AZ-SMART Architecture Draft

Center for Urban Simulation and Policy Analysis
University of Washington

February 22, 2007

1 Proposed Architecture for AZ-SMART

Overview

1.1 Data

Overview

1.1.1 Implementation

Insert stuff about Geodatabase and OPUS Cache.

1.1.2 Data Models

Insert narrative and E-R diagrams.

1.1.3 Data Diagnostics, Validation, and Imputation

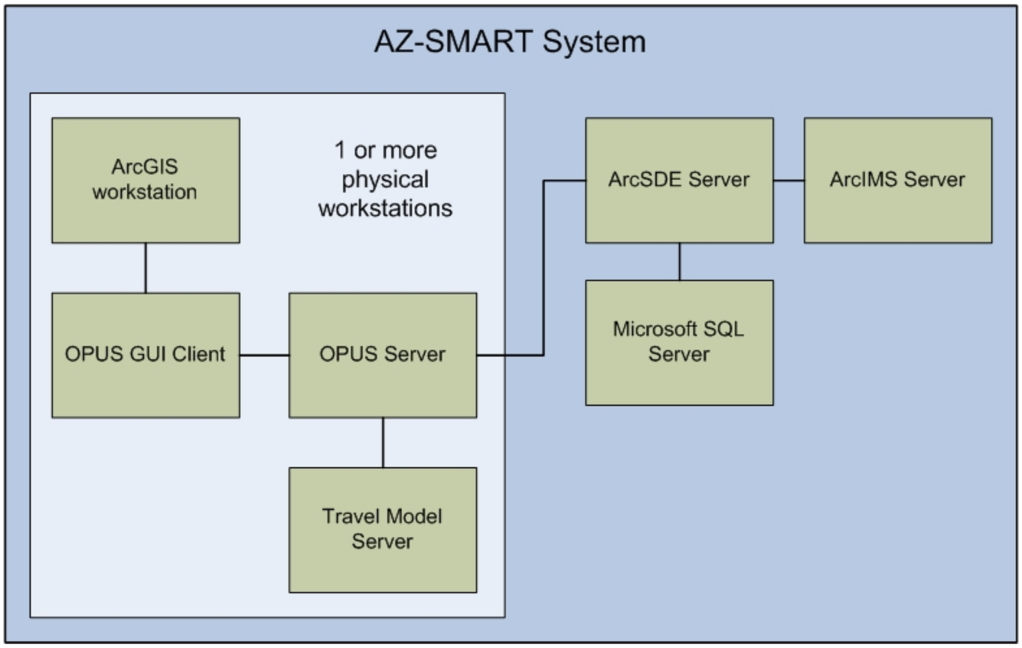
Insert stuff about data quality.

1.2 Software

Insert stuff about ArcGIS (ArcCatalog, ArcMap, ArcToolbox/ModelBuilder, dockable windows and toolbars), OPUS classes and methods, and how the two interact.

1.3 System

Insert stuff about the overall system.



1.4 Models

Insert stuff about the model system.

2 Core Modules in AZ-SMART

This section includes content from Appendix G of the AZ-SMART RFQ, along with with CUSPA's design recommendations for AZ-SMART core modules and including some questions that remain. The purpose of this section is to begin to reconcile the design, implementation, and feature ideas presented in Appendix G with functionality within ArcGIS and Opus, and to identify the nature of new development needed.

2.1 Data Manager

Overview

- Enhancements to ESRI ArcCatalog
- At minimum, maintain current functionality

  ArcCatalog will be used as the basis for the AZ-SMART Data Manager. An AZ-SMART directory structure would organize these data, scripts, configurations and other components.
- Access to, development, and maintenance of all data
- Create and track relationships (spatial and rule based) between datasets
- Uses tools from Tool Manager
- All data potentially used by more than one project. Examples include:
  - Land Use Codes
  - Base Year
  - Allocation Sector Names
  - Legends
  - Symbol table associated with global variables

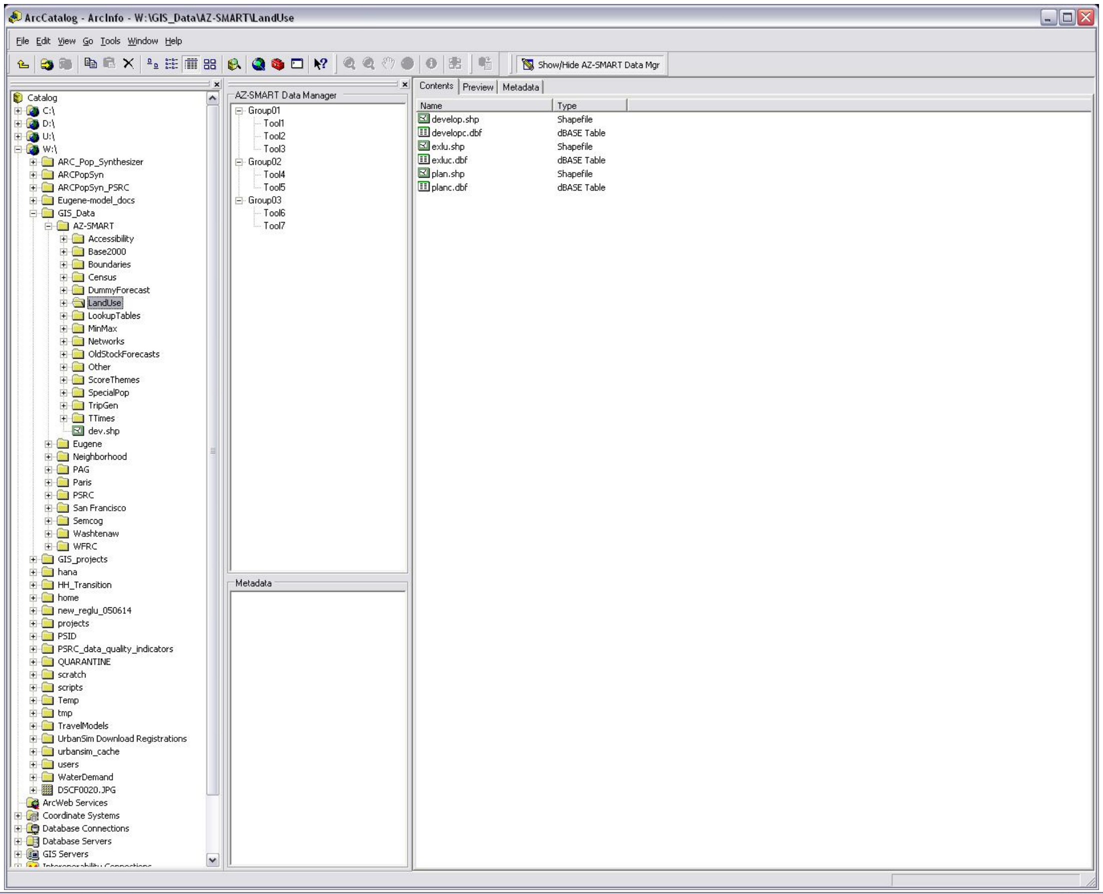
  CUSPA proposes to create a custom AZ-SMART ArcCatalog Tree structure, that would be a dockable window containing and organizing the data and tools used in AZ-SMART.
- Metadata must be maintained for all datasets

  Question for MAG: Need to jointly define the Metadata requirements and then devise a suitable design to address this need.
- Security

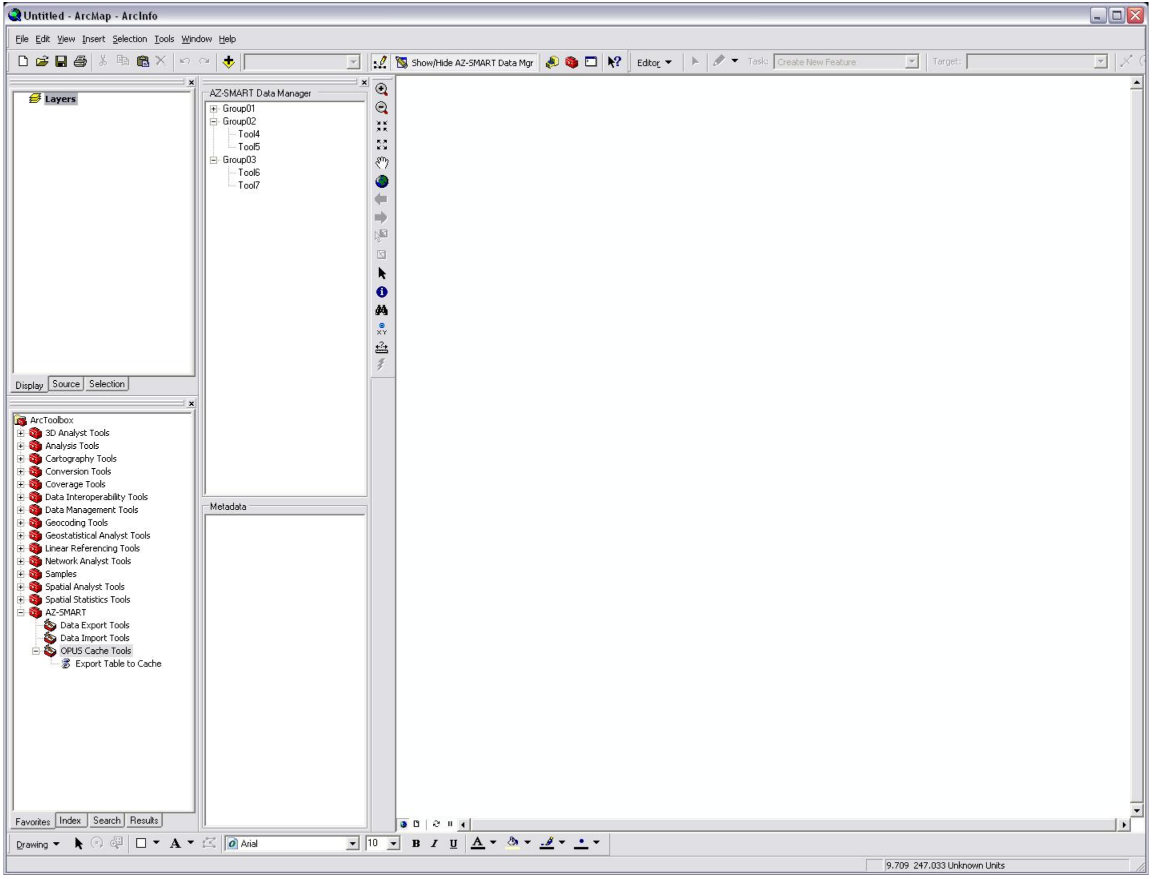
  Question for MAG: Please identify user groups and security levels that are desired throughout the system and the scope of permissions for each user group.
- Consultant to recommend directory structure

  A directory structure will be recommended and implemented in the AZ-SMART ArcCatalog Tree.
- Consultant to recommend and ultimately create data archiving procedures

  Archiving procedures will be recommended once the database and all items needing to be archived are identified.



Name	Type
develop.shp	Shapefile
developc.dbf	dBASE Table
exlu.shp	Shapefile
exluc.dbf	dBASE Table
plan.shp	Shapefile
planc.dbf	dBASE Table



2.2 Project Manager

Overview

Projects are defined as the database and model configuration used to support running a variety of scenarios that would share data and model specifications and parameters. Scenarios are defined as a set of input data, assumptions, and run configuration parameters used for a specific run of the model system.

CUSPA proposes to implement the Project Manager user interface as an OPUS GUI (developed using Envisage) that could be launched from an AZ-SMART Tool within ArcGIS. Compared to an approach of using only native ArcGIS GUI tools, this approach will make it more feasible to implement several of the requirements listed below, particularly the last three items, which are very focused on control of model operations – including during a simulation. It would need to be implemented in a way that could inter-operate transparently with other tools in the Tool Manager and Data Manager components, as noted in the first requirement below. Some initial tests of this approach should be developed early in the project to flesh out this aspect of the user interface.

- Create new projects and scenarios, or open projects and scenarios that have been created previously for further analysis
- Links tools from Tool Manager with data from Data Manager using ESRI ModelBuilder concepts
- Selects required components and limits execution of model to tools necessary for the scenario subset.
- Accesses all data relevant to potentially more than one project via the Data Manager
- Stores all data relevant to only that one project within a project. Examples include:
  - Projection Years
  - Switches utilized in the project
  - Status of the project
  - File and database names etc.
- Controls model execution: start, stop, and restart model execution
- Access the status of a model while executing
- Access various execution logs and error logs associated with a model run

2.3 Tool Manager

Overview

- Enhancements to ESRI ToolBox and ESRI ModelBuilder

  ESRI ArcToolbox and ModelBuilder will be used as the basis for the majority of the GUI for AZ-SMART. It will be enhanced by adding tools that leverage OPUS and UrbanSim functionality, generally through the OPUS GUI).
- Will have an indexing system by which end-users can find appropriate tools relevant to their needs

  The built-in ArcToolbox indexing system will be used for this purpose.
- Accesses Script Editor to code/implement tools

  The built-in editing functionality for Python will be used for this purpose.
- Uses ModelBuilder for computer-aided programming (i.e., data flow chart graphical interface)

  The built-in ModelBuilder functionality will be used for this purpose.
- Manage scripts: open, create, copy, delete, edit tools and their properties

  The built-in ArcToolbox functionality for managing scripts will be utilized here. This is accessible via context-menu selections at the toolset and tool level in ArcToolbox.

- Multiple security levels

  Question for MAG: Please identify user groups and security levels that are desired throughout the system and the scope of permissions for each user group.

- Multi-user system

  The system will be designed to use the ArcSDE multi-user environment with Microsoft SQL Server, which provides multi-user capabilities.

- Ability to share tool based on properties assigned

  Question for MAG: This functionality needs further clarification about what is needed. It may be possible to store tools in the Geodatabase, providing security and sharing capabilities within.

- Will perform conditionals and loops

  The built-in ArcToolbox and ModelBuilder functionality will be used to perform conditionals and loops, augmented by Python and OPUS capabilities.

- Unless specified, all tools should be capable of accepting input from user, script, output from other tools, or a database

  Tools will be constructed to meet this qualification.

- Consultant to recommend and ultimately create:
  - Tool archiving procedures

    This will require the use of a version control system such as Subversion. A Subversion repository has been set up for this project at CUSPA. An existing tool for accessing Subversion from Python could be linked from the user interface (www.pysvn.tigris.org).
  - Definition and management of production and development versions

    This could be accomplised by having two branches in the Subversion repostitory, one for production code, one for development code.
  - Prevention of the inadvertent modification and/or deletion of tools referenced elsewhere in the system

    This could be accomplished by placing relevant Toolboxes and Toolsets in a 'read-only' location.

Example Tools from MAG

2.3.1 Land Use Editor

Overview

- An ArcGIS application toolbar that provides controls specifically suited for editing and managing land use databases

  CUSPA proposes to use ArcGIS built-in editing functionality for editing of spatial features (e.g. existing land use, development projects), augmented by a toolset. If there is remaining functionality in SAM-IM that is not already addressed by built-in ArcGIS editing, then an ArcMap toolbar would be created to meet these needs.

- Maintains planar polygon topology/grid; Data model aware; Performs validation and domain checking

  CUSPA proposes to utilize built-in geodatabase functionality for maintaining topology, domain checking, etc. There are existing tools (e.g. Geodatabase designer 2) that can address these requirements.

- Advanced wizards for manipulating, validating, and assembling land use themes using interactive input and configurable rules

  CUSPA proposes to focus on developing tools and ModelBuilder models to meet these needs.

- Aggregation of parcels based on predefined rules (e.g. eliminate minor roads)

- Subdivision of parcels or polygons based on predefined rules

  The aggregation and subdivision of parcels or polygons are two complex tasks that need further design work and an assessment of alternative approaches.

- Variable sized grids with associated attribute data (e.g. areas with high vs. low resolution for modeling and analysis)

  CUSPA understands the potential benefits of variable sized grids for preserving small polygons in dense core of the urban area and avoiding wasted storage using small gridcells to represent large polygons in the urban periphery. However, implementing variable resolution grids is potentially a research project with considerable risk. Further discussion of the objectives, alternative strategies, and risks associated with these is needed.

- Can access data in Data Manager or Project Manager

  CUSPA proposes that all data in the geodatabase will be directly accessible in Data Manager and/or Project Manager as appropriate. Data generated during a simulation would be accessible in Data Manager and/or Project Manager through the use of tools to copy the data into the geodatabase or other tabular formats.

- Consistency checking across multiple themes

  Tools would be developed to check for consistency within and across themes as needed.

- Summary and indicator statistics

  This would be built using the Opus indicator framework.

- Completely configurable to suit any land use data model, coding scheme, and installation

  AZ-SMART will be designed to be highly modular and configurable and will use the flexibility inherent in the Opus sytesm.

- Similar to current editing capabilities in SAM-IM

  Existing ArcMap functionality will be utilized to provide this functionality.

2.3.2 Land Use and Socioeconomic Synthesizer

CUSPA proposes to develop tools for creating, manipulating, and synthesizing a base year database in the ArcToolbox/ModelBuilder environment.

- For creating and populating the base year (e.g., 2000) land use database by assembling multiple sources.
- For creating projected land use and socioeconomic datasets based on configurable rules.

  This would be done by implementing models in Opus and allowing them to be configured and run within ArcGIS.

2.3.3 Calibration and Validation

- Utilities for creating calibration data sets based on user supplied specifications
- Use 3rd party programs to perform regression analysis (e.g., ALOGIT, SPSS)
- Utilities for validating calibrated model data against observed data

  Opus includes tools for creating estimation datasets, estimating parameters of multiple regression and discrete choice models. These tools would be used for specifying and estimating models in AZ-SMART. Model validation will be supported by tools to compare predicted and observed data and visualize patterns of error in the results.

2.3.4 Analysis, Visualization, and Reporting

- A spatial calculator to perform computations on socioeconomic databases, examples:
  - Incorporate data from external sources;
  - Prorate projections to polygons based on a demographic property;
  - Drop point data into polygons (TAZ or land use polygons);
  - Perform row and column normalization and matrix balancing
  - Automatically summarize land use themes according to other polygon geographies (e.g., TAZs)
  - Calculate socioeconomic and land use statistics (population, employment, acres by type) for user defined areas based on geospatial rules.
  - Compute indicators and measures on land use or other polygon geographies (e.g., job-housing balance).

    Question to MAG: The needs detailed in this section warrant further discussion. CUSPA anticipates that some of this functionality will be handled by ArcGIS and Opus functionality.

- Capability to export tables to any file format, including custom format text files needed by travel models as well as ArcIMS; Users can define and save various file formats into a library of templates, and recall them for later exports.

  The system will provide a capacity for the user to define the particular data to be exported, variables, their sequence, the file format, and location. Export formats would include dbase, SQL server, ASCII (tab delimited, comma delimited, and fixed format). Question for MAG: Does this address the needs for ArcIMS?

- Provides methods by which end-users can define series of thematic maps to be generated automatically

- Provides methods by which end-users can define statistical tables and reports to be generated automatically

  The process of generating indicators and displaying them on a user configured base map will be automated.

2.3.5 Data Manipulation and Conversion Utilities

- Data available in a number of different file/DBMS formats: MS Excel spreadsheets, MS Access, Formatted ASCII files, Geodatabases, MySQL, etc.

- A library of utilities for accessing/converting data from one form to another so that it can be accessed directly by tools implementing models

  The system will support accessing and converting data among various data formats including but not limited to those formats listed above.

2.3.6 Accessibility

- Consultant to recommend and implement methodology or methodologies for travel times from geography to geography. Examples include:
  - Accesses travel times directly from third party systems used by MPOs (e.g., EMME/2, Cube)
  - Accesses travel times directly from modified third party systems using larger levels of geography
  - Creates travel times within AZ-SMART without using 3rd party systems

  CUSPA proposes to develop an interface to the forthcoming MAG TransCAD travel model if it is availabe within the timeframe of the AZ-SMART phase 1 project. It may be desirable to develop a sketch-level (fast running) travel model in later phases.

2.3.7 Submodels

CUSPA will need to work with MAG staff to develop clear specifications for sub models to be developed.

2.3.8 Site Suitability Tools

- Characterizes potential development sites throughout a region with respect to its suitability for development;
- A toolbox for portraying site characteristics from other GIS users (e.g., age and condition of structure, land value, proximity to highways, distance to developed land, residential market within 3 miles, etc.)
- Creates input datasets used in calibration
- An important component of allocation of lands during a projection, using calibrated factors

  CUSPA proposes to develop a set of tools organized within a Site Suitability Toolset to use themes for planned land use and various environmental or other features that would be used in determining suitability for each land use sector. These would be used to determine the capacity for development of each corresponding building type, such as Single Family units, Multi-family units, Office Sqft, etc, and could account for planning constraints such as minimum or maximum floor-area ratio (FAR) regulations.

2.3.9 Allocation Tool

This section describes functional requirements for a model system used to allocate land uses. CUSPA proposes to use existing functionality in OPUS to implement these models. The models and their specification would be configurable using dialog boxes or forms, and connected using a Model Builder style interface within the proposed OPUS GUI. Features described below will be incorporated into these tools. Comments inserted below are for clarification of the functional requirements. A more complete description of the design of the Allocation Models is provided in a subsequent section.

- A key tool for projecting growth in a region
- At minimum, maintain current functionality of SAM-IM

  The intent is to provide at least comparable functionality, though not using exactly the same approach.
- Process works by selecting lands, among candidates, to be built in order to absorb growth based on an evaluation of their inherent site suitability characteristics
- Features include but are not limited to:
  - Observes constraint layers that prohibit development due to environmental or policy factors
  - Observes general plan layers that designate acceptable conforming land uses and densities
  - Accepts any land use coding scheme that the user defines
  - Allocation sectors (variables of interest for projections) are user-defined
  - Sectors are allocated in a user-defined sequence.
  - Mechanism by which large development tracts are subdivided into parcels appropriate in size for the development considered

    Note earlier comments that aggregation and subdivision will require substantial design work and further discussion.
  - Ability to observe adopted land use plans and densities on a polygon/grid basis
  - Development Velocity Curve dictates the pace at which developments are built
  - Observes regional control totals of growth, or growth forecasts for subareas, as defined
  - Address ”mixed use” polygons
  - Address redevelopment and demolition
- Same process can be used, with different inputs, for vacating lands due to demolition and redevelopment
- Controlled by a number of different switches and rules supplied by the user that control how the allocation process specifically works

- Driven by a set of projected control totals of population and employment change that apply to the entire region or subareas of it
- Can control subarea growth at different geographic levels
- Capability for ”gravity effects” model projection mechanisms reacting to measures of accessibility, land use constraints and opportunities, growth trends, and other socioeconomic attributes
- Provides specific treatment of known developments scheduled to be underway
- Provides support for analysis of scenarios:
  - Generates alternative scenarios of land use and socioeconomic projections
  - Ability to work on complete area or revision-areas (sub-parts of complete modeling area)
  - Interactive designation of ”revision areas”; Capability to manipulate both polygon and grid
  - Migrates changes in downstream years; that is, changes made to a 2010 forecast migrated automatically to subsequent years;
- Provides different ways to react:
  - When build-out conditions are reached in individual subareas
  - How active developments are treated
  - With respect to policy initiatives
  - To demolition and redevelopment
- Different applications of the Allocation procedure in the projection model stream:
  - Regular production projections
  - ”Min-Max” procedure to create set of floors and ceilings to estimate reasonable growth potential
  - ”Scenario Builder” enabling analysis of changes to land use and other policy variables.

3 Nouns and Verbs

Following are lists of the nouns and verbs identified by reading Appendix G. (We are still assembling this.) We expect that each of these nouns will correspond to a class in our object-oriented system, so understanding the nouns and verbs is important for understanding what we are building. In general, a noun corresponds to a class, and a verb corresponds to a method of that class. Classes generally have additional methods and properties for internal use in the system.

3.1 Nouns from Appendix G

- Meta-data. Every noun has meta-data that covers its who, what, when, where, and why. In addition, a user may enter arbitrary key, value pairs?
- Project. A project defines the geographical scope, set of issues of concern, time-frame desired, etc. for a specific investigation into some set of issues.
- Scenario. A scenario is a particular configuration of input data, assumptions, and models to run to test a particular alternative future. Every project will eventually have at least one scenario. You can only ’run’ scenarios.
- Scenario run. Information about the running of a scenario. This includes meta-data and simulation results. Simulation results may be viewed by indicators.
- Indicator definition. Specification of how to compute a particular indicator.
  - Map indicator definition.
  - Table indicator definition.

  - Chart indicator definition.
  - Report indicator definition. This may consist of meta-data as well as a collection of other indicators (maps, tables, charts, etc.).

- Indicator result. Result of running an indicator or a scenario run. Synonym: prediction?
  - Map indicator results.
  - Table indicator results.
  - Chart indicator results.
  - Report indicator results.
- Indicator set. Multiple indicators. Allows multiple indicators to be operated on as a unit (e.g. create all of these).
- Data-flow diagram. This is a "model" in ModelBuilder. It is a visual representation of a data flow from a set of data sources, through a set of actions, to a set of data outputs. Includes conditionals and loops.
- Scenario subset. What is this?
- Model.
- Development.
- Custom procedure. For instance, a script, or a data-flow diagram.
- Tool. A software component that has a user interface.
- Software component.
- Data input.
- Data output.
- ...

3.2 Verbs from Appendix G

- New. Synonym: create.
- Save.
- Edit.
- Open/View.
- Delete.
- Copy.
- Run. Synonym: project.
- Manage. What does this mean?
- Analyze.
- ...

3.3 Combining Nouns and Verbs

These nouns and verbs could suggest the following menu items or tools that correspond to features or feature categories:

- Project
  - New project.
  - Open project.
  - Save project.
  - Modify project.
  - Copy project.
  - Delete project.
- Scenario
  - New scenario. Linked to a particular project.
  - Open scenario.
  - Save scenario.
  - Modify scenario.
  - Copy scenario.
  - Run scenario. Produces a scenario run.
  - Delete scenario.
- Scenario Run
  - Open scenario run (read only?).
  - Delete all or part of a scenario run. For instance, to prepare to re-run starting in 2020.
  - Copy all or part of a scenario run. For instance, to send to a colleague.
- Indicator Definition
  - New indicator definition. Defines how to compute an indicator results.
  - Open indicator definition.
  - Save indicator definition.
  - Delete indicator definition.
- Indicator Results
  - New indicator results. Compute an indicator results from an indicator definition.
  - Open/View indicator results.
  - ...
- Indicator Set
  - Create an indicator set.
  - View/edit/save an indicator set.
- Data-flow Diagram.
  - New data-flow diagram.
  - ...
  - Edit component from diagram. A componet may be a node or an edge. For instance, right-click on a data-source to set its properties.
  - Run data-flow diagram.

- Validate data-flow diagram.
- Step-over data-flow diagram. Runs next step in diagram.
- Step-into current node. During simulation.
- Set breakpoint. Execution will stop just before executing the component that has the breakpoint.
- Open selected node. Opens editor for the node, which may be a data-flow diagram itself.
- Open selected edge. Opens property editor for the edge.
- Select node. To edit, open, move, delete, etc.