1 Core Modules in AZ-SMART

The next 3 sub-sections synthesize narrative from RFG Appendix G with CUSPA's design recommendations for AZ-SMART core modules. The purpose of this is to begin to reconcile the design, implementation, and feature ideas presented in Appendix G with existing or future Opus functionality.

1.1 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 leveredge OPUS and UrbanSim functionality, generally through added tools within ArcToolbox that can be used interoperably within ModelBuilder (to be verified by CUSPA).

- 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.

- 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 may require the use of a version control system such as Subversion (e.g. open-source and free software application). A Subversion repository has been set up for this project at CUSPA.
- 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

1.1.1 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

  CUSPA suggests that these two items are a complex research task that requires further discussion with MAG.
- 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.

1.1.2 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.

1.1.3 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.

1.1.4 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.

1.1.5 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.

1.1.6 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.

1.1.7 Submodels

CUSPA will need to work with MAG staff to develop clear specifications for sub models to be developed. CUSPA will then implement them as mutually agreed upon.

1.1.8 Site Suitability Tools

Jesse says: To some extent this sounds like what Opus does automatically through its models, but perhaps MAG is looking for something more interactive than what we are used to.

- 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.) *Jesse says: This sounds a lot like what a GIS already does: symbolize the characteristics of spatial data, is there more to this?*

- Creates input datasets used in calibration *Jesse says: I do not understand what exactly this would create.*
- An important component of allocation of lands during a projection, using calibrated factors

1.1.9 Allocation Tool

Jesse says: This is the key simulation 'tool' (although I am not sure this is a tool as it is defined it the Tool Manager above). Many of these things are done by Opus during a simulation run. What I think is missing from Opus, and what MAG expects to see, is the ability to run the simulation in an interactive manner. Below I have inserted comments where I have a question on whether or not Opus does this.

- A key tool for projecting growth in a region
- At minimum, maintain current functionality of SAM-IM
- 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 *Jesse says: This is a little different than Opus, not sure how to reconcile Opus and SAM-IM here.*
  - Sectors are allocated in a user-defined sequence. *Jesse says: I think this is different than Opus too.*
  - Mechanism by which large development tracts are subdivided into parcels appropriate in size for the development considered *Jesse says: There is a lot of work here to subdivide and aggregate polygonal parcels*
  - 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 *Jesse says: We need to implement the 'Velocity Curve' idea into Opus.*
  - Observes regional control totals of growth, or growth forecasts for subareas, as defined *Jesse says: I am ignorant as to how this differs from densities allowed in the plan. I am not sure if Opus does this or not.*
  - Address "mixed use" polygons
  - Address redevelopment and demolition *Jesse says: Does Opus do this?*
- Same process can be used, with different inputs, for vacating lands due to demolition and redevelopment *Jesse says: This sounds like a 'negative growth' idea.*
- Controlled by a number of different switches and rules supplied by the user that control how the allocation process specifically works *Jesse says: Exactly what switches and rules?*
- Driven by a set of projected control totals of population and employment change that apply to the entire region or subareas of it *Jesse says: Again, is this fundamentally different than Opus?*
- Can control subarea growth at different geographic levels *Jesse says: I am unsure what this means.*
- 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 *Jesse says: This combined with the Velocity Curve represents new Opus functionality.*
- 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; *Jesse says: Does this mean that the user should have the ability to observe some prediction in 2010, decide that it is wrong, modify it manually, then restart the model from there? Or are changes to 2010 expected to automatically 'propagate' to subsequent years *without* re-running the model? This sounds complicated whatever it is.*

- Provides different ways to react: *Jesse says: This again implies mid-simulation run interactivity. How are we going to handle this? What should a user be able to change in the middle of a simulation run?*
  - 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: *Jesse says: I am fairly confused by this description of functionality.*
  - 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.

1.2 Data Manager

Overview

- Enhancements to ESRI ArcCatalog *Jesse says: An extension to ArcCatalog makes sense to me here, sort of. This raises the question of how are 'tools' accessible in and out of the ArcGIS environment, and specifically where do they reside?*
- At minimum, maintain current functionality
- Access to, development, and maintenance of all data *Jesse says: All data including Opus cache?*
- Create and track relationships (spatial and rule based) between datasets *Jesse says: ArcCatalog and the geodatabase framework have a lot of functionality built in for this sort of thing. I am not exactly sure what it means to 'track' a relationship though, perhaps just document it?*
- Uses tools from Tool Manager *Jesse says: Exactly how this is accomplished I do not know, see comment above.*
- All data potentially used by more than one project. Examples include: *Jesse says: Data being used across projects/scenarios makes sense, but is it the same data, copies of data, or just changes to data that are being used in different projects/scenarios?*
  - Land Use Codes
  - Base Year
  - Allocation Sector Names
  - Legends
  - Symbol table associated with global variables
- Metadata must be maintained for all datasets *Jesse says: What kind of metadata and in what format?*
- Security *Jesse says: This is a big issue that needs consideration up front and outside of simply the 'Data Manager' framework. What types of users are there and what can the do and not do?*

- Consultant to recommend directory structure *Jesse says: This should not be difficult once some other details are worked out.*
- Consultant to recommend and ultimately create data archiving procedures *Jesse says: Does archiving mean a database backup strategy? AZ-SMART will undoubtedly produce copious amounts of data in different formats.*

1.3 Project Manager

Overview

- Create new projects and scenarios, or open projects and scenarios that have been created previously for further analysis *Jesse says: I am a little unclear about the difference between projects and scenarios. Do you 'run' a project, or do you 'run' a scenario? David and Jesse said: Here is a proposal: A project defines the geographical scope, set of issues of concern, time-frame desired, granularity, etc. for a specific investigation into some set of issues. 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.*
- Links tools from Tool Manager with data from Data Manager using ESRI ModelBuilder concepts *Jesse says: This sounds like a version of ESRI's model builder that allows functions (tools) and data (tables, feature classes, etc.) to be linked in a flowchart fashion to compose a project/scenario.*
- Selects required components and limits execution of model to tools necessary for the scenario subset. *Jesse says: I think that the model builder type interface above implicitly limits projects/scenarios to running only tools that were included within them. Is there more to this feature that I am missing?*
- Accesses all data relevant to potentially more than one project via the Data Manager *Jesse says: I am not sure I understand this.*
- 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. *Jesse says: So do projects/scenarios 'store' data, copies of data, or references to data?*
- Controls model execution: start, stop, and restart model execution *Jesse says: This makes sense to me.*
- Access the status of a model while executing *Jesse says: This makes sense to me.*
- Access various execution logs and error logs associated with a model run *Jesse says: This makes sense to me.*

2 Overall open-ended questions and comments

- *Jesse says: My biggest question is this: What is the user going to see when they double-click 'AZ-SMART.exe' on their desktop? Does it bring up one of the above 'managers' (project, data, tool) or perhaps some wrapper application that incorporates those 'managers'?*
- *Jesse says: Are the managers as listed above the appropriate framework for AZ-SMART, or is there an alternative way of conceptualizing it? Unless we have a better way of splitting up the functionality I say we stick closely to this.*
- *Jesse says: I think we need some sort of 'security model' for AZ-SMART. Sprinkled throughout there is a desire for locking down certain parts of the system depending on what type of user you are. This is an issue we should address as we design and build the system. Adding security levels later could be a problem.*

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.