Introduction to Spatial Analysis and Spatial Modeling
Benefits of raster GIS

- ArcGIS raster support
- The ArcGIS Spatial Analysis extension
  - The seven interfaces
  - Installation and licensing
- Exercise 1
  - Introduction to the spatial analyst interfaces
Benefits of raster GIS

- **Location-based data model (cells)**
  - Better than features for many types of analyses

- **Especially sued for**
  - Surface creation and analysis
    - Elevation, rainfall, population
  - Location models
    - Best site for business, hospital
  - Distance measurement
    - Proximity, mobility, best path
  - Modeling movement
    - Flood inundation, fire spread
ArcGIS raster support

- **ArcGIS has out-of-the-box raster support**
  - Use with ArcMap
    - Draw, query, georeference
  - Manage with ArcCatalog
    - Copy, rename, delete,
  - Manipulate with ArcToolbox
    - Convert, project, merge, clip,
  - Store with a geodatabase
    - Raster datasets and catalogs
- **Add Spatial Analyst for analysis**
Spatial Analysis tools

- Over 100 tools organized into toolsets

[Image of toolset interface]

Seven interfaces for Spatial Analyst

- 1. ArcToolbox
  - Dialogs for all tools
- 2. Command Line
  - Type commands
- 3. Model Builder
  - Visual modeling
- 4. Scripts
  - Write easy programs
- 5. Spatial Analyst Toolbar
  - Dialogs for common tools
- 6. ArcObjects
  - More programming power
- 7. Map Algebra Tools
  - For all interfaces

- Most become Map Algebra
- Evaluated by Raster Engine

*These are all part of the geoprocessing framework*
The Spatial Analyst toolbar

Has its own environment (not part of the geoprocessing)

Can compose Map Algebra expressions

Dialogs for commonly used tools
Spatial Analyst and ArcToolbox

- Uses geoprocessing environments (right-click to set)
- Has Map Algebra tools

Notes:
- Opens ArcToolbox
- Hints and links to help
Spatial Analyst and Command Line

- Opens the Command line
- Command Interface for tools
- Supports code completion

Supports code completion
Spatial Analyst and Model Builder

- Click tools to set parameters
- Set model and diagram properties
- Save and run model
- Edit and change
- Uses geoprocessing environment

- Add a new toolbox
- Add a new model
Spatial Analyst and scripts

- Many languages are supported (Python shown)
- Is object-oriented (But easy!)
- Uses geoprocessor tools, environments
- May add scripts to ArcToolbox
- May use scripts in models
- Has a Map Algebra tool

```python
# Spatial Analyst and scripts example

# Import system modules
import sys, string, os, win32com.client

# Create the Geoprocessor object
gp = win32com.client.Dispatch("esriGeoprocessing.IGpDispatch.1")

# Check out any necessary licenses
gp.CheckOutExtension("spatial")

# Load required toolboxes...
gp.AddToolbox("C:\Program Files\ArcGIS\ArcToolbox\Toolboxes\Spatial Analyst Tools.tbx")

# Local variables...
cstopath = "c:\stop"  # Stop file path
Output_polyline_features = "c:\rice\Contour_stopol.shp"
Output_raster_2 = "c:\rice\HillShap_topol"

# Process: Aspect...
gp.Aspect_sa(cstopath, Output_raster)

# Process: Contour...
gp.Contour_sa(cstopath, Output_polyline_features, ",", "0", "1")

# Process: Hillshade...
gp.HillShade_sa(cstopath, Output_raster_2, "315", "45", "NO_SHADERS", "1")
```
Spatial Analyst and ArcObjects

- Many languages are supported (VBA shown)
- Is object-oriented (Powerful, but a lot of classes to learn)
- Must set your environments
- Build stand-alone applications, or build tools, or ...?
- Has classes for Map Algebra

```python
ctopobath = "ctopobath"
Output_raster = "D:/rice/Aspect ctopol"
Output_polyline_features = "D:/rice/Contour_ctopobal.shp"
Output_raster_2 = "D:/rice/HillSha_ctop1"

' Process: Aspect...
gp.Aspect_sa ctopobath, Output_raster

' Process: Contour...
gp.Contour_sa ctopobath, Output_polyline_features, ",", "0", "1"

' Process: Hillshade...
gp.HillShade_sa ctopobath, Output_raster_2, "315", "45", "NO_SHADOWS", "1"
```

Supports code completion
Spatial Analyst and Map Algebra

- An analysis language for raster data
  - Uses math-like expressions with operators and functions:
    
    \[
    \text{SmoothHill} = \text{Hillshade}(\text{FocalMean}([\text{Elevation}] \times 0.3048))
    \]
  
  - Can be more efficient than Spatial Analyst tools (one expression may use many functions and operators)

Spatial Analyst tools and Map Algebra

- Most tools implement Map Algebra functions and operators
  - Tools provide dialog and command line interfaces
  - Most tools implement a single function or operator (some implement many functions)
- Tools exist for writing Map Algebra expressions
Managing the extension

Start > Programs > ArcGIS > Desktop Administrator
- Select software product
- Select license manager
- Check license availability

Tools > Extensions
- Enable or disable extensions
Resources for self study

- **Online Help**
  - For ArcGIS Desktop users and for developers

- **Documentation**

- **ESRI Support Center**
  - Software information
  - Knowledge Base
  - Downloads
  - User forums
  - Developer support and tools

- **Virtual Campus courses**
  - Self-learning modules
Exercise 1 overview

- Check the license in the Desktop Administrator
- Enable the extension in ArcMap
- Run a tool with the Spatial Analyst toolbar
- Run a tool with the ArcToolbox
- Run a tool with the Command Line window
- Build and run a model with the Model Builder
- Create and run a script and view its code
- Run and view a VBA program that uses ArcObjects
- Write and run a Map Algebra expression
- Install several ArcObjects-based utilities (you will use them in future exercises)