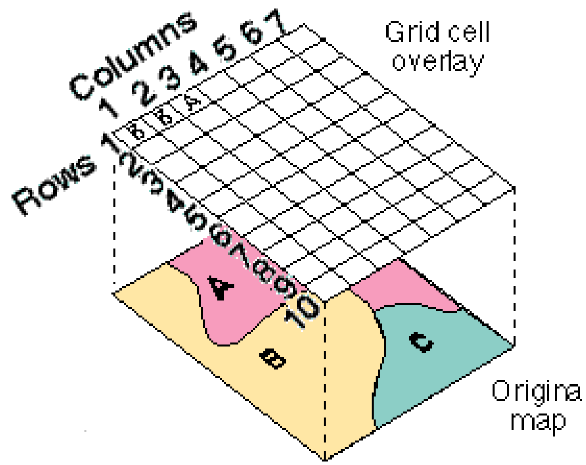
Raster Representations and Calculations

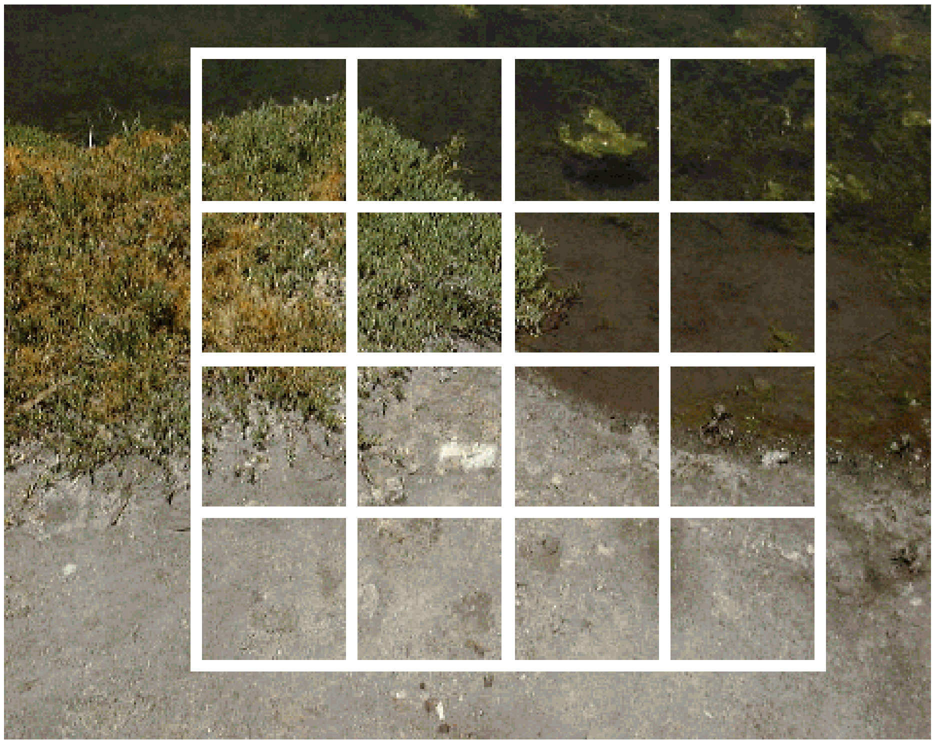
- The **raster concept**:
  - A **2-D array** of attributes
    - Each represented by **mathematical values**
  - **Locations** on the cells on the ground are **implicitly encoded** based on their **row-column** positions

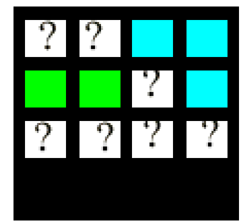


Raster data file

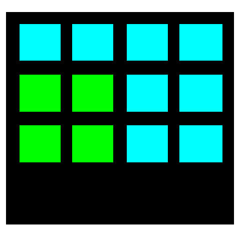
ROW	COLUMN	ATTRIBUTE
1	1	B
1	2	B
1	3	A

Spatially Straightfoward, But What About Value Encoding?

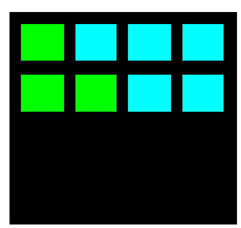




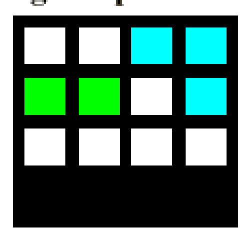
Water/Veg dominates



Winner takes all



Edges separate



Coding Strategies for Cell Values

- We are building a **model of reality** here:
  - We can make model **design decisions** based on what the **intended application** is
- All models **selectively throw information away**:
  - Whether presence/absence, or ordinal / interval / ratio categories or counts, any particular approach will be **selectively useful**

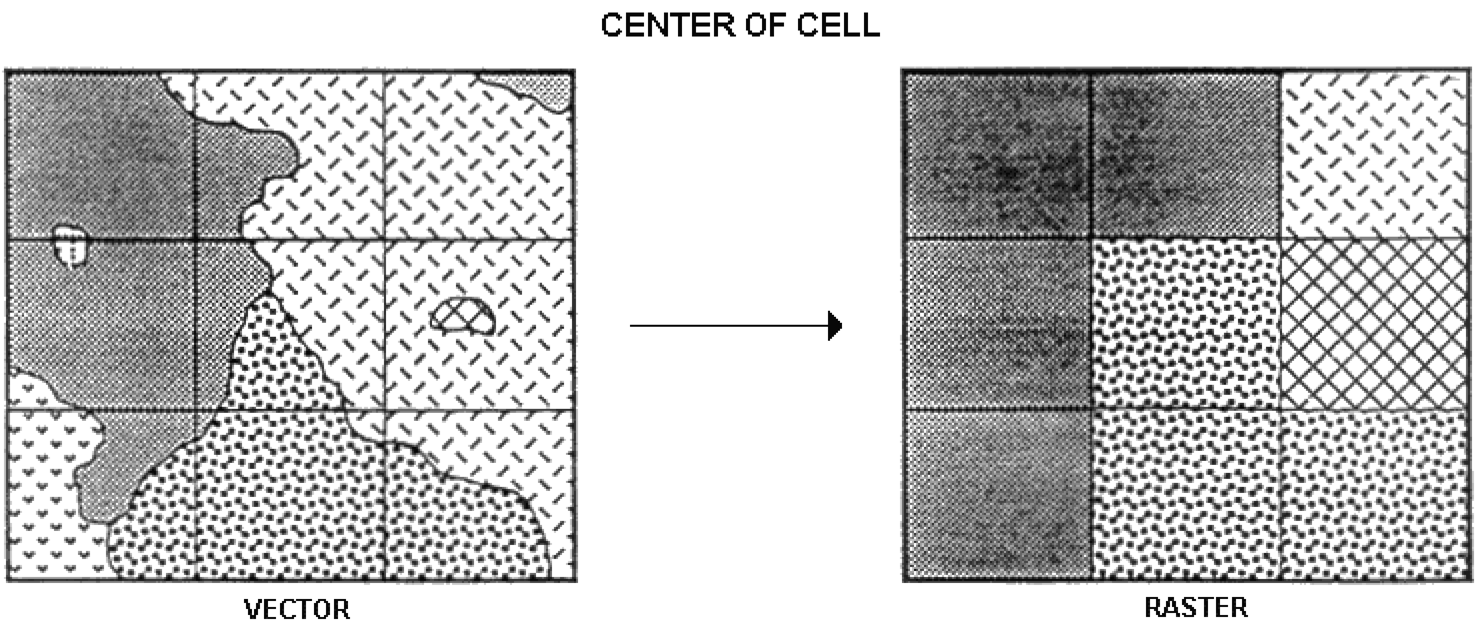
Systematic coding strategies for determining cell values:

1. Use the value from the cell center (centroid)
2. Use majority weighting within each cell
3. Calculate weighted values

(+ **non-systematic** most important type)

Systematic Coding Strategies for Cell Values

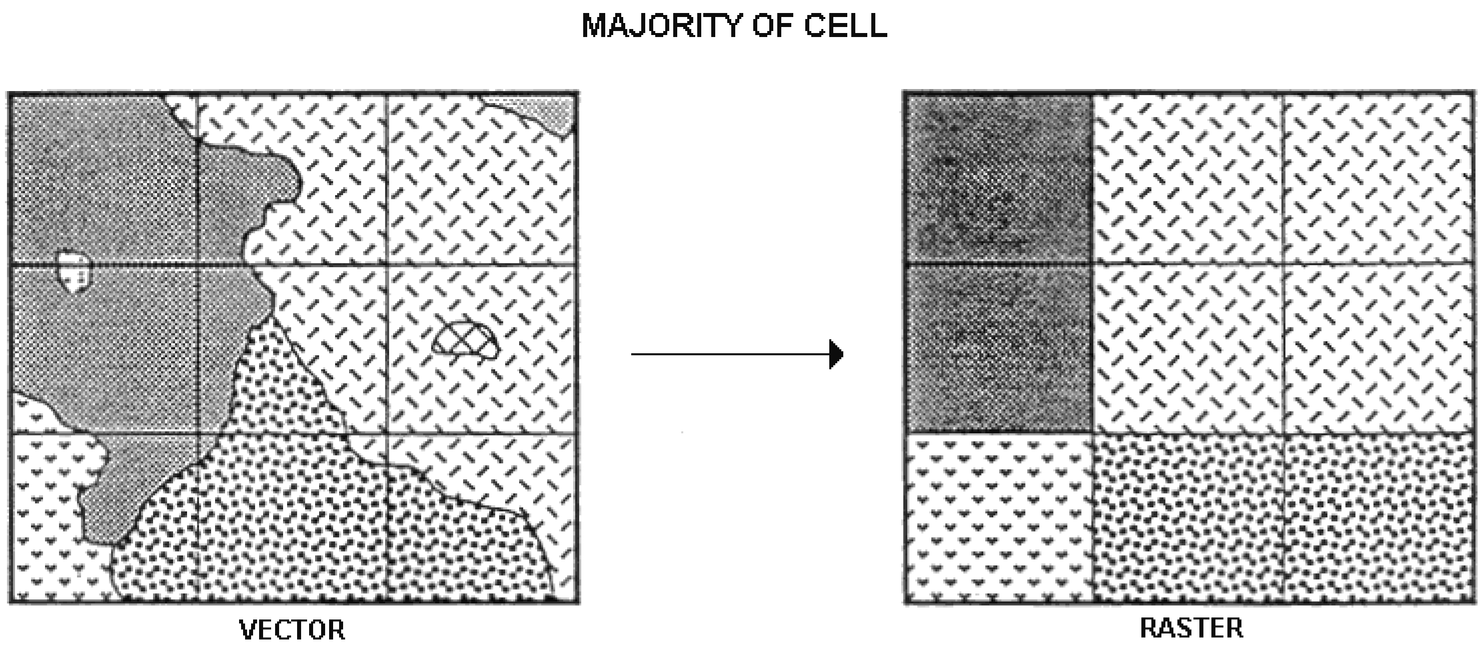
1. Use the value from the cell center (centroid)



- The value at the **centroid** is assigned to the cell
- This is a simple approach, but it can **over-represent** the values from small areas

Systematic Coding Strategies for Cell Values

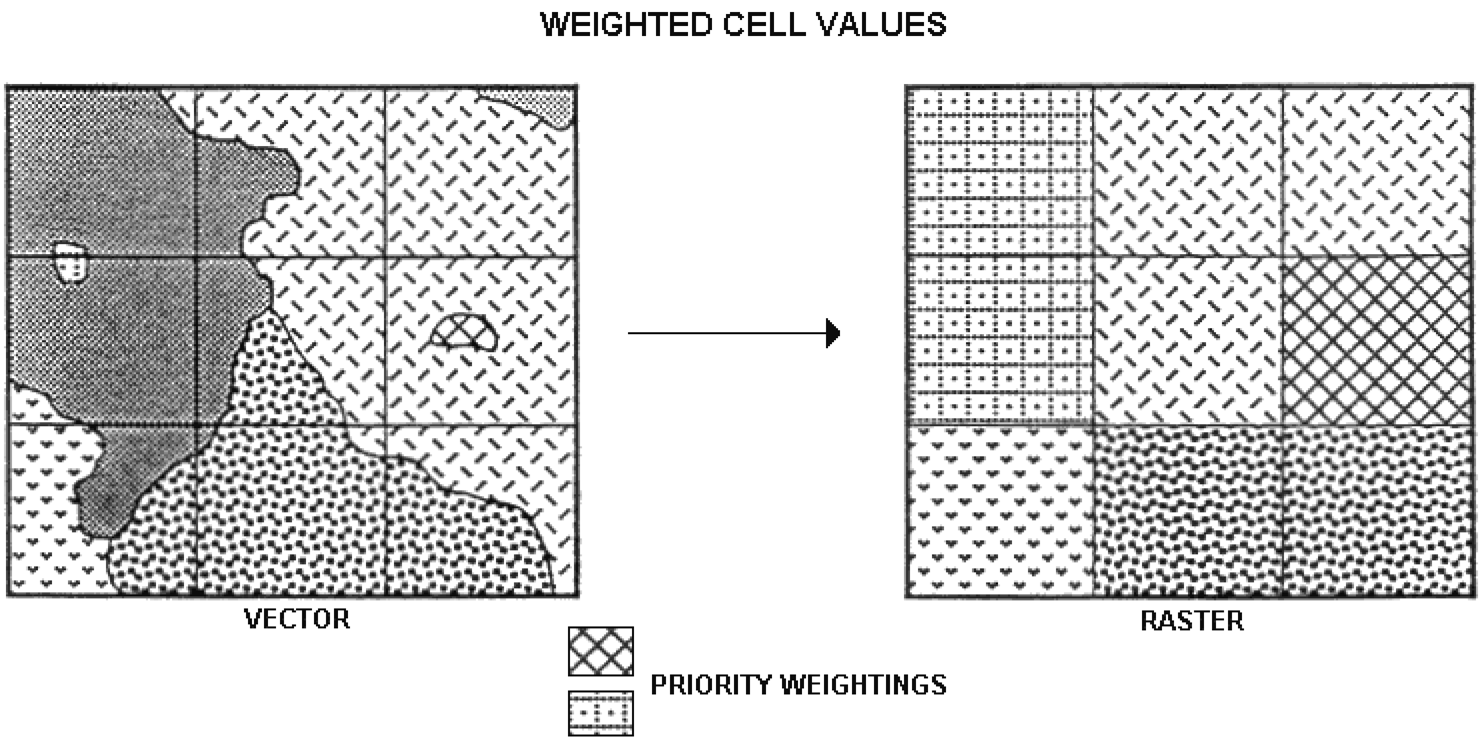
2. Use majority weighting within each cell



- The value covering the **majority** of the area is assigned to a cell
- This is a **“fairer”** representation than cell centers

Systematic Coding Strategies for Cell Values

3. Calculate weighted values



- Priority weights are based upon the **importance** of different values
  - The **"most important"** value present is assigned to a cell
- This ensures the representation of **crucial** geographic phenomena

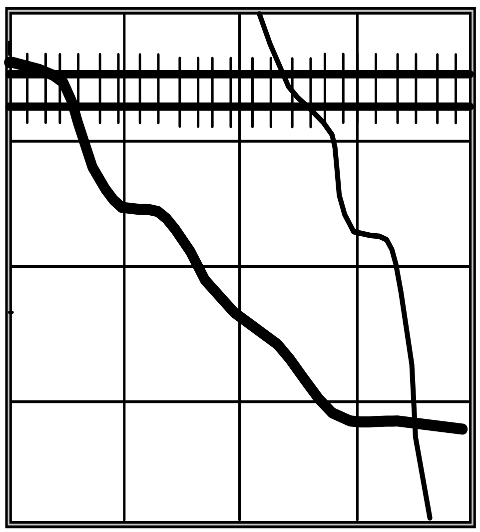
Dealing With Crossing Linear Objects

- What happens when **more than one linear object** occurs in a single cell?
  - If each theme were **separate**, there is **no problem**
    - i.e. roads = one theme, rail = another
    - Use **presence/absence** coding
  - Otherwise…
    - Use **most important type** method
      - Requires you to **decide** which is **most important**

Crossing Linear Features

Most Important Type Method

Transportation Matrix



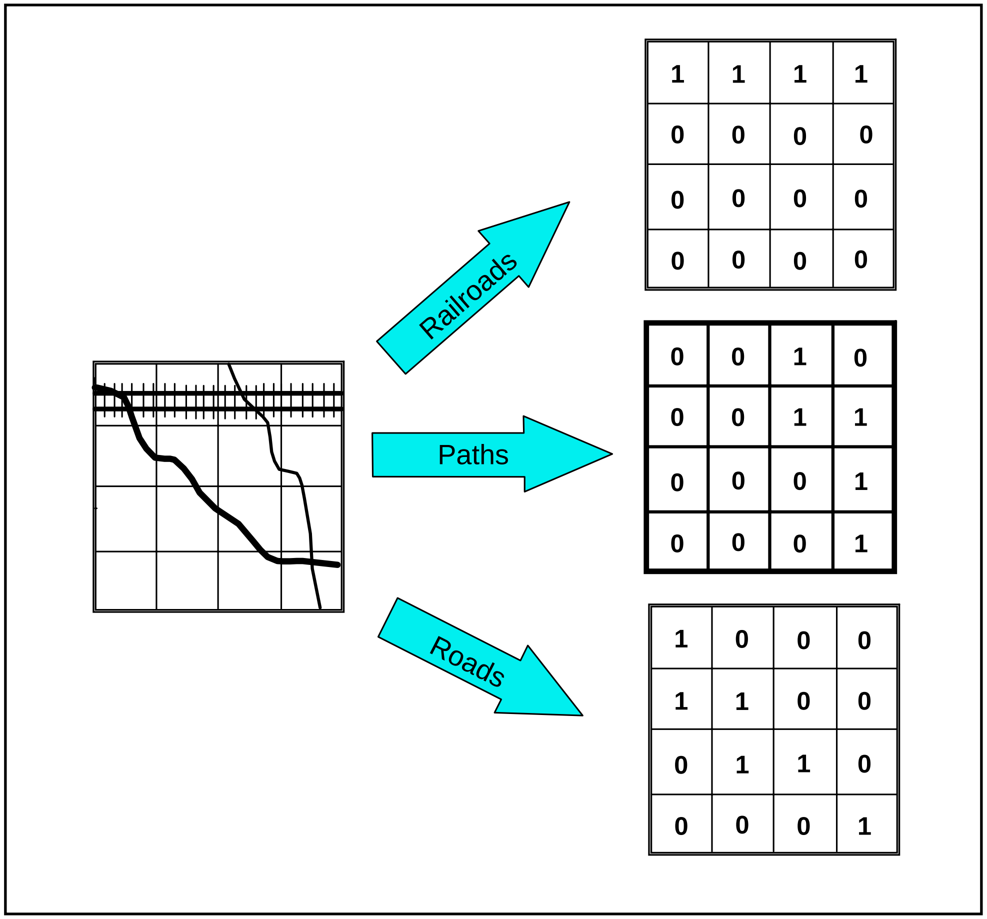
=

Data Matrix

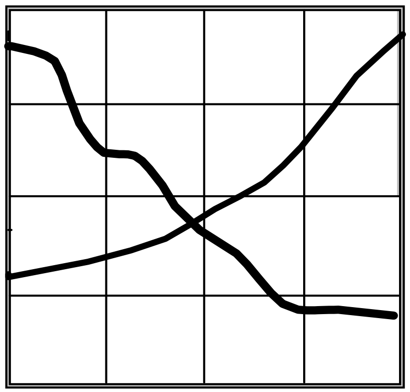
1	2	2	2
1	1	3	3
	1	1	3
		1	1

☐ = No Data

Separating The Objects into Individual Themes



Using Extended Raster Model with Crossing Linear Objects



Note how **category 3** can be used to represent when **roads cross**

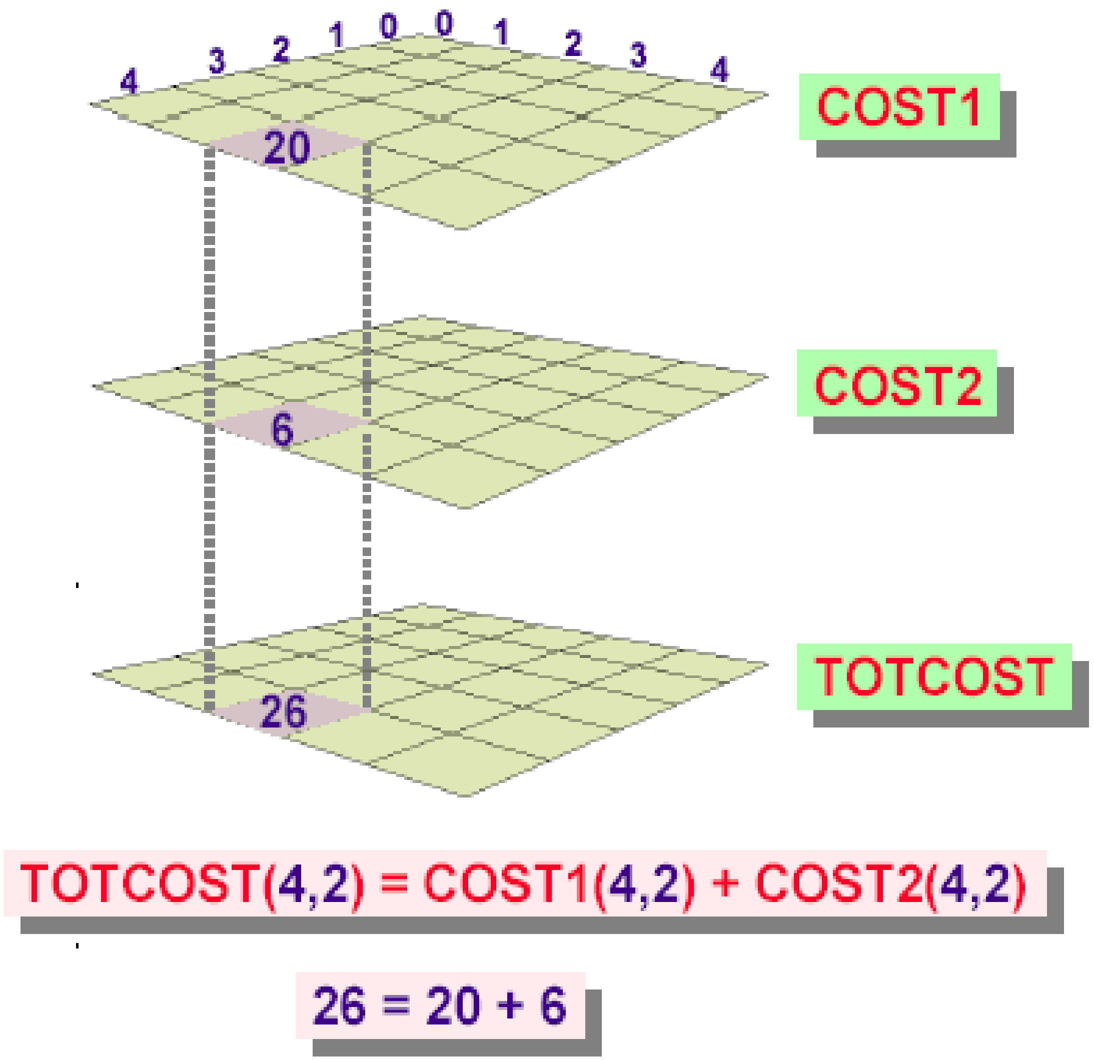
1	0	0	2
1	1	2	2
2	3	3	0
0	0	1	1

Value	Count	Description	Type	Construction
1	5	Highway	4 Lane	3/20/97
2	4	Road	2 Lane	9/17/99
3	2	Hwy/Road	N/A	N/A

Note also how **additional attributes** can be recorded with this method

Cell Coincidence

- The **raster concept**:
  - Each grid cell location for each theme **explicitly coincides** with its other thematic counterparts
  - The **efficiency** of raster GIS modeling **depends** on this



Matrix Algebra

$$\begin{pmatrix} 5 & 4 & 1 \\ 2 & 1 & 2 \\ 4 & 2 & 1 \end{pmatrix} + \begin{pmatrix} 3 & 2 & 1 \\ 1 & 4 & 5 \\ 2 & 7 & 3 \end{pmatrix} = \begin{pmatrix} 8 & 6 & 2 \\ 3 & 5 & 7 \\ 6 & 9 & 4 \end{pmatrix}$$

Map Algebra

5	4	1
2	1	2
4	2	1

+

3	2	1
1	4	5
2	7	3

=

8	6	2
3	5	7
6	9	4

Arithmetic operations: the same for -, but not *, /, mod

Matrix Algebra

$$\begin{bmatrix} 5 & 4 & 1 \\ 2 & 1 & 2 \\ 4 & 2 & 1 \end{bmatrix} * \begin{bmatrix} 3 & 2 & 1 \\ 1 & 4 & 5 \\ 2 & 7 & 3 \end{bmatrix} = \begin{bmatrix} 21 & 33 & 28 \\ ? & ? & ? \end{bmatrix}$$

Map Algebra

5	4	1
2	1	2
4	2	1

*

3	2	1
1	4	5
2	7	3

=

15	8	1
2	4	10
8	14	3

Arithmetic operations: the different for *, /, mod

Introduction to Map Algebra

- Language components
- Syntax and rules
- Objects
- Operators
- Commands

Language Components

- A **data manipulation** language for raster
  - **Math-like** expressions
    - AgSuit = (SoilSuit * 0.75) + (SlpSuit * 0.25)
- **Parts** of the language
  - **Objects**: Raster, numbers, constants, and so on
  - **Operators**: "+", "/", "GT", "LE", "AND", "OR", and so on
  - **Functions**: Slope, FocalMean, Sin, and so on
  - **Rules**: For building expressions and using functions
- Most operators & functions implemented as **tools**

Map Algebra operators

Arithmetic

+	Addition
−	Subtraction
*	Multiplication
/, DIV	Division
MOD	Modulus
−	Unary minus

Boolean

^, NOT	Logical complement
&, AND	Logical And
|, OR	Logical Or
!, XOR	Logical Xor

Combinatorial

CAND	Combinatorial And
COR	Combinatorial Or
CXOR	Combinatorial Xor

Relational

==, EQ	Equal
^=, <>, NE	Not equal
<, LT	Less than
<=, LE	Less than or equal
>, GT	Greater than
>=, GE	Greater than or equal

Logical

DIFF	Logical difference
IN {list}	Contained in list
OVER	Replace

These work with **two objects**, like: Slope GE 10

Relational Operators in Map Algebra

- **Relational Operators** (<,>,==,>=, <=)

A

1	0	3	5
6	9	3	1
0	2	7	0
2	8	5	1

>=

B

3	7	8	1
5	9	4	0
2	3	7	8
7	2	7	0

=

C

0	0	0	1
1	1	0	1
0	0	1	0
0	1	0	1

(A >= B) = C

Boolean Operators in Map Algebra

- The AND operation requires that the value of cells in **both** input layers be **equal to 1** for the output to have a value of 1:

0	1	1
0	0	1
1	0	1

AND

0	0	0
1	1	1
0	0	1

=

- The OR operation requires that the value of a cells in **either** input layer be **equal to 1** for the output to have a value of 1:

0	1	1
0	0	1
1	0	1

OR

0	0	0
1	1	1
0	0	1

=

Arithmetic Operators in Map Algebra

- We can **extend** this concept from Boolean logic to **algebra**
- Map algebra:
  - Treats input layers as **numeric inputs** to mathematical operations (each layer is a separate numeric input)
  - The result of the operation on the inputs is calculated on a **cell-by-cell basis**
- This allows for **complex overlay analyses** that can use as many input layers and operations as necessary
- A common application of this approach is **suitability analysis** where multiple input layers determine suitable sites for a desired purpose by **scoring cells** in the input layers according to their effect on suitability and combining them, often **weighting layers** based on their importance

Simple Arithmetic Operations

Summation

0	1	1
0	0	1
1	0	1

+

0	0	0
1	1	1
0	0	1

=

0	1	1
1	1	2
1	0	2

Multiplication

0	1	1
0	0	1
1	0	1

×

0	0	0
1	1	1
0	0	1

=

0	0	0
0	0	1
0	0	1

Summation of more than two layers

0	1	1
0	0	1
1	0	1

+

0	0	0
1	1	1
0	0	1

+

0	0	0
1	1	1
0	0	1

=

0	1	1
2	2	3
1	0	3

Raster (Image) Difference

The difference between two layers

5	4	3
6	5	6
7	1	5

\-

3	5	6
1	4	5
3	2	7

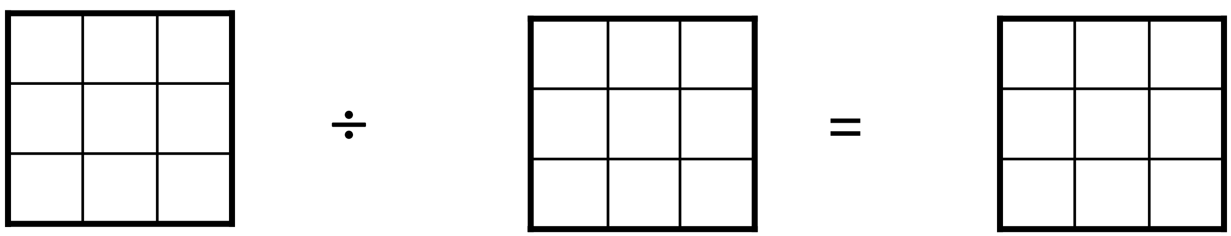
=

2	-1	-3
5	1	1
4	-1	-2

- An application of taking the differences between layers is **change detection**:
    - Suppose we have **two raster layers** that each show a map of the **same phenomenon** at a particular location, and each was generated at a **different point in time**
    - By taking the **difference** between the layers, we can **detect changes** in that phenomenon over that interval of time
- Question: **How** can the locations where changes have occurred be identified using the difference layer?

Raster (Image) Division

Question: **Can we** perform the following operation? Are there any **circumstances** where we **cannot** perform this operation? Why or why not?



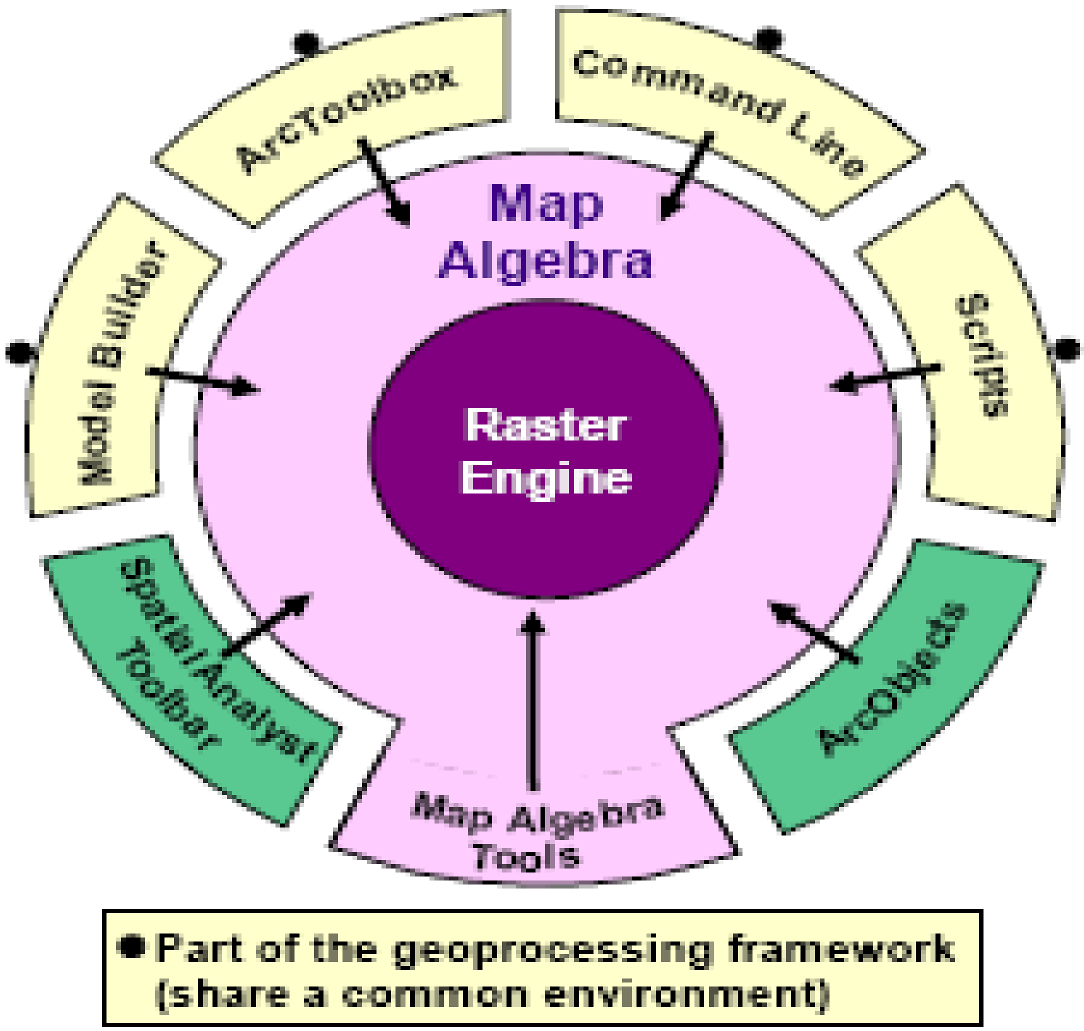
More Complex Operations

Linear Transformation

$$a \begin{bmatrix} 1 & 2 & 4 \\ 3 & 2 & 1 \\ 5 & 3 & 2 \end{bmatrix} + b \begin{bmatrix} 1 & 0 & 0 \\ 5 & 1 & 1 \\ 2 & 0 & 1 \end{bmatrix} + c \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} & & \\ & & \\ & & \end{bmatrix}$$

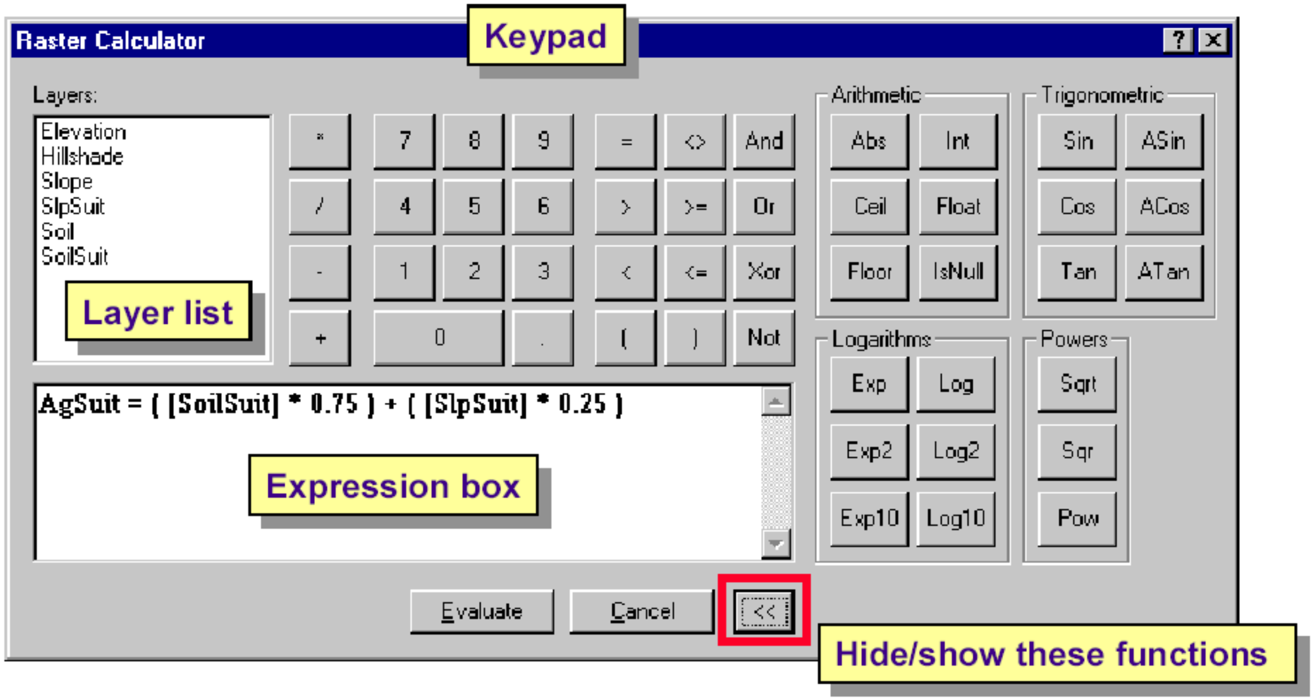
- We can multiply layers by **constants** (such as a, b, and c in the example above) before summation
- This could applied in the context of computing the results of a **regression model** (e.g. output y = a*x1 + b*x2 + c*x3) using raster layers
- Another application is **suitability analysis**, where individual **input layers** might be **various criteria**, and the **constants** a, b, and c determine the **weights** associated with those criteria

Seven Interfaces for Spatial Analyst



The Raster Calculator

- Use to enter **map algebra expressions**:
  - Build with **buttons or type** into expression box



Expression Syntax Rules

- **Delimit** operators and objects with **blanks**:

```
Wrong:  Layer+Layer2+Layer3
Right:  Layer1 + Layer2 + Layer3
```

- Operators evaluated by **precedence** level:

```
Layer1 + Layer2 * Layer3
```

- **Override** operator precedence with **parentheses**:

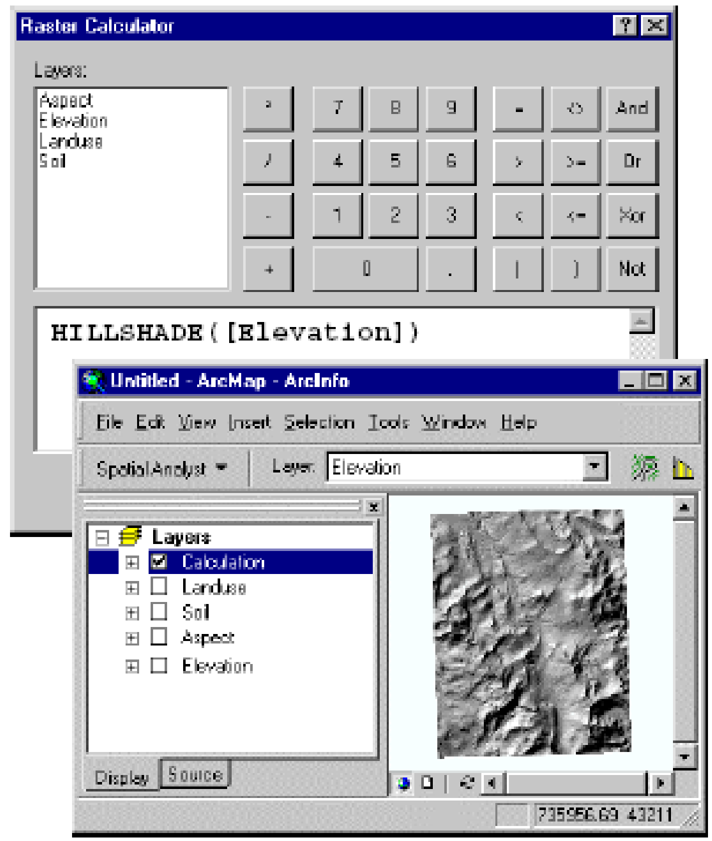
```
(Layer1 + Layer2) * Layer3
```

- **Nested** parenthetical expressions **evaluate first**:

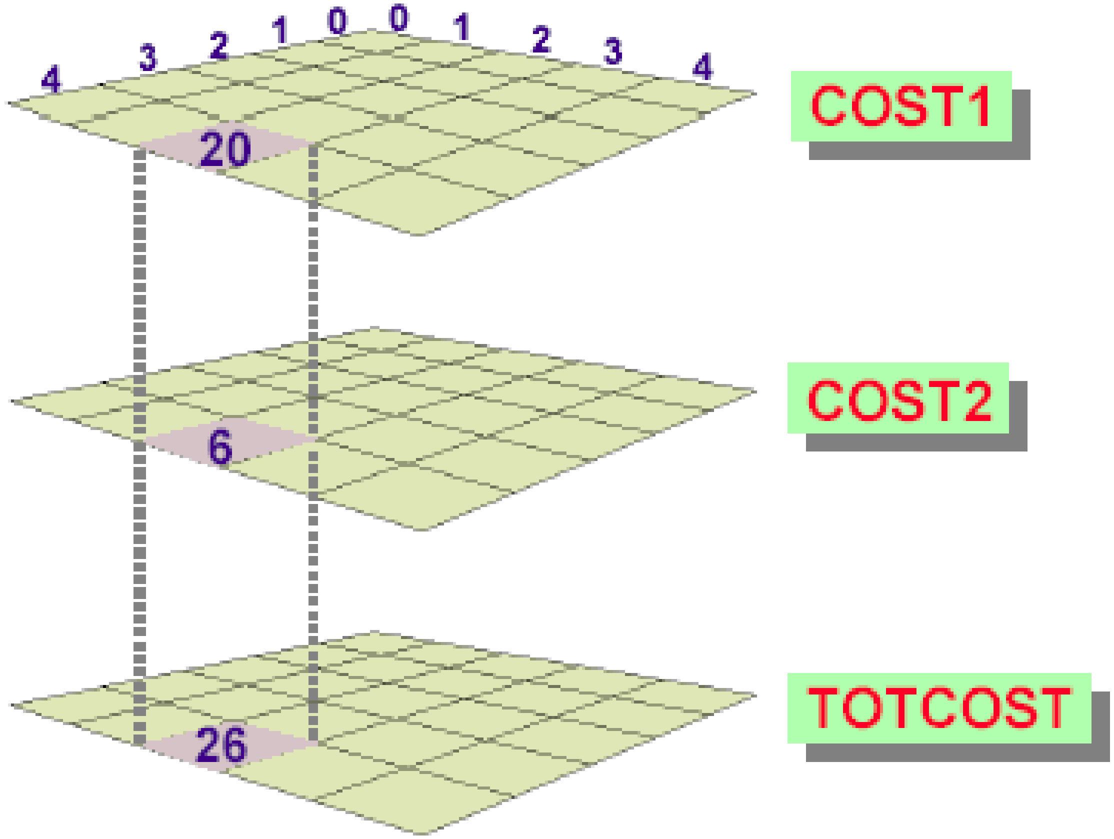
```
((Layer1 + Layer2) / 4 ) - Layer3
```

Expression Results

- Expressions **return** grids, vector data, tables, etc.
  - **Depends on functions** used
  - **Most** return **GRIDs**
- Temporary or permanent?
  - For **returned grids** only
  - **Temporary** GRID if unnamed
  - **Permanent** GRID if named
- Layers **added** to ArcMap:
  - Table of Contents



Cell coincidence

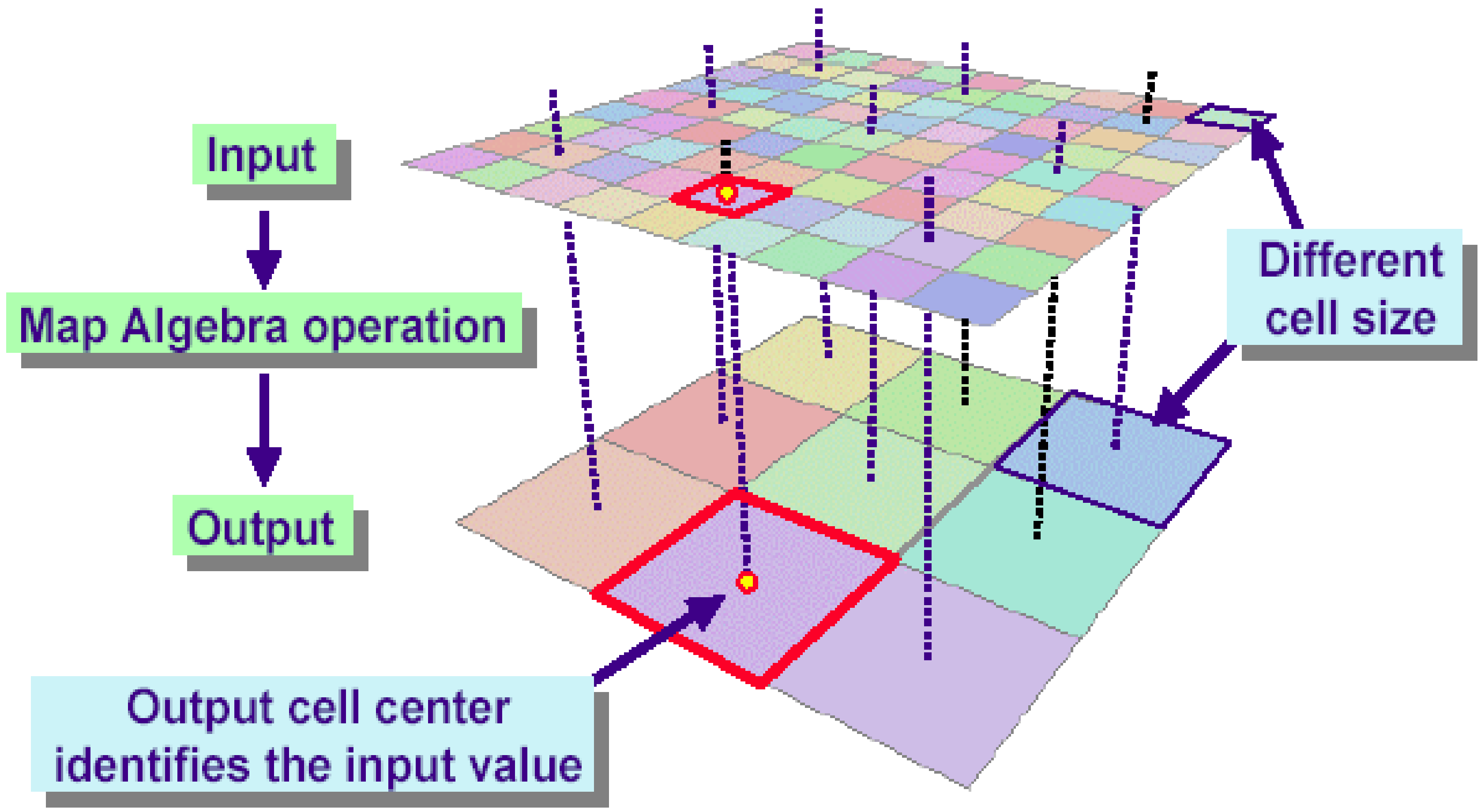


TOTCOST(4,2) = COST1(4,2) + COST2(4,2)

26 = 20 + 6

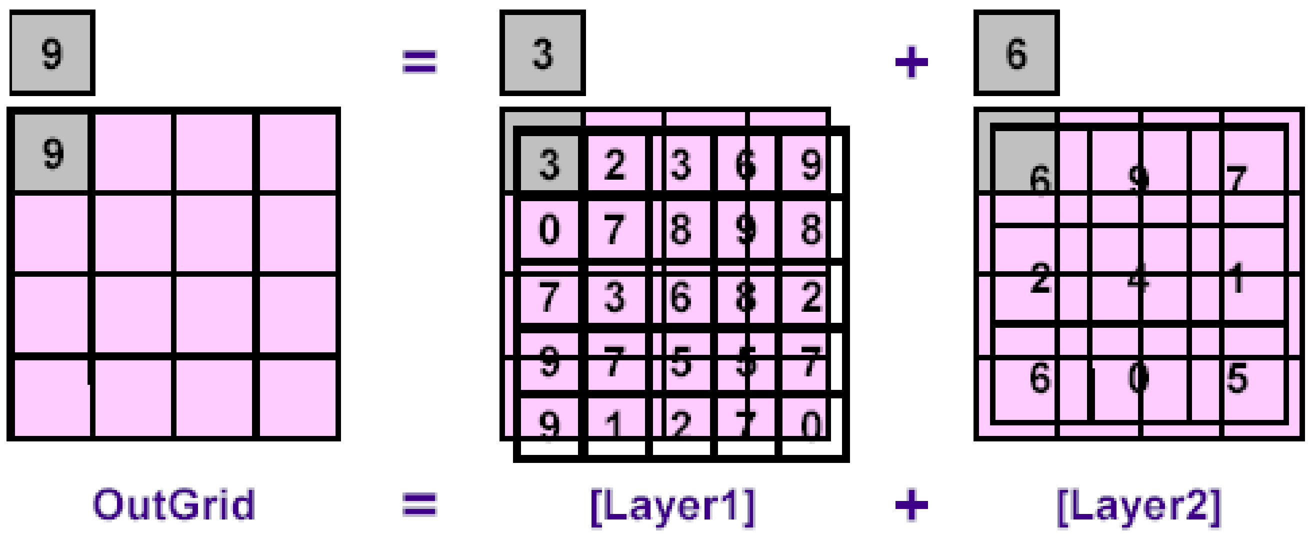
Resampling

- Automatically applied when combining rasters



Expression Evaluation (when Resampling)

- Expressions are processed as follows:



1. **Define** empty output GRID based on the analysis environment.
2. **Position** to the **next output cell** (start at row 0, column 0).
3. **Resample input raster(s)** to determine corresponding cell values.
4. **Evaluate** the expression and **write the result** to the output cell.
5. **Repeat** steps 2 - 4 for **all output cells**.

User attributes in expressions

- You may use numeric VAT fields in expressions
- Reference with [Layer].*field* notation

Vegetation.VAT

Value	Count	Desc	Suit
101	2450	Grass	1
201	65780	Mixed	3
301	32187	Pine	2
401	5433	Oak	5

Soil.VAT

Value	Count	Desc	Suit
23	2450	Sand	2
46	65780	Loam	1
87	32187	Clay	6
99	5433	Rock	9

```
[Vegetation].Suit + [Soil].Suit
```

  - [Layer] alone is assumed to be [Layer].Value
- You may join tables to grids VAT file
  - Use joined fields for symbology, selection
  - Cannot use in Map Algebra expressions

Special cell values in Map Algebra

- Logical: Non-zero values are *True*, zero is *False*

-2	-2	5	5	7
-2	-2	5	5	7
5	5	5	7	7
0	0	4	7	7
0	4	4	4	7

As Values

T	T	T	T	T
T	T	T	T	T
T	T	T	T	T
F	F	T	T	T
F	T	T	T	T

As True/False

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
0	0	1	1	1
0	1	1	1	1

As logical 0,1

- NoData: If any input is *NoData*, the output is *NoData*

1	1	1	1	2
1	ND	1	2	2
1	1	2	2	2
1	2	2	2	2
2	2	2	2	2

[Layer1]

+

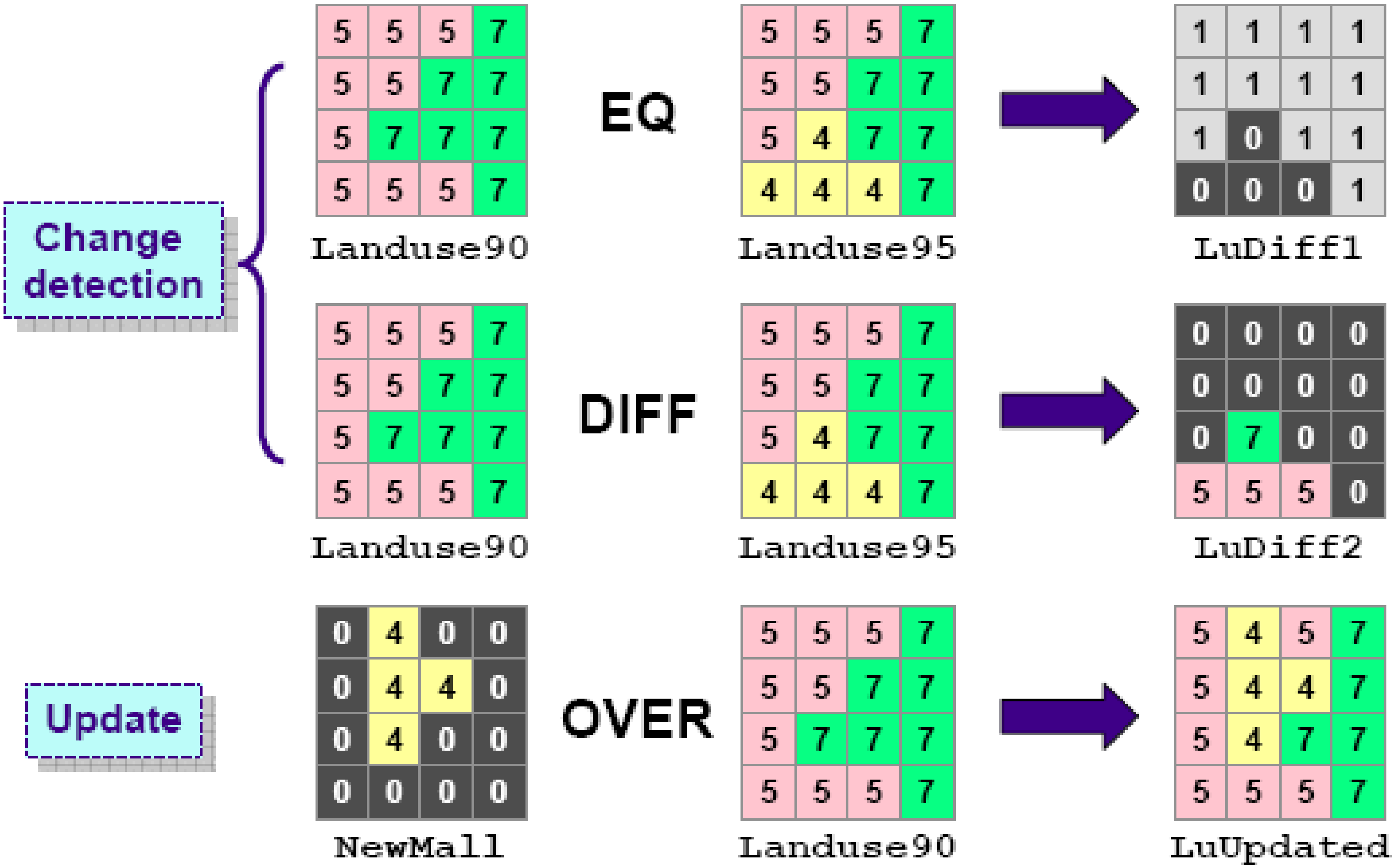
2	2	2	2	2
2	2	2	2	2
2	2	1	1	1
2	2	1	ND	1
2	2	1	1	1

[Layer2]

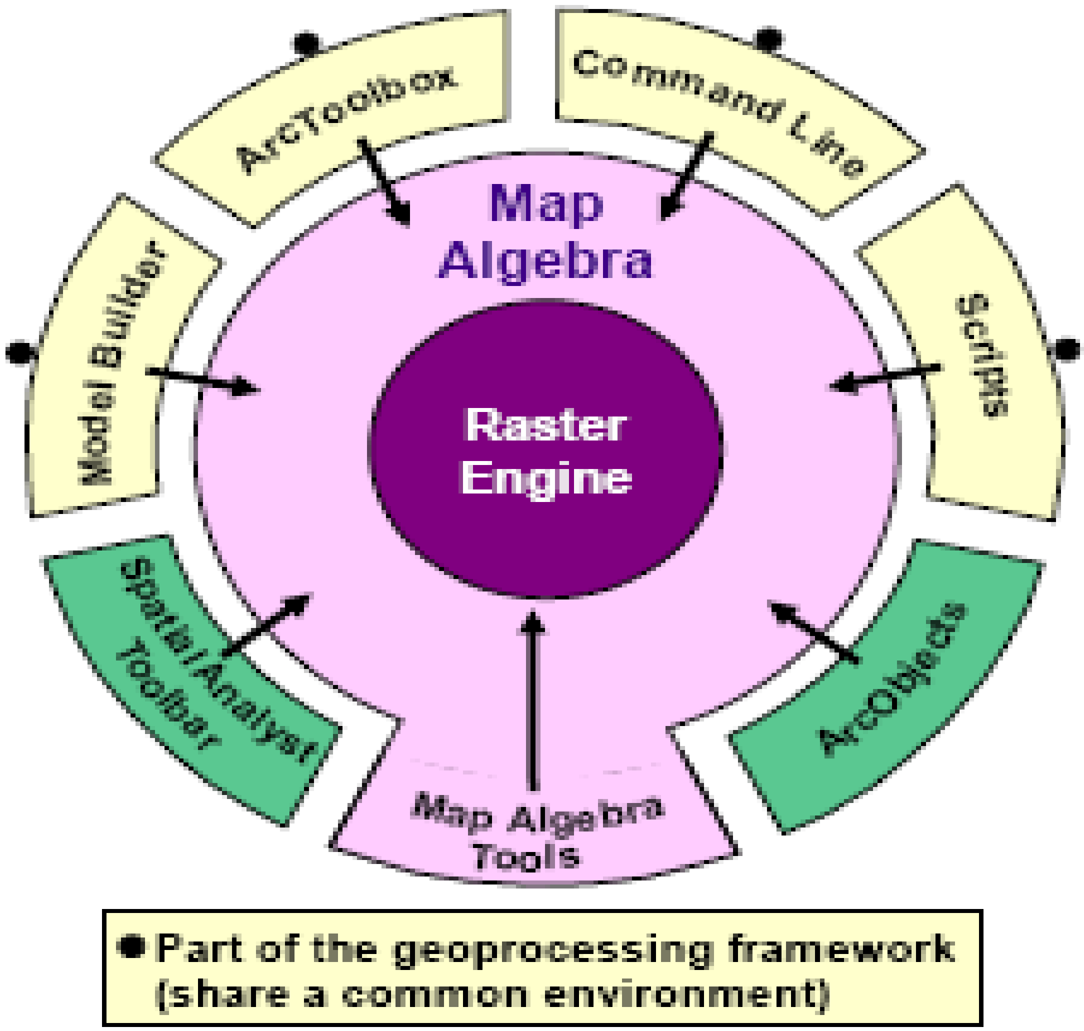
3	3	3	3	4
3	ND	3	4	4
3	3	3	3	3
3	4	3	ND	3
4	4	3	3	3

[Layer3]

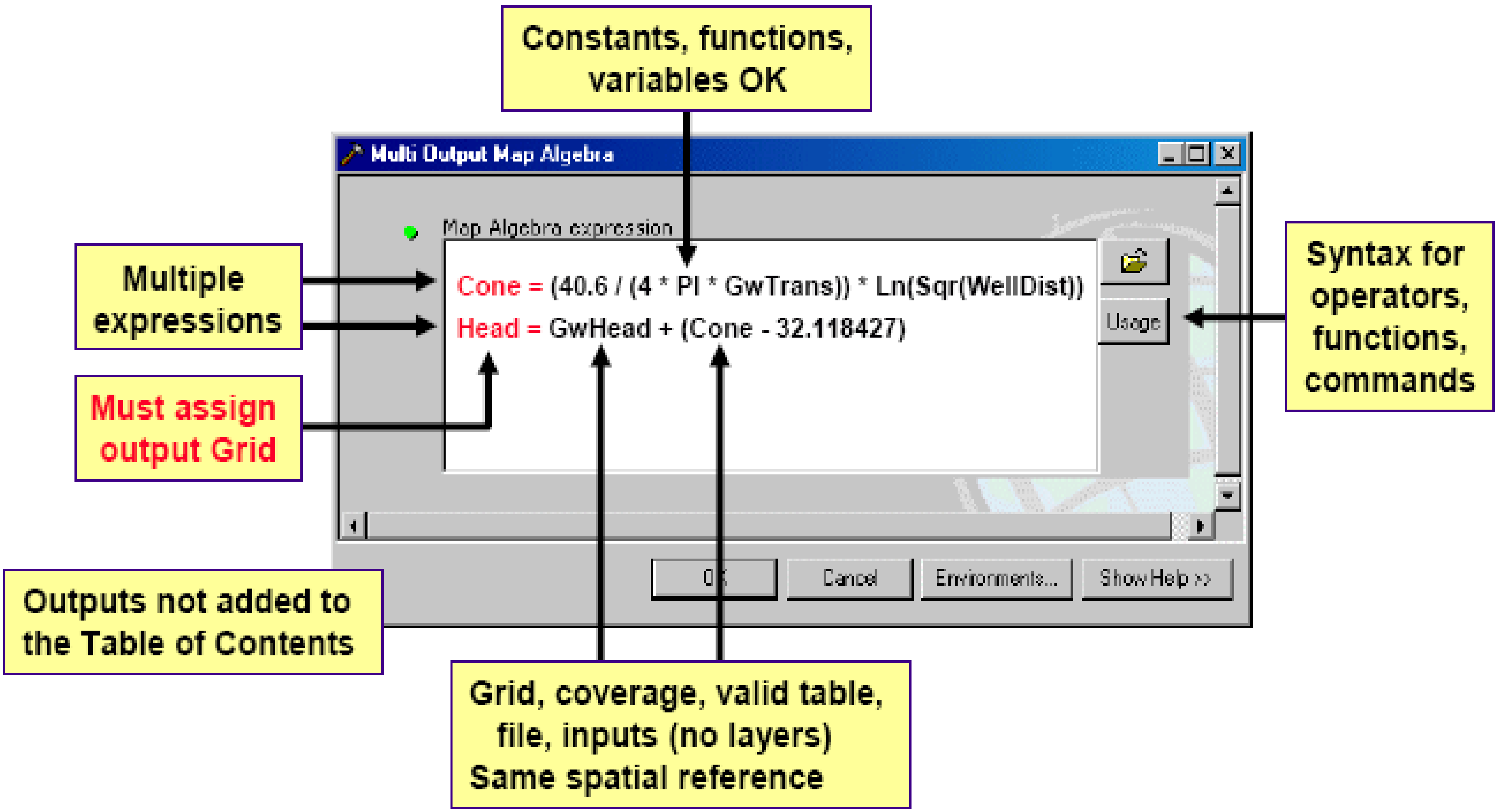
Examples of Operators



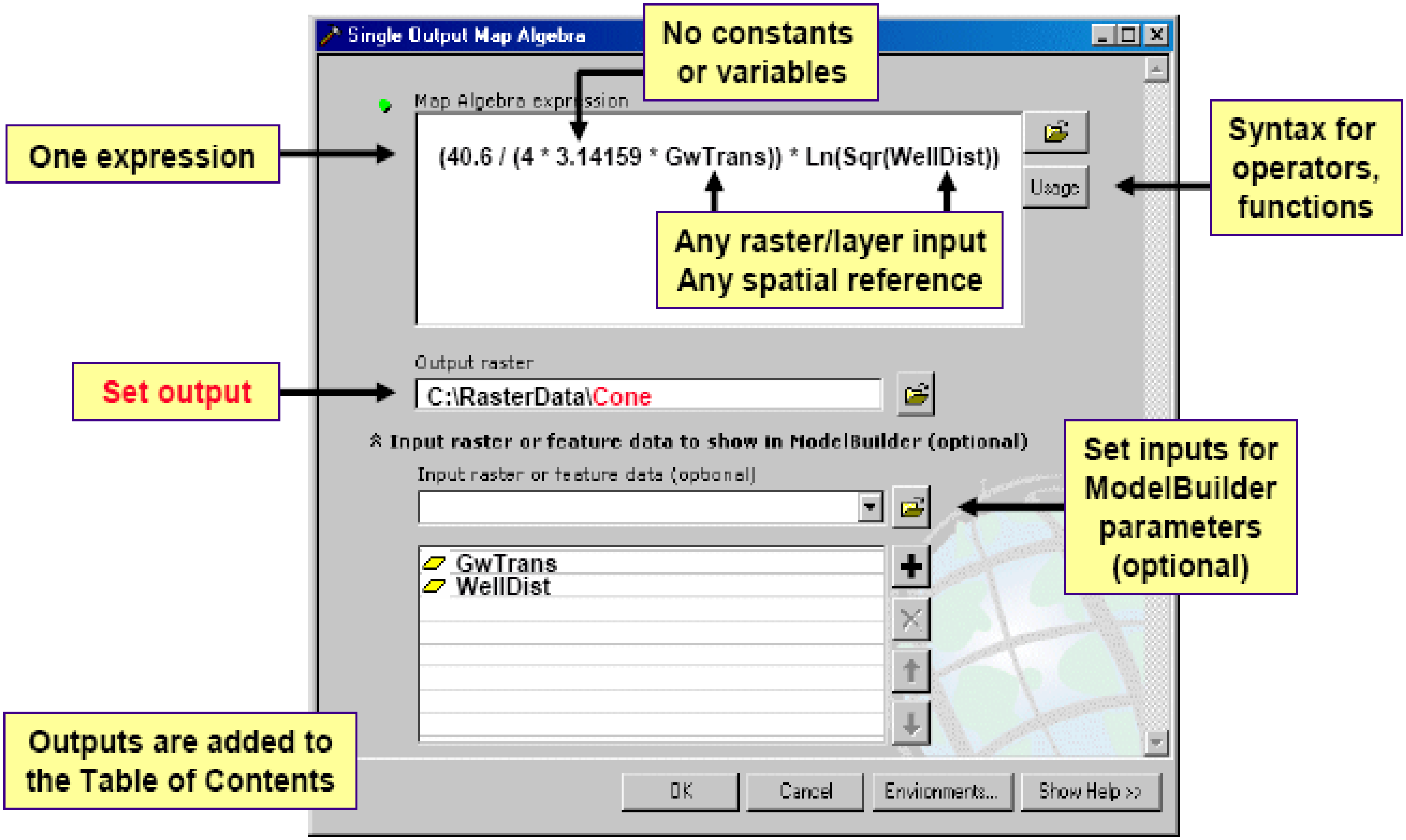
Seven Interfaces for Spatial Analyst



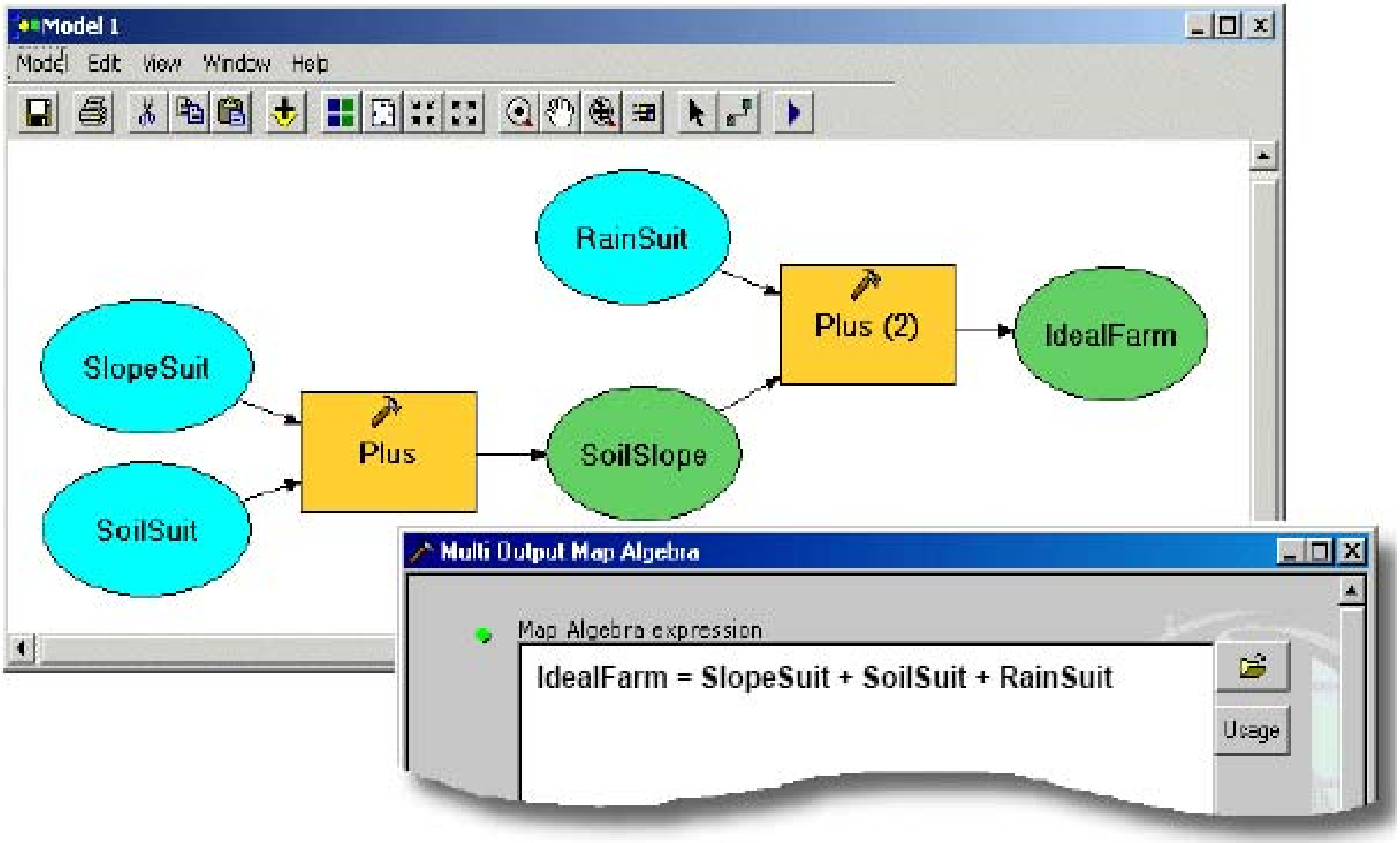
Multi Output Map Algebra Tool



Single Output Map Algebra Tool



Map Algebra vs. ModelBuilder



Exercise 3: Building a Raster Database

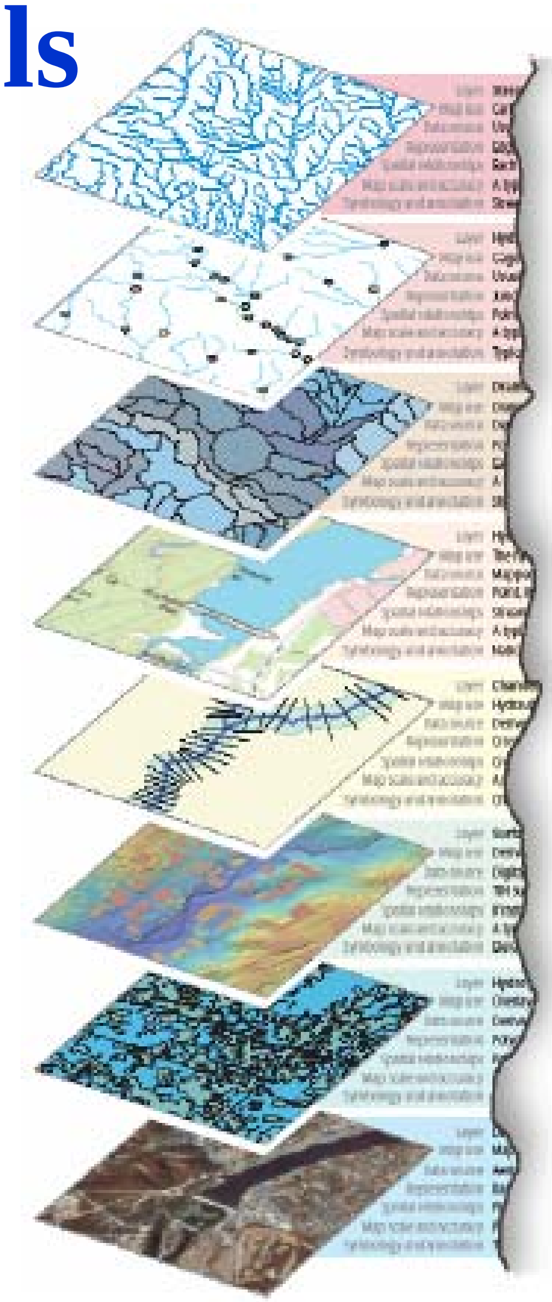
- EXERCISE 3A: BUILD A RASTER DATABASE
- EXERCISE 3B: GEOREFERENCE A RASTER

Building a Raster Database

- **Designing** a raster database
  - Poor design → **consequences, costs** in the future
- Need to **evaluate needs and plan accordingly**, before building a GIS database
- **Decisions** on the type of data to store, how to use the data, and on going maintenance
  - **Considering** these issues **beforehand** will help improve your design decisions

ESRI Data Models

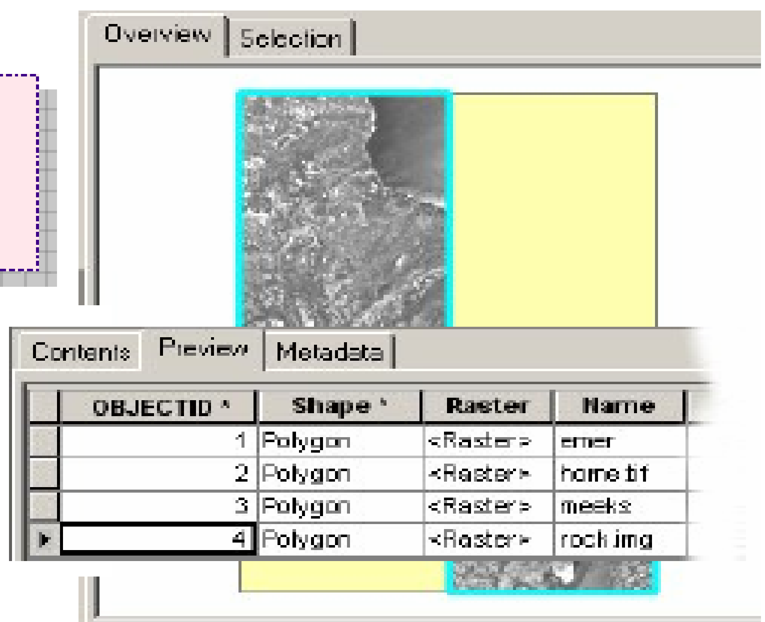
- Data models = schema **templates**
- Templates for **implementing** GIS projects
- **Speed up** development time
- Available for **many industries**
- Provided as **templates** to create
- Personal or file **geodatabases**

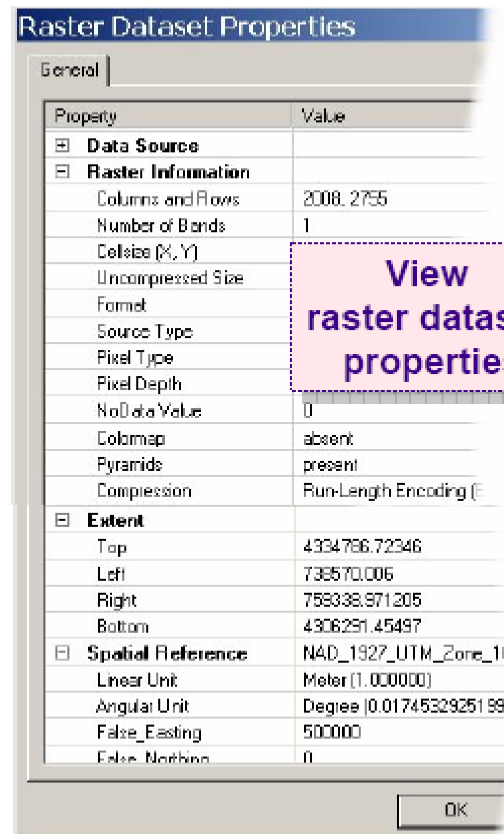


Rasters in ArcCatalog

- **Edit** spatial reference
- **Create** metadata
- Build **pyramids and statistics**

Examine raster catalog content





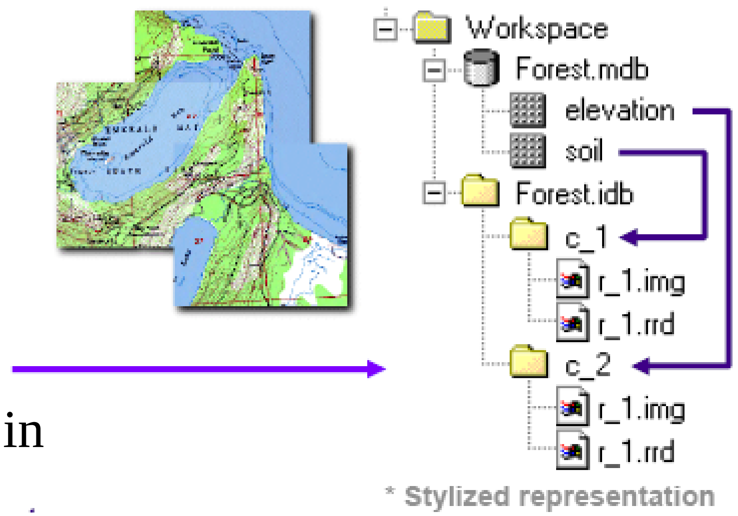
View raster dataset properties

Geoprocessing: Raster Management Tools

- General raster database **management tools**
  - Copy, paste, delete, calculate statistics, set spatial reference
- Data **organization/preparation**
  - Mosaicking
  - Raster catalogs
- Raster **data storage**
  - Pyramids
  - GDB technology

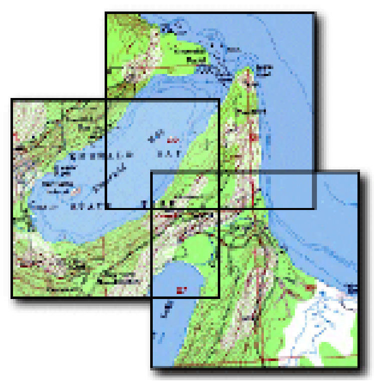
Geodatabase Raster Datasets

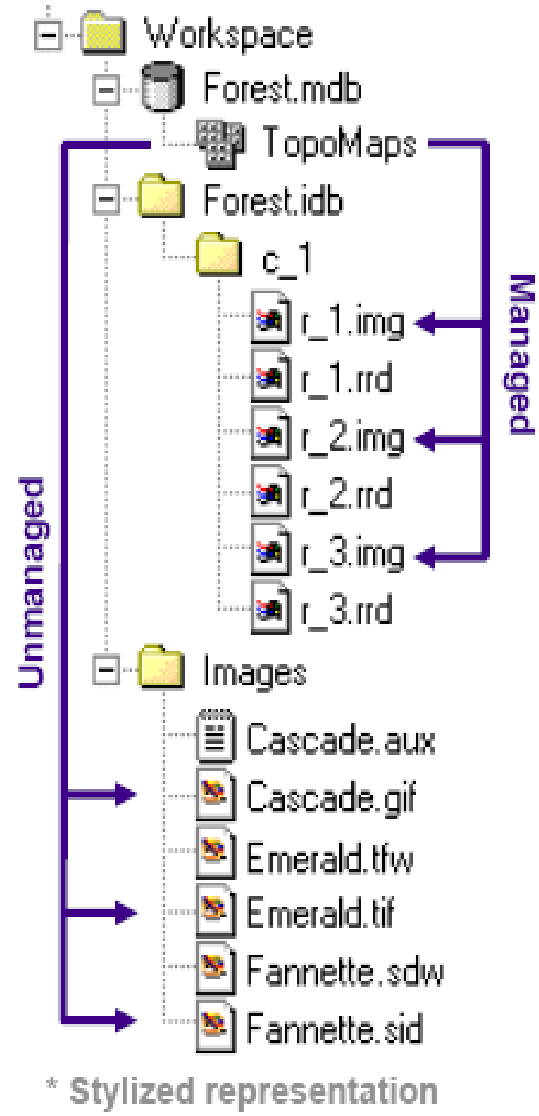
- A **single raster** in a geodatabase
  - May load many rasters into one raster dataset (mosaic)
  - Good for analysis and mapping
  - Seamless
  - Fast display at any scale
- **Personal geodatabase** format
  - Rasters converted to IMG format, stored in hidden .idb folder
- **File geodatabase** format
  - Rasters converted to FGDBR format, stored in the GDB folder
- **ArcSDE geodatabase** format
  - Rasters converted to ArcSDE raster format, stored in RDBMS
- All formats **preserve GRID attribute fields**



Geodatabase Raster Catalogs

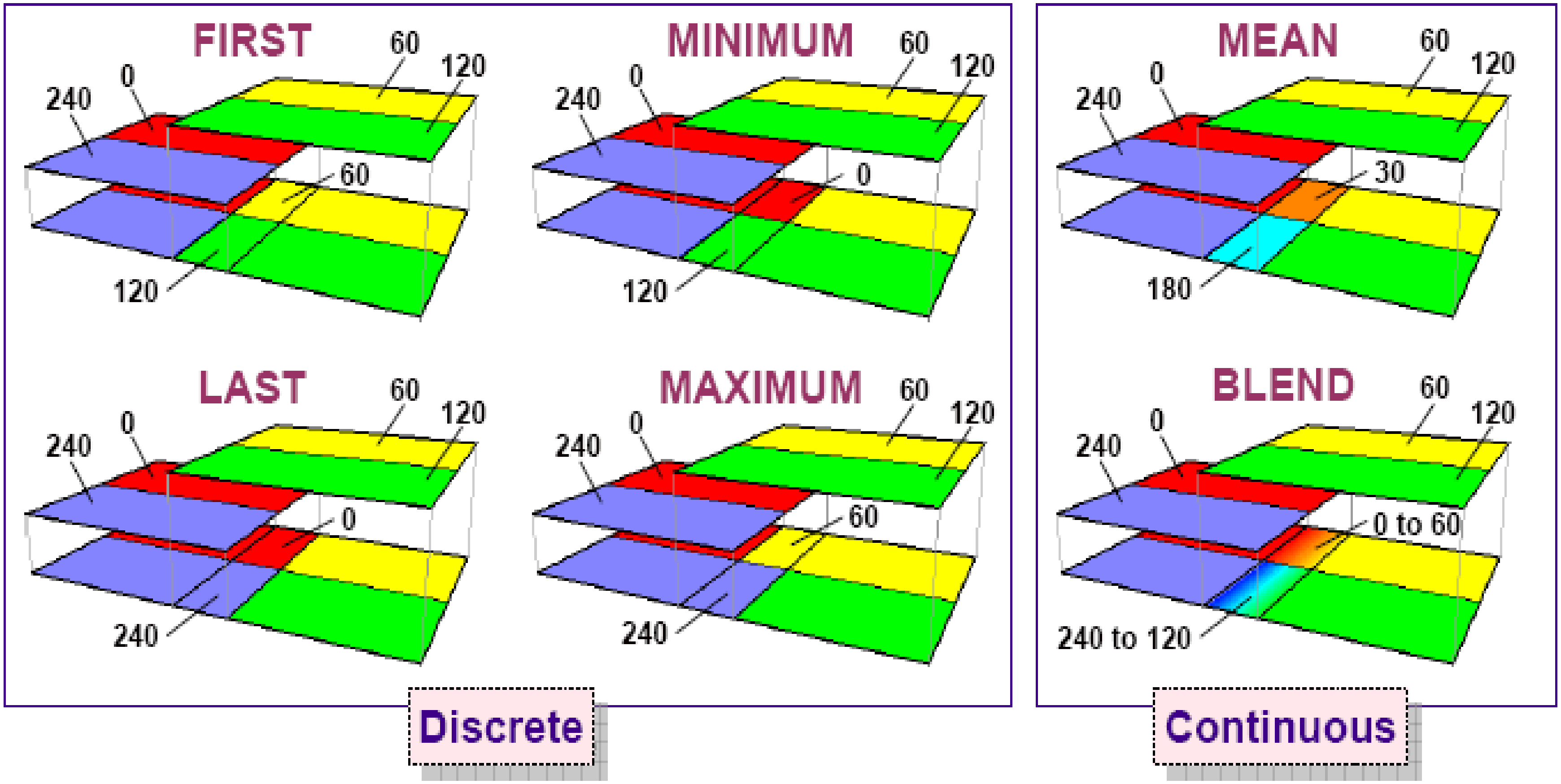
- A **collection** of raster datasets
  - Behave as one, but are **stored separately** (rows in a table)
  - May overlap, have gaps, different cell sizes, bit depths
  - Must have **same spatial reference**
  - Good for archives, display, and mapping
- **Personal geodatabase** format
  - **Managed**: Converted, stored in .idb folder
  - **Unmanaged**: Referenced by path name
- **File geodatabase** format
  - Can be managed or unmanaged
- **ArcSDE geodatabase** format
  - Necessarily managed





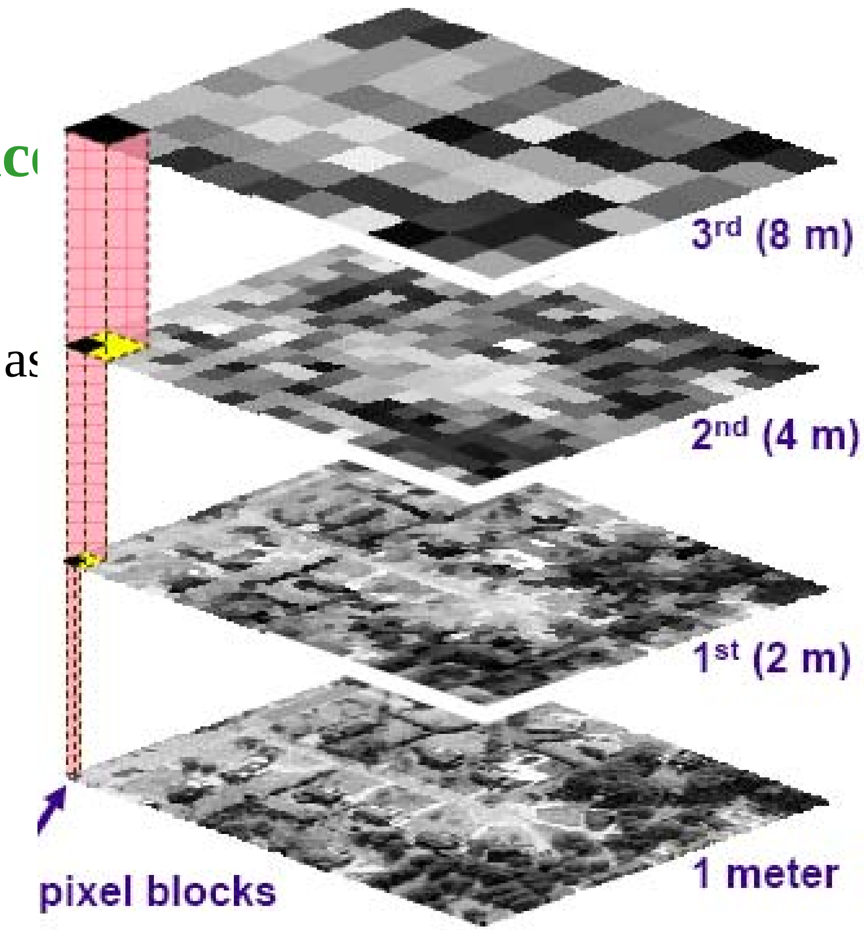
Merging Rasters

- Combine **multiple rasters** into one
  - **Six methods** to handle **overlapping** areas:



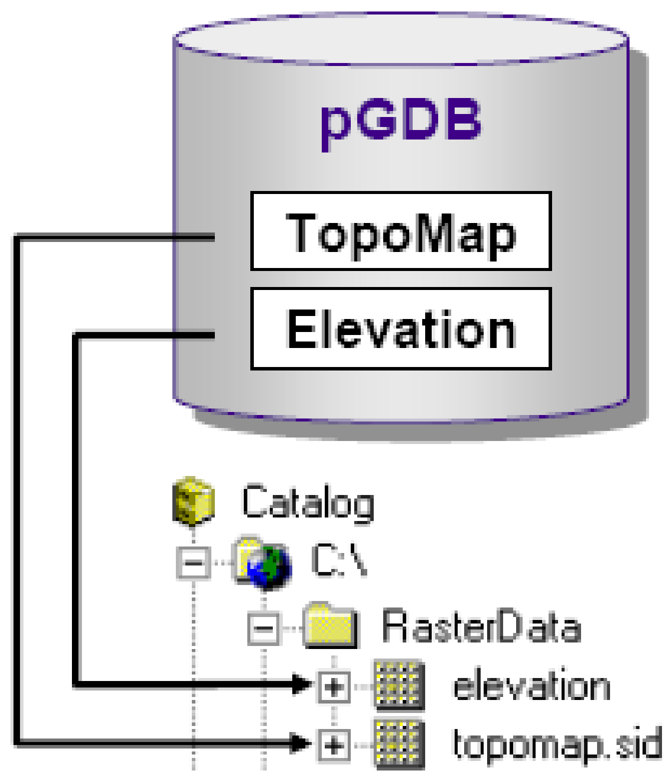
General Raster Properties: Pyramids

- **Reduced resolution copies** of original raster
  - Pixel size **doubles** at each level
- Improves query/display **performanc**
  - Returns **best resolution** for screen display
  - Returns about **same number of pixels** as **scale changes**
- Personal GDB
  - Stores pyramids in **RRD file**
- File GDB
  - Stores pyramids in **GDB folder**
- ArcSDE GDB
  - Stores pyramids **in tables**



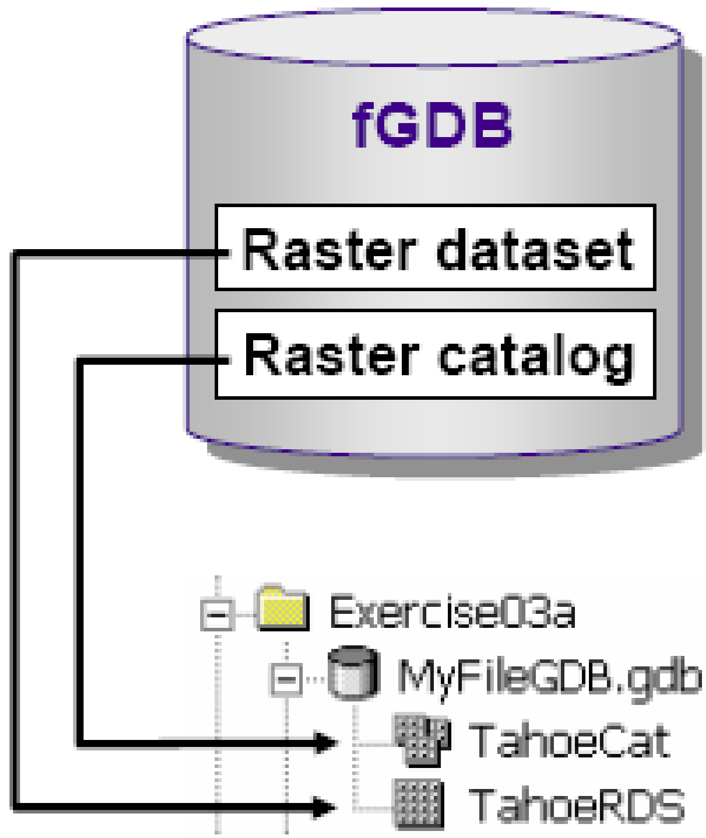
Rasters in a Personal Geodatabase

- Stores a **reference** to **external, file-based** rasters
  - Microsoft Access MDB file is **limited** to 2 GB total size
  - Provides **centralized access** to rasters
- Stores **raster datasets**
  - A single raster
  - Best for data
  - Can use in analysis
- Stores **raster catalogs**
  - A collection of rasters
  - Best for imagery archives
  - Cannot use in analysis



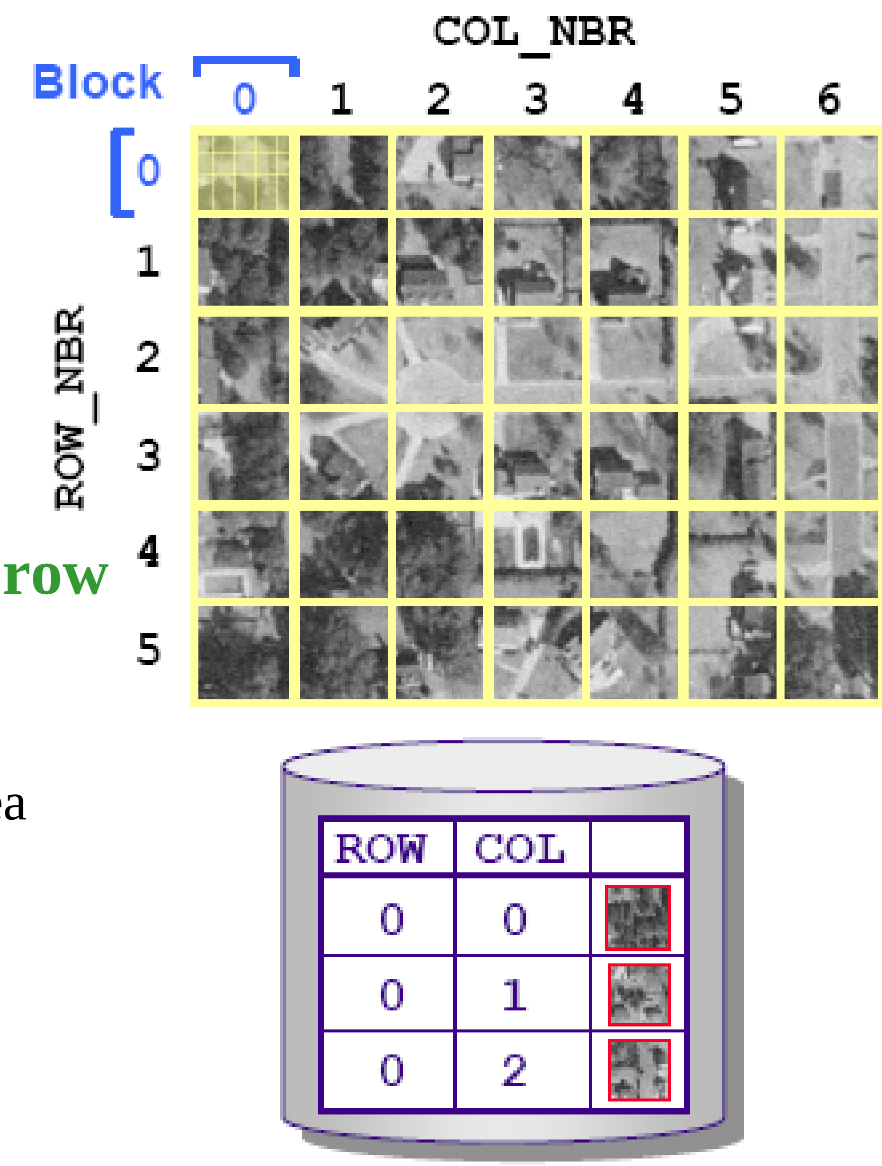
Rasters in a File Geodatabase

- Stores **raster catalogs** and **raster datasets**
  - Up to **one terabyte** for each raster dataset or raster catalog
  - Provides **centralized access** to rasters
- **Useful for**:
  - A single user and small work groups
  - Some readers and one writer



Rasters in an ArcSDE Geodatabase

- ArcSDE **subdivides** a raster into **blocks** for storage
  - Size set by user
  - Automatic and required
  - Invisible to end users
- The **raster is a table**; a **block is a row** in the table
- Provides **faster access** to data
  - ArcSDE returns blocks for visible area
  - Improves display performance



Next Topic:

Raster Analysis and Functions