Remediation and Treatment Technology Development and Support for DOE Oak Ridge Office: Geodatabase Development for Hydrological Modeling Support

FIU | **Applied Research Center**
FLORIDA INTERNATIONAL UNIVERSITY

Executive Summary

During FY11 (FIU Year 2), researchers at the Applied Research Center (ARC) at Florida International University (FIU) developed a geodatabase to support the hydrological work being performed at Oak Ridge Reservation (ORR), which serves as a centralized data management system, making the large amounts of data generated from the simulations of contaminant fate and transport accessible to all users. The geodatabase facilitates storage, concurrent editing and import/export of model configuration and output data that is specific to the hydrologic models being used. The work for FY12 (FIU Year 3) extended the geodatabase capabilities and created models using ArcGIS ModelBuilder and Python scripting that automate the process of querying the existing EFPC geodatabase and the generation of maps. Investigation of easily downloadable free/open source geographic information systems (GIS) software for viewing and querying the hydrological modeling data and for generating maps, graphs and reports, was then conducted to determine a simple way of sharing project derived data with other project stakeholders such as U.S. Department of Energy (DOE) personnel and ORR site contractors.

Table of Contents

List of Tables

List of Figures

Introduction

During FY11, FIU developed a geodatabase to support the hydrological modeling work performed by FIU, which involves the use of three integrated watershed models for the Y-12 NSC, White Oak Creek (WOC), and East Fork Poplar Creek (EFPC) to simulate the fate and transport of contaminants. More than a hundred simulations were completed for the purpose of calibrating the models, deriving model uncertainties, and for providing the analysis of remediation scenarios, resulting in gigabytes of simulation data. Therefore, there was a need for an advanced spatial data structure that would be used to address the management, processing, and analysis of spatial and temporal numerical modeling data derived from multiple sources, and to produce hydrogeological maps for visualization.

The ORR Geodatabase is a multiuser relational database management system (RDBMS) built upon a Microsoft SQL Server platform developed using Environmental Systems Research Institute (ESRI) ArcSDE technology. The system was deployed on an advanced Windows server with the latest technology and hardware and provides a user interface which facilitates data access, database connectivity, cryptography, web application development, numeric algorithms, and network communications. The ORR Geodatabase is based on the ArcHydro and ArcGIS Base Map data models. The ArcHydro data model contains a set of accompanying tools designed to support water resources applications within the ArcGIS environment. These data models were also used as templates as there were many input data types in common with the ORR Geodatabase. Modifications were then made for project specific input parameters. The geodatabase serves as a centralized data management system, making the mass amounts of data generated from the simulations of contaminant fate and transport accessible to all users and facilitates storage, concurrent editing and import/export of model configuration and output data.

The ORR geodatabase was configured for concurrent multi-user access and editing capability, adhering to the appropriate security and quality assurance protocols necessary to maintain data integrity. This process exerts control on the type of access all users have to the geodatabase and its datasets, and enables specification of user data management privileges. Connection to the geodatabase requires Windows-authenticated credentials, and permissions are assigned to users according to the existing roles contained in SQL Server which already have predefined sets of

permissions; however, the option to create new roles and set associated permissions is also possible if necessary. Windows authentication for access to the geodatabase was selected for simple configuration utilizing a certificate-based security mechanism, which is more secure than an operating system (OS)-based authentication. Besides the FIU-ARC firewalls and the built-in software and hardware security protocols, the geodatabase resides on computers which must adhere to the FIU's University Technology Services (UTS) Security and IT Policies which are outlined in detail at http://it.fiu.edu/security/index.shtml.

Technical Approach

The objective of this task during FY12, is to extend the geodatabase capabilities by creating a GIS-based model using ArcGIS ModelBuilder as well as Python scripting that will automate repetitive processes such as: (1) querying the existing ORR Geodatabase for files used in hydrological model development and performance of numerical simulations; (2) performing statistical analyses based on specified algorithms; and (3) generating maps with the spatial distribution of computed and observed data.

The use of PYTHON scripts provides capabilities including:

- Use of ArcGIS python geoprocessing libraries to iterate over any selected feature and to provide immediate analysis, updates, and modifications of the geodatabase.

- Creation of customized functions to handle various aspects of the analysis, creating more classes to implement object oriented capability and to promote code reuse within the application.

- Preprocessing of input shapefiles (sewer lines, nodes, links and any infrastructure) and creation of new shapefiles on the fly. This includes analysis of the attributes of the input files and creating of a new set of shapefiles using selected attributes.

- Iteration through a list of infrastructure elements (links, lines, nodes) selected by a polygon shapefile and for each selected component analysis of a number of attributes in the shapefiles, including total length of lines that will remain, are missing, undersized or to be abandoned, which provides analytic capabilities to derive information about each specific file.

- Automatic addition of multiple attributes to point, polygon and polyline shapefiles, including attributes that will be used for project tracking.

- Additional post-processing of relevant shapefiles, including selection, printing, formatting with a toolbox specifically designed to minimize the manual set up and efforts.

- Automatic zooming, mapping and exporting of selected spatial features to excel, pdf and CAD formats as needed.

- Creation of a special toolbox with required tools to provide GUI access to data analysis, creation of new shapefiles based on input, printing and merging information without requiring the user to spend time on entering data separately.

- Error checking and analysis of existing shapefiles and preparation for use in XPSWWM models (particularly nodes and lines).

- Creation of reports on information contained in map documents such as data frame coordinate system, layer data sources, layers with broken data sources, or layout element positioning; update, repair, or replacement of layer data sources in a map document or layer file; and update of layer symbology without having to physically open map documents.

- Creation of geographic data in batch using map export commands, such as a series of GeoTIFF images driven by a list of features in a data frame. Automation of the creation and management of map services to be published with ArcGIS Server.

- Building a variety of PDF map books, including a thematic or temporal map book with title page, multiple map pages, and any number of additional pages with supporting content such as tabular reports and contact lists. A reference map book based on Data Driven Pages which allows rapid processing and export of a series of layout pages from the map document (and selected ATLAS sheets).

Development of customized Python scripts for the purposes of this task require additional programming of built-in automated geoprocessing tools to call or retrieve data from the ORR geodatabase. The toolbox being developed will allow the ArcGIS program to iterate over a several GIS files for pre- or post-processing of data to be used in or that are derived from hydrological modeling. ArcGIS ModelBuilder will then be used to document the processes and workflows employed.

Figure 1 below, shows the overall ORR Geodatabase schema generated using the ArcGIS "Geodatabase Diagrammer" utility for ArcGIS 10. More details are visible in Figures 2-4. These images basically depict the data structure of the ORR geodatabase and the feature datasets and feature classes that are used or generated during hydrological model development as specified in Table 1.

Table 1. Model Configuration Files Stored in the ORR Geodatabase

| Spatial Data	Characteristics Represented
Admin_Features	EFPC, WOC, Y-12 & OSY Model domains (polygons)
	ORR Boundary
Admin_GRIDs	Model domains (GRIDs)
Conductivity_GRIDs	Hydraulic conductivity GRIDs
Contaminant_Conc_Features	Monitoring points (has associated timeseries attribute data)
	Plume Contours
Contaminant_Conc_GRIDs	Interpolated contaminant plumes (GRIDs)
DEMs	Clinch River, EFPC & WOC Watershed DEMs
Digital_Orthophotos	ORR DOQs (.bmp)
Drainage_GRIDs	Drainage Time Constant, Drainage Codes, Detention Storage (GRIDs)
GW_Features	Groundwater level contours
GW_GRIDs	Groundwater level GRIDs
Hydro Features	Watersheds, subwatersheds, catchments, hydroareas (lakes/ponds) (polygons)
	Floodplain polygons
	Hydrography, Hydrodrainage, hydrostructures (polylines)
Impervious_GRIDs	Paved runoff coefficient (GRID)
Landcover_Landuse_Features	Landuse/Landcover polygons
Landcover_Landuse_GRIDs	Vegetation grid codes
Manning	Manning's coefficients (GRIDs)
Monitoring_Stations	USGS SW monitoring stations, outfalls, GW monitoring wells
Network_Features	Rivers, streams, reaches, cross sections, diversion ditch, utilities (polylines)
	Nodes (points)
Physical_Features	Buildings, obscured areas, natural outlines, man-made outlines (polygons)
	Margins, man-made structures (polylines)
Soils	Geology, soils (polygons)
Topo_Features	Elevation contours
Transport_Features	Roads, railroads, transportation structures (polylines)
Temporal Data	Characteristics Represented
Monthly_RF_TS	Monthly rainfall timeseries
Flow_Aug_TS	Flow augmentation timeseries
DHI_Timeseries	Flow rate/discharge timeseries
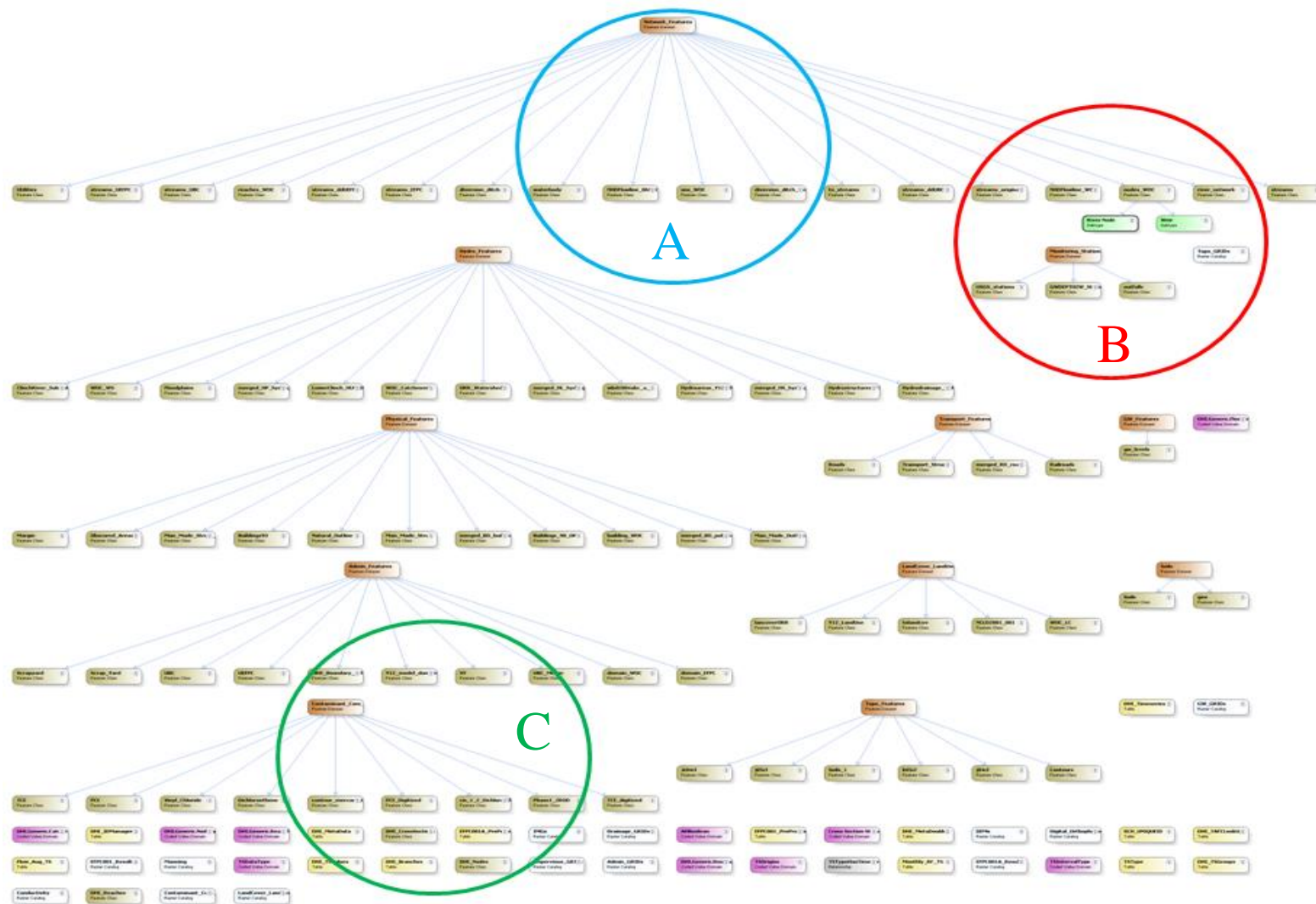
Figure 1. The ORR Geodatabase schema generated using the ArcGIS "Geodatabase Diagrammer" utility for ArcGIS 10.
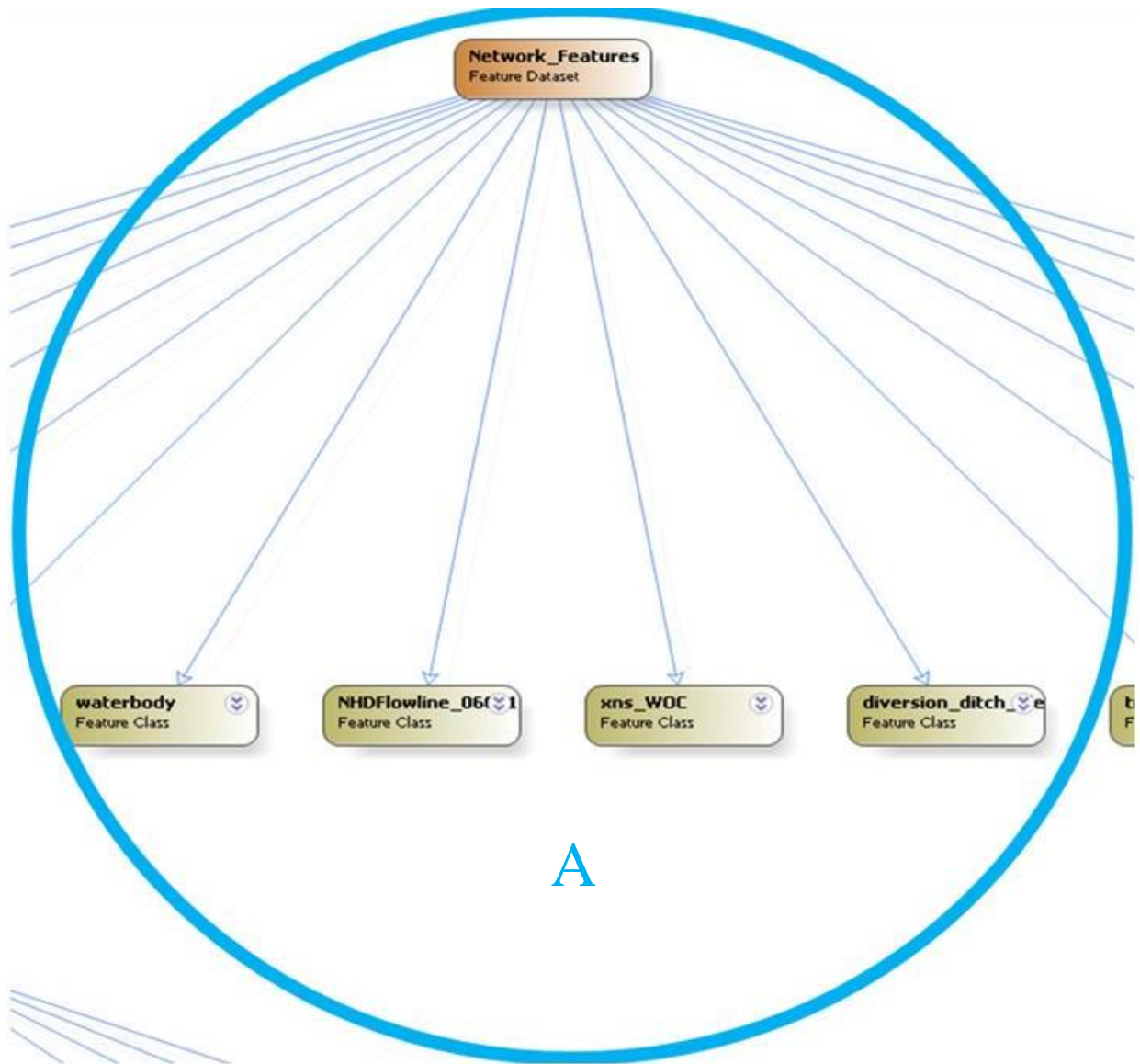
Figure 2. Closer view of circle A in the schema diagram shown in Figure 1 above.
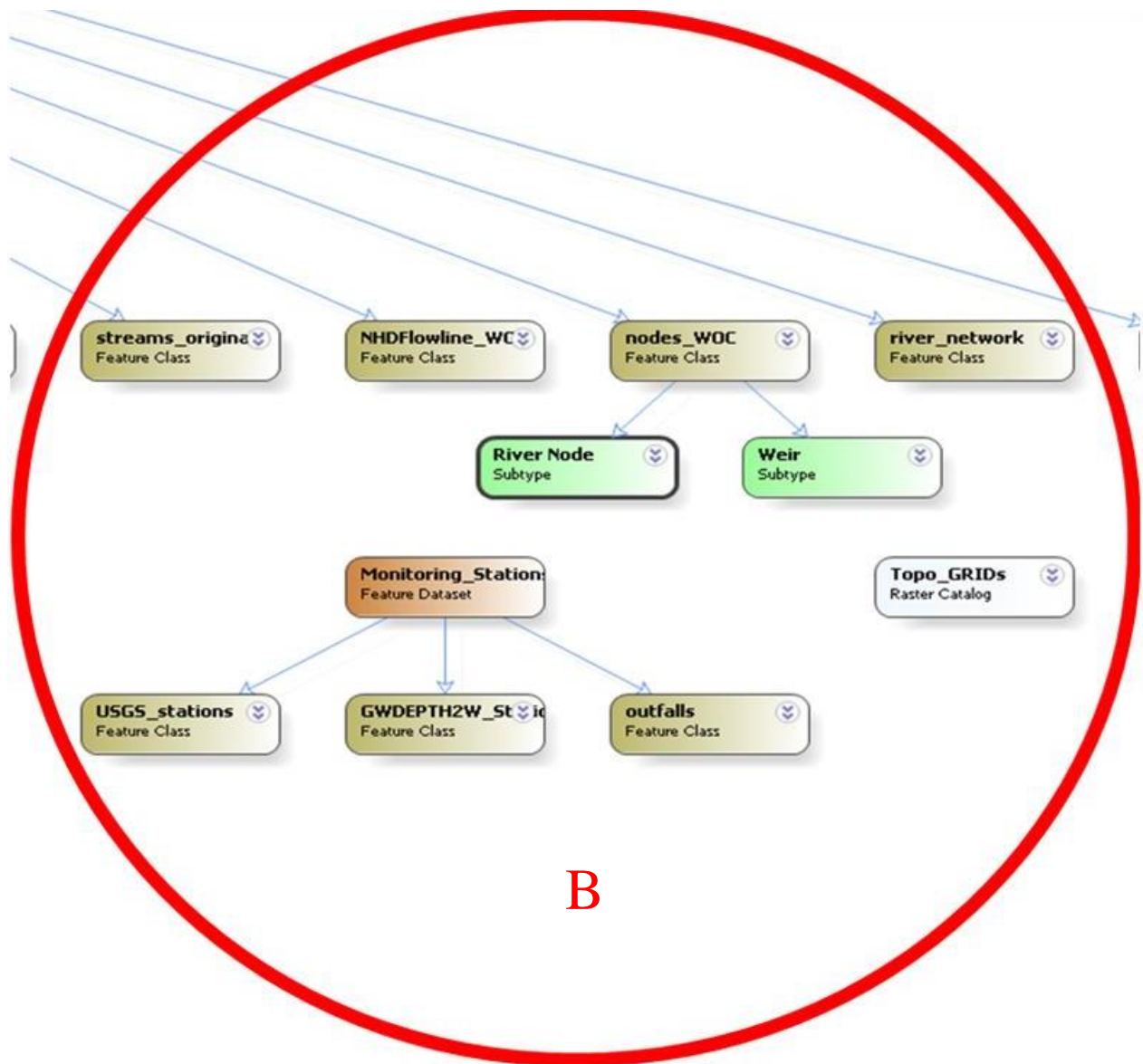
Figure 3. Closer view of circle B in the schema diagram shown in Figure 1 above.
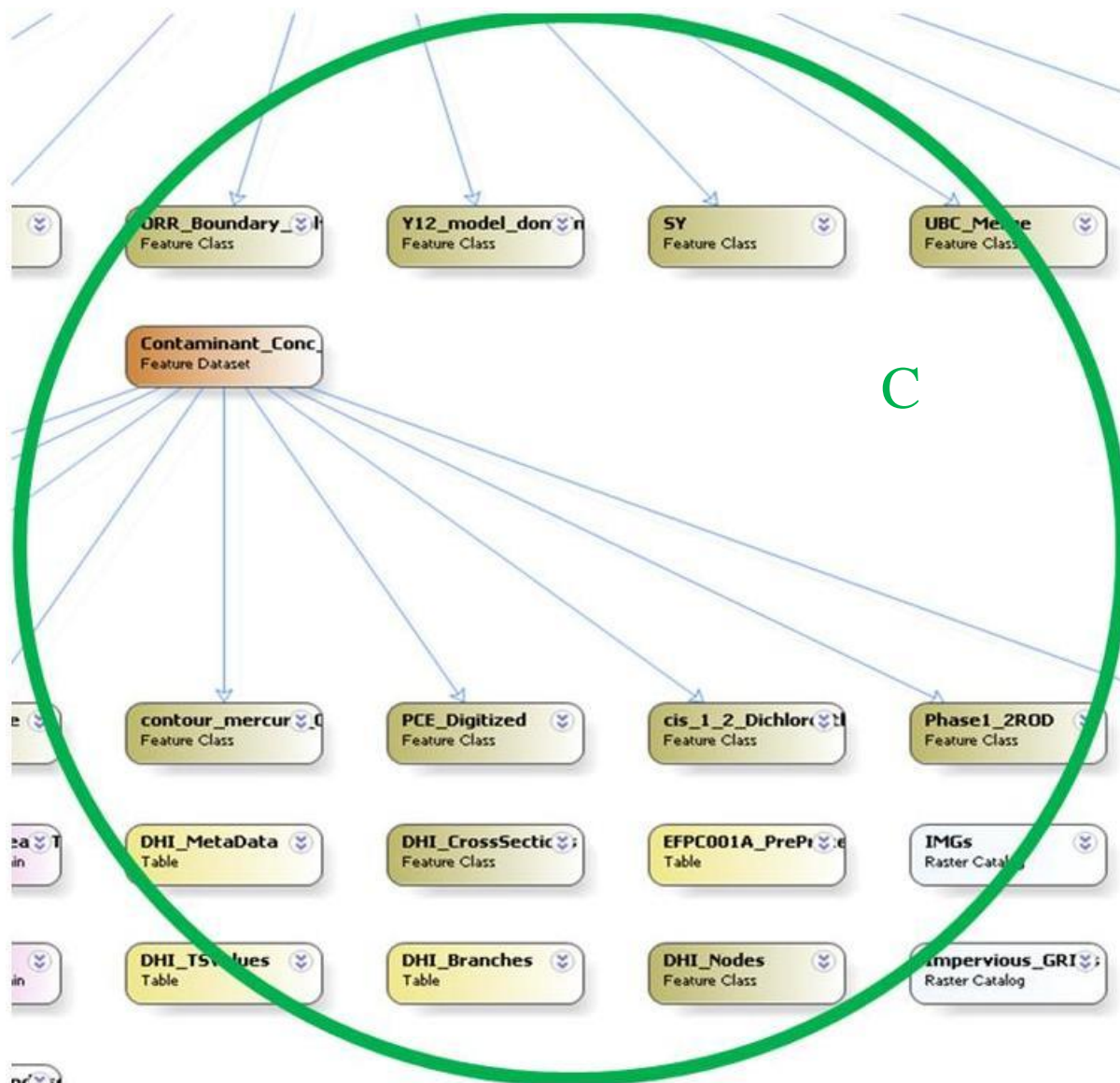
Figure 4. Closer view of circle C in the schema diagram shown in Figure 1 above.

The essence of this task is to provide modelers with ease of access to model data and pre-processing capability for files, like those featured in Figures 1-4, to be used as input into the MIKE SHE/11 models. Post-processing of model results is also possible for analysis, use in reports and map production. Simple tasks such as retrieving data from the geodatabase; pre-processing the data; exporting for use in hydrological model development; subsequent import and post-processing of model data; data analysis; and production of graphs, maps and reports are repetitive but necessary. The use of Python scripts and ArcGIS ModelBuilder assists in automating these tasks which saves time and can facilitate batch processing of this data. One of

the objectives of this task, therefore, was to also develop a reusable GIS tool, which can iterate over each set of the spatial input data parameters perform geoprocessing actions and calculate statistical parameters.

A toolbox which combines built-in ArcGIS geoprocessing tools coupled with customized Python scripts was also developed specifically for use with the East Fork Poplar Creek (EFPC) model. These tools and scripts automate the query and retrieval of timeseries data, including contaminant flow and transport parameters (e.g. mercury concentration, surface water and groundwater flow, discharge, groundwater level, etc.), from the existing ORR geodatabase. The ArcGIS data model iterates through selected features and exports the results in tabular format. Existing Python scripts can also be incorporated for statistical analysis of the exported data. Once a feature (e.g. GW well or outfall) has been selected, a field attribute such as station name can be used to extract and export all the data for that station in MS Excel or text format. Geoprocessing such as interpolation of the extracted values and generation of raster images for each day in ESRI GRID and TIFF formats can also be automated. ModelBuilder tools and Python scripts are also available in the toolbox to enable the export of maps from an ArcMap document within a specified data frame in PDF format. In addition, ArcGIS ModelBuilder can be used to generate model workflow diagrams which are a great way of documenting and visually representing the geoprocessing tools and scripts being incorporated into the data model as development progresses.

The GIS tool enables analysis of spatial and temporal monitoring data at ORR. It has been developed as a set of GIS toolboxes that use a set of input GIS files. The tool is calibrated to the project's location (i.e. the EFPC model domain) and is scalable and reusable. The GIS toolbox provides equivalent functionality and has capabilities to:

  i.   Add GIS files to ArcMap and create layer files.
 ii.   Select features within a specified area (e.g. the study domain) and zoom to selected features.
iii.   Clip/extract selected features and create new layer file of selected subset.
 iv.   Export clipped feature in format to be used by MIKE SHE/11 model.

v. Export attributes of clipped feature in MS Excel or text format for statistical analysis and generation of graphs and reports.

vi. Export map extent in various formats (e.g. JPEG, TIFF or PDF) for development of reports.

vii. Interpolate timeseries data collected at various monitoring points, generate gridded surfaces, and finally create and export mapped results (as seen in image below).

Work Summary

Work for FY12 has involved the creation of a model using ArcGIS ModelBuilder and Python scripting that automates the process of querying the geodatabase based on specific environmental parameters, performs analyses based on specified algorithms and generates maps with the spatial distribution of computed and observed data. A preliminary literature review was conducted for the use of Python scripting to automate various geoprocessing tasks. Existing and built-in scripts for some of these geoprocesses were also reviewed and in some cases modified to support external query and retrieval of mercury and hydrological model data from the existing ORR geodatabase. ArcGIS ModelBuilder was used to generate and document process flow diagrams. The following are some of the documents, presentations and technical workshops reviewed:

1. ESRI International User Conference Technical Workshops:
   - "Model Builder Advanced Techniques," Scott Murrary, July 2010.
   - "Working with Temporal Data in ArcGIS," David Kaiser, Hardeep Bajwa, July 2010.
2. ESRI Southeast Regional User Group (SERUG) Conference 2010 Technical Workshop:
   - "Intermediate ModelBuilder," Kevin Armstrong.
3. Wikihow: "Creating time-series raster mosaics in ArcGIS 10 for Eye on Earth."
4. "Model Builder Lab," Geoinformatics, Spring 2008, Purdue University Library.
5. "Time-Series Contaminant Interpolation using ArcGIS and Spatial Analyst," Mark K. Petersen, ESRI User Conference Proceedings 2006, Paper 1326.

These resources were used to develop a model using ArcGIS ModelBuilder which utilized customized Python scripts which served to automate the retrieval of groundwater level daily timeseries well data derived from OREIS by date; interpolate the extracted values; and generate raster images for each day in ESRI GRID and TIFF formats. The model also enabled the raster images produced to be stored in a raster catalog archived by date to facilitate visualization and animation of the temporal changes in groundwater levels for the specified study domain over a given timeframe. To date the scripts developed have enabled storage of the raster images

produced in a raster catalog, however, further development is necessary for automated archival of these images by date.

A separate model was also developed, again integrating customized Python scripts into ArcGIS ModelBuilder, to enable the export of maps from an ArcMap document within a specified data frame in PDF format. Python scripts have also been developed enable iteration through selected features followed by export of the results in tabular format. This can be utilized to extract model input and output data contained in the geodatabase such as groundwater level, discharge and mercury concentration. Once the feature (e.g. GW well or outfall) has been selected, a field attribute such as station name is used to extract all the data for that station and export it as a .dbf table which can be opened using MS Excel. The aim is to develop Python scripts to extend and refine the model by enabling data to be extracted from the ORR geodatabase and saved/exported as Excel or .txt files. Development of new or modification of existing Python scripts is also in progress to enable greater functionality such as statistical analysis of the exported data. This will be especially useful for extracting and analyzing timeseries data used in the EFPC model.

As previously mentioned, a toolbox has been implemented to combine the built-in ArcGIS geoprocessing tools coupled with the customized Python scripts that have been developed specifically for use with the East Fork Poplar Creek (EFPC) model. Implementation of this toolbox in ArcGIS uses the scripts to analyze water stages in rivers, flow rates, groundwater levels and water quality data from wells and rivers. The toolbox capabilities are being tested to extract and analyze timeseries data used in the EFPC model. Once the model is completed, a ModelBuilder workflow diagram will be generated and documented.

Review of Free/Open Source GIS Software

A literature and Internet search of easily downloadable free/open source GIS software that can be used for querying the ORR geodatabase online was carried out in order to facilitate accessibility to the hydrological modeling data as well as the generation of maps, graphs and reports. This way project derived data can be more easily shared with other project stakeholders such as DOE personnel and ORR site contractors. Comparative matrices were created by Bradt, 2008 [1], which compare various characteristics of free/open source GIS software packages, some of

which were used to assist in our own selection process taking into account, however, that improvements to some of the software products have been made which enhances some of their capabilities beyond what Bradt outlined in his 2008 analysis.

A similar comparison was also conducted by Zimmer, 2010 [10], who stated "….*keep it simple. If the software is simple to use and understand and provides a few essential functions, then people will readily adopt it.*" Like Zimmer, for the purposes of this project, although several products were reviewed, only software that was easy to find on the Internet, readily available, completely free, and simple to download and install were evaluated in depth. Aside from these characteristics, the software products selected also had to be able to perform certain operations and functions such as basic zooming and panning around the map; selection of features and querying of their attributes which contain project derived data stored in the existing ORR geodatabase; addition or removal of georeferenced vector layers and raster imagery to the map; addition of customized symbology and annotation; development, customization, export and printing of map layouts with the option to insert and edit map elements such as a north arrow, scale, legend, map title, etc. Tools for measuring distance and conducting simple buffer analyses were also a consideration. Although at this point of project development very simplistic operations and functions are required, scalable products were given preferential consideration to provide the option of more sophisticated editing and geoprocessing capabilities in the future, particularly with respect to timeseries and hydrological data. Table 2 below shows the various products reviewed by ARC-FIU.

Products developed by ESRI were the first to be evaluated due to the simple fact that the ORR geodatabase was developed on an ESRI platform using primarily ArcGIS software products and would therefore likely have greatest data compatibility. **ArcReader (v. 10.1)** is a free downloadable application provided by ESRI that allows users to view and print maps that are published as Published Map Files (PMF) by the ArcGIS Publisher extension to ArcView, ArcEditor, and ArcInfo. Although this software had the capability to execute many of the desired operations previously mentioned, it served as more of a data viewer and lacked any sophistication with respect to editing, geoprocessing and analytical capabilities, so was eliminated from further consideration.

Table 2. Free/Open Source GIS Software Reviewed by ARC-FIU

Software	Version	Operating System	Free/Open Source	Website
ArcReader	10.1	Windows Mac OS	Free	http://www.esri.com/software/arcgis/arcreader
ArcGIS Explorer Desktop	2500	Windows Mac OS	Free	http://www.esri.com/software/arcgis/explorer
Quantum GIS	1.8.0	Windows Mac OS Linux	Free/Open Source	http://www.qgis.org/
DIVA GIS	7.5	Windows Mac OS	Free/Open Source	http://www.diva-gis.org/
TatukGIS Viewer	4	Windows	Free	http://www.tatukgis.com/
MapWindow	4.x	Windows	Free/Open Source	http://www.mapwindow.org/
HydroDesktop	1.5	Windows	Free/Open Source	http://hydrodesktop.codeplex.com/
GRASS GIS	6	Windows	Free/Open Source	http://grass.osgeo.org/
Google Earth		Windows Mac OS Linux	Free	http://www.google.com/earth/index.html

ArcGIS Explorer Desktop (v. 2500) on the other hand had far more functionality in the performance of spatial analyses; the integration of various data formats such as geodatabases, shapefiles, KML/KMZ, GPX, and raster formats (JPEG 2000, GeoTIFF, MrSID); addition of layer files and packages created using ArcGIS for Desktop; and addition of coordinate data directly from your spreadsheets. Maps and globes in any projection or coordinate system can also be viewed with an integrated 2D/3D display. Most significantly, however, was the ability to extend the core capabilities of ArcGIS Explorer Desktop using Add-Ins and geoprocessing tools and models published using ArcGIS for Server, and the ability to connect to a wide variety of other available services from ArcGIS for Server, Open Geospatial Consortium, Inc. (OGC) WMS, and GeoRSS feeds. In addition, an advantage of using ArcGIS Explorer is that ESRI provides access to existing ready-to-use basemaps of topography, aerial/satellite imagery, roads, etc. via ArcGIS Online, which can add to the aesthetic quality of the final map product. Maps can then be exported or emailed directly from the application. Photos, reports, videos, and other information can also be added to your maps. ArcGIS Explorer Desktop is also customizable with no programming required. The look and feel, as well as the functionality can be customized to meet specific user needs. There is also an option to create presentation slideshows with the maps you have created, sharing them via live data and services. This feature can be very useful in providing stakeholders with quick and easy-to-assemble presentation material.

The Quantum GIS (QGIS) (v. 1.8.0 'Lisboa') desktop application is a user friendly, cross-platform (Windows, Mac, Linux) open source geographic information system (GIS) licensed under the GNU General Public License (GPL) that offers many GIS functions for data viewing, editing, and analysis and supports numerous vector, raster, and database formats and functionalities. Like ArcGIS Explorer Desktop, this application enables direct viewing of vector and raster data in different formats and projections, including ESRI shapefiles, MapInfo, SDTS and GML data, digital elevation models, aerial photography, Landsat imagery, GRASS locations and mapsets. QGIS mapping tools are also similar in nature to ArcGIS Explorer with capabilities to extend the application's functionality and customize its GUI through various plug-ins. Aside from the standard mapping tools available in most GIS data viewers, additional features such as on-the-fly reprojection, spatial bookmarks, digitizing tools for vector features, a field and raster calculator and GPS tools to import and export GPX format, convert other GPS formats to GPX,

or down/upload directly to a GPS unit are also available. Map features such as a north arrow, scale bar and copyright label are present as well as advanced vector and raster symbology, enabling customization of the final map product which can be published on the internet via the "Export to Mapfile" tool. There are also several readily available spatial analysis tools to conduct map algebra, terrain analysis, hydrologic modeling and network analysis.

DIVA-GIS (v. 7.5) is another free GIS that caters to those who are unable to afford generic commercial GIS software or are unwilling or unable to devote time to learning complicated mapping programs. Like ArcGIS Explorer Desktop, free spatial data is also provided that can be used with DIVA-GIS or other programs. Various vector and raster data types are also supported. DIVA-GIS was created particularly for analysis of biodiversity (i.e. the geographical and ecological distribution of organisms) with features that allow climate data to be extracted and ecological niche modeling to be carried out using the BIOCLIM and DOMAIN algorithms. Like the other map viewers being reviewed, it contained basic mapping tools and elements for map customization (e.g., scale, north arrow, legend, etc.). Maps could also be exported in various formats. It was found that DIVA-GIS, compared to other viewers such as ArcGIS Explorer Desktop, Quantum GIS and TatukGIS, was far more limited in the number of file formats it had the ability to read. In addition, the query and analysis tools were more geared to what the program was specifically created for and lacked some of the functionality desired for the nature of this project.

The **TatukGIS Free Viewer (VWR) (v. 4)** is a scaled down version of the TatukGIS Editor (EDT). It has a simple to use GUI for managing layer properties, legend display, thematic mapping features, spatial and attribute querying, custom labeling, measurement of map features, hyperlinking, and generation of graphs and charts. It features automatic coordinate system recognition of imported layers and on-the-fly reprojection to nearly 3,000 pre-defined coordinate systems. The user interface is also available in 16 different languages. Although it can open and view a very broad range of GIS/CAD vector, raster image, grid files and attribute data, as well as ArcView 3, ArcGIS/ArcMap 9 & 10, ArcExplorer, and MapInfo Professional projects, it does not have data editing, 3D viewing, layer rectification, and scripting/customization capabilities as contained in the TatukGIS Editor version which is not free. The free viewer also does not read ArcGIS File or SDE geodatabase files.

MapWindow GIS (v. 4.x) [12] is a free, open source GIS data viewer with built-in features that allow advanced users and developers to add additional functionality (models, special viewers, hot-link handlers, data editors, etc.) via existing plug-ins or development of their own which can be done either in notepad or an XML editor. It therefore functions as a mapping tool, a GIS modeling system and a GIS application programming interface (API) all accessible through a user friendly GUI, which can be very useful in engineering research and project software development and customization, without the need to purchase expensive GIS systems or have an extensive GIS technical background. The MapWindow GUI possesses the standard GIS data visualization features and is able to edit shapefiles and attribute tables and convert data to various formats. A broad range of spatial and text data formats are supported such as shapefiles, GeoTIFFs, ESRI ArcInfo ASCII and binary grids, however, like the TatukGIS Free Viewer, it does not read ArcGIS File or SDE geodatabase files.

HydroDesktop (v. 1.5) [13] is a free/open source GIS developed by the Consortium of Universities for the Advancement of Hydrologic Science, Inc. (CUAHSI) Hydrologic Information System (HIS) primarily for downloading, visualizing, and analyzing hydrologic and climate data registered with the CUAHSI HIS. Hydrologic and climate timeseries data can be retrieved from a metadata catalog maintained at HIS Central and/or individual HydroServers hosting WaterOneFlow web services. Users may also access data through third party data analysis applications that have the ability to read data from a relational database (e.g. R and MATLAB). Downloaded data can then be stored in a desktop data repository which follows a relational database schema. HydroDesktop has a similar interface to most desktop GIS programs with additional built-in tools and forms specifically for visualization and analysis of timeseries data. Topographical and aerial image basemaps are also readily available. HydroDesktop is based on the MapWindow GIS system and like MapWindow GIS, it has an extensible interface for developers to create custom data analysis and visualization plug-ins. Several GIS vector, raster, and image data types are supported, and functionality for navigating the map and customizing the map display are included. Similar to MapWindow GIS and the TatukGIS Free Viewer, HydroDesktop does not read ArcGIS File or SDE geodatabase files. Other GIS tools and features for searching, importing, and managing data are also available. An R statistics plug-in that supports manipulation and transformation of data, statistical analysis, and modeling using

data from the HydroDesktop database is also a useful feature as well as a plug-in for exporting selected observed data from the local desktop data repository to a delimited text file.

The Geographic Resources Analysis Support System, otherwise known as **GRASS GIS (v. 6)**, is a geographic information system used for data management, image processing, graphics production, spatial modeling, and visualization of various data types and formats. It is also a free/open source product released under the GNU General Public License like QGIS described above. It consists of a wide range of applications and is used in many different areas of scientific research in academic and commercial settings around the world by several governmental agencies including NASA, NOAA, USDA, DLR, CSIRO, the National Park Service, the U.S. Census Bureau, USGS, and many environmental consulting companies. GRASS GIS contains hundreds of modules for developing and managing new as well as existing data and rendering maps and images on monitor and paper. It has an intuitive graphical user interface as well as a command line syntax with the unique ability to interface with printers, plotters, digitizers, and databases. A broad range of data formats and database interfaces such as DBF, SQLite, PostgreSQL, mySQL, ODBC are supported and an extensive suite of vector and raster geoprocessing and statistical analysis tools for environmental applications such as erosion modeling, landscape structure analysis, solution transport, and watershed analysis are provided via built-in modules or plug-ins. The GRASS GIS 6 release has also introduced a new topological 2D/3D vector engine and support for vector network analysis. Maps can also be exported in image, Postscript or HTML formats.

Google Earth provides a platform for visualization and analysis of spatial data accessible over the Internet using freely available software and is a step towards decreasing programming and analytic costs of remediation. Use of the Google Earth plug-in and APIs allow display of maps and 3D models that will be accessible based on user security credentials. Observed and computed timeseries data could be made available through this readily available online mapping interface to facilitate the sharing and dissemination of model output data. Data such as contaminant concentrations and surface/subsurface flow rates can be displayed through development of a customized application which utilizes existing Google Earth APIs.

In summary, ArcGIS Explorer Desktop was found to be the most popular and most commonly used free GIS software, however, Quantum GIS was found to be a good competitor and another good option for its extremely user friendly interface, ability to read a broad range of GIS spatial data and text files, feature and attribute editing capabilities, map customization features and the overall quality of the final exported map product. Google Earth also provides a commonly used platform for visualization and analysis of spatial data with built-in base maps and a user friendly interface. Use of the Google Plug-In coupled with an existing "Time" API also facilitates visualization of timeseries data. MapWindow GIS and Tatuk GIS were also two useful data viewers, but had less functionality than ArcGIS Explorer Desktop and Quantum GIS. HydroDesktop, although limited for the same reasons as MapWindow GIS which it is based upon, is also a good consideration if users want to ability to access, visualize and analyze hydrologic and climate data, which may prove useful given the nature of the hydrological modeling work that this project is based upon. GRASS GIS, although extremely sophisticated in its level of functionality especially for a free product, was not as straightforward to set up as some of the other free/open source viewers reviewed and may have the potential to consume valuable time which can deter some end users. ArcReader and DIVA-GIS both lacked the level of desired functionality and were therefore omitted from further consideration.

Future Work

During FY12, the capabilities of the geodatabase developed by FIU to support the hydrological modeling work being performed for ORR were extended by creating a model using ArcGIS ModelBuilder and Python scripting to automate the process of querying the existing EFPC geodatabase, and to export data that can be used for statistical analysis and the generation of maps, graphs and reports. During FY13, the existing database will be updated with more recent ORR site monitoring data available from various sources including OREIS, USGS, NRCS STATSGO or SSURGO soil databases, and the U.S. EPA MRLC or NALC land cover databases. Training will also be provided to students on how to update and query the existing geodatabase within the ArcGIS environment. FIU will also improve the complexity of the existing Python scripts developed during FY12 and develop additional customized scripts which will serve to enhance database querying capabilities. FIU will provide a library of scripts which can be coupled with other existing libraries used for mathematics, science, and engineering such as NumPy and SciPy, to perform statistical analyses and which can be applied to similar databases used at other DOE sites. ArcGIS ModelBuilder will be used to document ArcGIS model workflow diagrams for reporting purposes. The existing geodatabase structure developed for the EFPC modeling work at ORR will also be used to create similar databases to support modeling work being conducted at the Moab Site, Utah and other EM cleanup sites that may be included in related tasks as pilot studies.

An investigation of downloadable freeware to facilitate online querying of the database was also conducted to determine methods by which project derived data can be more easily shared with other project stakeholders such as DOE personnel and ORR site contractors. FIU suggested that observed and computed timeseries data could be made available through a readily available online mapping interface to facilitate the sharing and dissemination of model output data. Data such as contaminant concentrations and surface/subsurface flow rates can be displayed through development of customized applications which utilize available software APIs. During a project review televideoconference between FIU and DOE held 04/30/13, DOE-HQ expressed interest regarding the security aspect of sharing the geodatabase with emphasis placed on the fact that the integrity of the data would need to be maintained and the cyber issues understood in addition to the software issues. Sharing via the web requires proper implementation of security protocols in

terms of sharing the geodatabase and ensuring integrity of the data. FIU clarified that in addition to the FIU-ARC firewalls and the built-in software and hardware security protocols, all computers must adhere to the FIU's University Technology Services (UTS) Security and IT Policies which are outlined in detail at http://it.fiu.edu/security/index.shtml. Security measures such as password protection can also be implemented. FIU also added that they would work with DOE to develop a security strategy for sharing the database within DOE or, if the issue arose, with an outside party if necessary.

References

1. Bradt, S. (2008). A comparison of open source or free GIS software packages. The New Hampshire Geospatial Extension Program, University of New Hampshire (UNH) Cooperative Extension.
2. Castle, E. (2003). Geodatabases in Design: A Floodpain Analysis of Little Kitten Creek. Thesis. Brigham Young University.
3. ESRI International User Conference Technical Workshop: "ModelBuilder Advanced Techniques," Scott Murrary, July 2010.
4. ESRI International User Conference Technical Workshop: "Working with Temporal Data in ArcGIS," David Kaiser, Hardeep Bajwa, July 2010.
5. ESRI Southeast Regional User Group (SERUG) Conference 2010 Technical Workshop: "Intermediate ModelBuilder," Kevin Armstrong.
6. "ModelBuilder Lab," Geoinformatics, Spring 2008, Purdue University Library.
7. Robayo, O. and D. Maidment, 2005. Map to Map: Converting a NEXRAD Rainfall Map into a Flood Inundation Map. Center for Research in Water Resources, University of Texas at Austin, CRWR Online Report 05-1.
8. Rosenzweig, I. and B. Hodges, 2011. A Python Wrapper for Coupling Hydrodynamic and Oil Spill Models. Center for Research in Water Resources, University of Texas at Austin, CRWR Online Report 11-09. Submitted to Texas General Land Office Oil Spill Prevention & Response. FY 2011 Report under Contract No. 10-097-000-3928, October 31, 2011.
9. "Time-Series Contaminant Interpolation using ArcGIS and Spatial Analyst," Mark K. Petersen, ESRI User Conference Proceedings 2006, Paper 1326.
10. Wikihow: "Creating time-series raster mosaics in ArcGIS 10 for Eye on Earth."
11. Zimmer, R. (2010). GIS Matters: Something for Nothing. The American Surveyor, Vol 7 No 8.
12. Ames, D.P., Michaelis, C.D., Anselmo, A., Chen, L. and H. Dunsford. (2008). MapWindow GIS. S. Shekhar, H. Xiong (Eds.) Encyclopedia of GIS. Springer, New York, pp. 633–634.
13. Ames, D.P., Horsburgh, J.S., Cao, Y., Kadlec, J., Whiteaker, T. and D. Valentine. (2012). HydroDesktop: Web Services-Based Software for Hydrologic Data Discovery, Download, Visualization, and Analysis. Environmental Modelling & Software. Vol 37, pp 146-156.

Appendix

Representative Python Scripts

mainORR.py

> *Imports the various Python libraries (in-built and customized) containing the scripts used for geoprocessing. Also sets the path to the data.*

```
# ---------------------------------------------------------------------------
# Read the parameters
# Import arcpy and set path to data
# ---------------------------------------------------------------------------

import arcpy
import os
import time
import libFunctions
from libFunctions import *
```

----

def main():

> *Specifies the input and output parameters and the geoprocessing actions to be carried out.*

```
  g.DEBUG = 0
  g.ARGIN = arcpy.GetParameterAsText(0)
  g.ARGA = arcpy.GetParameterAsText(1)
  g.ARGOUT = arcpy.GetParameterAsText(2)

  if g.ARGIN == '':
    iniInputFiles()
    ORR_clipSelected()

if __name__ == "__main__":
  main()
```

libFunctions.py

> *A customized library of Python scripts developed to automate geoprocessing activities that specifically support hydrological modeling work.*

----

def iniInputFiles():

> *Gets the specified parameter by its index position from the list of parameters.*

```
        result = []

## ARGIN = arcpy.GetParameterAsText(0)
        g.ARGIN =
"Z:/ORR_GIS/DOE_ORR_GeodB/Model_Builder/Model_Scripts/ORR_Project/DATA/TimeSe
ries.mdb"

## ARGA = arcpy.GetParameterAsText(1)
        g.ARGA =
"Z:/ORR_GIS/DOE_ORR_GeodB/Model_Builder/Model_Scripts/ORR_Project/DATA/TimeSe
ries.mdb/Model_Features/domain_EFPC"

## ARGOUT = arcpy.GetParameterAsText(2)
        g.ARGOUT =
"Z:/ORR_GIS/DOE_ORR_GeodB/Model_Builder/Model_Scripts/ORR_Project/DATA/TimeSe
ries.mdb"
```

----

def ORR_clipSelected():

> *Iterates the process of clipping features over a selected area (EFPC domain).*

```
        CLIP_FEATURE = g.ARGA
        print "Iterating Selected Area:", CLIP_FEATURE, "\n"
        for inN in g.LVARN:
                print inN
                clipArea(inN, CLIP_FEATURE)
```

----

def clipArea(inN, ATLASSHEET):

> *Creates a unique scratch path name; Creates a temporary copy in memory; Checks for*
>
> *existence of data before deleting; Clips features; Lists the fields in output feature.*

```
        sW = arcpy.env.scratchWorkspace = "in_memory"
        arcpy.overwriteOutput = True

        inF = g.ARGIN + "/" + inN
        outF = g.ARGOUT + "/CLIP_" + inN

  ## clipping is first used
  ##desc = arcpy.Describe(inF)
  ##oidName = str(desc.OIDFieldName)
```

```
  ##FTYPE = desc.ShapeType
  ## create temp copy in memory
  ##TMPFC = arcpy.CreateFeatureclass_management("in_memory","TMP",FTYPE,inF)
      TMP = sW + "/TMP"
  ##TMP = g.ARGOUT + "/TMP.shp"
        if arcpy.Exists(TMP):
     arcpy.Delete_management(TMP)
##   try:
##      arcpy.Delete_management(TMP)
##   finally:
##      print ""
       print " clipping: ", inF
       arcpy.Clip_analysis(inF,ATLASSHEET,TMP,"")
       ##output_fields = arcpy.ListFields(TMP)
       fieldnamesIN = [f.name for f in arcpy.ListFields(inF)]
       fieldnamesOUT = [f.name for f in arcpy.ListFields(TMP)]
       print "file IN=" , fieldnamesIN, " file OUT=" , fieldnamesOUT
  ## preferred use is to append the data to the existing file
       print " appending: ", outF
       if arcpy.Exists(outF):
               arcpy.Append_management(TMP,outF,"TEST","","")
       else:
               arcpy.CopyFeatures_management(TMP, outF)

  ##arcpy.Union_analysis (TMP, outF)
       if arcpy.Exists(TMP):
               arcpy.Delete_management(TMP)
       arcpy.AddMessage(inN + " Done...")
```
----

def ListFeatureClasses():

> *Sets the workspace for the ListFeatureClass function; Uses the ListFeatureClasses function to return a list of shapefiles; Copies shapefiles to a geodatabase.*

```
# Sets the workspace for the ListFeatureClass function
arcpy.env.workspace =
"Z:\ORR_GIS\DOE_ORR_GeodB\Model_Builder\Model_Scripts\ORR_Project\DATA\INPUT
DATA"

# Uses the ListFeatureClasses function to return a list of
#  shapefiles.
featureclasses = arcpy.ListFeatureClasses()

# Copy shapefiles to a geodatabase
for fc in featureclasses:
```

```
    arcpy.CopyFeatures_management(
        fc,
os.path.join("Z:\ORR_GIS\DOE_ORR_GeodB\Model_Builder\Model_Scripts\ORR_Project\D
ATA\TimeSeries.mdb",
                os.path.splitext(fc)[0]))
```
----

def AddLayer():

> *Adds a new layer from a layer (.lyr) file on disk and places it at the bottom of the data frame*
>
> *called EFPC_Data.*

```
import arcpy
mxd =
arcpy.mapping.MapDocument(r"Z:\ORR_GIS\DOE_ORR_GeodB\Model_Builder\Model_Scri
pts\ORR_Project\MAP.mxd")
df = arcpy.mapping.ListDataFrames(mxd, "EFPC_Data")[0]
addLayer =
arcpy.mapping.Layer(r"Z:\ORR_GIS\DOE_ORR_GeodB\Model_Builder\Model_Scripts\ORR_
Project\DATA\INPUTDATA\stations_OREIS_mercury.lyr")
arcpy.mapping.AddLayer(df, addLayer, "BOTTOM")
mxd.saveACopy(r"Z:\ORR_GIS\DOE_ORR_GeodB\Model_Builder\Model_Scripts\ORR_Proje
ct\DATA\OUTPUTDATA\ORR_Project.mxd")
del mxd, addLayer
```

TS_InterpolateToRaster_Script_080912.py

> *Interpolates GW elevation timeseries data from wells within Y-12 area to create rasters by*
>
> *date and exports map in PDF format from ArcMap document.*

```
# -------------------------------------------------------------------------
# TS_InterpolateToRaster_Script_080912.py
# Created on: 2012-08-09 09:40:35.00000
#   (generated by ArcGIS/ModelBuilder)
# Description: Interpolates GW elevation timeseries data from wells within Y-12 area to
create rasters by date and exports map in PDF format from ArcMap document.
# -------------------------------------------------------------------------

# Import arcpy module
import arcpy

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

# Load required toolboxes
```

```
arcpy.ImportToolbox("Model Functions")
arcpy.ImportToolbox("C:/Documents and Settings/lawrence/Application
Data/ESRI/Desktop10.0/ArcToolbox/My Toolboxes/ORR_GeodB.tbx")

# Local variables:
GW_ELEV_TS = "GW_ELEV_TS"
ORR_GeodB_Scratch_mdb = "C:\\Documents and Settings\\lawrence\\My
Documents\\ArcGIS\\ORR_GeodB_Scratch.mdb"
RasterSeries__2_ =
"Z:\\ORR_GIS\\DOE_ORR_GeodB\\Model_Builder\\TimeSeries.mdb\\RasterSeries"
Model_Builder_Testing_mxd =
"Z:\\ORR_GIS\\DOE_ORR_GeodB\\Model_Builder\\Model_Builder_Testing.mxd"
I_GW_ELEV_TS_DATE_ = "I_GW_ELEV_TS_DATE_"
GW_ELEV_n_ = "C:\\Documents and Settings\\lawrence\\My
Documents\\ArcGIS\\ORR_GeodB_Scratch.mdb\\GW_ELEV%n%"
GW_ELEV_n__tif =
"Z:\\ORR_GIS\\DOE_ORR_GeodB\\Model_Builder\\Raster_TS_Outputs\\GW_ELEV%n%.tif
"

# Process: Iterate Feature Selection
arcpy.IterateFeatureSelection_mb(GW_ELEV_TS, "DATE_ #", "false")

# Process: Natural Neighbor
arcpy.gp.NaturalNeighbor_sa(I_GW_ELEV_TS_DATE_, "GW_ELEV", GW_ELEV_n_,
"0.222827600002289")

# Process: Workspace To Raster Catalog
arcpy.WorkspaceToRasterCatalog_management(ORR_GeodB_Scratch_mdb,
RasterSeries__2_, "NONE", "NONE")

# Process: Copy Raster
arcpy.CopyRaster_management(GW_ELEV_n_, GW_ELEV_n__tif, "", "", "", "NONE", "NONE",
"")

# Process: ExportToPDF Script
arcpy.gp.toolbox = "C:/Documents and Settings/lawrence/Application
Data/ESRI/Desktop10.0/ArcToolbox/My Toolboxes/ORR_GeodB.tbx";
# And replace arcpy.gp.ExportToPDF(...) with arcpy.ExportToPDF_ALIAS(...)
arcpy.gp.ExportToPDF()
```

SelectData_ByAttribute.py

> *Selects GW wells by station name and exports all associated attributes in a .dbf table.*

```
# ----------------------------------------------------------------------
```

```
# SelectData_ByAttribute.py
# Created on: 2013-01-09 10:49:53.00000
#   (generated by ArcGIS/ModelBuilder)
# Usage: SelectData_ByAttribute <Value> <GW_ELEV_TS> <GW_ELEV_TS__2_>
# Description:
# This model selects GW wells by station name and exports all associated attributes in a
.dbf table.
# ---------------------------------------------------------------------------

# Import arcpy module
import arcpy

# Load required toolboxes
arcpy.ImportToolbox("Model Functions")

# Script arguments
Value = arcpy.GetParameterAsText(0)
if Value == '#' or not Value:
    Value = "1" # provide a default value if unspecified

GW_ELEV_TS = arcpy.GetParameterAsText(1)
if GW_ELEV_TS == '#' or not GW_ELEV_TS:
    GW_ELEV_TS = "GW_ELEV_TS" # provide a default value if unspecified

GW_ELEV_TS__2_ = arcpy.GetParameterAsText(2)
if GW_ELEV_TS__2_ == '#' or not GW_ELEV_TS__2_:
    GW_ELEV_TS__2_ = "GW_ELEV_TS" # provide a default value if unspecified

# Local variables:
I_GW_ELEV_TS = GW_ELEV_TS__2_
QueryTable_Value__dbf = I_GW_ELEV_TS
QueryTable_Value__Statistics_dbf = QueryTable_Value__dbf

# Process: Select Layer By Attribute
arcpy.SelectLayerByAttribute_management(GW_ELEV_TS, "NEW_SELECTION", "")

# Process: Iterate Feature Selection
arcpy.IterateFeatureSelection_mb(GW_ELEV_TS__2_, "", "false")

# Process: Copy Rows
arcpy.CopyRows_management(I_GW_ELEV_TS, QueryTable_Value__dbf, "")

# Process: Summary Statistics
arcpy.Statistics_analysis(QueryTable_Value__dbf, QueryTable_Value__Statistics_dbf,
"GW_ELEV MEAN;GW_ELEV MIN;GW_ELEV MAX;GW_ELEV RANGE;GW_ELEV
STD;GW_ELEV COUNT", "")
```

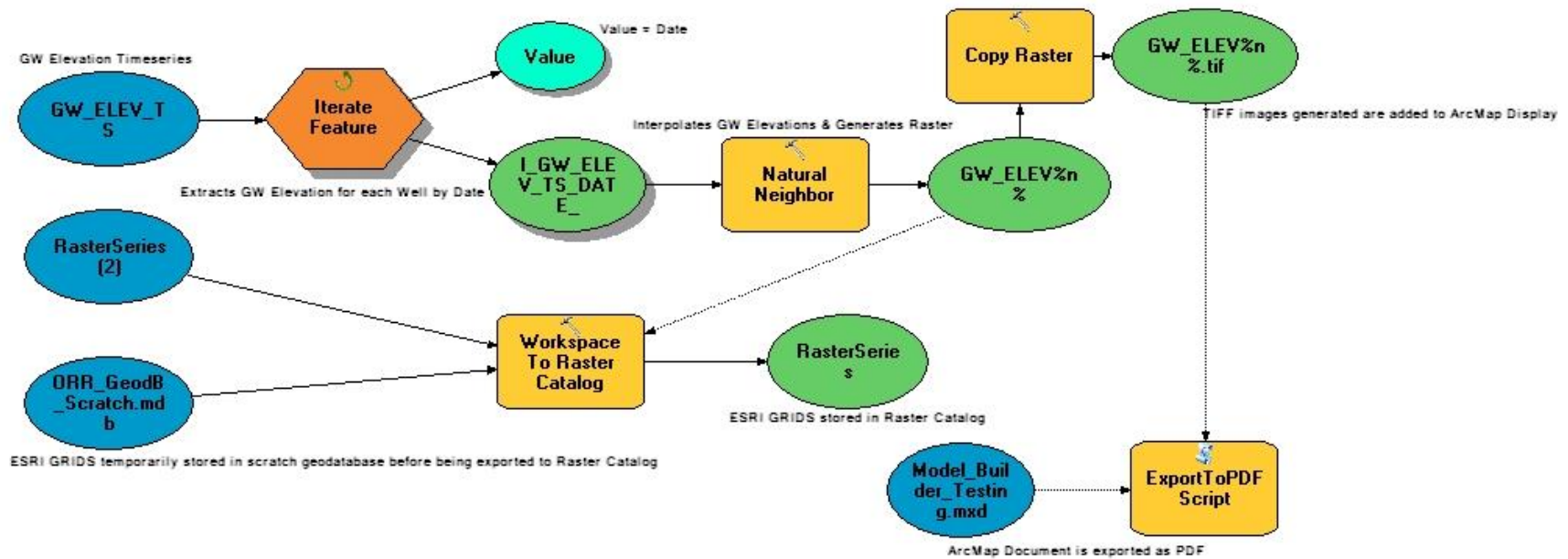**Sample ArcGIS ModelBuilder Process Workflow Diagram**

Figure 5. Process workflow diagram created using ArcGIS ModelBuilder.

This model interpolates groundwater elevation timeseries to create rasters by date and exports maps in PDF format.
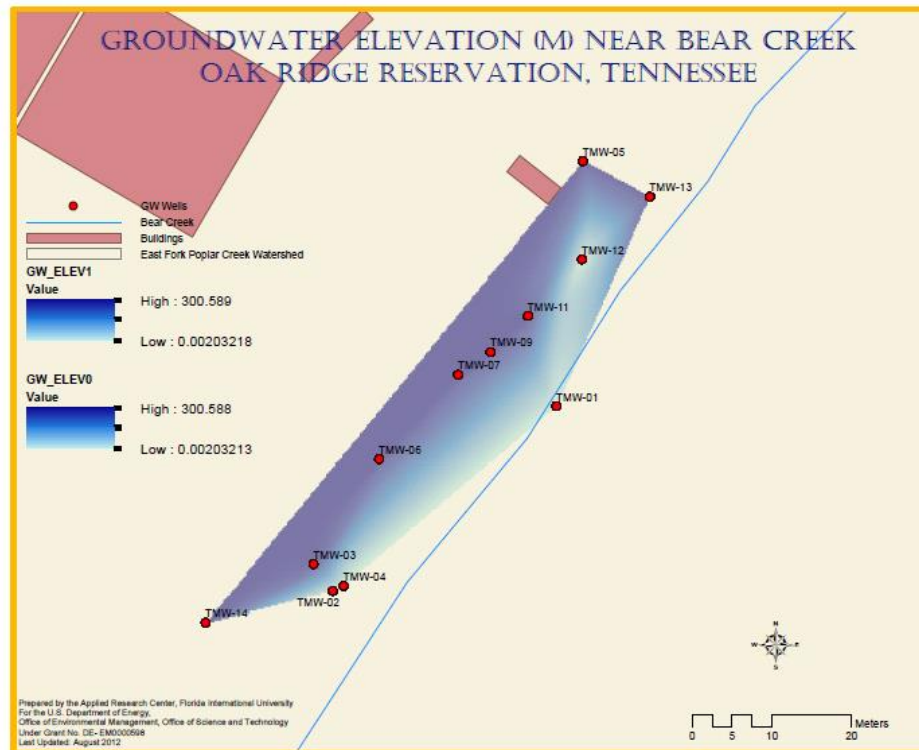
Figure 6. Map generated in PDF format following interpolation of GW elevation data.