩放范围；这允许用户设置地图以便在缩放时点显示得更大，道路显示得更宽。

要重命名图层树中的任意项目，请右键单击项目并选择“重命名”。此外，可以按下 F2 重命名选中的项目。注意，以该方式重命名图层不重命名初始表；重命名操作只更改显示在图层树中的文本。

图层控件选项卡

图层控件选项卡提供了可以应用到地图和地图中每个图层的其它设置和控件。选项卡根据图层或地图是否在图层控件窗口中突出显示来更改。对话框选项卡集用于控制地图、标记图层、图层和主题。每个选项卡的选项如下所述。

地图设置

选择图层控件窗口中的地图时，以下选项卡可用：视图、工具、编辑、边界和坐标系。

视图

“视图”选项卡允许用户控制总的地图外观。可以设置缩放级别、比例、中心点（以度为单位）和旋转角度。单击**应用**按钮以应用设置。

工具

“工具”选项卡允许用户控制 InfoTips 的显示，激活“对齐节点”并设置对齐公差，还激活“动态选择集”工具。

“动态选择集工具”复选框控制图元是立即选中（使用选择集工具时）还是在使用选择集工具释放鼠标按钮完成时选中：

如果“动态选择集工具”复选框未选中，则使用工具完成之前选择集工具实际上不选择地图上的任何图元。例如，指定半径并释放鼠标按钮之前“半径选择”工具不选择任何图元。

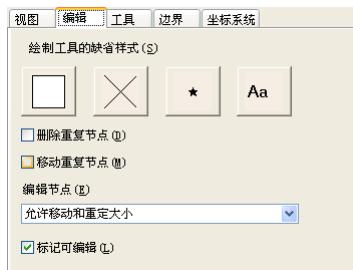
如果“动态选择集工具”复选框被选中，则拖动鼠标时图元动态选择或删除。例如，如果使用“半径选择”工具，则拖动鼠标扩大半径时可以选中更多图元。

如果“对齐节点”复选框被选中，则诸如“选择”工具等地图工具自动搜索附近的节点。如果节点在附近，则显示交叉线指示最近节点的位置。例如，使用“半径搜索”工具时可能要选择“对齐节点”复选框，且要确保以地图上点图元的精确位置为中心进行搜索。在提供绘制工具的应用程序中“对齐节点”图元尤其重要，因为用户通常需要在现有图元的精确位置绘制图元。

“对齐公差”设置指定了工具将搜索“可对齐”节点的范围。可以选择哪些图层使用“对齐节点”图元。例如，可能要打开“对齐节点”，但仅在特定图层中光标位于图元附近时才显示对齐交叉线。要打开或关闭特定图层的“对齐节点”图元，请选择图层树中的图层，然后在“选项”选项卡中选择或清除**对齐节点**复选框。

编辑

应用程序使用绘制工具、调整对象大小、移动和删除对象节点时，“编辑”选项卡中的选项允许用户控制某些地图编辑任务（例如用于绘制对象的样式）。



样式框允许用户指定应用程序使用的任何绘制工具的默认样式。单击样式框打开对应的样式对话框。选择的设置保存在工作空间中。用户在使用绘制工具的应用程序中打开工作空间且在地图上绘制对象时，应用程序使用这些样式设置。

也可以指定是否要删除或移动对象的重复节点。

最后，可以为地图指定编辑模式：

- 无 - 不可以在地图上执行任何编辑。
- 允许移动和重定大小 - 对象可以移动和调整。
- 允许编辑节点 - 节点可以移动或删除。
- 允许添加节点 - 可以将节点添加到对象。

选择的编辑模式应用到所有地图中可编辑的图层。

边界

在地图中，“边界”选项卡显示了当前地图视图的边界。单击[查看全部地图](#)按钮查看所有的地图。

坐标系统

“坐标系”选项卡指示地图的坐标系并允许用户更改坐标系。

请按以下步骤执行：

1. 单击**坐标系**按钮以显示“选择坐标系”对话框。
2. 从列表选择坐标系，然后单击**确定**。

图层设置

突出显示构成地图的其中一个图层时，显示四个选项卡：可视性、选项、边界和信息。

可视性

选择**在范围内显示**复选框以指定图层显示的缩放范围或比例范围。如果选择缩放范围，则以英里为单位指定最小和最大范围。图层显示在该范围内。如果选择比例范围，则指定最近和最远的比例。图层显示在该比例范围内。

选项

“选项”选项卡复选框简化了对地图图层的编辑和定制：

- **可选** - “可选”复选框选中时，可以使用“工具”菜单命令或工具栏中的“选择”工具来选中图层中的对象。清除不想从中选择的任意图层的“可选”复选框。
- **可编辑** - 选择复选框生成可编辑的图层。
- **DRAWING TOOLS 可将图元添加到该图层** - 如果准备将该工作空间用于提供绘制工具的应用程序中且要绘制工具在该图层中创建新的图元，则选择该复选框。
- **显示 INFO TIPS** - 选择“显示 InfoTips”复选框以显示选中图层中地图对象上的 InfoTips。InfoTip 文本由“InfoTip 表达式”字段中表达式的结果组成。例如，如果表达式是表中的列，则 InfoTips 组成该列的值。如果表达式使用表中的列信息计算，则 InfoTips 组成该计算的结果。
- **对齐节点** - 选择打开或关闭特定图层的“对齐节点”图元，选择图层树中的图层，然后选择或清除“对齐节点”复选框。

边界

对于图层，“边界”选项卡显示了选中图层的边界。单击**查看整个图层**按钮查看所有的图层，或单击**查看默认区域**来查看图层的默认视图。

信息

“信息”选项卡提供了与选中图层有关的信息。还给定了表（例如 MapInfo 表）的名称、路径和类型及其坐标系。

主题设置

可视性

主题图层选中时，“可视性”选项卡选项控制选中主题的显示。选择“可视”复选框显示主题图层；清除复选框关闭主题的显示。

选择**在范围内显示**复选框以指定主题显示的缩放范围或比例范围。如果选择缩放范围，则指定最小和最大缩放距离。主题显示在该范围内。如果选择比例范围，则指定最近和最远的比例。主题显示在该比例范围内。

Theme

“主题”选项卡指示主题地图的类型和用于获得值的表达式。“主题”选项卡还允许用户修改主题地图。单击**修改主题**以更改样式或图例。

标记设置

选择图层控件窗口中的标记图层时，“可视性”选项卡可用。扩展标记图层以查看标记源时，其它选项卡显示如何控制标记源中标记的外观和内容：Autolabel、Style、Expression、Position 和 Rules。

可视性

标记图层选中时，“可视性”选项卡中的选项控制标记的显示。选择**可视**复选框以显示标记；清除复选框以关闭标记显示。

选择**在范围内显示**复选框以指定标记显示的缩放范围或比例范围。如果选择缩放范围，则指定最小和最大缩放距离。标记显示在该范围内。如果选择比例范围，则指定最近和最远的比例。标记显示在该比例范围内。

自动标注

自动标注选项卡允许用户创建和管理自动标记的显示。选择**自动创建标记**复选框以生成地图的自动标注。选择**在范围内显示**复选框以指定自动标记显示的缩放范围或比例范围。如果选择缩放范围，则指定最小和最大缩放距离。如果选择比例范围，则指定最近和最远的比例。自动标注显示在该比例范围内。



样式

“样式”选项卡控制标记文本和标记线的样式。对于标记文本，使用“文本”样式框访问“文本样式”对话框。可以指定标记的字体、颜色、背景和其它文本效果。对于标记线，在设置标记线的样式的位置使用“直线”样式框访问“直线样式”对话框。在“标记线”组中，选择是否不要标记线、简单线或带有箭头的直线。

文本

“文本”选项卡允许用户指定表达式，该表达式从列或表中派生的信息生成标记文本。

位置

使用“位置”选项卡中的设置来设置标记的方向、偏移和旋转。

标记的方向是相对于其标记点的标记位置。单击其中一个按钮以选择方向。

标记偏移是标记离开其标记点的距离（以像素为单位）。

标记旋转是绘制标记的角度。有三种“旋转”设置：

- **与一个区段平行** - 如果正标记诸如高速公路的直线图元且要以某个角度（使得标记与高速公路的最近区段平行）绘制每个标记，则选择该选项。
- **与多个区段平行** - 如果正标记诸如高速公路的直线图元且要标记文本符合高速公路的形状，则选择该选项。
- **特定角度** - 以度为单位指定角度（例如零度）使得所有标记水平。

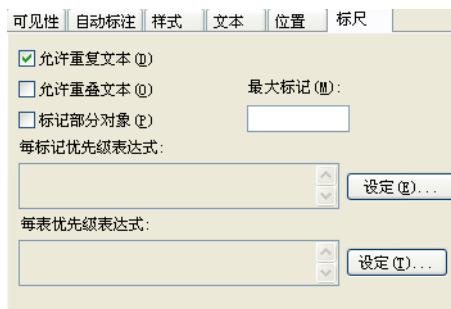
标尺

“规则”选项卡允许用户在地图上设置某些条件用于显示标记。

允许重复文本 - 选择“允许重复文本”复选框对于不同的对象允许重复的标记，例如 Portland 或 Portland, ME。

允许重叠文本 - 选择“允许重叠文本”复选框允许标记以互相重叠的方式绘制。某些标记不显示，因为这些标记与在地图上给定更高优先权的标记重叠。

标记部分对象 - 选择“标记部分对象”复选框标记折线和质心未显示在“地图”窗口中的对象。



每标记优先级表达式

该表达式字段可选。如果将该表达式字段保留为空，则单一标记源内的图元以不可预知的方式标记。例如，用户会发现某些小城市已标记，但一些大城市未标记（因没有足够的空间）。如果指定了表达式（必须是数值），则表达式将在地图的可视部分上为每个图元计算，且具有较大值的图元将被给定较高的标记优先权。要指定表达式，请单击**设定**按钮。

例如，假定用户正配置 WorldCapitals 图层的标记，该图层包含了表示城市的点图元。可能需要具有最多人口的城市具有最高的标记优先权。这种情况下，将指定标记表达式，例如：

```
cap_pop
```

cap_pop 列表示了每个首府的人口。指定 cap_pop 的“每标记优先级表达式”时，用户指定具有最多人口的城市应具有最高的标记优先权。因此，人口最多的城市首先标记，而只有留下足够的空间时才标记其它城市。

每表优先级表达式

该表达式字段可选。标记图层可以包含多个标记源；例如，可以具有一个表示一组标记世界国家的标记源，还可以具有另一个表示一组标记世界首都（城市）的标记源。默认情况下，列表顶部的标记源具有最高的优先权。如果要为标记源分配更高的优先权，则可以上移标记源（在标记源的图层控件列表中），或可以为每个标记源指定“每表优先级表达式”。

例如，如果为“世界国家”标记源给定的每表优先级表达式是 10，而为“世界首都”标记源给定的每表优先级表达式是 5，则国家标记比首都城市标记的优先权要高。

20

使用 GeoDictionary 管理器

GeoDictionary 管理器是 MapXtreme 2005 中的实用程序，用于地图绘制应用程序。本附录说明如何使用该工具。

本附录内容：

- ◆ 使用 *GeoDictionary* 管理器 342
- ◆ *GeoDictionary* 管理器中的变化 342
- ◆ *GeoDictionary* 管理器的用户界面 342
- ◆ *GeoDictionary* 文件 346

使用 GeoDictionary 管理器

GeoDictionary 管理器设计用于支持 GeoDictionary 文件的操控。GeoDictionary 是包含 MapInfo 表注册信息的 XML 文件，可以在数据自动绑定过程中和用户的应用程序匹配。只有可以或将要匹配的 MapInfo 表应该在 GeoDictionary 中注册。

不需要注册应用程序使用的 GeoDictionary 中的每个 .tab 文件，实际上，注册不必要的文件会增加系统开销。只需要注册那些想要匹配的表。

GeoDictionary 管理器中的变化

MapXtreme 2005 的 GeoDictionary 管理器在外观上类似于包含在 MapX 和 MapXtreme 先前版本中的实用程序。主要区别是现在基础表文件 (*.dct) 是 XML 文件且完全可编辑（使用文本或 XML 编辑器）。

GeoDictionary 管理器的用户界面

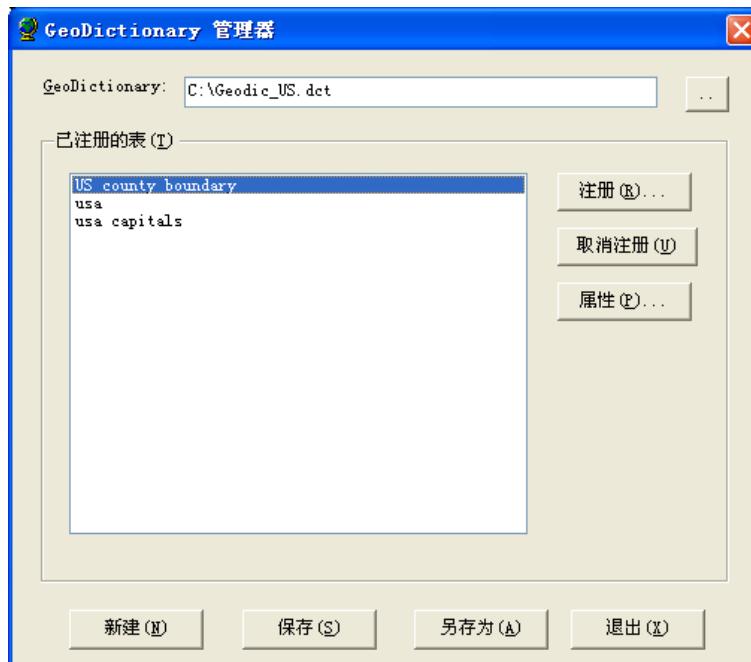
本节描述 GeoDictionary 管理器的用户界面。

运行 GeoDictionary 管理器

手动注册图层时运行 GeoDictionary 管理器：

- 开始 -> 程序 -> MapInfo -> MapXtreme 2005 -> GeoDictionaryManager。

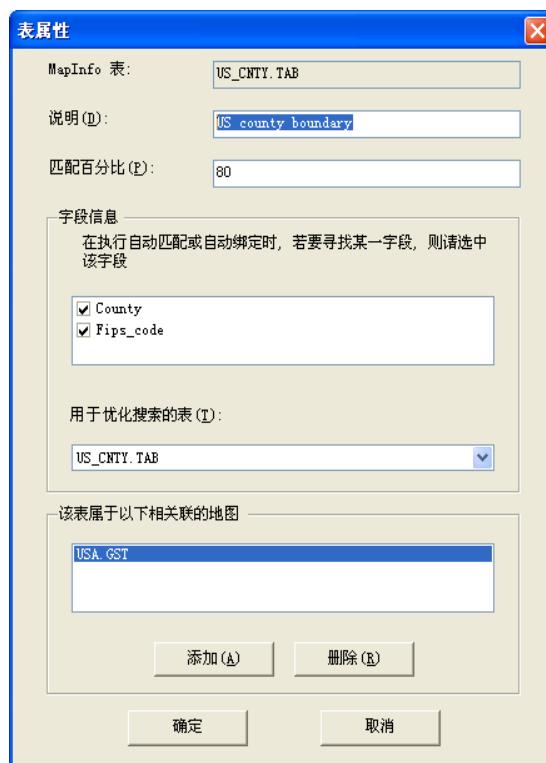
显示 GeoDictionary 管理器。启动时重新加载最后打开的文件。



部分	说明
GeoDictionary	GeoDictionary 编辑框包含当前管理的 GeoDictionary 的完整路径。GeoDictionary 编辑框右侧的按钮允许用户浏览并管理另一个 GeoDictionary。二进制和 XML GeoDictionary 文件都可以打开，但只能保存 XML 格式。
已注册的表	“已注册的表”列表框包含了所有注册到 GeoDictionary 的表的好记的名称列表。突出显示特定的表以取消注册或修改它的属性。双击表弹出“属性”对话框以编辑该表的属性。
注册	单击“注册”按钮弹出普通“打开文件”对话框，对话框上的“文件类型”组合框设置为“MapInfo Tables (*.tab)”。从文件选取器中选定表后，显示该表的“表属性”对话框。如果在“打开文件”对话框中选定多个文件，则对于每个添加的表都打开一个“属性”对话框。
取消注册	单击该按钮，从 GeoDictionary 中删除选中的表。也可以选中多个表（通过按住 shift 键单击）取消注册。使用“取消注册”按钮不会将文件从硬盘删除。
属性	单击“属性”按钮，弹出选中表的“表属性”对话框。参阅以下的“表属性”对话框。

部分	说明
新建	“新建”按钮用于清除屏幕上的字段，这样可以创建新的 GeoDictionary 文件。选择关闭该 GeoDictionary 文件时，将提示是否保存更改。
保存	“保存”按钮用于将当前文件另存为 XML 格式，无论该文件被打开时的格式是什么。
另存为	“另存为”按钮用于将当前文件另存为新名称或新位置。
退出	“退出”按钮用于关闭应用程序。如果 GeoDictionary 文件已经修改，将提示是否保存更改。

使用“属性”对话框为给定表设置匹配表属性。要访问“属性”对话框，单击“属性”按钮，或双击“注册的表”列表中的任意表名。



部分	说明
MapInfo 表	如果 MapInfo 表文件位于 GeoDictionary 的相同目录下，则只读编辑框包含 MapInfo 表的文件名，否则包含该文件的相对路径名。如果该文件位于不同的驱动器或卷，则显示完整路径。
说明	该字段用于更改表的好记的名称。该控件默认为 .TAB 文件中的 Description 标记，如果无法找到 Description 标记则为文件名，但用户可以更改。注意，在 GeoDictionary 管理器中对描述的更改只存储在 GeoDictionary 中并且不反映在表自身。这使得 GeoDictionary 管理器可以很容易地使用只读数据，例如 CD-ROM 上的数据。
匹配百分比	该字段最初填充 GeoDictionary 的默认值，并可以通过更改该值而改变以适应特定的表。值必须介于 1 和 100 之间。
字段信息	该列表框包含表中所有索引列的列表。如果选中某给定列的框，则将在匹配进程中搜索该字段。
用于优化搜索的表	该字段使用户可以设置优化表，决定特定索引中不惟一数据的精确匹配。 某些表，例如 US Counties，包含不惟一的索引列。在这种情况下，需要优化表以决定数据的精确匹配。如果表包含不惟一的索引列，使用该字段指定用于匹配的表，以找到惟一的条目。
该表属于以下相关联的地图	该列表框显示了包含该特定表的 Workspaces、GeoSets 或其它文件。
添加	显示了普通文件选取器，使用户可以选择 GeoSets、Workspaces 或与特定 Tab 文件关联的其它文件。选中一个或多个文件，将它们添加到关联地图列表。单击时，可以通过按住 shift 或 control 键来选择多个文件。
删除	删除选定的 geoset、workspace 或关联地图列表中的其它文件名。

GeoDictionary 文件

如果用户选择跳过使用 GeoDictionary 管理器，则可以手动操控 GeoDictionary 文件 (*.dct)。如果用户理解并可以熟练编写 XML，则该方法是使用 GeoDictionary 管理器之外的可行选择。.dct 文件是直接的 XML。

示例 .dct 文件

以下是示例 GeoDictionary 文件，GeoDic_US.dct。该文件是非常简单的 GeoDictionary，仅作为结构的示例提供。

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes"?>
<!--GeoDictionary file-->
<GeoDictionary>
    <DefaultMatchThreshold>80</DefaultMatchThreshold>
    <MatchTables>
        <MatchTable>
            <TableName>US_CNTY.TAB</TableName>
            <TableDescription>US county boundary</TableDescription>
            <RefineTableName>USA.TAB</RefineTableName>
            <MatchThreshold>90</MatchThreshold>
            <AssociatedMaps>
                <AssociatedMap>USA.GST</AssociatedMap>
            </AssociatedMaps>
            <MatchFields>
                <MatchField>
                    <FieldName>County</FieldName>
                </MatchField>
                <MatchField>
                    <FieldName>Fips_code</FieldName>
                </MatchField>
            </MatchFields>
        </MatchTable>
    </MatchTables>
</GeoDictionary>
```

结构中的元素对应 GeoDictionary 管理器中的独立字段和控件，如以下表格中定义：

XML 元素	表属性对话框的字段	说明
GeoDictionary	无	文件的根元素
DefaultMatchThreshold	无	特定的 MatchThreshold 不是为特定的表定义时，DefaultMatchThreshold 是匹配的阈值。 注： 该值不可以使用 GeoDictionary 管理器设置。
MatchTables	无	注册的 MatchTables 的容器元素。对于“已注册表”列表中的每一个项目，应该都有单一 MatchTable 元素与之对应。
MatchTable	MapInfo 表。	每个要匹配的表的文件名。
TablePath	MapInfo 表。	每个 MatchTable 的相对路径。
TableDescription	说明字段（也是出现在“已注册表”列表中的名称，该列表位于主对话框内）。	MatchTable 的好记的名称。
RefineTableName	用于优化搜索的表	用于优化匹配的相关表。
MatchThreshold	匹配百分比	该 MatchTable 的匹配阈值。如果没有指定值，则使用 DefaultMatchThreshold。
AssociatedMaps, AssociatedMap	该表属于以下相关联的地图。	该列表框显示了包含该特定表的 Workspaces 和 GeoSets。
MatchFields	无	该元素是 MatchField 和 FieldName 元素的容器。
MatchField, FieldName	字段信息列表框。	搜索列表框中选中的每个项目用于实现自动匹配和绑定。

术语表

A

本术语表定义本指南和 MapInfo 产品中有必要理解的术语，以及用于 MapInfo 产品和技术的特定术语。

术语

Codespace

请参阅第 350 页中的 *MapInfo Codespace*。

FeatureLayer

显示表中图元的 MapXtreme 2005 图层。例如表示世界国家的区域对象的图层是 FeatureLayer。FeatureLayer 必须通过 Map 的图层集合添加到 Map。FeatureLayers 可以是本地的 .TAB 数据、远程 RDB、无缝或光栅数据。

GeoDictionary

包含与表有关的信息的 MapXtreme 2005 文件（仅 TAB 文件）。GeoDictionary 用于自动确定应该将应用程序绑定到的表。

GeoDictionaryManager

维护 Geodictionary 的 MapXtreme 2005 工具。

Internet 信息服务 (IIS)

Microsoft 提供的软件服务，支持 web 站点的创建，配置和管理。特定地，某些辅助的 Internet 信息服务包括：FTP（文件传输协议）和 SMTP（简单邮件传输协议）。在 MapXtreme 2005 中，本公司提供的 WMS Server 必须配置为与 IIS 配合工作以运行服务器。

ISession 接口 (ISession Interface)

MapXtreme 2005 MapInfo.Engine 命名空间接口，提供所有 MapXtreme 应用程序的起点。ISession 管理 MapXtreme 应用程序所需的资源的初始化，并定义所有适用于程序实例的数据和功能。

MapControl

MapXtreme 2005 对象，使用户可以在窗体上查看地图。MapControl 拥有地图绘制的目的窗口。它还控制地图的大小并与地图工具交互。MapXtreme 2005 提供 MapControl 的桌面和 web 版本。

MapInfo Codespace

定义和标准的列表，常用于创建 MapInfo 地图和工作空间。MapInfo Codespace 包括坐标系设置；画笔、画刷、距离设置和缩写；图像大小设置；经常使用的类型和它们的缩写；可用的操作符的列表；时间、日期、温度单位设置和缩写。

Non-Earth 地图 (Non-Earth Map)

一种地图，其中的对象没有显式引用地球表面的位置。建筑平面图是典型的示例。

PointRef 构架 (PointRef Schema)

可以应用于不可制图的表，使其可以绘制地图的空间构架。该构架通过将不可制图表的列 (MatchColumn) 中的值和可制图表的列 (RefColumn) 中的值进行匹配，引用了另一个表的 Geometry 对象。表打开时，它包含只读 Geometry 列。然后表可以作为图层添加到 Map。

Unicode 转化格式 -8 (UTF-8)

Unicode 字符的 8 位无损编码。MapXtreme 2005 只按工作空间永久性构架指示的方式支持 UTF-8。

Web 地图服务 (WMS)

WMS 是符合 OGC 标准的 Web 服务，提供在绘制地图的应用程序中用作图层的地图图像。如果需要管理 WMS 服务器和客户端以访问任意的符合 OGC 标准 (1.0.0、1.1.0 或 1.1.1) 的 WMS，则 MapXtreme 2005 将提供 WMS 的服务器实现。

Web 图元服务 (WFS)

符合 OGC 标准的 Web 服务，提供用于绘制地图的应用程序的地理参考地图图元。MapXtreme 2005 提供 WMS Basic 的实现，包括只读服务和访问 WMS 服务器的客户端。

Web 服务器

运行超文本传输协议和 Web 服务器软件的计算机系统。Web 服务器接受来自 Web 用户浏览器的基于 URL 的 HTTP 请求，并将 HTML 页面发送回浏览器。一台 Web 服务器可以管理一个或多个 Web 站点。例如商用服务器，通常同时管理多个 Web 站点。

XY 构架 (XY Schema)

可以应用于不可制图的表，以使其可以绘制地图的空间构架。该表必须包含 X 和 Y 坐标值，该构架访问坐标值以创建表的 Geometry 列。

表 (Table)

以行和列格式组织的数据集合。在 MapXtreme 2005 中，表包含要显示在地图上的数据。表保存描述图元的信息，这些图元包括它们的几何体、样式和属性。MapXtreme 2005 支持各种数据源中的表，包括本地表 (MapInfo .TAB)、关系数据库管理系统 (RDBMS)、dBase、MS Access、ASCII 文件和 ESRI ShapeFile。特性表包括光栅、网格、无缝、视图、WMS 和 ADO.NET。可通过 TableInfo 类查看表的类型。可以通过 Data 命名空间中的 Catalog 打开或关闭表。请参阅第 10 章：处理数据。

超文本标记语言 (HTML)

纯文本 (Ascii) 语言，使开发人员可以创建 Web 页面，这些 Web 页面可以在不同的计算机平台上，通过不同的 Web 浏览器浏览。Html 使用标记以指定文档各部分的结构。Html 支持链接（使用 URL）指向其它 web 文档或文件。

超文本传输协议 (HTTP)

在 Web 客户端和 Web 服务器之间基于消息的网络接口。HTTP 运行于 TCP/IP 之上。

笛卡儿 (Cartesian)

一种坐标系，使用与任何真实世界系统都无关的 x、y 比例。大多数 CAD 绘图对于注册对象（例如，滚珠轴承组件或建筑平面绘图）使用该方法。如果绘图使用笛卡儿坐标，则绘图的一个角的坐标大致为 0, 0。

笛卡儿坐标 (Cartesian Coordinates)

平面上几何体对象通过 x 和 y 值的常规表示。

地理编码 (Geocode)

将 X 和 Y 坐标分配到表或数据库记录的过程，以便记录可以显示为地图上的对象。

地理标记语言 (GML)

特定于绘制地图的标记语言。GML 正在由 Open GIS Consortium (OGC) 开发，OGC 是开发和推广地理标准的国际组织。

地理信息系统 (GIS)

计算机硬件和软件的有条理的集合，专为有效创建、操控、分析和显示所有地理类型或空间参考数据而设计。

工作空间 (Workspace)

基于 XML 的永久性文件格式，使 MapXtreme 2005 的用户可以在各种环境下共享已经创建的地图。这是未来 MapInfo 所有产品都将符合的文件格式。关于创建工作空间的详细信息，请参阅第 19 章：*Workspace* 管理器。关于工作空间结构的详细信息，请参阅附录 G：理解 MapInfo 工作空间。

几何质心 (Geometric Centroid)

不需要包含在对象内部的中心点（请参阅第 355 页中的质心 (*Centroid*)），通常是 FeatureGeometry。

经度 (Longitude)

地图上的垂直直线，从北极点延伸到南极点，用于描述点的东-西位置。该位置以本初子午线（0 度）以东（至 -180.0 度）或以西（至 +180.0 度）的度数表示。经度的直线在赤道分开最远而在极点相交，所以它们是不平行的。

经度 / 纬度 (Longitude/Latitude)

MapInfo 产品的地图中表示地理对象的默认坐标系。

经度的度，纬度的度，小数度 (Degrees Longitude, Degrees Latitude, Decimal Degrees)

经度的度和纬度的度是用于表示地球表面位置的坐标。经度，或 X 坐标，表示位置的东西位置，任何在本初子午线以西位置的 X 值都为负。纬度，或 Y 坐标，表示位置的南北位置，任何在赤道以南位置的 Y 值都为负。

空间构架 (Spatial Schema)

一种可以应用到表来增强表的空间能力的服务。MapXtreme 2005 中有如下两种类型的空间构架: PointRef 和 XY。包含可以引用可制图表列的列, 或包含表示 XY 值的列的不可制图的表, 可以使用上述构架以创建 Geometry 列。然后这些表可以作为图层添加到 Map。关于每种方案类型的详细信息, 请参阅 PointRef 方案和 XY 方案术语表定义。

类 (Class)

在面向对象语言中, 类是包含执行某些函数类型方法的对象或对象集, 在含义上类似于过程语言的派生类型。

枚举 (Enumerate)

包含所有变量和它们可能的值的数据类型。

命名空间 (Namespace)

具有分层结构的命名系统, 用于将独立继承的类进行分组。例如, 两个不相关的同名类可以在不同的命名空间中存在: System.Utilities.FileFinder 和 MyCompany.Ultties.FileFinder 可以有相同的名称, 但是功能不同。此外, 命名空间有助于防止编译器引用错误的类 (即“冲突”)。

派生的类 (Derived Class)

包含其基类所有特性, 且包含关于基类的附加功能或增强功能的类。

区域 (Region)

区域是带有一个外部 Ring 和零个或多个内部 Ring (洞) 的 MultiPolygon。

事件句柄 (Event Handler)

页面对象的属性, 可以使用 JavaScript 或 VBScript 编写。例如, 事件句柄描述用户单击按钮或选中列表框中的文本时要执行的操作。VBScript 和 JavaScript 都支持显式定义的事件句柄, 例如在单击和选择中。此外, 可以定义函数替换显式事件句柄。这样的函数被称为隐式事件句柄。

数据绑定 (Data Binding)

数据源和服务器控件的关联。MapXtreme 2005 DataBinding 类包含关于 ASP.NET 服务器控件中单一数据绑定表达式的信息, 该控件使 Visual Studio .NET 的快速应用程序开发 (RAD) 设计器可以在设计阶段创建数据绑定表达式。

通用字符集 (UCS)

国际标准 ISO 10646 定义了通用字符集 (UCS)。UCS 是所有其它字符集标准的超集。UCS 也定义了多种将字符串编码为字节序列的方法, 例如 UTF-8 和 UTF-16。

统一资源定位符 (URL)

超文本链接或图像地图的基础实现, 包含万维网上 Web 页面或文件的地址。URL 包含要使用的网络协议信息 (通常是 HTTP) 和页面或文件的路径。URL 的示例如:

“<http://www.mycompany.com/index.html>” , 指向 “my company” web 站点的索引页面。

投影 (Projection)

将地球表面图元位置转换为二维表面位置（例如图纸地图）的数学模型。由于地图尝试在平面上表示球面对象（地球），所以所有投影都在一定程度上失真。地图投影可以保持面积、距离、形状或方向，但只有球形可以保持以上所有属性。某些投影（例如，Mercator）生成非常适合于导航的地图。其它投影（例如 Lambert 一类的等面积投影）生成非常适合于可视化分析的地图。

凸多边形缓冲区 (Convex Hull Buffer)

一种缓冲区类型，创建区域对象，表示基于输入对象节点的多边形。可以认为该凸多边形是在所有点周围放置橡皮圈的一个操作符。它由最小数目的点组成以便使所有点都位于多边形上或多边形内。使用凸多边形缓冲区，就没有大于 180 度的内角。

图层 (Layer)

MapInfo 产品中显示的地图基本组件，通常由多个有层次的图层组成（例如，街道数据图层叠放在郡的图层或邮政编码边界之上）。表在 Map 窗口出现时，它占用了该 Map 窗口的一个图层。通常，每个地图图层对应一个打开的表。

图元 (Feature)

表中具有几何体、样式和属性的行。Feature 通常具有 Table 和 Key 来标识所表示的行。

网格图 (Graticule)

显示在地球地图上的水平（纬度）直线和垂直（经度）直线构成的网格，间距为固定的距离（例如，每五度一格，每十五度一格）。用于建立参考框架。

纬度 (Latitude)

地图上的水平直线，从赤道的 0 度到北极点 (+ 90.0 度) 和南极点 (-90.0 度)。用于描述点的南 - 北位置，通常以在赤道上方或下方的度或小数度为单位度量。

小数度 (Decimal Degree)

度的分数部分的小数表示。很多地图以度、分、秒为单位（例如，40_30i10I）表示坐标，其中分和秒就是度的分数部分。30 分等于半度，30 秒等于半分。MapXtreme 2005 以小数度表示坐标（例如，72.558 度）。因此，经度：40 度 30 分，在 MapXtreme 2005 中将表示为 40.5 度。

修饰 (Adornment)

由图例、标题或比例栏组成的 MapXtreme 2005 地图元素。

永久性 (Persistence)

永久性是指 MapXtreme 2005（和其它 MapInfo 产品）管理数据的方法并确保使用该 API 创建的地图可以被其它 MapXtreme 用户使用。永久化和基于 XML 的工作空间的加载和保存有关，用于将 MapInfo Geometry 分析和发布到 GML，反之亦然。

质心 (Centroid)

通常是地图对象的中心。对于大多数地图对象，质心位于对象的中间位置（对象南北方向长度中点和东西方向长度中点的位置）。在某些情况下，因为有质心必须位于对象自身的限制，所以质心可能不在中点位置。因此，对于新月形区域对象的情况，实际上对象的中点可能位于限制区域之外；不过，本质心始终应该在限制区域内。

子午线 (Meridian)

从北极点延伸到南极点的直线或直线的一部分。子午线是经线。

字符编码 (Character Encoding)

一种将序列字节转换为序列字符的方法。另请参阅第 353 页中的通用字符集 (UCS) 和第 351 页中的 *Unicode* 转化格式 -8 (UTF-8)。

坐标 (Coordinate)

笛卡儿坐标系的 x、y 位置，或者地球坐标系的经度、纬度位置。坐标表示地图上相对于其它位置的位置。地球坐标系可以使用赤道和格林威治本初子午线作为固定参考点。平面坐标系根据到固定参考点的距离描述二维 x、y 位置，通常使用第一象限，所以所有坐标值都是正数。

坐标系统

坐标系用于创建几何体对象的数字表示。几何体对象的每个点由一对数字表示。这些数字是点的坐标。在绘图学中，坐标系与投影密切相关。可以通过提供投影参数的特定值来创建坐标系。

B

MapInfo SQL 参考

本参考定义了构成 MapInfo SQL 语言的函数和运算符。

本附录内容：

- ◆ *MapInfo SQL* 语言概述 358
- ◆ *IN/ANY/ALL* 运算符和子查询 359
- ◆ *MapInfo SQL* 命令 360
- ◆ 运算符 362
- ◆ 函数列表 365
- ◆ 日期格式化字符串 406
- ◆ 数字格式化字符串 408

MapInfo SQL 语言概述

MapXtreme 2005 对象通过访问数据的 MapInfo ADO.NET 数据提供方（特别是 `MICommand` 对象）向用户公开了 SQL 处理。此外，表达式也用于标注、主题、图例、`AddColumns`、图元搜索和选择集处理。MapInfo Corporation 从 MapXtreme 2005 开始新的基于 .NET 的产品，使用本公司最新开发的 SQL 语言。

关于表、图元和 MapInfo 数据提供方的详细信息请参阅第 10 章：处理数据。

MapX 和 MapXtreme 的较早版本使用 SQL 语法，该语法是 MapBasic 的子集。它通过 `Layer.Search` 方法公开给 MapX 和 MapXtreme 的 Windows 开发人员。MapX 也可以打开并查看 TAB 文件（包含 `Create View` 语句的 TAB 文件）。包含在 TAB 文件中的 `Select` 语句的语法也取决于该语法。

MapBasic SQL（和 MapX `Layer.Search` 语法）和新 MapInfo SQL 之间的不同之处主要在以下方面：

- 空间运算符
- 字符串常量
- 引用标识符
- 函数

空间运算符

两种语言都包括 `within` 和 `contain` 运算符，但它们的含义很不相同并影响到结果和性能。如果运算符左侧 (lhs) 的几何体质心位于运算符右侧 (rhs) 的几何体内部，则 MapBasic 的 `within` 运算符返回 `true`。

新 SQL (MapInfo SQL) 不使用左侧几何体的质心；而是测试判断整个左侧几何体是否在右侧几何体内。这等价于 MapBasic 中的“Entirely Within”运算符。同样，MapBasic 中的 `contain` 运算符测试判断左侧几何体是否包含右侧几何体的质心。MapInfo SQL 的 `contain` 运算符再次不使用质心判断，而是将判断应用于左侧和右侧的几何体（并执行与 MapBasic 的“Contains Entire”运算符等价的测试）。MapInfo SQL 公开了执行基于质心的测试的 `CentroidWithin` 和 `ContainsCentroid` 运算符。

字符串常量

MapBasic 使用双引号标识字符串常量。标准 SQL（由 SQL-3 规范指定）使用单引号标识字符串常量。双引号则用于引用标识符的名称（参阅以下的“引用标识符”）。

引用标识符

新 MapInfo SQL 遵循标准 SQL 约定并使用双引号以引用标识符，如果没有引号，就不能够准确地分析标识符。例如，名称中含有空格的表名“United States”如果没有引号将不能够正确分析。MapBasic 则禁用引用标识符。例如，由 MapInfo Professional 修改的表名、列等，通过将空格和其它标注字符替换为底线字符的方式分析。因此，打开表“United States.TAB”时，表名将变为 United_States。MapXtreme 2005 将不再为以上目的修改表名。

函数

MapX (MapBasic 的子集中) 支持的大多数函数现在已经更改。大多数函数更改了名称，其它的则更改了参数标志。

表示视图的 TAB 文件（例如，包含 Create View 语句）使用 SQL 语法以定义视图。取决于 TAB 文件的版本 (TAB 文件中的另一个显式属性集)，SQL 可以是 MapBasic SQL 或 MapInfo SQL。对于版本号小于 800 的 TAB 文件，视图语法必须是 MapBasic 语法以便保持文件与更早版本 MapInfo Professional 的兼容性。版本号 800 或更高版本的 TAB 文件必须包含新 MapInfo SQL。这些 TAB 文件将不再与早期版本的 MapInfo Professional 兼容。

IN/ANY/ALL 运算符和子查询

下面是与数据访问相关的新的和增强的功能。

- 现在提供带有文字列表和子查询的 IN、ANY 和 ALL 运算符。例如：

```
Select ... From T Where X IN (2, 6, 12)
Select ... From T Where X IN
(Select X From U Where B > 50)
```

- 现在提供相关联的子查询，其中子查询引用来自外部查询的列（或表达式）。例如：

```
Select ... From T Where X IN (Select X From U Where B = T.A)
```

- 空间 OR 列表已进行优化。例如：

```
{Select ... From T Where Obj Within @Var1 OR Obj Within @Var2 OR
Obj Within @Var3}
```

MapInfo SQL 命令

组成 MapInfo SQL 语言的命令有: Select、Insert、Update 和 Delete。

Select

```
SELECT < select_list >
    FROM { < table_source > } [ ,...n ]
    [ WHERE < search_condition > ]
    [ GROUP BY expression [ ,...n ] ]
    [ ORDER BY {expression | column_position [ ASC | DESC ] } [ ,...n ]]

< select_list > ::==
    {
        *
        | { table_name | table_alias }.*
        | { expression } [ [ AS ] column_alias ]
    } [ ,...n ]

< table_source > ::=
    table_name [ [ AS ] table_alias ]
```

Insert

```
{ VALUES ({expression | NULL} [, ...n]) | query_specification
```

Update

```
UPDATE { table_name }
    SET {{ column_name }} = { expression | NULL } } [, ...n]
    [WHERE < search_condition > ]
```

Delete

```
DELETE [FROM] { table_name } [ WHERE < search_condition > ]  
  
< search_condition > ::=  
  { [ NOT ] < predicate > | ( < search_condition > ) }  
  [ { AND | OR } [ NOT ] { < predicate > | ( < search_condition  
> ) } [ ,...n ] ]  
  
< predicate > ::=  
  {  
    expression [ { = | < > | != | > | >= | < | <= } expression ]  
    | string_expression [ NOT ] LIKE string_expression [ ESCAPE  
      'escape_character' ]  
    | expression [ NOT ] BETWEEN expression AND expression  
    | expression IS [ NOT ] NULL  
  }
```

表达式

表达式是列名、伪列、列的别名、常数、函数，或是用运算符连接的以上项目的任意组合。列名和伪列可以是以表名或表别名后接点 (".") 字符为前缀。

注： 子查询和关联的运算符 IN、ANY、SOME 和 ALL 当前不被 MapInfo 数据提供程序支持。

GROUP BY

GROUP_BY_expression 是选择列表的列或选择列表表达式的匹配副本的参考，别名 - 基于 1 的数字，指示列的位置，或 *n* 是表示列的数字的 col*n* 位置。

ORDER BY

ORDER_BY_expression 是选择列表的列或选择列表表达式的匹配副本的参考，别名 - 基于 1 的数字，指示列的位置，或 *n* 是表示列的数字的 col*n* 位置。

MapInfo SQL Language 基于 SQL-3 和 SQL/MM 标准。字符串文字必须用单引号封闭 ('example')，而标识符（列名称、表名称、别名等）应该以双引号封闭（“example identifier”）（如果有必要）。只有分析的逻辑不能正确分析标识符，标识符才需要加引号。这种情况包括名称中有空格或有其它特殊字符的标识符。

MapXtreme 2005 忽略使用 TableInfoView() 的查询中的 OrderBy 子句。但是并不会抛出异常。这样设计是为了响应 RDBMS 数据库中的不同行为（如，Oracle 允许该行为，而 SQL Server 不允许该行为）。

运算符

MapInfo SQL 支持以下运算符：

符号	用法
+	加法运算符；也是连接运算符。 注： MapBasic 使用 “&”
-	减法运算符；也是一元负号
*	乘法运算符
/	除法运算符
^	乘方运算符
=	相等运算符（对于字符串则是区分大小写）
<>	不等运算符
!=	不等运算符
<	小于运算符
>	大于运算符

符号	用法
<=	小于或等于运算符
>=	大于或等于运算符
(表达式分隔符
)	表达式分隔符
%	通配符
,	列表项与函数参数分隔符
@ 或 :	参数名 - 参数名需要以 @ 或 : 开头。符号。
"	字符串常量分隔符
'''	引用标识符分隔符
ALL	等于文字列表或子查询的所有值
ANY	等于文字列表或子查询的任一值
IN	等于文字列表或子查询的一个值
Within	第一个对象完全在第二个对象中。 (与 MapBasic 的 Entirely Within 相等)
Contains	第二个对象完全在第一个对象中。 (与 MapBasic 的 Entirely Within 相等)
Intersects	第一个对象与第二个对象相交 (与 MapBasic 的 Intersects、Partly Within 和 Partly Contains 相等)
CentroidWithin	第一个对象的质心在第二个对象内部 (与 MapBasic 的 Within 相等)
ContainsCentroid	第二个对象的质心在第一个对象内部 (与 MapBasic 的 Contains 相等)
EnvelopesIntersect	测试操作数的包络 (MBR) 是否相交

运算符优先级

MapXtreme 2005 计算 SQL 表达式时，需要明确表达式的哪个运算符先计算。这称为优先级。按约定，特定的运算符被分配到不同的优先等级。优先等级高的运算符先计算。下表按计算的先后顺序列出了 MI SQL 运算符。相同优先等级的运算符按从左到右的顺序计算，乘方例外，它是从右侧开始计算。

最高优先级：

- 括号
- 乘方
- 负号
- 乘法、除法
- 加法、减法
- 地理运算符
- 比较运算符
- Not
- And

最低优先级：

Or

例如，表达式 $3+4*2$ 计算的结果是 11。这是因为乘法的优先级比加法的优先级高，所以先执行乘法运算。要强制 MI SQL 先执行加法运算，必须使用括号： $(3+4)*2$ 。

现在考虑以下表达式，作用是选择 1999 年七月或九月的所有记录。

`Year(RECEIVED)=99 and Month(RECEIVED)=7 or Month(RECEIVED)=9`

由于“and”具有比“or”高的优先级，MapXtreme 2005 将该表达式

“`year(RECEIVED)=99 and month(RECEIVED)=7`”作为括号封闭对待。在这种情况下，将选择 1999 年七月或任意年九月的任意记录。这或许不是用户需要的结果。不过，通过为第二个表达式添加括号，用户可以获得预期的结果。

`(RECEIVED)=99 and (month(RECEIVED)=7 or month(RECEIVED)=9)`

用户不清楚 MapXtreme 2005 如何计算多个运算符的表达式时，应该使用括号将运算符元素按需要分组。

函数列表

下表定义了 MapInfo SQL 支持的所有函数，按标题分类。

- 数值函数
- 字符串函数
- 日期函数
- 转换函数
- 几何体构造函数
- 几何体谓词函数（布尔）
- 几何体观测函数（返回标量值的函数）
- 几何体测量函数
- 几何体函数（返回几何体的函数）
- 集合函数
- 几何体集合函数
- 光栅 / 网格函数
- 其它函数

关于上述函数的语法，请参阅第 370 页中的按字母顺序排列的函数列表。

函数名称	描述
数值函数	
Abs	计算绝对值
Acos	计算反余弦函数
Asin	计算反正弦函数
Atan	计算反正切函数
Ceil	始终返回大于参数的最小整数值
Cos	计算余弦函数
Exp	返回 e (自然对数为底数) 的参数次幂的值
Floor	始终返回小于参数的最大整数值
Log	计算自然对数函数

函数名称	描述
Mod	取模
Round	将 X 取整到最接近 m 的倍数的值。参数 m 是可选的，如果缺少则取整到最接近的整数
Sign	如果参数为负值、零或正值，则返回参数的对应符号 -1、0 或 1。
Sin	计算正弦函数
Sqrt	计算平方根函数
Tan	计算正切函数
Trunc	将数字截断到指定小数位数
字符串函数	
Char_Length	返回字符串参数中字符的数目
InStr	在一个字符串中搜索另一个字符串
Left	返回字符串中最左侧的指定个字符
Lower	将字符串转换到小写
Ltrim	从字符串中删除首部的空字符，返回生成的字符串
Right	返回字符串中最右侧的指定个字符
Rtrim	从字符串中删除尾部的空字符，返回生成的字符串
空间	返回由指定数目的空字符组成的字符串
Substring	提取字符串的一部分来形成新的字符串
Trim	从字符串中删除首部和尾部的空字符，返回生成的字符串
Upper	将字符串转换到大写
日期函数	
Current_Date	返回当前的日期。注：尚未确定
Day	以整数返回日期的日部分
Month	以整数返回日期的月部分
Weekday	返回从 1 到 7 的整数，指示指定日期是周几

函数名称	描述
Year	以整数返回日期的年部分
转换函数	
Ascii	返回字符串中首字符的字符代码
Char	返回长度为 1 的字符串，对应指定的字符代码
NumberToDate	从数字创建 Date
StringToDate	从字符串创建 Date
DateToNumber	将日期转换为数字
StringToNumber	将字符串转换为数字
DateToString	将日期转换为字符串
NumberToString	将数字转换为字符串
几何体构造函数	
MI_Box	创建新的 Feature Geometry 对象，表示矩形
MI_Point	创建新的 Feature Geometry 对象，表示一个点
几何体谓词函数（布尔函数）	
MI_Contains	确定一个几何体对象是否完全包含另一个几何体对象
MI_Intersects	确定一个几何体对象是否与另一个几何体对象相交
MI_Within	确定一个 Geometry 对象是否完全在另一个 Geometry 对象之内
MI_CentroidWithin	确定一个几何体对象的质心是否位于另一个几何体对象内
MI_ContainsCentroid	确定一个几何体对象的质心是否位于另一个几何体对象内
MI_EnvelopesIntersect	确定两个几何体对象的最小边界矩形 (MBR) 是否相交
几何体观测函数（返回标量值的函数）	
MI_CentroidX	返回几何体对象质心的 x 坐标
MI_CentroidY	返回几何体对象质心的 y 坐标
MI_MinX	返回几何体的 MBR 的最小 x 值

函数名称	描述
MI_MinY	返回几何体的 MBR 的最小 y 值
MI_MaxX	返回几何体的 MBR 的最大 x 值
MI_MaxY	返回几何体的 MBR 的最大 y 值
MI_GeometryType	返回指示几何体对象类型的字符串
MI_X	返回 Point 几何体的 x 坐标
MI_Y	返回 Point 几何体的 y 坐标
几何体测量函数	
MI_Area	返回几何体对象的地理面积
MI_CentroidDistance	返回两个几何体对象质心之间的地理距离
MI_Length	返回直线或折线几何体对象的地理长度
MI_Perimeter	返回几何体对象的地理周长
几何体函数（返回几何体的函数）	
MI_Buffer	返回 MultiPolygon 几何体对象，表示环绕另一个几何体对象的缓冲距离。有效缓冲区分辨率为 3 到 100（包括 3 和 100）。
MI_Centroid	以 Point 几何体对象返回几何体对象的质心
MI_Envelope	返回 Envelope 几何体对象，表示输入几何体的最小边界矩形 (MBR)
MI_Union	返回几何体对象，表示两个输入几何体对象的联合
MI_Intersection	返回几何体对象，表示两个输入几何体对象之间的交集
MI_ConvexHull	返回 MultiPolygon 几何体对象，表示在输入几何体所包含的点集中的凸多边形
MI_Transform	将几何体从一个坐标系转换到另一个坐标系
集合函数	
Avg	计算合计值的平均值
Count	计算合计值中值的数目
Sum	计算合计值之和

函数名称	描述
Min	计算合计值的最小值
Max	计算合计值的最大值
WtAvg	计算值和加权合计的加权平均值
ProportionSum	计算合计值的和，该值基于两个几何对象所占比例进行调节
ProportionAvg	计算合计值的平均值，该值基于两个几何对象所占比例进行调节
ProportionWtAvg	计算合计值的加权平均值，该值基于两个几何对象所占比例进行调节
MI_AggregateIndividual	从合计中返回任何单值
几何体集合函数	
MI_AggregateBuffer	返回 MultiPolygon 几何体对象，表示环绕合计几何体对象的缓冲距离
MI_AggregateConvexHull	返回 MultiPolygon 几何体对象，表示在几何体对象合集所包含的点集中的凸多边形
MI_AggregateEnvelope	返回 Envelope 几何体对象，表示几何体对象合集最小边界矩形 (MBR)
MI_AggregateIntersection	返回几何体对象，该对象表示几何体对象合集的交集
MI_AggregateUnion	返回几何体对象，该对象表示几何体对象合集的联合
光栅 / 网格函数	
MI_ImageFile	从 ImageInfo (Raster 或 Grid) 返回文件名
MI_ImagePixelWidth	返回光栅或网格图像的宽度 (单位为像素)
MI_ImagePixelHeight	返回光栅或网格图像的高度 (单位为像素)
MI_GridValueAtPixel	返回指定像素位置的 Grid 的值
MI_GridMinValue	返回最小网格值
MI_GridMaxValue	返回最大网格值
其它函数	
MI_EngineVersion	指示引擎的版本
MI_SQLVersion	指示引擎当前支持的 SQL 语言的版本

按字母顺序排列的函数列表

Abs

计算绝对值

语法

`Abs (d)`

参数

`d` 为双精度数

Acos

计算反余弦函数

语法

`Acos (d)`

参数

`d` 是要计算反余弦的双精度数，取值在 -1 到 1 之间。

Ascii

返回字符串中首字符的字符代码。

语法

`Ascii (str)`

参数

`str` 是字符串。

说明

`Ascii` 返回 `str` 中首字符的字符代码。如果 `str` 是空字符串，则 `Ascii` 返回零。如果 `str` 是空，则 `Ascii` 返回空。

Asin

计算反正弦函数

语法

`Asin (d)`

参数

`d` 是要计算反正弦的双精度数。

Atan

计算反正切函数

语法

`Atan (d)`

参数

`d` 是要计算反正切的双精度数。

Avg

计算合计值的平均值。

语法

`Avg (n)`

参数

`n` 是数字表达式。

说明

`Avg` 是仅可用于合计 Select 语句的合计函数。空值被忽略。如果仅在合计中提供空值，则返回空值。

Ceil

Ceiling - 始终返回大于参数的最小整数值

语法

`Ceil (d)`

参数

d 为双精度数

Char

返回长度为 1 的字符串，对应指定的字符代码。

语法

`Char (n)`

参数

n 表示与字符等价的二进制数。

说明

如果 n 为空，则 Char 返回空。

Char_Length

返回字符串参数中字符的数目

语法

`Char_Length (s)`

参数

s 是字符串表达式。

说明

如果 s 为空，则 Char_Length 返回 (0)。

Cos

计算余弦函数

语法

```
Cos ( d )
```

参数

d 是要计算余弦的双精度数。

Count

计算合计值中值的数目。

语法

```
Count ( expression )
Count ( * )
```

参数

expression 是任何数据类型的任意标量表达式。

说明

Count 是仅可用于合计 Select 语句的合计函数。除在特殊形式 Count(*) 外，空值被忽略。

Current_Date

返回当前的日期

语法

```
Current_Date ()
```

说明

注： Current_Date 函数尚未确定。

DateToNumber

将日期转换为数字。

语法

```
DateToNumber ( date )
```

参数

date 是日期。

说明

DateToNumber 将日期值 (date) 转换到八位整数，以 YYYYMMDD 表示日期。

DateToString

将日期转换为字符串。

语法

```
DateToString ( date, fmt )
```

参数

date 是日期型，而 fmt 是标识如何格式化 date (日期型) 的字符串。

说明

关于使用 fmt 参数格式化日期的详细信息，请参阅第 406 页中的日期格式化字符串部分。

Day

以整数返回日期的日部分。

语法

```
Day ( date )
```

参数

date 是 Date 表达式。

说明

Day 返回 1 到 31 之间的整型值。如果 date 为空，则 Day 返回空。

Exp

返回 e (自然对数为底数) 的参数次幂的值

语法

`Exp (d)`

参数

d 为双精度数

Floor

始终返回小于参数的最大整数值

语法

`Floor (d)`

参数

d 为双精度数。

InStr

在一个字符串中搜索另一个字符串。

语法

`InStr (str1, str2[, start_at[, occurrence]])`

参数

str1 是要在其中进行搜索的字符串,

str2 是要搜索的字符串,

start_at 是整数, 表示 str1 内 (从一开始) 开始搜索的部分,

occurrence 是整数, 表示 str2 在 str1 中出现的次数。

说明

返回 str1 中 (从一开始) 的位置, 它是 str2 所需次数的第一个字符。如果所需的 str2 没有找到, 或者 str1 或 str2 为空, 那么返回 0。参数 start_at 和 occurrence 是可选的, 均默认为 1。

Left

返回字符串中最左侧的指定个字符。

语法

```
Left ( str, len )
```

参数

str 是字符串，而

len 是大于或等于零的整数。

说明

Left 返回字符串，包含了字符串表达式 str 最左侧的 len 个字符。len 参数应该是大于或等于零的整数。如果 len 等于零，将返回空值。如果 len 大于或等于字符串 str 的长度，那么将返回字符串 str 的全部。

Log

计算自然对数函数

语法

```
Log ( d )
```

参数

d 为双精度数。

Lower

将字符串转换到小写。

语法

```
Lower ( str )
```

参数

str 是字符串

Ltrim

从字符串中删除首部的空字符，返回生成的字符串。

语法

```
Ltrim ( str )
```

参数

str 是字符串

Max

计算合计值的最大值。

语法

```
Max ( n )
```

参数

n 是数字表达式。

说明

Max 是仅可用于合计 Select 语句的合计函数。空值被忽略。如果仅在合计中提供空值，则返回空值。

MI_AggregateBuffer

返回 MultiPolygon 几何体对象，表示环绕合计几何体对象的缓冲距离。

语法

```
MI_AggregateBuffer ( geometry, dist, unit, calculationType, res )
```

参数

geometry 是返回几何体的表达式，

dist 是数值，表示缓冲区自源几何体的距离，

unit 是面积单位规范，

calculationType 是字符串，指示是执行“球面”还是“笛卡儿”计算。没有默认值，而

res 是数值，指定最小的线段数，用于从缓冲点创建圆。

MI_AggregateConvexHull

返回 MultiPolygon 几何体对象，表示在几何体对象合集所包含的点集中的凸多边形。

语法

```
MI_AggregateConvexHull ( geometry )
```

参数

geometry 是返回几何体的表达式，

MI_AggregateEnvelope

返回 Envelope 几何体对象，表示几何体对象合集最小边界矩形 (MBR)。

语法

```
MI_AggregateEnvelope ( geometry )
```

参数

geometry 是返回几何体的表达式。

说明

返回的几何体对象会在输入 geometry 的坐标系中。

MI_AggregateIndividual

从合计值中返回任何单值。

语法

```
MI_AggregateIndividual (expression)
```

参数

expression 是任何表达式

说明

当合计多个记录时，偶尔需要从列（或表达式）中想要的值而没有在 GroupBy 子句中指定列或表达式。可以使用 MI_AggregationIndividual 完成这个功能。

MI_AggregateIntersection

返回几何体对象，该对象表示几何体对象合集的交集。

语法

```
MI_AggregateIntersection ( geometry )
```

参数

geometry 是返回几何体的表达式。

MI_AggregateUnion

返回几何体对象，该对象表示几何体对象合集的联合。

语法

```
MI_AggregateUnion ( geometry )
```

参数

geometry 是返回几何体的表达式。

MI_Area

返回几何体对象的地理面积。

语法

```
MI_Area ( geometry, unit, calculationType )
```

参数

geometry 是返回几何体的表达式，

unit 是面积单位规范，

calculationType 是字符串，指示是执行“球面”还是“笛卡儿”计算。没有默认值。

说明

返回值为双精度数，以 units 参数指定的单位表示。MI_Area 仅适用于 Envelope、Polygon、Multipolygon 和 Ring 类型的几何体。任何其它类型的输入几何体都导致零值。

单位的有效值是下面的面积单位:

值	描述
sq mi	平方英里
sq km	平方公里
sq in	平方英寸
sq ft	平方英尺
sq yd	方码
sq mm	平方毫米
sq cm	平方厘米
sq m	平方米
sq survey ft	平方英尺 (US Survey)
sq nmi	平方海里
sq twip	平方 twips
sq pt	平方 points
sq pica	平方 picas
sq degree	平方度
acre	英亩
hectare	公顷
sq li	平方 links
sq ch	平方 chains
sq rd	方杆
perch	perches
rood	鲁德 (土地单位)

MI_Buffer

返回 MultiPolygon 几何体对象，表示环绕另一个几何体对象的缓冲距离。

语法

```
MI_Buffer ( geometry, dist, unit, calculationType, res )
```

参数

geometry 是返回几何体的表达式，

dist 是数值，表示缓冲区自源几何体的距离，

unit 是面积单位规范，

calculationType 是字符串，指示是执行“球面”还是“笛卡儿”计算。没有默认值，而

res 是数值，指定最小的线段数，用于从缓冲点创建圆。MI_Buffer 的有效缓冲区分辨率为 3 到 100（包括 3 和 100）。

说明

返回的几何体对象与提供的 geometry 参数处于相同的坐标系中。距离 (dist) 为 units 参数指定的单位。如果 res 为空或者为零，就使用相应的默认值。否则，res 必须最少为 3。用于分数圆线段的最小数值用于分数圆（例如当缓冲矩形的角时形成的圆）会按比例更小。（例如，半圆会至少有 res/2 段）。距离 dist 可以为正也可以为负。

单位的有效值是下面的距离单位：

值	描述
mi	英里
km	公里
in	英寸
ft	英尺
yd	码
mm	毫米
cm	厘米
m	米

值	描述
survey ft	英尺 (US Survey)
nmi	海里
twip	twips
pt	点
pica	picas
degree	度
li	links
ch	chains
rd	杆

MI_Box

创建新的 Feature Geometry 对象，表示矩形。

语法

```
MI_Box ( x1, y1, x2, y2, csys )
```

参数

x1、y1、x2 和 y2 是双精度数，标识框的边界，

csys 是坐标系规范

说明

csys 参数可以是对边界坐标系对象的引用，或是使用 codespace:code 表示法的坐标系字符串表示。

MI_Centroid

以 Point 几何体对象返回几何体对象的质心。

语法

```
MI_CentroidY ( geometry )
```

参数

geometry 是返回几何体的表达式。

说明

返回的点几何体对象会在输入 geometry 的坐标系中。

MI_CentroidDistance

返回两个几何体对象质心之间的地理距离。

语法

```
MI_CentroidDistance ( geom1, geom2, unit, calculationType )
```

参数

geom1 是返回几何体的表达式,

geom2 是返回几何体的表达式,

unit 是线性单位规范,

calculationType 是字符串, 指示是执行“球面”还是“笛卡儿”计算。没有默认值。

说明

计算两点之间的距离。如果任一几何体参数 geom1 或 geom2 不是点几何体, 则在计算中使用几何体的质心。返回值为双精度数, 以 units 参数指定的单位表示。

关于单位的有效值请参阅第 381 页中的 *MI_Buffer*:

MI_CentroidWithin

确定一个几何体对象的质心是否位于另一个几何体对象内。

语法

```
MI_CentroidWithin ( testGeometry, containerGeometry )
```

参数

testGeometry 和 containerGeometry 是几何体对象（或表达式）。

说明

如果 testGeometry 的质心位于 containerGeometry 内，则返回 true，否则返回 false。如果 containerGeometry 或 testGeometry 为空，则返回 false。函数使用第一个参数 (testGeometry) 的坐标系执行。

MI_CentroidX

返回几何体对象质心的 x 坐标。

语法

```
MI_CentroidY ( geometry )
```

参数

geometry 是返回几何体的表达式。

说明

返回值为双精度数，根据输入 geometry 的坐标系来表示值。

MI_CentroidY

返回几何体对象质心的 y 坐标。

语法

```
MI_CentroidY ( geometry )
```

参数

geometry 是返回几何体的表达式。

说明

返回值为双精度数，根据输入 geometry 的坐标系来表示值。

MI_Contains

确定一个几何体对象是否完全包含另一个几何体对象

语法

```
MI_Contains ( containerGeometry, testGeometry )
```

参数

containerGeometry 和 testGeometry 是几何体对象（或表达式）。

说明

如果 containerGeometry 完全包含 testGeometry，则返回 true，否则返回 false。如果 containerGeometry 或 testGeometry 为空，则返回 false。函数使用第一个参数 (containerGeometry) 的坐标系执行。

与 MI_Within (testGeoemtry, containerGeometry) 功能相同。

MI_ContainsCentroid

确定一个几何体对象的质心是否位于另一个几何体对象内。

语法

```
MI_ContainsCentroid ( containerGeometry, testGeometry )
```

参数

containerGeometry 和 testGeometry 是几何体对象（或表达式）。

说明

如果 testGeometry 的质心位于 containerGeometry 内，则返回 true，否则返回 false。如果 containerGeometry 或 testGeometry 为空，则返回 false。函数使用第一个参数 (containerGeometry) 的坐标系执行。

MI_ConvexHull

返回 MultiPolygon 几何体对象，表示在输入几何体所包含的点集中的凸多边形。

语法

```
MI_ConvexHull ( geometry )
```

参数

geometry 是返回几何体的表达式，

MI_EngineVersion

指示引擎的版本。

语法

```
MI_EngineVersion
```

MI_Envelope

返回 Envelope 几何体对象，表示输入几何体的最小边界矩形 (MBR)。

语法

```
MI_Envelope ( geometry )
```

参数

geometry 是返回几何体的表达式。

说明

返回的几何体对象会在输入 geometry 的坐标系中。

MI_EnvelopesIntersect

确定两个几何体对象的最小边界矩形 (MBR) 是否相交。

语法

```
MI_EnvelopesIntersect  
( geometry1, geometry2 )
```

参数

geometry1 和 geometry2 是几何体对象（或表达式）。

说明

如果 geometry1 的 MBR 与 geometry2 的 MBR 相交则返回 true，否则返回 false。如果 geometry1 或 geometry2 为空，则返回 false。函数使用第一个参数 (geometry1) 的坐标系执行。

MI_GeometryType

返回指示几何体对象类型的字符串。

语法

```
MI_GeometryType ( geometry )
```

参数

geometry 是返回几何体的表达式。

说明

返回值是以下之一（在 MapInfo.Geometry.GeometryType 枚举里也有）

- Envelope
- 点
- MultiPoint
- MultiCurve
- MultiPolygon
- FeatureGeometryCollection
- Curve
- Ring
- Polygon
- LineString

MI_Grid.MaxValue

以双精度数返回最大的网格值。

语法

```
MI_Grid.MaxValue ( GridInfo )
```

参数

GridInfo 是返回 GridInfo 的表达式。

说明

光栅表公开 RasterInfo，网格表以 MI_Raster 列的值公开 GridInfo。该函数只适用于 Grid 图像。

MI_GridMinValue

以双精度数返回最小的网格值。

语法

```
MI_GridMinValue ( GridInfo )
```

参数

GridInfo 是返回 GridInfo 的表达式。

说明

光栅表公开 RasterInfo，网格表以 MI_Raster 列的值公开 GridInfo。该函数只适用于 Grid 图像。

MI_GridValueAtPixel

以双精度数返回指定像素位置的 Grid 的值。

语法

```
MI_GridValueAtPixel ( GridInfo, XPixel, YPixel )
```

参数

GridInfo 是返回 GridInfo 的表达式，

XPixel 是返回表示 X 像素坐标的整数的表达式，

YPixel 是返回表示 Y 像素坐标的整数的表达式

说明

光栅表公开 RasterInfo，网格表以 MI_Raster 列的值公开 GridInfo。该函数只适用于 Grid 图像。

MI_ImageFile

以字符串从 ImageInfo（Raster 或 Grid）返回文件名。

语法

```
MI_ImageFile ( ImageInfo )
```

参数

ImageInfo 是返回 RasterInfo 或 GridInfo 的表达式。

说明

光栅表公开 RasterInfo，网格表以 MI_Raster 列的值公开 GridInfo。

MI_ImagePixelHeight

以整数返回光栅或网格图像的高度（单位为像素）。

语法

```
MI_ImagePixelHeight ( ImageInfo )
```

参数

ImageInfo 是返回 RasterInfo 或 GridInfo 的表达式。

说明

光栅表公开 RasterInfo，网格表以 MI_Raster 列的值公开 GridInfo。

MI_ImagePixelWidth

以整数返回光栅或网格图像的宽度（单位为像素）。

语法

```
MI_ImagePixelWidth ( ImageInfo )
```

参数

ImageInfo 是返回 RasterInfo 或 GridInfo 的表达式。

说明

光栅表公开 RasterInfo，网格表以 MI_Raster 列的值公开 GridInfo。

MI_Intersection

返回几何体对象，表示两个输入几何体对象之间的交集。

语法

```
MI_Intersection ( geometry1, geometry2 )
```

参数

geometry1 是返回几何体的表达式，

geometry2 是返回几何体的表达式。

MI_Intersects

确定一个几何体对象是否与另一个几何体对象相交。

语法

```
MI_Intersects ( geometry1, geometry2 )
```

参数

geometry1 和 geometry2 是几何体对象（或表达式）。

说明

如果 geometry1 与 geometry2 相交则返回 true，否则返回 false。如果 geometry1 或 geometry2 为空，则返回 false。函数使用第一个参数 (geometry1) 的坐标系执行。

MI_Length

返回直线或折线几何体对象的地理长度。

语法

```
MI_Length ( geometry, unit, calculationType )
```

参数

geometry 是返回几何体的表达式，

unit 是线性单位规范，

calculationType 是字符串，指示是执行“球面”还是“笛卡儿”计算。没有默认值。

说明

返回值为双精度数，以 units 参数指定的单位表示。MI_Length 仅适用于 Curve、LineString 和 MultiCurve 类型的几何体。任何其它类型的输入几何体都导致零值。

关于单位的有效值请参阅第 381 页中的 *MI_Buffer*:

MI_MaxX

返回几何体的 MBR 的最大 x 值。

语法

```
MI_MaxX ( geometry )
```

参数

geometry 是返回几何体的表达式。

说明

返回值为双精度数，根据输入 geometry 的坐标系来表示值。

MI_MaxY

返回几何体的 MBR 的最大 y 值。

语法

```
MI_MaxY( geometry )
```

参数

geometry 是返回几何体的表达式。

说明

返回值为双精度数，根据输入 geometry 的坐标系来表示值。

MI_MinX

返回几何体的 MBR 的最小 x 值。

语法

```
MI_MinX ( geometry )
```

参数

geometry 是返回几何体的表达式。

说明

返回值为双精度数，根据输入 geometry 的坐标系来表示值。

MI_MinY

返回几何体的 MBR 的最小 y 值。

语法

```
MI_MinY ( geometry )
```

参数

geometry 是返回几何体的表达式。

说明

返回值为双精度数，根据输入 geometry 的坐标系来表示值。

MI_Perimeter

返回几何体对象的地理周长。

语法

```
MI_Perimeter ( geometry, unit, calculationType )
```

参数

geometry 是返回几何体的表达式，

unit 是线性单位规范，

calculationType 是字符串，指示是执行“球面”还是“笛卡儿”计算。没有默认值。

说明

返回值为双精度数，以 units 参数指定的单位表示。MI_Perimeter 仅适用于 Envelope、Polygon、Multipolygon 和 Ring 类型的几何体。任何其它类型的输入几何体都导致零值。

关于单位的有效值请参阅第 381 页中的 *MI_Buffer*:

MI_Point

创建新的 Feature Geometry 对象，表示一个点。

语法

```
MI_Point( x, y, csys )
```

参数

x 与 y 是双精度数，标识点的坐标，

csys 是坐标系规范。

说明

csys 参数可以是对边界坐标系对象的引用，或是使用 codespace:code 表示法的坐标系字符串表示。

MI_SQLVersion

指示引擎当前支持的 SQL 语言的版本。

语法

```
MI_SQLVersion
```

说明

当前的最高值为 1.0。

MI_Transform

将几何体从一个坐标系转换到另一个坐标系。

语法

```
MI_Transform ( geometry, csys )
```

参数

geometry 是返回几何体的表达式，

csys 是坐标系规范。

MI_Union

返回几何体对象，表示两个输入几何体对象的联合。

语法

```
MI_Union ( geometry1, geometry2 )
```

参数

geometry1 是返回几何体的表达式，

geometry2 是返回几何体的表达式。

MI_Within

确定一个 Geometry 对象是否完全在另一个 Geometry 对象之内。

语法

```
MI_Within ( testGeometry, containerGeometry )
```

参数

testGeometry 和 containerGeometry 是几何体对象（或表达式）。

说明

如果 testGeometry 完全在 containerGeometry 之内，则返回 true，否则返回 false。如果 containerGeometry 或 testGeometry 为空，则返回 false。函数使用第一个参数 (testGeometry) 的坐标系执行。

与 MI_Contains (containerGeometry, testGeometry) 功能相同。

MI_X

返回 Point 几何体的 x 坐标。

语法

```
MI_X ( point_geometry )
```

参数

point_geometry 是表达式，返回 Point 类型的图元几何体。

说明

返回值为双精度数，根据输入 point_geometry 的坐标系来表示值。如果输入几何体不是 Point 几何体，那么返回空。

MI_Y

返回 Point 几何体的 y 坐标。

语法

```
MI_Y ( point_geometry )
```

参数

point_geometry 是表达式，返回 Point 类型的图元几何体。

说明

返回值为双精度数，根据输入 point_geometry 的坐标系来表示值。如果输入几何体不是 Point 几何体，那么返回空。

Min

计算合计值的最小值。

语法

```
Min ( n )
```

参数

n 是数字表达式。

说明

Min 是仅可用于合计 Select 语句的合计函数。空值被忽略。如果仅在合计中提供空值，则返回空值。

Mod

取模。

语法

```
Mod ( d, v )
```

参数

d 为双精度数

v 为双精度数

说明

一个数字表达式除以另一个数字表达式，返回余数。注：MapBasic 将其视为运算符。

Month

以整数返回日期的月部分。

语法

```
Month ( date )
```

参数

date 是 Date 表达式。

说明

Month 返回 1 到 12 之间的整数值。如果 date 为空，则 Month 返回空。

NumberToDate

从数字创建 Date。

语法

```
NumberToDate ( n )
```

参数

n 是按 YYYYMMDD 表示日期的八位整数。

NumberToString

将数字转换为字符串。

语法

```
NumberToString ( n, fmt )
```

参数

n 是数字，

fmt 是标识如何格式化数字 (n) 的字符串。

说明

关于使用 fmt 参数格式化数字的详细信息，请参阅第 408 页中的数字格式化字符串部分。

ProportionAvg

计算合计值的平均值，该值基于两个几何对象所占比例进行调节。

语法

```
ProportionAvg ( n, value_geometry, reference_geometry )
```

参数

n 是数字表达式，

value_geometry 是几何体表达式，

reference_geometry 是几何体表达式。

说明

ProportionAvg 是仅可用于合计 Select 语句的合计函数。n 的空值、value_geometry、与 / 或 reference_geometry 被忽略。如果仅在合计中提供空值，则返回空值。

n 的值会在乘以一定比例之后累计和平均。比例由 value_geometry 和 value_geometry 分割的 reference_geometry 的重叠区域决定。

ProportionAvg 合计函数可以被替换为以下的公式：

```
Avg ( n * AreaOverlap/MI_Area )
```

AreaOverlap 不是已定义的函数。但是，它相当于以下表达式：

```
MI_Area(MI_Intersection(value_geometry, reference_geometry), 'sq mi',  
'Spherical')
```

如果 value_geometry 的坐标系为 Non-Earth，则将进行笛卡儿面积计算。

ProportionSum

计算合计值的和，该值基于两个几何对象所占比例进行调节。

语法

```
ProportionSum ( n, value_geometry, reference_geometry )
```

参数

n 是指示要添加的值的数字表达式,

value_geometry 是几何体表达式,

reference_geometry 是几何体表达式。

说明

ProportionSum 是仅可用于合计 Select 语句的合计函数。n 的空值、value_geometry、与 / 或 reference_geometry 被忽略。如果仅在合计中提供空值，则返回空值。

n 的值会在乘以一定比例之后累计。比例由 value_geometry 和 value_geometry 分割的 reference_geometry 的重叠区域决定。

例如，引用几何体可以是表示与多个郡相交的防汛地区。每个郡的人口可以基于郡面积重叠风险地区的比例而按比例合计。

ProportionSum 合计函数可以被替换为以下的公式:

```
Sum ( n * AreaOverlap/MI_Area )
```

AreaOverlap 不是已定义的函数。但是，它相当于以下表达式:

```
MI_Area(MI_Intersection(value_geometry, reference_geometry), 'sq mi',  
'Spherical')
```

如果 value_geometry 的坐标系为 Non-Earth，则将进行笛卡儿面积计算。

ProportionWtAvg

计算合计值的加权平均值，该值基于两个几何对象所占比例进行调节。

语法

```
ProportionWtAvg ( n, w, value_geometry, reference_geometry )
```

参数

n 是数字表达式，

w 是指示加权的数字表达式，

value_geometry 是几何体表达式，

reference_geometry 是几何体表达式。

说明

ProportionWtAvg 是仅可用于合计 Select 语句的合计函数。n 的空值、value_geometry、与 / 或 reference_geometry 被忽略。如果仅在合计中提供空值，则返回空值。如果分母导致零值（在积累的每个记录的加权或重叠面积是零的情况下发生），则将返回空值。

ProportionWtAvg 合计函数可以被替换为以下的公式：

```
Sum ( n * w * AreaOverlap ) / Sum ( w * AreaOverlap )
```

AreaOverlap 不是已定义的函数。但是，它相当于以下表达式：

```
MI_Area(MI_Intersection(value_geometry, reference_geometry), 'sq mi',  
'Spherical')
```

如果 value_geometry 的坐标系为 Non-Earth，则将进行笛卡儿面积计算。

Right

返回字符串中最右侧的指定个字符。

语法

```
Right ( str, len )
```

参数

str 是字符串，而

len 是大于或等于零的整数。

说明

Right 返回字符串，包含了字符串表达式 str 最右侧的 len 个字符。len 参数应该是大于或等于零的整数。如果 len 等于零，将返回空值。如果 len 大于或等于字符串 str 的长度，那么将返回字符串 str 的全部。

Round

将 X 取整到最接近 m 的倍数的值。参数 m 是可选的，如果缺少则取整到最接近的整数

语法

```
Round ( X, m )
```

参数

X 是指示要取整的值的双精度数，

m 是指示取整 X 的最接近值的双精度数。

Rtrim

从字符串中删除尾部的空字符，返回生成的字符串。

语法

```
Rtrim ( str )
```

参数

str 是字符串

Sign

如果参数为负值、零或正值，则返回参数的对应符号 -1、0 或 1。

语法

```
Sign ( d )
```

参数

d 为双精度数

Sin

计算正弦函数

语法

```
Sin ( d )
```

参数

d 为双精度数

Space

返回由指定数目的空字符组成的字符串。

语法

```
Space ( len )
```

参数

len 为整数。

说明

如果 len 小于或等于零，则返回空值。

Sqrt

计算平方根函数

语法

```
Sqrt ( d )
```

参数

d 为双精度数

StringToDate

从字符串创建 Date。

语法

```
StringToDate ( str, fmt )
```

参数

str 是表示日期的字符串，

fmt 是标识 str 格式的字符串。

说明

关于 fmt 参数含义的详细信息请参阅第 406 页中的日期格式化字符串部分。

StringToNumber

将字符串转换为数字。

语法

```
StringToNumber ( str, fmt )
```

参数

str 是表示数字的字符串，

fmt 是标识 str 格式的字符串。

说明

StringToNumber 将字符串值 (str) 转换到数字表示，使用字符串模板 (fmt) 解释数字字符串 (str)。关于 fmt 参数含义的详细信息请参阅第 408 页中的数字格式化字符串部分。

Substring

提取字符串的一部分来形成新的字符串。

语法

```
Substring ( str1, start[, length] )
```

参数

str1 为源字符串，

start 是字符串 str1 从零开始的索引，从该位置开始提取，

length 是指示提取字符数目的整数。

说明

length 参数是可选的，如果缺少则默认到 str1 的结尾。

Sum

计算合计值之和。

语法

```
Sum ( n )
```

参数

n 是数字表达式。

说明

Sum 是仅可用于合计 Select 语句的合计函数。空值被忽略。如果仅在合计中提供空值，则返回空值。

Tan

计算正切函数

语法

```
Tan ( d )
```

参数

d 为双精度数

Trim

从字符串中删除首部和尾部的空字符，返回生成的字符串。

语法

```
Trim ( str )
```

参数

str 是字符串

Trunc

按照 d 指定的小数位数截断参数 X。

语法

```
Trunc ( X, d )
```

参数

X 为双精度数，

d 为整数

说明

例如，Trunc(12.345, 1) 返回 12.3。d 值可为负值，此时指小数点左侧的位数。Trunc(12.345, -1) 返回 10。d 是可选的，如果缺少则假定为零。

Upper

将字符串转换到大写。

语法

```
Upper ( str )
```

参数

str 是字符串

Weekday

返回从 1 到 7 的整数，指示指定日期是周几。

语法

```
Weekday ( date )
```

参数

date 是 Date 表达式。

说明

Weekday 返回整数值，表示是周几，1 = 星期天、2 = 星期一，等等。如果 date 为空，则 Year 返回空。

WtAvg

计算值和加权合计的加权平均值。

语法

```
WtAvg ( n, w )
```

参数

n 是指示要平均的值的数字表达式，

w 是指示应用于 n 的加权的数字表达式。

说明

WtAvg 是仅可用于合计 Select 语句的合计函数。n 的空值与 / 或 w 被忽略。如果仅在合计中提供空值，则返回空值。如果分母导致零值（在积累的每个记录的加权是零的情况下发生），则将返回空值。

可由以下公式替代 WtAvg

$$\text{Sum} (n * w) / \text{Sum} (w)$$

Year

以整数返回日期的年部分。

语法

```
Year ( date )
```

参数

date 是 Date 表达式。

说明

Year 返回整数值，表示年份，包含世纪（例如 2003 年）。如果 date 为空，则 Year 返回空。

日期格式化字符串

格式化字符串用作模板，用于将字符串转换为日期，将日期转换为字符串。以下的标记在格式化字符串 (fmt) 中解释，用来从字符串中读取日期值部分 (StringToDate) 或以指定的日期部分值替代来生成字符串 (DateToString)。

标记	值
yyyy	4 位年（例如 2003）
yy	2 位年（例如 03）。
mmmm	月份的全名 - 例如 January
mmm	月份的缩写名 - 例如 Jan
mm	2 位月份（包括零起始，如果适用）。January = 1, December = 12。
m	1 或 2 位月份（非零起始）。January = 1, December = 12。
dddd	星期几的全名，例如 - Tuesday。
ddd	星期几的缩写名 - 例如 Tues。
dd	2 位月份日期（包括零起始，如果适用）。
d	1 或 2 位月份日期（非零起始）。

标记	值
hh	2 位小时， 12 小时制。包括零起始，如果适用。
H	1 或 2 位小时， 12 小时制，非零起始。
HH	2 位小时， 24 小时制。包括零起始，如果适用。
H	1 或 2 位小时， 24 小时制，非零起始。
nn	2 位分钟。包括零起始，如果适用。
n	1 或 2 位分钟，非零起始。
ss	2 位秒。包括零起始，如果适用。
s	1 或 2 位秒，非零起始。
T	AM 或 PM
t	am 或 pm

注：此表定义用来翻译时间值各部分的标记。不过，MapXtreme 2005 目前不支持时间。这些标记将不被支持并在翻译中保持原样（例如，作为静态文本）。

当调用 `DateToString` 时，格式化字符串 (fmt) 可以包含其它标记和字符，它们将按原样传递到结果字符串。

使用日期格式化字符串的示例

给定日期 `Tuesday, October 7, 2003` 作为参数 ‘date’ 的值，如下的 `DateToString` 调用生成所示的结果。

```

DateToString(date, 'mm-dd-yy')           : 10-07-03
DateToString(date, 'm-d-yyyy')          : 10-7-2003
DateToString(date, 'dddd, mmmm d, yyy') : Tuesday, October 7, 2003
DateToString(date, 'Today is dddd')      : Today is Tuesday

```

如下的 `StringToDate` 调用都生成日期值 `October 7, 2003`。

```

StringToDate('10/7/2003', 'm/d/yyyy')
StringToDate('10-07-03', 'm-dd-yy')
StringToDate('Today is 2003-Oct-07', 'Today is yyyy-mmm-dd')

```

数字格式化字符串

格式化字符串用作模板，用于将字符串转换为数字，将数字转换为字符串。下面的元素可以组合为格式化字符串 (fmt)，用于从数字生成字符串 (NumberToString) 或者从字符串生成数字 (StringToNumber)。

元素	值
9	每个 9 都表示一个要显示的有意义的数字。零起始会显示或被视为空格。
0	表示一个有意义的数字。零起始会显示或被视为零。
\$	前缀：在数字前面加美元符号 (\$)。其它货币符号请参阅 L 元素。
, (逗号)	表示数字字符串中的逗号。逗号始终解释为组分隔符。其它组分隔符请参阅 G 元素。
. (句号)	表示数字字符串中的小数点。其它小数点符号请参阅 D 元素。
MI	后缀：如果为负则在数字后面加负号；为正则加空格。
S	前缀或后缀：在正数前面加正号 (+)，负数前面加负号 (-)。
D	指定小数点的位置。D 左侧所有的格式化元素都会格式化值的整数部分。D 右侧所有的格式化元素都会格式化值的小数部分。用于表示十进制字符的字符决定于特定的平台 - 例如 MapXtreme 2005 中这个值由当前线程的 Culture 设置决定。对于 Spatialware，它由数据库确定（通常为初始化参数）。
G	指定组分隔符的位置（例如 6,234 中逗号来按千分隔）。如同十进制字符 (D)，用于表示组字符的字符决定于特定的平台。
L	指定本地货币符号的位置（例如 \$）。如同十进制字符 (D)，用于表示组字符的字符决定于特定的平台。

使用数字格式化字符串的示例

下例说明了使用数字格式化元素来格式化数字或分析字符串。结果显示时带有引号，指示在结果中的空格（空位）的出现 - 引号不是结果的一部分。

```
NumberToString(123.4567, '999999.99')      : '123.46'  
NumberToString(123.4567, '099999.99')      : '000123.46'  
NumberToString(1234567, '99,999,999')       : ' 1,234,567'  
NumberToString(1234567, '9,999,999,999')     : '1,234,567'  
NumberToString(1234567, '999,999,999')       : ' 0,001,234,567'  
NumberToString(123.4567, '0.9999E99')        : '1.2346E2'  
NumberToString(123.4567, '0.9999E09')        : '1.2346E02'  
NumberToString(123.4567, '0.9999E+09')       : '1.2346E+02'  
NumberToString(12.345, '9.99')                : '####'  
StringToNumber(' 123.456', '9999.999')       : 123.456  
StringToNumber(' 123.456-', '9999.999')       : Error  
StringToNumber(' 123.456-', '9999.999MI')    : -123.456  
StringToNumber(' 1,234.56', '999,999.99')     : 1234.56  
StringToNumber(' 1,234.56', '999,999.9')      : Error
```


C

迁移到 MapXtreme 2005

本章节适用于 MapX 的现有用户，他们熟悉 MapXtreme 2005 中基于 .NET 的对象模型以及与 MapX 体系结构的不同之处。

本附录内容：

- ◆ 比较 *MapXtreme 2005* 和 *MapX* 的对象模型..... 412

比较 MapXtreme 2005 和 MapX 的对象模型

若已使用过 MapX 和 MapXtreme for Windows，将会发现它们与 MapXtreme 2005 工作方式的相似和不同之处。本节描述了 MapXtreme 2005 和先前 API 版本之间的一些主要不同之处。

特定对象模型实现的不同之处

源于 MapX 的 Map 对象现在用 3 或 4 个类表示。Map 包含了图层、重点区域（视图）、Adornment 和 Legend（无论以何种方式查看 Map）。MapControl 保存 Map 对象并作为 Map 的控件和与 Map 交互的工具。此外还具有既可用于 Windows Form（桌面）应用程序又可用于 ASP.NET (web) 应用程序的 MapControl。MapExport 类可用于将 Map 导出到文件或数据流。Session 保存 Maps、Selections 和 Tables 的集合。现在完全支持制图比例。标注图层可用于更好地控制标注的位置。组图层允许将大量图层在图层层次中视为一个图层。

MapX Dataset 概念在 MapInfo.Data 命名空间中已由一组更为灵活的选项替换。MapXtreme 2005 使用 ADO.NET 进行数据访问。ADO.NET 可以使用非常类似的代码访问各种数据格式，简化了数据源之间的转换。增加了更多光标用于通过数据库表向前和向后访问。

MapXtreme 2005 使用以表为中心的数据模型。在以前版本的 API 中，地图由图层组成，但没有特定的表类。在 MapXtreme 2005 对象模型中，表是用于访问数据的中心对象。Session.Catalog 用于打开和枚举表。图层可以引用表，但所有表特定的方法从 Layer 移动到到表。另外，搜索在 MapXtreme 2005 中的表上而不是在图层上执行。

修改样式的主题不再使用其自身的图层，例如范围主题和单值主题。在以前的模型中这种主题占用独立的图层，而且使用时独立于地图的其它图层。对象主题仍然保留其自身的图层且现在可以用选择的任何方式来显示，独立于主题所基于的原始地图对象。

几何体已更改为层次模型，可以更深入地访问对象的属性。样式也以层次表示。

此外还增加了大量地图工具供您使用。增强了这些工具的可扩展性和功能性，使得开发定制的工具更加容易。还有用于 web 应用程序（也可扩展）中的工具。

对于 web 应用程序，不再需要 MapXBroker 和 MapXServer。相反，MapXtreme 2005 使用标准的 COM+ 将 Session 对象置入对象池。

应用程序状态通过串行化和 ASP.NET 状态服务器来管理。有关状态管理和串行化的详细信息请参阅第 63 页第 5 章中的对象池应用程序开发模型。

下表列出了 MapX 5.0 对象模型中的对象和 MapXtreme 2005 当前对象模型中的等价对象。请注意，由于大范围重新设计了 MapXtreme 2005 产品，因此等价的对象可能是近似的。

MapX 5.0	MapXtreme 2004/2005
AffineTransform	AffineTransform 它是不可变的（无设置）。使用 CoordSysFactory.CreateAffineTransform 来创建。
AllFeaturesConstraint	
Annotation	已过时。新的 Adornments 类。MapControl 包含从 Control 中派生的修饰
Annotations	已过时。
BindLayer	已过时。替换对象位于 Table.AddColumn() 参数中。
BitmapSymbol	BitmapPointStyle
BitmapSymbols	
BoundsConstraint	
CoordSys	CoordSys。 它是不可变的（无设置）。使用 CoordSysFactory 创建。
Dataset	已过时。 替换对象： <ul style="list-style-type: none">• 表达式列：Table.AddColumn() 可以创建表达式列。• 手动数据绑定（动态或静态）：Table.AddColumn() 使用以 TableInfoClient 打开的 Table，指向 ADO.NET DataTable。• 自动数据绑定：由 Geodictionary 功能处理。• 非 ADO.NET 外部数据• XY 绑定：SpatialSchemaXY 应用到 TableInfo.SpatialSchema 属性。• PointRef 绑定：SpatialSchemaPointRef 应用到 TableInfo.SpatialSchema 属性。• RowValues：MIDataReader，MIScrollableReader，请参阅 M ICommand。• Refresh：Table.RefreshColumns()
Datasets	已过时。请参阅 Dataset。
Datum	Datum 它是不可变的（无设置）。使用 CoordSysFactory.CreateDatum() 来创建。
Feature	Geometry，样式和键信息是 Feature。
Features	请参阅 Feature。
FeaturesConstraint	

MapX 5.0	MapXtreme 2004/2005
FeatureFactory	<p>表级别对象处理</p> <ul style="list-style-type: none"> • Buffer: 对于每个图元，缓冲区都位于 Geometry 级别 (FeatureGeometry.Buffer)。 对于多个图元，请使用 FeatureProcessor.Buffer。 • Combine: 对于每个图元，组合都位于 Geometry 级别 (FeatureGeometry.Combine)。 对于多个图元，请使用 FeatureProcessor.Combine。 • Intersect: 对于每个图元，相交都位于 Geometry 级别 (FeatureGeometry.Intersect)。 对于多个图元，请使用 FeatureProcessor.Intersect。
Field	
Fields	
Find	Find
FindFeature	FindResult
FindMatch	FindCloseMatch, FindAddressRange
FindMatches	FindCloseMatchEnumerator, FindAddressRangeEnumerator
FindResult	FindResult
Geoset	已过时。类似功能: MapLoader
Geosets	已过时。MapXtreme 2005 会打开 geosets
Graphic	已过时。
IndividualValueCategory	IndividualValueThemeBin
IndividualValueCategories	ModifierThemeBins
Label	LabelLayer, LabelSource
Labels	LabelLayer, LabelSource
LabelProperties	LabelProperties

MapX 5.0	MapXtreme 2004/2005
Layer	<p>UserDrawLayer、LabelLayer、FeatureLayer、ObjectThemeLayer、GroupLayer...</p> <ul style="list-style-type: none">• AddFeature: 在使用 MICommand 执行插入操作期间, 当前的等价对象要将 Geometry 绑定到几何体列。• AllFeatures• AutoLabel: 不再存在。等价对象将 LabelSource.DefaultLabelProperties.Enabled 属性设置为 true。• Begin/EndAccess: 未记录• Bounds: FeatureLayer.Bounds• ClearCustomLabels: 未记录• ClippedBounds: 已过时。使用 Map.SetView()• CoordSys• Datasets: 已过时。图层具有 Table (FeatureLayer.Table.)• DeleteFeature: 等价对象是通过 MICommand 删除行。• DrawLabelsAfter: 此概念已很少使用。类似的功能是在每个 Layer 后面都使用 LabelLayer 实例。唯一的缺点是每个 LabelLayer 都清理其自己的标注缓存, 因而标注将不会在图层之间交互。• Editable: FeatureLayer.Editable• FeatureIDFromFeatureName: 等价对象将是通过使用 MICommand 的 SQL 命令。

MapX 5.0	MapXtreme 2004/2005
Layer (续)	<ul style="list-style-type: none"> • Find: Find • GetDrillDownFeaturesByID: 未记录 • GetFeatureByID: 等价功能是使用 MICommand 的键光标。 • Invalidate: IMapLayer.Invalidate • KeyField: 已过时。等价功能是 MICommand。 • LabelAtPoint • LabelProperties: LabelSource.DefaultLabelProperties. • Labels • Name: FeatureLayer.Name 或 Alias。 • NoFeatures • OverrideStyle: 等价对象是 FeatureLayer.Modifiers 集合。 • Pack: 等价对象是 FeatureLayer.Table.Pack • PredominantFeatureType: MISpatialColumnInfo.PredominantObjectType. 可以从 Table.TableInfo.ColumnInfos 属性获取列。 • Refresh: 刷新缓存。等价功能是: Table.Refresh() • Search: 等价功能是 MICommand。但是, 结果可能不仅仅包含图元键。 • SearchAtPoint: 无“简单的”等价对象。 • SearchWithinDistance: 无“简单的”等价对象。 • SearchWithinRectangle: 无“简单的”等价对象。 • SearchWithinFeature: 无“简单的”等价对象。 • Selectable: FeatureLayer.Selectable. • Selection: Selection 类, 位于 Session.Sessions 集合中。 • ShowCentroids: FeatureLayer.ShowCentroids • ShowLineDirection: FeatureLayer.ShowLineDirection • ShowNodes: FeatureLayer.ShowNodes • Style: 等价对象目前是 FeatureLayer.Modifiers 集合。 • SupportsPack: Table.SupportsPack • Type: IMapLayer.Type • UpdateFeature: 通过使用 MICommand 的等价对象 • Visible: IMapLayer.Enabled. IMapLayer.Visible 考虑: • ZoomLayer: IMapLayer.ZoomRangeEnabled • ZoomMax: IMapLayer.ZoomRange.End • ZoomMin: IMapLayer.ZoomRange.Start
Layers	图层 <ul style="list-style-type: none"> • AnimationLayer 等价对象: GroupLayer 和 BackingStoreState.Off • InsertionLayer 等价对象: AddMapTool.InsertionLayer • Selections: 位于 Session 对象中。
LayerInfo	已过时。等价对象是 TableInfo
Legend	Legend

MapX 5.0	MapXtreme 2004/2005
LegendText	LegendRow
LegendTexts	LegendRows
Map	<p>目前，Map 已分成两个实体：Map 和 MapControl。</p> <p>MapControl 包含适用于 System.Windows.Forms.Control 的功能；它是可嵌入 WinForm 的对象。</p> <p>Map 无论以何种方式查看（不管它是 WinForm 还是导出），均包含实际的地图绘制功能。</p> <p>下面是先前 Map 类中的属性 / 方法列表以及如何在 MapXtreme 2005 模型中进行处理：</p> <ul style="list-style-type: none"> • Annotations: 已过时。 • CenterX/Y: Map.Center • ConvertCoord: DisplayTransform.ToDisplay(), DisplayTransform.FromDisplay() • CreateCustomTool: 从 MapTool 或任何可用的 Custom* 工具继承，它们通过使用鼠标提供选取线绘制功能。 • CurrentTool: MapTools.LeftButtonTool、MapTools.MiddleButtonTool、MapTools.RightButtonTool。可以通过 MapControl.MapTools 属性访问 MapTools。 • Datasets: 已过时。可通过 MICatalog 访问表。 • Dataset: 已过时。 • DatasetGeoField: 已过时。请参阅 Map.Dataset. • DatasetTheme: 已过时。请参阅 Map.Dataset. • DefaultStyle • DisplayCoordSys: Map.GetDisplayCoordSys(). • Distance: CoordSys.Distance(). • DynamicSelectionSupport: MapTools.DynamicSelectionModeSelectMapTool.DynamicSelectionEnabled • EditableLabels • ExportMap: MapExport 类可以使用多种选项来导出地图。 • ExportSelection: MapExport.ExportSelection • FeatureEditMode 枚举: • miEditModeFeature -> MapTools.NodeMode = false • miEditModeNode -> MapTools.NodeMode = true • miMoveDuplicateNodes -> MapTools.MoveDuplicateNodes • miDeleteDuplicateNodes -> MapTools.DeleteDuplicateNodes • miEditModeAddNode -> MapTools.AddNodeMode • FeatureFactory: FeatureProcessor • GeoDictionary: Geodictionary • Geoset: 已过时。

MapX 5.0	MapXtreme 2004/2005
Map (续)	<ul style="list-style-type: none">• Geosets: 已过时。• GeosetWidth• hWnd• InfotipPopupDelay• InfotipSupport: 未记录。• IsPointVisible• Layers: Layers• MapPaperHeight/Width: PaperSize (通过 MapExport.ExportSize)• MapScreenHeight/Width: MapControl.Size, Map.Size• MapUnit: 显示坐标系的 CoordSys.Units• MatchNumericFields: GeoDictionary 功能。• MatchThreshold: GeoDictionary 功能。• MaxSearchTime: GeoDictionary 功能。• MilitaryGridReferenceToPoint: CoordSys.MilitaryGridToPoint()• MilitaryGridReferenceFromPoint: CoordSys.PointToMilitaryGrid()• MouseIcon: MapControl.Cursor• MousePointer: MapControl.Cursor• MouseWheelSupport: MapControl.MouseWheelSupport.MouseWheelBehavior• NumericCoordSys: 已过时。几何体现在有其自己的 CoordSys。具有坐标的任何内容都有坐标系。• Pan• PanAnimationLayer• PreferCompactLegends• PrintMap: Map.Draw()• PropertyPage• RedrawInterval: Map.IncrementalDraw.Interval.• Refresh• ReuseEquivalentOnRestore• Rotation: Map.Rotation.• SaveMapAsGeoset• SearchPath.• SelectionStyle: Selection.Style• SetSize• SnapToNodeSupport• SnapTolerance: MapTools.SnapTolerance• Title: Adornments• TitleText• Version• WaitCursorEnabled• Zoom• ZoomTo

MapX 5.0	MapXtreme 2004/2005
MultivarCategory	MultiVariableThemeCategory
MultivarCategories	MultiVariableThemeCategories
NotesQueryInfo	已过时。
NotesViewInfo	已过时。
OCIQueryInfo	已过时。
ODBCQueryInfo	已过时。
Parts	等价
点	DPoint
Points	已过时
RangeCategory	RangedThemeBin
RangeCategories	ModifierThemeBins
Rectangle	DRect
ResolveObject	MatchResolver
ResolveObjects	MatchResolver 集合。
RowValue	MICommand
RowValues	MICommand
Selection	Selection
SourceRow	SourceRow
SourceRows	SourceRows 集合
State	
Style	
Theme	ObjectTheme, FeatureStyleModifier
Themes	已过时。Themes 目前由 ObjectThemeLayer (对于 ObjectThemes) 包含或包含在 Layer 的 Modifiers 集合中 (对于 FeatureStyleModifier 主题图)。
ThemeProperties	已过时。Properties 位于主题图类本身内。

MapX 5.0	MapXtreme 2004/2005
Title	Adornments
Variable	MIPParameter。由于在表达式中绑定，MICommand.Parameters 属性可以定义在该命令中使用的变量。
Variable	MIPParameterCollection。

D

如何创建和部署 MapXtreme 2005 应用 程序

本附录是设计用于演示如何简化 MapXtreme 2005 应用程序的创建、打包和部署的教程。在本教程结尾处，您将查看到成功开发、打包和部署已实现地图绘制应用程序的所有步骤。

本教程假定您已成功安装 Visual Studio .NET 并熟悉该产品的使用。如果不熟悉 Visual Studio .NET，则可以从 Microsoft MSDN web 页面获得相关资源。详情请查看 <http://msdn.microsoft.com/developerscenter/>。此外，还应完全安装 MapXtreme 2005。本文档包含的代码片断通过 Visual Basic 实现。对于多数示例应用程序，我们还提供了 C# .NET 示例。

本附录内容：

- ◆ 构建桌面应用程序 422
- ◆ 生成 Web 应用程序 429
- ◆ 将应用程序打包 431
- ◆ 部署应用程序 436
- ◆ 授权分布式应用程序 437

创建应用程序

第一步，创建应用程序。本节首先从创建桌面应用程序的角度，再从创建 web 应用程序的角度来说明该处理的过程。

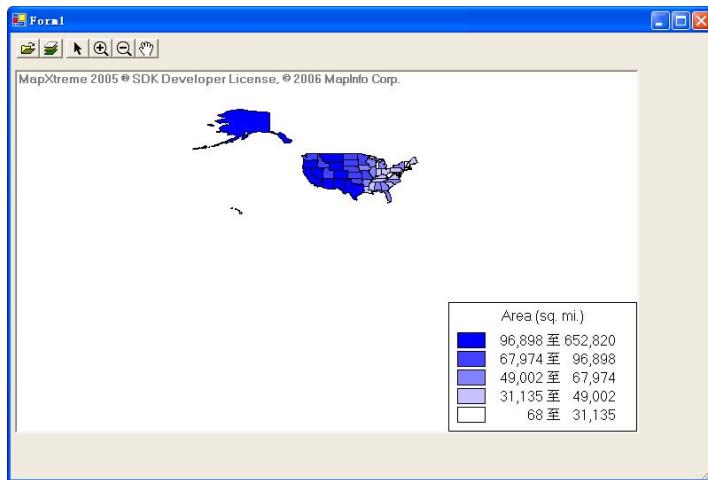
构建桌面应用程序

1. 选择模板

对于该示例，我们要选择 MapXtreme 2005 提供的其中一个主题图示例应用程序。

MapXtreme 2005 提供的所有示例应用程序都通过该产品提供的地图应用程序模板创建。使用 Visual Studio .NET，打开 ThemeLegend 示例应用程序的项目。该文件位于 C:\Program Files\MapInfo\MapXtreme\6.x\Samples\Desktop\Features\ThemeLegend\vb\ThemeLegendVB.vbproj（其中 x 代表正使用的 MapXtreme 2005 的发行版本号）。

项目打开后，请选择**调试 > 开始**或按下**F5**执行调试生成。



该应用程序创建的地图为标准美国地图，其主题图基于每个州的陆地面积。较大州显示为蓝色，较小州则显示为白色。图例显示细目分类。由于默认为显示整个地图，因此缩放级别用处不大。

使用提供的工具放大特定区域，或根据选择缩小。通过这些工具还可以选择区域并使用图层控件操控图层的显示方式。

如您所见，构建应用程序是非常简单的。

示例应用程序不足且需要更多定制信息时会发生什么情况？下文详细说明了可对该应用程序执行的某些修改，作为用于定制任意 MapXtreme 2005 项目的模型。

2. 修改应用程序

让我们从窗体后面的代码开始。使用 Visual Studio .NET 时，右键单击窗体任意位置并选择 **查看代码** 以显示代码页。

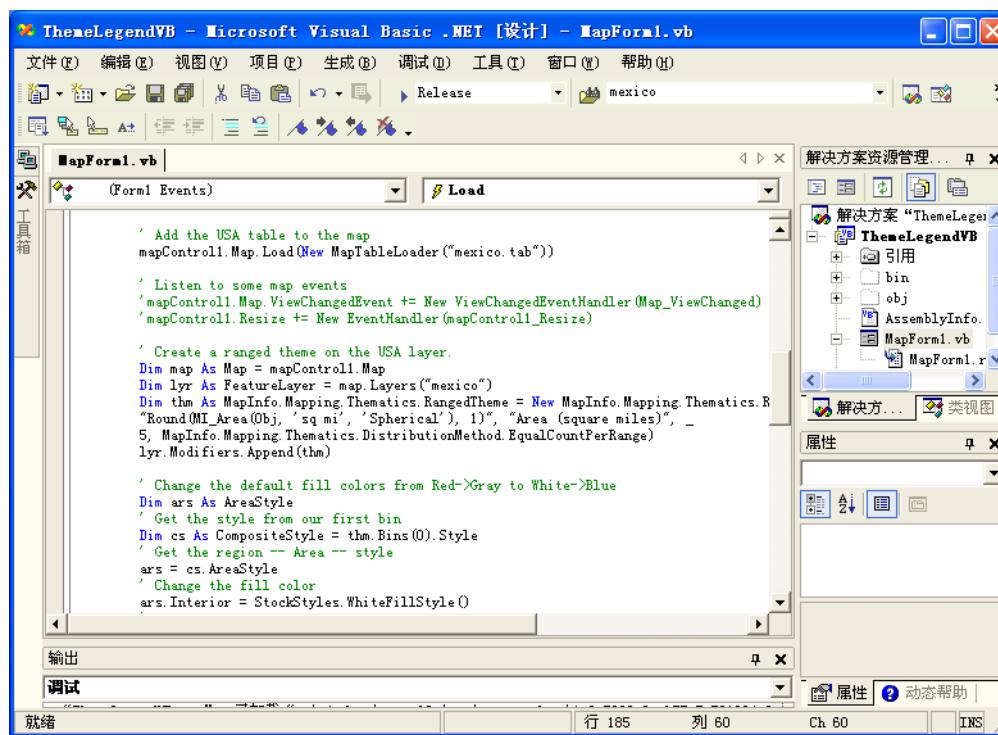
注： 本教程中讨论的所有示例应用程序均来自 MapXtreme 2005 的 6.5 发行版光盘。引用的行号可能与安装中的代码不完全对应。

向下滚动代码页时会发现应用程序中的主要操作会在 Form1() 类中执行。首先，我们要更改的是应用程序用于在其中创建主题图的数据。

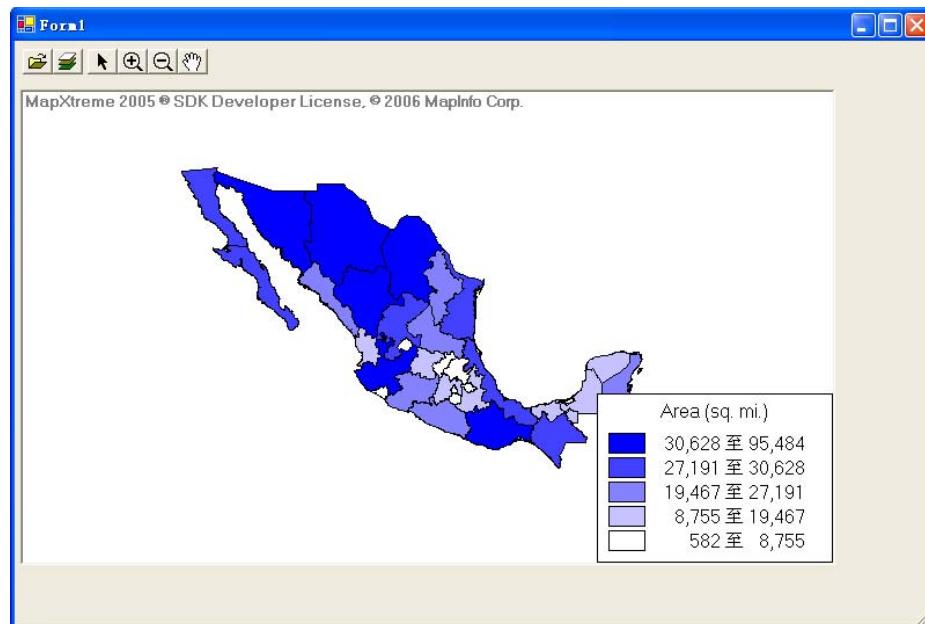
需要执行以下更改：

- 在第 185 行，将 usa.tab 更改为 mexico.tab
这将加载墨西哥地图，而不是美国地图。
- 在第 193 行，将 usa 更改为 mexico
这将更改正在使用的 FeatureLayer。

请参考下图，了解更改代码的正确位置。



构建并运行该版本的应用程序，查看发生的更改。



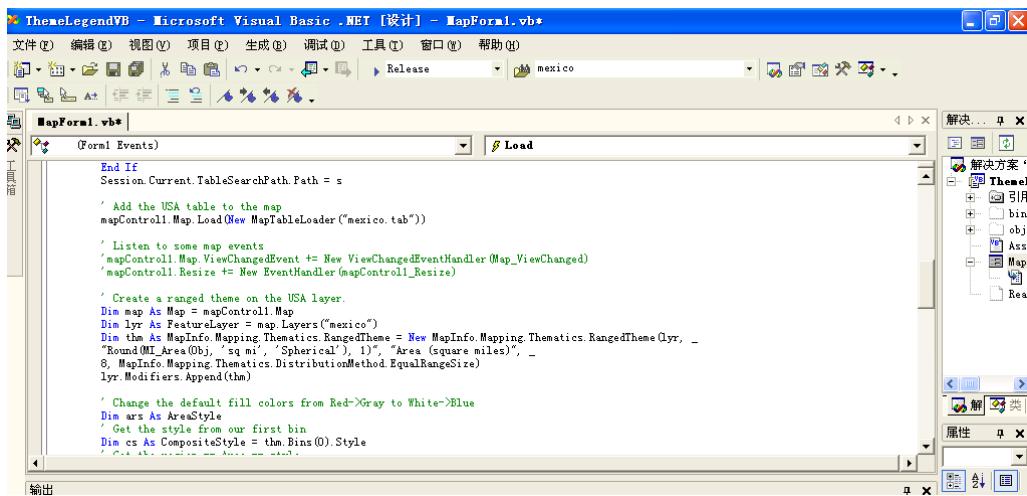
在地图中使用不同数据源时会出现一个明显差异。可以通过更改上述两行，传递适合该主题图的任何数据源。

接下来更改主题图的显示方式。通过控制传递到 RangedTheme 构造函数的参数，我们可以更改主题图的计算和显示方式。将光标置于创建主题图的代码所在位置（第 194 行）并按下 F1。这时将显示此调用的帮助主题，您可以通过阅读该主题了解每个参数传递到构造函数时执行的操作。

需要执行以下更改：

- 在第 195 行，将“5”更改为“8”
这将更改主题图中容器的数目。（容器是数据的特定部分，与给定参数相匹配。
通过更改该数目，则会将数据分割为更多“类别”。）
- 在第 196 行，将“EqualCountPerRange”更改为“EqualRangeSize”。
这将更改分布方法，以使得每个范围覆盖相似数量的值。

请参考下图，了解更改代码的位置。



按 **F5**（或 **调试 > 开始**）生成并运行修改的应用程序。如您所见，主题图容器的数目已增加且分布已更改。

现在，要更改每个显示范围的颜色。可以通过设置使每个容器具有特定颜色，或写入代码使颜色范围从一种颜色变换至另一种颜色，需要做的只是指定第一和最后一种颜色。此代码中的颜色范围是从白色至蓝色。现在让我们将颜色范围更改为从蓝色至红色。

需要执行以下更改：

- 在第 206 行，将 `WhiteFillStyle` 更改为 `BlueFillStyle`
这将更改第一个容器的颜色。
- 在第 216 行，将 `BlueFillStyle` 更改为 `RedFillStyle`
这将更改最后一个容器的颜色。每个位于这两个容器中间的容器都将具有在这两种颜色之间过渡的底纹。

请参考上图，它将指示更改代码的位置。

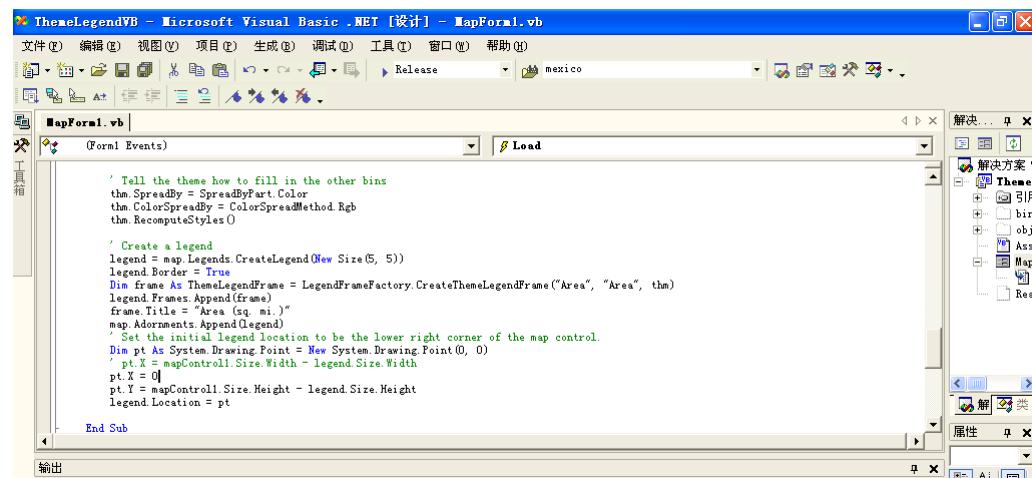
按 **F5**（或 **调试 > 开始**）生成并运行修改的应用程序。如您所见，地图颜色的更改反映了新的设置。

由于对地图的显示方式进行了多项修改，因此需要移动图例且不作任何改动，这样就会阻止部分地图的查看。可以使用平移工具移动地图，但更有效的方法是使用程序来移动图例。

需要执行以下更改：

- 在第 233 行，将 `"mapControl1.Size.Width - legend.Size.Width"` 更改为 `"0"`
这将图例位置的 X 坐标更改到框架的左侧。而 Y 坐标保持不变。

请参考下图，它将指示更改代码的位置。

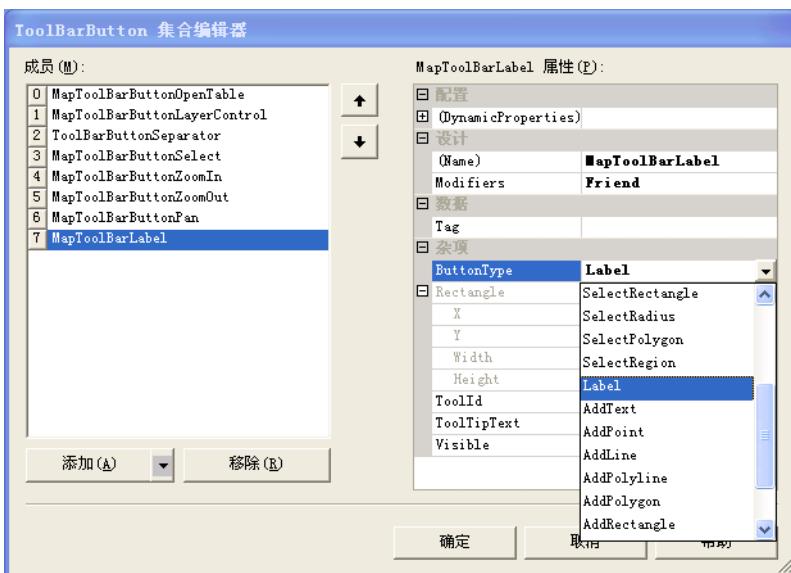


再次生成并运行应用程序，以查看更新的地图。此时，图例位于新的位置。

现在执行另一项修改。此次将 Label 工具增加到接口中的 ToolBar。在 Visual Studio .NET 中，单击 **MAPFORM1.VB [设计]** 选项卡，将“设计”页面显示到前端。在“属性”窗口的“按钮”字段中，单击集合后的省略号 (...) 按钮。



“ToolBarButton 集合”对话框显示。单击**添加**按钮，将新按钮添加到 ToolBar。该按钮在列表中最后一个按钮下显示。在**(NAME)**字段中重命名新按钮 mapToolBarLabel。从**BUTTONTYPE**下拉菜单中选择，将按钮类型更改为**标注**。



单击**确定**保存更改并关闭“ToolBarButton 集合”对话框。此时能够看见增加到 ToolBar 的新工具。

如您所见，通过修改现有的应用程序可以轻松地达到所需结果。可对该主题图地图执行的其它修改包括：

- 更改主题图基于的数据源列。
例如，通过检查表结构，可以选择人口。不要忘记更新要匹配的图例。
- 增加其它主题图和图例。
- 增加其它工具和按钮。

3. 在“发行”模式下生成

在修改应用程序后，可以准备执行最后的生成。将生成类型更改为“发行”（**生成 > 配置管理器**并从“可用解决方案配置”下拉列表中选择“发行”），然后重新生成。

生成 Web 应用程序

1. 选择模板

对于该示例，我们要选择 MapXtreme 2005 提供的 Web Thematics 示例应用程序。

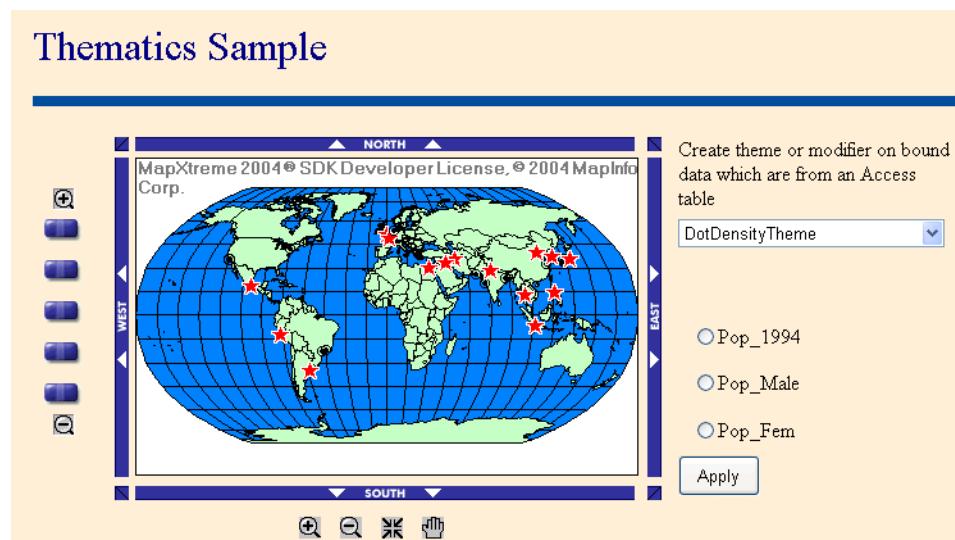
注： 对于 Visual Basic 示例应用程序，必须为 VB 版示例应用程序创建 IIS 虚拟目录。C# 版示例应用程序安装后即可运行，无需执行其它步骤。

要创建支持 VB 示例应用程序的 IIS 虚拟目录，请遵循以下步骤：

1. 选择**开始 > 控制面板 > 管理工具 > 计算机管理**，打开“计算机管理”对话框。
2. 选择 ThematicsWeb_6_5 web 应用程序，然后右键单击访问上下文菜单。在属性对话框中更改位置路径值，使其指向 Visual Basic 版示例的 “vb” 子目录。
3. 重新启动 IIS。在“计算机管理”对话框中选择 Internet 信息服务，从上下文菜单中选择**所有任务**，然后**重新启动 IIS...**

此时，就可以根据下述指示运行 VB 版示例应用程序。

从 ThematicsWeb 示例应用程序中打开项目文件（位于 C:\Program Files\MapInfo\MapXtreme\6.5\Samples\Web\Features\ThematicsWeb\vb\ThematicsWeb.vbproj）。按**F5**（或**调试 > 开始**），运行调试生成。



显示的 web 页面中包含一张世界地图、一个列出各种要应用主题图的下拉菜单、MS Access 表的属性列，以及应用这些主题图的“Apply”按钮。在底部中间位置，显示 MapXtreme 2005 web 模板中包括的标准工具按钮：ZoomIn、ZoomOut、Center 和 Pan。该示例中每个主题图都基于绑定的列，根本不使用任何列的 FeatureOverrideStyleModifier 除外。MS Access 数据库 eworld mdb 中绑定的列与 world.tab 绑定。

如您所见，创建高级的 web 应用程序十分简单。有难度的是在牢记最佳实践的基础上设计具有状态管理的 web 应用程序。由于在 web 环境下处理应用程序和用户状态的复杂性，所以必须了解 MapXtreme 2005 是如何处理它们以及如何在自己的开发中应用这些概念。有关对 web 应用程序“状态管理”的讨论，请参阅第 60 页第 5 章中的规划 Web 应用程序。

最佳实践中的这个示例，使用预载的地图工作空间和设置，手动管理应用程序和用户状态并使用池会话对象，使应用程序能够有效地服务于多项请求。

它设计用于检测应用程序用户为首次发出请求（在这种情况下应用程序显示其初始状态）还是再次访问该应用程序，同时预期该用户要维护的个人设置。由于本示例与多用户共享会话对象（称作对象池），因此它会检测会话时位于其初始状态（“干净的”）还是包含其它用户的更改（“不干净的”）。为了分别维护每个用户的设置，该示例使用 Session.State 并将其设置为手动，而不使用 HttpSessionState，因为 HttpSessionState 虽然可以自动保存状态，但同时会降低性能，因为它无法区分要保存哪些状态。它将保存所有状态。

关于该示例后面的代码的分析，请参阅附录 E：案例研究：在 Web 应用程序中管理会话状态。

2. 修改应用程序

有关如何修改该示例应用程序的详细信息，请在“桌面应用程序”讨论中参考第 423 页中的 2. 修改应用程序。可以通过与“桌面应用程序”类似的方式更改数据、范围颜色、分发方法等。

3. 在发行模式下生成

将生成选项更改为“发行”（**生成 > 配置管理器** 并从“可用解决方案配置”下拉列表中选择“发行”），然后重新生成整个项目。只有处于“发行”模式下才能将应用程序打包，以下部分将进行详细说明。

将应用程序打包

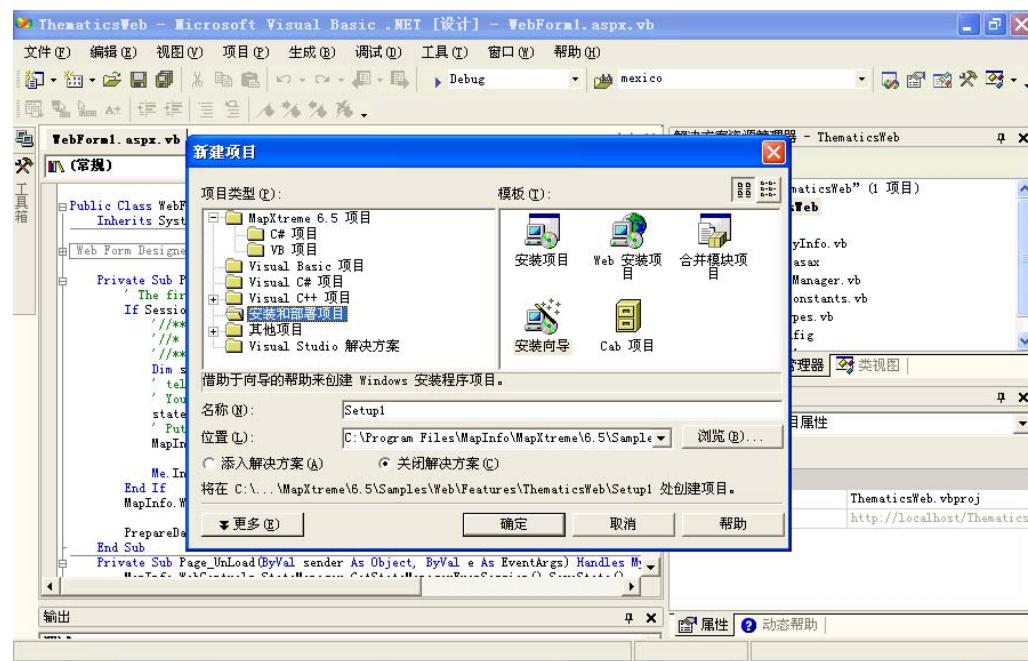
桌面应用程序的打包

设计、编码和生成桌面应用程序后，就需要将其打包发送给用户。MapXtreme 2005 在设计上可以简化该过程。使用 Visual Studio .NET 中的图元及 MapXtreme 2005 中的自动操作，可将正确的合并模块包括在程序包中。合并模块 (MSM) 是一个独立的程序包，包含了安装组件必需的所有文件、资源、注册表项和安装逻辑。在第 34 页中的部署应用程序中可以查找到 MSM 列表以及 MapXtreme 2005 中运行时安装程序和其它可用选项的相关讨论。

1. 创建安装项目

要完成打包请遵循以下步骤：

1. 在解决方案中创建新的安装项目。
 - a. 在“解决方案浏览器”中，突出显示“解决方案”并选择**文件 > 增加项目 > 新建项目 ...**
 - b. 从“安装和部署项目”中，选择**安装向导**，该向导将指导您完成项目创建过程。



- c. 按照屏幕提示操作。
2. 在步骤 2/5 中，选择**创建 WINDOWS 应用程序的安装**。单击**下一步**。

3. 在步骤 3/5 中，选择要包括在安装文件中的项目输出。
选择增加“主要”输出。单击**下一步**。
4. 在步骤 4/5 中，增加包含数据的文件。单击**增加**按钮并选择要增加到项目的文件。在这种情况下，需要增加以下位于

C:\Program Files\MapInfo\MapXtreme\6.5\Samples\Data 的文件：

- MEXICO.DAT
- MEXICO.ID
- MEXICO.IND
- MEXICO.MAP
- mexico.TAB

这些文件的.mws 将在当前应用程序目录下生成。

注：如果要将该应用程序部署到另一台计算机，则开发计算机和部署计算机上数据的绝对路径必须一致。

单击“下一步”按钮查看操作摘要。

5. 单击**完成**创建安装应用程序。

然后，Visual Studio .NET 将显示新安装项目的“文件系统”。

2. 增加许可证文件

为了使客户能够查看到您的应用程序，则需要在应用程序中安装桌面许可证文件(MapXtremeDesktop.lic)。请参阅第 437 页中的授权分布式应用程序。在该教程中包含了试用许可证¹的副本，以便您查看相关步骤。

1. 从 Visual Studio .NET 界面，激活**文件系统（安装）**选项卡。
2. 单击“应用程序文件夹”图标。
3. 右键单击并增加文件。在这种情况下，要选择名为 MapXtremeTrial.lic 的文件（位于 c:\Program Files\Common Files\MapInfo\MapXtreme\6.5）。在文件选取器中选择 MapXtremeTrial.lic 并单击“打开”。
此时，文件增加到安装项目并将与其它应用程序文件共同安装到应用程序文件夹。

3. 可选步骤：增加 Workspace 管理器和 GeoDictionary 管理器

如果需要，可在程序包中包括 MapXtreme 2005 实用程序 Workspace 管理器和 GeoDictionary 管理器，以便客户能够使用它们。执行第 432 页中的 2. 增加许可证文件中的步骤，增加 <MapXtreme 2005 installation folder>\WorkspaceManager.exe 和 GeoDictionaryManager.exe。

1. 如需为部署创建程序包，则 MapXtremeTrial.lic 必须来自 SDK 或 MapXtreme 2005 的运行时安装。使用 MapXtreme 2005 的评估版本无法创建安装项目。

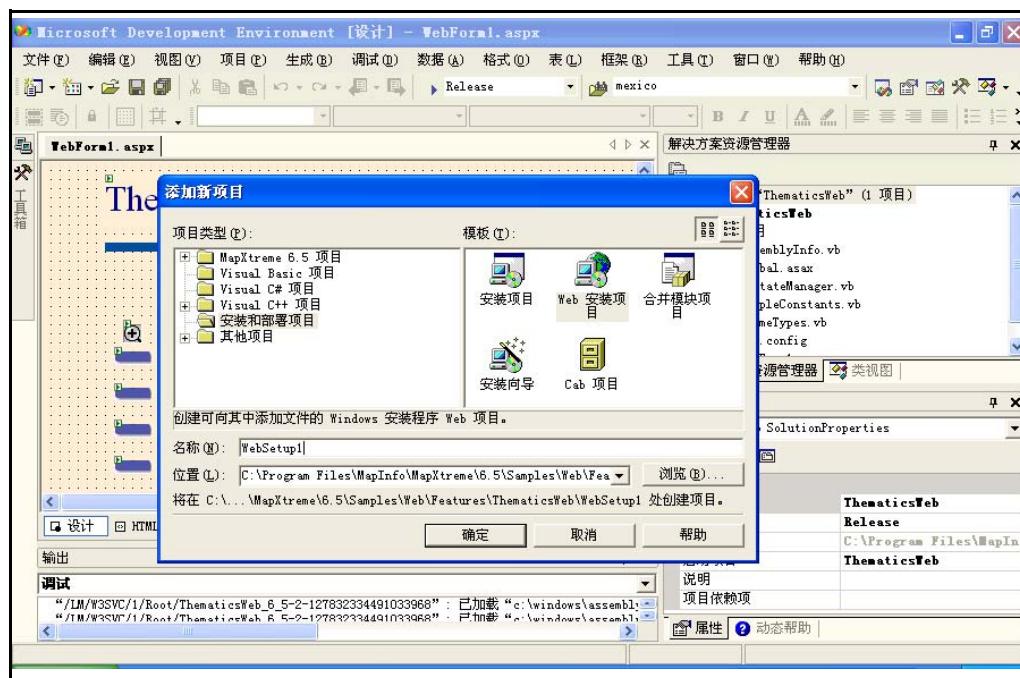
4. 生成

现在，生成该项目并创建 Setup.exe 文件，该文件包含所有数据、编译代码和项目必需的 MSM。该安装项目也是 Visual Studio .NET 解决方案的一部分，因此，在生成整个解决方案时，Visual Studio .NET 先编译和生成地图绘制应用程序，然后再编译和生成用于安装的 Setup 文件。如果在其它计算机上执行该安装文件，则会在其中安装整个应用程序，该应用程序中包含所有必需文件。以上就是本部分全部内容。

Web 应用程序打包

为 web 应用程序创建程序包与为桌面应用程序创建程序包类似。首先，要将新项目增加到解决方案。此次要选择“Web 安装项目”，而不是安装向导。请遵循以下步骤：

1. 在解决方案中创建新的 Web 安装项目。
 - a. 在“解决方案浏览器”中，突出显示“解决方案”并选择**文件 > 增加项目 > 新建项目 ...**
 - b. 从“安装和部署项目”中，选择**WEB 安装项目**，创建特定于 web 项目的安装应用程序。



2. 下一步，指示要包括解决方案的哪些部分。右键单击新创建的 Web 安装项目并选择**增加 > 项目输出 ...**。选择**主要输出**和**内容文件**。

必须包括内容文件，因为应用程序中的 web 页面是作为内容生成，不同于桌面应用程序中的可执行文件。

3. 增加本地目录用于保存所有数据，以便应用程序可以进行查找。
 - a. 在“解决方案浏览器”中，右键单击 Web 安装项目并选择**查看 > 文件系统**。
 - b. 在“文件系统”窗口中，右键单击**目标计算机上的文件系统**并选择“增加特殊文件夹”。
 - c. 从生成列表中选择**自定义文件夹**。将该文件夹的名称更改为 CDrive。在“属性”中，将 **DEFAULTLOCATION** 设置到 C:\。
 - d. 右键单击 CDrive 目录并增加嵌套文件夹，该文件夹与 web.config 中预载的 workspace 文件具有相同的文件夹结构。
 - e. 右键单击嵌套文件夹的最后一个文件夹并从在 web.config 中为预载 workspace 文件指定的目录增加数据。其中包括以下文件：

GRID15.DAT	GRID15.ID	-	GRID15.MAP	grid15.tab
OCEAN.DAT	OCEAN.ID	-	OCEAN.MAP	ocean.TAB
WLDCTY25.DAT	WLDCTY25.ID	WLDCTY25.IND	WLDCTY25.MAP	wldcty25.TAB
WORLD.DAT	WORLD.ID	WORLD.IND	WORLD.MAP	world.TAB
WORLDCAP.DAT	WORLDCAP.ID	WORLDCAP.IND	WORLDCAP.MAP	worldcap.TAB

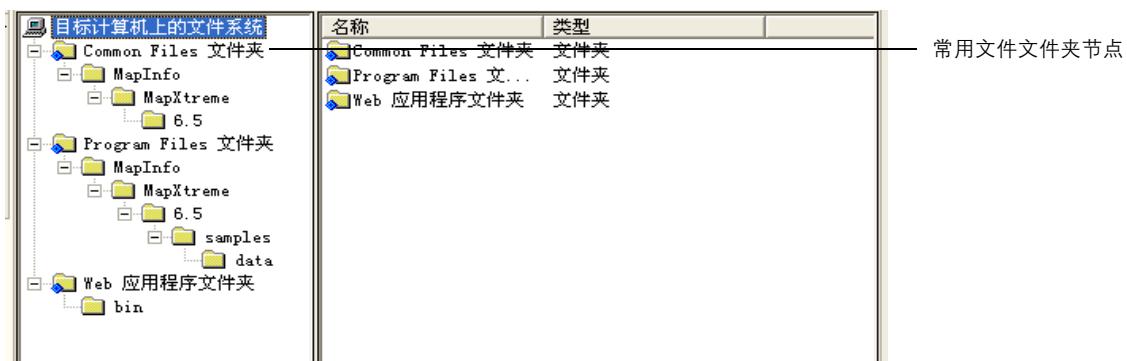
将在当前应用程序目录下生成这些文件的 world.mws。

注： 如果要将该应用程序部署到另一台计算机，则开发计算机和部署计算机上数据的绝对路径必须一致。

- f. 重复步骤 e，将 eworld.* 文件增加到同一文件夹，这些文件位于 C:\Program Files\MapInfo\MapXtreme\6.5\Samples\Web\Features\ThematicsWeb\vb
4. 从 Visual Studio .NET、Window 的 FileSystem 选项卡窗格中，选择“Web 应用程序”文件夹并打开“属性”窗口。将 VirtualDirectory 名称更改为 ThematicsWeb_6_5。我们在该位置指定目标计算的虚拟目录名称。
5. 现在，增加许可证文件。对于 web 应用程序，只有具有有效的 MapXtremeWeb.lic 副本用户才可以访问地图。在该示例中，使用了与 MapXtreme 2005 版本对应的试用许可证¹。有关部署许可证的详细信息请参阅第 437 页中的**授权分布式应用程序**。
 - a. 在“文件系统”窗口中，右键单击**目标计算机上的文件系统**并选择“增加特殊文件夹”。

1. 如需为部署创建程序包，则 MapXtremeTrial.lic 必须来自 SDK 或 MapXtreme 2005 的运行时安装。使用 MapXtreme 2005 的评估版本无法创建安装项目。

- b. 从生成列表中选择 **COMMON FILES** 文件夹。
- c. 右键单击 **COMMON FILES** 文件夹节点，然后增加名为 MapInfo 的文件夹。继续创建嵌套文件夹 MapXtreme 和 6.5，则文件树如下图所示。



- d. 右键单击 6.5 并增加文件。在这种情况下，要选择名为 MapXtremeTrial.lic 的文件（位于 c:\Program Files\Common Files\MapInfo\MapXtreme\6.5）。
6. 为使新生成的应用程序在外部服务器上正常运行，需要执行另一步骤。由于需要使用文件 eworld.tab、eworld.mdb 和 eworld.aid 创建某些主题图，因此它们必须能够访问该应用程序。在“解决方案浏览器”中查找这些文件并将其突出显示以检查其属性。（如果这些文件不可见，请单击位于“解决方案浏览器”顶部的**显示所有文件**按钮）。右键单击并选择**包括在项目中**。在“高级”组的“属性”窗口中，将每个文件的**生成操作**设置更改为**内容**。这样安装程序就会将这些文件以文件形式放置到容器目录，而不是将它们捆绑到.dll，这些文件就能够访问该应用程序。

现在，web 应用程序包的创建已完成。在“生成”菜单中选择**批处理文件生成**，然后选中与 Release 版 ThematicsWeb 和 安装项目对应的“生成”复选框来执行生成。单击**生成**并运行。在安装项目的 Release 目录下，将看到名为 Setup.exe 的文件和 / 或 setup.msi。使用该文件执行部署。

部署应用程序

生成 web 或桌面应用程序并将所有部署组件打包后要执行最后一个步骤，即将其放入运行的工作站。

部署桌面应用程序

对应用程序的有效部署，就是将通过本教程前几页创建的 Setup.exe 文件提供给最终用户。只需将 Setup.exe 文件复制到软盘或光盘中，然后发送出去。用户在桌面计算机上启动 Setup.exe 可执行文件时，安装开始并逐步引导他们完成简单的安装步骤。以上就是本部分全部内容！

部署 Web 应用程序

要部署 web 应用程序，请执行与桌面应用程序相同的步骤，即，将文件传输到我们的服务器并在此处运行安装。安装应用程序在 IIS 中创建所需的虚拟目录并将其所有必要组件放置到正确位置。在上述应用程序中，许可证文件应放置在 Program Files\Common Files\MapInfo\MapXtreme\6.5 目录并创建新目录，用于保存数据（与 web.config 中预载的 workspace 文件具有相同的文件夹结构）。

要查看 web 应用程序的运行情况，请打开浏览器，导航到服务器并选择虚拟目录名称。具体形式表现为 `http://MyMachine/ThematicsWeb_6_5/`。在 [步骤 4](#) 中将该名称指定为虚拟目录。此时就应该能够查看到新 web 应用程序的运行情况！

注：由于 Microsoft Framework 的问题，可能需要结束服务器上的 web 处理，即 `aspnet_wp.exe`，以便成功运行应用程序。此处理在将其结束时自动重启，因此在“任务管理器”中察觉不到任何变化。

授权分布式应用程序

MapXtreme 2005 要在每个部署中包括运行时许可证。要执行该操作，则必须在程序包中提供许可证或包括客户可用于获得自身许可证的许可证工具。运行时许可证可从 MapInfo Corporation 获得。

需要的运行时许可证类型取决于正在生成的应用程序类型和使用的 MapXtreme 2005 的版本。在下表中查看具体情况：

部署类型	MapXtreme 2005 的 SDK 版本	节点锁定？	如何获得运行时许可证
台式	无复制保护	否	联系 MapInfo Corporation 客户服务部门获得一种或多种情况下的运行时许可证。
台式	复制保护	是	生成每个坐席的节点 ID 文件并提交给 MapInfo Corporation 客户服务部门。
Web	无复制保护	是	生成每个坐席的节点 ID 文件并提交给 MapInfo Corporation 客户服务部门。
Web	复制保护	是	生成每个坐席的节点 ID 文件并提交给 MapInfo Corporation 客户服务部门。

要生成部署计算机的节点 ID，请运行许可证工具 MILicenseNodeID.exe 并将生成的文本文档提交给 MapInfo Corporation 客户服务部。许可证工具与 MapXtreme 2005 SDK 同时安装到 \Program Files\Common Files\MapInfo\MapXtreme\6.5\License Tool。请参阅 MapXtreme 2005 程序包中或 \MapInfo\MapXtreme\6.5\Documentation\PDF 下的“软件复制保护”许可证应用程序文档。

许可证工具包括在合并模块 MapInfoMXTConfig_6.5.msm 中。将 MapInfoCoreEngine.msm 包括在分发中时，您的用户将自动获得该工具。请参阅第 34 页第 2 章中的使用运行时安装程序部署应用程序的步骤。

E

案例研究：在 Web 应用 程序中管理会话状态

本附录介绍了如何在 MapXtreme 2005 web 应用程序中实现最佳状态管理的案例研究。在此案例研究中将使用 ThematicsWeb 示例应用程序。

本附录内容：

- ◆ 案例研究：管理状态 440

案例研究: 管理状态

状态管理的原则和最佳实践位于第 60 页第 5 章中的规划 Web 应用程序。本案例研究说明了如何在 web 应用程序中应用这些最佳实践。本讨论着眼于 MapXtreme 2005 随附的 ThematicsWeb 示例应用程序上下文的状态管理。

ThematicsWeb 示例说明了如何使用以下内容:

- 预载的地图图层工作空间和设置
- 从 MS Access 表绑定数据
- 手动状态, 用于管理会话状态
- 池会话对象, 用于获取高效性能

本示例设计用于检测应用程序用户是否正在生成首次请求（在这种情况下应用程序应位于其初始状态），或用户是否正在重新访问该应用程序并想要维护其个人设置。由于本示例与多个用户共享池对象，因此它会检测池会话是位于其初始状态（“干净的”）还是包含其它用户的更改（“不干净的”）。为了分别维护每个人的设计，本示例使用设置为手动的 Session.State，而不使用 HttpSessionState。这样开发人员就可以控制为每个用户保存的状态数。

ThematicsWeb 示例应用程序将会加载工作空间文件 world.mws，该文件包括五个 FeatureLayers 和一些 LabelLayers。该示例应用程序为空间表使用 world.tab。主题图的属性数据来自 eworld.mdb。请注意在创建 MapInfo Session 时会预装此 world.mws，且在访问该应用程序时会增加 eworld 列。要获得更好的性能并使管理状态变得更容易，请将属性列信息放置到工作空间文件中，那么在创建 MapInfo Session 对象时，所有需要的数据将会预装到此 MapInfo Session 对象中。另请参阅第 85 页第 5 章中的相关表。

应用程序设置

首先, 查看 Web.config 文件的以下部分内容, 该文件包括了应用程序设置:

```
<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\6.5\Samples\Data\World.mws" />
    .....
  </appSettings>
  .....
  <sessionState mode="StateServer" ... />
  .....
</configuration>
```

黑体文本显示的设置的说明如下:

- MapInfo.Engine.Session.Pooled 设置为 "true" 意味着 MapInfo Session 对象由 COM+ 服务放入池中。通过使用 MapInfo.Engine.Current 检索的 Session 对象可能是刚从空白创建的新 Session 对象（“干净的”）或由其它请求创建的“不干净的”Session 对象。干净的和不干净的 Session 对象对您的请求来说可能都包含了过多或过少的数据。要避免出现这样的情况, 应维护您自己的初始状态并遵循状态管理。
- MapInfo.Engine.Session.State 设置为 "Manual" 意味着您必须自己管理应用程序状态。有关示例, 请参阅第 441 页中的实现状态管理器。手动设置有助于提高性能, 因为将仅保存和恢复更改的数据。不必为每个请求都保存未更改的数据（请参阅第 64 页第 5 章中的背景地图）。
- MapInfo.Engine.Session.Workspace 值是一个或多个在创建每个 MapInfo Session 对象时要预装的工作空间文件。在工作空间之间使用分号以指定多个工作空间。若选择不预装工作空间, 则必须增加代码以确定会话对象是否已包含数据。若不预装工作空间, 还必须保存和恢复更多数据。

实现状态管理器

若在 Web.config 文件中将 MapInfo.Engine.Session.State 设置为 Manual, 则必须通过从抽象基类 MapInfo.WebControls.StateManager 继承来实现您自己的状态管理器类。若不执行此操作, 则 MapXtreme 2005 会抛出异常。有如下两种必须实现的方法:

```
Public Overrides Sub RestoreState()
Public Overrides Sub SaveState()
```

MapXtreme 2005 提供了StateManager 的 VB 示例, 位于 ThematicsWeb 示例应用程序文件夹中。

若要使用 HttpSessionState 管理状态，则会具有包括您自己的状态管理器的选项。要在此情况下放置和获取您自己的 StateManager，则使用以下静态方法。

```
MapInfo.WebControls.StateManager.PutStateManagerInSession(yourOwnStateManager)
MapInfo.WebControls.StateManager.GetStateManagerFromSession()
```

初始状态管理

ASP.NET Session.IsNewSession 为 true 时会初始化应用程序状态。请遵循以下步骤。

1. 创建 MyStateManager 对象并将地图别名增加到状态管理器的 ParamsDictionary 属性。若在 Web.config 文件中将 MapInfo.Engine.Session.State 设置为 "Manual"，则该步骤是强制性的。

还可以将多个键 / 值对增加到 ParamsDictionary 中。这些键 / 值对将由 ASP.NET 保存和恢复，并且它们应是可串行化的。

```
Dim stateManager As New MyStateManager

stateManager.ParamsDictionary.Item(stateManager.ActiveMapAliasKey)
    =
    = Me.MapControl1.MapAlias
```

2. 将此状态管理器对象放置到 ASP.NET Session 中，那么 ASP.NET 框架可以自动保存和恢复该对象。

```
MapInfo.WebControls.StateManager.PutStateManagerInSession(stateManager)
```

3. 使用 InitState() 方法初始化应用程序数据。

- a. 若 MapInfo.Engine.Session.Current.Catalog 没有打开的表，则加载 eworld.mdb (Microsoft Access 表)。请注意，仅为每个池 MapInfo Session 对象打开一次 Microsoft Access 表，原因是为该示例应用程序加载该表后，表内容并不发生更改。若将入池设置为 false，由于将会为每个请求创建新的 MapInfo Session 对象，那么每次都会打开 Microsoft Access 表。

Microsoft Access 表的三个数字列会绑定到由预装的工作空间文件打开的 world.tab。

- b. 获取 MapInfo.Mapping.Map 对象的初始状态。执行该操作有两种方法。一种方法用于该请求是应用程序的首次请求。另一种方法用于后续请求。

若请求是应用程序的首次请求，则设置地图的缩放、中心和大小，并使用 MapInfo.WebControls.ManualSerializer 将地图对象初始化为字节数组并将该字节数组放置到 HttpSessionState 中，从而使其可以全局访问。带有字节数组格式的地图对象是地图的初始状态。

```
Dim originalMap As String = "original_map"
' Set zoom, center and size here...
' Serialize MapInfo object into byte array and put it into -
HttpApplicationState.
Me.Application.Item(originalMap) = _
```

```
MapInfo.WebControls.ManualSerializer.BinaryStreamFromObject (myMap)
```

若请求不是应用程序的首次请求，则检索 `HttpApplicationState` 中使用字节数组保存的地图对象，并将该对象反串行化为正常地图对象以获取初始状态。

`MapInfo` 地图对象反串行化会尝试确定是否有带有相同地图别名的地图对象存在。若有一个带有相同地图别名的地图对象存在，则地图对象反串行化会自动将其值应用到现有的具有相同地图别名的地图对象；否则，地图对象反串行化将创建新的地图对象并将其值应用到新创建的地图对象。

```
Dim bytes As Byte() = CType(Me.Application.Item(originalMap),
Byte())
' This step will deserialize myMap object back and all original
states will be put back to
' myMap if myMap has same alias name as the one stored in
HttpApplicationState.
Dim obj As Object = _
```

```
MapInfo.WebControls.ManualSerializer.ObjectFromBinaryStream(bytes)
```

c. 在地图图层列表的顶部创建 `GroupLayer` 以保存为该示例临时创建的所有 `ObjectThemeLayers` 或 `LabelLayers`。将 `GroupLayer` 对象放置到 `HttpSessionState` 以维护那些临时创建的对象的状态。

将 `GroupLayer` 放置在地图图层顶部列表的原因是要显示那些临时的 `ObjectThemeLayers` 或 `LabelLayers`。因此，在每次接收到请求时，必须将 `GroupLayer` 的位置调整到顶部。将会在 `MyStateManager` 类 `RestoreState()` 方法中看到详细信息。

```
If (Not myMap.Layers.Item(SampleConstants.GroupLayerAlias) Is
Nothing)
    Then myMap.Layers.Remove(SampleConstants.GroupLayerAlias)
End If
' put the GroupLayer on the top of Layers collection, so contents
within it could get displayed.
myMap.Layers.InsertGroup(0, "grouplayer",
SampleConstants.GroupLayerAlias)
```

正常状态管理

正常状态管理描述了建立初始状态后如何管理应用程序状态。这将用到 MapInfo.WebControls.StateManager 中的抽象方法 SaveState() 和 RestoreState()。必须在子类中实现它们以满足您自己的应用程序的需要。若 MapInfo.Engine.Session.State 设置为 "Manual"，则必须获取从 MapInfo.WebControls.StateManager 继承的您自己的类。否则，不必继承 MapInfo.WebControls.StateManager。

保存状态:

在页面 Unload 事件句柄中，将通过使用从 HttpSessionState 检索的 StateManager 对象来保存应用程序状态。

注： 若将 Server.Transfer 或带有 endResponse 标记的 Response.Redirect 用作 true，则将不会调用 Page_UnLoad 方法，原因是那些方法将会忽略该事件句柄。可以在将 Server.Transfer 或带有 endResponse 标记的 Response.Redirect 用作 true 之前显式地调用 MapInfo.WebControls.StateManager.GetStateManagerFromSession().SaveState() 以解决该问题。

```
Private Sub Page_UnLoad(ByVal sender As Object, ByVal e As EventArgs) _
Handles MyBase.Unload
If (Not MapInfo.WebControls.StateManager.GetStateManagerFromSession() Is Nothing) Then
    MapInfo.WebControls.StateManager.GetStateManagerFromSession().SaveState()
End If
End Sub
```

请检查 MyStateManager 的 SaveState() 方法并注意将表分配到 HttpSessionState 的顺序。必须先分配 Access 表，然后再分配主题图表(world.tab)，原因是在本示例中主题图表取决于 Access 表。接收到新的请求时，将会先反串行化 Microsoft Access 表。（请注意，在接收到新的请求时，ASP.NET 框架将自动反串行化保存在 HttpSessionState 中的对象。）此外，必须在主题图图层（本案例中为 GroupLayer）之前保存表，原因是这些主题图图层全都取决于表。

```
Public Overrides Sub SaveState()
    Dim mapAlias As String =
    CType(ParamsDictionary.Item(MapInfo.WebControls.StateManager.ActiveMap
        AliasKey), String)
    Dim myMap As MapInfo.Mapping.Map = Me.GetMapObj(mapAlias)
    MapInfo.WebControls.StateManager.SaveZoomCenterState(myMap)
    ' The order matters in this case since theme_table contains some temp
    columns constructed from mdb_table.
    ' and some themes/modifiers in the theme_layer are based on the
    theme_table.
```

```
HttpContext.Current.Session.Item("mdb_table") =
MapInfo.Engine.Session.Current.Catalog.Item(SampleConstants.EWorldAlias)
HttpContext.Current.Session.Item("theme_table") =
MapInfo.Engine.Session.Current.Catalog.Item(SampleConstants.ThemeTableAlias)
HttpContext.Current.Session.Item("theme_layer") =
myMap.Layers.Item(SampleConstants.ThemeLayerAlias)
HttpContext.Current.Session.Item("group_layer") =
myMap.Layers.Item(SampleConstants.GroupLayerAlias)
End Sub
```

恢复状态:

在页面 Load 事件句柄中，通过使用从 HttpSessionState 检索的 StateManager 对象来恢复应用程序状态。

```
If (Not MapInfo.WebControls.StateManager.GetStateManagerFromSession() Is Nothing) Then
    MapInfo.WebControls.StateManager.GetStateManagerFromSession().RestoreState()
End If
```

请检查 MyStateManager 的 RestoreState() 方法。已保存的表、主题图图层和 GroupLayer 由 ASP.NET 框架自动恢复，原因是在 SaveState() 方法中，它们直接分配给 HttpSessionState。恢复后，必须更改 GroupLayer 的位置，原因是未保存整个图层列表（且当前地图的图层列表没有 GroupLayer）。因此，必须手动将 GroupLayer 调整到正确位置，在本案例中，正确的位置是图层列表的顶部。

```
Public Overrides Sub RestoreState()
    Dim mapAlias As String =
    CType(ParamsDictionary.Item(MapInfo.WebControls.StateManager.ActiveMap
        AliasKey), String)
    Dim myMap As MapInfo.Mapping.Map = Me.GetMapObj(mapAlias)
    MapInfo.WebControls.StateManager.RestoreZoomCenterState(myMap)
    ' the position of the groupLayer inside the Layers collection
    ' is not guarantee when it get deserialized back
    ' since we don't save the whole Layers collection for
    ' performance purpose in this sample.
    Dim indexOfGroupLayer As Integer =
    myMap.Layers.IndexOf(myMap.Layers.Item(SampleConstants.GroupLayerAlias))
    If (indexOfGroupLayer > 0) Then
        myMap.Layers.Move(indexOfGroupLayer, 0)
    End If
End Sub
```


F

定制 MapXtreme 2005

本附录包含 MapXtreme 2005 中可定制的类、接口和其它元素的示例。

本附录内容：

- ◆ 可定制的类，第 448 页
- ◆ *MapXtreme 2005* 其它可定制的方面，第 455 页

可定制的类

以下类设计用于简化子类，以创建更适合开发需要的定制子类。

Engine.CustomProperties

使用 CustomProperties 类将定制信息添加到对象。可以由 CustomProperties 添加信息的不同种类的对象是：FeatureCollection、FeatureStyleModifier、GmlFeatureCollection、GroupLayer、IfeatureCollection、ImapLayer、Isession、LabelModifier、LabelSource、Legend、LegendFrame、Map、MapLayer、MapTool、MultiFeatureCollection、Session.PooledSession 和 Table。使用 Add 方法添加信息。使用 Item 方法检索信息。CustomProperties 可以是其中的任意类型。

注：不要直接将 MapXtreme 2005 对象增加到 CustomProperties 集合中。如果这样做，会在串行化期间引起错误。相反，应该添加 Alias。例如，不将 Map 添加到 CustomProperties 集合。而是添加 Map.Alias。

以下是一个示例，说明了 CustomProperties 如何用于添加和检索属性：

```
using MapInfo.Engine;

CustomProperties bag = new CustomProperties();
bag.Add("One", 1);
bag.Add("DateNow", DateTime.Now);
int i = (int)bag["One"];
DateTime ts = (DateTime)bag["DateNow"];
```

搜索

有一些类可以用于定制搜索函数，以实现几乎任何种类的搜索需要。QueryFilter 类用于创建定制“where”子句用于 SQL 查询。QueryDefinition 类用于执行定制的自定义 SQL 查询语句； SearchResultProcessor 类则设置查询结果的后处理。

关于如何使用上述类的示例，请参考 MapXtreme 2005 安装的 Samples 目录中的 Search 示例应用程序。（默认安装位置是：

C:\Program Files\MapInfo\MapXtreme\6.x\Samples\Desktop\Features\Search）。

QueryFilter

IqueryFilter 接口定义所有查询筛选器必须支持的接口。QueryFilter 用于定义 QueryDefinition 的全部或部分“where”子句。

QueryDefinition

QueryDefinition 类定义搜索执行的查询。QueryDefinition 由筛选器 (IQueryFilter)、Columns 和 OrderBy 组成。如果不指定 Columns，则使用“*”。

I SearchResultProcessor 或 SearchResultProcessor

SearchResultProcessor 实现 IsearchResultProcessor 接口，该接口用于执行搜索结果的后处理以缩小选定行的范围。

FeatureStyleModifier 或 FeatureOverrideStyleModifier

创建定制 FeatureStyleModifiers 以通过 FeatureStyleModifier 或 FeatureOverrideStyleModifier 的派生更改图层外观。上述类用于指定特定图层中对象的特定 FeatureStyle 修改。

以下示例代码创建修饰符，如果人口从 1990 到 2000 年之间在减少，则修饰符将 USA.TAB 中的区域设置为红色。

```
using MapInfo.Mapping;
using MapInfo.Style;

internal class UsaPopulationDecreasedModifier :
MapInfo.Mapping.FeatureStyleModifier
{
    public UsaPopulationDecreasedModifier() : base(null, null)
    {
        // use 2 columns for expressions
        Expressions = new string[] { I18N.Wrap(
            "Pop_1990", I18N.WrapType.Column),
            I18N.Wrap("Pop_2000", I18N.WrapType.Column) };
    }

    // called during draw events
    protected override System.Boolean
        Modify(MapInfo.Styles.FeatureStyleStack styles,
        object[] values)
    {
        // compares the values from each column
        if (double.Parse(values[0].ToString()) >
            double.Parse(values[1].ToString()))
        {
            // if population decreased, color the region red
            CompositeStyle style = styles.Current;
            if (style.AreaStyle.Interior is SimpleInterior)
            {
                ((SimpleInterior)style.AreaStyle.Interior).ForeColor =
                    Color.Red;
            }
        }
    }
}
```

```
        // modifies the region
        return true;
    }
}
// does not modify the region
return false;
}
}
```

UserDrawLayer

UserDrawLayer 类用于绘制地图中的定制图层。可以在该图层中填充任何需要的对象，包括定制徽标，点的透明覆盖等。要使用该类，从该类派生新类并重载绘制方法。以下示例代码说明了上述方法：

```
using MapInfo.Mapping;

class MyUserDrawLayer : UserDrawLayer, ISerializable
{
    string myLayerName = "My User Draw Layer Name";
    public string MyName { get { return myLayerName; } }
    set { myLayerName = value; } }
    // This constructor must be present for the class to be
    // serializable
    public MyUserDrawLayer(string Name,
                           string Alias) : base(Name, Alias) {}
    public override void Draw(MapInfo.Geometry.LRect ClientRect,
                           MapInfo.Geometry.LRect UpdateRect, Graphics graphics)
    {
        // Create pen.
        Pen blackPen = new Pen(Color.Black, 3);
        // Create location and size of rectangle.
        float x = 10.0F;
        float y = 10.0F;
        float width = 200.0F;
        float height = 200.0F;
        // Draw rectangle to screen.
        graphics.DrawRectangle(blackPen, x, y, width, height);
    }
    public override void GetObjectData(SerializationInfo info,
                                      StreamingContext context)
    {
        base.GetObjectData(info, context);
        info.AddValue("myLayerName", myLayerName);
    }
    public override void SetObjectData(SerializationInfo info,
                                      StreamingContext context)
    {
```

```
        base.SetObjectData(info, context);
        myLayerName = info.GetString("myLayerName");
    }
}
```

VB UserDrawLayer 示例应用程序与 ..\MapInfo\MapXtreme\6.x\Samples\Desktop\Features 文件夹中的 MapXtreme 2005 同时提供。

Windows.Controls

Windows.Controls 命名空间中的很多类可以创建子类以实现定制行为。以下是使用该途径定制 LayerControl 的两个示例。

MapInfo.Windows.Controls.PropertiesUserControl

如果需要将定制的选项卡添加到 LayerControl，则创建 PropertiesUserControl 的子类。然后通过调用 GetLayerTypeControls() 和 SetLayerTypeControls() 方法，将定制类添加到其它选项卡类的集合中。该方法工作的示例，请参阅包含在 MapXtreme 2005 安装路径 Samples\Features 目录下的示例应用程序。关于 PropertiesUserControl 的子类 MapBackgroundControl 类，请参阅 LayerControl 示例应用程序中的 MapBackgroundControl.cs 文件。

MapInfo.Windows.Controls.LayerNodeHelper 和它的所有子类

LayerNodeHelper 类控制 LayerControl 的图层树中节点的外观和行为。例如，这些帮助程序类指示用户可以删除的图层。如果 ILayerNodeHelper 接口的属性不能对图层节点的外观或行为提供足够的控制（例如，如果需要应用按表逻辑，该逻辑允许用户删除部分地图图层，但不能删除其它），可以将任意帮助类子类化，重载适当的方法，执行定制重载方法中的按表逻辑。然后通过调用 LayerControl.SetLayerTypeHelper 方法应用新的帮助程序类。

默认情况下，用户删除 GroupLayer 时，将出现确认对话框，除非在 GroupLayer 完全为空的情况下。在这种情况下，空的 GroupLayer 将被简单删除而不出现确认提示（LabelLayers 的工作方式类似）。要强制 Layer Control 始终显示确认提示，甚至在 GroupLayer 为空时也显示，可以创建从 GroupLayerNodeHelper 派生并重载 IsRemovalConfirmed 方法的类。以下代码示例说明如何实现该方法：

```
using System;
using MapInfo.Windows.Controls;

namespace MapInfo.Samples.LayerControl
{
    public class CustomLayerNodeHelper : GroupLayerNodeHelper
    {
        public CustomLayerNodeHelper()
        {
```

```
        }
        public override bool IsRemovalConfirmed(object obj)
        {
            return true;
        }
    }
}
```

然后，要在应用程序中使用该新帮助程序类，使用如下的语法调用 LayerControl 的 SetLayerTypeHelper 方法：

```
layerControlDlg.LayerControl.SetLayerTypeHelper(
    typeof(GroupLayer), new CustomLayerNodeHelper() );
```

LayerNodeHelper 的子类是：

- MapNodeHelper
- MapLayerNodeHelper
- GroupLayerNodeHelper
- LabelLayerNodeHelper
- LabelSourceNodeHelper
- LabelModifierNodeHelper
- FeatureStyleModifierNodeHelper
- RangedThemeNodeHelper
- DotDensityThemeNodeHelper
- IndividualValueThemeNodeHelper
- PieThemeNodeHelper
- BarThemeNodeHelper
- GradSymbolThemeNodeHelper

工具

可通过两种方式定制工具：通过现有工具的子类化或通过将代码添加到定制基本工具。有关桌面工具的详细信息，请参阅第 111 页第 7 章中的可用工具。有关 web 工具的详细信息，请参阅第 93 页第 6 章中的运用 *MapXtreme 2005 Web 控件*。

样式

以下类可以用于将定制信息添加到应用于对象的样式集。

BitmapPointStyleRepository

BitmapPointStyleRepository 类用于重复当前所有位图点样式，并允许用户从指定的目录添加新位图图像。使用 Refresh() 方法将任意新位图图像放置到保存所有图像的库中。以下代码示例演示了“C:\MyData\MyBitmapImages”目录下用于 BitmapPointStyles 的图像集合的重新加载。

```
using MapInfo.Styles;

StyleRepository styleRepository = Session.Current.StyleRepository;
BitmapPointStyleRepository bmpPointStyleRepository =
    styleRepository.BitmapPointStyleRepository;
bmpPointStyleRepository.Reload("C:\MyData\MyBitmapImages");
```

InteriorStyleRepository

InteriorStyleRepository 类用于重复当前所有内部样式，并允许用户从指定的目录添加新内部样式的新位图图像。要添加到用于填充区域的内部图案的当前集上，请使用 AppendBitmapPattern() 方法。以下代码示例演示了将新位图图像添加到 InteriorStyleRepository。

```
using MapInfo.Styles;

StyleRepository styleRepository = Session.Current.StyleRepository;
InteriorStyleRepository interiorStyleRepository =
    styleRepository.InteriorStyleRepository;
// AppendBitmapPattern returns the zero-based index of the pattern in
// the repository. The index returned may be used to
// retrieve a SimpleInterior style using the repository indexer.
int index = interiorStyleRepository.AppendBitmapPattern(
    "C:\MyImages\trees.bmp"
);
if (index >= 0)
{
    SimpleInterior style =
        (SimpleInterior)interiorStyleRepository[index];
}
```

LineStyleRepository

LineStyleRepository 类用于重复当前所有直线样式并允许用户添加新直线样式（以 MILISTED.EXE 创建，可以从 [MapInfo.com](#) 下载该程序）。新 MAPINFO.PEN 文件是由 MILISTED.EXE 应用程序创建，然后可以使用 Reload() 方法将新文件放置到库中。

“C:\MyData” 目录下 PEN 文件中的示例代码演示了用于 SimpleLineStyles 的直线样式图案集的重新加载。

```
using MapInfo.Styles;

StyleRepository styleRepository = Session.Current.StyleRepository;
LineStyleRepository lineStyleRepository =
    styleRepository.LineStyleRepository;
lineStyleRepository.Reload("C:\MyData\MyLineStyles.PEN");
```

VectorPointStyleRepository

VectorPointStyleRepository 类用于重复当前所有矢量符号，并允许用户从指定的文件添加新矢量符号。要创建符号集的定制版本，需要使用 Symbol Maker MapBasic 应用程序。矢量符号集更改后，使用 Reload() 方法将符号集读取到库中。“C:\MyData” 目录下 FNT 文件中的示例代码演示了 SimpleVectorPointStyles 集的重新加载。

```
using MapInfo.Styles;

StyleRepository styleRepository = Session.Current.StyleRepository;
VectorPointStyleRepository vecPointStyleRepository =
    styleRepository.VectorPointStyleRepository;
vecPointStyleRepository.Reload("C:\MyData\MyVectorSymbols.FNT");
```

GmlFeatureCollection

GmlFeatureCollection 类用于从 XML 文件导入图元。可以通过创建 FeatureCollection 对象并构造 FeatureObjects 加入创建的对象的方法，将 Features 添加到地图。然后可以将 FeatureCollection 插入到地图中。

WorkSpacePersistence 和 WorkSpaceLoader

WorkSpaceLoader 可以被子类化，以便使用的永久文件可以保存从应用程序保存的附加信息。开发人员可以使用应用程序写入任何需要保存到永久 XML 文件中的信息，做法是通过在 UserData 节点下添加信息。WorkSpacePersistence 类写入持续文件且该节点中有某些内容时，该内容将自动写入文件。WorkSpaceLoader 类或该类的某些派生类读取的数据包含 UserData 节点的子节点。如下代码是它的示例。

```
using MapInfo.Persistence;

class myWorkSpaceLoader : WorkSpaceLoader
{
    public myWorkSpaceLoader(string fileName) : base(fileName)
    {
        // empty
    }
    override public void Load()
    {
        // called parent Load() method.
        base.Load();

        XmlNode userData = this.UserDataXmlNode;
        foreach (XmlNode childNode in userData.ChildNodes)
        {
            string text = childNode.Name;
            text = childNode.InnerText;
            // user can do their own load stuff here
        }
    }
}
```

MapXtreme 2005 其它可定制的方面

ADO.NET

如果访问存储在 DataTable 中的数据，该数据无法通过 ADO.NET 数据提供方访问，可以使用 TableInfoAdoNet 类以访问不可制图的数据的 TableTypes（尚未为其提供 DataSource）。关于使用该类的更详细信息，请参阅联机帮助中的 TableInfoAdoNet 类。

应用程序数据文件的位置

任何 MapXtreme 2005 应用程序都使用以下文件中存储的数据。

文件类型	文件名
缩写文件	MAPINFOW.ABB
画笔文件	MAPINFOW.PEN
投影文件	MapInfoCoordinateSystemSet.xml
矢量符号文件	MapInfow.fnt
定制符号目录	CustSymb
Nadcon 文件	*.las、*.los
jgd2000 文件	jgd2000.*

默认情况下，MapXtreme 应用程序在以下目录寻找数据文件：

- C:\Program Files\Common Files\MapInfo\MapXtreme\6.0 - MapXtreme 2005 安装程序存放这些文件的目录。
- 用户的应用程序存放的目录。对于 Windows 应用程序，这是存放 .exe 文件的目录。对于 web 应用程序，这是存放 Web.config 文件的目录。
- MapInfo.CoreEngine 程序集文件夹 - 该目录是应用程序寻找数据文件的最后位置。

对应以上位置的目录列表可以通过 ISessions.AppDataPaths 属性获得，该属性将列表以字符串数组的形式返回。当不能通过编程方式添加到该列表时，可以通过在应用程序的配置文件中定义定制部分的方法添加到列表。以下示例代码说明了该方法。注意，附加路径定义的顺序确定它们搜索的顺序，并且所有定制路径在以上列出的默认位置之前搜索。

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=4.0.0.350, Culture=neutral,
            PublicKeyToken=93e298a0f6b95eb1" />
    </configSections>
    <MapInfo.CoreEngine>
        <ApplicationDataPaths>
            <SpecialPath>
                <LocalApplicationData>YourCorporation\Your
Application<\LocalApplicationData>
                <\SpecialPath>
            </ApplicationDataPaths>
        </MapInfo.CoreEngine>
    </configuration>
```

在该示例中，MapXtreme 2005 将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> 元素不应包括 CustSymb 文件夹名称。如果该元素包含了文件夹名称，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在相对于 .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在“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。

以上配置元素中的每个元素按以下方式定义：

<configSections>

这是声明定制部分句柄的位置。这种情况下，程序集 MapInfo.CoreEngine 中定义的 ConfigSectionHandler 类负责分析配置文件的 MapInfo.CoreEngine 部分。

<MapInfo.CoreEngine>

该部分包含 MapInfo.CoreEngine 程序集的设置。特定地，应用程序数据搜索路径。

<ApplicationDataPaths>

该部分定义 MapXtreme 2005 在搜索应用程序数据文件时使用的附加路径。

<Path>

如果应用程序数据的完整路径已知则要使用的元素。

<SpecialPath>

如果应用程序数据存储在众所周知的系统文件夹的相对位置，则使用该语法。例如，如果应用程序数据存储在名为 MyAppData 的文件夹中，位于“My Documents”目录下，则配置文件的条目形如 <Personal>MyAppData</Personal>，此处“Personal”是 Environment.SpecialFolder 枚举的值，表示“My Documents”目录。任何 Environment.SpecialFolder 中的枚举值都可以用于在配置文件中定义应用程序数据路径。

查找缩写文件

可以定制 Abbreviation 文件以匹配用户数据，使查找操作更有效率。有关详细信息，请参阅第 263 页第 14 章中的微调查找过程。

G

理解 MapInfo 工作空间

MapXtreme 2005 提供了工作空间格式，可以移植和互操作并使用 MapInfo codespace 定义。本附录介绍工作空间的定义、功能和使用。

本附录内容：

- ◆ 什么是 *MapInfo* 工作空间? 460
- ◆ 工作空间的结构 461

什么是 MapInfo 工作空间？

使用 MapXtreme 2005，用户可以将创建的地图作为基于 XML 的工作空间保存（使用 .MWS 扩展名），工作空间可以移植和互操作。然后可以和其他任何使用 MapInfo 产品的用户共享这些地图，无论他们处于何种工作环境。

如果用户熟悉 MapXtreme Java、MapX 或 MapInfo Professional，则会了解到目前用户使用不同的文件格式保存地图：

- MapXtreme Java 地图保存为 Map Definition Files (.MDF 文件)
- MapX 地图保存为 geosets (.GST 文件)
- MapInfo Professional 地图保存为工作空间 (.WOR 文件)

上述文件已成为独立的 MapInfo 产品用户共享地图的方式。MapInfo Professional 工作空间也可以包含浏览器的设置并布局窗口、图形、图例甚至打印机的设置。

用户需要单一的永久性文件格式，可以被所有 MapInfo 产品共享。未来所有的 MapInfo 产品都将支持并使用 MapXtreme 2005 中基于 XML 的新型 MapInfo 工作空间。XML 提供了用户所需的可移植和互操作性，以实现本地和网络的工作空间共享。

在 MapXtreme 2005 中，用户可以创建命名连接以定义共享新工作空间的首选路径。命名连接使用户可以指定可选的驱动器和路径信息，或共享设备上数据的特定数据库连接。在不同计算机上工作的用户可以建立上述基于他们自身环境的命名连接路径，以使用其他用户创建的工作空间。路径可以是绝对路径或相对路径，但不包括文件名。命名连接可以建立到 URI、WMS、本地连接和 RDBMS（使用 ODBC、OCI 或 JDBC）的路径。此后，用户获得包含命名连接路径的工作空间时，可以定义正确的路径以定位工作空间中引用的数据。

MXP_MapDataSource_1_0.xsd 是完全文档化的 XML 构架文件，指定用于定义命名连接的 XML 文件的格式。定义工作空间和命名连接的 XML 构架 (.XSD) 随 MapXtreme 2005 一起提供。构架文件中的备注有助于使用和理解工作空间的结构。虽然可以使用任意文本查看工具查看这些文件（例如记事本），不过，本公司推荐使用更稳定的 XML 查看和编辑工具。

本公司的 XML 文档使用文本标识符，并且尽可能的使用了 XML 标准化组织定义的标识符。不过，XML 是可扩展的，任何文档作者都可以添加标识符以阐明文档使用的数据的含义。为确保这些标识符按照 MapInfo Corporation 定义的规范准确标注，本公司在名为 MapInfo Codespace 的文档中对它们作了详细说明。

关于创建工作空间和使用 Workspace 管理器的详细信息，请参阅第 19 章：*Workspace 管理器*。

注： 不能通过 MapXtreme 2005 使用在 MapInfo Professional 中创建的工作空间。
MapXtreme 2005 可以读取 MapX geosets。

工作空间的结构

要理解工作空间的结构，可以通过一个具体的工作空间及其相关的 XML 代码。下面将以 MapXtreme 2005 安装路径的 Sample 文件夹下的 World.MWS 工作空间作为示例。下文描述了工作空间中五种数据类型中的四种。

1. Header 包含了基本文件信息，包括版本类型和创建日期。
2. Connection 部分定义了包含在工作空间中的独立文件的位置。这里是命名连接信息存储的位置。
3. DataSourceDefinition 部分列出了工作空间中数据的 TAB 文件源。
4. MapDefinition 部分包含了和工作空间关联的图层、主题和标注图元的定义，例如标注的详细信息、缩放范围、使用的颜色等等。
5. UserData 部分使应用程序可以永久化自身的数据。用户数据是“通配符”元素，因为构架不验证它，所以可以输入任意需要的内容。

Header 部分

文件以必需的 XML 文件头信息开始，该文件头信息描述了文件的 XML 版本，编码描述和指示文件是否独立的字段。文件中的最高级元素是 Workspace Element，包含文件版本、日期和 XML 命名空间定义的属性。

注： MapXtreme 2005 工作空间永久性支持的编码方式是 UTF-8。

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<WorkSpace version="MXP_WorkSpace_1_0" date="2004-02-03T14:00:00"
  xmlns:gml="http://www.opengis.net/gml" xmlns="http://www.mapinfo.com/
  mpx">
```

Connection 部分

ConnectionString 部分定义了使用工作空间必需的数据库连接、文件连接和 WMS 连接。目前支持的数据库连接包括 MS Windows ODBC 连接、Oracle C 接口 (OCI) 连接和 JDBC 驱动程序连接。文件连接标识文件的路径。以下是这些连接的一些示例。这些连接不在示例工作空间中出现，而是用作数据库和文件连接的示例。

```
<ConnectionString>
  <FileConnection dbType="file">
    <ConnectionName>mts_data_japanese</ConnectionName>
    <FilePath>\servername\DIR\AP\DATA\TestScripts\Japanese
    </FilePath>
  </FileConnection>
  <DBConnection dbType="xy">
    <ConnectionName>Project_xy_ids_informix_9.3</ConnectionName>
```

```
<ODBCConnectionString>Driver={INFORMIX 3.81 32 BIT};  
    UID=privateuser;PWD=privateuser;DATABASE=xy;  
    HOST=HOST;SRVR=Project_ds;SERV=1800;PRO=onsoctcp;  
    CLOC=en_US.CP1252;DLOC=en_US.CP1252;VMB=0;CURB=0;  
    OPT=;SCUR=0;ICUR=0;OAC=1;OPTOFC=0;RKC=0;ODTYP=0;DDFP=0;  
    DNL=0;RCWC=0  
</ODBCConnectionString>  
</DBConnection>  
<DBConnection dbType="oracle">  
    <ConnectionName>Server_japanese_oracle_9i_release_2  
    </ConnectionName>  
    <OCICConnectionString>SRVR=INTL;UID=USER;PWD=USER;SCROLL=NO  
    </OCICConnectionString>  
</DBConnection>  
<DBConnection dbType="oracle">  
    <ConnectionName>SERVER_oracle_9i_release_1</ConnectionName>  
    <OCICConnectionString>SRVR=SERVER;UID=USER;PWD=USER  
    </OCICConnectionString>  
</DBConnection>  
<DBConnection dbType="sqlserver">  
    <ConnectionName>server_sqlserver2000_server</ConnectionName>  
    <ODBCConnectionString>DRIVER={SQL Server};  
        DATABASE=QADB;Server=Server;UID=User;PWD=user;  
        QuotedID=No;Trusted_Connection=No;Network=DBMSN;  
        Address=Server,1433</ODBCConnectionString>  
</DBConnection>  
<DBConnection dbType="informix">  
    <ConnectionName>spyro_ius_9.2_hpux</ConnectionName>  
    <ODBCConnectionString>Driver={INFORMIX 3.81 32 BIT};  
        UID=user;PWD=user;DATABASE=thedb;HOST=host;  
        SRVR=server_ds;SERV=1800;PRO=onsoctcp;CLOC=en_US.CP1252;  
        DLOC=en_US.CP1252;VMB=0;CURB=0;OPT=;SCUR=0;ICUR=0;  
        OAC=1;OPTOFC=0;RKC=0;ODTYP=0;DDFP=0;DNL=0;  
        RCWC=0</ODBCConnectionString>  
</DBConnection>  
</ConnectionSet>
```

DataSource Definition 部分

DataSource Definition 部分定义了在每个连接位置检索的数据文件（例如， TAB 文件）和数据库表。它也可以包括可选的元数据以描述要检索的数据。

```
<DataSourceDefinitionSet>  
    <DBDataSourceDefinition id="id1" readOnly="false">  
        <DataSourceName>alias_Line_ontario_oracle_9i_rel  
        </DataSourceName>  
        <ConnectionMember>  
            <ConnectionName>ontario_oracle_9i_release_1</ConnectionName>
```

```
</ConnectionMember>
<DBQuery>
    <Query>select * from us_hiway_extra</Query>
</DBQuery>
<DBDataSourceMetadata>
    <FeatureGeometryAttribute srsName="mapinfo:coordsys 1,62">
        OBJECT
    </FeatureGeometryAttribute>
    <KeyAttributes />
</DBDataSourceMetadata>
</DBDataSourceDefinition>
<TABFileDataSourceDefinition id="id2" readOnly="false">
    <DataSourceName>alias_Line_us_hiway_extra</DataSourceName>
    <FileName>FeatureLayerBuilder\us_hiway_extra.tab</FileName>
</TABFileDataSourceDefinition>
<DBDataSourceDefinition id="id3" readOnly="false">
    <DataSourceName>alias_Point_ontario_oracle_9i_re
    </DataSourceName>
    <ConnectionMember>
        <ConnectionName>ontario_oracle_9i_release_1</ConnectionName>
    </ConnectionMember>
    <DBQuery>
        <Query>select * from usa_caps_extra</Query>
    </DBQuery>
    <DBDataSourceMetadata>
        <FeatureGeometryAttribute srsName="mapinfo:coordsys 1,62">
            OBJECT
        </FeatureGeometryAttribute>
        <KeyAttributes />
    </DBDataSourceMetadata>
</DBDataSourceDefinition>
<TABFileDataSourceDefinition id="id4" readOnly="false">
    <DataSourceName>alias_Point_usa_caps_extra</DataSourceName>
    <FileName>FeatureLayerBuilder\usa_caps_extra.tab</FileName>
</TABFileDataSourceDefinition>
<DBDataSourceDefinition id="id5" readOnly="false">
    <DataSourceName>alias_Region_ontario_oracle_9i_r
    </DataSourceName>
    <ConnectionMember>
        <ConnectionName>ontario_oracle_9i_release_1</ConnectionName>
    </ConnectionMember>
    <DBQuery>
        <Query>select * from usa_extra</Query>
    </DBQuery>
    <DBDataSourceMetadata>
        <FeatureGeometryAttribute srsName="mapinfo:coordsys 1,104">
            OBJECT
        </FeatureGeometryAttribute>
        <KeyAttributes />
    </DBDataSourceMetadata>
```

```
</DBDataSourceMetadata>
</DBDataSourceDefinition>
<TABFileDataSourceDefinition id="id6" readOnly="false">
    <DataSourceName>alias_Region_usa_extra</DataSourceName>
    <FileName>FeatureLayerBuilder\usa_extra.tab</FileName>
</TABFileDataSourceDefinition>
</DataSourceDefinitionSet> ...
```

Map Definition 部分

Map Definition 部分定义一张或多张地图和组成它们的图层。每张地图都具有显示条件，包括图像大小、缩放和中心设置、渲染地图的坐标系。

```
<MapDefinitionSet>
    <MapDefinition id="id7" name="tab_map_632140027761992151"
        alias="alias_tab_map_632140027761992151">
        <DisplayConditions>
            <MapSize uom="mapinfo:imagesize pixel">
                <ImageWidth>600</ImageWidth>
                <ImageHeight>400</ImageHeight>
            </MapSize>
            <ZoomAndCenter>
                <MapZoom uom="mapinfo:length mi">3000</MapZoom>
                <gml:Point srsName="mapinfo:coordsys 1,62">
                    <gml:coordinates>-79.771366,38.003251000000006
                    </gml:coordinates>
                </gml:Point>
            </ZoomAndCenter>
            <DisplayCoordSys>
                <SRSName>mapinfo:coordsys 1,62</SRSName>
            </DisplayCoordSys>
            <ResizeMethod>preserveZoom</ResizeMethod>
            <RasterConditions dither="halfTone" trueColor="true">
                optimizeRasterLayers="screen"
                optimizeVectorLayers="screen" />
        </DisplayConditions>
        <LayerList>
            <FeatureLayer id="id8" name="Tab_Point_632140027762460886"
                alias="alias_Tab_Point_632140027762460886"
                volatile="unknown">
                <DataSourceRef ref="id4" />
            </FeatureLayer>
            <FeatureLayer id="id9" name="Tab_Line_632140027764492071"
                alias="alias_Tab_Line_632140027764492071"
                volatile="unknown">
                <DataSourceRef ref="id2" />
            </FeatureLayer>
        </LayerList>
    </MapDefinition>
</MapDefinitionSet>
```

```
<FeatureLayer id="id10" name="Tab_Region_632140027764960806"
    alias="alias_Tab_Region_632140027764960806"
    volatile="unknown">
    <DataSourceRef ref="id6" />
</FeatureLayer>
</LayerList>
<MapLegendSet />
</MapDefinition>
<MapDefinition id="id11"
    name="ontario_oracle_9i_release_1_632140027766054521"
    alias="alias_ontario_oracle_9i_release_1_632140027766054521">
    <DisplayConditions>
        <MapSize uom="mapinfo:imagesize pixel">
            <ImageWidth>600</ImageWidth>
            <ImageHeight>400</ImageHeight>
        </MapSize>
        <ZoomAndCenter>
            <MapZoom uom="mapinfo:length mi">3000</MapZoom>
            <gml:Point srsName="mapinfo:coordsys 1,62">
                <gml:coordinates>-79.771366,38.0032510000000006
                </gml:coordinates>
            </gml:Point>
        </ZoomAndCenter>
        <DisplayCoordSys>
            <SRSName>mapinfo:coordsys 1,62</SRSName>
        </DisplayCoordSys>
        <ResizeMethod>preserveZoom</ResizeMethod>
        <RasterConditions dither="halfTone" trueColor="true"
            optimizeRasterLayers="screen"
            optimizeVectorLayers="screen" />
    </DisplayConditions>
    <LayerList>
        <FeatureLayer id="id12"
            name="oracle_usa_caps_extra_632140027766054521"
            alias="alias_oracle_usa_caps_extra_632140027766054521"
            volatile="unknown">
            <DataSourceRef ref="id3" />
        </FeatureLayer>
        <FeatureLayer id="id13"
            name="oracle_us_hiway_extra_632140027768398196"
            alias="alias_oracle_us_hiway_extra_632140027768398196"
            volatile="unknown">
            <DataSourceRef ref="id1" />
        </FeatureLayer>
        <FeatureLayer id="id14"
            name="oracle_usa_extra_632140027768866931"
            alias="alias_oracle_usa_extra_632140027768866931"
            volatile="unknown">
            <DataSourceRef ref="id5" />
        </FeatureLayer>
    </LayerList>
</MapDefinition>
```

```
</FeatureLayer>
</LayerList>
<MapLegendSet />
</MapDefinition>
</MapDefinitionSet>
```

该图层列表显示了为该地图设置的标注属性。这些属性包括标注所在的图层、标注的文件源、标注的可视性规则（包括标注可视的范围，如果适用）、标注的字体和大小属性、标注字符限制和标注线（如果适用）、标注位置设置和对齐方式、标注的原点（关于标注的点）。

通过编程方式创建工作空间

在 MapX 中，曾经使用 `Map.SaveMapAsGeoset` 方法从现有的地图创建 geoset。用户也可以通过编程方式保存他们的工作空间，以后在 Workspace 管理器中通过使用如下所示的代码片断查看这些工作空间。

注： 用户需要打开一张包含一个或多个图层的地图，以将其作为工作空间保存。

```
// Reads a MapX geoset, writes a MapXtreme 2005 workspace.
using MapInfo.Persistence;
using MapInfo.Mapping;

.

.

MapLoader MapLoader = MapLoader.CreateFromFile("my.gst");
MapExport MapExport = new MapExport();
MapExport.Map.Load(MapLoader);
MapInfo.Persistence wsp = new WorkSpacePersistence();
wsp.Save("c:\\\\temp\\\\newwork.mws");
```

H

从 MapXtreme 应用程序进行打印

本附录将指导您从 MapXtreme 开发项目中打印出最好的地图图像。首先会介绍打印功能概述和一些有用的技巧和方法，然后帮助您解决在使用 MapXtreme 应用程序打印时出现的一些问题。

本附录内容：

- ◆ 概述 468
- ◆ 了解 *MapXtreme* 中的打印选项 469
- ◆ 在应用程序中实现打印功能 472
- ◆ 已知打印问题的解决方案 478

概述

从 MapXtreme 开发的应用程序通常可以直接打印。但是随着打印设备多样性持续扩展，将会产生特定于某些设备的问题。我们为开发人员提供了多种功能以便自定义打印过程。这些选项设计为可以根据将要打印的地图及使用的设备来自定义打印。这些不同设置用于满足多种打印机和绘图仪的需要和它们之间的细微差别。

MapXtreme 支持:

- 打印地图 - 使用 MapPrintDocument 类。
- 打印图例 - 使用 LegendPrintDocument 类。
- 直接打印到设备或通过使用增强元文件 (Enhanced Metafile, EMF) 进行打印。
- 打印不同大小的地图。

MapXtreme 不支持:

- 使用打印选项对话框。
- 打印布局。
 打印地图中的图例时布局是非常重要的。有关如何处理该限制，请参阅示例第 474 页中的打印地图中的图例。
- 打印多页面地图。
 如果地图的幅面为多个页面，则只打印第一页。
- 通过编程方式打印到文件。
 需要使用 Microsoft.NET 1.1 Framework System.Printing.PrinterSettings 类来实现此功能。
- 打印半透明光栅图像。

在 MapXtreme 应用程序中处理打印的类是从 Microsoft.NET 1.1 Framework System.Drawing.PrintDocument 类派生而来并继承了该功能。设备输出控制由 Microsoft.NET 1.1 Framework System.Printing.PrinterSettings 和 System.Printing.PageSettings 类管理。MapInfo.Printing.MapPrintDocument 类专门用于打印地图，MapInfo.Printing.LegendPrintDocument 则专门用于打印图例。

了解 MapXtreme 中的打印选项

从 MapXtreme 应用程序打印地图或图例时，可以向用户提供以下打印选项。为了使用户可以控制这些设置，需要将此功能构建在应用程序的打印对话框中。如果不是特定地允许用户更改这些选项，则使用默认设置。也可以通过编程方式调节这些设置。

打印大小

MapXtreme 提供这些选项用于打印不同大小的地图。要改变地图打印的大小，需要将 MapPrintDocument.MapPrintSize 属性设置为 MapPrintSize 枚举中的一个值。这些值如下所示：

适合于页面

该选项为默认选项，打印的地图保持其纵横比，但缩放至适合于页面。

```
MapPrintDocument.MapPrintSize=MapPrintSize.FitPage
```

填充页面

使用此选项打印的地图填充整个页面。此方法不保持地图原来的纵横比，打印的地图可能会变形。

```
MapPrintDocument.MapPrintSize=MapPrintSize.FillPage
```

当前地图大小

此选项以地图的原始大小打印地图。此方法可能会正合适，但是如果地图比所在页面大，则打印时会丢失超出页面的部分。

```
MapPrintDocument.MapPrintSize=MapPrintSize.MapSize
```

注： 以下属性通用于 MapInfo.Printing.MapPrintDocument 和
MapInfo.Printing.LegendPrintDocument 类。

特殊透明光栅处理

用于将应用程序设置为内部管理透明像素显示和光栅图像的打印。在屏幕上，透明图像被渲染为使用光栅操作 (ROP) 来处理透明像素。打印时此方法可能起作用，也可能不起作用，因为 MapXtreme 使用不同的方法来确定打印的透明方法。必须确定特殊的打印驱动程序是否可以正确处理 ROP，并相应将其设置为 true 或 false。大多数的打印情况下，此值应设置为 true。当绘制到屏幕上时，此值通常设置为 false。使用

DrawingAttributes.SpecialTransparentRasterHandling 来设置此值。此值设置为 true（默认）使应用程序可以内部管理打印过程。

注： 打印矢量图层时此设置不起作用。

特殊透明矢量处理

用于将应用程序设置为打印时内部管理透明矢量填充图案。使用 DrawingAttributes.SpecialTransparentVectorHandling 来启用或禁用此功能。将此功能设置为 true 使 MapXtreme 在打印透明填充图案或透明位图符号时执行特殊的处理。此设置通常用于打印。将此功能设置为 false 让打印设备来处理如何打印透明填充图案和透明位图符号。请参阅第 476 页中的使用填充图案时需要速度吗？详细了解何种填充图案是位图，何种填充图案是矢量图。

尽可能地以真彩色显示光栅

某些打印机不支持 24 位（真彩色）图像。MapXtreme 具有内部功能可以对此进行调节。使用 DrawingAttributes.TrueColorRaster 来启用或禁用此功能。

将此功能设置为 true（默认）来使用 24 位（真彩色）打印光栅和网格图像。如果图像是 24 位，同时打印机支持 256 色（8 位）以上，就可能实现此功能。如果打印机不支持 24 位图像，可将此属性设置为 false，仅使用 256 色用 DitherMethod 中指定的抖动来渲染光栅图像。将此值设置为 false 打印 24 位图像时，打印出来的图像将会严重恶化。将此属性设置为 false 时，同时需要设置抖动方法（参见下文）。

抖动方法

抖动是一种电脑混合像素的技术，当颜色深度减小时可以保持图像的外观。将 24 位图像转换为 256 色时选择一种抖动方法。

- **半色调抖动**计算图像中高对比度元素之间颜色的一系列半色调的差异，以此来创建颜色的平滑过渡。此选项在显示、打印和导出选项中默认选择。
- **误差扩散抖动**计算对比颜色之间的中间色，并将周围像素逐渐均匀混合为中间色。

通过使用 DrawingAttributes.RasterDitherMethod 属性设置此值。可以将此值设置为 DitherMethod 枚举中的一个值。

示例：

```
mapPrintDocument.DrawingAttributes.RasterDitherMethod =  
MapInfo.Mapping.DitherMethod.HalfTone;
```

特殊多边形孔处理

MapXtreme 应用程序可以使用多个单独的块并将它们混合在一起绘制复杂的多边形 (Polygon) 对象。当多边形带有孔或岛状物打印时，这些块可能无法很好地匹配，可能会叠加或留有缝隙。有一些内部编程可以调节此错误行为，以确保这些多边形正常显示和打印。要利用此功能，需要将 `DrawingAttributes.SpecialPolygonHoleHandling` 属性的值设置为 `true`。当绘制到屏幕上时，此值应设置为 `false`。

缩放图案

此设置即可以打印不透明位图填充图案，正如屏幕上所看到的一样，还允许打印机驱动程序额外对渲染该填充图案进行控制。用 `DrawingAttributes.ScaleBitmapPatterns` 设置此值。如果此值为 `true`，则位图填充图案将缩放到可以补偿打印机和屏幕分辨率的差异。此值为 `False` 则不会缩放图案，而是依靠打印机驱动程序来缩放该填充图案。如果打印机驱动程序可以缩放填充图案，则应当将此值设置为 `false`。如果打印机驱动程序无法缩放填充图案，则应当将此值设置为 `true` 以便防止打印输出显示得过小，且显示为接近实心填充图案。请参阅第 476 页中的使用填充图案时需要速度吗？详细了解何种填充图案是位图，何种填充图案是矢量图。

如果此属性设置为 `true`，则打印预览和打印出来的文档看起来可能会不同，因为打印机驱动程序缩放了图案，而显示时并没有缩放图案。即使显示不正常，也可以正常打印。

注： 如果填充图案有背景颜色，则认为图案是不透明的，而没有背景颜色的图案认为是透明填充图案，无论其属性值是什么，都始终被缩放。

直接打印到设备

使用此选项可以将图像直接打印到打印机。用 `PrintMethod.Direct` 设置此值。

使用增强元文件 (EMF) 打印

使用此选项生成图像的增强元文件，然后将其发送到打印机。此选项利用当前打印机技术来缩小假脱机大小，并且可以在不牺牲质量的前提下更快地打印文件。用 `PrintMethod.Emf` 设置此值。

在应用程序中实现打印功能

MapXtreme 2005 中的打印命名空间包含一组类，可以帮助您打印地图。MapXtreme 类从 .NET 框架打印类中派生而来，但是具有 MapXtreme 应用程序打印所需要的更多信息。使用这些类可以向任何连接的打印设备上处理地图的基本打印。

使用 MapPrinting 类来访问对话框和打印机控制对话框。为应用程序分配单独的 MapPrinting 对象，并使用相同的例子来用于打印 / 打印预览 / 页面设置，这一点非常重要。MapXtreme 附带的示例应用程序可以演示这些类以及如何将文本、图形或徽标添加到打印输出中。

设置 MapPrinting 并分配地图：

```
this.mapPrinting = new MapPrinting();
this.mapPrinting.Map = this.mapControl1.Map;
```

下面为打印地图的代码示例：

```
this.mapPrinting.ShowDialog = true;
this.mapPrinting.Print();
```

下面为显示地图打印预览的代码示例：

```
this.mapPrinting.PrintPreview();
```

下面为显示“页面设置”对话框的代码示例：

```
this.mapPrinting.PageSettingsDialog();
```

可以通过打印文档访问打印机设置，打印文档是主要打印类的一部分。因为要向支持打印需要的派生类中添加设置和选项，所以必须从地图打印对象中将该类去除，再重新将其类型转换到派生类中。

```
mapPrinting = new MapPrinting();
mapPrinting.Map = mapControl1.Map;
MapPrintDocument mapPrintDocument = mapPrinting.PrintDocument
as MapPrintDocument;
if (mapPrintDocument != null) {
    // here are some examples of how to set print options
    // Set these based on your needs

mapPrintDocument.DrawingAttributes.SpecialTransparentRasterHandling =
true;
mapPrintDocument.PrintMethod = PrintMethod.Direct;
// and set other properties of mapPrintDocument
}
```

不提供任何更改地图打印选项的 UI。这些对话框仅用于用户更改系统打印机设置。如果想要提供给用户附加选项，则必须提供自己的 UI 来设置 MapPrintDocument（例如 DrawingAttributes、PrintMethod 和 PrintSize）的属性。

在打印文档中还可以访问打印进程，还可以添加打印事件发生时的回调。例如，您也许想要向每个打印的页面上添加徽标或其它图形。

```
mapPrinting = new MapPrinting();
mapPrinting.Map = mapControl1.Map;
mapPrinting.PrintDocument.PrintPage += new
PrintPageEventHandler(mapPrintDocument1_PrintPage);

private void mapPrintDocument1_PrintPage(object sender,
PrintPageEventArgs e)
{
    // add customization for each page (ie; title, page #, etc.)
    Graphics g = e.Graphics;
}

mapPrinting = New MapPrinting()
mapPrinting.Map = mapControl1.Map
mapPrinting.PrintDocument.PrintPage += New
PrintPageEventHandler(mapPrintDocument1_PrintPage)

Private Property sender,() As mapPrintDocument1_PrintPage(object
End Property
Private Function e)() As PrintPageEventArgs
    ' add customization for each page (ie; title, page #, etc.)
    Dim g As Graphics = e.Graphics
End Function
```

使用此方法可以访问图形对象，并可以使用任何可用图形程序来打印附加图形或文本。

常规打印技巧与方法

注: MapInfo 不会推荐某种打印机/绘图仪优于其它种类的打印机/绘图仪，也不会特别证明某种打印机可以永久正常运行。有太多种因素可以影响到输出，从而我们无法做出推荐。

用户在打印时碰到了一些特定的常见问题并提供了解决方案，以下技巧和建议就是根据这些信息而得出的。

解决打印故障的良好的第一步是要确保您已经下载和安装了该 MapXtreme 版本的补丁（如果有补丁），并且要确保正使用的打印机驱动程序是您的打印机/操作系统驱动程序的最新版本。此文档中对一些异常已作出说明。

如何重载打印页面事件

.NET 打印 API 提供了一种重载打印页面事件的机制。因此可以添加对每个打印页面的自定义设置。例如，可以添加标题、页码、徽标等。

需要实现 PrintPageEventHandler 才能完成此功能。示例：

```
this.mapPrinting = new MapPrinting();
this.mapPrinting.Map = this.mapControl1.Map;
this.mapPrinting.PrintDocument.PrintPage += new
System.Drawing.Printing.PrintPageEventHandler(this.mapPrintDocument1_P
rintPage);

private void mapPrintDocument1_PrintPage(object sender,
System.Drawing.Printing.PrintPageEventArgs e)
{
    // TODO - add your code here to alter the printed page
}
```

打印地图中的图例

正如本节的开始部分所述，MapXtreme 不支持布局的打印。如果想要打印地图中的图例，则需要将该图例作为修饰添加到地图中。下面是一些代码示例演示此操作如何执行：

```
//create cartographic legend
MapInfo.Mapping.Legends.Legend legend =
mapControl1.Map.Legends.CreateLegend(new System.Drawing.Size(5, 5));
legend.Border = true;
MapInfo.Mapping.LayerType[] normalLyr = new
MapInfo.Mapping.LayerType[1];
normalLyr[0] = MapInfo.Mapping.LayerType.Normal;
MapInfo.Mapping.IMapLayerFilter filter =
MapInfo.Mapping.MapLayerFilterFactory.FilterByLayerType(normalLyr);
MapInfo.Mapping.Legends.LegendFrame frame;
```

```
foreach(MapInfo.Mapping.FeatureLayer ftrLayer in  
mapControl2.Map.Layers.GetMapLayerEnumerator(filter))  
{  
    frame =  
    MapInfo.Mapping.Legends.LegendFrameFactory.CreateCartographicLegendFra  
me(ftrLayer);  
    legend.Frames.Append(frame);  
}  
  
//set legend location on the map  
System.Drawing.Point pt = new System.Drawing.Point(220, 200);  
pt.X = mapControl2.Size.Width - legend.Size.Width;  
pt.Y = mapControl2.Size.Height - legend.Size.Height;  
legend.Location = pt;  
  
//append legend as map adornment  
mapControll1.Map.Adornments.Append(legend);  
  
mapPrinting.print();
```

旧的驱动程序能用，新的驱动程序不能用

有疑问时，如果旧的驱动程序能用而新的不能用，那么请使用旧的驱动程序。

有关 HP 755 驱动程序的建议

如果使用此型号的绘图仪时打印有困难，请尝试用 HP 650C (C2859B) 的绘图仪驱动程序来代替。多数情况下，如果使用某种打印机或绘图仪型号时有问题，还会有另一种相似的型号，您可以用相似型号的驱动程序来代替当前的驱动程序，效果会非常好。例如，您可以将 HP8550 color LaserJet 彩色激光打印机驱动程序用作 HP 8500 DN Color LaserJet 彩色激光打印机的驱动程序。

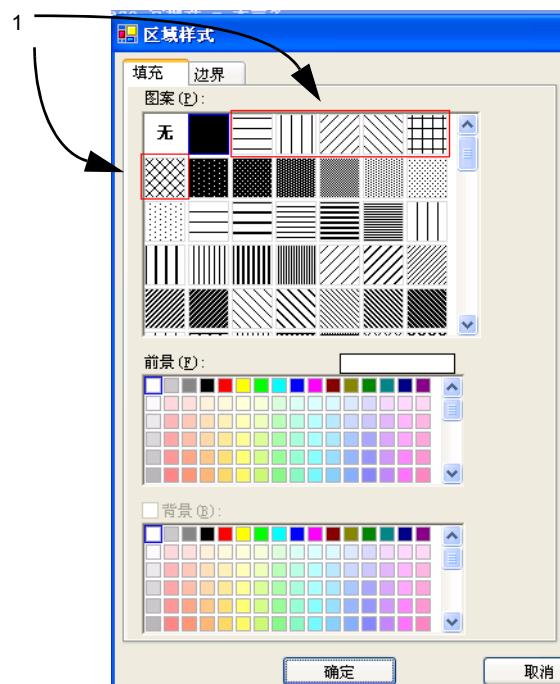
有效缩放图案的建议

打印时，了解您正在使用的打印机驱动程序的类型。许多 PCL6 和某些 HPGL 驱动程序可以进行填充图案的缩放，并提供此功能的控制。关闭其缩放可能会造成打印结果和监视器上显示的有所不同。建议尝试关闭驱动程序的缩放选项而先试用我们提供的缩放功能，因为我们已经增强了更好地打印地图和图例的方法。打开我们的图案缩放选项，启用应用程序中的“缩放图案”（请参阅第 471 页中的缩放图案）。再将缩放功能关闭，看一看哪一种效果更好。测试显示使用我们的缩放功能所产生的颜色输出与屏幕上的显示更接近。

我们发现，如果使用 LanguageLevel 2 或 3 打印到 PostScript 驱动程序，某些 Microsoft 驱动程序不支持图案缩放。那么我们的缩放方法可能就起不到作用了。Microsoft 建议将 PostScript 驱动程序的语言级别重新设置到 LanguageLevel 1 来除去此限制。此情况下出现过异常。在 Windows 2000 和 Windows NT 上，使用我们的缩放选项，一些 HP Laser Jet 激光打印机和 Color Laser Jet 彩色激光打印机的 PostScript 驱动程序可以正常打印。

使用填充图案时需要速度吗?

请注意 AreaStyle 对话框中的前六项填充图案（实心填充之后）是 Windows 标准，有助于更快速地打印。这些填充图案是基于矢量的。MapXtreme 附带的其它图案是位图。选择填充图案时可能需要考虑这些。



1 使用这些填充图案以实现快速打印。

速度仍然不理想吗?

如果想要改进打印机速度，而您的打印机有**快速**、**普通**和**最佳** 三种打印质量选项，建议您选择**快速**。但是这样也会降低输出的分辨率。

磁盘空间不足吗?

确保有足够的临时磁盘空间，特别是在使用“使用增强元文件打印”选项时。系统会尝试在磁盘上本地创建一个带图层的位图。

全局打印？本地假脱机！

尝试在本地假脱机¹打印作业，而不是在绘图仪上。这样计算机可以将输出光栅化，而不是由打印机来做此项工作，这样会更有效。

1. 假脱机是存储数据的过程，在内存中或文件中组成将被打印的文档，直到打印机准备好对其进行处理。



1 单击假脱机打印文档和最后一页假脱机后开始打印按钮。

1. 要设置本地假脱机打印作业, 请选取**开始 > 设置 > 控制面板 > 打印机**。
2. 右键单击打印机并在菜单中选取**属性**, 显示打印机的属性。
3. 单击**高级**选项卡, 显示高级属性选项。

注: 如果您没有计算机的管理员权限, 可能无法使用假脱机选项。如果想要进行更改, 请与 IT 部门联系以获取他们的支持。

4. 单击**确定**保存所作更改。

已知打印问题的解决方案

有许多种因素影响应用程序的打印和绘图，MapXtreme 应用程序也不例外。MapXtreme 不提供打印机驱动程序，而是使用当前 Windows 操作系统中现有的驱动程序。本节介绍用户及合作伙伴没有涉及的特定的打印机/绘图仪问题。我们提供这些信息的目的是支持您，使得您可以支持您的用户。

平台独立问题

如果没有特别注明，任何操作系统或硬件/软件都会发生平台独立问题。在下文中查找您正要解决的问题，并参阅解决方案。

阴影线图案打印为黑色多边形

打印机 / 绘图仪：HP Designjet 系列绘图仪

问题：打印带阴影线图案的地图时，一些区域显示为黑色实心。

分辨率：依据 Hewlett Packard 支持部门，新的 HP 打印机驱动程序 (4.63) 可以正确处理非 Windows 标准的阴影图案。我们建议您下载并安装新的 HP 4.63 驱动程序来解决此问题。

打印时阴影图案压缩了

操作系统：WindowsNT 4.0/2000

打印机 / 绘图仪：HP LaserJet 系列，用 HP LaserJet 4050 系列 PCL 6 驱动程序测试

问题：使用 MapInfo 阴影图案（非前六种图案 AreaStyle 对话框的其中一种）时，可能会注意到该图案大大地压缩了。这是因为这些图案是位图，是以较高的打印机分辨率绘制而成的。标准 Windows 图案不会出现此问题。打印机分辨率会影响压缩量。

分辨率：一些打印机驱动程序具有一个新的输出设置称为“缩放图案 (WYSIWYG)”，使图案可以按正确的比例缩放。在测试的 HP 打印机中，可通过选择**打印首选项 > 打印质量 > 详细情况**选项来使用此选项。

用户应当关闭驱动程序的缩放功能或关闭我们的缩放功能，因为两个缩放同时工作会将图案填充两次。尝试两种缩放类型，选出更好用的一种。

特定平台问题

如果没有特别注明，这些问题出现在特殊的操作系统和 / 或硬件。这些问题根据操作系统分组。

打印光栅时，如果使用 HP 500, 800, 5000 DesignJet 绘图仪，每 2-4 厘米将重复打印输出

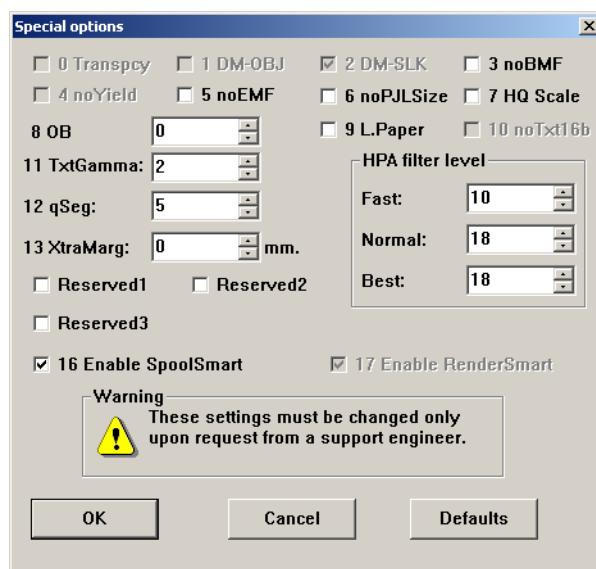
有两种建议可用于解决此问题。

注： 必须具备打印机和打印机驱动程序的读 / 写权限才能解决此问题。

要解决此问题，可以在驱动程序属性中更改一个高级隐藏设置。

要避免在打印矢量覆盖的光栅图时打印机每 2-4 厘米重复自己：

1. 在**控制面板**中确定打印机驱动程序的属性。要获取此设置，右键单击打印机图标并选取**属性**。
2. 单击**高级**选项卡。选择以下选项：
 - 选择**避免内存不足**选项。
 - 如果使用 Windows 2000 或 XP 操作系统，则请选择缩放因子为 100%。
3. 单击此对话框中的**打印默认值**按钮。
4. 在此对话框中，选择**关于**按钮。关于 < 驱动程序名称 > 对话框显示。
5. 按下键盘上的 F8 键，单击**确定**按钮。“Special options”对话框显示。



6. 清除**ENABLE SPOOLSMART**复选框，然后单击**OK**：

注: 修改此驱动程序选项框中的设置时,一定要留意原始设置,以备在需要时改回原始设置。这些是专门为 HP 支持工程师设计的,普通公众一般不使用。

可以将这些设置保存为“快速设置”,以确保每次打印光栅文件时都起作用。

1. 浏览至 HP 打印机驱动程序,访问驱动程序首选项。使用这些方法中适合您的操作系统的一种:
 - 使用 **Windows NT 4.0** - 从 Windows 桌面选择**开始 > 设置 > 打印机**。然后,右键单击合适的绘图仪驱动程序图标,并选取**文档默认值**。
 - 使用 **Windows 2000/XP (经典视图)** - 从 Windows 桌面选择**开始 > 设置 > 打印机**。然后,右键单击合适的绘图仪驱动程序图标,并选取**打印首选项**。
2. 在隐藏菜单中按照您所想要的方式设置首选项。
3. 在**快速设置**框中,键入所选设置的名称(例如,“光栅打印设置”),并单击**保存**。当前所有的驱动程序设置(纸张类型、隐藏菜单设置等)都保存在**快速设置**名称下。打印机驱动程序会记住这些设置,可以用于将来的打印作业。

要在稍后以相同设置打印光栅地图:

1. 选取“打印”命令来显示“打印”对话框。
2. 选取**属性**按钮。“属性”对话框显示。
3. 确保输入的名称(如示例中的“光栅打印设置”)显示在**快速设置**下拉列表中,然后单击**确定**。“打印”对话框重新显示。
4. 单击**确定**以打印。

注: 如果“快速设置”中不需要此特殊设置,请记住要将此设置改回到默认机器设置。

不能使用 HP Designjet 打印机打印 (驱动程序 5.31 或 5.32)。

当尝试使用这些驱动程序时会发生以下一种情况:

- 如果选择**最高性能**选项,则会显示内存出错信息,打印机仅打印部分图像。
- 如果选择**避免内存不足**选择,则计算机会崩溃,并显示“硬盘空间不足”和“虚拟内存不足”信息。

此问题的解决方案与前述问题的解决方案相似。“特殊选项”对话框为同一个对话框。

1. 编程设置以下打印选项:

```
MapPrintDocument mapPrintDocument = this.mapPrinting.PrintDocument
as MapPrintDocument;
mapPrintDocument.PrintMethod = PrintMethod.Direct;
mapPrintDocument.DrawingAttributes.SpecialTransparentVectorHandle
ing = true;
mapPrintDocument.DrawingAttributes.
SpecialTransparentRasterHandling = false;
mapPrintDocument.DrawingAttributes.TrueColorRaster = true;.
```

2. 从“控制面板”选择“打印机”或“打印机和传真”，找出您用于打印的打印机。
3. 右键单击该打印机并选择“属性”选项。“属性”对话框显示。
4. 单击**关于**按钮来显示关于<打印机驱动程序>对话框。
5. 按下键盘上的 F8 键，单击**确定**按钮。“特殊选项”对话框显示。
6. 取消选取“启用 SpoolSmart”复选框。

样式查找

本附录包含支持样式的查找表，包括填充图案、直线样式、矢量符号和自定义位图符号。有关样式说明和如何使用的详细信息，请参阅第 15 章：为地图设置样式。

本附录内容：

- ◆ 填充图案，第 484 页
- ◆ 直线样式，第 494 页
- ◆ 矢量符号，第 495 页
- ◆ 定制符号，第 499 页

填充图案

下表概述了 MapXtreme 填充图案（也称为 MapXtreme 编程 API 的内部样式）。每个填充图案都有一个关联的“索引编号”（用于对 InteriorStyleRepository 对象的可编程访问）和一个“图案值”（填充图案的内部描述名）。

具体如下：

填充图案

自身的图形填充图案。

索引编号

基于 0 的索引，用于检索表示 InteriorStyleRepository 中填充图案的内部样式。

图案值

填充图案的数字标识符，可用于构造内部样式对象。这只是一个内部值，不能通过编程方式访问 InteriorStyleRepository。不过，这些图案值可用于指示 MapXtreme 2005 工作空间文件 (.mws) 中的填充图案。

了解索引编号结构

如“填充图案”表中所见，填充图案索引（列于“索引编号”列）的范围从 1 至 172。基于 1 的索引结构用于其它所有 MapInfo 产品（如 MapInfo Professional 和 MapBasic）。不过，在 MapXtreme API 中，填充图案（也称为内部样式对象）通过 InteriorStyleRepository 对象（该对象使用基于 0 的索引）获得。

特别是，InteriorStyleRepository 对象允许用户通过基于 0 的索引结构（范围从 0 至 172）在所有可用内部样式对象上重复。因此，实际上有 173 个内部样式对象可供选择，不过，基于 0 和 1 的内部样式对象是等价的。基于以下原因，需要位于元素 0 和 1 的等价对象：

首先，我们必须提供 InteriorStyleRepository 类自身的基于 0 的内部索引功能。即，InteriorStyleRepository 类只允许基于 0 的索引访问，用于从其 IList 检索元素。（IList 是 InteriorStyleRepository 类实现的标准 .NET 接口。）

其次，基于 0 的索引结构必须与“图案值”列中显示的基于 1 的索引结构保持同步。因此，只有 172 个独立的填充图案，其索引范围从 1 至 172。已添加位于索引 0 的可编程填充图案，以给出该索引元素的定义。

实际上，客户将从工作空间文件中特有的一句话得到启发。

注: 您可能会注意到前八个填充图案都具有相同的“索引编号”和“图案值”。随后，“图案值”通常比对应的“索引编号”大 3 (因此，“图案值”序列达到 175，而“索引编号”以 172 结束)。这不是出错。仍然有 172 个惟一的填充图案。表中未丢失任何内容。

与索引和图案值关联的填充图案

填充图案	索引编号	图案值	填充图案	索引编号	图案值
N (无)	1	1		12	15
	2	2		13	16
	3	3		14	17
	4	4		15	19
	5	5		16	18
	6	6		17	20
	7	7		18	21
	8	8		19	22
	9	12		20	23
	10	13		21	24
	11	14		22	25

与索引和图案值关联的填充图案

填充图案	索引编号	图案值	填充图案	索引编号	图案值
	23	26		35	38
	24	27		36	39
	25	28		37	40
	26	29		38	41
	27	30		39	42
	28	31		40	43
	29	32		41	44
	30	33		42	45
	31	34		43	46
	32	35		44	47
	33	36		45	48
	34	37		46	49

与索引和图案值关联的填充图案

填充图案	索引编号	图案值	填充图案	索引编号	图案值
	47	50		59	62
	48	51		60	63
	49	52		61	64
	50	53		62	65
	51	54		63	66
	52	55		64	67
	53	56		65	68
	54	57		66	69
	55	58		67	70
	56	59		68	71
	57	60		69	72
	58	61		70	73

与索引和图案值关联的填充图案

填充图案	索引编号	图案值	填充图案	索引编号	图案值
	71	74		83	86
	72	75		84	87
	73	76		85	88
	74	77		86	89
	75	78		87	90
	76	79		88	91
	77	80		89	92
	78	81		90	93
	79	82		91	94
	80	83		92	95
	81	84		93	96
	82	85		94	97

与索引和图案值关联的填充图案

填充图案	索引编号	图案值	填充图案	索引编号	图案值
	95	98		107	110
	96	99		108	111
	97	100		109	112
	98	101		110	113
	99	102		111	114
	100	103		112	115
	101	104		113	116
	102	105		114	117
	103	106		115	118
	104	107		116	119
	105	108		117	120
	106	109		118	121

与索引和图案值关联的填充图案

填充图案	索引编号	图案值	填充图案	索引编号	图案值
○ ○	119	122	△ △	130	133
· ·	120	123	▽ ▽	131	134
// //	121	124	+	132	135
\\\\ \\\	122	125	+	133	136
◇ ◇	123	126	▪ ▪	134	137
▲ ▲	124	127	— —	135	138
▣ ▣	125	128	▨	136	139
■ ■	126	129	▽ ▽	137	140
· ·	127	130	V V V V V V V V V V V V	138	141
— —	128	131	● ● ● ● ● ● ● ●	139	142
▽ — — ▽	129	132	× ×	140	143

与索引和图案值关联的填充图案

填充图案	索引编号	图案值	填充图案	索引编号	图案值
	141	144		153	156
	142	145		154	157
	143	146		155	158
	144	147		156	159
	145	148		157	160
	146	149		158	161
	147	150		159	162
	148	151		160	163
	149	152		161	164
	150	153		162	165
	151	154		163	166
	152	155		164	167

与索引和图案值关联的填充图案

填充图案	索引编号	图案值	填充图案	索引编号	图案值
	165	168		169	172
	166	169		170	173
	167	170		171	174
	168	171		172	175

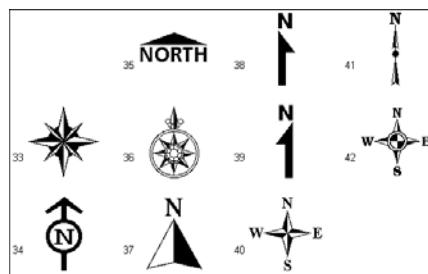
直线样式

0	24	48	72	96	
1	25	49	73	97	
2	26	50	74	98	
3	27	51	75	99	
4	28	52	76	100	
5	29	53	77	101	
6	30	54	78	102	
7	31	55	79	103	
8	32	56	80	104	
9	33	57	81	105	
10	34	58	82	106	
11	35	59	83	107	
12	36	60	84	108	
13	37	61	85	109	
14	38	62	86	110	
15	39	63	87	111	
16	40	64	88	112	
17	41	65	89	113	
18	42	66	90	114	
19	43	67	91	115	
20	44	68	92	116	
21	45	69	93	117	
22	46	70	94	118	
23	47	71	95		

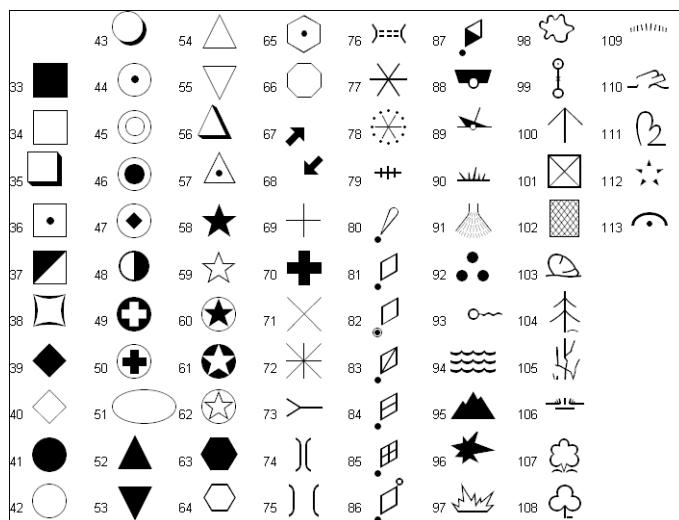
矢量符号

MapXtreme 2005 在其安装过程中自动安装 10 种 MapInfo 特定的 TrueType 字体。这些字体为用户提供了字形符号选择，范围涉及天气、房地产、交通等。

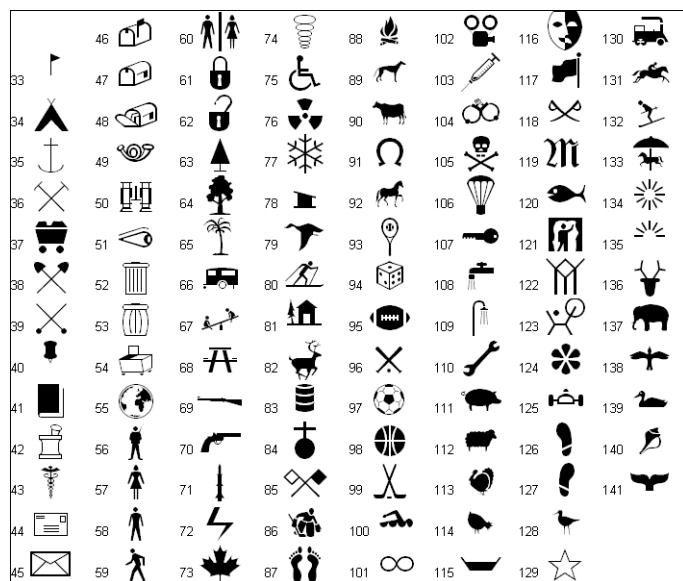
MapInfo 箭头



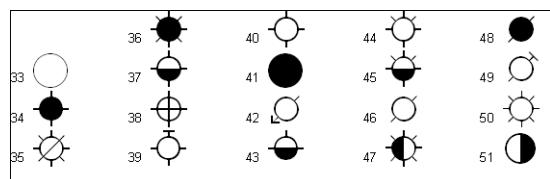
MapInfo 制图



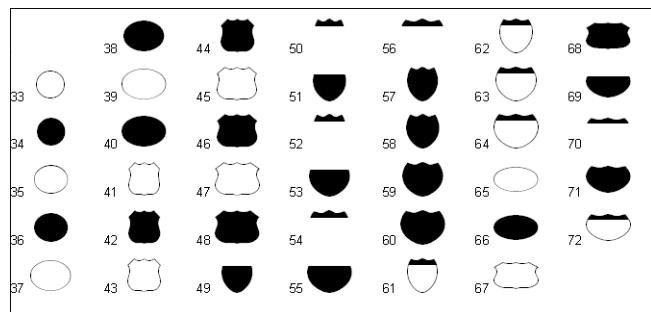
MapInfo 杂类



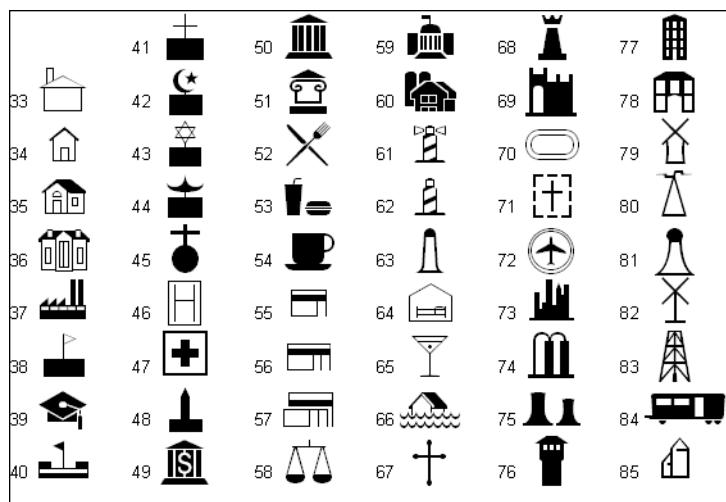
MapInfo 油气



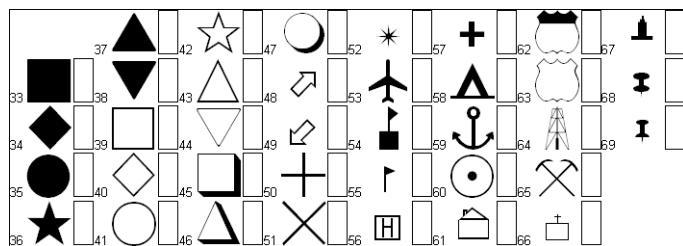
MapInfo 遮罩



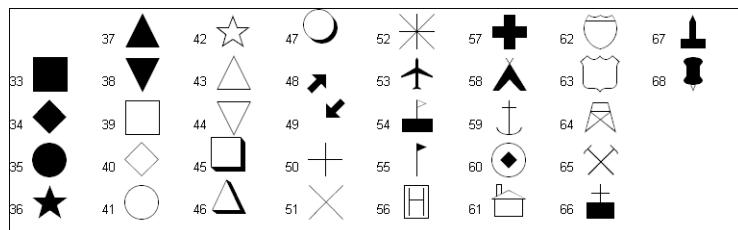
MapInfo 房地产



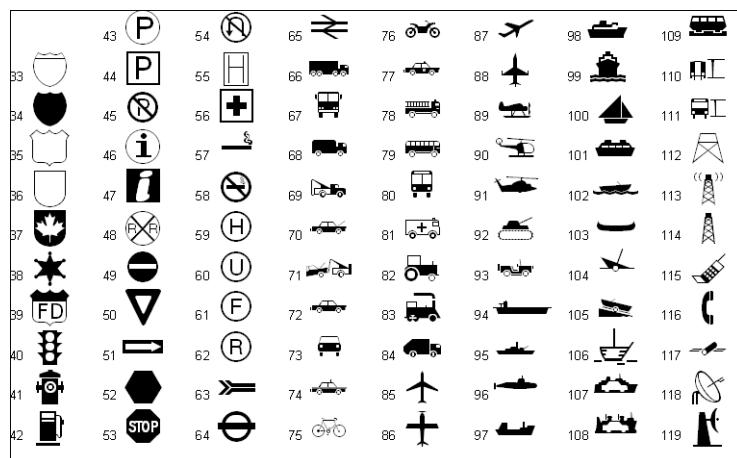
地图符号



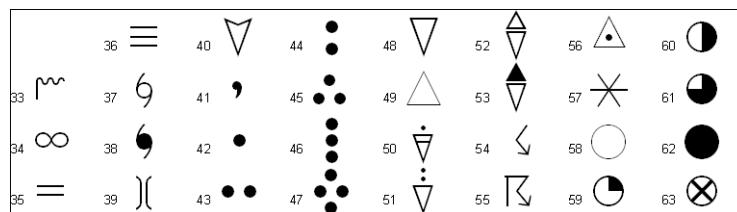
MapInfo 符号



MapInfo 交通



MapInfo 天气



定制符号

以下符号位于 C:\Program Files\Common Files\MapInfo\MapXtreme\6.5\CustSymb。每个图像的文件扩展名都是 .BMP。

可以用编程方式通过 MapInfo.Styles 命名空间中的 BitmapPointStyleRepository 集合类访问这些符号。

可以创建自己的位图图像并将其添加到 CustSymb 目录。尽管事实上对创建的图像没有大小限制，不过 MapXtreme 2005 显示图像的能力取决于可用的内存。图像不一定必须是方形，而且还可以具有最多 24 位颜色深度。要确保图像以其高度和宽度显示，则必须在各自图像的 BitmapPointStyle 对象中将 Boolean "NativeSize" 属性设置为 true。



AMBU1-32



AMBU-64



BADG1-32



BADG2-32



BANK1-32



BANK2-32



BANK-64



BOOK1-32



BUIDLINGS



CAMP1-32

CAR1-32



CAUT1-32



CHUR1-32



COMP1-32



FARM1-32



FAST1-32



FIRE1-32



FIRE-64



FOOD-64



GLOB1-32



GOLF1-32



HOSP1-32



HOUS1-32



HOUS2-32



HOUS3-32



HOUS-64



HYDR1-32



INTE1-32



LITE1-32



LITE2-32



MAIL1-32



MBOX1-32



MBOX2-32



MOSQ1-32



ONEW1-32



ONEW2-32



PENC1-32

PIN1-32
(青色)PIN2-32
(红色)PIN1-32
(黄色)PIN4-32
(绿色)PIN5-32
(蓝色)PIN6-32
(紫色)PINB-64
(蓝色)PING-64
(绿色)PINGY-64
(灰色)PINR-64
(红色)

POLI1-32



RAIL1-32



RAIL2-32



RAIL3-32



RAIL-64



RED-CAR



REST1-32



STAT1-32



STOP1-32



SYNA1-32



TARG1-32



TAXI1-32



TEMP1-32



TOWE1-32



TOWE2-32



TRAF1-32



TRUC1-32



TRUC2-32



TRUC-64



YIEL1-32



YIEL2-32

J

MapXtreme 2004 Web 控件

本章讨论 MapInfo.Web.UI.WebControls 命名空间下的 MapXtreme 2004 web 控件和工具。

推荐使用 MapXtreme 2005 中最近重新设计的 web 控件创建新的应用程序。请参阅第 6 章：运用 *MapXtreme 2005 Web 控件*。新控件的 API 位于 MapInfo.WebControls。

本章内容：

- ◆ *Web.UI.WebControls* 命名空间概述 502
- ◆ *Web* 控件 502
- ◆ 使用 *Web* 控件 504
- ◆ 在用户控件中使用 *MapControl* 507
- ◆ *Web* 工具体系结构 508

Web.UI.WebControls 命名空间概述

MapInfo.Web.UI.WebControls 命名空间包括用户界面工具和 Web 控件，用于设计和使用由 MapXtreme 2004 构建的 Web 窗体应用程序。可方便地在 Web 窗体上拖放 Web 控件。这些工具在 MapControl 中提供了服务器端和客户端行为。这些控件和工具包括了各种功能：地图显示和导航、图元选择以及图层管理。

Web 控件

Web 控件在 Microsoft Visual Studio .NET 中提供交互式（拖放）开发环境。使用这些控件可以创建简单的地图绘制应用程序，而不必编写大量代码。

Web 控件包括在 MapInfo.Web.UI.WebControls 命名空间中。它们包括：MapControl、Toolbar、ZoomInControl、ZoomOutControl、CenterControl、PanControl、DistanceControl、LayerControl、InfoToolControl 和 LegendControl。

这些控件位于“MapXtreme Web 控件”选项卡下面的 Visual Studio 工具箱中。

Web 控件	目的
MapControl	显示地图图像并允许基于选定的工具进行鼠标交互。
ToolbarControl	提供用于一个 / 多个地图工具的面板。
ZoomInToolControl	用于放大地图或图层的区域视图。
ZoomOutToolControl	用于缩小地图视图。
CenterToolControl	用于将地图的中心设置到单击的点上。
PanToolControl	用于通过拖动地图，在窗口中重新定位地图。
Selections	鼠标单击操作在某位置上选择图元。可以使用以下功能： <ul style="list-style-type: none">• PointSelectionToolControl• RadiusSelectionToolControl• PolygonSelectionToolControl• RadiusSelectionToolControl
InfoTool	用于获得单击点的属性信息。

Web 控件	目的
DistanceTool	用于获得两个点或多个点之间的距离。
LayerControl	允许在地图图层中添加 / 删除 / 编辑属性。
LegendControl	允许在窗体上查看 Legend。

Web 控件类型

有两种类型的 Web 控件可用于开发应用程序：服务器控件和客户端 HTML 控件。

服务器端控件

通过使用服务器控件，可以在服务器端处理大部分的绘制和事件。

某些服务器控件（如下拉列表框或文本框）具有称为 autopostback 的属性。在发生选择更改或文本更改事件时，autopostback 允许将控件转到服务器。

注：要提高 Web 应用程序性能，可考虑将此属性更改为 False（并且使用提交按钮一次提交整个页面，而不是服务器控件的每个事件）。

HTML 控件

HTML 控件是 HTML 页面上由浏览器绘制的元素。通过使用这些控件，可在客户端通过脚本访问对象模型。

注：可使用 HTML 控件代替服务器控件来提高性能。有关更多性能调整的技巧，请访问以下 Web 站点：<http://samples.gotdotnet.com/quickstart/aspplus/doc/perftuning.aspx>

使用 Web 控件

本节描述 MapXtreme 2004 中的几个关键 Web 控件: MapControl、LegendControl 和 LayerControl。

MapControl

MapControl 是包含 Map 对象实例的 Web 控件。通过使用 Map 对象指向的 MapAlias 属性, Map 对象可从 MapFactory 获得。可以通过将地图导出到图像并且该图像使用 IMG html 标记来绘制地图。工具控件的回帖也在该控件中处理。

下面描述 MapControl 在设计阶段和运行时的行为:

设计阶段

在设计阶段可执行下列操作:

1. 使用 LoadMap 动词可从表/geoset/工作空间中加载 MapControl 指向的地图, 或通过更改属性或使用 LayerControl 更改 MapControl 指向的地图。
2. 对地图进行任何更改时, 地图状态存储在与 Web 项目位于同一个文件夹的工作空间文件中。项目关闭和重新打开时会从该工作空间读取和加载地图, 从而将地图恢复到前一个状态。在由窗体名和 MapControl 唯一 ID 限定的项目中, 工作空间文件的名称是唯一的。可将属性 WorkspaceFileName 用于此目的。不建议在运行时设置此属性。
3. 在此过程中所使用的别名在 .aspx 文件中会串行化为 MapAlias 属性。该别名与工作空间文件中的别名相同。但是, 若 MapAlias 属性不匹配, 则可能导致不可预知的行为。在 .aspx 文件的 html 部分更改 MapAlias 属性。
4. 属性 UseDesigntimeWorkspace 用于指定是否在第一次请求页面的运行时加载此工作空间。如果该属性设置为 false, 则将忽略在设计阶段创建的工作空间。

运行阶段

MapControl 在运行时的生命周期如下所示:

1. 在构造函数中，在创建所有默认工具的同时创建工具集合
2. 在 OnInit 中执行下列任务：
 - a. 尝试使用 Map 对象指向的 MapAlias 从 Session 中获取 Map 对象。如果该对象不存在，则创建一个空地图。使用 Map 对象还可创建用于导出的 MapExport 对象。
 - b. 如果是第一次访问此页面并且 UseDesigntimeWorkspace 标记为 True，则将工作空间加载到地图中。
3. 在 OnLoad 中，显示客户端上需要的所有信息，例如 javascript 和工具信息。
4. 在 RenderContents 中，渲染保存 MapControl 的标记和表示图像的 IMG 标记。工具使用的其它信息用于客户端操作。LayerControl 以 javascript 变量的形式渲染。

在 Web 窗体的 Page_Load 方法运行时，可以从默认方法偏离并使用自己的 Map 对象。但是，创建地图的别名必须与 MapControl 的 MapAlias 匹配。在将另一个工作空间加载到地图时，请注意该地图的别名更改为工作空间中的别名。下面是一个示例：

```
private void Page_Load(object sender, System.EventArgs e)
{
    if (!Page.IsPostBack) {
        // Remove the map from session; we want to use ours
        MapInfo.Engine.Session.Current.MapFactory.Remove
            (MapControl1.MapAlias);
        // Create a empty map with same alias as MapControl
        MapInfo.Engine.Session.Current.MapFactory.CreateEmptyMap
            (MapControl1.MapAlias, MapControl1.MapAlias,
             new Size(200,200));
        // Load the map with table.
        MapInfo.Mapping.MapLoader ml =
            MapInfo.Mapping.MapLoader.CreateFromFile(@"World.tab");
        // Get the map object out of session we just created
        MapInfo.Mapping.Map map =
            MapInfo.Engine.Session.Current.MapFactory
                [MapControl1.MapAlias];
        if (map != null) {
            ml.Load(map);
            MapControl1.Map = map;
        }
    }
}
```

LegendControl

LegendControl 用于绘制地图和图例对象的图例。

这个控件与其它控件不同，因为此时它不具有任何设计阶段行为。Map 和 Legend 对象在运行时设置。使用该 LegendControl 的正确方法是：

1. 在设计阶段拖放该控件。将仅显示带有正确信息的矩形。
2. 在 Page_Load 方法中编写代码以设置 Map 和 Legend 对象。

下面是一个示例。

```
private void Page_Load(object sender, System.EventArgs e)
{
    if (!IsPostBack) {
        LegendControl1.Map = MapControl1.Map;
        if (MapControl1.Map.Legends.Count > 0) {
            LegendControl1.Legend = MapControl1.Map.Legends[0];
        }
    }
}
```

LayerControl

LayerControl 允许在相关地图上更改设置。LayerControl 以树视图结构显示 Feature 图层和 Label 图层。这样就可以将这些图层的属性更改为可视的、可选择的和缩放图层。在未按下“应用更改”按钮之前，这些更改不会反映到地图中。所有更改只有在使用 LayerControl 的应用程序处于活动状态时永久，这些更改不能保存，也不能永久存在于工作空间文件中。

要使用 LayerControl，请将控件拖动到已包含地图的设计窗体上。以树视图突出显示图层并按需更改设置。

在用户控件中使用 MapControl

用户控件是一个控件集合。它们在项目中显示为条目并可拖放到窗体上。创建用户控件的目的是可以将其重复用于不同的窗体中。用户控件还具有代码后面的文件，可以在其中编写用户控件的代码。

用户可能希望将 Web 控件包括在用户控件中。以下章节描述如何在用户控件中使用 MapControl。

将 MapControl 拖动到窗体上或用户控制它时，在项目中创建工作空间文件。该工作空间存储对 MapControl 指向的地图所做的全部更改。该工作空间文件名称的格式是：

<Web Form name>_<MapControl.UniqueID>.mws，这使该名称在整个项目中是惟一的。将 MapControl 拖动到用户控件时，如果 MapControl 的惟一 ID 包含冒号（“:”），则可能在工作空间文件的路径名中导致问题。

在用户控件中使用 MapControl 的正确方法是：

1. 将 MapControl 和所有相关的控件拖放到用户控件中。
2. 在“属性”窗口中，将 MapControl 的 UseDesignTimeWorkspace 属性更改为 False。
3. 在用户控件代码文件中，添加首次访问页面时用于加载地图的代码。例如：

```
Page_Load 方法  
'Put user code to initialize the page here  
Dim wl As MapInfo.Mapping.MapWorkSpaceLoader  
  
If Not Page.IsPostBack Then  
    wl = New MapInfo.Mapping.MapWorkSpaceLoader  
        ("c:\data\world.mws")  
    wl.Load(MapControl1.Map)  
End If
```

确保工作空间文件中地图的别名与 MapControl 的别名匹配。

Web 工具体系结构

Web 对象模型中的工具在 MapControl 中提供服务器和客户端行为。每个工具都具有相应的控件。这些控件将在第 502 页中的 *Web 控件*一节中讨论。工具包含要执行行为的逻辑。该控件允许在 Web 窗体中使用工具。

服务器端的工具负责:

- 提取数据
- 执行工具的业务逻辑。

客户端的工具负责:

- 绘制和鼠标操作（例如橡皮圈矩形和鼠标单击）
- 从鼠标操作收集数据
- 为数据创建隐藏字段
- 将页面提交到服务器

初始化时，MapControl 具有包括一套默认地图导航和选择工具的工具集合: ZoomIn、ZoomOut、Pan、Center、Distance、Info、PointSelection、RectangleSelection、RadiusSelection 和 PolygonSelection 工具。要激活其中一个工具，将该工具设置为工具集合的当前工具。

要将服务器上的工具更改为放大工具:

```
MapControl1.MapTools.CurrentTool = ZoomInMapTool.Toolname;
```

要将定制工具添加到工具集合:

```
MapControl1.MapTools.Add(new CustomTool());
```

Web 工具体系结构非常灵活，您可以编写自己的工具，将它们添加到工具集合并通过将其设置为当前工具来使用它们。在初始化 MapControl 时不能使用定制工具。它们在 Page_Load 的时候添加。

Info 工具

将 Info 工具添加到 MapControl 集合并将其设置为当前工具后，必须添加事件方法以检索图元，这些图元被选作使用 Info 工具进行单击的结果。要执行该操作，在您的 IDE 中选择 Info 工具图标，右键单击该图标以打开“属性”窗口。单击闪电形状的事件图标来显示有效事件。找到 InfoRetrieved 事件并双击。然后，添加类似如下的代码片断以使用工具并检索选择的图元集合。

```
private void InfoToolControl1_InfoRetrieved(object sender,
                                             MapInfo.Web.UI.WebControls.InfoToolEventArgs e)
{
    // using MapInfo.Data
    MultiResultSetFeatureCollection multiCollection = null;
    multiCollection = e.Features;
    foreach ( FeatureCollection fCollection in multiCollection )
    {
        foreach ( Feature feature in fCollection )
        {
            ListBox1.Items.Add(feature.Table.Alias);
        }
    }
}
```

距离工具

与 Info 工具类似，在用户双击该工具以后，必须添加事件来检索计算过的距离。要执行该操作，在 IDE 中选择距离工具图标，右键单击该图标以打开“属性”窗口。单击闪电形状的事件图标来显示有效事件。双击 DistanceComputed 事件以将该事件添加到工具。添加类似如下示例的代码以检索距离并在文本框控件中显示。

```
private void DistanceToolControl1_DistanceComputed(object sender,
                                                   MapInfo.Web.UI.WebControls.DistanceToolEventArgs e)
{
    TextBox1.Text= string.Format("Distance calculated: {0}",
                                 e.Distance);
}
```

创建自定义工具

创建定制工具的一般过程是：

1. 从 MapInfo.Web.UI.Controls.MapTool 派生工具类。
2. 指定用于客户端绘制的 Start 和 Stop 方法。
3. 重载 Execute 方法并在其中编写业务逻辑。

下面的两个示例取自随 MapXtreme 2005 一起提供的 ToolsSample。示例 1 说明了服务器端的操作，而示例 2 则侧重于客户端的操作。

示例 1：编写服务器端定制工具

该示例显示如何编写您自己的定制工具，而不需要从头开始编写大量代码。创建一个名为 ZoomOutRectangleMapTool 的工具，可以通过在地图上绘制矩形进行缩小。

现在，为了创建工具，我们需要下面的信息：

- 如何实现该类？
- 如何在 Web 页面的地图上绘制矩形？
- 如何处理服务器端的信息以实际地执行缩放？
- 如何将它添加到工具集合中以便它们可以使用？

首先编写如下所示的类：

```
// This class is a custom tool which allows users to draw a
// rectangle and then zoom out. ZoomOut tool for example just
// allows users to zoom out on a click.
public class ZoomOutRectangleMapTool : MapTool
{
    private string _clientStartMethod;
    private string _clientStopMethod;
    private string _clientCodeSource = null;
    private string _name;
    public static readonly string Toolname = "ZoomOutRectangleTool";
    public ZoomOutRectangleMapTool()
    {
        //
        // TODO: Add constructor logic here
        //
        // Name of the tool
        _name = Toolname;
        // We use MapInfo's rectangle drawing client side code
        ClientStartMethod = "MapInfoWebRectangleStart";
        ClientStopMethod = "MapInfoWebRectangleStop";
    }
    /// Name of the tool, by default it is ZoomInTool
    public override string Name
    {
        get { return _name; }
        set { _name = value; }
    }

    // Name of the client side method to set up mouse event handlers
    // for drawing
    public override string ClientStartMethod
    {
        get { return _clientStartMethod; }
        set { _clientStartMethod = value; }
    }
}
```

```
// Name of the client side method to cancel the mouse event
// handlers
public override string ClientStopMethod
{
    get { return _clientStopMethod; }
    set { _clientStopMethod = value; }
}

// Source or url where the client drawing code is located
public override string ClientCodeSource
{
    get { return _clientCodeSource; }
    set { _clientCodeSource = value; }
}

// Url of the cursor image when this tool is active
public override string CursorUrl {
    get {return "";}
    set {;}
}

// This method sets the zoom value
public override void Execute(string dataString,
    System.Collections.ArrayList arrayList,
    MapInfo.Mapping.Map map) {
    // Extract points from the string
    System.Drawing.Point [] points =
        base.ExtractPoints(dataString);

    // There will be two points, end of rectangle in screen
    // coordinates, convert them to spatial
    MapInfo.Geometry.DPoint from;
    map.DisplayTransform.FromDisplay(points[0], out from);
    MapInfo.Geometry.DPoint to;
    map.DisplayTransform.FromDisplay(points[1], out to);

    // If first point and second points are very close to each
    // other treat it like a click
    if(((points[0].X - points[1].X) <= 20) &&
        ((points[0].Y - points[1].Y) <= 20)) {
        map.Zoom = new MapInfo.Geometry.Distance
            (map.Zoom.Value * 2, map.Zoom.Unit);
        map.Center = to;
    }
}
```

```
else {
    // Determine the zoom factor as the ratio of the drawn
    // rectangle to the bounds of the map and set the center
    // of the map as the center of the rectangle
    DRect r1 = new DRect(from, to);
    map.Center = r1.Center();

    DRect rect = new DRect();
    double mh = Math.Abs(map.Bounds.y2 - map.Bounds.y1);
    double mw = Math.Abs(map.Bounds.x2 - map.Bounds.x1);
    double rh = Math.Abs(to.y - from.y);
    double rw = Math.Abs(to.x - from.x);

    double ratio = Math.Min(mh, mw) / Math.Min(rh, rw);
    map.Zoom = new MapInfo.Geometry.Distance
        (map.Zoom.Value * ratio, map.Zoom.Unit);
}
}
```

正如所见，ClientStartMethod 和 ClientStopMethod 指定要绘制矩形的函数的名称。绘制选取框矩形、收集信息以及操作完成后将信息发送到服务器的功能都在随 MapXtreme 一起安装的 javascript 文件中实现。javascript 源文件安装在称为 MapXtremeWebResources（指向实际位置）的 IIS 虚拟目录中。

而服务器上，该代码的一大部分位于 Execute() 中。该函数自动从 MapControl 中调用，并带有正确的信息。该函数从隐藏的变量中提取点并执行该代码以变化缩放率。

创建这个类以后，将它添加到项目中。要使用该工具，首先通过更改该页面类的 Page_Load 方法将它添加到 MapControl 的工具集合。

```
private void Page_Load(object sender, System.EventArgs e)
{
    // Add your custom tool
    MapControl1.MapTools.Add(new ZoomOutRectangleMapTool());
}
```

要使用这个新的定制工具，将其设置为当前工具。例如，对于窗体上单击时调用句柄的按钮，可按如下所示设置当前工具：

```
private void Button1_Click(object sender, System.EventArgs e)
{
    MapControl1.MapTools.CurrentTool =
        ZoomOutRectangleMapTool.Toolname;
}
```

完成该操作后，用户可以在地图上绘制矩形并可以进行缩小。

示例 2：在客户端安装当前工具

上述示例的最后一部分说明了如何在服务器端以编程方式设置工具。如果想要执行的操作是设置当前工具，则可以使用能在绝大多数浏览器中工作的 Javascript 代码。这比进入服务器设置当前工具更加有效。注意：即使如此，仍需要编写更改地图的服务器端代码。

下面的示例代码取自 ToolSample 项目中的 WebForm3.aspx。

```
<%@ Register TagPrefix="mapinfowebuiwebcontrols"
Namespace="MapInfo.Web.UI.WebControls" Assembly="MapInfo.Web" %>
<%@ Page language="c#" Codebehind="WebForm3.aspx.cs"
AutoEventWireup="false" Inherits="WebApplication4.WebForm3" %>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN" >
<HTML>
    <HEAD>
        <title>WebForm3</title>
        <meta content="Microsoft Visual Studio .NET 7.1"
            name="GENERATOR">
        <meta content="C#" name="CODE_LANGUAGE">
        <meta content="JavaScript" name="vs_defaultClientScript">
        <meta content="http://schemas.microsoft.com/intellisense/ie5"
            name="vs_targetSchema">
    </HEAD>
    <body MS_POSITIONING="GridLayout">
        <asp:Label id="Label2" style="Z-INDEX: 106; LEFT: 704px;
            POSITION: absolute; TOP: 328px" runat="server"
            Height="16px" Width="56px" Font-Bold="True">Distance:
        </asp:Label>
        <br>
        <strong>This project contains two custom controls which extend
            MapInfo Web MapTool to do custom work.
        <br>
        The first one uses MapInfo client side drawing code
        <br>
        The second one uses the combination of custom client side
            drawing code and MapInfo client side API
        <br>
        <br>
        This Page demonstrates the flexibility of how tools can be
            used in any UI, and current tool can be set on server
            side or on the client side through javascript.
        <br>
        It also makes use of DistanceComputed event of
            DistanceMapTool to display distance in a textbox.
    </strong>
    <br>
    <br>
    <script src="JScript1.js"></script>
    <form id="Form1" method="post" runat="server">
```

```
<MAPINFOWEBUIWEBCONTROLS:MAPCONTROL id="MapControl1"
    style="Z-INDEX: 100; LEFT: 160px; POSITION: absolute;
    TOP: 176px" runat="server" Width="523px" Height="232px"
    WorkspaceFileName="WebForm3MapControl1.mws"
    BorderStyle="Solid" BorderWidth="1px" MapAlias="Map3"
    Format="Gif"></MAPINFOWEBUIWEBCONTROLS:MAPCONTROL>
<asp:dropdownlist id="DropDownList1" style="Z-INDEX: 101;
    LEFT: 704px; POSITION: absolute; TOP: 240px"
    runat="server" Width="160px" Height="40px"
    AutoPostBack="True"></asp:dropdownlist>
<table style="Z-INDEX: 102; LEFT: 176px; POSITION: absolute;
    TOP: 432px">
    <TR>
        <TD><STRONG>Select Current Tool:</STRONG></TD>
        <TD vAlign="middle"><INPUT id="radio1" type="radio"
            align="middle" value="MapInfoWebZoomIn"
            name="click" onclick="SetCurrentTool
                (this, 'MapControl1')">MapInfoWebZoomIn
        </TD>
        <TD vAlign="middle"><INPUT id="radio2" type="radio"
            align="middle"
            value="ZoomOutRectangleCustomClientDrawingTool"
            name="click" onclick="SetCurrentTool
                (this, 'MapControl1')">
            ZoomOutRectangleCustomClientDrawingTool
        </TD>
    </TR>
</table>
<asp:Label id="Label1" style="Z-INDEX: 104; LEFT: 704px;
    POSITION: absolute; TOP: 208px" runat="server"
    Height="16px" Width="136px" Font-Bold="True">
    Select Current Tool:
</asp:Label>
<asp:TextBox id="TextBox1" style="Z-INDEX: 105; LEFT: 704px;
    POSITION: absolute; TOP: 352px" runat="server"
    Height="24px" Width="160px">
</asp:TextBox>
</form>
<script>
    if (_map.MapTools.CurrentTool != null) {
        var rad1 = document.getElementById('radio1');
        if (rad1 != null && _map.MapTools.CurrentTool.Name ==
            rad1.value) {
            rad1.checked = true;
        }
        var rad2 = document.getElementById('radio2');
        if (rad2 != null && _map.MapTools.CurrentTool.Name ==
            rad2.value) {
            rad2.checked = true;
    }
}
```

```

        }
    }
    function SetCurrentTool(radio, map)
    {
        MapInfoWebSetCurrentTool(map, radio.value);
        var v = document.getElementById('currentName');
        if (v != null) {
            v.value = radio.value;
        }
    }
</script>
</body>
</HTML>

```

客户端工具示例

下面的 Javascript 示例说明了想要在客户端上执行的典型工具操作。

下面这些在窗体上显示的变量可用于工具名称:

```

// Globals for default tool names
var MapInfoWebZoomInTool = "MapInfoWebZoomIn";
var MapInfoWebZoomOutTool = "MapInfoWebZoomOut";
var MapInfoWebPanTool = "MapInfoWebPan";
var MapInfoWebCenterTool = "MapInfoWebCenter";
var MapInfoWebPointSelectionTool = "MapInfoWebPointSelection";
var MapInfoWebRadiusSelectionTool = "MapInfoWebRadiusSelection";
var MapInfoWebRectangleSelectionTool =
    "MapInfoWebRectangleSelection";
var MapInfoWebPolygonSelectionTool = "MapInfoWebPolygonSelection";
var MapInfoWebDistanceTool = "MapInfoWebDistance";
var MapInfoWebInfoTool = "MapInfoWebInfo";

```

要使地图对象获得给定 ID, 使用 MapInfoWebGetMap(mapid)。例如:

```
var obj = MapInfoWebGetMap('MapControl1');
```

要使客户机端上的当前工具接受给定地图 ID 以及工具名, 使用

MapInfoWebSetCurrentTool(mapid, toolName)。例如:

```
MapInfoWebSetCurrentTool('MapControl1',
    MapInfoWebRectangleSelectionTool);
```

下面的示例演示如何访问客户端上的地图对象和工具集合。

```
var map = document.getElementById('MapControl3_Image');
map.MapTools // to get tools collection
map.MapTools.CurrentTool // to get current tool
```

下面是可用于客户端绘制的开始和停止方法。

绘制矩形:

```
MapInfoWebRectangleStart  
MapInfoWebRectangleStop
```

处理单击:

```
MapInfoWebPointStart  
MapInfoWebPointStop
```

处理地图平移:

```
MapInfoWebPanStart  
MapInfoWebPanStop
```

绘制圆形:

```
MapInfoWebCircleStart  
MapInfoWebCircleStop
```

绘制折线:

```
MapInfoWebDistanceStart  
MapInfoWebDistanceStop
```

绘制多边形:

```
MapInfoWebPolygonStart  
MapInfoWebPolygonStop
```

K

Web 地图服务器示例

本附录包含示例 `wmsconfig.xml` 文件，该文件说明了 Web 地图绘制服务。

本附录内容：

- ◆ *WMS 配置文件示例* 518

WMS 配置文件示例

以下的 XML 文件描述了 Web 地图服务 (WMS) 的配置示例。如果想要管理 WMS，则需要名为 wmsconfig.xml 的类似文件。关于 WMS MapXtreme 2005 实现的概述，请参阅第 18 章：[WMS 和 WFS](#)。MapXtreme 2005 中定义 WMS 配置的构架位于产品 CD 上（文件名 MXP_WMS_Configuration_1_0.xsd）。

该示例服务器管理地图数据，WMS 客户端发出请求时，服务器将返回光栅图像，该图像可以在绘制地图的应用程序中作为图层使用。键元素用黑体标注以突出显示需要包含在文件中的信息类型。有关详细信息请参阅文档备注。

```
<?xml version="1.0" encoding="utf-8"?>
<WmsConfig xmlns:gml="http://www.opengis.net/gml" xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:mxp="http://www.mapinfo.com/mxp"
version="MXP_WMSConfig_1_0" disableFeatureInfo="false" xmlns="http://www.mapinfo.com/mxp/wms" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.mapinfo.com/mxp/wms
C:\SANDBOX\schemas\WMS\1_0\MXP_WMS_Configuration_1_0.xsd">
    <mxp-wms:Service>
        <mxp-wms:Name>Sample WMS Server</mxp-wms:Name>
        <mxp-wms:Title>Sample WMS Server</mxp-wms:Title>
        <mxp-wms:Abstract>This is a sample WMS server
            </mxp-wms:Abstract>
        <mxp-wms:KeywordList/>
        <!-- The following is the URL of the WMS server -->
        <mxp-wms:OnlineResource>http://localhost/WMS/GetMap.ashx
            </mxp-wms:OnlineResource>
        <mxp-wms:Fees>$10</mxp-wms:Fees>
        <mxp-wms:AccessConstraints>none</mxp-wms:AccessConstraints>
    </mxp-wms:Service>
    <!-- The following MIME types will not be used by the server -->
    <mxp-wms:WmsRasterFormatExcludeSet>
        <mxp-wms:Format>image/jpeg2000</mxp-wms:Format>
        <mxp-wms:Format>image/photoshop</mxp-wms:Format>
        <mxp-wms:Format>image/jp2</mxp-wms:Format>
    </mxp-wms:WmsRasterFormatExcludeSet>
    <!-- The following connection is referenced by one of the data
        sources -->
    <ConnectionSet xmlns="http://www.mapinfo.com/mxp">
        <DBConnection dbType="informix">
            <ConnectionName>spyro</ConnectionName>
            <ODBCConnectionString>Driver={INFORMIX 3.81 32
                BIT};UID=atsmipro;PWD=atsmipro;DATABASE=qadb;
                HOST=threemile;SRVR=threemile_ds;SERV=1800;
                PRO=onsoctcp;CLOC=en_US.CP1252;DLOC=en_US.CP1252;
                VMB=0;CURB=0;OPT=;SCUR=0;ICUR=0;OAC=1;OPTOFC=0;
```

```
RKC=0;ODTYP=0;DDFP=0;DNL=0;RCWC=0
  </ODBCConnectionString>
</DBConnection>
</ConnectionSet>
<DataSourceDefinitionSet xmlns="http://www.mapinfo.com/mxp/wms">
  <!-- The following data sources reference local TAB files -->
  <TABFileDataSourceDefinition id="id1" readOnly="false"
    xmlns="http://www.mapinfo.com/mxp">
    <DataSourceName>STATES</DataSourceName>
    <FileName>C:\data\MIPro65\Tut_Data\Tut_Usa\USA\STATES.TAB
    </FileName>
  </TABFileDataSourceDefinition>
  <TABFileDataSourceDefinition id="id2" readOnly="false"
    xmlns="http://www.mapinfo.com/mxp">
    <DataSourceName>US_HIWAY</DataSourceName>
    <FileName>C:\data\MIPro65\Tut_Data\Tut_Usa\USA\US_HIWAY.TAB
    </FileName>
  </TABFileDataSourceDefinition>
  <TABFileDataSourceDefinition id="id4" readOnly="false"
    xmlns="http://www.mapinfo.com/mxp">
    <DataSourceName>CANHIWAY</DataSourceName>

  <FileName>C:\data\MIPro65\Tut_Data\Tut_Usa\CANADA\CANHIWAY.TAB
    </FileName>
  </TABFileDataSourceDefinition>
  <TABFileDataSourceDefinition id="id5" readOnly="false"
    xmlns="http://www.mapinfo.com/mxp">
    <DataSourceName>OCEAN</DataSourceName>
    <FileName>C:\data\MIPro65\Tut_Data\Tut_Usa\WORLD\OCEAN.TAB
    </FileName>
  </TABFileDataSourceDefinition>
  <!-- The following data source references a local named
       connection -->
  <DBDataSourceDefinition id="id3" readOnly="false"
    xmlns="http://www.mapinfo.com/mxp">
    <DataSourceName>alias_Line_spyro</DataSourceName>
    <ConnectionMember>
      <ConnectionName>spyro</ConnectionName>
    </ConnectionMember>
    <DBQuery>
      <Query>select * from usa_caps_extra</Query>
    </DBQuery>
    <DBDataSourceMetadata>
      <FeatureGeometryAttribute srsName="mapinfo:coordsys 1,62">
        sw_geometry</FeatureGeometryAttribute>
        <KeyAttributes/>
      </DBDataSourceMetadata>
    </DBDataSourceDefinition>
```

```
<!-- The following data source uses an inline connection to  
an Oracle server -->  
<DBDataSourceDefinition id="id6" readOnly="false"  
xmlns="http://www.mapinfo.com/mxp">  
  <DataSourceName>ASIA</DataSourceName>  
  <ConnectionMember>  
    <InlineDBConnection>  
      <OCIConnectionString>SRVR=ontario;UID=atsmipro;PWD=atsmipro  
      </OCIConnectionString>  
    </InlineDBConnection>  
  </ConnectionMember>  
  <DBQuery>  
    <Query>select * from "ATSJAVA"."ASIA"</Query>  
  </DBQuery>  
  <DBDataSourceMetadata>  
    <FeatureGeometryAttribute  
      srsName="mapinfo:coordsys 1,0">  
      OBJECT  
    </FeatureGeometryAttribute>  
    <KeyAttributes/>  
  </DBDataSourceMetadata>  
</DBDataSourceDefinition>  
</DataSourceDefinitionSet>  
<mxp-wms:WmsLayer queryable="false">  
  <mxp-wms:Name>World</mxp-wms:Name>  
  <mxp-wms:Title>The World</mxp-wms:Title>  
  <mxp-wms:Abstract>This is a root layer that represents the whole  
  world.  
  </mxp-wms:Abstract>  
<!-- The following SRS is available to all layers -->  
  <mxp-wms:SRSNameSet>  
    <mxp:SRSName>EPSG:4326</mxp:SRSName>  
  </mxp-wms:SRSNameSet>  
  <mxp-wms:WmsStyleSet/>  
  <mxp-wms:WmsLayerList>  
    <mxp-wms:WmsLayer queryable="false">  
      <mxp-wms:Name>USA</mxp-wms:Name>  
      <mxp-wms:Title>United States</mxp-wms:Title>  
      <mxp-wms:Abstract>Contains layers for the United States  
      </mxp-wms:Abstract>  
    <!-- The following SRS is available to this layer and all of the  
    layers underneath it -->  
    <mxp-wms:SRSNameSet>  
      <mxp:SRSName>EPSG:4267</mxp:SRSName>  
    </mxp-wms:SRSNameSet>  
    <mxp-wms:WmsStyleSet/>  
    <mxp-wms:WmsLayerList>  
      <mxp-wms:WmsLayer>  
        <mxp-wms:Name>States</mxp-wms:Name>
```

```
<mxp-wms:Title>States</mxp-wms:Title>
<mxp-wms:SRSNameSet/>
<mxp-wms:WmsStyleSet>
    <mxp-wms:WmsStyle>
        <mxp-wms:Name>BlueFill</mxp-wms:Name>
        <mxp-wms:Title>Blue Fill</mxp-wms:Title>
        <mxp-wms:Abstract>This is a blue area fill
            with a red border.</mxp-wms:Abstract>
        <AreaStyle
            xmlns="http://www.mapinfo.com/mxp">
            <!-- The following defines the red border -->
            <LineStyle stroke="red" width="1"
                width-unit="mapinfo:imagesize pixel">
                <Pen>mapinfo:pen 2</Pen>
            </LineStyle>
            <!-- The following defines the blue fill -->
            <Interior fill-opacity="0" fill="#id7">
                <Defs>
                    <Pattern id="id7">
                        <Bitmap uri="mapinfo:brush 2">
                            <ColorAdjustmentSet>
                                <ColorAdjustment color-1="nonWhite"
                                    color-2="blue" opacity="1"/>
                                <ColorAdjustment color-1="white"
                                    opacity="1"/>
                            </ColorAdjustmentSet>
                        </Bitmap>
                    </Pattern>
                </Defs>
            </Interior>
        </AreaStyle>
    </mxp-wms:WmsStyle>
</mxp-wms:WmsStyleSet>
<FeatureLayer id="id8" name="STATES"
    alias="STATES" volatile="unknown"
    xmlns="http://www.mapinfo.com/mxp">
    <DataSourceRef ref="id1"/>
</FeatureLayer>
</mxp-wms:WmsLayer>
<mxp-wms:WmsLayer>
    <mxp-wms:Name>USHighways</mxp-wms:Name>
    <mxp-wms:Title>US Highways</mxp-wms:Title>
    <mxp-wms:Abstract>The major highways for the
        United States</mxp-wms:Abstract>
    <mxp-wms:SRSNameSet/>
    <mxp-wms:WmsStyleSet/>
    <FeatureLayer id="id9" name="US_HIWAY"
        alias="US_HIWAY" volatile="unknown"
        xmlns="http://www.mapinfo.com/mxp">
```

```
<DataSourceRef ref="id2" />
</FeatureLayer>
</mxp-wms:WmsLayer>
<mxp-wms:WmsLayer>
    <mxp-wms:Name>USCapitals</mxp-wms:Name>
    <mxp-wms:Title>US Capitals</mxp-wms:Title>
    <mxp-wms:Abstract>The capitals for the United
        States.</mxp-wms:Abstract>
    <mxp-wms:SRSNameSet/>
    <mxp-wms:WmsStyleSet/>
    <FeatureLayer id="id10" name="STATECAP"
        alias="STATECAP" volatile="unknown"
        xmlns="http://www.mapinfo.com/mxp">
        <DataSourceRef ref="id3" />
    </FeatureLayer>
    </mxp-wms:WmsLayer>
</mxp-wms:WmsLayerList>
</mxp-wms:WmsLayer>
<mxp-wms:WmsLayer queryable="false">
    <mxp-wms:Name>Canada</mxp-wms:Name>
    <mxp-wms:Title>Canada</mxp-wms:Title>
    <mxp-wms:Abstract>Contains layers for
        Canada</mxp-wms:Abstract>
    <mxp-wms:SRSNameSet/>
    <mxp-wms:WmsStyleSet/>
    <mxp-wms:WmsLayerList>
        <mxp-wms:WmsLayer>
            <mxp-wms:Name>CanadaHighway</mxp-wms:Name>
            <mxp-wms:Title>Canada Highway</mxp-wms:Title>
            <mxp-wms:Abstract>The major highways in Canada
            </mxp-wms:Abstract>
            <mxp-wms:SRSNameSet/>
            <mxp-wms:WmsStyleSet>
                <mxp-wms:WmsStyle>
                    <mxp-wms:Name>ThickGreenLine</mxp-wms:Name>
                    <mxp-wms:Title>Thick Green Line
                    </mxp-wms:Title>
                    <mxp-wms:Abstract>A thick, green line
                        style.</mxp-wms:Abstract>
                <!-- The following defines a black line style -->
                <LineStyle stroke="black" width="1"
                    width-unit="mapinfo:imagesize pixel"
                    xmlns="http://www.mapinfo.com/mxp">
                    <Pen>mapinfo:pen 2</Pen>
                </LineStyle>
            </mxp-wms:WmsStyle>
        </mxp-wms:WmsLayer>
    </mxp-wms:WmsLayerList>
</mxp-wms:WmsStyleSet>
```

```
<FeatureLayer id="id11" name="CANHIWAY"
    alias="CANHIWAY" volatile="unknown"
    xmlns="http://www.mapinfo.com/mxp">
    <DataSourceRef ref="id4" />
</FeatureLayer>
</mfp-wms:WmsLayer>
<mfp-wms:WmsLayerList>
</mfp-wms:WmsLayer>
<mfp-wms:WmsLayer>
    <mfp-wms:Name>Oceans</mfp-wms:Name>
    <mfp-wms:Title>Oceans</mfp-wms:Title>
    <mfp-wms:Abstract>The oceans of the world
    </mfp-wms:Abstract>
    <mfp-wms:SRSNameSet/>
    <mfp-wms:WmsStyleSet/>
    <mfp:FeatureLayer id="id12" name="OCEAN" alias="OCEAN"
        volatile="unknown">
        <mfp:DataSourceRef ref="id5"/>
    </mfp:FeatureLayer>
</mfp-wms:WmsLayer>
<mfp-wms:WmsLayer>
    <mfp-wms:Name>Asia</mfp-wms:Name>
    <mfp-wms:Title>Asia</mfp-wms:Title>
    <mfp-wms:SRSNameSet/>
    <mfp-wms:WmsStyleSet/>
    <FeatureLayer id="id13" name="ASIA" alias="ASIA"
        volatile="unknown" xmlns="http://www.mapinfo.com/mxp">
        <DataSourceRef ref="id6" />
    </FeatureLayer>
</mfp-wms:WmsLayer>
</mfp-wms:WmsLayerList>
</mfp-wms:WmsLayer>
</WmsConfig>
```




坐标系的元素

本附录内容：

- ◆ 投影及其参数..... 526
- ◆ 投影基准面..... 531
- ◆ 基准面转换..... 542
- ◆ 有关坐标系和投影的详细信息 542
- ◆ 定制基准面..... 544

投影及其参数

下表指示了适用于每个投影的参数，它们以出现在 MapInfoCoordinateSystemSet.xml 中的相关坐标系行中的顺序列出。该文档位于
C:\Program Files\Common Files\MapInfo\MapXtreme\6.x 目录下。

	基准面	单位	经原点	纬原点	标准纬线 1	标准纬线 2	方位角	比例因子	东伪偏移	北伪偏移	范围
阿伯斯等积圆锥	X	X	X	X	X	X			X	X	
方位等距	X	X	X	X							X
Cassini-Soldner	X	X	X	X					X	X	
贝尔曼圆柱等积	X	X	X		X						
球面双像	X	X	X	X				X	X	X	
埃克脱 IV	X	X	X								
埃克脱 VI	X	X	X								
等距圆锥	X	X	X	X	X	X			X	X	
高爾	X	X	X								
Hotine Oblique Mercator	X	X	X	X			X	X	X	X	
兰勃特方位等积	X	X	X	X							X
兰勃特等角圆锥	X	X	X	X	X	X			X	X	
经度 / 纬度	X										
墨卡托	X	X	X								
米勒	X	X	X								
摩尔魏特	X	X	X								
新西兰地图网格	X	X	X	X				X	X		
多圆锥	X	X	X	X				X	X		

	基准面	单位	原点, 经度	原点, 纬度	标准纬线 1	标准纬线 2	方位角	比例因子	东伪偏移	北伪偏移	范围
区域墨卡托	X	X	X		X						
罗宾逊	X	X	X								
正弦曲线	X	X	X								
球面	X	X	X	X				X	X	X	
Swiss Oblique Mercator	X	X	X	X					X	X	
横墨卡托	X	X	X	X				X	X	X	

投影

投影是坐标系使用的等式或等式集。下表命名了 MapInfo 使用的投影并给出了用于标识 MapInfoCoordinateSystemSet 文件中投影的编号:

编号	投影
9	Albers Equal-Area Conic
28	Azimuthal Equidistant (all origin latitudes)
5	Azimuthal Equidistant (polar aspect only)
30	Cassini-Soldner
2	Cylindrical Equal-Area
31	Double Stereographic
14	Eckert IV
15	Eckert VI
6	Equidistant Conic, 也称做 Simple Conic
17	Gall
7	Hotine Oblique Mercator
4	Lambert Azimuthal Equal-Area (polar aspect only)

编号	投影
29	Lambert Azimuthal Equal-Area
3	Lambert Conformal Conic
19	Lambert Conformal Conic (modified for Belgium 1972)
1	Longitude/Latitude
10	Mercator
11	Miller Cylindrical
13	Mollweide
18	New Zealand Map Grid
27	Polyconic
26	Regional Mercator
12	Robinson
16	Sinusoidal
20	Stereographic
25	Swiss Oblique Mercator
8	Transverse Mercator, (也称做 Gauss-Kruger)
21	Transverse Mercator, (modified for Danish System 34 Jylland-Fyn)
22	Transverse Mercator, (modified for Danish System 34 Sjaelland)
23	Transverse Mercator, (modified for Danish System 34/45 Bornholm)
24	Transverse Mercator, (modified for Finnish KKJ)

例如，Longitude/Latitude 投影在 MapInfoCoordinateSystemSet.xml 中列出如下：

```
<GeographicCRS>
    <srsName>Longitude / Latitude (Porto Santo 1936)</srsName>
    <srsID>
        <code>coordsys 1,94</code>
        <codeSpace>mapinfo</codeSpace>
        <remarks>Longitude / Latitude (v 6.0 and later projections)</remarks>
    </srsID>
    <usesEllipsoidalCS/>
    <usesGeodeticDatum/>
</GeographicCRS>
```

上述 MapInfoCoordinateSystemSet.xml 中的投影编号，可以通过将常数值添加到 Projection 表中所列基数的方法修改。有效值及其含义列表以下：

常数	含义	参数
1000	系统具有仿射变换	仿射单位指定符和系数在系统常规参数之后出现。
2000	系统具有显式边界	边界在系统常规参数之后出现。
3000	系统既有仿射又有边界	仿射参数跟随系统参数；边界跟随仿射参数。

示例：

假定要以简单系统工作，该系统基于 Transverse Mercator 投影并使用 NAD 1983 基准面。可以采用一行代码，例如 MapInfoCoordinateSystemSet.xml 文件中的以下示例：

```
<ProjectedCRS>
    <srsName>UTM Zone 1 (NAD 83)</srsName>
    <srsID>
        <code>coordsys 8,33,7,-177,0,0.9996,500000,0</code>
        <codeSpace>mapinfo</codeSpace>
        <remarks>Universal Transverse Mercator </remarks>
    </srsID>
    <baseCRS/>
    <definedByConversion/>
    <usesCartesianCS/>
</ProjectedCRS>
```

需要基于以上的系统，但是要有以下参数指定的仿射变换：

Units=meters; A=0.5; B=-0.866; C=0; D=0.866; E=0.5; and F=0.

需要的 MapInfoCoordinateSystemSet.xml 文件中的代码是:

```
<ProjectedCRS>
    <srsName>UTM Zone 1 (NAD 83) - rotated 60 degrees</srsName>
    <srsID>
        <code>coordsys 8,33,7,-177,0,0.9996,500000,0, 7, 0.5, -
            0.866, 0, 0.866, 0.5, 0 </code>
        <codeSpace>mapinfo</codeSpace>
        <remarks>Universal Transverse Mercator </remarks>
    </srsID>
    <baseCRS/>
    <definedByConversion/>
    <usesCartesianCS/>
</ProjectedCRS>
```

或者, 如果要将系统边界设置为 (x1, y1, x2, y2)=(-500000, 0, 500000, 1000000), 需要的代码是:

```
<ProjectedCRS>
    <srsName>UTM Zone 1 (NAD 83) - bounded</srsName>
    <srsID>
        <code>coordsys 8,33,7,-177,0,0.9996,500000,0, -500000, 0,
            500000, 1000000
        </code>
        <codeSpace>mapinfo</codeSpace>
        <remarks>Universal Transverse Mercator </remarks>
    </srsID>
    <baseCRS/>
    <definedByConversion/>
    <usesCartesianCS/>
</ProjectedCRS>
```

要使用以上的修改将系统定制, 代码如下:

```
<ProjectedCRS>
    <srsName>UTM Zone 1 (NAD 83)</srsName>
    <srsID>
        <code>coordsys 8,33,7,-177,0,0.9996,500000,0, 7, 0.5, -
            0.866, 0, 0.866, 0.5, 0, -500000, 0, 500000,
            1000000 </code>
        <codeSpace>mapinfo</codeSpace>
        <remarks>Universal Transverse Mercator </remarks>
    </srsID>
    <baseCRS/>
    <definedByConversion/>
    <usesCartesianCS/>
</ProjectedCRS>
```

投影基准面

基准面是通过将参考椭面关联到地球上特定点的方法建立的。下表列出了每个基准面的详细信息：

- MapInfoCoordinateSystemSet.xml 文件中用于定义基准面的数字。
- 基准面的名称
- 通常使用基准面的地图
- 基准面的参考椭面

编号	基准面	区域地图	椭面
1	Adindan	埃塞俄比亚、马里、塞内加尔、苏丹	Clarke 1880
2	Afgooye	索马里	Krassovsky
1007	AGD 66, 7 parameter	澳大利亚、A.C.T.	Australian National
1008	AGD 66, 7 parameter	澳大利亚、塔斯马尼亚	Australian National
1009	AGD 66, 7 parameter	澳大利亚、维多利亚 /NSW	Australian National
1006	AGD 84, 7 parameter	澳大利亚	Australian National
3	Ain el Abd 1970	巴林群岛	International
118	American Samoa	美属萨摩亚群岛	Clarke 1866
4	Anna 1 Astro 1965	可可斯群岛	Australian National
119	Antigua Island Astro 1943	安提瓜、顺风群岛	Clarke 1880
5	Arc 1950	博茨瓦纳、莱索托、马拉维、斯威士兰、扎伊尔、赞比亚、津巴布韦	Clarke 1880
6	Arc 1960	肯尼亚、坦桑尼亚	Clarke 1880
7	Ascension Island 1958	阿森松岛	International
9	Astro B4 Sorol Atoll	燕鸥岛	International
8	Astro Beacon “E”	硫磺岛	International
10	Astro DOS 71/4	圣赫勒拿岛	International

编号	基准面	区域地图	椭面
11	Astronomic Station 1952	马库斯岛	International
151	ATS77 (Average Terrestrial System 1977)	加拿大	ATS77
12	Australian Geodetic 1966 (AGD 66 - 3 param)	澳大利亚和塔斯马尼亚岛	Australian National
13	Australian Geodetic 1984 (AGD 84 - 3 param)	澳大利亚和塔斯马尼亚岛	Australian National
1006	Australia-National-AGD84 7 param	澳大利亚	Australian National
1007	Australia (A.C.T. AGD66 7-param)	澳大利亚	Australian National
1008	Australia (Tasmania AGD66 7-param)	塔斯马尼亚	Australian National
1009	Australia (Victoria/NSW AGD66 7-param)	维多利亚、 NSW	Australian National
151	Average Terrestrial System 1977 (ATS77)		
120	Ayabelle Lighthouse	吉布提	Clarke 1880
110	比利时	比利时	International
14	Bellevue (IGN)	埃法特和埃罗蒙哥群岛	International
15	Bermuda 1957	百慕大群岛	Clarke 1866
16	Bogota Observatory	哥伦比亚	International
121	Bukit Rimpah	邦加和勿里洞群岛 (印度尼西亚)	Bessel 1841
17	Campo Inchauspe	阿根廷	International
18	Canion Astro 1966	凤凰群岛	International
19	Cape	南非	Clarke 1880
20	Cape Canaveral	佛罗里达和巴哈马群岛	Clarke 1866
1005	Cape, 7 parameter	南非	WGS 84

编号	基准面	区域地图	椭面
21	Carthage	突尼斯	Clarke 1880
22	Chatham 1971	查塔姆岛（新西兰）	International
23	Chua Astro	巴拉圭	International
122	Co-Ordinate System 1937 of Estonia	爱沙尼亚	Bessel 1841
24	Corrego Alegre	巴西	International
123	Dabola	几内亚	Clarke 1880
124	Deception Island	迷幻岛、南极洲	Clarke 1880
1000	Deutsches Hauptdreiecksnetz (DHDN)	德国	Bessel
25	Djakarta (Batavia)	苏门达腊岛（印度尼西亚）	Bessel 1841
26	DOS 1968	Gizo 岛（新乔治亚群岛）	International
27	Easter Island 1967	复活节岛	International
115	EUREF 89	欧洲	GRS 80
28	European 1950 (ED 50)	奥地利、比利时、丹麦、芬兰、法国、德国、直布罗陀、希腊、意大利、卢森堡、荷兰、挪威、葡萄牙、西班牙、瑞典、瑞士	International
29	European 1979 (ED 79)	奥地利、芬兰、荷兰、挪威、西班牙、瑞典、瑞士	International
108	European 1987 (ED 87)	欧洲	International
125	Fort Thomas 1955	Nevis、St. Kitts、顺风群岛	Clarke 1880
30	Gandajika Base	马尔代夫共和国	International
116	GDA 94	澳大利亚	GRS 80
32	Geodetic Reference System 1967 (GRS 67)	全球	GRS 67
33	Geodetic Reference System 1980 (GRS 80)	全球	GRS 80

编号	基准面	区域地图	椭面
126	Graciosa Base SW 1948	Faial、Graciosa、Pico、Sao Jorge 和 Terceira 群岛（亚速尔群岛）	International 1924
34	Guam 1963	关岛	Clarke 1866
35	GUX 1 Astro	瓜达康纳尔岛	International
150	Hartbeesthoek 94	南非	WGS 84
127	Herat North	阿富汗	International 1924
128	Hermannskogel	南斯拉夫（1990 年解体）、斯诺文尼亞、克罗地亚、波斯尼亞、黑塞哥维那、塞尔维亚	Bessel 1841
36	Hito XVIII 1963	南智利（靠近 53° S）	International
37	Hjorsey 1955	冰岛	International
38	中国香港特别行政区 1963	中国香港特别行政区	International
1004	匈牙利大地基准面 (HD 72)	匈牙利	GRS 67
39	Hu-Tzu-Shan	中国台湾地区	International
40	Indian	泰国和越南	Everest (India 1830)
41	Indian	孟加拉、印度、尼泊尔	Everest (India 1830)
129	Indian	巴基斯坦	Everest (Pakistan)
130	印度 1954	泰国	Everest (India 1830)
131	印度 1960	越南	Everest (India 1830)
132	印度 1975	泰国	Everest (India 1830)
133	印度尼西亚 1974	印度尼西亚	印度尼西亚 1974
42	Ireland 1965	爱尔兰	Modified Airy
134	ISTS 061 Astro 1968	南乔治亚岛	International 1924
43	ISTS 073 Astro 1969	Diego Garcia	International
152	Japanese Geodetic Datum 2000 (JGD2000)	日本	GRS80

编号	基准面	区域地图	椭面
44	Johnston Island 1961	约翰斯顿岛	International
45	Kandawala	斯里兰卡	Everest (India 1830)
46	凯尔盖朗岛	凯尔盖朗岛	International
47	Kertau 1948	西马来西亚和新加坡	Everest (W. Malaysia and Singapore 1948)
135	Kusaie Astro 1951	卡罗琳群岛、密克罗尼西亚联邦	International 1924
48	L.C. 5 Astro	大开曼群岛	Clarke 1866
136	Leigon	加纳	Clarke 1880
49	Liberia 1964	利比里亚	Clarke 1880
113	Lisboa (DLx)	葡萄牙	International
50	Luzon	菲律宾(不含棉兰老岛)	Clarke 1866
51	Luzon	棉兰老岛	Clarke 1866
52	Mahe 1971	马希岛	Clarke 1880
53	Marco Astro	Salvage 群岛	International
54	Massawa	厄立特里亚(埃塞俄比亚)	Bessel 1841
114	Melrica 1973 (D73)	葡萄牙	International
55	Merchich	摩洛哥	Clarke 1880
56	Midway Astro 1961	中途岛	International
57	Minna	尼日利亚	Clarke 1880
137	Montserrat Island Astro 1958	蒙特塞拉特、顺风群岛	Clarke 1880
138	M' Poraloko	加蓬	Clarke 1880
58	Nahrwan	Masirah 岛(阿曼)	Clarke 1880
59	Nahrwan	阿拉伯联合酋长国	Clarke 1880
60	Nahrwan	沙特阿拉伯	Clarke 1880
61	Naparima, BWI	特立尼达和多巴哥	International

编号	基准面	区域地图	椭面
109	荷兰	荷兰	Bessel
31	新西兰 Geodetic 大地基准面 1949 (NZGD 49)	新西兰	International
1010	(New Zealand (NZGD49 7-parameter))	新西兰	International
62	北美洲 1927 (NAD 27)	美国大陆	Clarke 1866
63	北美洲 1927 (NAD 27)	阿拉斯加	Clarke 1866
64	北美洲 1927 (NAD 27)	巴哈马 (不含圣萨尔瓦多岛)	Clarke 1866
65	北美洲 1927 (NAD 27)	圣萨尔瓦多岛	Clarke 1866
66	北美洲 1927 (NAD 27)	加拿大 (包括纽芬兰岛)	Clarke 1866
67	北美洲 1927 (NAD 27)	运河区	Clarke 1866
68	北美洲 1927 (NAD 27)	加勒比 (特克斯和凯科斯群岛)	Clarke 1866
69	北美洲 1927 (NAD 27)	中美洲 (伯利兹、哥斯达黎加、萨尔瓦多、危地马拉、洪都拉斯、尼加拉瓜)	Clarke 1866
70	北美洲 1927 (NAD 27)	古巴	Clarke 1866
71	北美洲 1927 (NAD 27)	格陵兰 (Hayes Peninsula)	Clarke 1866
72	北美洲 1927 (NAD 27)	墨西哥	Clarke 1866
73	北美洲 1927 (NAD 27)	密歇根州 (只用于 State Plane Coordinate System 1927)	Modified Clarke 1866
74	北美洲 1983 (NAD 83)	阿拉斯加、加拿大、中美洲、美国大陆、墨西哥	GRS 80
139	North Sahara 1959	阿尔及利亚	Clarke 1880
107	Nouvelle Triangulation Francaise (NTF) Greenwich Prime Meridian	法国	Modified Clarke 1880
1002	Nouvelle Triangulation Francaise (NTF) Paris Prime Meridian	法国	Modified Clarke 1880

编号	基准面	区域地图	椭面
111	NWGL 10	全球	WGS 72
117	NZGD 2000	新西兰	GRS 80
1010	NZGD 49, 7 parameter	新西兰	International
75	Observatorio 1966	Corvo and Flores 群岛 (亚速尔群岛)	International
140	Observatorio Meteorologico 1939	Corvo and Flores 群岛 (亚速尔群岛)	International 1924
76	Old Egyptian	埃及	Helmert 1906
77	Old Hawaiian	夏威夷	Clarke 1866
97	OldTokyo	日本、韩国、冲绳	Bessel 1841
78	阿曼	阿曼	Clarke 1880
79	Ordnance Survey of Great Britain 1936	英格兰、怀特岛、苏格兰、设得兰 群岛、威尔士	Airy
80	Pico de las Nieves	加拿利群岛	International
81	Pitcairn Astro 1967	皮特克恩岛	International
141	Point 58	布基纳法索和尼日尔	Clarke 1880
142	Pointe Noire 1948	刚果	Clarke 1880
143	Porto Santo 1936	波尔图桑塔和马德拉群岛	International 1924
1000	Potsdam	德国	Bessel
82	Provisional South American 1956	玻利维亚、智利、哥伦比亚、厄瓜 多尔、圭亚那、秘鲁、委内瑞拉	International
36	Provisional South Chilean 1963	南智利 (靠近 53° S)	International
83	Puerto Rico	波多黎各和维京群岛	Clarke 1866
1001	Pulkovo 1942	德国	Krassovsky
1012	PZ90	俄罗斯	PZ90
84	Qatar National	卡塔尔	International

编号	基准面	区域地图	椭面
85	Qornoq	南格陵兰岛	International
1000	Rauenberg	德国	Bessel
86	Reunion	马斯克林岛	International
112	Rikets Triangulering 1990 (RT 90)	瑞典	Bessel
1011	Rikets Triangulering 1990 (RT 90), 7 parameter	瑞典	Bessel
87	罗马 1940	撒丁尼亚岛	International
88	Santo (DOS)	圣埃斯皮里托岛	International
89	São Braz	São Miguel, Santa Maria 群岛（亚速尔群岛）	International
90	Sapper Hill 1943	东福克兰岛	International
91	Schwarzeck	纳米比亚	Modified Bessel 1841
144	Selvagem Grande 1938	Salvage 群岛	International 1924
145	Sierra Leone 1960	塞拉里昂	Clarke 1880
146	S-JTSK	捷克共和国	Bessel 1841
1013	SK42	俄罗斯	PZ90
1024	SK95	俄罗斯	PZ90
92	南美洲 1969	阿根廷、玻利维亚、巴西、智利、哥伦比亚、厄瓜多尔、圭亚那、巴拉圭、秘鲁、委内瑞拉、特立尼达和多巴哥	南美洲 1969
93	南亚	新加坡	Modified Fischer 1960
94	东南基准	Porto Santo and Madeira 群岛	International
95	西南基准	Faial、Graciosa、Pico、Sao Jorge、Terceira 群岛（亚速尔群岛）	International
1003	瑞士 (CH 1903)	瑞士	Bessel

编号	基准面	区域地图	椭面
147	Tananarive Observatory 1925	马达加斯加	International 1924
96	Timbalai 1948	文莱和东马来西亚 (沙捞越和沙巴)	Everest (India 1830)
1015	东京	日本	Bessel 1841
98	Tristan Astro 1968	崔斯坦火山岛	International
99	Viti Levu 1916	梅地来雾群岛 (斐济群岛)	Clarke 1880
148	Voirol 1874	突尼斯 / 阿尔及利亚	Clarke 1880
149	Voirol 1960	阿尔及利亚	Clarke 1880
100	Wake-Eniwetok 1960	马歇尔群岛	Hough
101	World Geodetic System 1960 (WGS 60)	全球	WGS 60
102	World Geodetic System 1966 (WGS 66)	全球	WGS 66
103	World Geodetic System 1972 (WGS 72)	全球	WGS 72
104	World Geodetic System 1984 (WGS 84)	全球	WGS 84
105	Yacare	乌拉圭	International
106	Zanderij	苏里南	International

单位

下表列出了 MapInfoCoordinateSystemSet.xml 文件中可用的坐标单位和用于标识单位的编号:

编号	单位
6	厘米
31	链
3	英尺 (也称为国际英尺) *

编号	单位
2	英寸
1	千米
30	令
7	米
0	英里
5	毫米
9	海里 [†]
32	杆
8	美国测量英尺（用于 1927 State Plane） [‡]
4	码

* 一国际英尺精确等于 30.48 厘米。

—海里精确等于 1852 米。

** —美国测量英尺精确等于 12/39.37 米，或近似等于 30.48006 厘米。

坐标系原点

原点是以经度和纬度指定的点，所有坐标都从该原点引用。选择原点的目的是优化特定坐标系的精确性。从原点向北移动时，Y 值增加。从原点向东移动时，X 值增加。这些坐标值通常称为北偏移和东偏移。

对于 Transverse Mercator 投影，原点的经度定义了中心子午线。在构造 Transverse Mercator 投影的过程中，定位一个柱面与地球相切。中心子午线是切线。沿着中心子午线，投影地图的比例为真。

在创建 Hotine Oblique Mercator 投影的过程中，有必要指定一个既不是赤道也不是子午线的大圆周。MapInfo 通过指定一个椭面上的点和从该点发出的方位角以实现该功能。那个点正是坐标系的原点。

标准纬线（圆锥投影）

在圆锥投影中，一个锥形沿着纬线的两条平行线穿过与它相交的地球区域。这两条平行线就是标准纬线。一条在投影区域的北面，另一条在投影区域的南面。要使用单一标准纬线两次指定那条纬线。它们都是用纬线的度数表示。

倾斜方位角 (Hotine Oblique Mercator)

使用点和方位角指定上述的大圆周 (Hotine Oblique Mercator) 时，方位角被称为 Oblique Azimuth 并用度数表示。

比例因子 (Transverse Mercator)

比例因子应用于柱面坐标，以均分沿着东部和西部边界减少错误时地图中心区域的比例错误。比例因子有使柱面凹进地球区域的效果，所以它有两条相交线。沿着这两条相交线比例为 true。

比例因子可以表示为比值的形式，例如 1:25000。这种情况下它通常被称比例减少。比例因子和比例减少之间的关系是：

$$\text{比例因子} = 1 - \text{比例减少}$$

这种情况下比例因子就是 $1 - (1/25000)$ 或 0.99996。

北伪偏移和东伪偏移

如果不涉及负数，则坐标的计算会更简单。要在计算 State Plane 和 Universal Transverse Mercator 坐标的过程中排除这个问题，常用的方法是将测量偏移量增加到北偏移和东偏移。这些偏移量被称为北伪偏移和东伪偏移。它们以坐标单位表示，而不是以度表示。
(坐标单位由 Units 参数指定。)

范围 (Azimuthal 投影)

以度为单位指定的范围，表示观测的地球区域的大小。范围可以在 1 和 180 之间。指定的是 90 度时，就可以看到半球区域。指定的是 180 度时就可以看到整个地球，尽管它的大部分区域是扭曲的。

Polyconic 投影

以下描述节选自 *Map Projections - A Working Manual*, USGS Professional Paper 1395, 作者 John P. Snyder。

Polyconic 投影，通常被称为 American Polyconic in Europe，因为地图上每个平行圆弧的曲率是相同的而得名。它是以纬线的特定平行线为切线的地球外切锥形，沿着其表面的平行轨迹分解而生成的。因此就涉及到很多 ("poly-") 锥形，而不是每个圆锥投影的单一锥形。

Polyconic 投影既不是等区域投影也不是等角度投影。然而沿着中心子午线，就没有失真且比例为 true。每条平行线的比例为 true，但是子午线被各种数量延长以在平行线的正确位置穿过每条平行线。所以没有平行线符合标准等角度投影（或正确角度）的意义，除了中心子午线以外。靠近中心子午线的位置，失真量极其微小。

该投影方法不适合绘制大区域地图。绘制经度范围大的地图时，转换算法将失效。例如，如果使用 Polyconic 重投影，则样本表 WORLD.TAB 可能显示异常。

基准面转换

将坐标从一个基准面转换到另一个基准面时，MapInfo 已使用的有 Molodensky（3 参数）和 Bursa-Wolf（7 参数）方法。它们是可以将坐标从任何基准面转换到任何其它的基准面的多用途方法。

在引入 NAD 83 基准面之后，NOAA 开发了名为 NADCON 的程序，是 North American Datum CONversion 的缩写。它是非常特定的程序，只能将 NAD 27 转换为 NAD 83，或将 NAD 83 转换为 NAD 27。对于这项特定的任务，它比 Molodensky 常用方法具有更高的精确性；NADCON 的精确度可以达到大约 0.1 米，而 Molodensky 的精确度只能达到 10-30 米。美国政府的大多数机构，包括人口调查局，已经将 NAD 27 和 NAD 83 之间转换的 NADCON 方法标准化。

从 MapInfo 4.1.2 开始，如果地图坐标位于 NADCON 覆盖的区域之内（美国、波多黎各和维京群岛），则使用 NADCON 算法在 NAD 27 和 NAD 83 之间转换坐标。如果坐标位于以上区域之外，或如果它们使用 NAD 27 或 NAD 83 以外的其它基准面，则 MapInfo 使用 Molodensky 或 Bursa-Wolfe 转换方法。

由于需要访问文件，NADCON 转换方法的速度将比 Molodensky 方法略微缓慢。如果要关闭 NADCON 转换，则要在 MapXtreme 程序或 MapXtreme Common 目录下删除 *.las 和 *.los 文件。

有关坐标系和投影的详细信息

以下列出的前三种出版物是相对简短的小册子。最后两种是内容充实的书籍。此外，我们还提供了 American Congress of Surveying and Mapping（小册子）和 U.S. Geological Survey（书籍）的地址和电话号码。

- American Cartographic Association. Choosing a World Map ttributes, Distortions, Classes, Aspects. Falls Church, VA: American Congress on Surveying and Mapping. Special Publication No. 2. 1988.

- American Cartographic Association. Matching the Map Projection the Need. Falls Church, VA: American Congress on Surveying and Mapping. Special Publication No. 3. 1991.
- American Cartographic Association. Which Map is Best? Projections for World Maps. Falls Church, VA: American Congress on Surveying and Mapping. Special Publication No. 1. 1986.
- John P. Snyder. Map Projections - A Working Manual. Washington: U.S. Geological Survey Professional Paper 1395. 1987
- John P. Snyder and Philip M. Voxland. An Album of Map Projections. Washington: U.S. Geological Survey Professional Paper 1453. 1989.
- 联系信息
- American Congress on Surveying and Mapping, 5410 Grosvenor Lane, Suite 100, Bethesda, MD 20814 2212; (301) 493-0200
- Earth Science Information Center, U.S. Geological Survey, 507 National Center, Reston, VA 22092; (703) 860-6045 or (800) USA-MAPS
- Austin Texas 大学地理系的 Peter H. Dana 还建立了一个关于地图投影、Geodetic 基准面和坐标系的网站。因为该网站的很多说明也是使用 MapInfo Professional 展示的，所以很有价值。网站的材料可以用于学习、研究和教育目的，但请保留作者的版权：
- Peter H. Dana, The Geographer's Craft Project, Department of Geography, The University of Texas at Austin.

关于 Geodetic 基准面的信息和说明，请访问：

<http://www.colorado.edu/geography/gcraft/notes/datum/datum.html>

关于坐标系和其它原理的信息，请访问：

<http://www.colorado.edu/geography/gcraft/notes/datum/datum.html>

关于地图投影的信息请访问：

<http://www.colorado.edu/geography/gcraft/notes/datum/datum.html>

定制基准面

基准面是地球的形状和方位的数学描述。因为地球的形状是不规则的，所以有很多不同的局部基准面用于世界的不同地区。这些局部基准面提供在特定区域与地球表面接近的近似值。

每个地球坐标系使用特定的基准面以近似表示地球的表面。将坐标从一个坐标系转换到另一个坐标系时，如果两个坐标系使用了不同的基准面，则 MapInfo 绘制的地图必须执行基准面变换。MapInfo 使用 Bursa-Wolfe 基准面变换方法，其精确度通常在 10 米之内。（当转换是在两个使用相同基准面的坐标系之间进行时，无需执行基准面变换，其结果的精确度通常在 0.1 米之内。）

定义定制基准面

大多数坐标系使用本附录列出的 MapInfo 预定义基准面之一。如果需要使用不在列表中的基准面，并知道该基准面的数学参数是什么，则可以使用定制基准面定义坐标系。MapInfo 使用以下信息定义基准面：

- 椭面，也称为球面。这是绕它的短轴旋转以构成三维表面的椭圆。该椭面由两个数学参数描述：它的短半轴长度，单位为米（由字母 a 表示）和它的修整度（由字母 f 表示）。MapInfo 支持 40 种以上的预定义椭面，在下一个表中列出。
- 三个平移参数，指定沿着其每个轴平移椭面的距离（单位为米）。这些参数通常由 dX 、 dY 和 dZ 表示，也可以由 DX 、 DY 和 DZ 或 u 、 v 和 w 表示。
- 三个旋转参数，指定绕着其每个轴旋转椭面的角度（单位为弧度 - 秒）。这些参数通常由 EX 、 EY 和 EZ 表示。也可以由 eX 、 eY 和 eZ 或 e 、 y 和 w 表示。
- 比例校正因子，指定每百万部分中的数量，以调节椭面的大小。该参数由字母 m 表示，有时也由 k 表示。
- 本初子午线的经度，以东格林威治为基准面，单位为度。本初子午线指定地球上哪个位置分配经度 0° 。大多数基准面使用格林威治作为本初子午线，所以该参数通常为 0。不过，某些基准面采用不同位置做为本初子午线。例如，NTF 基准面将巴黎做为其本初子午线，它是格林威治以东 2.33722917 度。如果在坐标系中使用 NTF 基准面，则该坐标系中所有经度都相对于巴黎，而不是格林威治。

可以在任何坐标系定义中定制自定义的基准面。使用基准面编号 9999，后跟基准面参数，按以下顺序：

```
9999, EllipsoidNumber, dX, dY, dZ, EX, EY, EZ, m, PrimeMeridian
```

某些基准面只指定椭面和平移参数 (dX 、 dY 、 dZ)，没有旋转参数、比例校正因子或本初子午线。这种情况下，可以使用基准面编号 999 替代 9999，以简化定义：

```
999, EllipsoidNumber, dX, dY, dZ
```

椭面编号必须从以下列表选择。目前尚没有方法定制自定义的椭面。如果需要使用本列表中没有出现的椭面，请通知 MapInfo 技术支持，我们可以将您的椭面增加到未来的 MapInfo 版本中。

编号	椭面	a	1/f
9	Airy 1930	6377563.396	299.3249646
13	Airy 1930 (modified for Ireland 1965)	6377340.189	299.3249646
51	ATS7 77	6378135.0	298.257
2	Australian	6378160.0	298.25
10	Bessel 1841	6377397.155	299.1528128
35	Bessel 1841 (modified for NGO 1948)	6377492.0176	299.15281
14	Bessel 1841 (modified for Schwarzeck)	6377483.865	299.1528128
36	Clarke 1858	6378293.639	294.26068
7	Clarke 1866	6378206.4	294.9786982
8	Clarke 1866 (modified for Michigan)	6378450.047484481	294.9786982
6	Clarke 1880	6378249.145	293.465
15	Clarke 1880 (modified for Arc 1950)	6378249.145326	293.4663076
30	Clarke 1880 (modified for IGN)	6378249.2	293.4660213
37	Clarke 1880 (modified for Jamaica)	6378249.136	293.46631
16	Clarke 1880 (modified for Merchich)	6378249.2	293.46598
38	Clarke 1880 (modified for Palestine)	6378300.79	293.46623
39	Everest (Brunei and East Malaysia (Sabah and Sarawak))	6377298.556	300.8017
11	Everest (India 1830)	6377276.345	300.8017
40	Everest (India 1956)	6377301.243	300.80174
50	Everest (Pakistan)	6377309.613	300.8017
17	Everest (W. Malaysia and Singapore 1948)	6377304.063	300.8017

编号	椭面	a	1/f
48	Everest (West Malaysia 1969)	6377295.664	300.8017
18	Fischer 1960	6378166.0	298.3
19	Fischer 1960 (modified for South Asia)	6378155.0	298.3
20	Fischer 1968	6378150.0	298.3
21	GRS 67	6378160.0	298.247167427
0	GRS 80	6378137.0	298.257222101
5	Hayford	6378388.0	297.0
22	Helmert 1906	6378200.0	298.3
23	Hough	6378270.0	297.0
31	IAG 75	6378140.0	298.257222
41	Indonesian	6378160.0	298.247
4	International 1924	6378388.0	297.0
49	Irish (WOFO)	6377542.178	299.325
3	Krassovsky	6378245.0	298.3
32	MERIT 83	6378137.0	298.257
33	New International 1967	6378157.5	298.25
43	NWL 10D	6378135.0	298.26
42	NWL 9D	6378145.0	298.25
44	OSU86F	6378136.2	298.25722
45	OSU91A	6378136.3	298.25722
46	Plessis 1817	6376523.0	308.64
52	PZ90	6378136.0	298.257839303
24	South American	6378160.0	298.25
12	Sphere	6370997.0	0.0

编号	椭面	a	1/f
47	Struve 1860	6378297.0	294.73
34	Walbeck	6376896.0	302.78
25	War Office	6378300.583	296.0
26	WGS 60	6378165.0	298.3
27	WGS 66	6378145.0	298.25
1	WGS 72	6378135.0	298.26
28	WGS 84	6378137.0	298.257223563

平移和旋转参数描述椭面在空间中的方位（与 WGS 84 基准面相比较）。确保这些参数含有正确的符号（正号或负号）非常重要。通常，描述局部基准面的文档会列出将坐标从该局部基准面转换为 WGS 84 所需的参数。（也就是说，这些参数是通过从 WGS 84 减去局部基准面的方法派生。）这种情况下，可以准确使用

文档中出现的参数。不过，如果文档中列出的转换坐标的参数是反方向的（从 WGS 84 转换为局部基准面），则必须将平移、旋转和比例校正因子的符号取反。

以正确的顺序列出参数也非常重要。某些文档以 EZ 为首列出旋转参数，例如：EZ, EY, EX。在这些情况下，定义定制基准面时必须将旋转参数的顺序反向。文档使用希腊字母表示参数时，这种问题尤其容易被忽略。如果文档以 w、y、e 的顺序列出参数，则必须在定制基准面的定义中将它们的顺序反向。

以下是局部基准面描述的示例（名为 LD-2），可出现在技术文章中：

LD-2 椭面：Krassovsky

a	6,378,245.0 m
f	1 / 298.3

从 LD-2 转换为 WGS 84

u	+24 m
v	-123 m
w	-94 m
w	+0.13

y	+0.25
e	-0.02
m	+1.1 ± 10-6

该基准面使用 Krassovsky 椭面，在上文的椭面表中的编号为 3。因为参数描述的是从局部基准面到 WGS 84 的转换，所以不需要将参数的符号取反。不过，旋转参数是以 w 作为首参数，所以必须在定制基准面的定义中将旋转参数的顺序反向。

9999, 3, 24, -123, -94, -0.02, 0.25, 0.13, 1.1, 0

最后的示例 LD-3，只提供了椭面和平移参数：

LD-3 椭面：Clarke 1880

a	6378249.145 m
f	1 / 293.465

WGS 84 减去局部基准面 LD-3

dX	-7 m
dY	36 m
dZ	225 m

该基准面使用 Clarke 1880 椭面，在上文的椭面表中的编号为 6。不需要将参数的符号取反或担心旋转参数的顺序（因为它们没有出现）。在这种情况下，可以在定制基准面的定义中使用基准面编号 999 替代 9999。以下两种定义是等价的，可以使用任意一种：

999, 6, -7, 36, 225
9999, 6, -7, 36, 225, 0, 0, 0, 0, 0

对于其它定制基准面的定义，会在 MapInfoCoordinateSystemSet.xml 行的基准面编号位置插入这些定义中的一个，如下所示：

```
"Longitude / Latitude (LD-3)", 1, 999, 6, -7, 36, 225

<srsName>UTM Zone 30 (LD-3)</srsName>
- <srsID>
  <code>coordsys 8, 999, 6, -7, 36, 225, 7, -3, 0, 0.9996, 500000, 0
    </code>
  <codeSpace>mapinfo</codeSpace>
  <remarks>Universal Transverse Mercator (LD-3)</remarks>
</srsID>
```

M

定义 MapInfo Codespace

MapInfo Codespace 是定义和标准的列表，常用于创建 MapInfo 地图和工作空间。可以参考这些定义以帮助使用 MapXtreme 2005。也可以将本公司的 codespace 定义与另一种常用的 codespace - European Petroleum Survey Group (EPSG) 作比较，该文档位于：

<http://www.epsg.org/>。

本附录内容：

- ◆ 定义 *MapInfo Codespace* 550

定义 MapInfo Codespace

在下表中，将说明目前的 MapInfo Codespace。这些定义将允许基于 XML 的地图文档使用本公司的 codespace 并参考公共值，例如 srsName。点几何体定义的示例可以如下所示：

```
<gml:Point srsName="mapinfo:coordsys 1,74"> ...</gml:Point> 或  
<gml:Point srsName="epsg:4269"> ...</gml:Point>
```

MapInfo Codespace 定义

类别	位置 1	位置 2	位置 3	位置 4	解释与示例
coordsys	CoordSys 文件*				坐标参考系。 示例： mapinfo:coordsys 1,74
pen	[1-255]				简单 Mapinfo 画笔图案；编号 [1-255] 基于默认的 MapInfo MapInfo.pen 定义。遵循所有已建立的规则。 示例： mapinfo:pen 46
brush	[1-8, 12-175]				简单 Mapinfo 画刷图案编号 [1-8, 12-175]。（注：9-11 是保留的。）基于默认的 MapInfo 位图定义。 示例： mapinfo:brush 17

MapInfo Codespace 定义 (续)

类别	位置 1	位置 2	位置 3	位置 4	解释与示例
length	in ft yd mi mm cm m km sft nmi li ch rd pt twip pica deg pt= 1/72 in pica = 12 pt twip = 1/20 pt 1 deg = (pi/180) * 6370997 meter = 69.09329 mi (基于大圆周)				长度单位。 支持 inch foot yard mile 和 millimeter centimeter meter kilometer 和 survey foot nautical mile 和 link chain rod point twip pica degree。 示例: mapinfo:length m mapinfo:length pt mapinfo:length deg
imagesize	pixel				图像尺寸的非线性单位。 示例: mapinfo:imagesize pixel

MapInfo Codespace 定义 (续)

类别	位置 1	位置 2	位置 3	位置 4	解释与示例
type					数据类型。
	boolean				示例: mapinfo:type boolean
	byte				示例: mapinfo:type byte
	date				示例: mapinfo:type date
	decimal	[(n	n)]	n.m n 是总的位数 m 是小数点后保留的位数 m<=n	最小值可选 最大值可选 精度可选 示例: mapinfo:type decimal (100 -- 所有 d>100 mapinfo:type decimal 100) -- 所有 d<100 mapinfo:type decimal [200 300] -- 所有 200<=d<=300 mapinfo:type decimal 10.7 -- 所有总长度不超过 10 位的小数, 小数点后不超过 7 位, 那么整数部分最多有 3 位。 mapinfo:type decimal [200 300] 10.7 -- 前两种示例的结合。
	double	[(n	n)]		示例: mapinfo:type double (0 150)
	float	[(n	n)]		示例: mapinfo:type float 30.25]
	int	[(n	n)]		示例: mapinfo:type int [0 180]
	short	[(n	n)]		示例: mapinfo:type short
	string	n			最大长度 n 可选。 示例: mapinfo:type string 256
	char				示例: mapinfo:type char

MapInfo Codespace 定义 (续)

类别	位置 1	位置 2	位置 3	位置 4	解释与示例
operators					标量属性比较运算符或几何体运算符或布尔运算符。
	eq neq lt gt lteq gteq				二进制标量值运算符。 示例: mapinfo:op eq
	in not_in				在标量值枚举设置中的标量值 in/not, 例如 value in {2, 50, 88, 95} value not_in { "NY" , "NJ" }。 示例: mapinfo:op in mapinfo:op not_in
	between not_between				标量值是在两个基于该标量值类型的顺序相关标量值之间, 例如, value between {5, 25} value not_between {1may2000, 30may2000} 示例: mapinfo:op between mapinfo:op not_between
	like not_like				字符串值匹配 RHS 图案。RHS 图案的格式可以指定为数据源, 例如, lastname like "Jo%" 。(使用 Oracle 通配字符 '%') 。 示例: mapinfo:op like mapinfo:op not_like
	intersects				图元几何体 fg 和给定的 Polygon 相交。 示例: mapinfo:op intersects
	mbr_intersects				图元几何体 (fg) 的 mbr (最小绑定矩形) 和给定的 Polygon 的 mbr 相交。 示例: mapinfo:op mbr_intersects

MapInfo Codespace 定义 (续)

类别	位置 1	位置 2	位置 3	位置 4	解释与示例
operators (续)	contains				图元几何体 fg 包含给定的 Point。给定的 Polygon 包含图元几何体 fg。 示例: mapinfo:op contains
	contains_centroid				图元几何体 fg 包含给定的 Polygon 的质心。给定的 polygon 包含图元几何体 (fg) 的质心 示例: mapinfo:op contains_centroid
	and				布尔 and。 示例: mapinfo:op and
	or				布尔 or。 示例: mapinfo:op or

MapInfo Codespace 定义 (续)

类别	位置 1	位置 2	位置 3	位置 4	解释与示例
area					面积单位。
	sq	in ft yd mi mm cm m km sft nmi li ch rd pt twip pica degree			平方线性单位。 示例: mapinfo:area sq mi
	acre a ha perch rood a = are (10m x10m 的 面积) ha = hectare (100m x 100m 的面积) perch = 272.25 sq ft (1 平方 rod 16.5 ft) rood = 40 perch (1/4 acre)				预定义面积单位。 示例: mapinfo:area acre
time	msec sec min hr day week month year				时间单位。 Millisecond Second Minute, Hour Day Week Month Year 示例: mapinfo:time hr
angle	deg rad				角度度量单位。 degree radian 示例: mapinfo:angle rad
temp	K F C				温度单位。 Kelvin Fahrenheit Celcius 示例: mapinfo:temp C

* MapXtreme 2005 的坐标系信息在 MapInfoCoordinateSystemSet.xml 文件中，该文件位于 C:\Program Files\Common Files\MapInfo\MapXtreme\6.x 目录下。

N

示例数据的版权

以下资源提供了和 MapXtreme 2005 一起发布的数据。关于 MapInfo 产品的详细信息，请参阅“MapInfo Data Products Catalog”或联系 MapInfo Corporation。

地图名称	版权所有
阿根廷 MWS/Geoset	
阿根廷城市	Digital Chart of the World 的 MapInfo
阿根廷国境线	Digital Chart of the World 的 MapInfo
阿根廷主要城市	Digital Chart of the World 的 MapInfo
南美洲国境线	Digital Chart of the World 的 MapInfo
世界海洋 (经度 / 纬度)	MapInfo
亚洲 MWS/Geoset	
亚洲	Digital Chart of the World 的 MapInfo
亚洲首都	Digital Chart of the World 的 MapInfo
亚洲主要城市	Digital Chart of the World 的 MapInfo
欧洲国境线	MapInfo 1998
世界海洋 (亚洲)	MapInfo
澳大利亚 MWS/Geoset	
澳大利亚城市	Digital Chart of the World 的 MapInfo
澳大利亚高速公路	Digital Chart of the World 的 MapInfo
澳大利亚主要城市	Digital Chart of the World 的 MapInfo
澳大利亚州界	MapInfo
澳大利亚州首府	Digital Chart of the World 的 MapInfo
世界海洋 (经度 / 纬度)	MapInfo
巴西 MWS/Geoset	
巴西主要城市	Digital Chart of the World 的 MapInfo
巴西城市	Digital Chart of the World 的 MapInfo
巴西国境线	Digital Chart of the World 的 MapInfo
南美洲国境线	Digital Chart of the World 的 MapInfo

地图名称	版权所有
世界海洋 (经度 / 纬度)	MapInfo
加拿大 MWS/Geoset	
加拿大城市	Digital Chart of the World 的 MapInfo
加拿大高速公路	Digital Chart of the World 的 MapInfo
加拿大主要城市	Digital Chart of the World 的 MapInfo
加拿大省界	Digital Chart of the World 的 MapInfo
加拿大省首府	Digital Chart of the World 的 MapInfo
美国州界	Digital Chart of the World 的 MapInfo
世界海洋 (经度 / 纬度)	MapInfo
中国 MWS/Geoset	
亚洲	Digital Chart of the World 的 MapInfo
亚洲主要城市	Digital Chart of the World 的 MapInfo
中国城市	Digital Chart of the World 的 MapInfo
中国国境线	Digital Chart of the World 的 MapInfo
中国高速公路	Digital Chart of the World 的 MapInfo
中国主要城市	Digital Chart of the World 的 MapInfo
世界海洋 (亚洲)	MapInfo
Dallas, TX Geoset	
Crime 郡地图	MapInfo
Crime 演示地图	MapInfo
Dallas 城市边界 A	USGS 的 MapInfo
Dallas 城市边界 B	USGS 的 MapInfo
Dallas 位置	MapInfo
Dallas 主要高速公路 A	USGS 的 MapInfo

地图名称	版权所有
Dallas 主要高速公路 B	USGS 的 MapInfo
Dallas 街道 A	USGS 的 MapInfo
Dallas 街道 B	USGS 的 MapInfo
Dallas 街道 C	USGS 的 MapInfo
Dallas 水体	USGS 的 MapInfo
Dallas 河流	USGS 的 MapInfo
Dallas, TX Geoset	版权所有 © 1997, Horizons Technology, Inc.

DC MWS/Geoset

Dc 区域路标	MapInfo
Dc 城市边界	MapInfo
DC 高速公路	Bureau of Transportation Statistics 的 MapInfo
DC 州际公路	Bureau of Transportation Statistics 的 MapInfo
DC 路标	MapInfo
DC 公路	Bureau of Transportation Statistics 的 MapInfo
DC 州公路	Bureau of Transportation Statistics 的 MapInfo
Dc 水层	MapInfo
DC 邮政编码	© 1995 Geographic Data Technology, Inc.

欧洲 MWS/Geoset

亚洲	Digital Chart of the World 的 MapInfo
欧洲首都	Digital Chart of the World 的 MapInfo
欧洲城市	Digital Chart of the World 的 MapInfo
欧洲国境线	MapInfo 1998
欧洲高速公路	MapInfo 1998
欧洲主要城市	Digital Chart of the World 的 MapInfo

地图名称	版权所有
世界海洋 (经度 / 纬度)	MapInfo
法国 MWS/Geoset	
欧洲国境线	MapInfo 1998
法国城市	Digital Chart of the World 的 MapInfo
法国高速公路地图	MapInfo 1998
法国主要城市	Digital Chart of the World 的 MapInfo
世界海洋 (经度 / 纬度)	MapInfo
德国 MWS/Geoset	
欧洲国境线	MapInfo 1998
德国城市	Digital Chart of the World 的 MapInfo
德国高速公路	MapInfo 得到的 GISdata Limited GisNET 数据许可。 © GDC Ltd. 1993
德国主要城市	Digital Chart of the World 的 MapInfo
世界海洋 (经度 / 纬度)	MapInfo
印度 MWS/Geoset	
亚洲	Digital Chart of the World 的 MapInfo
印度首都城市	Digital Chart of the World 的 MapInfo
印度区域边界	Risk Management Solutions, Inc.
印度主要城市	Digital Chart of the World 的 MapInfo
印度次要城市	Digital Chart of the World 的 MapInfo
印度州界	Risk Management Solutions, Inc.
世界海洋 (亚洲)	MapInfo
意大利 MWS/Geoset	
欧洲国境线	MapInfo 1998
意大利城市	Digital Chart of the World 的 MapInfo

地图名称	版权所有
意大利高速公路	MapInfo 得到的 GISdata Limited GisNET 数据许可。 © GDC Ltd. 1993
意大利主要城市	Digital Chart of the World 的 MapInfo
世界海洋 (经度 / 纬度)	MapInfo

日本 MWS/Geoset

日本 Cased 公路	MapInfo 得到的 GISdata Limited GisNET 数据许可。 © GDC Ltd. 1993
日本城市	Digital Chart of the World 的 MapInfo
日本国家背景	MapInfo 得到的 GISdata Limited GisNET 数据许可。 © GDC Ltd. 1993
日本高速公路	MapInfo 得到的 GISdata Limited GisNET 数据许可。 © GDC Ltd. 1993
日本主要城市	Digital Chart of the World 的 MapInfo
日本河流湖泊	MapInfo 得到的 GISdata Limited GisNET 数据许可。 © GDC Ltd. 1993
世界海洋 (经度 / 纬度)	MapInfo

墨西哥 MWS/Geoset

墨西哥城市	Digital Chart of the World 的 MapInfo
墨西哥高速公路	Digital Chart of the World 的 MapInfo
墨西哥主要城市	Digital Chart of the World 的 MapInfo
墨西哥州界	Bureau of Transportation Statistics 的 MapInfo
墨西哥州首府	Digital Chart of the World 的 MapInfo
美国州界	Digital Chart of the World 的 MapInfo
世界海洋 (经度 / 纬度)	MapInfo

大西洋中部 MWS/Geoset

大西洋中部首都	Digital Chart of the World 的 MapInfo
---------	--------------------------------------

地图名称	版权所有
大西洋中部城市	Digital Chart of the World 的 MapInfo
大西洋中部国家	MapInfo
大西洋中部高速公路	Bureau of Transportation Statistics 的 MapInfo
大西洋中部主要城市	Digital Chart of the World 的 MapInfo
大西洋中部州	Digital Chart of the World 的 MapInfo
世界海洋 (经度 / 纬度)	MapInfo
美国州界	Digital Chart of the World 的 MapInfo

北美洲 MWS/Geoset

加拿大城市	Digital Chart of the World 的 MapInfo
加拿大高速公路	Digital Chart of the World 的 MapInfo
加拿大主要城市	Digital Chart of the World 的 MapInfo
加拿大省界	Digital Chart of the World 的 MapInfo
加拿大省首府	Digital Chart of the World 的 MapInfo
墨西哥城市	Digital Chart of the World 的 MapInfo
墨西哥高速公路	Digital Chart of the World 的 MapInfo
墨西哥主要城市	Digital Chart of the World 的 MapInfo
墨西哥州界	Bureau of Transportation Statistics 的 MapInfo
墨西哥州首府	Digital Chart of the World 的 MapInfo
美国城市	Digital Chart of the World 的 MapInfo
美国郡界	Census Bureau 的 MapInfo
美国高速公路	Bureau of Transportation Statistics 的 MapInfo
美国主要城市	Digital Chart of the World 的 MapInfo
美国州界	Digital Chart of the World 的 MapInfo
美国州首府	Digital Chart of the World 的 MapInfo

地图名称	版权所有
世界海洋 (经度 / 纬度)	MapInfo
葡萄牙 MWS/Geoset	
欧洲国境线	MapInfo 1998
葡萄牙高速公路	MapInfo 得到的 GISdata Limited GisNET 数据许可。 © GDC Ltd. 1993
葡萄牙主要城市	Digital Chart of the World 的 MapInfo
世界海洋 (经度 / 纬度)	MapInfo
韩国 MWS/Geoset	
亚洲	Digital Chart of the World 的 MapInfo
韩国城市	Digital Chart of the World 的 MapInfo
韩国国境线	Digital Chart of the World 的 MapInfo
韩国主要城市	Digital Chart of the World 的 MapInfo
世界海洋 (亚洲)	MapInfo
西班牙 MWS/Geoset	
欧洲国境线	MapInfo 1998
西班牙城市	Digital Chart of the World 的 MapInfo
西班牙高速公路	MapInfo 得到的 GISdata Limited GisNET 数据许可。 © GDC Ltd. 1993
西班牙主要城市	Digital Chart of the World 的 MapInfo
世界海洋 (经度 / 纬度)	MapInfo
StreetPro 示例	
StreetPro 比利时	2004 Tele Atlas B.V. 's-Hertogenbosch. 保留所有权利。
StreetPro 德国	2004 Tele Atlas B.V. 's-Hertogenbosch. 保留所有权利。
StreetPro 西班牙	2004 Tele Atlas B.V. 's-Hertogenbosch. 保留所有权利。
StreetPro 法国	2004 Tele Atlas B.V. 's-Hertogenbosch. 保留所有权利。

地图名称	版权所有
StreetPro 英国	2004 Tele Atlas B.V. 's-Hertogenbosch. 保留所有权利。
StreetPro 意大利	2004 Tele Atlas B.V. 's-Hertogenbosch. 保留所有权利。
StreetPro 荷兰	2004 Tele Atlas B.V. 's-Hertogenbosch. 保留所有权利。
StreetPro 瑞典	2004 Tele Atlas B.V. 's-Hertogenbosch. 保留所有权利。

英国 MWS/Geoset

欧洲国境线	MapInfo 1998
英国城市	Digital Chart of the World 的 MapInfo
英国 A 级公路	MapInfo 得到的 GISdata Limited GisNET 数据许可。 © GDC Ltd. 1993
英国高速公路	MapInfo 得到的 GISdata Limited GisNET 数据许可。 © GDC Ltd. 1993
英国标准区域	版权所有 URPI 1998
世界海洋 (经度 / 纬度)	MapInfo

美国 MWS/Geoset

Urban_CitiesPop10K_Plus	MapInfo
USA_Caps	Digital Chart of the World 的 MapInfo
US_Int_Shields1	MapInfo
US_Int_Shields2	MapInfo
US_Int_Shields3	MapInfo
Interstates	US Department of the Interior: nationalatlas.gov
Secondary_Rds	US Department of the Interior: nationalatlas.gov
Streams_Rivers	US Department of the Interior: nationalatlas.gov
Lakes	US Department of the Interior: nationalatlas.gov
Urban_Areas	US Department of the Interior: nationalatlas.gov
Wilderness_Areas	US Department of the Interior: nationalatlas.gov

地图名称	版权所有
USA_Outline	MapInfo
世界 MWS/Geoset	
世界首都	Digital Chart of the World 的 MapInfo
世界国家	Digital Chart of the World 的 MapInfo
世界网格图	MapInfo
世界海洋 (Robinson)	MapInfo
世界前 25 位城市	Digital Chart of the World 的 MapInfo
Mgrs_100K	MapInfo
Mgrs_Nor_100K	MapInfo
Mgrs_sou_100K	MapInfo
Mgrs_UPS_Nor	MapInfo
Mgrs_UPS_Sou	MapInfo
MGRS_Zones	MapInfo
其它文件	
MapXsite Streets 1.0 的 Dallas 城市边界	USGS 的 MapInfo
MapXsite Streets 1.0 的 Dallas 主要高速公路	USGS 的 MapInfo
MapXsite Streets 1.0 的 Dallas 街道	USGS 的 MapInfo
MapXsite Streets 1.0 的 Dallas 水体	USGS 的 MapInfo
MapXsite Streets 1.0 的 Dallas 河流	USGS 的 MapInfo
美国 5 位邮政编码点	(c) 2005 GDTTeleAtlas
美国 5 位邮政编码点 (压缩点文件)	(c) 2005 GDTTeleAtlas
MapStats.mdb	
亚洲人口统计	版权所有 United Nation Demographic Yearbook, 1994
澳大利亚人口统计	MapInfo 澳大利亚

地图名称	版权所有
DC 人口统计	MapInfo 加拿大
大西洋中部州人口统计	MapInfo 加拿大
世界人口统计	版权所有 United Nation Demographic Yearbook, 1994
美国人口统计	MapInfo 加拿大
DC 客户数据库	MapInfo
美国客户数据库	MapInfo

索引

A

Abs 函数 370
Acos 函数 370
ADO.NET 接口
 分析数据 91
ADRG
 支持的光栅格式 295
Ascii 函数 370
Asin 函数 371
ASP.NET 应用程序
 创建但不使用 MapXtreme 2005 模板 33
 会话管理 138–139
 使用 ASP.NET 模板 30, 32
 Web 控件 32
ASRP
 支持的光栅格式 295
Atan 函数 371
Avg 函数 371
Azimuthal 投影 541
案例研究
 管理状态 440
安装
 程序 20–24
 系统要求 16
 先决条件 19
安装程序
 部署应用程序 34, 37
安装的位图图像
 访问 275

B

.bil 文件
 支持的光栅格式 295
BitmapPointStyleRepository 类
 定制 453
.bmp 文件
 支持的光栅格式 295
北偏移量 541

比例栏
 修饰 227
比例因子
 Transverse Mercator 投影 541
变化
 网格图像 273
编辑
 LineStrings 284
表
 别名 149
 打包 154
 打开 151
 地图绘制术语说明 41
 定义 351
 关闭 154
 关联 86, 88–90
 结果集 90
 可制图 162
 列 149–150
 MI_Geometry 列 150
 MI_Key 列 150
 MI_Style 列 150
 目录 151
 事件 155
 视图 89
 添加表达式列 160
 相关 85
表, 创建
 本地的 157
 临时 MemTable 159
 临时本地的 158
表达式
 AddColumns 180
 标记优先权 338–339
 标注 182
 布尔 179
 创建 178–182
 函数 179

- InfoTip **182**
MapInfo SQL 定义 **361**
示例 **180–182**
使用 **179**
搜索图元 **181**
在缓冲区外选择对象 **180**
在缓冲区中选择对象 **180**
主题 **181**
- 标记**
 Workspace 管理器 **337–338**
 样式 **338**
 优先权表达式 **338–339**
- 标注**
 单值主题 **242**
 地图绘制术语说明 **42**
 范围主题 **240**
 概述 **223**
 UML 图 **223**
- 标准结线**
 圆锥投影 **540**
- 布尔表达式** **179**
- 部署**
 web 应用程序 **436**
 桌面应用程序 **436**
- 部署应用程序**
 安装程序 **34, 37**
- C**
- CADRG**
 支持的光栅格式 **295**
- Ceil 函数** **372**
- Char 函数** **372**
- Char_Length 函数** **372**
- CIB**
 支持的光栅格式 **295**
- codespace**
 定义 **350**
- codespace 定义** **550–555**
- COM 对象模型**
 .NET 可互操作性 **58**
- Cos 函数** **373**
- Count 函数** **373**
- CPU**
 性能 **72**
- CreateThemeWizard**
 Windows 对话框 **130–131, 133–134**
- Current_Date 函数** **373**
- Curve**
 创建 **287**
- 参数**
 投影 **526**
- 操作系统**
 支持 **16**
- 查询**
 以 QueryDefinition 类定制 **448**
- 查询类型** **175**
- 查找地图图元**
 错误的地址元素 **268**
 街道地址 **252–254**
 十字路口 **252**
 位置名称 **252**
 优化边界表 **254**
- 产品功能**
 发生变化的内容 **9**
 新增内容 **4**
 主要功能 **3**
- 产品体系结构** **48**
- 常用对话框**
 由 MapXtreme 提供 **130**
- 常用样式**
 默认样式 **274**
- 超文本标记语言 (HTML)**
 定义 **351**
- 超文本传输协议 (HTTP)**
 定义 **351**
- 程序集**
 重定向到新的 **26–27**
- 池大小**
 配置 **70**
- 持续**
 保存信息 **454**
- 重定向程序集** **26–27**
- 创建**
 LineStrings **284**
 MapInfo Map Catalog **206**
 MapXtreme 2005 应用程序 **421**
 web 应用程序 **60–71, 429–430**
 桌面应用程序 **58–60, 422, 425–428**

创建标注主题**单值** **242****范围** **240****创建地图****MapFactory** **218****创建工作空间****编程方式** **466****通过 Workspace 管理器** **324****创建几何体****多边形** **288****环** **288****LineStrings** **284****MultiPoint 对象** **282****Points** **282****曲线** **287****创建图层****UserDrawLayer** **类** **450****创建应用程序****不使用模板** **33****在 Visual Studio .NET 中** **27–30, 32****创建主题****饼图** **235****单值** **241****点密度** **243–244****范围** **238****分级符号** **234****条形图** **237****图例** **246****D****Data.Find 命名空间****概述** **50****UML 图表** **255****Data.TableInfo****UML 图** **156****DateToNumber 函数** **374****DateToString 函数** **374****Day 函数** **374****DBMS 数据库****访问远程表** **194****将空间表导入** **206****Delete 命令****MapInfo SQL** **示例** **361****.dem 文件****支持的网格格式** **299****DTED 文件****支持的网格格式** **300****打包****web 应用程序** **433–434****桌面应用程序** **431–433****打包表** **154****打开表****目录** **151****单值标注主题****创建** **242****单值主题****创建** **241****导出****地图到图像** **216****导入****XML 文件的图元** **454****地理编码****定义** **352****概述** **45****显示地图上的数据** **45****地理标记语言 (GML)****定义** **352****地理信息系统 (GIS)****定义** **352****笛卡儿坐标****定义** **352****地图****地图绘制术语说明** **40****制图图例** **247****地图标题****修饰** **227****地图绘制术语和概念** **40–41, 43–44****地图类****地图内容** **217****地图视图****View 工具** **112****地图图例** **245****对齐框架** **248****地图应用程序模板****快速原型开发** **59****地址****查找结果代码** **261****用 Find 匹配** **252****地址范围****Find 类结果** **259**

地址号码
 用 Find 匹配 254

点
 创建 282

点密度主题
 创建 243–244

点样式 495
 BasePointStyle 类 274
 将位图图像用作 275
 在远程空间表中 210

定义可制图的表 201–202

定制
 对话框 135
 EngineCustomProperties 类 448
 FeatureOverrideStyleModifier 449
 FeatureStyleModifier 449
 GMLFeatureCollection 454
 工具 453
 搜索函数 448
 web 工具（版本 6.5 之前） 509
 Windows 工具 115
 Windows 控件 451–452
 WorkSpaceLoader 454
 UserDrawLayer 450
 样式 453

定制工具
 创建 99
 行为 113

定制位图
 点样式 272

动态选择 112

东伪偏移 541

对话框
 CreateThemeWizard 130–131, 133–134
 定制 135
 添加至应用程序 129
 预构建 59

对象池
 测试案例 74
 会话对象 71
 InProc 62–63
 开发模型 63
 优势 63

对象模型
 MapXtreme 2005 和 MapX 412

命名空间 49–51
实现 412
数据访问 79–86, 88–92
体系结构 49

对象主题 233
饼图 235
分级符号 234
条形图 237

多边形
 创建 288

E

ECW 文件
 支持的光栅格式 295

.emf 文件
 支持的光栅格式 295

Engine 命名空间
 概述 50
 异常类 146

Excel
 代码示例 84

ExecuteFeatureCollection 173

Exp 函数 375

F

FeatureCollection 184

FeatureGeometry 183

FeatureLayer
 定义 350

FeatureOverrideStyleModifier
 定制 449
 覆盖样式 277
 重载样式 230

FeatureStyleModifier
 定制 449
 更改图层外观 449
 UML 图 229
 修改样式 229

Find 类
 返回对象 259, 261
 示例应用程序 256
 要求 255

Floor 函数 375

- 范围**
在投影中 **541**
- 范围标注主题** **240**
 创建 **240**
- 范围主题**
 创建 **238**
 分布类型 **238–239**
- 访问不可制图的数据** **455**
- 访问数据**
 IDBCommand **168**
- 访问远程表**
 不使用 .tab 文件 **195**
 DataTable **166**
 DBMS 数据库 **194**
 定义可制图的表 **201**
 缓存管理 **204–205**
 IDBCommand **167**
 可制图的表 **201–202**
 MapInfo Map Catalog **206**
 ODBC 连接字符串格式 **199**
 使用 .tab 文件 **194**
 属性数据 **203**
 性能 **92, 203**
 映射到 X/Y 列 **195**
 指定样式 **210**
- 非地球地图**
 定义 **350**
- 分布式应用程序**
 特许 **437**
- 分发 web 控件** **108**
- 分级符号主题**
 创建 **234**
- 分析数据** **90**
 ADO.NET 接口 **91**
 MI 命令 **91**
 目录 **90–91**
 OGC 查询接口 **91**
- 覆盖样式** **277**
- 符号样式**
 在远程空间表中 **210**
- 符号字体设置** **495**
- 服务器表**
 缓存数据 **204**
- 服务器表查询**
 定义可制图的表 **201–202**
- 辅助底纹**
 请参阅 hillshading
- G**
- .gen 文件**
 支持的光栅格式 **295**
- Geocoding 命名空间**
 概述 **52**
- GeoDictionary** **169–170**
 定义 **350**
 实用程序 **342, 344, 346**
- GeoDictionary 管理器** **346**
 定义 **350**
- Geometry 类** **280–281**
- Geometry 命名空间**
 概述 **50**
 UML 图 **281**
- GeoTIFF 文件**
 支持的光栅格式 **295**
- GIF 文件**
 支持的光栅格式 **295**
- GmlFeatureCollection 类** **454**
 定制 **454**
- .grc 文件**
 支持的光栅格式 **295**
- .grd 文件**
 支持的光栅格式 **295**
 支持的网格格式 **300**
- group by**
 MapInfo SQL 定义 **361**
- GTOPO30 文件**
 支持的网格格式 **299**
- 更新到 MapXtreme 2005**
 web 应用程序 **25–26**
 桌面应用程序 **25**
- 工具**
 创建自定义（版本 6.5 之前） **509**
 定制 **115, 453**
 工具事件 **116–117**
 设计定制 **113**
 web 体系结构 **96**
 web 应用程序 **96**
 web 应用程序（版本 6.5 之前） **508**
 Workspace 管理器 **330–332**
 桌面应用程序 **115**

- 工作空间
 - 定义 **352**
 - 加载 **70**
 - 结构 **461–462, 464–465**
 - 通过编程方式创建 **466**
 - XML 格式 **325**
- 构建
 - web 应用程序 **60–71**
 - 桌面应用程序 **58–60**
- 关闭表 **154**
- 关键字
 - Mapinfow.abb 文件替换 **264**
- 管理状态
 - 案例研究 **440**
- 关联表 **86, 88–90**
- 光栅 / 网格函数
 - MapInfo SQL 语言 **369**
- 光栅句柄 **297**
- 光栅图像
 - 添加到地图 **296**
 - 限制 **296**
 - 样式 **273**
 - 支持的文件格式 **295**
 - 主要的类 **295–296**
 - 坐标系 **296**
- 规划
 - 数据访问 **79–86, 88–92**
- 规划应用程序
 - web **60–71**
 - 桌面 **58–60**
- H**
 - hillshading**
 - 网格参数 **273**
 - Hotine Oblique Mercator 投影**
 - 倾斜方位角 **541**
 - 函数**
 - MapInfo SQL 支持 **359, 365, 367–369**
 - 合并模块**
 - 创建安装程序 **35, 37**
 - 画笔样式** **494**
 - 在远程空间表中 **210**
 - 画刷样式**
 - 在远程空间表中 **210**
- I**
 - IIS**
 - 调节应用程序 **75–76**
 - 服务器设置 **75–76**
 - 应用程序池 **75**
 - IIS 应用程序池设置** **75**
 - InfoTool**
 - web 工具（版本 6.5 之前） **508**

InProc 应用程序
 开发模型 **62**
 有对象池 **62**

Insert 命令
 MapInfo SQL 示例 **360**

InStr 函数 **375**

InteriorStyleRepository 类
 定制 **453**

Internet 信息服务 (IIS)
 定义 **350**

I SearchResultProcessor 接口 **449**

ISession 接口
 定义 **350**
 会话管理 **138**
 UML 图表 **138**

J

JPEG 文件
 支持的光栅格式 **295**

JPEG2000 文件
 支持的光栅格式 **295**

集合函数
 MapInfo SQL 语言 **368–369**

几何体
 坐标系 **290**

几何体测量函数
 MapInfo SQL 语言 **368**

几何体对象
 转换为 FeatureGeometry 对象 **287**

几何体构造函数
 MapInfo SQL 语言 **367**

几何体观测函数
 MapInfo SQL 语言 **367–368**

几何体函数
 MapInfo SQL 语言 **368**

几何体集合函数
 MapInfo SQL 语言 **369**

几何体列
 可制图的表 **201**

几何体谓词函数
 MapInfo SQL 语言 **367**

几何质心
 定义 **352**

技术支持, 联系 **14**

基准面
 定义定制 **544–545, 547–548**
 列表 **531–538**
 投影 **531**
 转换坐标 **542**

加载数据
 MapLoader **218**

检索图元 **184**

教程
 创建 web 应用程序 **429–430**
 创建桌面应用程序 **422, 425–428**
 打包 web 应用程序 **433–434**
 打包桌面应用程序 **431–433**

街道地址
 查找结果代码 **261**
 用 Find 匹配 **252**

街道名称
 用 Find 匹配 **252**

街道缩写
 用 Find 匹配 **253–254**

结果代码 **262**

结果集 **90**
 select 语句 **90**

结果属性
 查找匹配 **261**

接近的匹配
 Find 类结果 **260**

经度
 定义 **352**

经度 / 纬度
 定义 **352**

经度的度
 定义 **352**

距离工具
 web 工具 (版本 6.5 之前) **509**

矩形工具
 绘制矩形 **115**

K

Key
 MI_Key 列 **150**

开发框架支持 **16**

开发模型
 InProc **62**

可制图的表

- 将行添加到 Map Catalog **208–209**
- 使表 **162**
- 使用服务器表查询进行定义 **201–202**

控件

- web **94–103, 105–108**
- web (版本 6.5 之前) **502**
- 桌面 **59**

空间构架

- 定义 **353**
- PointRef **87, 162–163**
- XY 构架 **162**

空间运算符

- MapInfo SQL 语言 **358**

库中的类 **453****L****LabelLayer** **223****LabelModifier** **224****LabelProperties** **224****LabelSource** **223****LayerControl**

- 创建定制选项卡 **451**
- 定制 **451**
- 图层节点 **451–452**
- web 控件 (版本 6.5 之前) **506**
- Windows 控件 **128–129**
- Workspace 管理器 **332–337**
- 样式 **276**

LayerNodeHelper 类

- 定制 **451**

Left 函数 **376****LegendControl**

- web 控件 (版本 6.5 之前) **506**

LineStrings

- 创建 **284**

LineStyleRepository 类

- 定制 **454**

Log 函数 **376****Lower** 函数 **376****Ltrim** 函数 **377****类**

- 定义 **353**

联系技术支持 **14****列**

- 添加 **180**
- 添加表达式 **160**
- 支持的数据类型 **149–150**

路径规划

- 概述 **45**

M**MapBasic SQL** 语法

- v. MapInfo SQL 语言 **358**

MapControl

- 定义 **350**
- 规划桌面应用程序 **59**
- web 工具 (版本 6.5 之前) **508**
- web 控件 (版本 6.5 之前) **504**
- Windows 控件 **124, 219**
- 在表单元格中 **101**

MapExport

- 导出到图像 **216**

MapFactory

- 创建地图 **218**

MapForm 应用程序

- 使用 MapForm 模板 **28–29**
- Windows 控件 **32**

MapInfo 3.0 兼容字体

- 绘制点 **274**
- VectorSymbolRepository 类 **275**

MapInfo ADO.NET 数据提供方 **79**

- MICommand **171–172**
- MIConnection **170**
- 数据访问 **170**

数据源 **81****MapInfo Codespace**

- 定义 **350**

MapInfo codespace 定义 **550–555****MapInfo Grid** 文件

- 支持的网格格式 **299**

MapInfo Map Catalog

- 创建 **206**
- 添加行 **208–210**

MapInfo SQL 语言 **174**

- 参考 **91**
- 查询类型 **175**
- MIDataReader **173**
- 命令 **360**

日期格式化字符串 **406–407**
数字格式化字符串 **408–409**
v. MapBasic SQL **358**
运算符 **362, 364**

MapInfo Workspace 格式 (.MWS) 324

MapInfoCoordinateSystemSet.xml 文件
投影 **529**
投影 ID 编号 **527**
坐标单位 **539**

Mapinfo.abb 文件
编辑 **263–264, 268**
替换方法 **265–267**
替换关键字 **264**
与街道缩写匹配 **253–254**

MapLoader
加载数据 **218**

Mapping 命名空间
概述 **50, 216**
UML 图 **216**

Mapping.Legends 命名空间
概述 **50**

Mapping.Thematics 命名空间
概述 **51**
UML 图 **232–233**

MapPrinting
类 **472**

MapToolBar
Windows 控件 **126–127**

MapTools
体系结构 **96**
用于桌面应用程序 **60**

MapView
视图的列表 **218**

MapX 5.0
迁移到 MapXtreme 2005 **413–414, 416–420
411**

MapXtreme 2004
web 控件的体系结构 **94**

MapXtreme 2004 web 控件
迁移到 MapXtreme 2005 **106**

MapXtreme 2005
变化的功能 **9**
产品概述 **2**
创建应用程序 **421**
对象模型 **49–51, 412**

功能概述 **3**
迁移到 **11**
升级 **25–27**
体系结构概述 **48**
web 控件的体系结构 **94**
新增功能 **4**
支持资源 **13**

MapXtreme 2005 ASP.NET Web 应用程序
构建但不使用模板 **33**
使用模板 **30**

MapXtreme 2005 web 控件
从 MapXtreme 2004 迁移 **106**

MapXtreme 2005 Windows 地图应用程序
使用模板 **28**

Max 函数 377

MI 连接
数据访问 **80**

MI 命令
分析数据 **91**
数据访问 **80**

MI_AggregateBuffer 函数 377

MI_AggregateConvexHull 函数 378

MI_AggregateEnvelope 函数 378

MI_AggregateIndividual 函数 378

MI_AggregateIntersection 函数 379

MI_AggregateUnion 函数 379

MI_Area 函数 379

MI_Box 函数 382

MI_Buffer 函数 381–382

MI_Centroid 函数 383

MI_CentroidWithin 函数 384

MI_CentroidX 函数 384

MI_CentroidY 函数 384

MI_Contains 函数 385

MI_ContainsCentroid 函数 385

MI_ConvexHull 函数 385

MI_EngineVersion 函数 386

MI_Envelope 函数 386

MI_EnvelopesIntersect 函数 386

MI_Geometry 列 150

MI_GeometryType 函数 387

MI_Grid.MaxValue 函数 387

MI_Grid.MinValue 函数 388

MI_Grid.getValueAtPixel 函数 388

MI_ImageFile 函数 388

- MI_ImagePixelHeight 函数 389**
MI_ImagePixelWidth 函数 389
MI_Intersection 函数 389
MI_Intersects 函数 390
MI_Key 列 150
MI_Length 函数 390
MI_MaxX 函数 391
MI_MaxY 函数 391
MI_MinX 函数 391
MI_MinY 函数 392
MI_Perimeter 函数 392
MI_Point 函数 393
MI_SQLVersion 函数 393
MI_Style 列 150
MI_Transform 函数 393
MI_Within 函数 394
MI_Union 函数 394
MI_X 函数 394
MI_Y 函数 395
MICommand 172
 SQL 语句 171
MIConnection
 MapInfo ADO.NET 数据提供方 170
Microsoft Data Access Components (MDAC)
 部署应用程序 38
MIDataReader 173
Min 函数 395
Mod 函数 395
Month 函数 396
MrSID 文件
 支持的光栅格式 295
MSI 安装程序
 创建安装程序 35, 37
MultiPoint 对象
 创建 282
.mws
 MapInfo Workspace 格式 324
MXTRunNCP.exe
 请参阅运行时安装程序。
每记录样式 212
枚举
 定义 353
命名空间 49–51
 定义 353
- 命名连接**
 工作空间永久性 460
 类型 326
- 模板**
 ASP.NET 30, 32
 MapForm 28–29
 用于 MapXtreme 项目 58
- 默认样式**
 组合样式 272
- 目录**
 表的注册表 151
 分析数据 90–91
 数据访问 79, 81–85
 搜索方法 185
 相关表 85
- N**
- NAD 83 基准面**
NADCON
 在 NAD 27 和 NAD 83 之间转换 542
- .NET 对象模型**
 COM 可互操作性 58
- NITF**
 支持的光栅格式 295
- North American Datum Conversion (NADCON)**
 参阅 NADCON
- NumberToDate 函数 396**
NumberToString 函数 396
- 内部样式**
 BaseInterior 类 274
 绘图区域 272
 SimpleInterior 272
- 内存**
 性能 72
- O**
- ODBC 连接字符串格式 199**
OGC 查询接口
 分析数据 91
- Oracle Spatial**
 几何体转换 196–198
 连接字符串格式 200
- Oracle 应用程序**
 故障诊断 213

order byMapInfo SQL 定义 **362****P****PCX 文件**支持的光栅格式 **295****Persistence 命名空间**概述 **51****Photoshop 文件**支持的光栅格式 **295****PNG 文件**支持的光栅格式 **295****PointRef 构架 **87, 162–163****定义 **351****Polyconic 投影 **541******ProportionAvg 函数 **397******ProportionSum 函数 **398******ProportionWtAvg 函数 **399******派生的类**定义 **353****匹配数据**GeoDictionary **169–170**自动匹配 **169–170****Q****QueryDefinition**定制 **449****QueryFilters**定制 **448****起始状态**应用程序 **64****迁移到 MapXtreme 2005 **11, 411, 413–414, 416–420******倾斜方位角**Hotine Oblique Mercator 投影 **541****R****Raster 命名空间**概述 **51**UML 图 **294****Right 函数 **400******Round 函数 **400******Routing 命名空间**概述 **52****Rtrim 函数 **400******日期格式化字符串**MapInfo SQL 语言 **406–407****日期函数**MapInfo SQL 语言 **366****S****Search 函数**定制 **448****SearchResultProcessor 类**定制 **449****Select 命令**MapInfo SQL 示例 **360****select 语句**结果集 **90**MICommand **171–172****Selection 类 **142–143******.sid 文件**支持的光栅格式 **295****Sign 函数 **401******Space 函数 **401******SpatialWare 应用程序**故障诊断 **213****Spot 文件**支持的光栅格式 **295****SQL 查询类型 **175******SQL 语句**MICommand **171–172****SQL 语言**MapInfo 参考 **91****Sqrt 函数 **402******StringToDate 函数 **402******StringToNumber 函数 **402******StyleRepository**样式存储 **275****StyleRepository 类**访问预定义的样式 **275****Styles 命名空间**概述 **51**UML 图 **270****Substring 函数 **403******Sum 函数 **403******升级**到 MapXtreme 2005 **25–27****事件**工具 **116–117**

事件句柄
 定义 **353**
 web 控件 **106**

示例数据版权信息 **557–560, 562–565**

示例应用程序
 MapXtreme 附带 **58**
 搜索 **188**
 ThematicsWeb **440**

视图
 表 **89**
 MapView **218**

视图工具
 更改地图视图 **112**

实用程序
 GeoDictionary 管理器 **342–343, 345–347**
 Geodictionary 管理器 **169–170**
 Workspace 管理器 **324**

十字路口
 用 Find 匹配 **252**

属性数据
 访问 **203**

术语表 **350–351, 353–355**

数据
 分析 **90**
 在 MapXtreme 2005 中的使用 **164**

数据绑定 **164**
 定义 **353**
 在 GeoDictionary 中注册表 **342**

数据访问 **79, 168**
 表达式 **178–182**
 部署应用程序 **38**
 DataTable **166**
 ExecuteFeatureCollection **173**
 Geodictionary 管理器 **169–170**
 规划 **79–86, 88–92**
 IDbCommand **167**
 MapInfo ADO.NET 数据提供方 **170**
 MapInfo SQL 语言 **174**
 MI 连接 **80**
 MICommand **171–172**
 MIConnection **80**
 MIDataReader **173**
 SQL 命令 **171–172**
 体系结构 **79**
 图元 **183**

源行 **169**
最优化性能 **92**

数据命名空间
 概述 **50**
 UML 图 **148**

数据源
 TableInfo **81**
 选取 **81**
 支持 **161**

数值函数
 MapInfo SQL 语言 **365–366**

数字格式化字符串
 MapInfo SQL 语言 **408–409**

双变量主题 **244–245**

搜索示例应用程序 **188**

搜索图元 **185**
 目录方法 **185**
 SearchInfo **187**
 SearchInfoFactory **187**

缩放图案 **475**

缩写文件
 编辑 **263–264, 268**
 替换方法 **265–267**
 替换关键字 **264**

T

TableInfo
 元数据 **156**
 支持的数据源 **81**

Tan 函数 **403**

Targa 文件
 支持的光栅格式 **295**

ThematicsWeb
 示例应用程序 **440**

TIFF 文件
 支持的光栅格式 **295**

Tools 命名空间
 概述 **52**

Transverse Mercator 投影
 比例因子 **541**

Trim 函数 **404**

TrueType 字体
 StyleRepository 类 **275**

Trunc 函数 **404**

- 特殊字符
- Mapinfow.abb 文件替换 **267**
- 特许
- 分布式应用程序 **437**
- 应用程序 **18, 437**
- 替换
- Mapinfow.abb 文件关键字 **264**
- 体系结构
- 对象模型实现 **412**
- 概述 **48**
- 数据 **79**
- web 应用程序 **53–54**
- 应用程序 **52–55**
- 桌面应用程序 **55**
- 填充图案 **484**
- 填充样式 **484**
- 绘图区域 **272**
- 在远程空间表中 **211**
- 添加
- 将对话框添加至应用程序 **129**
- 将光栅图像添加到地图 **296**
- 将几何体对象添加到地图 **113**
- 将网格图像添加到地图 **300**
- 添加工具
- 绘制几何体图元 **114**
- 将几何体对象添加到地图 **113**
- 条形图主题
- 创建 **237**
- 统一资源定位符 (URL)
- 定义 **353**
- 通用字符集 (UCS)
- 定义 **353**
- 投影
- 参数 **526**
- 出版物和资源 **542–543**
- 地图绘制术语说明 **44**
- 定义 **354**
- 基准面 **531–538**
- MapInfoCoordinateSystemSet.xml **529**
- MapInfoCoordinateSystemSet.xml ID 编号 **527**
- 凸多边形
- 定义 **354**
- 图案缩放 **475**
- 图层
- 地图绘制术语说明 **41**
- 定义 **354**
- 定制外观 **449**
- 类型 **219**
- UML 图 **219**
- 图层覆盖样式
- 组合样式 **272**
- 图层类型
- FeatureLayer **219**
- GroupLayer **220**
- MapLayer **220**
- ObjectThemeLayer **220**
- UserDrawLayer **220**
- 图层, 创建
- UserDrawLayer 类 **450**
- 图例 **245**
- 创建主题 **246**
- 地图绘制术语说明 **42**
- 对齐框架 **248**
- 修饰 **227**
- 制图 **247**
- 图元 **187**
- 从表检索 **184**
- 地图绘制术语说明 **41**
- 定义 **354**
- ExecuteFeatureCollection **173**
- FeatureCollection **184**
- FeatureGeometry **183**
- 描述 **183**
- Selection 类 **142–143**
- 搜索 **185**
- 修改 **184**
- 修饰符主题 **232**
- 椭圆
- 绘制椭圆 **114**
- U**
- UML 图
- Data.TableInfo **156**
- 地图化 **216**
- Geometry **281**
- Mapping.FeatureStyleModifier **229**
- Mapping.Labels **223**
- Mapping.Layers **219**
- Mapping.Thematics.IModifierTheme **232**
- Mapping.Thematics.ObjectTheme **233**

- Raster **294**
- 数据命名空间 **148**
- 样式 **270**
- UML 图表
 - Data.Find **255**
- Unicode 转化格式 (UTF-8)
 - 定义 **351**
 - 在工作空间中支持 **461**
- Update 命令
 - MapInfo SQL 示例 **360**
- Upper 函数 **404**
- UserDrawLayer 类 **450**
 - 定制 **450**
- USGS DEM 文件
 - 支持的网格格式 **299**
- V**
 - VectorPointStyleRepository 类
 - 定制 **454**
 - Vertical Mapper Classified Grid
 - 支持的光栅格式 **295**
 - Vertical Mapper Continuous Grid
 - 支持的光栅格式 **295, 300**
 - 支持的网格格式 **300**
- Visual Basic .NET
 - ASP.NET 应用程序模板 **30, 32**
 - 地图应用程序模板 **28**
 - 开发框架 **16**
 - MapForm 应用程序模板 **28–29**
- Visual C# **16**
- Visual Studio .NET
 - 创建应用程序 **27–30, 32**
 - 开发框架 **16**
- W**
 - Web 地图服务 (WMS)
 - 测试案例 **75**
 - 地图请求 **306**
 - 定义 **351**
 - 服务器操作 **305**
 - 服务器设置 **304**
 - 客户端性能 **77**
 - 示例配置文件 **518–522**
 - 使用客户端 **306**
 - 图像 **308**
 - Web 服务器
 - 定义 **351**
 - web 工具
 - 创建自定义 (版本 6.5 之前) **509**
 - DistanceTool (版本 6.5 之前) **509**
 - 服务器端示例 (版本 6.5 之前) **509**
 - InfoTool (版本 6.5 之前) **508**
 - 客户端示例 (6.5 版本之前) **513**
 - 客户端示例 (版本 6.5 之前) **513, 515**
 - 使用 MapControl **101**
 - 使用 MapControl (版本 6.5 之前) **508**
 - 体系结构 **96**
 - web 控件 **94–103, 105–108**
 - ASP.NET Web 应用程序 **32**
 - 本地化 **101**
 - 从 MapXtreme 2004 迁移 **106**
 - 错误管理 **102**
 - 更改源代码 **107–108**
 - LayerControl (版本 6.5 之前) **506**
 - LegendControl (版本 6.5 之前) **506**
 - 类型 (版本 6.5 之前) **502–503**
 - MapControl (版本 6.5 之前) **504**
 - 目录结构 **97**
 - 事件句柄 **106**
 - 使用 **97–103, 105–108**
 - 使用 (版本 6.5 之前) **504**
 - 体系结构 **94–95**
 - web 应用程序 (版本 6.5 之前) **502–503**
 - 在框架中 **100**
 - 状态管理 **103**
 - 自定义工具 **99**
 - 自定义命令 **99**
 - Web 图元服务 (WFS)
 - 定义 **351**
 - 服务器 **315**
 - 客户端 **316, 318**
 - web 应用程序
 - 部署 **436**
 - 创建 **429–430**
 - 打包 **433–434**
 - 打包教程 **433–434**
 - 更新到 MapXtreme 2005 **25–26**
 - 规划 **60–71**

- 教程 **429–430**
- 设计考虑 **60–71**
- 体系结构 **53–54**
- Web.config 文件**
 - 会话管理 **139**
 - WFS 的配置 **314**
 - 预装工作空间 **98**
- WebControls 命名空间**
 - 概述 **51, 94**
- Weekday 函数 **405****
- WFS**
 - 请参阅 Web 图元服务 (WFS)
- where 子句 **179****
- Windows 2000**
 - 支持的操作系统 **16**
- Windows 2003**
 - 支持的操作系统 **16**
- Windows 对话框**
 - CreateThemeWizard **130–131, 133–134**
 - 定制 **135**
 - 添加至应用程序 **129**
- Windows 控件**
 - LayerControl **128–129**
 - MapControl **124**
 - MapForm 应用程序 **32**
 - MapToolBar **126–127**
 - 桌面应用程序 **124**
- Windows 图元文件**
 - 支持的光栅格式 **295**
- Windows 位图文件**
 - 支持的光栅格式 **295**
- Windows XP**
 - 支持的操作系统 **16**
- Windows 增强图元文件**
 - 支持的光栅格式 **295**
- Windows.Controls 命名空间**
 - 概述 **51**
- Windows.Dialogs 命名空间**
 - 概述 **51**
- .wmf 文件**
 - 支持的光栅格式 **295**
- WMS**
 - 请参阅 Web 地图服务 (WMS)
- Workspace 管理器**
 - 标记 **338**
- 菜单命令 **325–327, 329, 331–332**
- 工具菜单命令 **330**
- LayerControl **332–333, 335–337**
- LayerControl 工具 **333–334**
- 另存为 XML **324**
- 命名连接 **326**
- 特性摘要 **324**
- 主题 **336**
- WorkSpaceLoader 类**
 - 定制 **454**
- WtAvg 函数 **405****
- 网格图**
 - 定义 **354**
- 网格图像 **299****
 - 变化 **273**
 - hillshade 参数 **273**
 - 检索数据 **301**
 - 添加到地图 **300**
 - 样式 **273**
 - 主要的类 **300**
- 纬度**
 - 定义 **354**
- 纬度的度**
 - 定义 **352**
- 位图**
 - 创建定制 **276**
- 位图符号 **499****
- 位置名称 **252****
 - 查找结果代码 **262**
- 文本**
 - 绘图 **273**
 - 绘制样式 **274**
- 文档集 **13****
- 文件格式**
 - 光栅图像 **295**
 - 网格图像 **299**
- 无对象池**
 - 测试案例 **74**
- X**
- XML 文件**
 - 导入图元 **454**
 - GeoDictionary **342, 344, 346**
 - Workspace 格式 (.MWS) **324**

XY 构架 162
 定义 351
XY 空间 86
XY 空间构架
 空间构架 86
系统要求 16
相关表
 目录 85
项目模板
 MapXtreme 附带 58
小数度
 定义 354
性能
 CPU 72
 测试摘要 72
 内存 72
 应用程序 72–78
 优化 74–75, 77–78
 最优化数据访问 92
修改图元 184
修饰
 地图元素 227
 定义 354
修饰符样式
 组合样式 272
修饰符主题 232
 单值标注主题 242
 单值主题 241
 点密度主题 243–244
 范围 238
 范围标注主题 240
许可证
 获得 19
 类型 18, 437
 SDK 18, 437
 试用 18, 437
 应用程序 19
 运行时 18, 437
选择对象
 缓冲区外 180
 在缓冲区中 180
选择工具
 选择点和区域 112

Y
Year 函数 406
样本连接字符串 200
样式
 变化 273
 标记 338
 查找表 483
 常用 274
 创建定制位图 276
 点 272, 274
 覆盖 277
 光栅 273
 LayerControl 276
 区域 272
 矢量点 274
 填充 272, 274
 Workspace 管理器 338
 网格 273
 网格 hillshading
 文本 274
 修改 229
 以库中的类定制 453
 在远程空间表中 210
 支持的定制位图符号 499
 支持的矢量符号 495
 支持的填充图案 484
 支持的直线样式 494
 直线 274
 重载 230
 字体 273
 字体点 273
 组合 272
异常类
 Engine 命名空间 146
引用标识符
 MapInfo SQL 语言 359
应用程序
 部署 436
 打包 web 433–434
 打包桌面 431–433
 规划 web 60–71
 规划桌面 58–60
 启动工作空间 70
 起始状态 64

- 特许 **18–19, 437**
- 体系结构 **52–55**
- web 应用程序体系结构 **53–54**
- 性能 **72–78**
- 优化 **74–75, 77–78**
- 桌面体系结构 **55**
- 应用程序池设置
 - 标识 **75**
 - 回收 **75**
 - IIS **75**
 - 健康状况 **75**
 - 性能 **75**
- 应用程序模板
 - ASP.NET **30, 32**
 - MapForm **28–29**
- 应用程序设置
 - 状态管理 **441**
- 用户创建的安装程序
 - 部署应用程序 **35, 37**
- 永久性
 - 定义 **354**
- 优化边界表
 - 查找街道地址 **254**
 - 查找结果代码 **262**
- 预定义的样式 **275**
- 预构建对话框
 - MapXtreme 附带 **59**
- 远程表
 - 不使用 .tab 文件 **195**
 - DBMS 服务器 **194**
 - 定义可制图的表 **201**
 - 访问属性数据 **203**
 - 缓存管理 **204–205**
 - 可制图的表 **201–202**
 - ODBC 连接字符串格式 **199**
 - 使用 .tab 文件 **194**
 - 性能 **203**
 - 映射到 X/Y 列 **195**
- 远程空间表
 - 导入 DBMS 数据库 **206**
 - 将行添加到 Map Catalog **208–210**
 - MapInfo Map Catalog **206**
 - 每记录样式 **212**
 - 样式 **211**
 - 指定样式 **210**
- 圆圈工具
 - 绘制圆圈 **115**
- 元数据
 - TAB 文件 **157**
 - TableInfo **156**
- 源行 **169**
- 原型开发
 - 快速桌面应用开发 **59**
- 圆锥投影
 - 标准纬线 **540**
- 运算符
 - MapInfo SQL 语言 **362, 364**
- 运算符优先级
 - MapInfo SQL 语言 **364**
- 运行时安装程序
 - 部署应用程序 **34**
- Z
- 杂项函数
 - MapInfo SQL 函数 **369**
- 折线
 - 绘制样式 **274**
- 折线工具 **114**
- 正弦函数 **401**
- 支持的文件格式
 - 光栅图像 **295**
 - 网格图像 **299**
- 制图图例 **247**
- 直线工具 **114**
- 直线样式 **494**
 - BaseLineStyle 类 **274**
 - 绘制折线 **274**
 - 在远程空间表中 **211**
- 质心
 - 定义 **355**
- 主题
 - 饼图 **235**
 - CreateThemeWizard **130–131, 133–134**
 - 单值标注主题 **242**
 - 单值主题 **241**
 - 地图绘制术语说明 **43**
 - 点密度主题 **243–244**
 - 对象 **233**
 - 范围 **238**
 - 范围标注主题 **240**

- 分级符号 **234**
- 双变量 **244–245**
- 条形图 **237**
- 图例 **245**
- Workspace 管理器 **329, 336**
- 修饰符 **232**
- 主题图例
 - 创建 **246**
- 转换函数
 - MapInfo SQL 语言 **367**
- 状态管理
 - 案例研究 **440**
 - 保存状态 **444**
 - 初始 **442**
 - 恢复状态 **445**
 - 会话对象 **139**
 - 手动 **441**
 - web 控件 **103**
 - 正常 **444**
- 桌面应用程序
 - 部署 **436**
 - 创建 **422, 425–428**
 - 打包 **431–433**
 - 打包教程 **431–433**
 - 更新到 MapXtreme 2005 **25**
 - 规划 **58–60**
 - 会话管理 **139**
 - 教程 **422, 425–428**
 - MapControl **59**
 - MapTools **60**
 - 体系结构 **55**
 - 原型开发 **59**
- 自定义命令
 - 创建 **99**
- 自动标注
 - Workspace 管理器 **337**
- 字符编码
 - 定义 **355**
 - 工作空间中的 UTF-8 **461**
- 字符串函数
 - MapInfo SQL 语言 **366**
- 字体点样式
 - 用 MapInfo.Styles.Font 类绘制点 **273**
- 子午线
 - 定义 **355**
- 字形
 - 绘图文本 **273**
- 组合样式 **272**
- 最佳实践
 - 状态管理 **440**
- 坐标
 - 定义 **355**
 - 在 NAD 27 和 NAD 83 之间转换 **542**
 - 转换为另一个基准面 **542**
- 坐标单位 **539**
- 坐标系
 - 出版物和资源 **542**
 - 地图绘制术语说明 **44**
 - 定义 **355**
 - 光栅图像 **296**
 - 几何体 **290**
 - 原点 **540–541**
 - 注册到 CoordSysFactory 对象 **290**