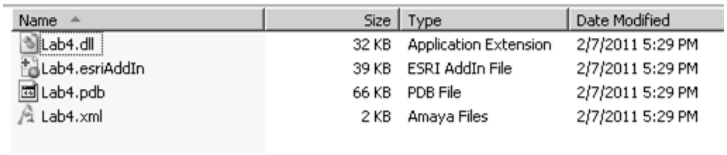
Releasing ArcObjects

```
Private Sub BtnSummary_Click(ByVal sender As Syste
    Dim pMap As IMap = My.Document.Maps.Item(0)
    Dim pFLayer As IFeatureLayer
    Dim aFeature As IFeature
    Dim pCursor As ICursor

    Try
        'Do stuff with the ArcObjects

    Catch ex As Exception
        MessageBox.Show("BtnSummary_Click Exception: " & ex.Message)
    Finally
        pMap = Nothing
        pFLayer = Nothing
        aFeature = Nothing
        pCursor = Nothing
        GC.WaitForPendingFinalizers()
        GC.Collect()
    End Try
```

ProjectName\Bin

- Debug subdirectory
  - Compiled files go here when you run the AddIn in dubug mode
- Build subdirectory
  - Compiled files go here when you build the AddIn
- These are temporary files that are recreated each time the project is built
- Okay to delete when troubleshooting

File listing



Name	Size	Type	Date Modified
Lab4.dll	32 KB	Application Extension	2/7/2011 5:29 PM
Lab4.esriAddIn	39 KB	ESRI AddIn File	2/7/2011 5:29 PM
Lab4.pdb	66 KB	PDB File	2/7/2011 5:29 PM
Lab4.xml	2 KB	Amaya Files	2/7/2011 5:29 PM

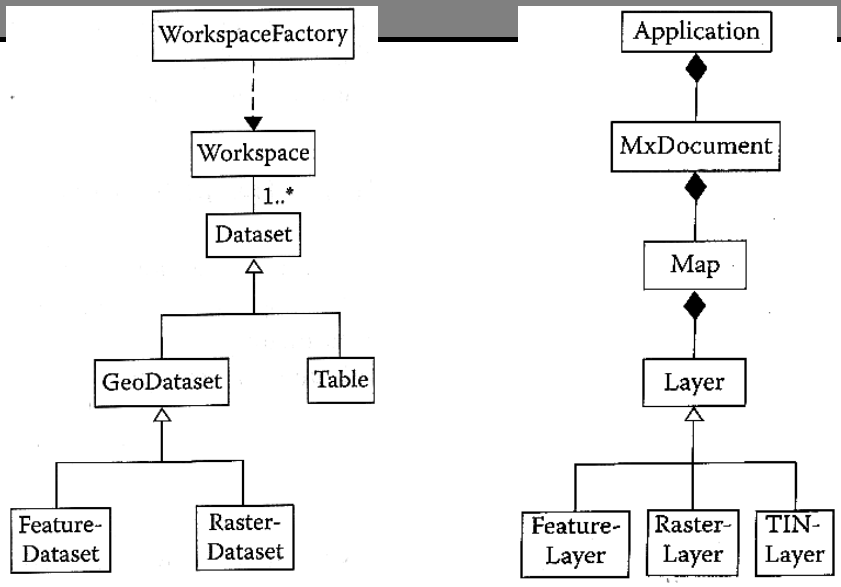
- If things get weird try
  1) Using the AddIn manager to uninstall
  2) Quitting ArcMap (all sessions)
  3) Reinstalling by double-clicking on .esriAddIn in debug or bin folder
  4) Restart ArcMap

* ArcObjects: Raster Data and Analysis
1. Raster Con Operations
2. Raster Math Operations
3. Raster Local Operations
4. Raster Neighborhood Operations

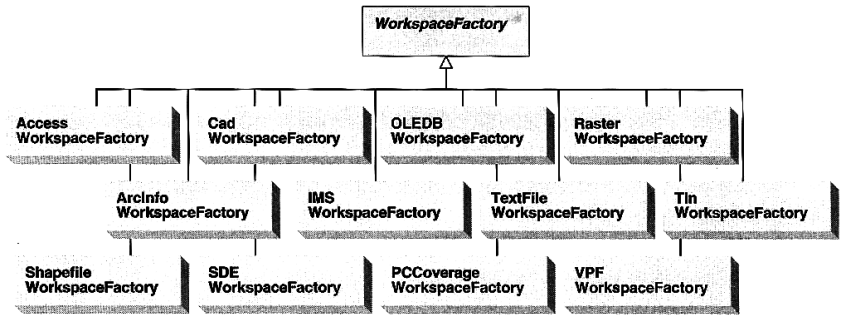
GEOG 4/590: GIS Programming

4

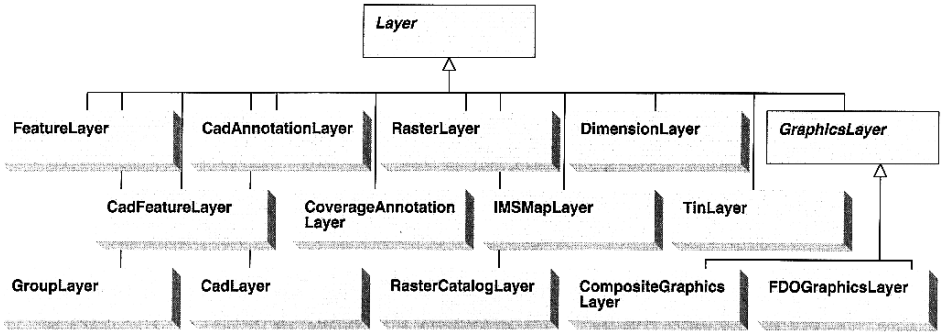
Access Raster Layer



RasterWorkspaceFactory



RasterLayer



Access Raster Layers

Direct Access

```
Dim pRLayer As IRasterLayer = New RasterLayer
pRLayer.CreateFromFilePath("C:\temp\data\covden5")
```

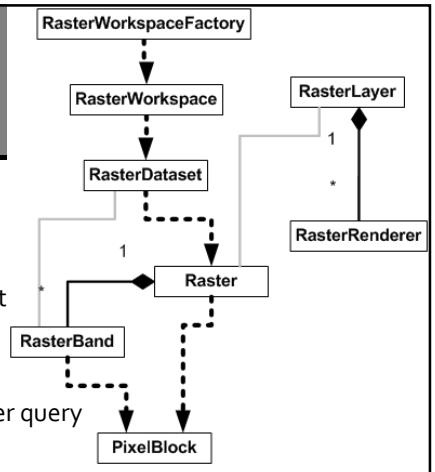
Via WorkspaceFactory

```
Dim pWkspFactory As IWorkspaceFactory = New RasterWorkspaceFactory
Dim pWksp As IRasterWorkspace = pWkspFactory.OpenFromFile("C:\temp\data", 0)
Dim pRDataset As IRasterDataset = pWksp.OpenRasterDataset("covden5")
Dim pRLayer As IRasterLayer = New RasterLayer
pRLayer.CreateFromDataset pRDataset
```

Via GxDialog

```
Dim pFilter As IGxObjectFilter = New GxFilterRasterDatasets
Dim pGxObject As IEnumGxObject
'bObjectSelected = pGxDialog.DoModalOpen(ThisDocument.Parent.hWnd, pGxObject)
Dim pGxDataset As IGxDataset = pGxObject.Next
Dim pRLayer As IRasterLayer = New RasterLayer
pRLayer.CreateFromDataset pGxDataset.Dataset
```

Raster Objects



RasterDataset: e.g., GRID, Tiff
RasterLayer: visual representation of a raster

Raster: runtime virtual copy of a RasterDataset
RasterBand: Bands in a RasterDataset
PixelBlock: A container of pixels

RasterDescriptor: The "selectionset" of a raster query

Operator Objects:

- RasterExtractionOp
- RasterReclassOp
- RasterMathOp
- RasterLocalOp
- RasterNeighborhoodOp
- RasterZonalOp

IRasterDescriptor Interface

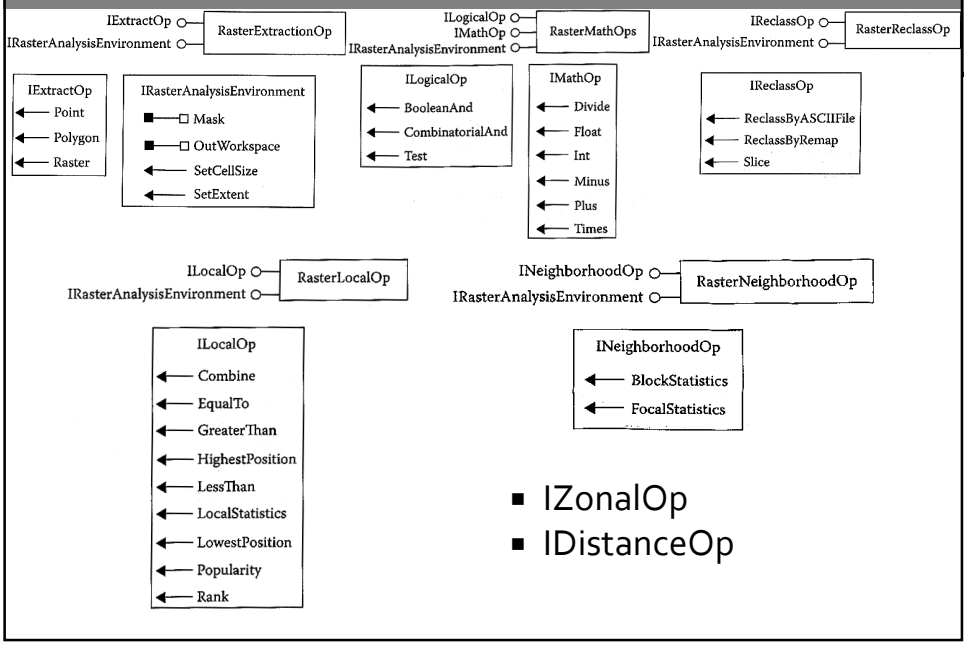
	All	Description
←	Create	Create a Raster descriptor.
←	CreateFromSelectionSet	Creates a GeoDataset descriptor with a SelectionSet.
■	Field	The field of the GeoDataset descriptor.
■	FieldName	The field name of the GeoDataset descriptor.
■	QueryFilter	The query filter of the GeoDataset descriptor.
■	Raster	The Raster in the descriptor.
■	SelectionSet	The SelectionSet of the GeoDataset descriptor.

10

IConOp.Con with RasterQuery

```
'Setup QueryFilter
Dim nValue As Integer = InputBox("Enter the a number between 1 and 5")
Dim pQFilter As IQueryFilter = New QueryFilter
pQFilter.WhereClause = "Value = " & nValue

'Create Raster and RasterDescriptor
Dim pRaster As IRaster = inRDataset.CreateDefaultRaster
Dim pRasDes As IRasterDescriptor = New RasterDescriptor
pRasDes.Create(pRaster, pQFilter, "value")

'Use RasterDescriptor
Dim pConOp As IConditionalOp = New RasterConditionalOp
Dim pOutputRaster As IGeoDataset = pConOp.Con(pRasDes, inRDataset)

Dim pOutputLayer As IRasterLayer = New RasterLayer
pOutputLayer.CreateFromRaster(pOutputRaster)
pOutputLayer.Name = "QueryOutput"

Dim pMxDoc As IMxDocument = My.ArcMap.Document
Dim pMap As IMap = pMxDoc.FocusMap
pMap.AddLayer(pOutputLayer)
pMxDoc.ActiveView.Refresh()
```

Raster Operations



- IZonalOp
- IDistanceOp

IRasterAnalysisEnvironment Interface

	All	Description
	DefaultOutputRasterPrefix	The default output raster prefix.
	DefaultOutputVectorPrefix	The default output vector prefix.
	GetCellSize	Gets the type and value of cell size in the RasterAnalysis.
	GetExtent	Gets the type and values of extent in the RasterAnalysis.
	Mask	Mask allows processing to occur only for a selected set of cells.
	OutSpatialReference	The output spatial reference of GeoAnalysis.
	OutWorkspace	The output workspace of GeoAnalysis.
	Reset	Remove all previously stored default rasteranalysis environments.
	RestoreToPreviousDefaultEnvironment	Restores to the previous default raster analysis environment.
	SetAsNewDefaultEnvironment	Sets the raster analysis environment of the object as new default environment.
	SetCellSize	Sets the type and value of cell size in the RasterAnalysis.
	SetExtent	Sets the type and values of extent in the RasterAnalysis.
	VerifyType	The verify type of the RasterAnalysis.

Save a Raster Dataset – ISaveAs Interface

```
Public Shared Sub SaveRasterDataset( _
    ByRef geoDataset As IGeoDataset, ByVal rasterPath As String, _
    ByVal rasterName As String)

    Dim pWSF As IWorkspaceFactory = New RasterWorkspaceFactory
    Dim pWS As IRasterWorkspace = pWSF.OpenFromFile(rasterPath, 0)

    Dim pRasterSaveAs As ISaveAs2 = geoDataset 'casting
    pRasterSaveAs.SaveAs(rasterName, pWS, "GRID")

    pRasterSaveAs = Nothing
    pWS = Nothing
    pWSF = Nothing
End Sub
```

14

IMathOp Example - SquareRoot

```
' Create the RasterMathOps object
Dim mathOp As IMathOp = New RasterMathOps

' Create the output raster object
Dim geoDataset_output As IGeoDataset = mathOp.SquareRoot(geoDataset)
```

15

FocalStatistics Example

```
'Dim geoDataset As IGeoDataset
'nbrWd As Integer,  nbrHt As Integer
'filterType is an esriGeoAnalysisStatisticsEnum value

'Specify a rectangle neighborhood
Dim pRasterNeighborhood As IRasterNeighborhood = New RasterNeighborhood
pRasterNeighborhood.SetRectangle(nbrWd, nbrHt, esriUnitsCells)

'Create a RasterNeighborhoodOp operator
Dim pNeighborhoodOp As INeighborhoodOp = New RasterNeighborhoodOp

'Perform focal statistics
Dim pOutRaster As IGeoDataset = New RasterDataset

For index As Integer = 1 To iterations
  pOutRaster = pNeighborhoodOp.FocalStatistics( _
    geoDataset, filterType, pRasterNeighborhood, True)  'set ignore nodata to true
  geoDataset = pOutRaster
Next
```

FocalStatistics - esriGeoAnalysisStatisticsEnum

esriGeoAnalysisStatisticsEnum Constants

Statistics types in GeoAnalyst.

Constant	Value	Description
esriGeoAnalysisStatsMajority	1	Majority.
esriGeoAnalysisStatsMaximum	2	Maximum.
esriGeoAnalysisStatsMean	3	Mean.
esriGeoAnalysisStatsMedian	4	Median.
esriGeoAnalysisStatsMinimum	5	Minimum.
esriGeoAnalysisStatsMinority	6	Minority.
esriGeoAnalysisStatsRange	7	Range.
esriGeoAnalysisStatsStd	8	Standard Deviation.
esriGeoAnalysisStatsSum	9	Sum.
esriGeoAnalysisStatsVariety	10	Variety.

ILocalOp Example - Combine

```
'geoDataset_1, geoDataset_2 , geoDataset_3 are of IGeoDataset
'Get RasterBand objects from RasterBandCollection Interface of a Rasterdataset
Dim rasterBC_1 As IRasterBandCollection = CType(geoDataset_1, IRasterBandCollection)
Dim rasterBC_2 As IRasterBandCollection = CType(geoDataset_2, IRasterBandCollection)
Dim rasterBC_3 As IRasterBandCollection = CType(geoDataset_3, IRasterBandCollection)
Dim rasterBand_1 As IRasterBand = rasterBC_1.Item(0)
Dim rasterBand_2 As IRasterBand = rasterBC_2.Item(0)
Dim rasterBand_3 As IRasterBand = rasterBC_3.Item(0)

' Create a RasterBandCollection
Dim rasterBandCollection As IRasterBandCollection = New RasterClass
rasterBandCollection.Add(rasterBand_1, 0)
rasterBandCollection.Add(rasterBand_2, 1)
rasterBandCollection.Add(rasterBand_3, 2)

Dim pInputDataset As IGeoDataset = CType(rasterBandCollection, IGeoDataset)
Dim localOp As ILocalOp = New RasterLocalOpClass

'setup raster analysis environment parameters
Dim pRAE As IRasterAnalysisEnvironment = localOp
With pRAE
   .SetCellSize(esriRasterEnvSettingEnum.esriRasterEnvValue, 60)
   .SetExtent(esriRasterEnvSettingEnum.esriRasterEnvMinOf)
End With

' Declare the output raster object
Dim geoDataset_output As IGeoDataset = localOp.Combine(pInputDataset)
```

18