Review of ArcGIS Object Modeling and UML
Presentation Overview

- Review object modeling process
- Review UML constructs
- Review UML conventions
Review of Terms

- **Data model**: UML-compliant object model created and maintained in a CASE tool.

- **Logical model**: shows objects, properties, and relationships in graphic format.

- **Physical model**: contains UML-compliant definition of model, can be used to generate schema.

- **Schema**: an empty Geodatabase that contains the feature classes, relationships, and associated entities.

- **Repository**: A database that contains the information required to generate a schema.
Schema creation process

Tool: Microsoft Visio

1. Review Logical Model with Project Team
2. Modify Model To fit Requirements
3. Export Model To Repository

Tool: ESRI ArcCatalog

1. Import Model
2. Configure Schema
3. Load Data
4. Configure Network

MSRepository Database
Pilot Data Migration Process

1. **Existing Databases**
2. **Conversion Planning**
3. **Data Conversion**
4. **Workshop**
5. **Prototype Model**
6. **Load Data**
7. **Configure**
8. **Review Data**
9. **ArcEnergy Model**
10. **Requirements**
11. **Repository**
12. **Generate Schema**

**Tools Used:**
- ArcCatalog
- ArcMap
- Visio
Modify The Object Model

Classes
Fields
Relationships
Domains

ESRI Classes:: SimpleJunctionFeature

Metadata Classes:: ArcFMUserMetaData
+CreationUser : esriFieldTypeString
+DateCreated : esriFieldTypeDate
+DateModified : esriFieldTypeDate
+LastUser : esriFieldTypeString

SimpleJunctionFeatures:: GasValve
+Material : GasValveMaterial
+ValveDiameter : GasValveDiameter
<<SubtypeField>>ValveType : GasValveType
+Depth : esriFieldTypeDouble

SimpleJunctionFeatures:: Gate Valve

SimpleJunctionFeatures:: Ball Valve

ESRI Classes:: Object

ObjectClass:: Valvelnspcetion
+InspectDate : esriFieldTypeDate
Export Schema

- Build *physical* model in Visio
- Export from Visio using UML tool
- Creates repository in MS Access or SQLServer (we’ll be using MS Access)
Generating the Geodatabase Schema

- Reads information from repository
- Creates schema (tables and relationships)
- Can build schema in ArcSDE or personal Geodatabase
Loading Data in ArcCatalog

- Load data by feature class
- RDBMS joins can be used for attributes
- Establish relationships
Geodatabase Structure

GeoDatabase
- Collection of FeatureDatasets
  - Like a workspace or instance, e.g. CE Gas Geodatabase

Feature Dataset
- Collection of FeatureClasses
  - Share same spatial reference, e.g. GasFacilities, Landbase

Feature Class
- Collection of Features
  - Share same attributes, e.g. RegulatorStations

Feature
- Geographic Object
  - Row in a DBMS Table, e.g. Valve #V156
Configuring Behavior in ArcCatalog

- **ArcCatalog Tools**

  1. **Feature Class Properties:** In ArcCatalog, you can set the properties for a feature class, including defining which domains go with which fields.

  2. **Connectivity Rules:** In ArcCatalog, you can define a geometric network the properties of it. The properties include what feature classes participate in the network and what can connect to what (and how).

  3. **User Roles:** Using ArcCatalog, you can define user access by dataset. (You can also use Oracle roles.)
Viewing Data in ArcMap

- Connect to Geodatabase
- Load layers into document
- Define display settings and save
  - Symbology
  - Labels
- Set up templates
- View, query, trace
Basic UML Constructs

- UML: Unified Modeling Language
- A notation to express a design or model
  - Standard for ESRI and industry
  - Syntax based, not process-based
  - Visio 5.0 Enterprise / Visio 2000 (also Rational Rose, MSVisual Modeler)

(Data) Models

- Logical – Graphic representation of objects, properties, relationships
- Physical – ‘Smart’ representation that specifies objects, properties, relationships, relationship classes, and domains
An **AbstractClass** cannot be used to create objects, but is a specification for instances of subclasses.

A **Class** can create new objects.

**Associations** represent relationships between classes.

An Association has **Multiplicity** which is an indication of how many objects may participate in the given relationship.

**Generalization** defines similar properties and methods among classes.

**Aggregation** defines a relationship where one of the objects is considered the ‘whole’ and objects from the other class are considered ‘parts.’
Object Model Concepts

- ESRI Defined Class
  - Abstract Class
    - Property1
    - Property2
  - Concrete Class1
    - Property1
    - Property2
  - Concrete Class2
    - Property1
    - Property2

Wormhole to Abstract Class (italicized)

Wormhole to Concrete Class (non-italicized)

Defined subtypes

Notes
UML Generalizations

- Connection between abstract and concrete class
- Promote inheritance
- Attributes in abstract are inherited by subclass
Associations

Types
- Binary (or Peer-to-Peer)
- Simple Aggregation
- Composite Aggregation

Cardinality
- 1:1
- 1:M
- M:N

Keys
- Primary/foreign keys
- Destination/Source
Subtypes

- Break down concrete class into more specific categories
- Don’t have overhead of adding more classes
- Must share properties and methods of concrete class
- Example: Distribution Main

Distribution Main

- Bare Steel Main
- Cast Iron Main
- Coated Steel Main
- Copper Main
- Plastic Main
ESRI Object Classes

ArcInfo provides a hierarchy of object classes ready for use:

- Object (Non-spatial)
- Feature (Spatial, non-networked, point, line, or polygon)
- Simple junction (Spatial, networked, point)
- Complex junction
- Simple edge (Spatial, networked, line)
- Complex edge
ESRI Features

- Features are spatial objects (classes)
- Geometry, location are properties
- Attributes (e.g., height, nominal voltage) are properties
- How feature behaves is determined by methods defined for it
  - Display, edit, query, etc.
- Data structures defined as part of object
ESRI Object Classes

Object

Feature

Geometry

NetworkFeature

EdgeFeature

SimpleEdge Feature

ComplexEdge Feature

JunctionFeature

SimpleJunction Feature

ComplexJunction Feature
Network Features

- **SimpleJunctionFeature**
  Features represented at a single insertion point that have a single point of connection with the logical network.

- **ComplexJunctionFeature**
  Features represented at a single insertion point that may have many points of connection with the logical network.

- **SimpleEdgeFeature**
  Linear features to which network connectivity may be established at either endpoint.

- **ComplexEdgeFeature**
  Linear features to which network connections may be made at multiple locations along its length.
Defining the model

ArcGIS Object Modeling
Compositon
Objects from the 'whole' class control the lifetime of the objects from the 'part' class.

Data Model Review: Objects

Objects

UnitObject

Manufacturer
Model
SerialNumber

CapacitorUnit

FluidType
RatedKVAR

CapacitorControl

LowerVoltageSetting
DeactivateSaturday
DeactivateSunday
Time[On,Off]
UpperVoltageSetting
SCADAControlled
SCADAMonitored
CurrentLevel[On,Off]
Temperature[On,Off]
VoltageLevel[On,Off]
KVARLevel[On,Off]

PFCorrectingEquipment

ControlType
FacilityID
LocationID
PhaseDesignation
OverheadOrUnderground
RatedKVAR

CapacitorBank

ConnectionConfiguration
NominalVoltage
Reactance
Switched
Grounded

SeriesCapacitor

Object

UnitObject

Object

Composition
Objects from the 'whole' class control the lifetime of the objects from the 'part' class.
Wormholes
Subclasses and Subtypes

- Subtype whenever possible
- Reduce the total number of classes
  - Increase performance
  - Better organization
- Differentiate within an object class
  - Same attributes and behavior
  - Can set default values, edit tasks, domains by subtype
  - Can configure relationships and connectivity rules at subtype level
Domains

Describe the legal values of a field type

- Used to promote attribute integrity

Types:

- Range: Pipe Length can be 1 to 1,000,000
- Coded Value: Gas Valve Use Type can be Bypass, Isolation, Purge, or Relief

Can be shared among classes

Can be defined at the feature class or subtype level
Domains and Subtypes

Feature class called Gas Pipe

Assign Domains by Subtype:

- **Plastic** – Domain for Diameter: ¾”, 1”, 1 ¼”, 2”, 3”, 4”, 6”, 8”, 10”, 12”
- **Copper** – Domain for Diameter: 3/8”, ½”, ¾”, 1”, 1 ¼”, 1 ½”
- **Cast Iron** – Domain for Diameter: 3”, 4”, 6”, 8”, 10”, 12”, 16”
- **Coated Steel** – Domain for Diameter: Values from ¼” to 36”
- **Steel** – Domain for Diameter: Values from ¼” to 36”
Default Values

- Can be established at the feature class or subtype level
- Enable default initialization of objects once a feature class or subtype is selected

Example: NonControllable Fitting

- Default Subtype: Coupling
- Default Material: Plastic
- Default Insulated Indicator: 1 (True)
- Default Bonded Indicator: 0 (False)
Relationships

Can Relate:
- Feature class to a object class
- Feature class to a feature class
- Object class to an object class

Can Define by:
- Concrete class to concrete class
- Subtype to subtype
Geometric Networks

- Topological relationship between feature classes
- Used to model network systems
- Traceable

Feature classes in a network must all be in the same feature dataset

Features can only participate in one network

Enabled field controls flow for simple traces – Trace Weights store information for more complex traces
Questions?
Association (or Relationship) Class

- Used with M..N associations
- Is an object (table in DB)
- Stores keys relating the two classes
- Can store other attributes as well
- Is Invisible to user

<table>
<thead>
<tr>
<th>Parcel</th>
<th>1..M</th>
<th>1..N</th>
<th>Owner</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
<td></td>
<td></td>
<td>Name</td>
</tr>
<tr>
<td>Area</td>
<td></td>
<td></td>
<td>Address</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Parcel_Owner</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Deed #</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Date</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ParcelObjectID</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>OwnerObjectID</td>
<td></td>
</tr>
</tbody>
</table>
Configuring Behavior in ArcCatalog

• **ArcFM Tools**
  
  1. **Properties Manager**: The Properties Manager assists you in determining field display order, feature behavior, and field behavior.
  
  2. **Snapping Manager**: Configure snapping preferences, priorities, and tolerances for the each feature class.
  
  3. **Favorites Manager**: Favorites allow you to create a system-wide (Geodatabase) library of frequently used new features which will be added to your map documents while editing in ArcMap.
Custom Object Classes

Build on ESRI hierarchy:

- Model real world facilities
- Define as subclasses to ESRI classes
  - Odorizers – point features
  - Pipes – complex edges
  - Valves – junctions
  - Inspection -- objects
  - Parcels – polygon features
- Define lowest differentiation for methods and properties
UML Classes

- **Encapsulation**
  - Data hiding, encapsulate internal details and present higher level of structure

- **Inheritance**
  - Consolidation of attributes, operations -- inherit behavior and properties of parent class(es)

- **Polymorphism**
  - Objects can take different forms, methods may be re-used
Validation Options

- **Domains** – restrict values user can enter in field
  - Can be defined at class or subtype level

- **Relationship Rules** – restrict cardinality, subtypes
  - Can be defined at class or subtype level
  - Association must already exist between feature classes

- **Connectivity Rules** – define network connections
  - Can be defined at class or subtype level
  - Edge/edge rules
  - Edge/junction rule