

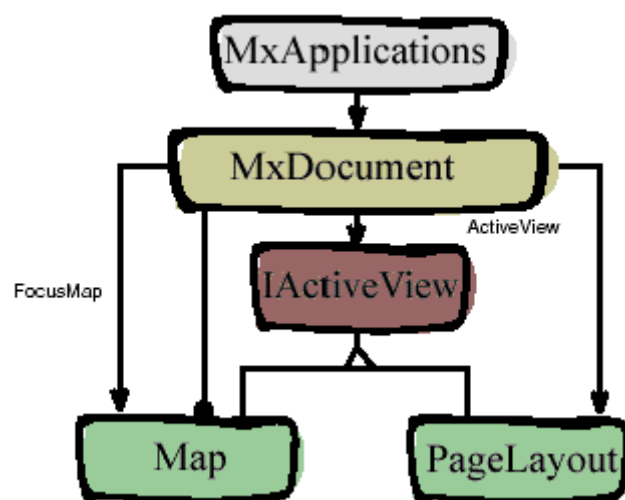
ArcGIS World第一期

ArcObjects 初探	2
ArcMap中的表格操作	8
ArcGIS和ArcView 3.x中的表格.....	9
地图配置文件，地图服务，请求和响应之间的关系	11
基于Oracle的ArcSDE配置	17
将注记从coverage格式转化为geodatabase格式.....	19
设置热键加速ArcMap操作	21
使用ArcGIS 8.2 加载ArcSDE栅格数据.....	23
压缩Geodatabase	28
ArcGIS中的影像配准	30
使用sderaster命令向ArcSDE中加载栅格图像	36
ArcIMS 4 新功能和扩展模块.....	41
从ArcView 3.x工程导入符号化方案	43
在ArcMap中用地图提示浏览数据	44

ArcObjects 初探

ArcObjects 是一个 COM 组件的集合，它是 ArcInfo 8 的两个新应用程序的基石，即 ArcMap 和 ArcCatalog。这一系列的组件包含了超过 1200 个对象，它们可以用来定制、扩展和构建 GIS 应用程序。乍一看来，这么大的一个对象模型看起来真有点让人喘不过气来。仔细看一下 ArcMap 的对象模型你会发现它和其它一些大的对象模型的基本组成很相似。

1、基础构架



图片 1

图 1 显示了 ArcMap 的概念对象模型。顶端的对象 MxApplication 代表 ArcMap 本身。MxApplication 只管理一个文档 MxDocument。与 ArcView GIS 不同的是 ArcMap 使用单文档界面。MxDocument 管理一组 Map 对象和一个 PageLayout 对象。在数据视图下，ActiveView 是一个 Map；而在页面视图下，ActiveView 是 PageLayout。无论在何种视图下，总是只有一个 FocusMap。

例 1

```
Private Sub Zoom()
    'get the active view
    Dim mxDoc As IMxDocument
    Set mxDoc = Application.Document
    Dim activeView As IActiveView
    Set activeView = mxDoc.activeView
    ' get the active view's extent
    Dim ext As IEnvelope
    Set ext = activeView.Extent
    ' shrink the extent
    ext.Expand 0.75, 0.75, True
    ' set the extent
```

```

activeView.Extent = ext
activeView.Refresh
End Sub

```

如果 ArcMap 在页面视图下显示一个有三个数据框 (data frame) 的文档, 将会有三个 Map 与 MxDocument 相关, ActiveView 属性将返回 PageLayout, FocusMap 将返回当前被选中的数据框。例 1 中的 VBA 代码说明了这一部分对象模型。这段代码会改变 ActiveView 的显示范围, 放大 75%。这段代码非常的浅显。真正的工作是要从 ActiveView 中获的当前的显示范围, 它是一个 IEnvelope 接口对象。然后缩放, 并把调整的封装边界作为新的显示范围。这段代码在 ActiveView 上通过 IActiveView 接口进行操作。ActiveView 能在 PageLayout 或 Map 中使用, 这完全取决于最终用户的选择。这段代码对于两种情况都有效。

2、使用接口

有时候对于 Map 或 PageLayout, 你可能要使用不同的接口。在这种情况下, COM 对象模型与传统的面向对象模型, 像 Avenue 或是 MapObjects, 有很大的不同。接口是一组相关方法和属性的集合。COM 对象暴露多个接口。比如说, Map 对象不但有 IActiveView 接口, 而且有 IMap 接口和其它一些接口。

例 2

```

Private Sub ClearLayers()
    ' get the active view
    Dim mxDoc As IMxDocument
    Set mxDoc = Application.Document
    Dim activeView As IActiveView
    Set activeView = mxDoc.activeView
    If TypeOf activeView Is IMap Then
        Dim map As IMap
        Set map = activeView ' acquire the IMap interface
        map.ClearLayers ' remove the layers
        mxDoc.UpdateContents ' refresh the table of contents
        activeView.Refresh ' refresh the display
    End If
End Sub

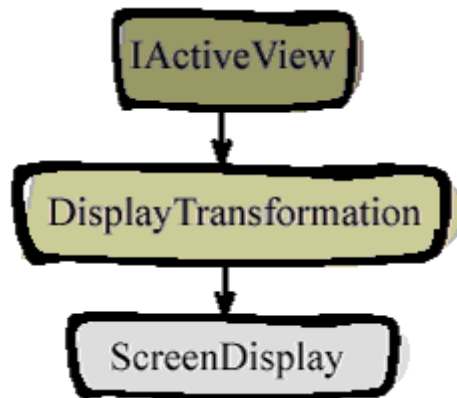
```

例 2 显示了用来从 activeView 变量获取 IMap 接口的 Set 语句。注意这段代码使用 TypeOf 语句来判断 ActiveView 是不是 Map 的情况, 通常情况下 IMap 接口是不显示的。如果不使用 TypeOf 语句, 那么在运行代码时当 ActiveView 是一个 PageLayout, 在遇到 Set 语句的时候就会得到一个类型不匹配的错误。

3、屏幕显示

PageLayout 和 Map 都有一个相关的对所有图形进行描述的 ScreenDisplay 对象。ScreenDisplay 包含一个 DisplayTransformation。你能获取 DisplayTransformation 的属性比如 VisibleBounds, 并且可以使用相应的方法在地图和显示单位间转换坐标和距离。图 2 说明了这一点。

比如向地图中增加一个新图层的事件将使 ScreenDisplay 触发一个重绘事件。这一绘制过程发生在三个方面。绘制顺序是地理数据、选择集和注记。绘制缓存 (drawing cache) 里存放了地图的一个非屏幕位图, 只要窗口刷新, 它就会被显示在屏幕上。ScreenDisplay 管理几个缓存。如果使用 esriNoScreenCache 常量, 图形将直接绘制在 ScreenDisplay 的窗口上, 不进行缓存。如果其它窗口获得显示, 图形将消失。下面的代码使用 esriNoScreenCache 在 ScreenDisplay 的中间绘制文本 “Hello”。



图片 2

注意例 3 中的例子，在 StarDrawing 和 FinishDrawing 中间调用了 SetSymbol 和 DrawText。

例 3

```
Private Sub Hello()  
Dim mxDoc As IMxDocument  
Set mxDoc = Application.Document  
Dim activeView As IActiveView  
Set activeView = mxDoc.activeView  
    ' set up a text symbol to draw with  
Dim sym As ITextSymbol  
Set sym = New TextSymbol  
    ' change the font size to 18  
Dim fnt As IFontDisp  
Set fnt = sym.Font  
fnt.Size = 18  
sym.Font = fnt  
    ' draw into the active view's display  
With activeView.ScreenDisplay  
    .StartDrawing .hDC, esriNoScreenCache  
    .SetSymbol sym  
Dim bnds As IArea  
Set bnds =  
    .DisplayTransformation.VisibleBounds  
    .DrawText bnds.Centroid, "Hello"  
    .FinishDrawing  
End With  
End Sub
```

4、使用图层

地图维护了一组图层。图层是实现了 ILayer 接口的对象。图层的实例包括了 FeatureLayer、GroupLayer、GraphicsLayer、AnnotationLayer、CadLayer、TinLayer 和 RasterLayer。IMap 接口的 Layer(index) 和 LayerCount 属性能够用来浏览图层集合。例 4 所示的功能是找到并返回指定名字的图层。

例 4

```
Function FindLayer(map As IMap, name As String) As ILayer
Dim i As Integer
For i = 0 To map.LayerCount - 1
If map.Layer(i).name = name Then
Set FindLayer = map.Layer(i)
Exit Function
End If
Next
End Function
```

FeatureLayer 通过 FeatureClass 对象来管理一个矢量数据的连接，比如 coverages、shapefile 或 SDE 图层。例 5 中的代码从一个 shapefile 中获取一个 FeatureClass，通过打开一个 Workspace，这个 Workspace 用 ShapefileWorkspaceFactory 包含了 shapefile。然后创建一个 FeatureLayer 与图层的 FeatureClass 相关，在把它加到 FocusMap 中。

例 5

```
Sub AddLayer()
    ' use a workspaceFactory to open the workspace
Dim wksFact As IWorkspaceFactory
Set wksFact = New ShapefileWorkspaceFactory
Dim wks As IFeatureWorkspace
Set wks = wksFact.OpenFromFile("c:\Data\shp", 0)
    ' open the featureClass
Dim fc As IFeatureClass
Set fc = wks.OpenFeatureClass("BigCypress")
    ' create a featureLayer
Dim lyr As IFeatureLayer
Set lyr = New FeatureLayer
Set lyr.FeatureClass = fc
    ' name the layer with the featureClass name
Dim ds As IDataset
Set ds = fc
lyr.Name = ds.Name
    ' add the layer to the map
Dim mxDoc As IMxDocument
Set mxDoc = Application.Document
Dim map As IMap
Set map = mxDoc.FocusMap
map.AddLayer lyr
End Sub
```

FeatureLayers 管理的一个要素的选择集，并且暴露 IFeatureSelection 接口来操作这个选择集。例 6 的代码通过使用一个 SQL 的 WHERECLAUSE 查询语句得到一个选择集。

例 6

```
Sub SelectFeatures()
```

```

Dim mxDoc As IMxDocument
Set mxDoc = Application.Document
' find the layer we're going to select from
Dim lyr As IFeatureLayer
Set lyr = FindLayer(mxDoc.FocusMap, "BigCypress")
' get the layer's selection interface
Dim sel As IFeatureSelection
Set sel = lyr
' create a query filter
Dim filter As IQueryFilter
Set filter = New QueryFilter
' set up the where clause and the spatial reference
filter.WhereClause = "NAME = 'Hampton' "
Dim shapeField As String
shapeField = lyr.FeatureClass.ShapeFieldName
Set filter.OutputSpatialReference(shapeField) = _
mxDoc.FocusMap.SpatialReference
' invalidate the current selection on the display
mxDoc.ActiveView.PartialRefresh _
esriViewGeoSelection, Nothing, Nothing
' select features and invalidate the new selection
sel.SelectFeatures filter, _
esriSelectionResultNew, False
mxDoc.ActiveView.PartialRefresh _
esriViewGeoSelection, Nothing, Nothing
' notify everyone that we've modified the selection
Dim selEvents As ISelectionEvents
Set selEvents = mxDoc.FocusMap
selEvents.SelectionChanged
End Sub

```

这段代码中的 IFeatureSelection 接口上的 SelectFeatures 方法完成了所有的工作。QueryFilter 对象为选择定义了标准。WhereClause 属性为 SQL 表达式而设置，OutputSpatialReference 属性用来控制要素如何进行动态投影，当它们从数据库中取出的时候。为了指明一个空间限制条件，你可以使用 SpatialFilter 对象来替换 QueryFilter。

调用 IActiveView 的 PartialRefresh 方法有目的地重绘显示区，旧的选择区被擦除并且新的选择区被绘制。最后三行通过 ISelectionEvents 接口调用 SelectionChanged 方法。这一方法通知那些系统中的通过地图注册的组件，选择已经改变了。属性表窗口（Attribute Table Window）侦听选择事件，以致它能同步的高亮显示与选择相匹配的行。如果图 8 所示的代码没有调用 SelectionChanged 方法，表窗口（Table Window）将不会反映激活视图中的选择。这种在 ArcObjects 中普遍使用的交流方式被称为处理事件过程（working with events）。

5、响应事件

直到现在我们只考虑了用代码操作 ArcObjects，从最顶端的对象开始，浏览了用到的对象。ArcObjects 一个有趣的方面是能够将对象模型和关联代码进行整合，这些代码只要有一些特别的事

情发生就会执行，比如说地图选择集的改变。很多对象模型都能触发事件。COM 提供了一个机制监听和响应事件。

如果你是用 VBA 或 Visual Basic，大部分幕后的线程都需要响应事件。你只需要为每个你感兴趣的事件编写代码，这个代码就开始侦听事件。例 7 所示的例子表明了 VBA 环境下如何处理事件。下面的代码用 WithEvents 语句声明了一个全局变量。WithEvents 告诉开发环境对象变量将用来响应对象事件。

```
Dim WithEvents g_Map As Map
```

变量声明后，就可以从 VBA 的对象下拉列表中选择。单击过程下拉列表框将显示 Map 对象所暴露的变量事件。这个例子使用 SelectionChanged 方法来编写代码。例 7 所示的代码响应 SelectionChanged 事件，只要选择集发生了改变，地图就能触发这个事件，并且用户窗体的列表框会显示选中要素的名字。

例 7

```
Private Sub g_Map_SelectionChanged()  
    ' show the form if it's not visible  
    If Not UserForm1.Visible Then  
        UserForm1.Show vbModeless  
    End If  
    ' clear any previous results in the list  
    UserForm1.ListBox1.Clear  
    ' get the map's selection  
    Dim activeView As IActiveView  
    Set activeView = g_Map  
    ' enumerate over the selected features  
    Dim featureEnum As IEnumFeature  
    Set featureEnum = activeView.Selection  
    featureEnum.Reset ' reset the cursor  
    Dim feat As IFeature  
    Set feat = featureEnum.Next  
    Do While Not feat Is Nothing  
        ' add the feature's name to the list  
        Dim index As Long  
        index = feat.Fields.FindField("Name")  
        If index <> -1 Then  
            UserForm1.ListBox1.AddItem feat.Value(index)  
        End If  
        Set feat = featureEnum.Next ' get next feature  
    Loop  
End Sub
```

在显示用户窗体，清除列表框以后，代码从地图的 IActiveView 接口获得一个选择集。选择集的 IEnumFeature 接口用来浏览选择要素的集合，返回一个 IFeature 接口。代码搜索每一个要素字段的 Name 字段。如果找到 Name 字段，它的值就加到列表框中。

最后一步是初始化全局变量 g_Map，如例 8 所示。一旦这段代码被运行，只要选择集发生变化例 7 中的代码就执行，并且列表框将被选择集所更新。

例 8

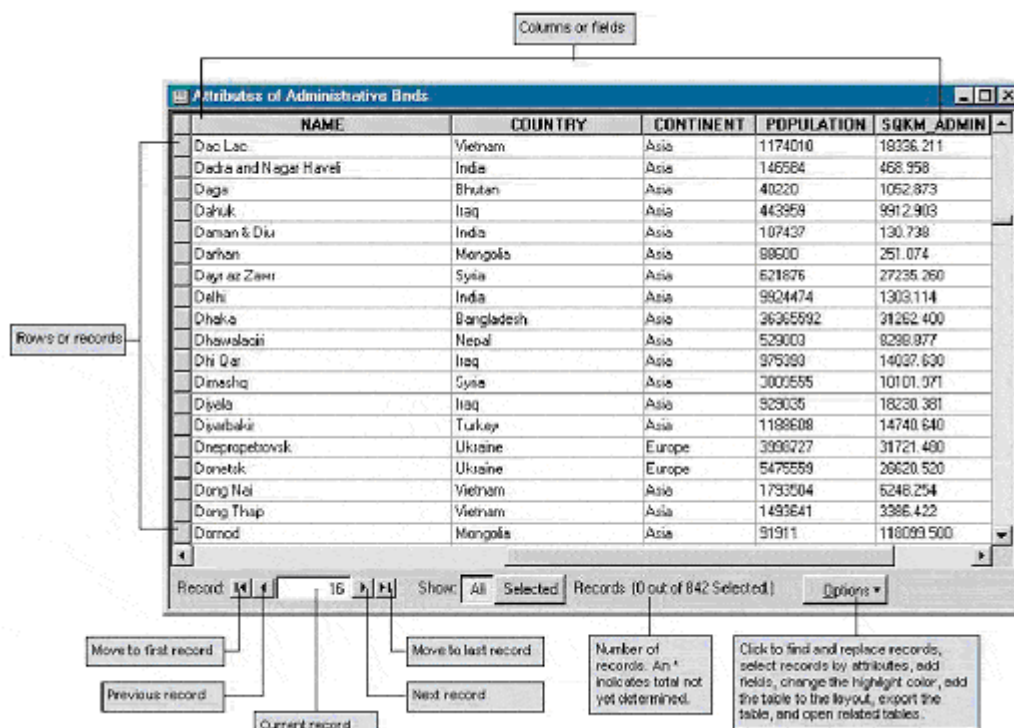
```
Sub StartListening()  
Dim mxDoc As IMxDocument  
Set mxDoc = Application.Document  
Set g_Map = mxDoc.FocusMap  
End Sub
```

注：ArcObjects 的更新很快，目前的版本已到 8.2，不久就会有 8.3，甚至是 9。随着每次版本的更新都会有新的对象和接口发布，所以对于开发者来说，要密切注意版本的变化。

（张俊、庄怀耀 编译）

ArcMap 中的表格操作

表格是由行和列组成的数据库组件。在 ArcGIS 的桌面应用程序 ArcMap 中，每一行或每个记录包含了描述地图上显示的一个地理要素的信息。例如，在一个与地图相联系的表中，地图显示的是消防栓的位置，表中的每行描述了某一个消防栓的信息，每列包含描述消防栓的一类属性像识别号码、制造商、安装日期、最后检视日期等。表格存储在像 Microsoft Access、dBASE、Oracle 或 Microsoft SQL Server 等数据库中。



图片 1

表格窗口的底部显示了表格的相关信息，Options 菜单提供了执行常见表格任务的选项

ArcGIS 和 ArcView 3.x 中的表格

ArcView 3.x 中要素的属性数据存储存储在表格中。表格包含存储在 dBASE 文件中的非空间数据，能够加载到 ArcView 3.x 和 ArcMap 中。ArcMap 的表格能显示和 ArcView 3.x 同样的数据格式。表格也可以在 ArcCatalog 中用 Preview 来查看。

常见的表格任务

在 ArcMap 中，能通过属性值查询表格来选择要素，或者为要素记录更新属性字段。表格也可以为显示在地图上的要素提供附加属性来加强地图的版面布局。

在 ArcMap 中打开图层的属性表，只需在目录表中右键点击图层，从环境菜单中选择 Open Attribute Table。表格窗口底部的一栏显示了被选中记录的编号、表格中的记录数目以及被选中记录的数目。使用栏中的箭头可以滚动查看所有记录。使用 All 和 Selected 按钮能在查看所有记录和查看被选中记录之间切换。点击位于表格窗口右下方的 Options 按钮，会发现像 Find、Replace 和 Add a Field 这样的常见的表格任务的菜单选项。

通过属性选择要素

尽管可以在查看表格时点击所需的记录来选择要素，但通常这不是最精确和有效的定位记录的方法。属性查询，可以从 ArcMap 窗口的菜单中或表格窗口上 Options 菜单中找到，能够识别所有满足指定的查询规则的记录。

1. 打开表格窗口，点击 Options 按钮，从菜单中选择 Select By Attribute。
2. 在 Select By Attribute 对话框中，点击 Method 旁边的下拉列表框，选择一种查询方案（例如，Create a new selection、Add to current selection）。
3. 在属性字段列表中，双击用于查询的字段。
4. 点击选择一个逻辑运算符（如 =、>、<）。
5. 从 Unique Values 列表中为选中的属性字段选择一个值，或者直接在文本框里输入一个值。
6. 点击 Verify，验证 SQL 语句格式正确而且能正确执行。
7. 点击 Apply。满足查询规则的记录会在表格中高亮显示。

使用 Select By Attribute 对话框中的 Save 和 Load 按钮，可以保存查询语句，并能通过读取它来快速重建这个选择集。SQL 查询也可以直接在文本框中输入。点击 SQL Info 按钮，ArcMap 会提供 SQL 的关键字、保留字符及其它针对所用类型数据库的查询限制。

更新要素属性

要素属性有时存储的是变化的数据——最后检视日期、人口或税额等。存储在 ArcMap 表格中的属性数据能直接通过编辑属性表来更新。

1. 在 ArcMap 中打开地图，通过 View>Tollbar>Editor 显示 Editor 工具条，开始一个编辑会话。在 Editor 工具条上点击 Editor 菜单，选择 Start Editing。
2. 打开要编辑的表格。
3. 滚动或查询表格，定位到包含需要编辑的属性值的记录。
4. 在记录中选择要编辑的字段。输入新的值，回车。

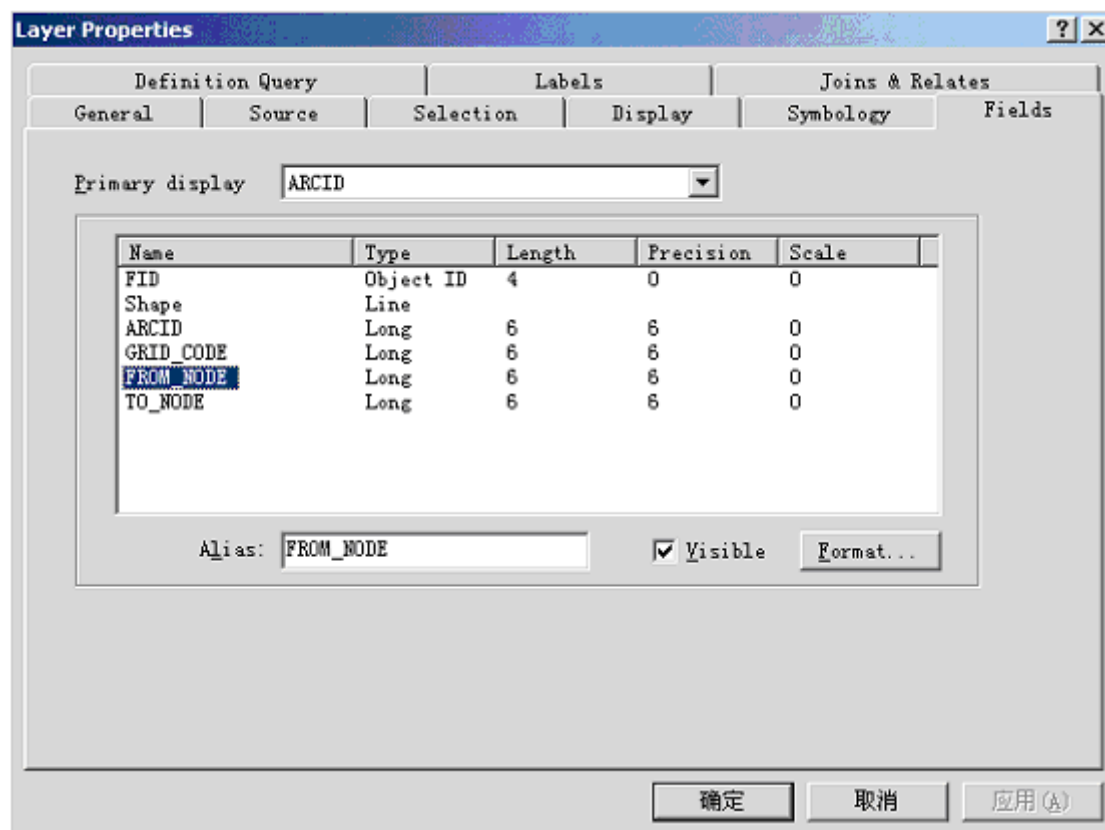
5. 用同样的方式编辑其它的属性字段。点击标准工具条上的 Undo 能撤消一次编辑。
6. 编辑结束时，点击 Editor 菜单选择 Stop Editing。

尽管可以不启动一个编辑会话就能对表格进行修改，但这样的改动不能像在编辑会话中那样被撤消。

向版面添加表格

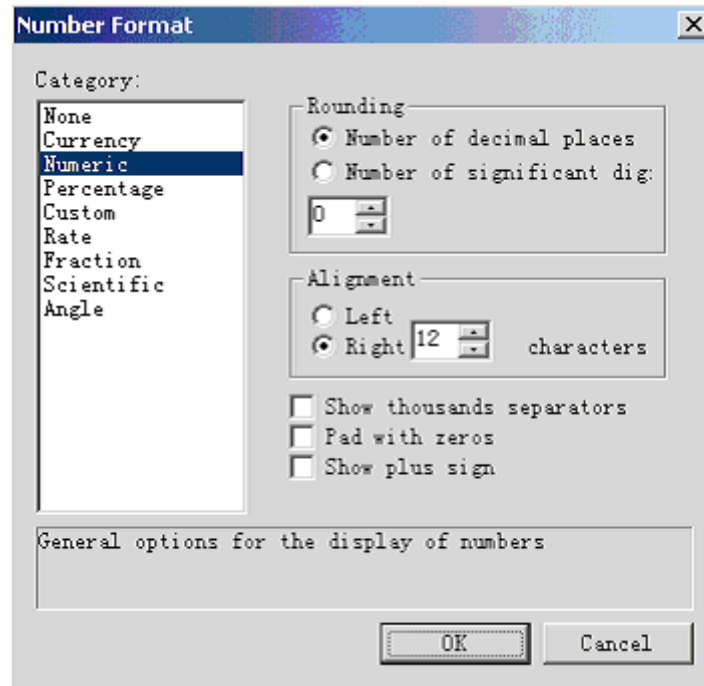
将表格添加到版面能提供地图上显示的要素的信息。在添加之前，可以修改表格的外观，通过改变文本的字体和大小、只显示被选中记录或者隐藏列。一旦将表格添加到版面，它的特征就不能被修改了。如果要做进一步的修改，得从版面中删除表格，在表格窗口进行修改，再重新把表格添加到版面。

1. 打开要添加到版面的表格。
2. 在表格窗口的右下角，点击 Options 按钮，选择 Appearance。
3. 在 Table Appearance 对话框中，设置表格中文本的字体、颜色和大小。
4. 如果表格中的数字需要格式（例如，添加千位分割符、正号），在 ArcMap 目录表中右键点击图层，选择 Properties。
5. 在 Layer Properties 对话框，点击 Fields 标签。



图片 2

6. 选择要格式化的数字型字段。点击 Format 按钮。
7. 在 Number Format 对话框中，选择一个类型 (Category)，设置小数位 (Rounding)、对齐方式 (Alignment) 和其它一些格式特征。点击 OK。



图片 3

8. 如果在版面中的表格只需显示被选中的要素,可以像前面描述的那样选择和显示记录。
9. 隐藏不需要的列,把指针移动到要隐藏的列和它右边的列之间。指针变成了双竖线。按住鼠标左键向左拖动竖线来隐藏该列。要解除隐藏可在两列之间双击左键。
10. 对表格窗口做最后的调整。点击 Options 按钮,选择 Add Table to Layout。
11. 点击版面上表格出现的区域,拖动它的边框。按需要调整表格边框的位置。
12. 将表格插入到版面之后,回到表格窗口,选择 Options>Appearance,将表格文本的特征修改成原始的设置。

其它对表格的常见操作在这里就不讨论了,包括连接表格、添加新的字段和生成报告。包含在表格中的数据能与地图上的要素基于共同字段联系起来。要素的新信息会被添加在属性表中新建的字段中。使用 ArcMap 软件内置的报告制造器,能把属性表中的信息组织、格式化及打印到报告中。更多关于 ArcMap 中表格操作的内容,可以查看 ArcMap 在线帮助或参考 GIS Store(www.esri.com/gisstore)上的《Getting to Know ArcGIS》。

地图配置文件,地图服务,请求和响应之间的关系

本文不仅讨论了地图配置文件,地图服务,请求和响应之间的关系,而且介绍了它们与 ArcIMS 空间服务器之间是如何相互作用的。ArcIMS 空间服务器是 ArcIMS 的支柱。空间服务器能将地图和数据打包成适当的格式,然后在将它发送给客户端浏览器。

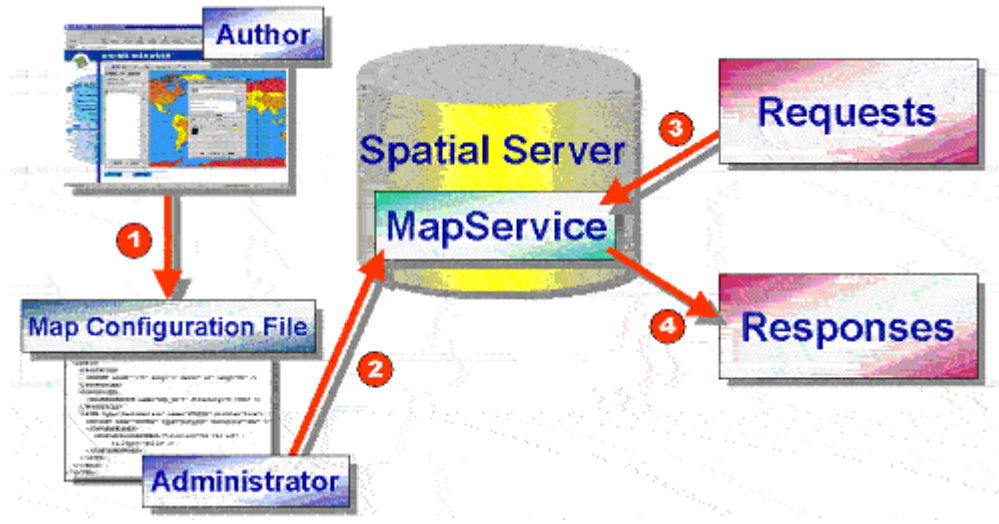
与空间服务器之间的所有的交流都是使用 ArcXML,图 1 显示了空间服务器和地图配置文件,地图服务,请求和响应之间的相互关系。图表中的数字反映了与空间服务器交流的不同步骤。

第 1 步，使用 Author 工具，文本编辑器或 XML 编辑器来创建一个地图配置文件。该地图配置文件是用 ArcXML 写的。

第 2 步，使用 Administrator 在 Spatial Server 上发布一个地图服务，第 1 步产生的地图配置文件是针对这个地图服务的。

第 3 步，空间服务器收到一个 ArcXML 格式的请求。

第 4 步，空间服务器生成一个 ArcXML 格式的响应。



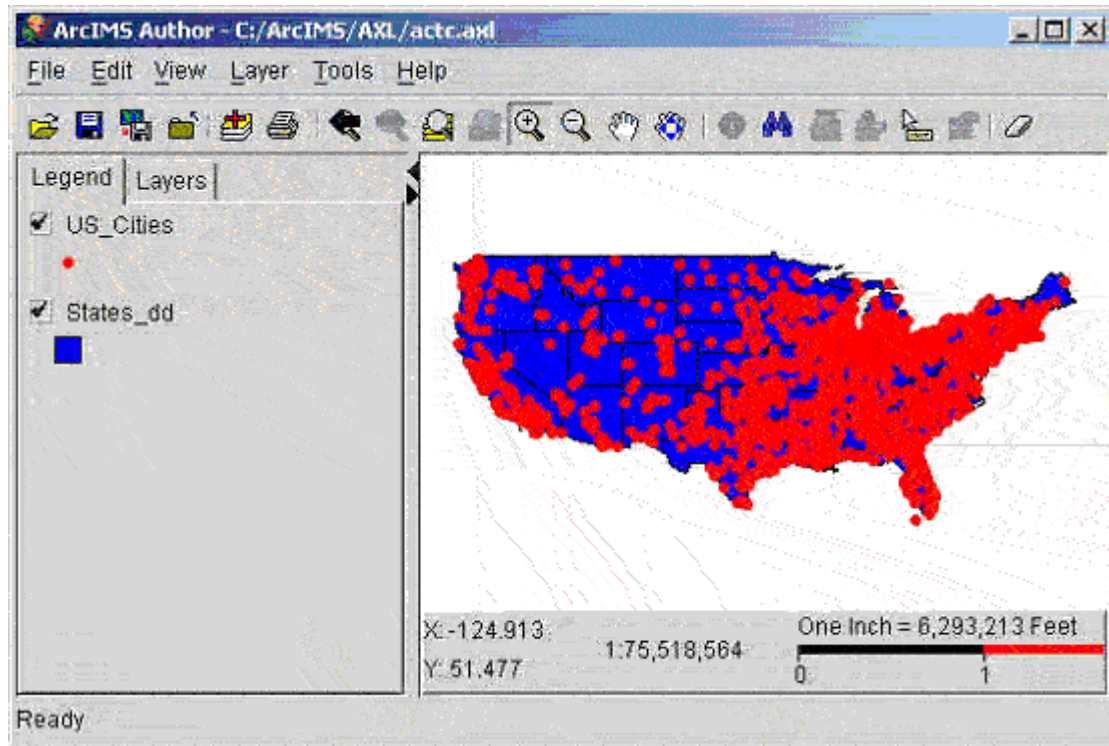
图片 1

图 1，按照图表显示地图配置文件，地图服务，ArcIMS 空间服务器，请求和响应之间的关系

上面的每一个步骤都会在下面详细讨论到。

第 1 步 创建一个地图配置文件

在图 1 的第 1 步，生成一个地图配置文件，你可以用 Author 工具，文本编辑器或 XML 编辑器创建一个地图配置文件，该文件是以 ArcXML 格式的，它包含地图的描述信息，如使用什么数据层，每一层是如何显示的。图 2 显示了 ArcIMS Author 中具有两个层，US_Cities 层和 States_dd 层。



图片 2

当你在 Author 中保存该文件，其结果就是包含层信息的一个地图配置文件，如图 3 所示。

```
<?xml version="1.0" encoding="UTF-8"?>
<ARCXML version="1.1">
<CONFIG>
<ENVIRONMENT>
<LOCALE country="CN" language="zh" variant="" />
<UIFONT color="0,0,0" name="sansserif.plain" size="12"
style="regular" />
<SCREEN dpi="96" />
</ENVIRONMENT>
<MAP>
<PROPERTIES>
<ENVELOPE minx="-128.86515248134154" miny="18.924781799316406"
maxx="-66.96927103600244" maxy="58.38488352301343"
name="Initial_Extent" />
<MAPUNITS units="decimal_degrees" />
</PROPERTIES>
<WORKSPACES>
<SHAPEWORKSPACE name="shp_ws-0" directory="C:\student\arcims\Data"
/>
</WORKSPACES>
<LAYER type="featureclass" name="States_dd" visible="true" id="0">
<DATASET name="States_dd" type="polygon" workspace="shp_ws-0" />
```

```

<SIMPLERENDERER>
<SIMPLEPOLYGONSYMBOL                                boundarytransparency="1.0"
fillltransparency="1.0" fillcolor="0,0,255" boundarycaptype="round"
/>
</SIMPLERENDERER>
</LAYER>
<LAYER type="featureclass" name="US_Cities" visible="true" id="1">
<DATASET name="US_Cities" type="point" workspace="shp_ws-0" />
<SIMPLERENDERER>
<SIMPLEMARKERSYMBOL color="255,0,0" width="6" />
</SIMPLERENDERER>
</LAYER>
</MAP>
</CONFIG>
</ARXML>

```

地图配置文件区别请求或响应的标志就是 CONFIG 元素。CONFIG 元素的子元素是用于辅助定义地图的特征，如 PROPERTIES，WORKSPACE，和 LAYER 等子元素。

步骤 2：发布一个地图服务

地图服务是运行在 ArcIMS 空间服务器上的一个应用程序。你可以将地图服务作为空间服务器的一个入口来看待。空间服务器只对运行在其上的地图服务提供相应的服务。

对于一个地图服务而言，一个地图配置文件是必要的（图 1 第 2 步）。当发布一个地图服务的时候，你必须指派地图服务给 FeatureServer 或 ImageServer, 在本文中的例子是发布成影像地图服务。无论哪种情况，地图配置文件都给地图服务中的每一层提供了绘制的规则。这些规则是地图服务的默认选项。比如，如果地图服务采用图 3 所示的地图配置文件，那么地图服务将给 States_dd 层以蓝色的多边形填充显示，US_Cities 层以红色的标记符号显示。

第 3 步：客户端发送请求

地图服务一旦运行在空间服务器上，客户端就能够向地图服务发送请求（图 1 中的第 3 步）。请求由 ArcIMS HTML 客户端浏览器、Java 客户端浏览器和 ColdFusion 及 ActiveX 连接器产生。

客户端请求与其它类型的 ArcXML 文件区别的标志是 REQUEST 元素。图 4 是客户端请求的典型的一个例子。

图 4：客户端请求影像的例子

```

<?xml version="1.0" encoding="UTF-8"?>
<ARXML version="1.1">
<REQUEST>
<GET_IMAGE>
<PROPERTIES>
<ENVELOPE minx="-125" miny="25" maxx="-67" maxy="50" />
<IMAGESIZE width="700" height="500" />
</PROPERTIES>
</GET_IMAGE>
</REQUEST>

```

</ARCXML>

当上述的请求发送到图 3 的地图服务，其结果看起来与 Author 中初始定义的地图一样。
使用的地图服务定义范围和图层符号。

图 5：使用地图服务的默认值生成的地图服务

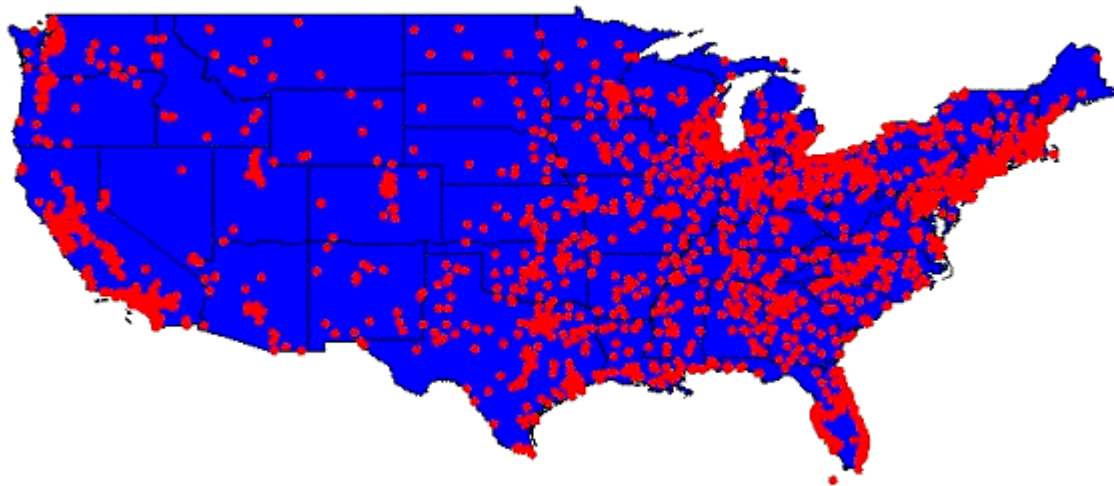


图 3

客户端的请求也可以通过请求不同比例尺，改变图层的颜色，显示或隐藏图层，请求属性数据的子集或改变投影，来覆盖掉地图服务的一些信息。图 6，客户端给同一个地图服务发出请求，获取新的缩放比例并将 US_Cities 层的要素符号改成星形，颜色从蓝色改变为绿色。

图 6：客户端请求改变地图服务缩放区域和图层要素的符号及颜色

```
<?xml version="1.0" encoding="UTF-8"?>
<ARCXML version="1.1">
  <REQUEST>
    <GET_IMAGE>
      <PROPERTIES>
        <ENVELOPE minx="-88" miny="30" maxx="-67" maxy="50.0" />
        <IMAGE_SIZE width="500" height="350" />
        <LAYERLIST>
          <LAYERDEF id="1" visible="true" >
            <SIMPLERENDERER>
              <SIMPLEMARKERSYMBOL type="star" color="0,255,0" width="6"/>
            </SIMPLERENDERER>
          </LAYERDEF>
        </LAYERLIST>
      </PROPERTIES>
    </GET_IMAGE>
  </REQUEST>
</ARCXML>
```



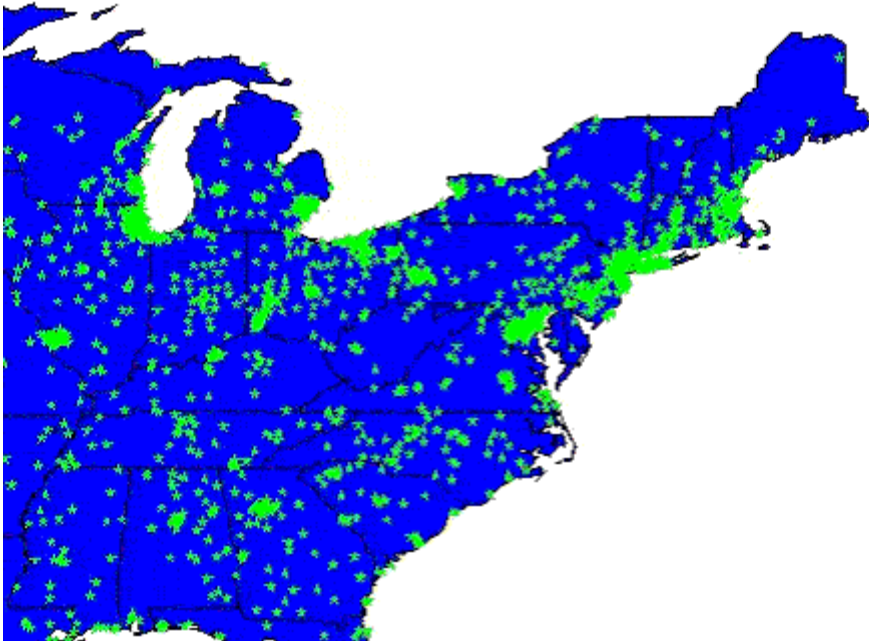
```

</PROPERTIES>
</GET_IMAGE>
</REQUEST>
</ARXML>

```

客户端请求获得的新地图，将如图 7 所示显示：

图 7：使用新的缩放因子并给 US_Cities 层更改符号和显示颜色



图片 4

第 4 步：客户端接受服务器的响应

当空间服务器处理一个请求，其结果是返回一个响应（图 1 的第 4 步）。服务器的响应与其它类型的 ArcXML 文件区别的标志是 RESPONSE 元素。图 8 是影像地图服务对地图请求响应的一个例子。响应包含空间服务器生成的地图的名称和位置。该响应是对图 4 和图 6 请求的一个典型的例子。

图 8：响应的例子

```

<?xml version="1.0" encoding="UTF8"?>
<ARXML version="1.1">
  <RESPONSE>
    <IMAGE>
      <ENVELOPE minx="-91.7857142857143" miny="30" maxx="-63.2142857142857"
maxy="50" />
      <OUTPUT url="http://zhuanghy/output/actc_ZHUANGHY1208107213.jpg" />
    </IMAGE>
  </RESPONSE>
</ARXML>

```


基于 Oracle 的 ArcSDE 配置

在关系数据库管理系统（DBMS）中，ArcSDE 就像管理地理数据的“大门”，使得 DBMS 中数据可以快速向 ArcMap、ArcCatalog、ArcIMS，以及其它互连网络客户端传输。ArcSDE 和 DBMS 的組合使用，有利于将基于传统文件的矢量，栅格和 CAD 数据都移植到一个空间数据及属性数据集成的数据库中。这样，所有的空间数据及一般的商业数据都被管理在传统 IT 框架内，有利于数据的一体化。

ArcSDE 支持许多的数据库软件，因此选择配置的方案是很灵活的。当然，客户端应用从 ArcSDE 获取空间数据的速度快慢的性能取决于多个因素，包括硬件、数据库配置、ArcSDE 配置、客户端处理能力和网络带宽等等。本文将对基于 Oracle 数据库的 ArcSDE 系统的一些基本运行配置方案进行描述。

1、Oracle 及 ArcSDE 简介

Oracle 数据库是由一个例程（Instance）和存储在硬盘上的文件组成的。Oracle 例程是由进程和内存结构组成的。服务器进程（Server Process）从 SGA 与 Oracle 客户端交互，比如 ArcSDE。它负责处理用户提交的 SQL 语句以及读写用户数据。后台进程（Background Process）代表服务器进程与组成数据库的物理文件交互。构成 Oracle 这样大容量数据库的是一系列数据文件，其中存储了如表和索引等这样的对象。

通过 Oracle 服务端程序，ArcSDE 使用 giomgr 进程与数据库交互。每个 ArcSDE 服务都有一个 giomgr 进程在监听用户应用的连接请求、清理断开的用户进程。每个连接 ArcSDE 的客户端应用程序都被指定一个 gsrvr 进程，该进程由 giomgr 生成。通过 Oracle 服务端程序，Gsrvr 提交用户所有的数据库查询及编辑的请求。

2、逻辑数据库设计

如何正确配置 ArcSDE Oracle 数据库，首先要分析这个数据库将的用途。一般地，数据库要么用于联机分析处理（OLAP），要么用于决策支持系统（DSS）。OLAP 数据库的特征就是频繁的数据更改，例如典型的插入或更新操作，这对大用户量的数据库是很习以为常的。这样类型的系统包括订单输入系统、订票系统和打卡系统。DSS 数据库则有所不同。它可以从大量的数据中产生报表信息。一般来说，OLAP 系统的用户群一般都比较大，DSS 用户群相对要少得多。

DSS 系统的主要特性是能够快速地获取大量的数据。大部分的 GIS 应用都选用 DSS 系统，因为它们需要获取用于分析和显示的大量地理数据和属性信息。当然，很多的 GIS 系统在数据更新时候也可以像 OLTP 系统那样工作。由于大多数的 GIS 都采用 DSS 数据库的配置，因此可以根据相应的应用进行数据库的物理设计和逻辑设计。

逻辑设计的目的就是优化配置数据库，使数据库对象按照其活动性相互分离。比如，表和其索引分离，高活动性表和低活动性表也不要存放在一起。如此配置将会获得运行效率极高、并且易于管理的数据库。

一般情况下，只有表和索引被视为对象，其实还存在其它很多对象，比如视图、序列和同义词等等。在操作系统中，Oracle 将对象逻辑地存储在一个或多个数据文件所支撑的表空间内。对于不同类型数据，在 Oracle 中都应当应该分别给它们各自独立的表空间。比如，典型的 ArcSDE 配置将提供最少五个表空间用于存储地理数据，它们分别是 FEATURE，ATTRIBUTE，SPATIAL_INDEX，ORACLE_INDEX 和 SDE。每一个表空间都存储 ArcSDE 和其相关客户端所使用的不同的表或索引。FEATURE 表空间用于存储“F[n]”（F 表），即 ArcSDE 创建的要素表。ATTRIBUTE 表空间存储包含属性信息的业务表。SPATIAL_INDEX 表空间存储“S [n]”（S 表），即空间索

引表。ORACLE_INDEX 表空间存储的是基于要素、属性和空间索引表创建的 ORACLE 索引。SDE 表空间存储 ArcSDE 系统表。除了上述这些 ArcSDE 表空间外,Oracle 数据库应该至少还包含 SYSTEM, ROLLBACK, TEMP 和 USER 表空间, 分别用来存储用于 ORACLE 内部进程的不同对象。

3、物理数据库设计

通常情况下,并不需要考虑数据库的物理布局。只有当数据库运行遇到性能问题时,物理布局才被考虑到。其实,要配置一个高性能的数据库,正如数据库的逻辑设计中的精心安排一样,数据库文件的物理布局设计必须也要同时进行。在任何 Oracle 数据库中, I/O 争用往往是数据库性能的瓶颈。因此数据文件的物理布局应该始于对数据 I/O 情况的识别。要减少 I/O 争用,必须将不同的对象组安排在不同磁盘驱动器和表空间中。

一般情况下, 客户端访问的表空间中的表和索引将会占据大量的 I/O。在 ArcSDE 配置中, 这种情况在 FEATURE、ATTRIBUTE、SPATIAL_INDEX 和 ORACLE_INDEX 表空间中都会出现。如果可能的话, 应当将这些表空间的数据文件各自单独存放到不同磁盘中。典型的 GIS 客户端应用(如, ArcMAP)在普通的会话过程中都会频繁访问这些表空间里的表和索引文件。比如, 当用户在 ArcMAP 中进行缩放或移动操作时, 查询语句将提交给 ArcSDE, 以获取应用程序窗口范围内的要素。执行之一过程的动作是从 S 表的索引表中获取符合条件的要素标识码。然后通过使用 FID 字段的索引, 从 F 表中提取相应的要素几何图形。如果查询中还包含属性查询, 业务表也将被提取。所以, 如果系统存储中每个表空间在物理上都分别存放, 对于 ArcMap 这样的客户端将获得更好的操作性能。

SYSTEM 表空间保存 Oracle 的数据字典, 因此应该被安排在活动性适中的磁盘上。每当 Oracle 对一个 SQL 语句进行解析的时候, 都会检查存放在数据字典中的表。SYSTEM 表空间中只存储数据字典是最好的。

ROLLBACK 表空间用于维护数据库中数据的一致性。对于数据库中的每个事务处理, Oracle 在回退段中生成一个数据的 Before Image。它的作用除了提供回退机制以外, 还得保证在该事务处理提交之前其它客户端查询的读取一致性(read consistency)。一般来说, 给 ROLLBACK 表空间分配 50MB 的空间。并且, ROLLBACK 表空间应该分割存储到不同的磁盘上。

TEMP 表空间用于存储动态由 SELECT DISTINCT, UNION 和 CREATE INDEX 等大数据量的排序操作产生的数据。创建 TEMP 表空间时, 一定要记住, 创建一个索引需要两倍于该索引存储的空间大小。在 ArcSDE 数据库中的空间索引中, 一般情况下, S 表的索引(S_[x])是需要存储空间最大的索引。因此, 为 TEMP 表空间分配的空间至少是该索引存储量大小的两倍。建议把临时表空间与其它表空间物理地分开。

与 SYSTEM 表空间很相似, SDE 表空间是用于存储 Oracle 数据字典的, 这里保存了管理 ArcSDE 例程的元数据。SDE 表空间可以和 SYSTEM 表空间存放在同一个磁盘上。

4、其它数据库文件

Oracle 使用控制文件来保存数据库信息, 如数据库标识符、创建日期、表空间名字、数据文件名, 以及其它用于数据库正常启动和运行的重要细节信息。在数据库使用过程中, 这些文件被 Oracle 不断更新, 因此数据库打开的时候, 它必须是可写的。如果由于某些原因导致该文件不可用, 数据库将不能够正常运行。由于这个原因, 这个控制文件应该包括最少三个镜像。

联机重作(Online redo)日志文件用于记录数据库的所有事务处理操作, 这些操作包括: 插入, 更新和删除。在数据库例程崩溃的时候, Oracle 使用这些文件来最小化数据损失。在一个例程崩溃时, Oracle 读取这些文件, 用以恢复已提交但尚未写入数据文件中的数据。Oracle 至少需要两个 redo 日志文件。一旦一个日志文件写满了, Oracle 开始写另一个日志文件。这个处理过程称为日志切换, 它会触发检查点事件, 而检查点是相当耗资源的。大多数 ArcSDE 数据库保持静态, 随时间推移, 数据库一般是变化不大的。为了把日志切换的次数减至最少, 一般来说, 需

要创建两个 50 兆的联机重作日志文件。重做日志文件的使用频率并非特别高，可以和数据库的其它文件存放在一块。

5、Oracle 进程参数配置

数据库例程的配置在很大程度上决定了数据库的性能。在数据库启动的时候，一个系统全局区域（SGA）都会分配，并且数据库的后台进程同时启动。SGA 是存放数据库信息的一块内存区域，供数据库用户共享。Oracle 后台进程和 SGA 的组合称为 Oracle 例程。在启动的时候，Oracle 使用 init[sid].ora 参数文件来配置该进程。

在 Oracle 上运行 ArcSDE 数据库的时候，一些关键参数设置是尤其举足轻重的。DB_BLOCK_SIZE 和 DB_BLOCK_BUFFERS 参数控制 SGA 中 BLOCK 缓存的大小。在很多情况下，给该缓存分配 1 / 3 或更多的系统可用内存。在内存中，BLOCK 缓存存储了先前访问过的表和索引数据块，减少系统的 I/O 需求。将 DB_BLOCK_SIZE 设置为 16,384 字节是非常合适的，当这个参数低于 8,192 字节时，可能会降低数据库性能。DB_BLOCK_BUFFERS 参数乘以 DB_BLOCK_SIZE，就是 BLOCK 缓存的总大小。例如，将 DB_BLOCK_BUFFERS 设置为 100,000，并将 DB_BLOCK_SIZE 设置为 16K，那么 BLOCK 缓存的大小就是 1.6G。

ArcSDE 和 DSS 数据库的另外一个重要配置参数，是 DB_FILE_MULTIBLOCK_READ_COUNT。该参数决定了在一次读取操作的时候，从硬盘读取的数据量。磁盘 I/O 过频，将降低系统的性能，磁盘读取的次数越少越好。在 DSS 数据库中该参数经常被设置成一个很高的值，但是当该参数被设置为 8 时，ArcSDE 似乎能运行得更好。偏高的参数值反而会影响 Oracle 的执行计划。

共享池（shared pool）是 SGA 中的一个重要部分，它包括数据字典缓存（data dictionary cache）和库缓存（library cache）。库缓存存储了最近解析过的 SQL 语句。共享池的大小由 SHARED_POOL_SIZE 参数控制。推荐设置大小为 55MB，最大值为 75MB。

SGA 的还有一个重作日志缓冲（log buffer），它保存着尚未提交地数据更新。日志缓冲由 LOG_BUFFER 参数所控制。如果内存空间允许，可以将该参数设置为 10MB。

6、结论

以上是基于 Oracle 的 ArcSDE 的一些通用配置方案，然而，要让你的空间数据库能够向客户端提供高性能的服务，以上这些是远远不够的，还需要很多很多尝试。作好 Oracle 数据库物理布局是获得高性能数据库的起点。ArcGIS 中国培训中心开设为期五天的 ArcSDE Oracle 管理员教程，可以为你提供一套完整的空间数据库配置解决方案。

将注记从 coverage 格式转化为 geodatabase 格式

地图注记的创建有这么几种方式：从 coverage 中导入、将地图标注转化为注记、或者在 ArcCatalog 中创建新的注记要素类。本文将着重介绍第一种方法的使用，将已有的 coverage 注记转化成 geodatabase 注记。

与其它的地图要素类一样，annotation 要素类可以作为要素集的一部分或者作为一个单独的要素类存在。Geodatabase 中的注记可以是要素关联也可以是非要素关联的。要素关联的注记参与它所描述的要素类（也叫源要素类）的复杂的关系。源要素控制相关联注记的位置和生命范围。

转化为 geodatabase annotation 的 coverage annotation 不能是要素关联的,但可以通过在 ArcCatalog 中创建关联类而与另一要素类中的要素相关联。但是,对于转化的 coverage annotation 的要素关联行为,比如注册的自动创建和更新,可以通过定制代码来实现。

何为 Coverage Annotation

Coverage 包含要素的要素类包括弧段、结点、label 点、多边形还有注册。注册还可以进一步组织成子类存储。比如,描述公路的 coverage 可能含有线状要素的注册以及点要素的单独注册。

Coverage annotation 在 TXT 文件中存储了字符串文本标注,连同包括文本符号的数目、位置、配置以及大小规格的相应的字符。在 coverage 中每个注册子类都有一组这样的文件。注册子类可能有一个存储属性的文本属性表(TAT)。TAT 文件可用于将同一个标注与多个地理要素关联,并且,用 TAT 文件或者一个相关的表代替包含在 TXT 文件中的字符串来绘制文本。TAT 文件是将 coverage 注册子类转化成 geodatabase 注册要素类所必需的。

用于转化 coverage 的 Convert Coverage Annotation 工具舍弃了所有的 TAT 条目。但是,通过在目标注册要素类中定义与 TAT 条目相对应的字段可以保持这些条目。然而这些替代的条目在转化过程中并没有转化,使用 Arc 命令: ADDTEXT,通过定义目标注册要素类的字段,就可将 TAT 文件中的那些条目将依次转化进入 geodatabase 中。

创建要素类

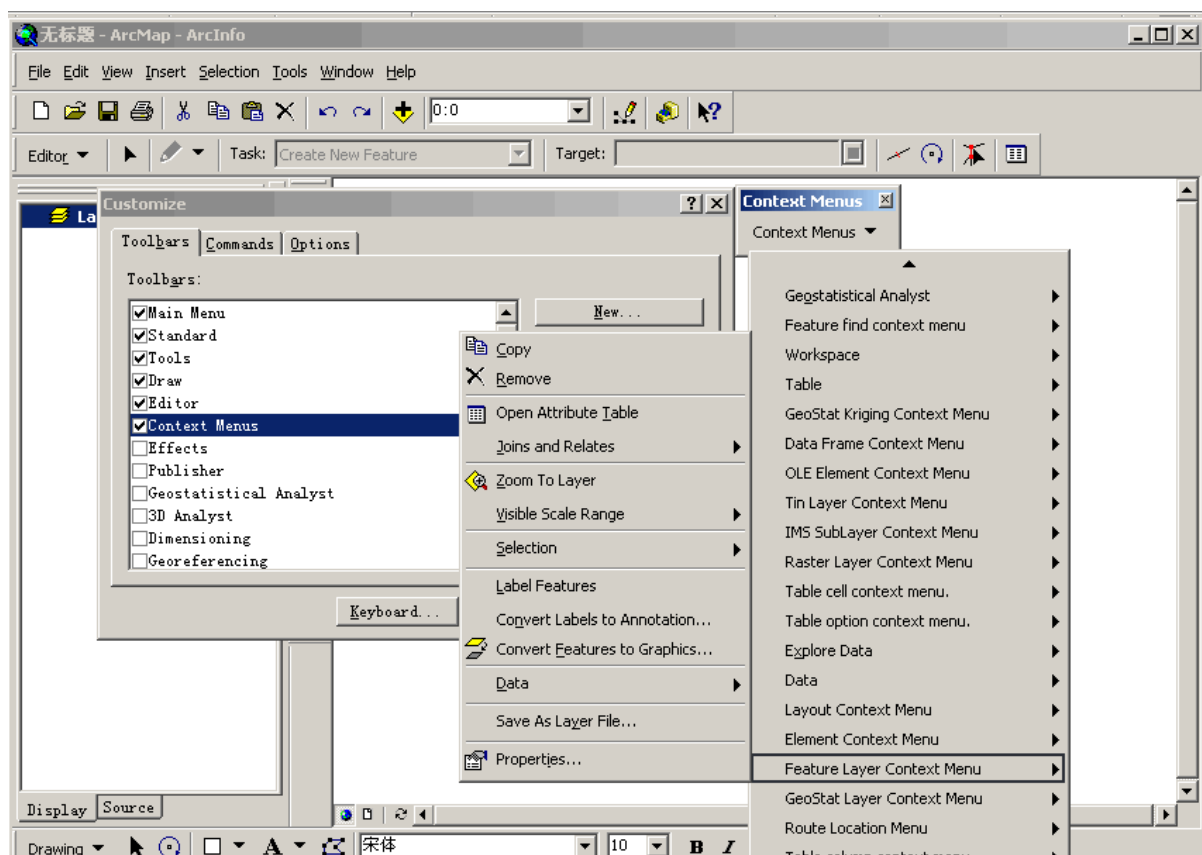
使用 ArcCatalog 创建要素类来存储转化的注册。因为在 coverage 中与注册相关联的空间要素通常也被转化,并且在同一坐标空间保持一致,建议创建一个要素集,把这些要素类保存在一起。创建了新的要素集,从原始 coverage 中导入空间参考,再创建一个新的注册要素类。

指定参考比例尺也是很重要的。它决定了注册显示的字体大小且作为标记比例尺,在放大缩小的时候文本也会随着放大缩小。地图比例尺与预期的输出或视图比例尺应该相等。比如说,如果参考比例尺设置为 1:10,000,当地图显示是 1:5,000,注册将会放大两倍。如果注册要素类是在一个要素集中创建的,参考比例尺的单位将自动与空间参考中指定的单位匹配。在新的要素集、注册及其它要素类创建成功后,关闭 ArcCatalog 并且打开 ArcMap。

定制 ArcMap

在 ArcMap 中转化注册需要一些小的定制,因为 Convert Coverage Annotation 命令在任何标准工具条上都是找不到的。但是可以通过 ArcMap 中的 Customize 对话框将这个命令加到操作界面上。

因为注册转化命令被添加到环境菜单中,只有 coverage 注册显示在数据框中时,这个命令才会被访问。一旦创建了要素集并且定制了 ArcMap 界面,coverage 转化就可以开始了。



图片 1

Coverages 的转化

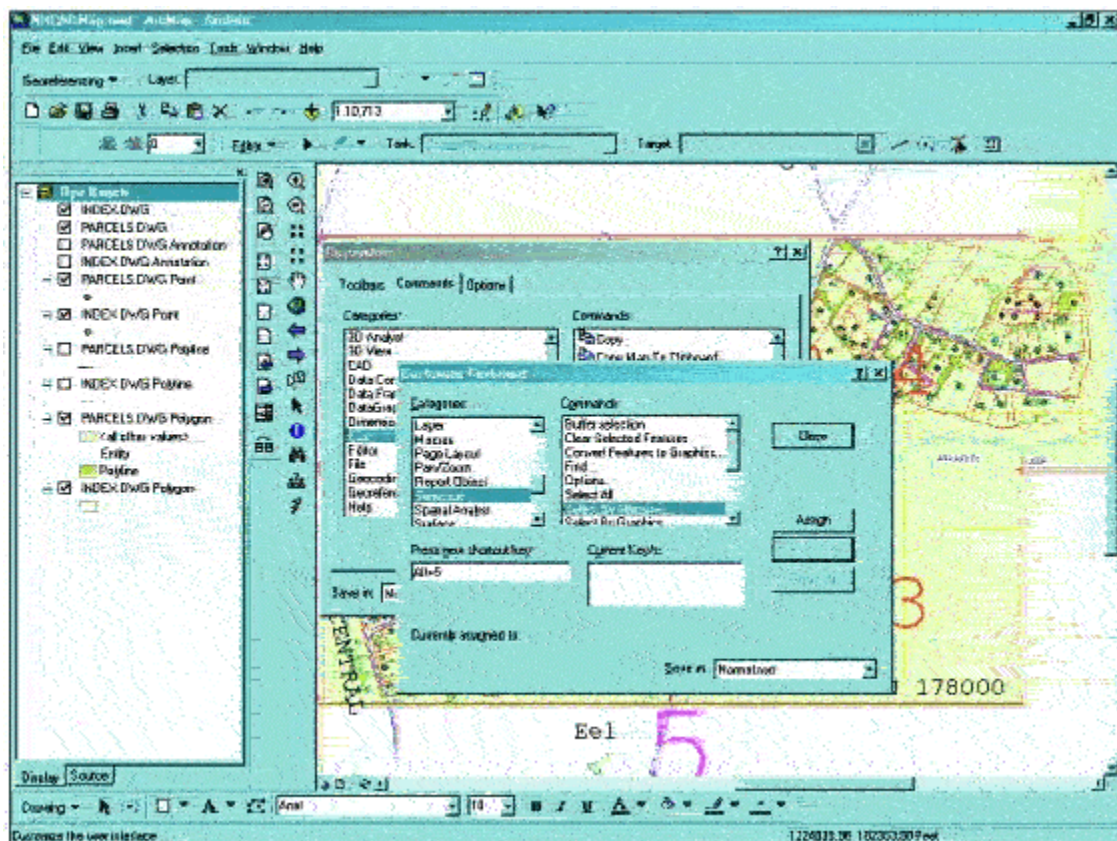
在 ArcMap 中，annotation coverage 是以图层加入数据框中。为新图层设置属性，指定注记如何显示，右键单击 annotation 图层并在弹出的环境菜单中选择 **Convert Coverage Annotation** 开始转化操作。在 **Convert Coverage Annotation** 对话框中，选中需要转化的 coverage 注记要素类。多个 coverage 注记要素类可以转化成一个 geodatabase 注记要素类。在转化处理完毕后，查看新的 geodatabase 注记。转化得到的注记可以被编辑，更改字体、大小或者其它特性。

设置热键加速 ArcMap 操作

加速 ArcMap 操作的途径之一就是为常用的菜单选项创建热键。例如，在 ArcMap 中，频繁使用要素属性信息检索要素，这时可以为 **Select by Attribute** 这个菜单选项设置热键，以便迅速获得 **Select by Attribute** 对话框。设置热键对于 ArcMap 操作的快速完成是非常有效的，对于一些多级菜单里的菜单选项，设置热键将明显减少鼠标的单击动作，而且节约操作时间。与鼠标操作以及一般的快捷键相比，热键效率明显要高得多，因为它无需展开菜单便可以直接执行命令了。ArcMap 中的热键设置是非常容易的。需要注意的是，可以给一个菜单选项同时设置多个热键，但是每个热键只对应一个菜单选项。

ArcMap 拥有及其可灵活的工作环境，用户可以根据功能需求定制自己的操作界面。所有的定制都可以存储在三个不同级别的地图文件中，即 Normal 模板(Normal.mxt)、基础模板(*.mxt)以及地图文档(.mxd)。这三个文档的读取顺序依次是 Normal 模板、基础模板、地图文档，这个顺序是非常重要的，因为前面加载文档的定制内容将被随后载入的文件所覆盖。比如，在 Normal 模板中你将“Select by attribute”热键设置为“CTRL+M”。同时，在一个名为 test.mxd 文档中你又将“CTRL+M”设置成为“Select by location”选项的热键。这样，你在启动 ArcMap 后，如果直接用 Add data 选项加载数据，那么此时“CTRL+M”依旧是“select by attribute”的热键。一旦你在启动 ArcMap 后，就选择添加 test.mxd 地图文档，那么此时的“CTRL+M”将变为“select by location”选项的热键。

Normal 模板存储了所有 ArcMap 的图形界面设置。Windows NT 用户的 Normal 模板存放在该用户相应的个人 file 文件夹中，而 Windows 2000 中，则存放在 Documents and Settings 的登陆用户的相应文件夹内。也就是说，尽管同一台机器上所有的登陆用户运行的都是同一个 ArcGIS 程序，他们还是能够拥有自己独立的个性化界面定制。基础模板可以存储所有的地图元素，这样，同一部门或一项工程中生产的所有地图都能够做到标准化。ArcMap 给用户提供了专题基础模板，当然，任意一个地图文档都可以保存为基础模板。



图片 1

设置热键

- 1、ArcMap 中，从菜单中选择 Tools > Customize。
- 2、在 Customize 对话框中，选择 Commands 标签，点击 Keyboard 按钮。
- 3、在 Customize Keyboard 对话框中，选择对话框左边方格中即将用于制作热键的命令所在的菜单或工具类别。
- 4、这时与该菜单或者工具类别相应的命令将会出现在右边的方格中。选择你所需要设置热键的

命令。所有原先为这个命令指定的热键，将在 Current key/s 文本框中列出。

5、点击 Press new shortcut key 文本框，在键盘上按要使用的热键。如果这些热键已被指定给其它命令，那个命令名称将出现在“Currently Assigned To:”旁边的对话框中。

6、在 Save in 的下拉列表中选择你所需要存储这些热键的模板。

7、点击 Assign 按钮。新的热键将出现在 Current Key/s: 列表中。

8、关闭 Customize Keyboard 和 Customize 对话框。

热键设置完成。这时热键组合会出现在菜单中相应的命令后面。如果这个热键被定制到 Normal 模板中，那么这个热键可以应用到所有新建以及已有的地图文档中，前提是当然是没有热键冲突。如果将热键定制到基础模板中，那么对于基于该基础模板创建的任一地图文档，这个热键都可用。如果只是将热键定制到某个地图文档中，那么它也就仅仅在该地图文档中可用。

同理，ArcCatalog 在 Normal 模板中 (.gxt) 存储所有的界面定制内容。从菜单中选择 Tools>Customize，然后使用与 ArcMap 相同的方法，也可以为 ArcCatalog 菜单命令设置热键。在 ArcGIS 的帮助文档中搜索 “customizing” 关键字，你将学会定制 ArcMap 界面的更多方法。

使用 ArcGIS 8.2 加载 ArcSDE 栅格数据

ArcGIS 8.2 提供了两种将栅格数据加载入 ArcSDE 的方式——Raster to Geodatabase 向导以及 Raster to Geodatabase 工具。向导是面向新手的，逐步介绍数据加载的过程。工具是为更多用户准备的，因为所有的参数以及选项都在一个对话框中出现。

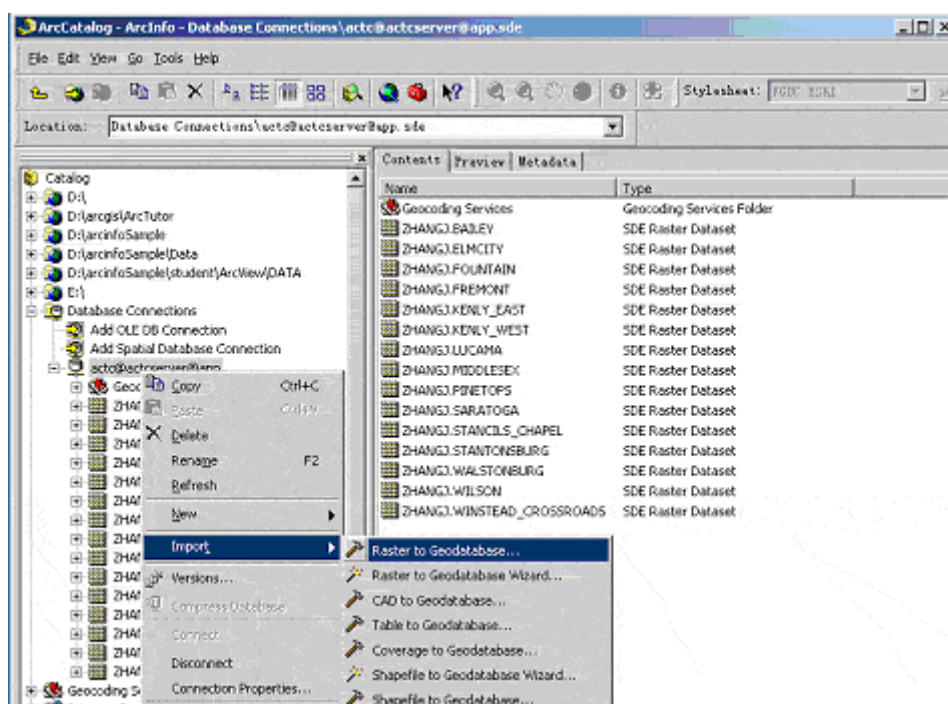
ArcInfo 和 ArcEditor 中的 ArcCatalog 与 ArcToolbox 都有将一个或多个栅格数据加载入 ArcSDE 数据库的功能。从 ArcToolbox 中获得 Raster to Geodatabase 的工具或向导，只要展开 ArcToolbox 中的 Import to Geodatabase 目录就可以选择所需要的工具了。在 ArcCatalog 中，右击单击需要导入到 ArcSDE 中的栅格数据集，在弹出的环境菜单中选择 Raster to Geodatabase 的向导或工具即可。

向导和工具都支持所有 ArcGIS 中可用的栅格格式的加载，包括 GRID、IMG、TIFF、MrSID、JPEG、ESRI BIL_BIP_BSQ、GIF、PNG、ERDAS7.5、LAN 以及 GIS、BMP、ERMapper ERS、RAW、ESRIGridStack、CADRG、ADRG、DTED levels 1 和 2、NITF2.0 以及 2.1、CIB 等等。

本文描述如何以批处理方式使用 Raster to Geodatabase 工具来创建一个无缝栅格图以及栅格目录表。在第一个应用实例中，许多空间上相邻和（或者）重叠的共有同一空间参考的栅格数据将被镶嵌成为一个影像，这个影像可以作为 GIS 数据及其它应用的背景图，在应用过程中，它将保持稳定的存储状态。第二个实例是制作一个栅格目录表，用以满足数据的频繁更新，原始栅格数据的存储。在栅格目录表中，尽管每个栅格图象都是单独存储的，但是目录表管理使得这些栅格数据就像在同一个数据集中一样。

制作无缝栅格图

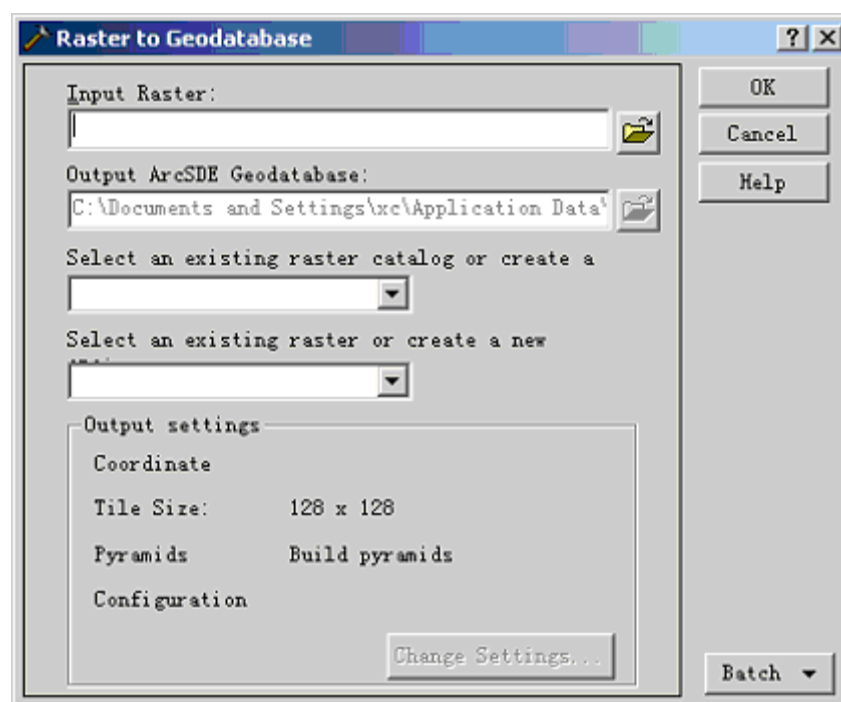
以批处理方式将一系列栅格文件加载入数据库并做成无缝栅格图或者栅格目录表是非常方便快捷的。在 ArcCatalog 使用 Raster to Geodatabase 工具对话框来批处理加载栅格数据是最简便的方法。将所有的栅格文件放到同一个文件夹下还可以简化这个处理过程，在这里的例子假设这一切都已做好。



图片 1

图 1: 启动 ArcCatalog, 右键单击选择栅格数据所需要导入的数据库, 选择 Import>Raster to geodatabase
第一步: 启动工具

在 ArcCatalog 中, 右键单击数据库连接并从弹出的环境菜单中选择 Import>Raster to Geodatabase 工具。Raster to Geodatabase 工具对话框弹出, 并且路径已经填好。

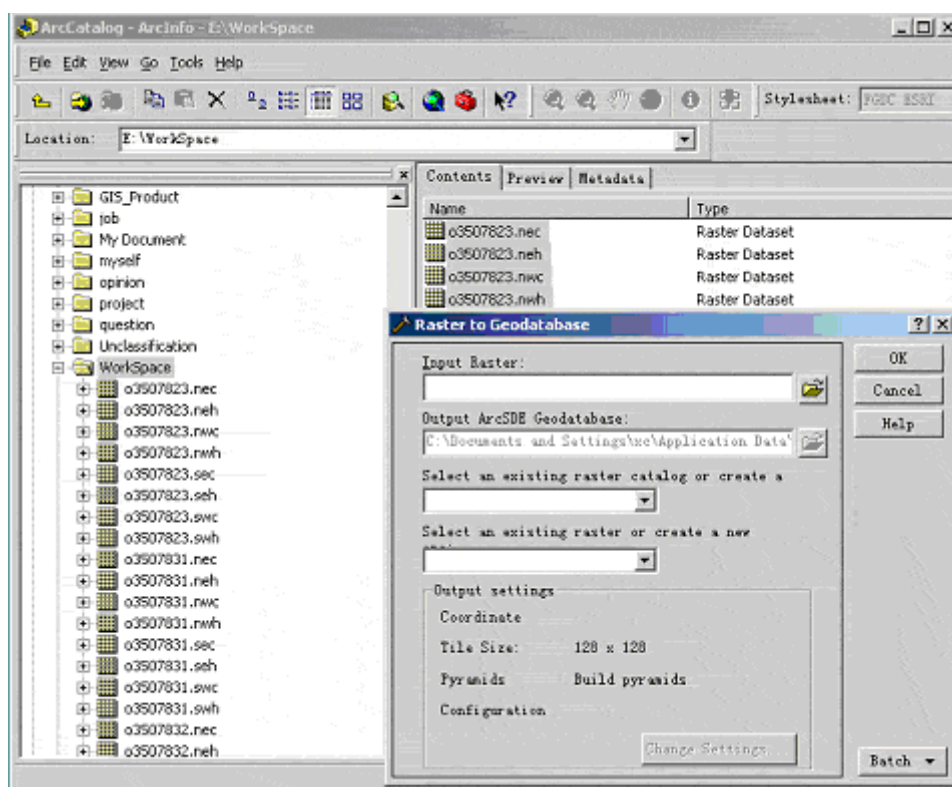


图片 2

图 2: Raster to Geodatabase 工具对话框

第二步: 选择输入栅格数据

找到栅格数据存储目录并将所有栅格选中, 将它们拖至对话框的 Input Raster 输入框。这将使对话框底部的批处理 (batch) 方式自动显示, 并且输入、输出路径已经填好。



图片 3

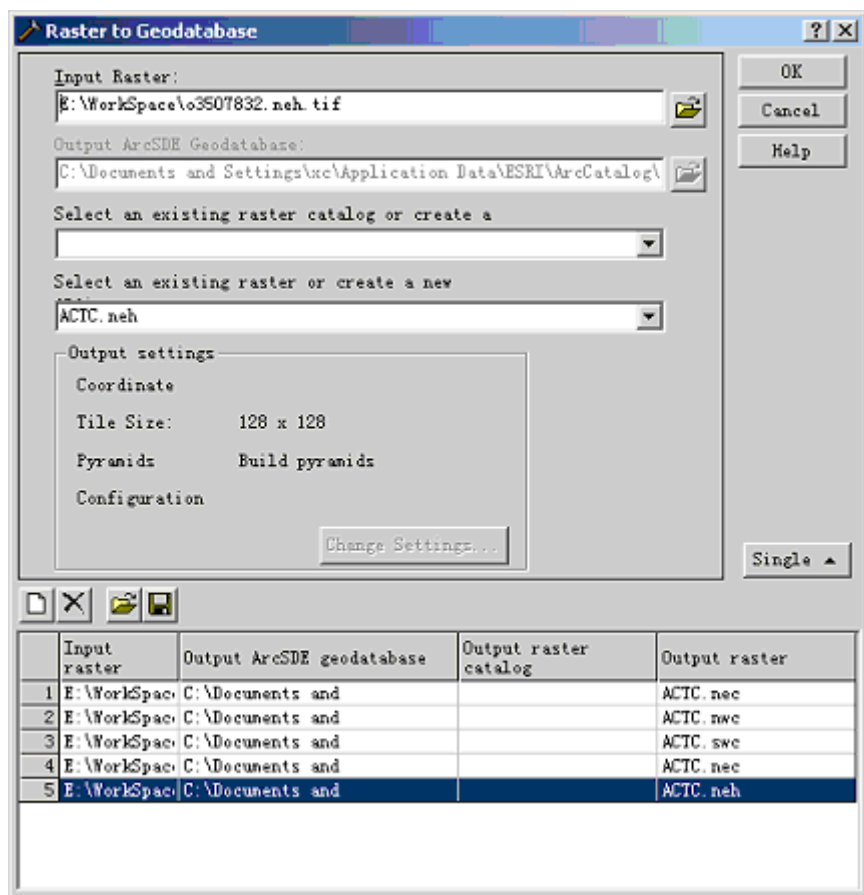
图 3: 在 Input Raster 对话框中, 定位到所需加载的所有栅格数据

第三步: 选择输出栅格文件名

在批处理方式中选择输出栅格列, 并键入所需栅格数据文件名。

第四步: 改变金字塔以及统计值设置

默认情况下, 加载工具都将为每个加载的栅格文件建立金字塔以及计算统计值。然而, 栅格镶嵌操作会删除加载过程中每个的栅格的金字塔层以及统计值, 为了节省资源和处理时间, 只要设定最后一个添加的栅格制作金字塔以及计算统计值就行了。要进行这个调整, 只要在批处理模式下选择最后一行除外的所有行, 选中后单击 **Change Settings** 按钮获得 **Output Settings** 对话框。在对话框的 **Storage** 标签中, 不选 **Built Statistics** 复选框并选择 **Do not build pyramids** 选项。单击 **OK** 关闭 **Output Settings** 对话框。



图片 4

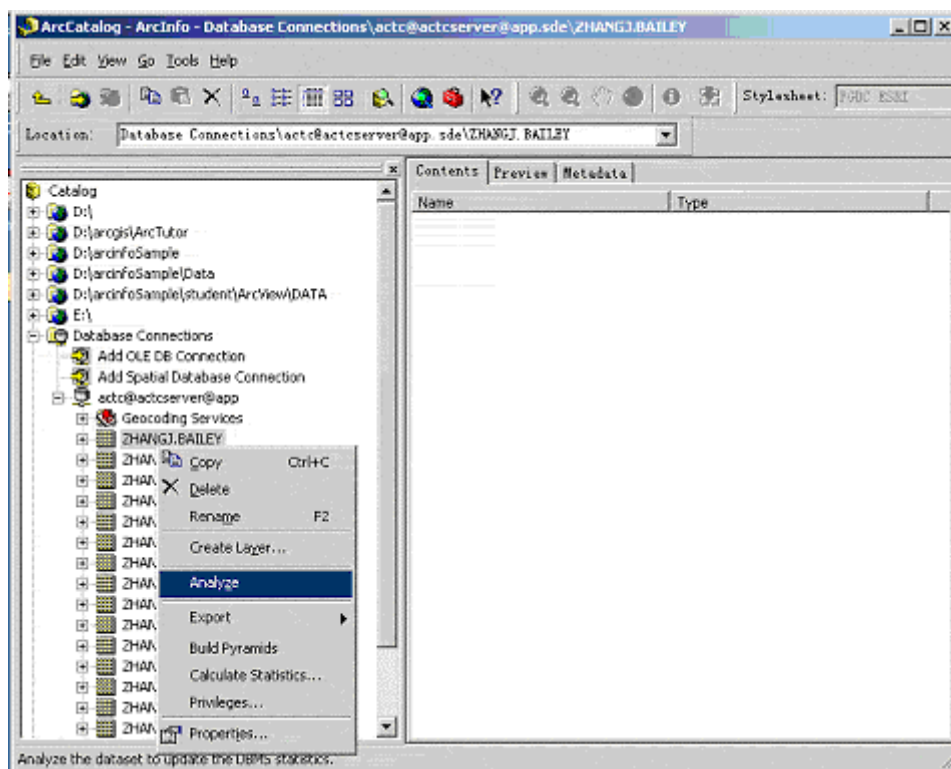
图 4: 给 output raster 列中的所有文件命名

第五步: 运行批处理程序

在 Raster to Geodatabase 对话框中, 单击 OK, 进行栅格数据导入 ArcSDE。

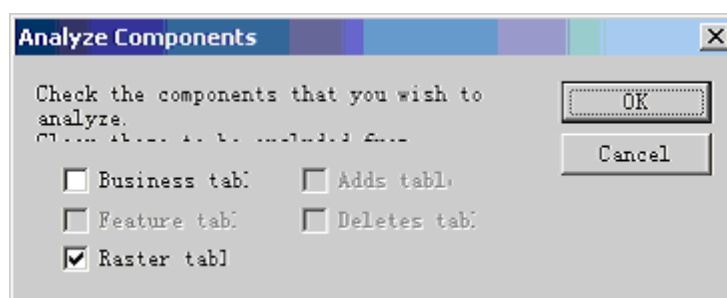
第六步: 更新 DBMS 级统计值

在加载操作完成后, 分析栅格表并进行 DBMS 级统计以获得最佳性能。刷新数据库, 右键单击新建的栅格镶嵌图, 在弹出菜单中选择 Analyze。在弹出的 Analyze Components 对话框中, 选择 Raster table 作为分析对象, 最后单击 OK。



图片 5

图 5: 加载完成后选择 Analyze 进行 DBMS 分析



图片 6

图 6: 选择 Raster table 作为分析对象

上述操作介绍了如何生成一幅无缝栅格图，它适合作底图或者其它只读情况。对于那些需要频繁更新的数据，制作一个栅格目录表是一个更好的选择。制作栅格目录表过程也可以使用 Raster to Geodatabase 工具或向导完成。

制作栅格目录表

第一、二步：启动 Raster to Geodatabase 工具并选择输入的栅格文件，同前例。

第三步：选择输出栅格目录的表名

一旦批处理模式出现在对话框的底部（这时输入的栅格文件已全部列出），选择输出栅格目录表列，键入输出的栅格目录表名。

第四步：检查设置

在栅格目录表中，每个栅格都是单独实体。金字塔和统计值都必须单独创建，并且不影响目录表中的其它栅格数据的金字塔以及统计值。因此，没有必要更改统计值以及金字塔选项的默认设置。要更改栅格目录中哪个栅格文件名，选择批处理中的 Output raster catalog 的列，为各个栅

格数据键入所需文件名。

第五步：运行批处理加载程序

在 Raster to Geodatabase 对话框中，单击 OK 执行批处理加载任务。

第六步：更新 DBMS 统计

要分析栅格目录表，使用 ArcSDE 提供的 sdetable 命令行工具，这个语句是

```
sdetable -o update_dbms_stats -t doq_catalog -K r -m .....
```

结论

现在，ArcGIS 仅仅适用于 Windows 环境中，这样 Raster to Geodatabase 工具或者向导仅能使用于 Windows 操作平台上。在不要求栅格地图色彩映射表而且输入数据具有同一空间参考的时候，这些工具是非常管用的。命令行工具，如 sderaster 命令，加载的参数要比 Windows 界面的栅格加载复杂得多，可以应用于 UNIX 环境中。

压缩 Geodatabase

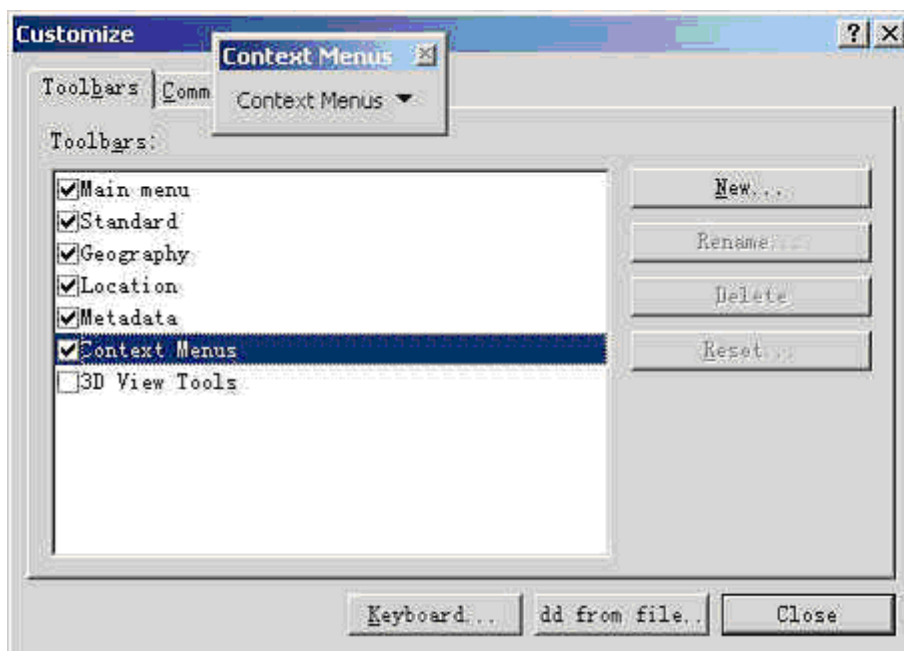
ArcSDE Geodatabase 支持长事务处理，通过提供版本满足了多用户同时访问数据库的需求。版本实质上就是 Geodatabase 命了名的状态，并且多个版本可以交互出现。版本 Geodatabase 通常有一个最高级别的叫做 DEFAULT 的版本，表示数据库中数据的初始状态。其它版本可以创建并且组织成为分层状、环状或者工作流扩展历史状态的形式。

对要素类或者对象类的操作改变并没有直接应用到版本 Geodatabase 的业务表（B 表）中。相反，这些编辑变化都保存在添加表（A 表）和删除表（D 表）中。这些表是根据被访问操作并有所更改的已注册的要素类和对象类创建的。因为每次对要素类或者对象类编辑后，A 表和 D 表中的行数都会增加，时间久了执行效率就会下降。

周期性的压缩 geodatabase——将不需要的数据库状态（也就是版本）及那些已经没有存在意义的行状态删除——将会提高运行效率。压缩操作需要数据库中所有状态的一个独特锁定，这个锁定将使 geodatabase 不可用，直到压缩操作完成为止。这个操作权限只有 SDE 管理员拥有，同时只有 ArcInfo 或者 ArcEditor 版本的 ArcCatalog 中才有这个操作命令。压缩命令必须要先添加到 ArcCatalog 中。

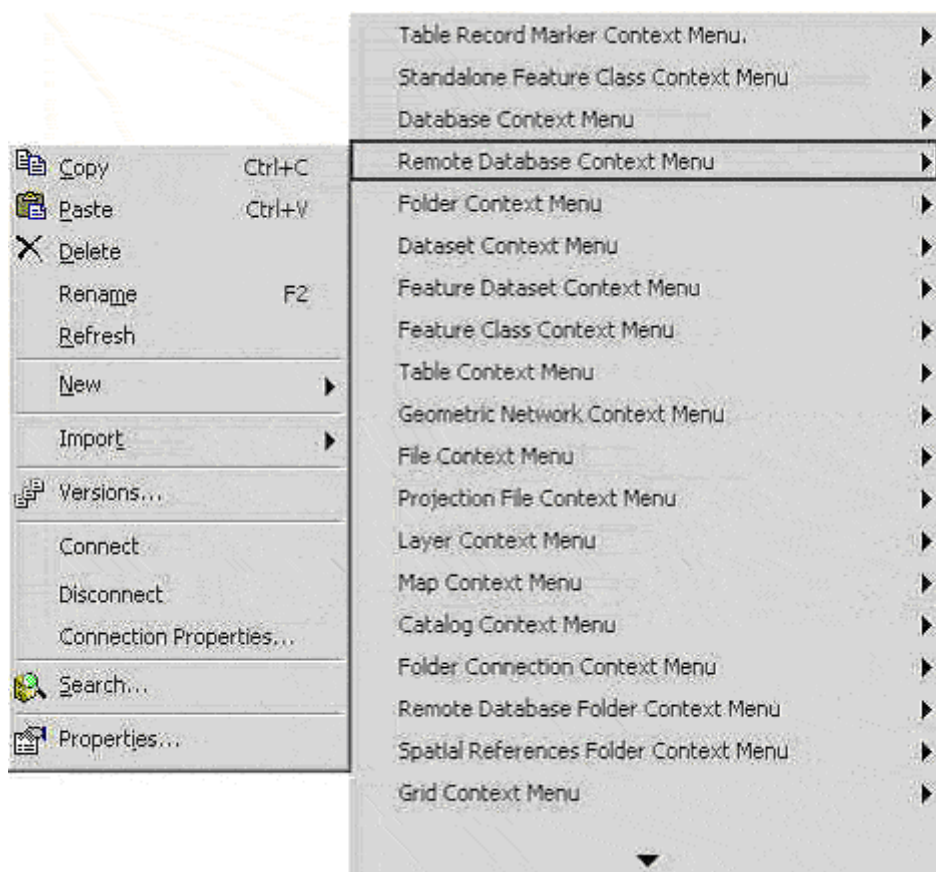
- 1、ArcCatalog 中，单击 Tools>Customize。

- 2、在 Customize 对话框中选择 Toolbars 标签，勾上 Context Menus 的复选框。如图 1 所示。



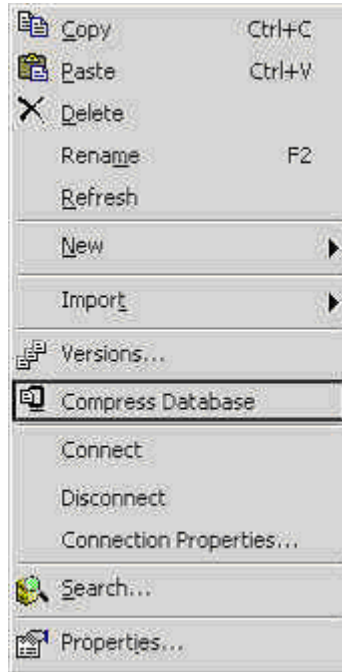
图片 1

3、单击 Context Menus，在下拉列表中选择 Remote Database Context Menu。如图 2 所示。



图片 2

4、在 Customize 对话框中，选择 Commands 标签。从左边的长方框中选择 Geodatabase Tools 并把右边框中的 Compress Database 命令拖至 Remote Database context menu 中。如图 3 所示。



图片 3

5、选择 Close 按钮，关闭 Customize 对话框。

为了得到最优结果，最好先通过协调每个未完成版本与 DEFAULT 版本的冲突，得到新的 geodatabase。并没有必要把所有版本提交到 DEFAULT 版本中或者删除所有的未完成的版本。在 ArcCatalog 中，以 SDE 管理员身份创建一个 geodatabase 的连接。右键单击这个新建的数据库连接并从环境菜单（context menu）中选择选择 Compress Database。

压缩操作结束后，如果需要，重新创建版本并且使用 ArcCatalog 中的 Analyze 命令更新统计。这个 Analyze 命令更新的统计包括业务表（B 表）、要素表（F 表）、delta 表（A 表和 D 表合称），以及这些表的索引表。

进行一个要素集的 analyze 操作的时，这个所有要素集中所有的要素类同时完成 analyze。如果要素集中包含几何网络，网络中的表也被 analyze。定期对 geodatabase 进行 analyze 操作（尤其是对数据库进行改动后），也不失为十分可行的方法。关于版本的更多信息，可以查看《Modling our world》以及《Building a Geodatabase》这两本书。

ArcGIS 中的影像配准

ArcGIS 拥有强大的交互式栅格及影像数据配准工具。如果你已经用过 ArcView 影像分析扩展模块的影像配准功能，你会发现 ArcGIS 中提供的工具与之类似。这个练习中使用到的影像和矢量数据描述了犹他州盐湖城的一个高速公路建设工程中的一些变化。

因主办 2002 冬季奥运会，犹他州已经为此准备多年——建设比赛地点、为与会者提供住处以及其它配套设施。设施中有一个很重要的改进，即 15 号州际公路约 16 英里的重建，刚好穿过盐湖城市区的西部。重建工作始于 1997 年并于 2001 年 7 月完成。全部工程的花费接近二十亿美元。工程是在犹他州交通部的指导和授权下，由 Parsons Brinkerhoff 主管，工程如期完成，并未超出预算。

模型简介

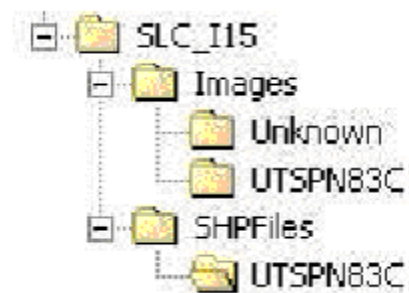
本练习所采用的高质量数据是由 AirPhotoUSA 和 Tele Atlas North America 提供的，它们是 ESRI 的两个商业合作伙伴。练习数据包含两张经过重采样的航空相片，以 JPG 格式存储（Slc_9910.jpg 和 Slc_0107.jpg），一幅影像的 world 文件（Slc_0107.jgw），一个街道中心线的 shapefile（teleatll），以及影像配准控制点（controlpt）。矢量数据是以 Utah State Plane NAD83 Central Zone 配准的，单位是英尺。

街道的矢量数据是从 Tele Atlas North America 的犹他州盐湖城数据集中裁切出来的。矢量数据集被合成为单个的 shapefile，还在一个 ArcMap 的图层文件中含有简单的图例。其中一个 Shapefile 文件包含七个控制点，位于城市干道交叉点上，用于影像配准。

由 AirPhotoUSA 提供的影像包括两张航空相片，描述了在 I-80 立交枢纽周围的 I-15 的重建范围。这两个影像是分别在 1999 年 10 月和 2001 年 7 月获取的。为了缩减文件大小以使数据能更容易从网上下载，这些图像经过重采样，由原来的两英尺地面分辨率降到了大约八英尺并以 JPG 文件存储。1999 年 10 月的影像是未配准的。2001 年 7 月的影像是以 Utah State Plane NAD83 Central Zone 配准的。在对 1999 年的影像配准后，可以比较两幅影像，会观察到新的建筑物和一些季节差异。

获取练习数据

启动 ArcGIS 之前，在资源管理器或其它文件管理程序中创建一个目录存放练习中使用的数据。创建一个 SLC_115 文件夹。再创建 Images 和 SHPFiles 两个子文件夹。在 Images 下，创建 Unknown 和 UTSPN83C 文件夹。在 SHPFiles 下，创建一个 UTSPN83C 文件夹。完整的目录结构显示在图 1 中。下载 SHPFiles.exe 和 JPGFiles.exe 两个自解压文件（www.casw.com.cn）。双击 JPGFiles.exe，将 Slc_9910.jpg 解压到 \Images\Unknown 目录下，将 Slc_0107.jpg 解压到 \Images\UTSPN83C 目录下。双击 SHPFiles.exe，将其内容解压到 \SHPFiles\UTSPN83C 目录。



图片 1

在 ArcMap 中配准影像

栅格数据可以通过扫描地图、航片及卫片来获取。扫描的地图通常不包含表明影像的地表位置信息。从航空相片和卫星相片上获得的位置信息往往不适合执行分析，或者与其它数据对齐显示。与其它空间数据一起使用栅格数据，需要把栅格数据对齐或配准到地图的坐标系统。

配准栅格数据定义了它的地图坐标位置，并指定了联系数据与其地表位置的坐标系统。配准栅格数据使它能与其它地理数据一起被查看、查询和分析。本练习将描述激活 ArcMap 配准工具、配准一幅航空相片以及将其保存为配准好的影像的过程。关于配准的更多内容，可以查看 ArcMap 在线帮助中的“About Georeferencing”。

加载数据

开始这个练习，要激活 Georeferencing 工具条以及添加矢量数据和 1999 年的航片。

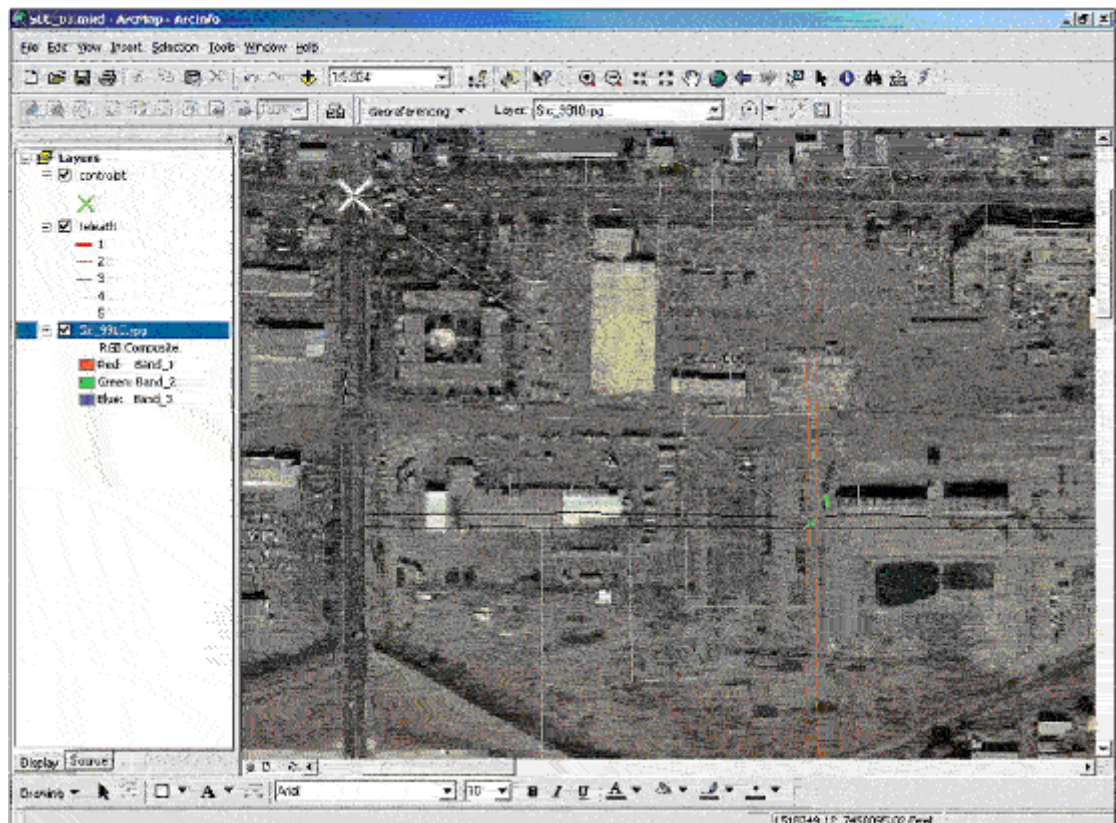
1. 打开 ArcMap, 加载 Georeferencing 工具条, 通过选择 View > Toolbars > Georeferencing。把工具条放在完全可见的位置。
2. 点击 Add Data 按钮, 漫游到\SHPPFiles\UTSPN83C 目录, 选择 teleatl.lyr 和 controlpt.lyr。每个图层上的红色惊叹号表示这些图层文件需要与它们对应的 shapefile 连接起来。右键点击每一图层, 选择 Properties, 点击 Sources 标签上的 Set Data Source 按钮, 漫游到\SHPPFiles\UTSPN83C 目录, 点击相应的 shapefile。
3. 右键点击 controlpt.lyr, 从环境菜单中选择 Open Attribute Table。查看每一层的数据。EASTING 值是从 1,151,600 到 1,153,300, NORTHING 值是在 7,440,300 和 7,542,200 之间。teleatl1 和 controlpt 都是在犹他 State Plane NAD83 Central Zone。为了让数据正确显示, 改变地图单位和显示单位。右键点击目录表中的 Layers 数据框, 选择 Properties, 点击 General 标签, 从地图单位和显示单位的下拉框中选择 feet。
4. 向模型中添加 1999 年 10 月的影像, 点击 Add Data 按钮, 从\Images\Unknown 目录选择 Slc_9910.jpg。在 Create Pyramids for Slc_9910.jpg?对话框中选择 Build Pyramids。忽略指明影像丢失空间参考信息的提示, 点击 OK。
5. 在目录表中右键点击 Slc_9910.jpg, 从环境菜单中选择 Zoom to Layer。注意显示在地图区右下方的地图单位。影像的坐标很小, 大概在-1,000 到+1,500 之间, 表明这一层没有用 Utah State Plane 来投影。

测量者经常有意识地摆放地面控制板, 用易于辨认的点来辅助影像纠正和立体建模。注意在 Slc_9910.jpg 的角上和中心线上的七个白色的小×号。这些是代表真实控制点的假想点, 它们创建在影像上, 辅助显示在 controlpt.lyr 上的测量控制点的配准。你可以放大到这些点看一看。将当前地图文档存为 SLC_01。

配准 1999 年的影像

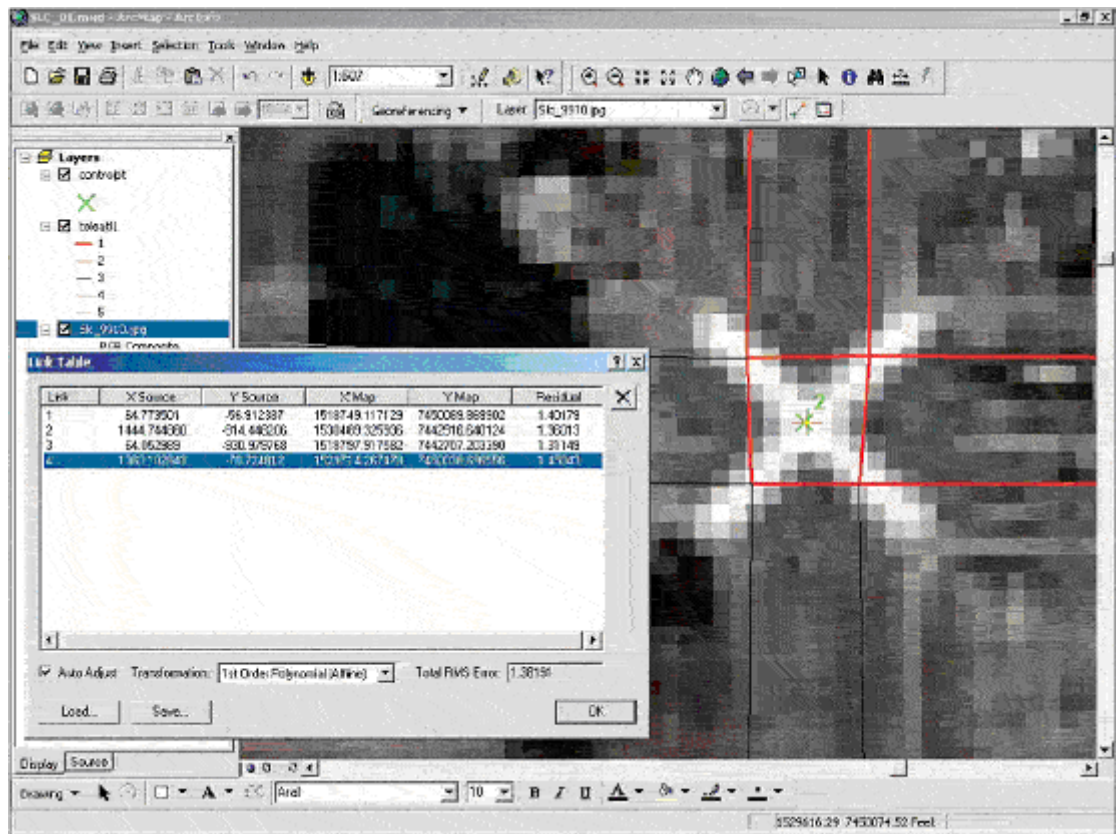
下一步是用影像和街道图层上的控制点将 1999 年 10 月的影像配准到犹他 State Plane 坐标空间。在 ArcGIS 中配准影像与使用 ArcView Image Analysis 扩展模块配准影像的过程很相似。影像被拉伸和扭曲到矢量数据的投影控制点。

1. 右键点击街道图层, 即 teleatl1, 选择 Zoom to Layer。
2. 在 Georeferencing 工具条上, Layer 设置为 Slc_9910.jpg, 点击下拉菜单, 选择 Fit to Display。这会使 Slc_9910.jpg 的进行全局显示。
3. 放大到影像的西北角, 使数字 1 边上的绿色×(第一个控制点)与影像上的白色×都可见。点击 Georeferencing 工具条上的 Add Control Points 按钮。鼠标指针变成了一个十字丝。将十字丝定位在白色的×上点击。鼠标指针变成了一个有要素的十字丝。将鼠标指针移动到绿色的×控制点, 点击。影像会移动, 两个×排在了一起。这个影像平移是个单点的转换, 它是基于栅格数据上的一个控制点和目标数据(controlpt.lyr)上相应控制点之间的连接实现的。在目录表中右键点击 Slc_9910.jpg, 选择 Zoom to Layer 查看影像的全图。



图片 2

4. 放大到影像的东南角，按照上一步描述的方法，用 Add Control Points 按钮将白色×移动到控制点 3。缩小影像会看到其它控制点看起来都已经对齐了。用前面的方法，配准影像东北和西南角的控制点 2 和 4。
5. 结束时，点击 Georeferencing 工具条上的 View Link Table 按钮。在这个过程中，ArcGIS 通过计算控制点 1、2、3、4 做一个多项式转换，应用最小二乘法使输入位置近似到输出位置。最佳多项式转换生成两个公式——一个是为输入位置 (x,y) 计算其输出的 x 坐标、一个是计算输出的 y 坐标。最小二乘法是从一个应用到所有点的一般公式得到的。



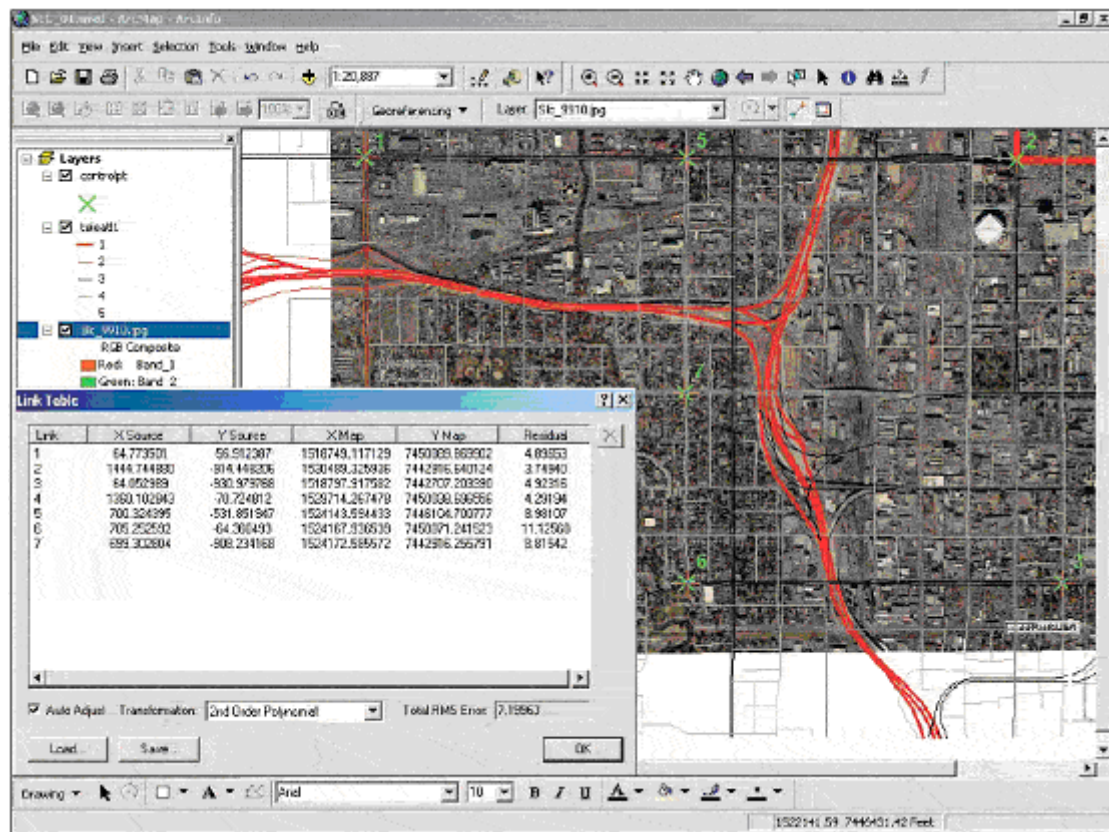
图片 3

图 3

当这个公式应用到一个控制点，会返回一个误差。误差是从点位置与到点位置之间的差异。使用更多质量等同的控制点，就能更精确地用多项式把输入数据转换到输出的坐标。因为目前我们只用了四个控制点，ArcGIS 只能执行一个 1st Order Polynomial（或仿射）变换。

转换栅格数据需要的连接数目依赖于使用的方法。但是，更多的连接不等同于更好的配准。理想情况下，连接应该分散在影像上，至少每个角上有一个点。转换的精确程度是通过比较真实的地图坐标位置与栅格中转换的位置来测量的。这个对每个连接的测量称为残留误差。点击 Georeferencing 工具条上的 Link Table 按钮来显示 Slc_9910.jpg 的误差。粗略地看，配准完成的很好。总体误差的计算是对所有连接的残留误差求均方差（RMS）。RMS 误差是对转换精度的估计值，但不一定意味着地图被精确配准。

1. 配准控制点 5、6、7 后再查看 Link Table。每个添加的点都增大 RMS 误差，但是增加穿过影像中心的点会加强配准的整体精度。
2. 点击 Link Table 按钮。在 Link Table 对话框中，选择 2nd Order Polynomial。ArcMap 能应用一个更复杂的数学运算来调整数据和校正 RMS 误差。有着不能接受的高误差点将会被选中、删除、替换。执行 2nd Order 变换需要 6 个或更多的点。



图片 4

- 保存地图文档。这样也保存了配准信息，存储在一个单独的文件里，与影像文件同名，扩展名是.aux。为 1999 年的影像创建一幅新的犹他 State Plane 影像，从 Georeferencing 工具条的下拉菜单中选择 Rectify。选择使用 Nearest Neighbor 的重采样方式，把图像以 TIFF 格式保存在 Images\UTSPN83C 目录，名称为 SLC_9910。TIFF 格式的影像比原来的 JPG 文件大。添加 TIFF 影像到地图中。它保持了位置精度，但色彩有些变动。由于图像中保存了配准信息，所以就可以把它的 world 文件传送给其它需要使用该图象的用户。

末期建设：I-15 重新投入使用

1999 年 10 月的影像配准之后，显示完成后的 I-15 工程的 2001 年 7 月的图像可以添加到地图做比较。点击 Add Data 按钮，从 Images\UTSPN83C 目录选择 Slc_0107.jpg。允许 ArcGIS 为它创建金字塔。在目录表中将它拖到 Slc_9910.jpg 的上面。Slc_0107.jpg 已经配准到犹他 State Plane 坐标，因此它能直接加载到模型中。

在 Salt Lake Valley 的西部放大和移动时，打开和关闭 2001 年 7 月的影像。打开街道图层（teleatl1）并放在影像的上面。或者，选择目录表中最上面的图层，用 Effects 工具条来设置它的透明度。

仔细观察高速公路，注意它的变化，特别是被修改的盐湖城市区的入口和 I-25/I-80 立交枢纽周围的区域。在控制点 7 的西北方出现了一个新的建筑物。注意影像之间的色彩差异。1999 年 10 月的影像，树木将要落叶了，草坪也在消退。而 2001 年，树叶茂盛、草地碧绿。

总结

这个练习演示了如何在 ArcGIS 中配准影像，在坐标空间中用矢量数据来为航空相片定向，以及保存纠正后的影像。利用 ArcGIS 对要素的一些制图显示，这幅影像就可以拿来研究，与该区域不同时间的影像做比较，从而研究空间关系在时间上的变化。

区域不同时间的影像做比较，从而研究空间关系在时间上的变化。

使用 `sderaster` 命令向 ArcSDE 中加载栅格图像

栅格数据在 ArcSDE 用户中已经获得了很高的声誉，在某种程度上这归功于 ArcGIS8 提供的创建和浏览 ArcSDE 栅格数据的，直观的、基于图形界面的工具。ArcGIS8 除了图形用户界面操作外，在 ArcSDE 的 8.2 版本中还增加了管理栅格数据的名为 `sderaster` 的命令行工具。本文主要介绍如何使用 `sderaster` 来创建和管理栅格数据。

创建栅格数据库

在单个文件中存储几百兆的栅格数据已经很不合适，要存储达到几个 G 大小的栅格数据几乎是不可能。然而，栅格数据的容量通常又是非常巨大的。这样，ArcSDE 和后台数据库的组合就是非常必要的。ArcSDE 和后台数据库管理系统(DBMS)将栅格数据物理的分割为小的文件存储，并提供良好的访问机制。与单机操作系统的文件共享机制相比，DBMS 的缓存机制提供了在高并发环境下有效存取共享数据的方法。

ArcSDE 管理员也许都已认识到，在栅格数据中即便是简单的像素的改变也需要计算机和 I/O 设备进行大量的运算操作，比如重建金字塔和重新计算栅格统计。保持数据的稳定是非常重要的，数据的频繁更改是非常不利于数据处理的，比如在文件增大的过程中，尤其是在 DBMS 事务进行锁定并恢复数据的时候这种变化是非常危险的。因此，绝大多数的栅格数据大部分时候都设定为只读模式。

ArcSDE 管理员提供两种向 ArcSDE 加载栅格数据方法：ArcCatalog 和 ArcToolbox 界面加载工具，以及 `sderaster` 命令行工具。

`Sderaster` 给管理员提供了一个快速的、灵活的加载相连影像文件(TIFF)以及多波段连续影像(BSQ)的方法。命令行工具适用于任何 ArcSDE 平台。

两种方案

本文使用 `sderaster` 命令行加载两幅相邻的美国地质调查局 (USGS) 的 O 系列数字栅格图 (DRG)，介绍了 `sderaster` 加载栅格数据的方法，并对其相关命令行操作选项进行描述。

在加载这些栅格数据文件的时候，一个可选方案是创建两个各自独立的 ArcSDE 栅格。然而这种方案只适用小容量数据输入。当加载的数据容量扩大，比如要加载的数据是一个县或者一个州的 DRG 时，这种存储方案无论对于用户访问浏览数据库，还是对于管理员管理大量的支持表和目录表，都是不可行的。因为加载的栅格数据在逻辑上就像一个地图系列一样，并且这些栅格数据共有很多相同属性（比如投影参数、像素大小、像素深度、色彩图）。另一个更佳方案就是将这两个栅格数据装载进入同一 ArcSDE 栅格中。

为了完成这个任务，ArcSDE 管理员有两个选择方案：即将所有文件存储为单一栅格镶嵌图、或是将每幅图象作为单独个体存储在栅格目录表中。以下逐一对上述两种选择方案进行说明。

创建栅格镶嵌图

DRG 文件实质上是 1:24000 的 USGS 标准分幅的原始地形图，同时包括相邻图幅上重叠的注记部分。为了进行镶嵌图制作，必须先进行影像调整，去掉地图边幅的注记说明，留下地图主体部分即可。由于地图投影旋转造成的无数据区域应该全部像素值均赋为 99。

第一步：导入第一个文件

从一个文件创建一个 ArcSDE 栅格最简单的办法是使用 sderaster 命令中的 import 操作 (sderaster -o import)。这个操作将会创建业务表 (B 表), 增加栅格列, 加载像素值。同时, 你可以选择另一种效率稍差的方法, 即使用组合的 sdetable 和 sderaster 命令。与之相关 ArcSDE 管理员命令的语法和选项, 请参阅“ArcSDE Administration Commands”。导入第一个栅格文件的语法如图 1 所示。为了能更好的阅读, 所有的例子都用前斜杠 (\) 把命令行分成了多行, 并且已经假定了 SDEUSER (-u)、SDEPASSWORD (-p)、SDESERVER (-s)、SDEINSTANCE (-i)、SDEDATABASE (-D) 这些环境变量都已设置完成。在这个 Windows 命令行中, 可以用字符 (^) 表示语句结束。

图 1 导入第一个栅格文件的语法

```
sderaster -o import \  
-f o35077e6.tif \  
-l drg_mosaic,image \  
-g \  
-k large \  
-a 99 \  
-t 128,128 \  
-c lz77 \  
-I nearest \  
-N \  
-G file=utm17_nad27.prj \  
-L 0
```

图 1 中的命令将 o35077e6.tif 的文件加载到一个命名为 drg_mosaic 的新 ArcSDE 栅格中。参数 -N 表示 sderaster 不加载源文件的色彩, -G 表示该数据的空间参考元数据, -L 表示不构建金字塔。要镶嵌第二个文件, 新建 ArcSDE 栅格必须不含色彩, 同时还必须定义空间参考。如果上述任一条件不能满足, 那么镶嵌第二幅地图的操作都会失败, 并产生 SE_MOSIC_NOT_ALLOWED 的出错信息。若是原栅格数据没有 ArcSDE 格式的投影文件, 但是它的投影又是已知的, 那么可以使用 -G 参数, 从 ArcCatalog Coordinate System 文件夹下复制一个合适的文件。另一种方法是通过投影的 ID 号来指定已经预定义好的坐标系统。投影 ID 号的列表可以在 ArcSDE C API 的 pedef.h 头文件中找到。这个文件的副本也能在 ArcSDE Developer Help 中找到 (Developer Interfaces>C API>C API Function Reference>Include Files>pedef.h)。

尽管建金字塔不会导致镶嵌的失败, 但它很浪费镶嵌的时间和进程资源。因为在建立金字塔时, ArcSDE 需要扫描整个栅格图象, 并且后台数据库一旦改变, 所建的金字塔就作废了。在做镶嵌图的过程中, 加载栅格文件的同时建立金字塔的工作量是非常巨大的, 无异于在 ArcEdit 中编辑每个要素后建立 Coverage 的拓扑关系, 或是在一张表中插入一行后重建 DBMS 的列目录, 都需要开销大量的时间。关于金字塔的其它信息将在本文的第四步中列出。

在加载含有 GeoTIFF 标记的 TIFF 文件时会产生警告。这些警告信息表明 sderaster 忽略嵌入的 GeoTIFF 标记, 并用相关的、作为影像地理参考的 world 文件和投影文件来替换。出现这样的警告信息很正常, 并不会有什么问题。

第二步: 拼接第二个文件

向已有的 ArcSDE 栅格追加栅格文件, 使用 sderaster -o mosaic 命令。因为输出文件的结构已经在输入过程中就定义好了, 如图 2 所示, 这条命令的参数较少。

图 2 sderaster -o mosaic 的参数

```
sderaster -o mosaic \  
^
```

```
-f o35077e7.tif \
-l drg_mosaic,image \
-v 1 \
-N \
-L 0 \
-a 99
```

图 2 中的命令表示把 o35077e7.tif 中的像素向 drg_mosaic 栅格添加。特别要说明的是，像素增加到了栅格 ID 号为 1 的 drg_mosaic 文件的业务表中。为了确定栅格 ID，执行 SQL 语句

```
SELECT * FROM DRG_MOSAIC
```

或者使用命令：

```
sdetable -o list \
-t drg_mosaic \
-c name \
-v 'ESRI_SDERASTERDATASET'
```

因为 -g 参数在输入时已经指明，所以在 drg_mosaic 业务表中的一个栅格名字将是 ESRI_SDERASTERDATASET。同时也要注意以下三点。

- 1、如有重叠区域，新文件的像素将覆盖原栅格中的已有像素。
- 2、-N 参数是必须要有的，以便在镶嵌过程中忽略栅格色彩。
- 3、在镶嵌操作过程中将金字塔等级设置为 0，这样确保在镶嵌操作中栅格数据不会建立金字塔。

如果此时产生 ArcSDE -303 的错误信息，即 SE_MOSAIC_NOT_ALLOWED，很有可能是错误的栅格像素配置造成的。有可能是输入栅格文件和输出文件的栅格大小不一致产生的，更有可能的是，输入栅格文件的原点并未准确偏移到输出的 ArcSDE 栅格的对应像素点上。此时，必须查看 world 文件确定加载不成功的原因。如果像素大小一致，但是原点位置并未偏移准确，那么在第二个影像的 world 文件中，应该将其手工编辑修改到正确的相应位置。这一处理过程逻辑上等于进行了一个重采样操作。如果有两个以上的栅格文件需要镶嵌，那么对于每个需要镶嵌的文件都要执行 sderaster -o mosaic 命令，唯一需要改变的是源文件的名称，用 -f 参数。

第三步：增加色彩信息

原有栅格文件的色彩可能不相同，因此 sderaster 命令不允许带色彩信息的栅格进行镶嵌，当然将 sderaster 的 -N 参数设置为忽略色彩信息便可以执行这个操作。比如说，某个栅格色彩图将像素值 1 设置为红色，而另一个将像素值 1 设置为蓝色。但是由于一个 ArcSDE 栅格数据只能拥有一个色彩信息，所以对于上述假设，只能为输入的栅格文件创建两个不同的 ArcSDE 栅格。

图 3 加载色彩信息

```
sderaster -o colormap \
-f o35077e6.tif \
-l drg_mosaic,image \
-v 1
```

用我们的方案镶嵌 DRG 文件时，每个输入文件的色彩信息都是一样的。因此，色彩信息可以从任一个输入文件中作为镶嵌 ArcSDE 栅格图的属性载入。在载入色彩信息之前，ArcSDE 栅格以灰阶显示影像。完成色彩信息加载后，它将与原来 TIFF 文件一样，已经完成了符号化。

第四步：计算金字塔、栅格统计值和 DBMS 统计值

所有需要镶嵌的文件加载完成后，就可以计算金字塔、栅格统计值和 DBMS 的统计值了。

严格来说，ArcSDE 栅格影像不一定需要计算这几个值，当然构建金字塔能够获得很好的显示效果，而且统计值对于执行某些任务操作是非常必要的。需要注意的是，ArcSDE 栅格中如果像素值改变，金字塔和统计值都必须得重新计算。对于数据量大的栅格数据，这个操作过程需要的时间有可能是几小时、甚至超过一天的时间。因此，无论是定制某些加载策略，还是制作栅格目录来取代镶嵌栅格影像，以避免这些时间开销惊人的操作，还是非常值得的。当然，这些方法各有好处，可以根据我们的实际应用需要来指定。

图 4 建立金字塔

```
sderaster -o pyramid \  
    -l drg_mosaic,image\  
    -v 1 \  
    -L -1
```

图 5 计算栅格统计值

```
sderaster -o stats \  
    -l drg_mosaic,image \  
    -v 1
```

图 6 计算 DBMS 统计值

```
sdetable -o update_dbms_stats \  
    -t drg_mosaic \  
    -m <DBMS-specific string>
```

图 4、5 和 6 分别列出了计算金字塔、栅格统计值和 DBMS 统计值所需的命令行。在图 4 中，-L -1 参数指出 ArcSDE 所需计算的最合适的金字塔等级数，这是系统基于栅格的宽度和高度计算得出的。当然，你也可以指定确切的整数值建立金字塔等级。创建统计值无需修改用户的输入，ArcSDE 会自动生成基本的统计值和图表。另一方面，计算 DBMS 级统计值需要用户提供一个说明后台数据库的字符串，以此控制后台数据库引擎对栅格表的采样。需要得到更多的关于特定 DBMS 的 -m 参数信息，请参阅 ArcSDE Developer Help。

创建栅格目录表

创建栅格目录表是将众多栅格文件载入 ArcSDE 的另一个方法。在实际应用中数据用户需要获取整张地图的目录表，包括补充的文本和地图周围的图幅符号，使用栅格目录表，管理员能够保留重叠的输入影像。通过栅格目录，用户能够查询所需的地图目录及其源文件，同时像素无需进行预先配置。

第一步：输入第一个文件

给栅格目录表加载第一个文件和创建镶嵌图的处理过程基本上是一样的，其中三个参数需要改变，-g 改成了 -n，-N 被省略，-L -1 指定 ArcSDE 构建金字塔。

图 7 为创建栅格目录表加载第一个文件

```
sderaster -o import \  
    -f o35077e6.tif \  
    -l drg_catalog,image \  
    -n o35077e6 \  
    -k catalog \  
    -t 128,128 \  
    -c lz77 \  
    -L -1
```

```
-I nearest \
-G file=utm17_nad27.prj \
-L -1
```

图 7 的语句列出在栅格目录中加载第一个文件的命令。这里使用了 `-n` 替换了 `-g`，`-n` 是为栅格创建用户定义名字的参数，这个参数可以区分同一目录下的各个栅格文件。如果使用 `-g`，栅格将被默认命名为 `ESRI_SDERASTERDATASET`。在创建栅格目录时，ArcGIS 会删除 `ESRI_SDERASTERDATASET` 的名字，并且不会把这个表作为目录存储。可以向栅格目录表随意添加它的成员所包含的附加属性。这些属性可以在制作栅格目录表的时候添加，通过构建非空间业务表的对应列并且使用 `sderaster -o` 命令进行添加，或者使用 DBMS 的工具向已有栅格目录表增加。

`-N` 参数这时应该去掉，使得 `sderaster` 能够载入源文件的色彩信息。尽管在栅格目录中每一个栅格都能存储各自的色彩信息，但是 ArcGIS 客户端使用从该栅格目录中得到的第一个栅格的色彩信息进行目录表中所有文件的符号化。

`-L -1` 表示 ArcSDE 建立金字塔作为加载的一部分。尽管每一个目录表成员的像素都存储在上一张表里，但 ArcSDE 独立追踪和管理它们。因而，改变目录中一个栅格的像素不影响其它成员的金字塔，这样，向一个栅格目录中加载新栅格数据时，无需为已有的数据重建金字塔。对于需要频繁增长变化的栅格数据库，这是个非常重要的需要考虑的事项。

第二步：向栅格目录表添加第二个文件

要向栅格目录表中增加一个新文件，需要使用 `sderaster -o` 命令。图 8 所示是添加第二个栅格的语法，所有后面田间的栅格同输入第一个栅格文件的方法几乎是一样的。

图 8 在栅格目录表中增加成员

```
sderaster -o insert \
-f o35077e7.tif \
-l drg_catalog,image \
-n o35077e7 \
-t 128,128 \
-c lz77 \
-I nearest \
-G file=utm17_nad27.prj \
-L -1
```

这里值得注意的是 `-k` 参数已经缺省。因为所有的目录表成员都已经存在同一张 DBMS 表中，`dbtune` 关键字没有也不能指定。`-G` 参数还是需要的，因为目录表中的所有成员都共有相同的投影参数，这样做是防止了加载文件时出项不一致或是漏掉的投影信息。

图 9 用 `sdetable` 和操作系统命令自动计算统计值

```
raster_id=`sdetable -o list \
-t drg_catalog \
-c name \
-v $name | \
grep IMAGE | \
awk -F: '{ print $2 }'`

sderaster -o stats \
```



```
-l drg_catalog,image \  
-v $raster_id
```

如果客户端应用程序需要栅格统计值，可以考虑在输入或添加新的栅格数据后进行计算。像色彩信息一样，统计值是独立存储在 DBMS 中的，目录表中任一文件的计算不影响栅格目录中的其它成员。当新近加载的文件都准备好了以后，可以使用 ArcSDE 命令行进行自动处理，计算栅格统计值。比如，假设使用一个循环将大量的文件载入。在循环中，变量 \$name 存储了当前加载的栅格的名字。使用的 sdetable 和操作系统的 built-in 组合命令，栅格 ID 值需要使用 sderaster -o stats 命令确定，并且存储在名为 raster_id 的变量里，如图 9 所示在加载下一个文件前马上就会用到。这一技术提供了独立于 DBMS 的确定栅格 ID 以及使用这些 ID 来计算栅格统计值的方法。如果为此使用 Windows 的批处理，那么就用“find”替代 grep 命令，用“for/f”替代 awk。

第三步：计算 DBMS 统计值

要为整个栅格目录计算 DBMS 统计值，只要在载入过程的最后执行一次 sdetable -o update_dbms_stats 命令。语法与制作镶嵌栅格图的图 6 所示语句相同。

总结：

sderaster 命令补充了 ArcCatalog 和 ArcToolbox 图形界面工具的不足。由于具有多平台支持、脚本加载的易用性和管理金字塔及栅格色彩的灵活性等等优点，命令行工具成为很多应用的不错选择。在需要将栅格文件整合制作成为无缝镶嵌图或者作为个体存储在栅格目录表的时候，sderaster 和 sdetable 命令可以用来加载 TIFF 和 BSQ 文件。

ArcIMS 4 新功能和扩展模块

对于 ArcIMS 4，ESRI 进行了较大的改进，包括新的 ArcIMS 扩展模块和新版本的 ArcIMS Site Starters。

ArcIMS 4 的一大特色是新的 ArcIMS 元数据服务器 (Metadata Server)，它可以被用来创建一个中央化、在线的元数据资料档案库，从而极大地方便了 Internet 上元数据发布和浏览。元数据由 ArcGIS 8.2 ArcCatalog 创建，存储在关系数据库中，可以通过 ArcSDE 访问。

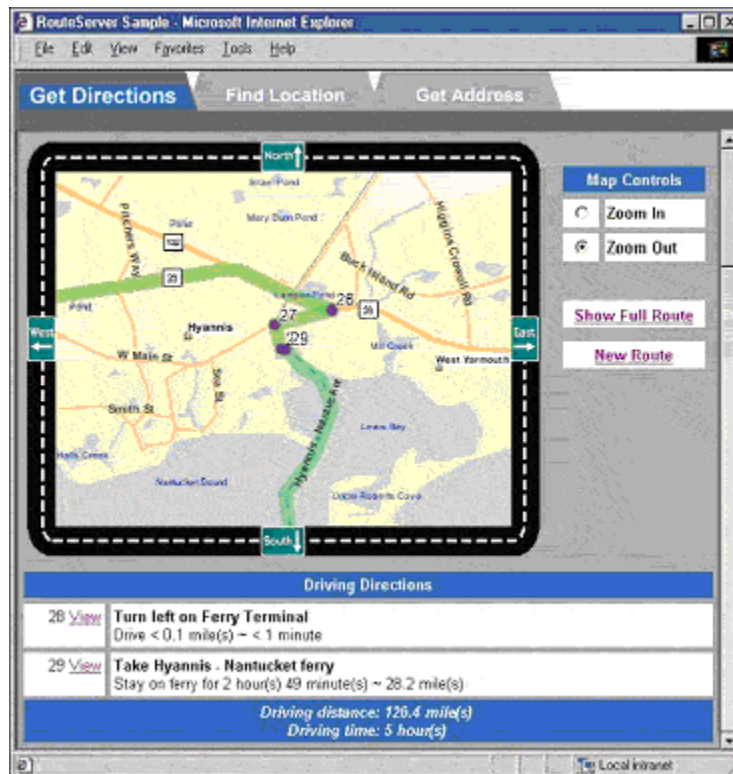
ArcIMS 4 的另一个重大改进是新的 Java 连接器。它全面替代 ArcIMS 3.1 的应用程序服务器连接。该连接器允许用户在客户端使用 Java 编程，调用服务器端的一系列 Java Beans。它包括新的对象模型，Java 服务端脚本语言 (JSP)，基于新对象模型的标签库和很多的例子。

ArcIMS 4 的用户对操作系统具有更多的选择性。新版本 ArcIMS 4 除了继续支持现有平台 IBM AIX、Microsoft Windows NT/98/2000/XP 和 Sun Solaris 外，还将支持 Linux、HP-UX 和 SGI 操作系统。

ArcIMS 支持两种新的可选扩展模块—ArcIMS Route Server 和 ArcMap Server。

ArcIMS Route Server 扩展模块

ArcIMS Route Server 扩展模块为 ArcIMS 用户在他们的 Web 站点上提供了高性能的路径分析能力。ArcIMS Route Server 不仅在 ArcIMS 站点中引入高速、准确的路程分析和点到点的行驶方向指示，而且可以基于时间和距离计算最佳路径，并支持按照多个停留来计算路径。



图片 1

ArcIMS Route Server为ArcIMS站点集成了高效的路径分析功能

ArcIMS 产品经理, Bernie Szukalski 是这样说的, “由于 Route Server 是 ArcIMS 的一个扩展模块, ArcIMS 用户将从软件特有的该功能中获益。ArcIMS Route Server 用户可以定制地图输出, 使用影像或要素流传输方式, 并执行空间和属性查询。”

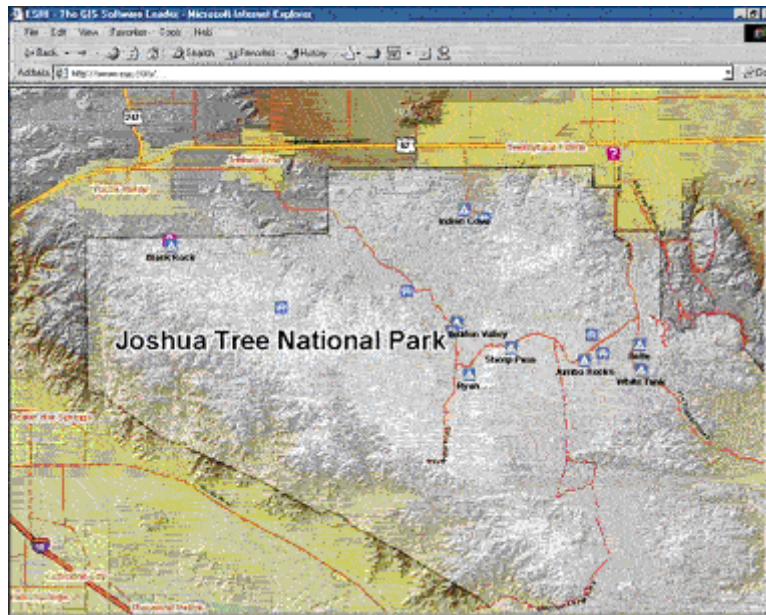
ArcIMS Route Server 的特色包含执行快速 crosscountry 路径查询, 设置路径优先 (高速路优先, 最短路径, 最快路径), 并使用精确设置以微调其结果。

ArcIMS Route Server 使用各种客户端, 从标准桌面应用程序, 基于浏览器的客户端到定制的浏览器, 桌面或无线应用程序。为了使之更快地运行, ArcIMS 开发者可以使用软件中的示例程序。或者他们可以选择 Active Server Pages (ASP), ColdFusion, Java, Java 服务端脚本 (JSP) 或者是 ArcXML, 一种 GIS 对可扩展标志语言 (XML) 的扩充, 建立他们自己的定制方案。

ArcIMS Route Server 支持使用空间数据压缩格式 (SDC) 优化过的数据集, 这样可以得到高速的地址匹配和路径选择运算。当前支持 ArcIMS Route Server 的数据提供商包括 Geographic Data Technology, Inc. (GDT) 和 Tele Atlas。数据覆盖美国的 50 个州。

ArcMap Server 扩展模块

ArcIMS 的 ArcMap Server 扩展模块允许 ArcGIS (ArcMAP 或 ArcGIS Publisher) 中编辑过的文件通过 ArcIMS 4 的服务在 Internet 上共享。通过指向 mxd 和 pmf 文件, ArcIMS 利用 ArcMap Server 支持更多的数据格式。这些额外的数据格式包括 geodatabase 要素类、要素数据集、几何网络、coverages、ArcSDE 中的基于版本的图层、DGN、DWG、DXF 以及注记。



图片 2

ArcMap Server 将 ArcMap 或 ArcGIS Publisher 生成的文件利用 ArcIMS 服务在 Internet 提供共享

只要是能够访问 ArcIMS 服务的客户端都能使用这项新技术。用户可以在任何比例尺下进行漫游并可利用 ArcMAP 的版本设置功能打印高质量的地图。

安装 ArcMAP 服务器的用户需要清楚它必须运行在与 ArcGIS 不同的机器上。

新版本的 ArcIMS Site Starter 要素缓冲功能

ArcIMS Site Starters 1.1 可以从 ESRI 的 Web 站点上下载。ArcIMS Site Starters 是一套应用程序模板，它们中的每一个都与特定的商业功能相匹配（比如数据库搜索）。Web 开发者可以对 ArcIMS Site Starter 应用程序进行简单地配置，以便使用他们自己特定的数据。

对 ArcIMS Site Starters 1.1 最重要的改进是引入 buffer 应用程序。这种新的应用程序将典型的数据库搜索和地理信息系统的空间查询功能结合在一起。用户可以定位要素，定义缓冲参数，并生成整合在地图中的缓冲报告。

ArcIMS Site Starters 1.1 同时也增强了动态地图显示的页面，它同时也支持 Internet Explorer 6.0 和 Netscape 6.2 浏览器。另外，用户在显示样式中具有一个更多的选择。

ArcIMS Site Starters 1.1 是一块里程碑，标志着 ESRI 在 ArcIMS 方面坚持不懈的目标：为 ArcIMS 用户提供简单、有效和智能的应用程序解决方案。对于普通用户而言，能够从直观地、一步一步地生成地图和报告中获益；对于开发者而言，他们能够利用可定制的应用程序框架。ArcIMS Site Starters 可能从 www.esri.com/arconline 上下载。

从 ArcView 3.x 工程导入符号化方案

从一个 ArcView 3.x 的工程（project）向 ArcGIS 8.1（ArcInfo、ArcEditor 或 ArcView）中新的地图图层导入已有的符号化方案，这会节省时间和精力。这些符号化方案来自 ArcView 3.x 工程的图例，是以 AVL 文件存储的。

1. 在目录表中，双击你要符号化的图层。
2. 在弹出的 Layer Properties（图层属性）对话框中，选择 Symbology（符号化）标签，点击 Import（导入）按钮。
3. 在 Import Symbology（导入符号化方案）对话框中，选择 Import symbology definition from an ArcView 3 legend file。
4. 点击 Folder（文件夹）按钮，浏览并选择要导入的 AVL 文件。
5. 点击 OK 关闭 Layer Properties 对话框，应用符号化方案。

在 ArcMap 中用地图提示浏览数据

ArcMap 中的地图提示（Map Tips）提供了快速浏览地图上要素的途径。地图提示生效时，移动鼠标到要素上会显示存储在字段中的任何信息，或者指定作为要素主显示字段（primary display field）的信息。当所需要的信息只是像名称这样的属性时，地图提示消除了对众多单个要素做查询的必要。

1. 在目录表中，右键点击你要显示地图提示的图层，从弹出的环境菜单中选择 Properties（属性）。
2. 在 Layer Properties（图层属性）对话框中，点击 Display（显示）标签，选中 Show map tips（显示地图提示）单选框。
3. 在 Layer Properties 对话框中，点击 Fields（字段）标签。
4. 点击 Primary display field（主显示字段）旁边的下拉列表框，选择作为地图提示显示的属性字段。
5. 点击 OK。

要使用地图提示，shapefile 或其它数据源必须建立索引。如果 Show map tips 单选框是灰色的，可以在 ArcCatalog 中为数据源建立空间索引。

1. 从 ArcMap 中删除要建立索引的数据层。
2. 打开 ArcCatalog，漫游到数据源的位置。
3. 右键点击数据源，选择 Properties（属性）。
4. 在 Properties 对话框中，选择 Index（索引）标签。点击 Add 按钮。
5. 点击 OK 完成操作。
6. 重新把图层添加到 ArcMap，如前所述使地图提示生效。

可以使用 ArcCatalog 更新和删除空间索引。在 ArcMap 中，空间索引会在添加或删除要素时自动更新。