Extending the Ecosystem Management Decision Support System to Enable Data Migration

by

Kenneth Crouse

A RESEARCH PAPER

submitted to

THE GEOSCIENCES DEPARTMENT

in partial fulfillment of the
requirements for the
degree of

MASTER OF SCIENCE
GEOGRAPHY PROGRAM

December 2003

directed by
Dr. A. Jon Kimerling

Acknowledgements

Thanks to Dr. Kimerling for always being there with sound advice and encouragement and for keeping an open door policy in his office across the hall that allowed me to keep up with the latest Geosciences news and happenings without leaving my office. I am grateful to Dr. Wright for allowing me to participate in some very interesting projects and work with several exceptional people in the process. A summer in American Samoa wasn't bad either! Thanks to Dr. Rosenfeld, who demonstrated early in my Geosciences days that you can be a geographer and still have fun, even while doing geography. Special thanks to Keith Reynolds of the Forestry Sciences Laboratory for being open enough to share his software and his expertise with someone who was initially an unknown quantity. His support gave me the opportunity to do the type of work I really wanted to do, and he kept me grounded in reality when I started heading for blue sky. Finally, I want to thank Dave Hohler and the people with NRIS Tools for providing me with a place to work and making sure I had everything I needed to do the job.

Table of Contents

Introduction 1

Ecosystem Management 3

The nature of ecosystem management challenges 3

Requirements for solutions 5

Application of Decision Support Systems 7

Statement of Problem 9

The Ecosystem Management Decision Support system 9

Need for data migration 21

Approach Taken 22

Results 25

Conclusions 30

References 32

Appendix: Visual Basic code 34

Figures

Figure 1. Components of EMDS system 11

Figure 2. Chewaucan Basin study area 14

Figure 3. Knowledge base in Network Window 15

Figure 4. Graph of relative influence scores 16

Figure 5. Map created for bank stability Truth Values 17

Figure 6. Graph generated by Priority Analyst 20

Figure 7. Data Frames for chewKMR project 26

Figure 8. Dynamic map layers for Reach scenario 27

Extending the Ecosystem Management Decision Support System to Enable Data Migration

ABSTRACT: *In this work software is developed to take an ecosystem management tool used by the US Forest Service, the Ecosystem Management Decision Support (EMDS) system, and enhance its utility by capturing the results of an ecosystem assessment and saving them in a format that facilitates importation into other management tools, such as planning applications. EMDS is written as an ArcMap extension, and the new software runs on top of EMDS to extract the desired data without the need to modify the existing code. The capture is performed with a very little impact on the user, although the user can exercise some discretion over what data are saved.*

Introduction

In this work software is developed to extend an existing decision support system, the Ecosystem Management Decision Support (EMDS) system, to facilitate data migration of results to other applications. EMDS is implemented as an extension to the ArcMap component of ArcGIS, a commercial Geographic Information System from the Environmental Systems Research Institute (ESRI), and performs ecological analyses using knowledge-based and decision-modeling tools. EMDS was developed at the U.S. Forest Service's Pacific Northwest Research Station and is used in forest resources management within the Agency.

The results of an EMDS analysis include a knowledge base of data specifications for the study area, tables and graphs reflecting the outcomes of decision processes, and maps that spatially depict present conditions in the region and hypothesized conditions under different scenarios. Because EMDS was built as a stand-alone application, its outputs were designed for presentation to the person running the program. As such, some of the output is transitory, appearing only on the display monitor, and other output is stored in a format that is not conducive to sharing across applications.

The present extension to EMDS runs under ArcMap and extracts selected results, organizing them in a hierarchical structure that reflects the structure of the problem. Results are stored in standard formats that can be handled by most applications, and they can be transferred to a Web page for viewing or downloading into other programs. The code is written in Visual Basic, and because it is external to EMDS, no modifications or revisions to the configured EMDS software release were necessary.

The first section of the paper examines the nature of ecosystem management and some of the unique problems that confront it. The need for a regional approach is discussed, and the importance of horizontal and vertical information flow is presented. The role of Decision Support Systems (DSSs) in ecosystem management is reviewed and their evolution is traced. Other types of software tools that are used in an ecosystem management framework are also discussed.

The next section goes into more detail about EMDS, describing its salient features and how it is used. This is followed by a discussion of data migration issues and why they are particularly relevant in the ecosystem management arena.

The particular approach taken to extending EMDS is presented next. The requirements and design choices for the software are discussed, and the data model used to organize EMDS results is presented. The section concludes with an explanation of the data export process that prepares data for transfer to a Web-accessible venue.

The next section looks at an example EMDS session to show how the new software works with EMDS and how the analysis results are processed. The structure and content of the exported data are shown diagrammatically and explained. Finally, some conclusions and suggestions for further work are offered.

Ecosystem Management

The nature of ecosystem management challenges

Ecosystem management has become a key tool for many agencies that are charged with managing our nation's natural resources. Its strategic goal is to develop policies and practices that provide a balance between ensuring the long-term health and sustainability of ecosystems and allowing for the use of natural resources for the benefit of public and private interests (Rauscher, 1999). While there may be widespread agreement that this is an important goal, there is anything but consensus as to what constitutes a proper "balance".

Several factors contribute to making ecosystem management a challenging task. It involves numerous stakeholders, some of which have a long history of contentious relationships. The ecosystems being managed are large, complex systems undergoing continuous change. The scientific understanding of how ecosystems operate under different natural and human-imposed stresses is still very incomplete. The long-term effects of different management practices are not well understood. There are few clear, objective indicators that can be used to measure success. Indeed, the meanings of the terms "health" and "sustainability" used in connection with ecosystem goals are subject to debate.

Another characteristic of ecosystem management that makes it different from most natural resources management approaches of the past is the regional nature of its spatial scope. While many systems have been developed for managing individual administrative units, such as a forest or park, ecosystems are defined based on communities of organisms and can cut across political and administrative boundaries. Because of this, ecosystem management often requires the cooperative efforts of distinct groups that must share data and expertise in making decisions about how to manage ecosystem resources.

The fact that the goals of ecosystem management require looking beyond the near-term consequences of actions means that an extended temporal scope must be considered in devising management solutions. This increases the complexity of the management task because the results of a given management policy may not be fully realized until some time far into the future. In the mean time, changes due to human impacts, natural disturbances, climate changes and sociopolitical shifts may significantly alter the state of the ecosystem. If management is to have the necessary information at its disposal to make decisions in this dynamic framework, the temporal dimension of data must be properly managed. The underlying conditions and rationale contributing to a decision at a particular point in time must be preserved in order to make a meaningful evaluation of the long-term consequences of a decision.

Requirements for solutions

In light of the many challenges presented by ecosystem management, it is clear that an effective approach requires a change from traditional resource management paradigms characterized by being carried out within a single agency, having a narrow scientific focus, and implementing management decisions in a purely top-down manner. Ecosystem management involves scientists from multiple disciplines, often including geologists, hydrologists, foresters, fish and wildlife experts, atmospheric scientists, as well as specialists in information sciences and social scientists. Because ecosystems are not defined by political boundaries, ecosystem problems can involve multiple natural resources organizations. Within a given geographic domain, agencies representing different disciplines, such as forestry and hydrology, may both have jurisdictions within an ecosystem. Also, an ecosystem may cross county or state lines and encompass both rural and urban communities.

In addition to the need to involve scientists and administrators from multiple agencies in addressing ecosystem management problems, other stakeholder groups must be included. Most environmental issues will have people vigorously advocating the preservation of the environment and another group defending economic interests with equal fervor.

Stakeholders can be individual landowners or huge corporations. A certain amount of tension is expected among stakeholder groups simply because they represent different interests, but the divisions can be much more serious when participants have had an adversarial relationship in the past. Viable ecosystem management solutions under these circumstances will not come from science alone; they require a decision-making process that engenders the participation and support of the various stakeholder groups (Fall, *et al.*, 2001).

One implication of the special requirements identified for ecosystem management solutions is that isolated, stand-alone systems will not be sufficient. The nature of ecosystem problems and solutions dictates that data and information must flow readily among the various entities involved in ecosystem management. When multiple scientific disciplines and agencies are participating in the process, they need efficient means of sharing data and expertise. When an ecosystem crosses administrative boundaries, such as adjacent counties, data from each organization must be combined to form a cohesive picture of the ecosystem.

The fact that diverse groups of stakeholders must not only provide input into a successful ecosystem management plan, but must also buy into the plan imposes additional requirements for information sharing. There must be some mechanism in place that allows stakeholders to provide input into the decision-making process. This is especially critical for stakeholders such as private citizens, who would not normally be part of the process. Merely supplying input, however, is not sufficient. Stakeholders need to see how their concerns are taken into consideration in creating a management plan. One way to accomplish this is to have some stakeholders involved at all steps of the process. But even when this is possible, just a few, select representatives can be personally involved. To reach a broader spectrum of stakeholders, there needs to be a means of communicating the scientific findings used in an ecosystem investigation, as well as in the decision-making process. When this level of communication has been absent, the result has often been a management plan that falls short of its goals because of a lack of support from the public and private sectors.

Application of Decision Support Systems

Historically, natural resources management has taken a very hands-on approach. Management decisions were made by a select few individuals based largely on their own personal experience in the field. With this approach, management tended to take a myopic view of resources, and a synergistic view on a regional scale was generally not possible.

The evolution of desktop computing in the 1980's brought about significant changes in the tools and methods of natural resources management. Orders-of-magnitude gains in processor power and falling hardware costs brought the power of mainframe computers to the desktops of scientists and managers. Advances in graphical user interfaces and user-friendly applications software meant that persons without extensive computer backgrounds could effectively use a host of new software tools for management.

One such tool that has had a major impact on the evolving field of ecosystem management is the Geographic Information System (GIS). Geographic Information Systems provide a framework for the integration of diverse spatial data related to natural resources which enables the overlaying of spatial data to reveal relationships and perform analyses not feasible with manual methods (Croteau, *et al.*, 1997). Advances in the collection of ground-based and remotely sensed aerial and satellite data have provided economical coverage over much larger areas than previously possible and in a much timelier manner. In short, GIS has helped make available to the ecosystem manager a very rich source of information on which to base decisions.

Another tool that became available to resource managers as a result of the computer revolution is the Decision Support System. Decision Support Systems were initially developed in the business world to assist managers in utilizing financial and other information in making corporate decisions. As the potential of such systems became recognized, their use spread into other disciplines. In natural resources management, DSS software has helped decision makers manage complex sets of information and precisely define the process of distilling the information into a decision (Fedra, 1995; Varma *et al.*, 2000). This has been particularly valuable in situations where the values and interests of several competing groups must be taken into account and where there is a legal obligation to document that process (Jankowski *et al.*, 2001; Kangas *et al.*, 2000).

Given that ecosystem management problems often involve the assemblage of a wide variety of spatial data over a large area and the complex analysis of the data from multiple, competing viewpoints, it was natural for GIS and DSS technologies to come together at some point (Densham and Goodchild, 1989; Walsh, 2000). The nature of this union can take many forms. A GIS can simply be used as a source of spatial data to be used in a DSS, or a GIS can be used to provide a spatial presentation of the data or the results of an analysis. At a somewhat deeper level of integration, a GIS can be used as an analytical arm of a DSS, performing spatial analyses on various data layers representing key factors affecting the decision. If visualization techniques are employed, a GIS can be used to present a 2- or 3-dimensional depiction of the results of different potential outcomes of the decision process.

Statement of Problem

The Ecosystem Management Decision Support system

The Ecosystem Management Decision Support (EMDS) system was developed at the USDA Forest Service's Pacific Northwest Research Station and released in 1997 as an application framework for knowledge-based decision support in ecological assessments at any geographic scale (Reynolds *et al.*, 2002). The current version, EMDS 3.02, integrates the NetWeaver logic engine for knowledge-based reasoning with the decision-modeling engine of Criterium DecisionPlus (InfoHarvest, 2001) within the ArcMap GIS environment. EMDS is described as a "framework", because the user must provide the knowledge base, data layers, and decision rules for the particular assessment at hand to perform an assessment (Reynolds *et al.*, 2000).

To be able to conduct an assessment in EMDS, the following steps must be carried out:

- A knowledge base is designed that describes how to interpret the available data
- One or more Data Frames are constructed in ArcMap that contain all the spatial data layers required for analysis
- A decision model is developed that reflects management priorities for the landscape features in the assessment

The NetWeaver Developer System is used to construct a knowledge base that describes the logical relationships of ecosystem states and processes (Reynolds, 1999). The ecosystem data used by the knowledge base consist of feature-based spatial data loaded into the layers of an ArcMap Data Frame. EMDS automatically compares the data layers to the requirements of the knowledge base and performs a spatial intersection on multiple layers if necessary.

Criterium DecisionPlus is used to build the decision model, which consists of a structured set of criteria for evaluating features according to management priorities. Two types of models are available, both variations of multicriteria decision analysis methods. The Analytic Hierarchy Process (AHP) model is simpler to use in a stable environment, while the Simple Multiattribute Rating Technique (SMART) model is often used when additional alternatives are likely to be added after the model is built (Saaty, 1994; Kamenetzky, 1982).

Figure 1 shows the key components of EMDS. EMDS is written as an ArcMap extension, so all the functionality of ArcMap for manipulating and displaying spatial data is available to the EMDS routines. Note that the NetWeaver and Criterium DecisionPlus tools are external to EMDS. For information on how to obtain these products, visit the EMDS Web site: http://www.fsl.orst.edu/emds/.

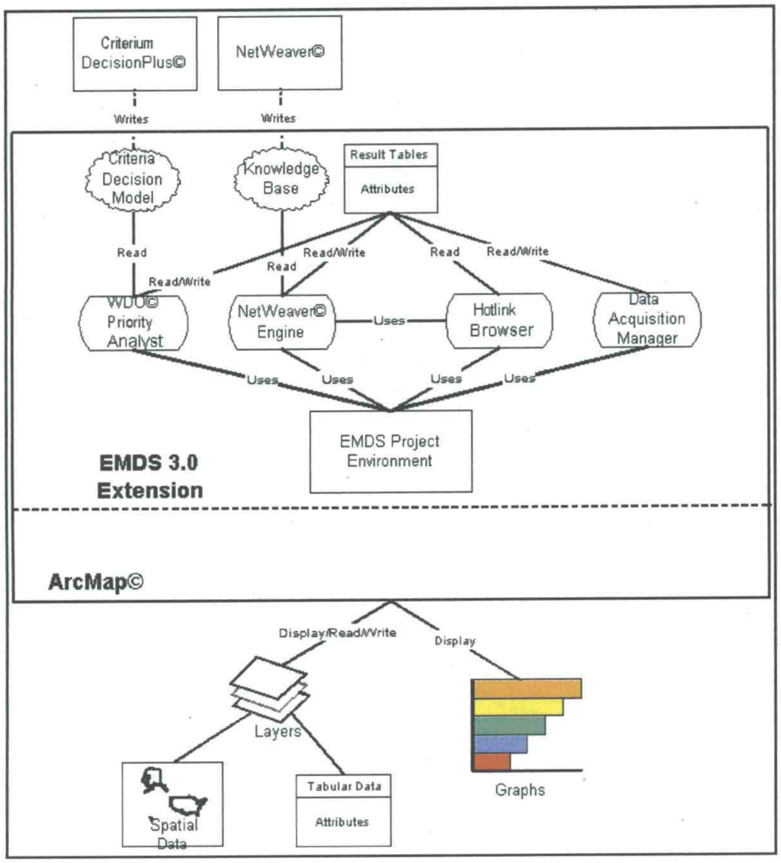


Figure 1. Components of the EMDS system (from Reynolds, 2002)

The EMDS Project Environment provides control and coordination for an EMDS session. The NetWeaver Engine uses the knowledge base and the GIS data layers to perform an analysis that indicates the state of features in the study area, e.g., which stream reaches are in violation of water temperature standards. The Hotlink Browser provides a convenient means of exploring the knowledge base, which is helpful in understanding results presented in tables and maps. The Data Acquisition Manager is used to examine the effects of missing data. This unique feature of EMDS enables the user to determine when additional data need to be collected and helps establish priorities for data collection.

The Priority Analyst enables EMDS to go beyond feature evaluation toward analysis and planning. It uses the decision model built by Criterium DecisionPlus and the results from a landscape analysis to study the effects of priority settings, trade-offs of environmental indicators, and potential effects of management decisions. Priority Analyst results can be viewed as graphical displays or tables and are written to new data layers in ArcMap, so that the spatial nature of the results is immediately observable.

An EMDS application is structured around a project. The two essential parts of a project are the ArcMap map file (.mxd) and the project folder. The map file provides the means for loading an EMDS project into ArcMap. The EMDS project folder is organized according to the project hierarchy. The hierarchy of folders and files that make up a project is given below.

```
Project.mxd (ArcMap map file)
Project folder:
    EmdsProject.xml
    Assessment1 folder:
        EmdsAssessment.xml
        Layers folder (input data layers)
        Assessment1 Shapefile (assessment study area)
        Analysis1 folder:
            EmdsAnalysis.xml
            KnowledgeBase.nw (NetWeaver knowledge base)
            ALIASES.DBF (knowledge base names and aliases)
```

```
                FLAG.DBF (input flag table)
                INF.DBF (output influence table)
                INFRNK.DBF (output influence rank table)
                infsum.dbf (output influence summary table)
                INN.DBF (input data table)
                OUT.DBF (output data table)
                Analysis1 Shapefile (analysis overlay)
                Scenario1 folder:
                        EmdsScenario.xml
                        KnowledgeBase.nw
                        ALIASES.DBF
                        FLAG.DBF
                        INF.DBF
                        INFRNK.DBF
                        infsum.dbf
                        INN.DBF
                        OUT.DBF
                        PAManager.mdb (Priority Analyst Access data base)
                Scenario2 folder: ...
        Analysis2 folder: ...
Assessment2 folder: ...
```

At the top level is the project folder, which contains subfolders for assessments and the project XML file, that describes all project-level data. A project can have one or more assessments, and each assessment has an associated study area. Multiple assessments for a single project may be necessary, for example, when a region needs to be studied at different scales. Each assessment folder contains an optional layers folder for maps, a Shapefile of the study area, and an XML file.

Each assessment consists of one or more analyses. An analysis involves the actual evaluation of landscape features in the study area. Multiple analyses may be necessary when one desires to look at different sets of features in the same assessment, for instance, stream reaches versus sub-basins. An analysis folder contains a Shapefile and an XML file, similar to an assessment, but it also contains a NetWeaver knowledge base for the evaluation of features and a set of tables for storing evaluation results. A Microsoft Access database contains data used by the Priority Analyst. In addition, the analysis folder contains a subfolder for each scenario created during the analysis.

A scenario provides a mechanism for exploring the consequences of various alternative approaches to the management of resources in the study. Within a scenario, the input data may be modified to reflect a hypothetical condition, or the structure of the knowledge base may be altered to indicate different assumptions about the data. The scenario folder contains the results tables similar to an analysis, an XML file, and a Priority Analyst database.

An example of using EMDS for watershed assessment is given by a study of the Chewaucan Basin, which consists of nine subwatersheds as shown in Figure 2.

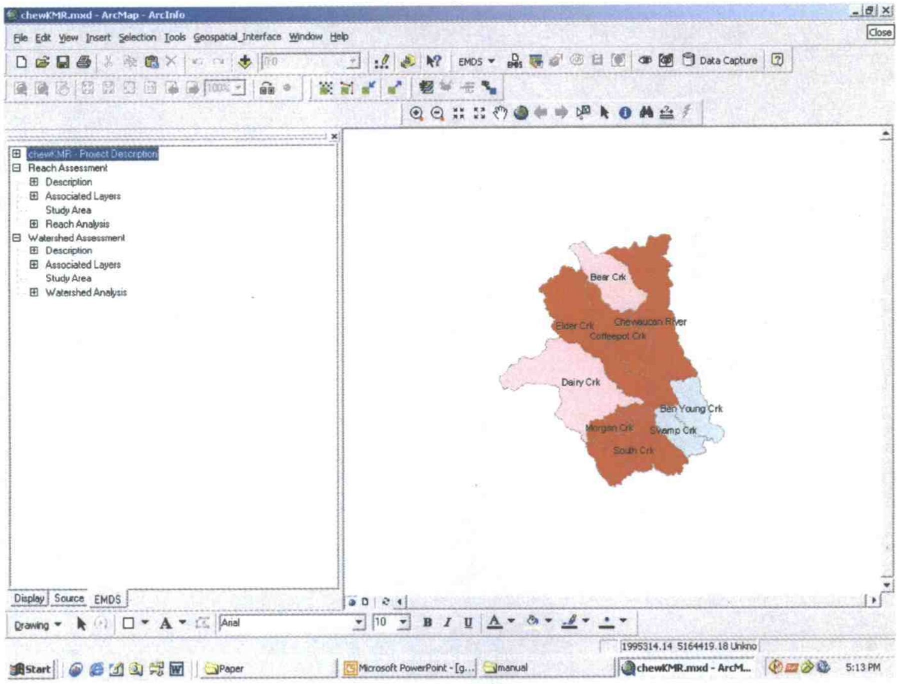


Figure 2. Chewaucan Basin study area

The Chewaucan Basin is a 5th-code hydrologic unit, located on the Fremont National Forest in south-central Oregon. In this project, named chewKMR, two assessments were performed, a fine-scale assessment based on stream reaches and a higher-level assessment based on subwatersheds, as indicated by the EMDS hierarchy in the left panel of Figure 2.

Figure 3 shows a view of the knowledge base as depicted in the NetWeaver Network Window. The logic diagram may be used to examine which attribute values caused a particular state to appear.

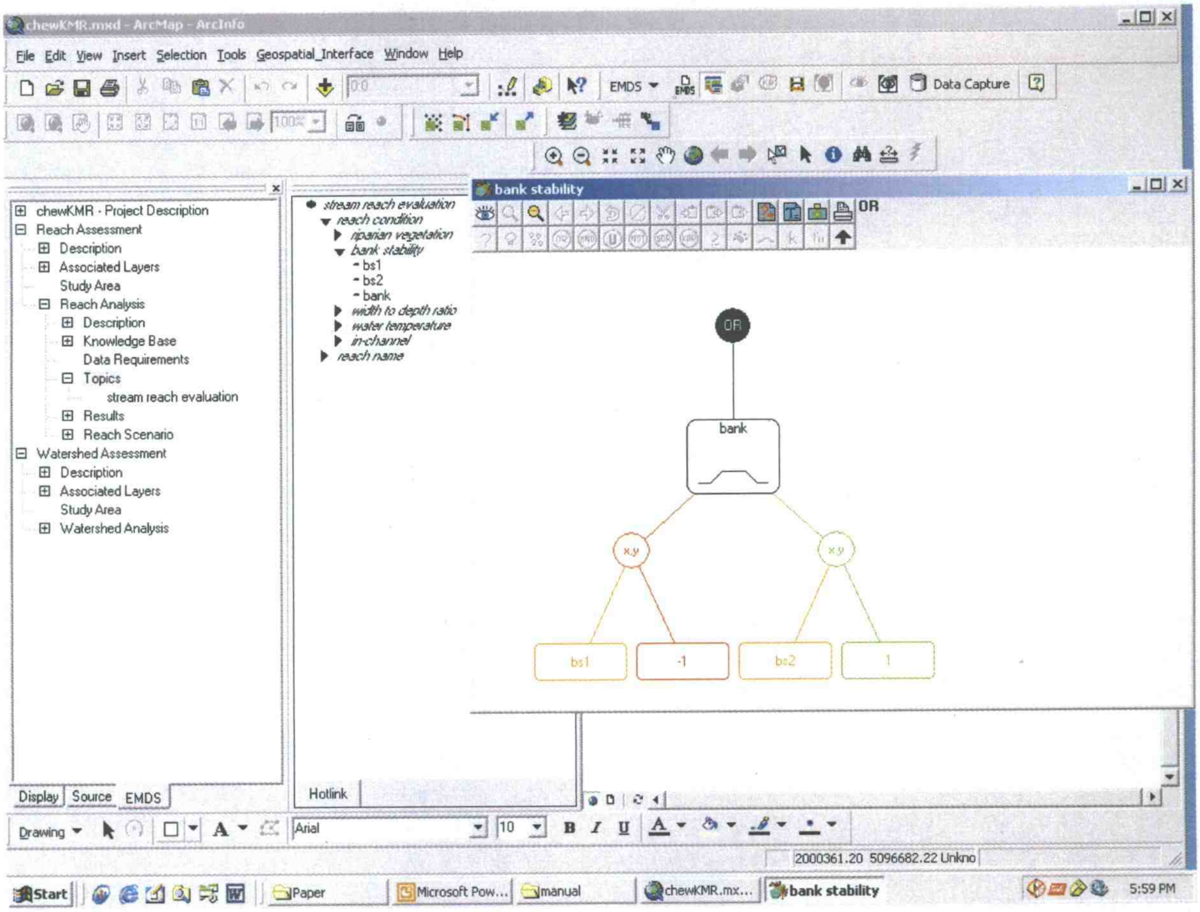


Figure 3. Knowledge base in Network Window

An example of the graphical output resulting from an evaluation of the reach-scale knowledge base is shown in Figure 4, where the relative influence of different attributes is depicted.

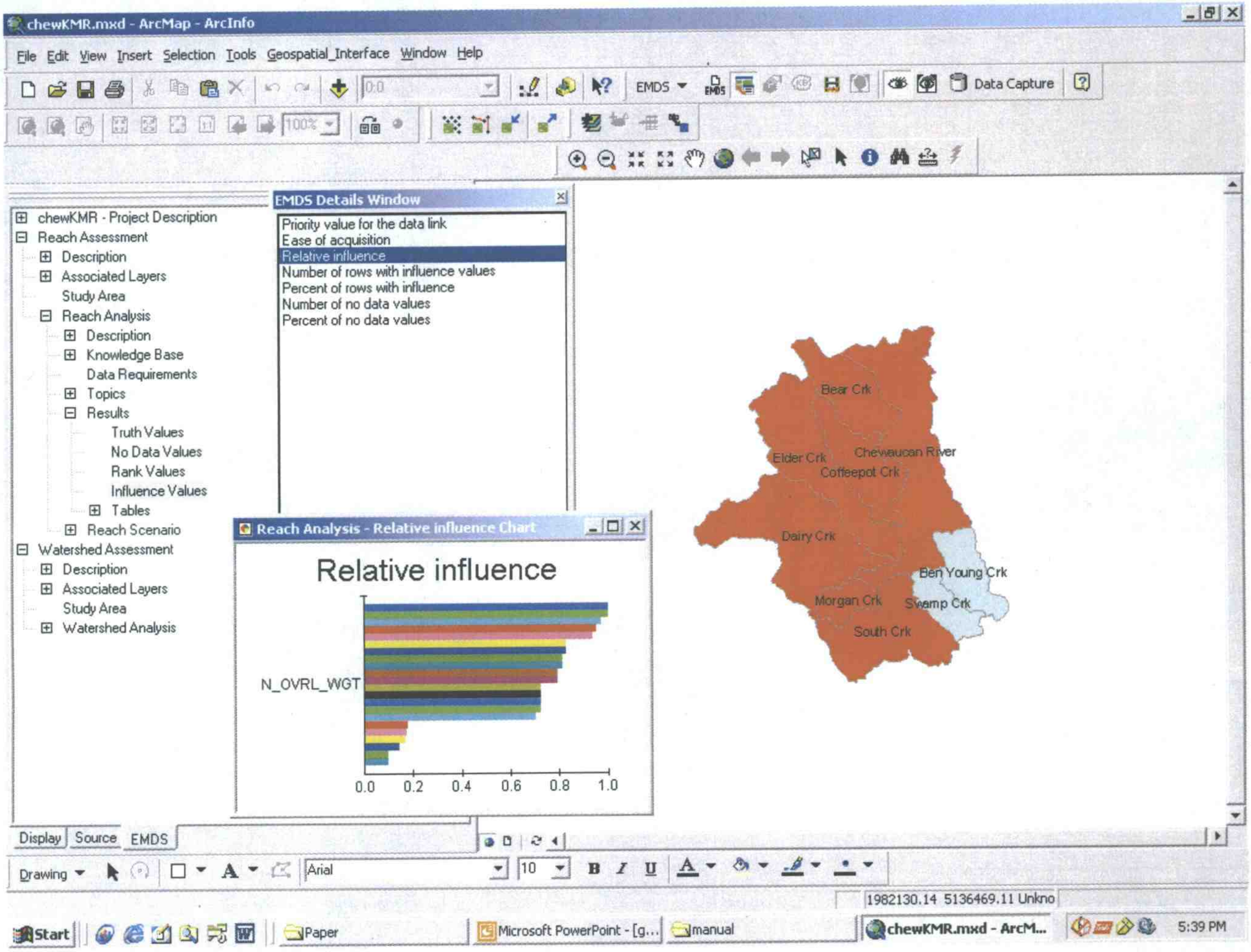


Figure 4. Graph of relative influence scores

Figure 5 shows the map layer created as a result of displaying TruthValue scores for the bank stability criterion as determined by applying the knowledge base to the stream reach data.

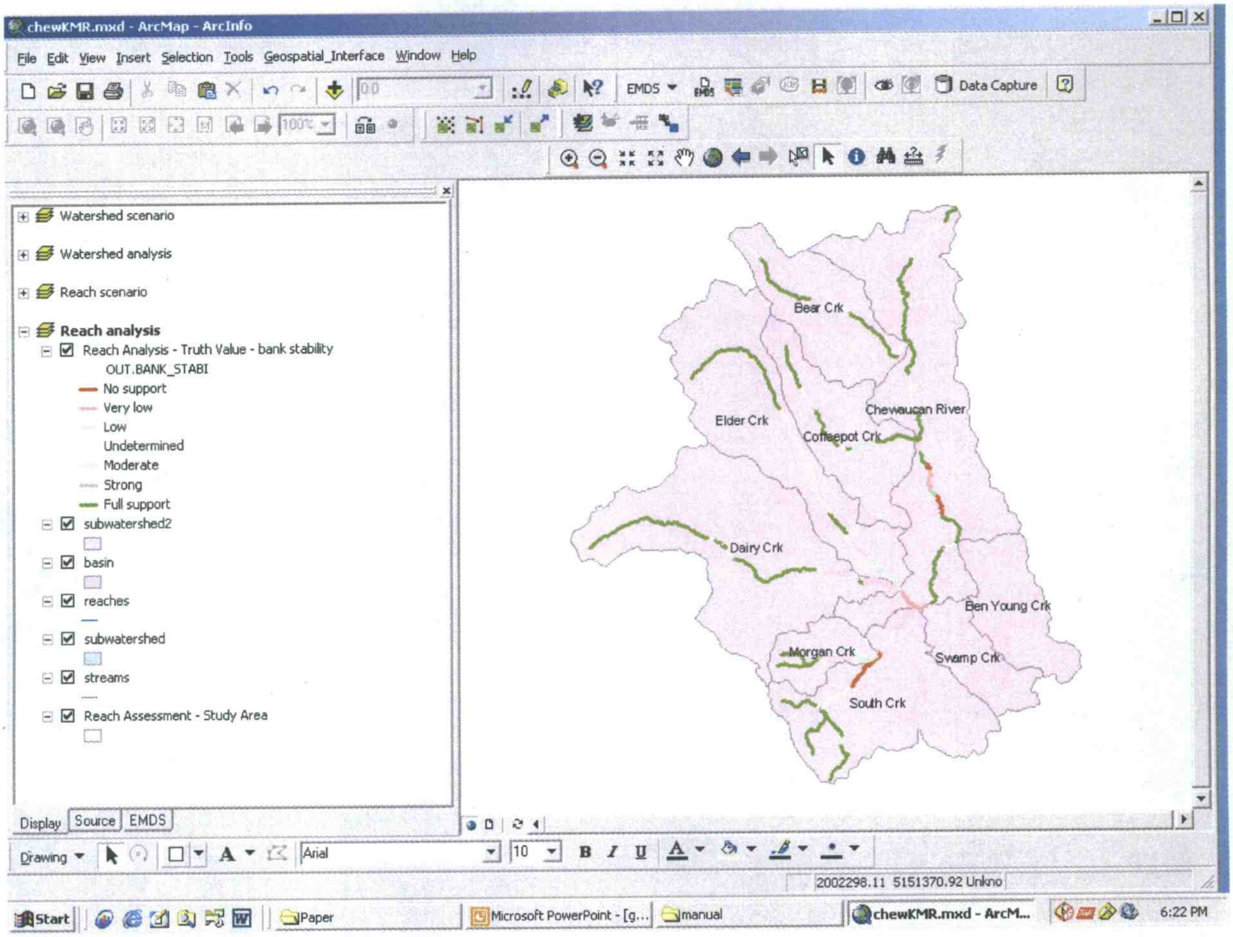


Figure 5. Map created for bank stability Truth Values

The contents of the project folder for chewKMR are shown below.

```
chewKMR.mxd
chewKMR folder:
        EmdsProject.xml
        Reach Assessment folder:
                EmdsAssessment.xml
                Layers folder:
                        basin.lyr
                        reaches.lyr
                        subwatershed2.lyr
                Reach Assessment Shapefile
                Reach Analysis folder:
                        EmdsAnalysis.xml
                        KnowledgeBase.nw
        ALIASES.DBF
        FLAG.DBF
        INF.DBF
        INFRNK.DBF
        infsum.dbf
        INN.DBF
        OUT.DBF
        Reach Analysis Shapefile
        Reach Scenario folder:
            EmdsScenario.xml
            KnowledgeBase.nw
            ALIASES.DBF
            FLAG.DBF
            INF.DBF
            INFRNK.DBF
            infsum.dbf
            INN.DBF
            OUT.DBF
            PAManager.mdb
Watershed Assessment folder:
    EmdsAssessment.xml
    Layers folder:
        basin.lyr
        subwatershed.lyr
    Watershed Assessment Shapefile
    Watershed Analysis folder:
        EmdsAnalysis.xml
        KnowledgeBase.nw
        ALIASES.DBF
        FLAG.DBF
        INF.DBF
        INFRNK.DBF
        infsum.dbf
        INN.DBF
        OUT.DBF
        ScenarioIgnore folder:
            EmdsScenario.xml
            KnowledgeBase.nw
            ALIASES.DBF
            FLAG.DBF
            INF.DBF
            INFRNK.DBF
            infsum.dbf
            INN.DBF
            OUT.DBF
            PAManager.mdb
```

Standard output products generated and saved any analysis or scenario include the database files INN.DBF, INFRNK.DBF, infsum.dbf, OUT.DBF, and OUT.MDB. These files are created when the data layers for the study area are evaluated using the information in the NetWeaver knowledge base. There is some overlap in their contents, as the OUT file contains all the fields in INF and INFRNK.

The Shapefile at the assessment level contains the defining geometry for the study area designated by the user, but does not contain additional attribute information. Similarly, the Shapefiles at the analysis level retain only the geometry data from the files supplied by the user at the outset of the session, and must be joined on the fly with attribute data from NetWeaver's out.dbf file to display results from the evaluations. The files in the layers folder at the assessment level are the result of writing out the input Shapefiles as layer files.

The XML files at each level of the project hierarchy contain annotations produced by the user during a session, such as a description of the scope and purpose of each level, and the author and date. The XML files for the analysis and scenario levels also contain information about the content of the associated knowledge base. Note that in the case of a scenario, this information may be altered from the original knowledge base to reflect the particular conditions of the scenario, such as choosing to ignore certain logic pathways in an evaluation.

In addition to outputs that are normally saved for an EMDS project, several data products are produced in the course of performing an assessment and are not necessarily retained at the conclusion of the session. These include graphs, such as shown in Figure 6, that the user may choose to display when using the Priority Analyst.

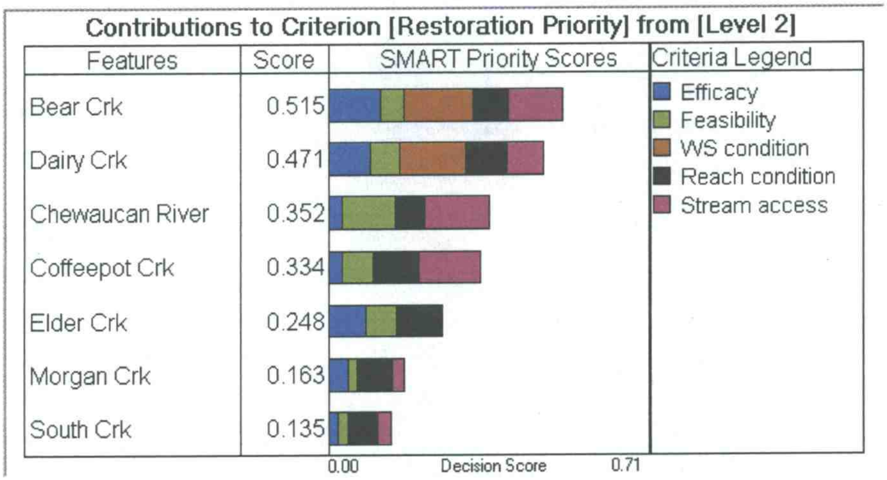


Figure 6. Graph generated by Priority Analyst

This same information may be presented in tabular form as well. When the user clicks on the Display button after selecting one of the attributes or criteria in the Details window, a new data layer is produced to present the result as a map. This layer is produced by performing a spatial join on the fly and appears in the ArcMap Table of Contents, but is not saved unless the user explicitly does so.

The fact that some information is not automatically retained in EMDS is not a design oversight. EMDS was originally designed as a stand-alone application whose primary products are the results of performing ecological analyses and examining alternative scenarios. As a consequence not all intermediate products are retained, and some of the results that are saved are not in a format that is conducive for importing by another application. Within the realm of an ecosystem assessment where the person running EMDS is the consumer of the results, the stand-alone design works well; however, evolving developments in natural resources management have created new data requirements.

Need for data migration

Emerging public awareness of environmental issues in the 1960's and 1970's led to the passage of landmark legislation that changed the way federal agencies approached natural resource management. The National Environmental Policy Act (1969), the Forest and Rangeland Renewable Resource Act (1974) and the National Forest Management Act (1976) helped establish guidelines for environmental policy making (US Department of Agriculture, Forest Service, Ecosystem Management Coordination (a)). Two key elements of the prescribed approach are environmental assessments and public involvement. Within the Forest Service these elements are addressed in Land and Resources Management Plans, also known as Forest Plans (US Department of Agriculture, Forest Service, Ecosystem Management Coordination (b)).

A Forest Plan is one component of a continuous adaptive management process that includes assessment, planning, implementation, and monitoring. A Forest Plan is required for each administrative unit within the National Forest System. It defines a strategy for achieving goals subject to design criteria, but on-the-ground decisions are made at the level of individual projects. A successful Forest Plan requires a smooth flow of information among components of an adaptive management approach as well as between the plan and project levels.

A tool recently developed by the Forest Service to aid in the planning process is called e-Planning (US Bureau of Land Management, ePlanning Project). E-Planning creates an online interactive document that treats public comments as managed data objects. Comments are captured electronically at public forums or submitted online, and are organized and tracked in the database. An e-Planning Web document is database-driven, so the authoritative version of the plan is electronic, rather than paper-based.

There is a natural progression from an environmental assessment to a resource management plan. An assessment produces a report card on the current state of a resource unit. A tool like EMDS can also produce scenarios of what the resource would look like under various management approaches. These results present a manager with a collection of choices and expected outcomes from adopting each choice, but they do not specifically give guidance on how to make a choice or how to implement one once it is selected. The latter issues are addressed in a planning document that identifies goals, strategies for achieving those goals, and monitoring measures to determine progress. If a planning tool such as e-Planning can readily assimilate the results generated by EMDS, the transition from assessment to planning can become more efficient and effective.

Approach Taken

One possible approach to coupling assessments and planning would be to develop software for each component that is designed from the ground up to be tightly integrated and interoperable. This may well be the best strategy if software development for both components is in the design phase and under the auspices of a single organization;

however, such is not the case for EMDS and e-Planning. EMDS is an operational product of the Forest Service's Natural Resource Information System (NRIS) Tools Group, and e-Planning is in the advanced stages of prototyping and testing as a joint venture of the Forest Service and the Bureau of Land Management. A major re-design of either product at this point would be quite costly. A much more feasible approach is to capture the relevant outputs of EMDS and organize them for input into e-Planning. This would make it possible to migrate the results of an ecosystem analysis, including alternative scenarios, into the planning process without major modifications of existing software.

As mentioned earlier, EMDS is written as an ArcMap extension, and it is under configuration management, currently at Version 3.02. The EMDS code consists of 97 Visual Basic modules, and the source code was made available for this study. Modifying the source code directly to effect the desired output changes was considered, but any such modifications, even minor ones, would have entailed a new release of the software with its associated overhead.

A design decision was made to accomplish the necessary data capture without making changes to the existing EMDS code. This dictated that the new code would have to run within the ArcMap environment "on top of" the EMDS extension and be able to extract data from objects being used by EMDS. Because EMDS is written in Visual Basic, this was a natural choice for the language to be used in the present project. Access to the ArcMap environment is accomplished by compiling the Visual Basic code to create a

Dynamic Link Library (DLL) object, which can be added to ArcMap as a new command on a toolbar.

The key to accessing the desired data lies in utilizing the ArcObjects data structures that underlie all ArcGIS products as well as the new objects defined in EMDS. To illustrate, the class structure of the EMDS code parallels the hierarchy described earlier, namely; project – assessments – analyses – scenarios. A method of the class *project* returns an *assessment,* and a property of an assessment is a textual attribute called *description.* Thus, to extract the text describing an assessment, one needs to access the EMDS project, access the assessment class, and read its description property. Other data items are more complicated to reach, but access to them depends on being able to navigate through the ArcObjects data structures and extract the desired information, which is then written to disk at a location designated by the user.

The new code operates by making two passes through the data. In the first pass it traverses the EMDS hierarchy to construct an output folder structure that mirrors the hierarchy. As the program goes through the hierarchy, it extracts certain text fields, such as author, date and description, at each level and writes them to the output folder. The tables containing the results of evaluating the study area data against the NetWeaver knowledge base, INFSUM.DBF and OUT.DBF, are stored in the appropriate analysis or scenario folder. Also, if any Excel spreadsheets or graphs are found in traversing the hierarchy, these are saved to disk as well.

In the second pass, the code examines the ArcMap Table of Contents to capture the data layers that the user has chosen to display. Recall that the maps depicting the results of using the Priority Analyst to study different environmental criteria are created on the fly and are not automatically retained by EMDS. The maps captured on this pass represent a snapshot of a dynamic collection of maps viewed by the user. It is up to users to initiate data capture when the maps are displayed that they wish to save. If a user views ten maps, but is only interested in five, he or she would perform the capture after deleting all but the five desired maps. This approach gives the user more control over what is saved than just taking what EMDS provides or saving every map. The output EMDS provides automatically, does not include any of the dynamic maps, while saving every map that gets created may include many that are not of interest.

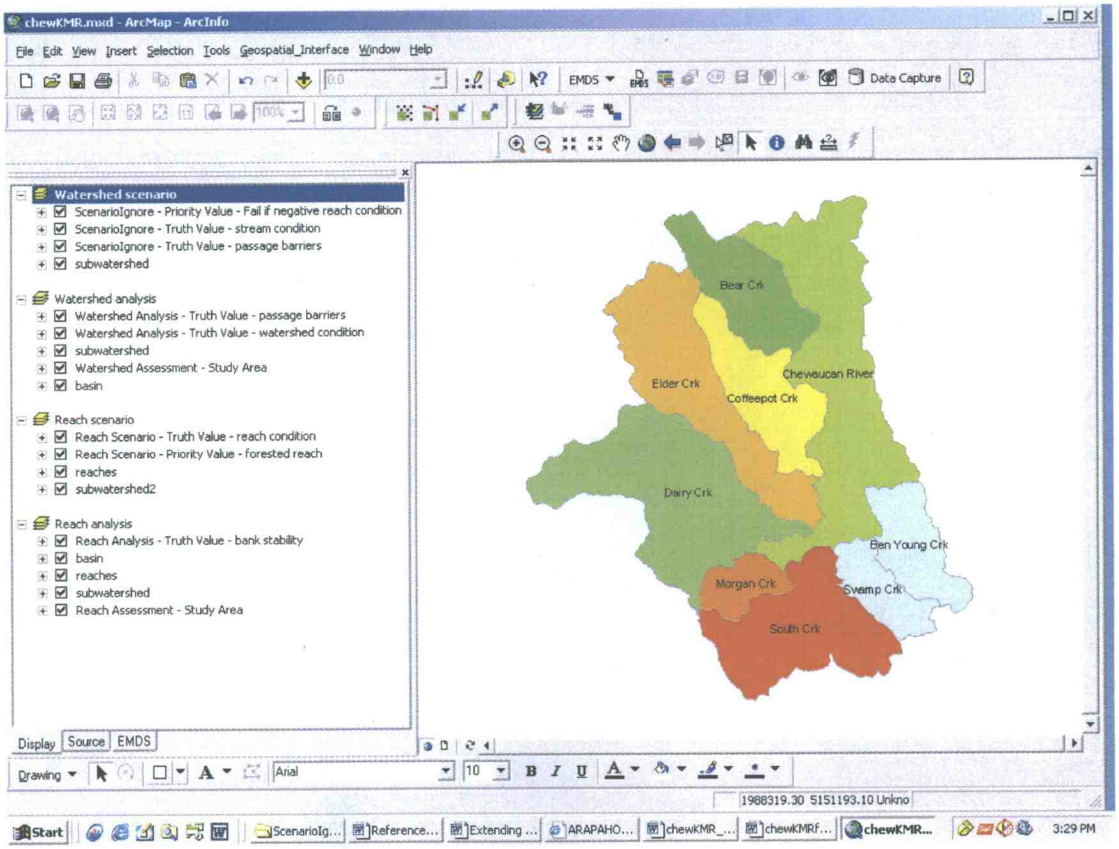
All maps captured in the second pass are stored as Shapefiles in a project-level folder, whose structure follows that of the ArcMap Table of Contents. There will be a folder for each Data Frame, within which maps will be grouped by analysis or scenario. Since the user has the freedom to shuffle maps between Data Frames, there is no guarantee that the arrangement of maps within Data Frames will follow the EMDS hierarchy, or any other scheme. As long as the user does not change the map names assigned by EMDS, the maps will be organized by analysis and scenario within a Data Frame.

Results

The EMDS data capture module was run on the Chewaucan Basin data discussed earlier. The Chewaucan basin project consists of two assessments: a reach assessment based on stream reaches and a watershed assessment based on subwatersheds. Each assessment consists of a single analysis, and each analysis, in turn, contains a single scenario.

In order to add the data capture functionality to the EMDS extension, a new command button is created for the EMDS toolbar and the compiled capture code in the form of a DLL file is associated with the button. The libraries referenced in the code must be available on the user's machine, and the user must designate a directory where the results folder will be created.

An ArcMap session was started and the Chewaucan map file was loaded. Figure 7 shows the four Data Frames for the two analyses, two scenarios, and the study area.



Watershed scenario
- ☑ ScenarioIgnore - Priority Value - Fail if negative reach condition
- ☑ ScenarioIgnore - Truth Value - stream condition
- ☑ ScenarioIgnore - Truth Value - passage barriers
- ☑ subwatershed

Watershed analysis
- ☑ Watershed Analysis - Truth Value - passage barriers
- ☑ Watershed Analysis - Truth Value - watershed condition
- ☑ subwatershed
- ☑ Watershed Assessment - Study Area
- ☑ basin

Reach scenario
- ☑ Reach Scenario - Truth Value - reach condition
- ☑ Reach Scenario - Priority Value - forested reach
- ☑ reaches
- ☑ subwatershed2

Reach analysis
- ☑ Reach Analysis - Truth Value - bank stability
- ☑ basin
- ☑ reaches
- ☑ subwatershed
- ☑ Reach Assessment - Study Area

Figure 7. Data Frames for chewKMR project

The EMDS toolbar is on the right side of the second header row. The Data Capture command is on the toolbar beside the Help command. To test the capture features, a Priority Analyst graph and table were created during the session. Several dynamic map layers were generated, and the Table of Contents in Figure 8 shows two of them within the Reach scenario. The Data Capture command was executed with these new layers present.

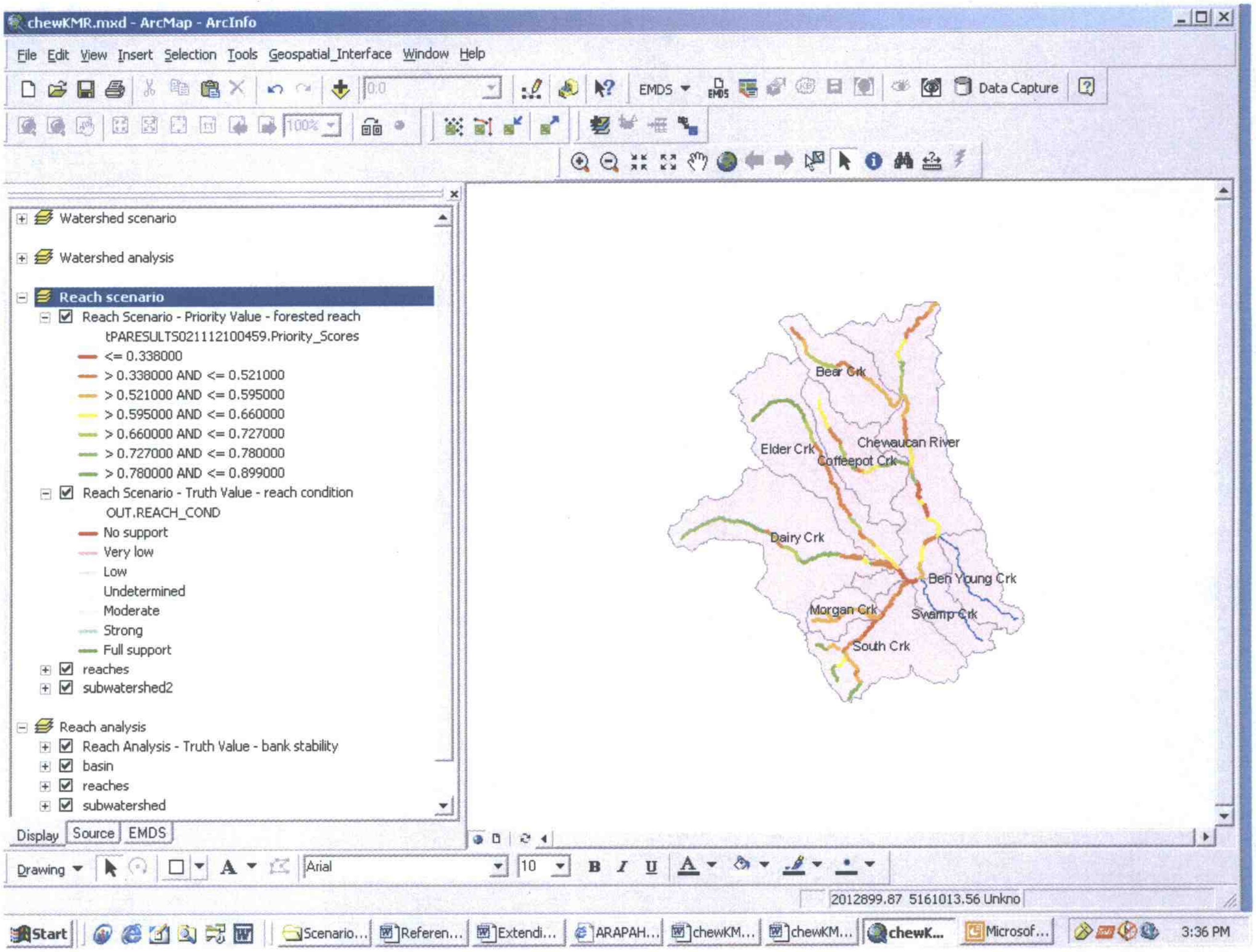

chewKMR.mxd - ArcMap - ArcInfo
File Edit View Insert Selection Tools Geospatial_Interface Window Help
EMDS
Data Capture
100%
⊞ Watershed scenario
⊞ Watershed analysis
⊟ Reach scenario
⊟ ☑ Reach Scenario - Priority Value - forested reach
tPARESULTS021112100459.Priority_Scores
<= 0.338000
> 0.338000 AND <= 0.521000
> 0.521000 AND <= 0.595000
> 0.595000 AND <= 0.660000
> 0.660000 AND <= 0.727000
> 0.727000 AND <= 0.780000
> 0.780000 AND <= 0.899000
⊟ ☑ Reach Scenario - Truth Value - reach condition
OUT.REACH_COND
No support
Very low
Low
Undetermined
Moderate
Strong
Full support
⊞ ☑ reaches
⊞ ☑ subwatershed2
⊟ Reach analysis
⊞ ☑ Reach Analysis - Truth Value - bank stability
⊞ ☑ basin
⊞ ☑ reaches
⊞ ☑ subwatershed
Display
Source
EMDS
Bear Crk
Elder Crk
Chewaucan River
Coffeepot Crk
Dairy Crk
Ben Young Crk
Morgan Crk
Swamp Crk
South Crk
Drawing
Arial
10
B I U
2012899.87 5161013.56 Unkno
Start
Scenario...
Referen...
Extendi...
ARAPAH...
chewKM...
chewKM...
chewK...
Microsof...
3:36 PM

Figure 8. Dynamic map layers for Reach scenario

The results of the capture are shown below, where, for brevity, only the Reach assessment folders are fully expanded.

```
ChewKMR
    Maps
        Reach analysis
            Reach Analysis
                Reach Analysis – Truth Value – bank stability Shapefile
            basin Shapefile
            reaches Shapefile
            subwatershed Shapefile
            Reach Assessment – Study Area Shapefile
        Reach scenario
            Reach Scenario
                Reach Scenario – Priority Value – forested reach Shapefile
                Reach Scenario – Truth Value – reach condition Shapefile
            reaches Shapefile
            subwatershed2 Shapefile
        Watershed analysis
        Watershed scenario
    Reach Assessment
        Reach Analysis
            Reach AnalysisGraphs
            Reach AnalysisTables
                infsum.dbf
                OUT.DBF
            Reach Scenario
                Reach ScenarioGraphs
                    PA – Forested Reach – Contributions.gif
                    PA – Forested Reach – Priorities.gif
                Reach ScenarioTables
                    Infsum.dbf
                    OUT.DBF
                    PA – Forested Reach – Contributions.xls
                    PA – Forested Reach – Priorities.xls
                Reach Sceanrio.txt
            Reach Analysis.txt
        Reach Assessment.txt
    Watershed Assessment
    chewKMR.txt
```

At the top level is the project folder, which contains the project textual information, ChewKMR .txt, folders for each assessment and the Maps folder. Looking in the Reach

Assessment folder, one finds the reach assessment text file and a folder for the analysis. In the Reach Analysis folder are the analysis text file, a folder for analysis graphs, a folder for analysis tables, and the scenario folder. The Reach AnalysisGraphs folder contains the graph, RelativeInfluence.jpg, created by the Priority Analyst. The Reach AnalysisTables folder contains the output tables, OUT.DBF and infsum.dbf, which are automatically generated in EMDS. The Reach Scenario folder, in turn, contains the scenario text data, and folders for scenario graphs and tables.

Back at the project level, the Maps folder contains a folder for each of the four Data Frames in ArcMap. The Reach analysis maps folder contains the following Shapefiles:

basin (input)
Reach Assessment – Study Area (user-defined)
reaches (input)
subwatershed (input)

It also contains the Reach Analysis subfolder, which holds Shapefiles for the *dynamic* maps generated when the user requested to display the Priority Analyst result:

Reach Analysis – Truth Value – bank stability

The other three Data Frame folders within Maps are similarly constructed. Note that the reaches Shapefile occurs in more than one Data Frame folder, so there is some duplication. Also, the dynamic maps are grouped into the same folder based on the first segment of their file name, which is either the name of an analysis or a scenario. These three-segment names are the names assigned by EMDS to be self-descriptive.

Conclusions

This work has looked at an existing tool for ecosystem management, EMDS, and added to its utility by building the front end of a data migration path to other tools that are part of an adaptive management approach to natural resources. It has done this by creating a simple, easy-to-use enhancement to the existing EMDS ArcMap extension that allows a user to capture and save relevant text, graphics, tabular, and map data for any EMDS project. All the data are saved in common formats that most current software tools can handle.

While the capture operation takes place automatically once the user initiates it, the user does exert some control over what gets saved. The user determines which dynamic maps displaying the results of analyses are created and which maps are present in the Data Frames when the capture occurs. Graphs and tables that present analysis results on-the-fly are also generated at the user's discretion, and can also be included in the captured data, but require the user to explicitly save them in the project folder prior to executing the capture.

One continuation to the current work, which would increase the utility of the results without much additional investment, would be to transfer the captured data from the folder it resides in to a database that is the backend for a Web page. This would make the results of an assessment much more accessible, and they could be examined without having any application software besides a browser. Transfer to another tool, such as a planning tool, could be greatly facilitated as well.

To realize the full benefit of a data path between assessment and planning, an end-to-end demonstration is desirable that begins with an assessment in EMDS, captures the results of the assessment, imports the results into e-Planning, and incorporates the results in the formulation of a Forest Plan. Then the real issues of application integration would surface and could start to be addressed.

References

Croteau, K.G., Faber, B.G., and Thomas, V.L. 1997. SMART PLACES: A tool for design and evaluation of land use scenarios. In *Proceedings of the 1997 ESRI User Conference*, Ref Paper #121. San Diego, CA: Environmental Systems Research Institute, Inc.

Densham, P., and Goodchild, M. 1989. Spatial decision support systems: A research agenda. In *Proceedings of GIS/LIS'89*, pp. 707-716. Orlando, FL.

Fall, A., Daust, D., and Morgan, D.G. 2001. A framework and software tool to support collaborative landscape analysis: Fitting square pegs into square holes. *Transactions in GIS* 5(1):67-86.

Fedra, K. 1995. Decision support for natural resources management: Models, GIS, and expert systems. *AI Applications* 9(3):3-19.

InfoHarvest Company. 2001. Criterium DecisionPlus users guide. InfoHarvest Company, Seattle, WA.

Jankowski, P., Andrienko, N., and Andrienko, G. 2001. Map-centered exploratory approach to multiple criteria spatial decision-making. Int. *J. Geographical Information Science* 15(2):101-127.

Kangas, J., Mehtatalo, L., Store, R., and Leskinen, P. 2000. Improving the quality of landscape ecological forest planning by utilizing advanced decision-support tools. *Forest Ecology and Management* 132(2-3):157-171.

Kamenetzky, R. 1982. The relationship between the analytical hierarchy process and the additive value function. *Decision Sciences* 13:702-716.

Rauscher, H.M. 1999. Ecosystem management decision support for federal forests in the United States: A review. *Forest Ecology and Management* 114:173-197.

Reynolds, K.M. 1999. NetWeaver for EMDS version 2.0 user guide: A knowledge base development system. General Technical Report PNW-GTR 471, P.75. US Department of Agriculture, Forest Service, Pacific Northwest Research Station, Portland, OR.

Reynolds, K.M., Jensen, M., Andreasen, J., and Goodman, I. 2000. Knowledge-based assessment of watershed condition. *Computers and Electronics in Agriculture* 27:315-333.

Reynolds, K.M., Rodriguez, S., and Bevans, K. 2002. Ecosystem Management Decision Support 3.0 user guide. US Department of Agriculture, Forest Service, Pacific Northwest Research Station, Forestry Sciences Laboratory, Corvallis, OR.

Saaty, T.L. 1994. *Fundamentals of Decision Making and Priority Theory with the Analytical Hierarchy Process.* RWS Publications, Pittsburgh, PA.

US Bureau of Land Management, ePlanning Project. ePlanning: E-Gov for planning and NEPA. http://web.blm.gov/eplanning/ [accessed November 26, 2003]

US Department of Agriculture, Forest Service, Ecosystem Management Coordination (a). List of important laws, policy and regulations. http://fsweb.wo.fs.fed.us/em/nfma/index.htm [accessed November 20, 2003]

US Department of Agriculture, Forest Service, Ecosystem Management Coordination (b). Principles of national forest planning. http://maps.fs.fed.us/fp/r2/arnf/ [accessed November 20, 2003]

US Department of Agriculture, Forest Service, Pacific Northwest Research Station, Forestry Sciences Laboratory. Ecosystem Management Decision Support. http://www.fsl.orst.edu/emds/ [accessed November 21, 2003]

Varma, V.K., Ferguson, I., and Wild, I. 2000. Decision support system for the sustainable forest management. *Forest Ecology and Management* 128(1-2):49-55.

Walsh, M.R. 2000. Toward spatial decision support systems in water resources. *Journal of Water Resources Planning & Management – ACSE* 119(2):158-169.

Appendix: Visual Basic code

```
Option Explicit
'*******************************************************************************
' This code captures text, map, tabular, and graphical data used in an
' EMDS session and saves it to a directory disignated by the user.  In the
' first segment the folder structure is established that closely follows
' the EMDS hierarchy, and text, tables and graphs are captured. In the
' second part, the LayerToShape subroutine traverses the Data Frames in
' ArcMap and creates Shapefiles from the layers and adds them to the
' project folder of the captured data.
'
' Author: Ken Crouse
' Revision History: 11/24/03    baseline version
'
' Development Environment:
'   ArcGIS 8.3
'   EMDS 3.02
'
' Referenced Libraries:
'   Visual Basic for Applications
'   Visual Basic runtime objects and procedures
'   Visual Basic objects and procedures
'   OLE Automation
'   EmdsCore
'   EmdsExtension
'   ESRI Object Library
'   Microsoft Scripting Runtime
'
'*******************************************************************************


Implements esriCore.ICommand
Private m_pApp As esriCore.IApplication
'Private m_pBitmap As IPictureDisp

Private Sub Class_Initialize()
   'Set m_pBitmap = frmEmdsTOC.ilsResources.ListImages("CDP_Map").Picture
End Sub

Private Sub Class_Terminate()
   Set m_pApp = Nothing
    'Set m_pBitmap = Nothing
End Sub

Private Property Get ICommand_Bitmap() As esriCore.OLE_HANDLE
```

```
    'ICommand_Bitmap = m_pBitmap
End Property

Private Property Get ICommand_Caption() As String
    ICommand_Caption = "Data Capture"
End Property

Private Property Get ICommand_Category() As String
    ICommand_Category = "EMDS Add-On"
End Property

Private Property Get ICommand_Checked() As Boolean
    ICommand_Checked = False
End Property

Private Property Get ICommand_Enabled() As Boolean
    ICommand_Enabled = True
End Property

Private Property Get ICommand_HelpContextID() As Long

End Property

Private Property Get ICommand_HelpFile() As String

End Property

Private Property Get ICommand_Message() As String
    ICommand_Message = "Write file hierarchy to disk"
End Property

Private Property Get ICommand_Name() As String
    ICommand_Name = "EmdsExtension_EmdsWriteFile"
End Property

Private Sub ICommand_OnClick()
    Dim pUID As esriCore.UID
    Dim pExtension As clsEmdsExt
    Dim strPathCode As String

    Dim fso As New FileSystemObject  ' File system objects
    Dim drv As Drive
    Dim fld As Folder

    Dim strDirProj As String        ' Project variables
    Dim filProj As TextStream
Dim strProjName As String
Dim filName As TextStream
Dim filPath As TextStream

Dim pAsmt As clsEmdsAssessment   ' Assessment variables
Dim vAsmt As Variant
vAsmt = 0
Dim strDirAsmt As String
Dim filAsmt As TextStream
Dim strAsmtName As String

Dim pAnal As clsEmdsAnalysis     ' Analysis variables
Dim vAnal As Variant
vAnal = 0
Dim strDirAnal As String
Dim filAnal As TextStream
Dim strAnalName As String
Dim strAnalInTables As String
Dim strAnalOutTables As String
Dim strAnalInGraphs As String
Dim strAnalOutGraphs As String

Dim pScen As clsEmdsAnalysis     ' Scenario variables
Dim vScen As Variant
vScen = 0
Dim strDirScen As String
Dim filScen As TextStream
Dim strScenName As String
Dim strScenInTables As String
Dim strScenOutTables As String
Dim strScenInGraphs As String
Dim strScenOutGraphs As String

'Dim strFolders(1 To 50, 1 To 2) As String ' Array for folder name, path
'Dim lFldrIndex As Long
'lFldrIndex = 0

' Get EMDS extension

Set pUID = New esriCore.UID
pUID.Value = "EmdsExtension.clsEmdsExt"
Set pExtension = m_pApp.FindExtensionByCLSID(pUID)

Dim pMxDoc As IMxDocument
Set pMxDoc = m_pApp.Document
pMxDoc.UpdateContents
```

```
   ' Check for project

   If Not pExtension.EmdsProject Is Nothing Then
      MsgBox " Project: " & pExtension.EmdsProject.Description.Name
    Else
      MsgBox "No project found"
      Exit Sub
   End If

   ' Request path from user; use default if none
   strDirProj = InputBox("Enter path of output directory, ending in \ ", _
"Output Directory",  "C:\temp\")

   ' Set up to create menu on new toolbar
   'Dim pCmdTool As ICommandBar
   'Set pCmdTool = m_pApp.Document.CommandBars.Create("Toolbar",
'esriCmdBarTypeToolbar)
   'Dim pCmdMenu As ICommandBar
   'Set pCmdMenu = pCmdTool.CreateMenu("Select Maps")
   'Dim pSubAsmt As ICommandBar
   'Dim pSubAnal As ICommandBar
   'Dim pSubScen As ICommandBar

   ' Set up project directory
   strProjName = pExtension.EmdsProject.Description.Name
   strDirProj = strDirProj & strProjName       'Append project name to path
   Set fld = fso.CreateFolder(strDirProj)      'Create new project folder
   Set fld = fso.CreateFolder(strDirProj & "\" & "Maps")  'Folder for all maps

   ' Create text file for project data
   Set filProj = fso.CreateTextFile(strDirProj & "\" & strProjName & ".txt", True)
   ' Write project name, comments, and ID to text file
   filProj.WriteLine (strProjName)
   filProj.WriteLine (pExtension.EmdsProject.Description.Comments)
   filProj.WriteLine (strPathCode)

   ' Create menu entry for project
   'pCmdMenu.CreateMacroItem "Project: ", 0, "m_pApp.Document.ProjMacro"
   'Dim pSubProj As ICommandBar
   'Set pSubProj = pCmdMenu.CreateMenu(strProjName)
   'pSubProj.CreateMacroItem "Assessments:", 0, "ProjMacro"

   ' Look for assessments
   vAsmt = 0
```

```
  While (pExtension.EmdsProject.AssessmentCount > vAsmt) 'Total count exceeds no.
'processed
   vAsmt = vAsmt + 1
   Set pAsmt = pExtension.EmdsProject.Assessment(vAsmt)
   strAsmtName = pAsmt.Description.Name
   strDirAsmt = strDirProj & "\" & strAsmtName
   Set fld = fso.CreateFolder(strDirAsmt)     'Create assessment folders & text file
   Set filAsmt = fso.CreateTextFile(strDirAsmt & "\" & strAsmtName & ".txt", True)
   filAsmt.WriteLine (strAsmtName)
   filAsmt.WriteLine (pAsmt.Description.Comments)
   filAsmt.WriteLine (pAsmt.Description.CreatedBy)

   ' Create menu entry for assessment
   'Set pSubAsmt = pSubProj.CreateMenu(strAsmtName)
   'pSubAsmt.CreateMacroItem strAsmtName & "Shapefile", 0,
"EmdsExtension.clsEmdsHelpStartCmd"
   'pSubAsmt.CreateMacroItem "Analyses:", 0,
"EmdsExtension.clsEmdsHelpStartCmd"

   ' Process each analysis within an assessment
   vAnal = 0
   While (pAsmt.AnalysisCount > vAnal)
    vAnal = vAnal + 1
    Set pAnal = pAsmt.Analysis(vAnal)
    strAnalName = pAnal.Description.Name
    strDirAnal = strDirAsmt & "\" & strAnalName
    Set fld = fso.CreateFolder(strDirAnal)  'Create analysis folders and text file
    Set fld = fso.CreateFolder(strDirAnal & "\" & strAnalName & "Graphs")
    Set fld = fso.CreateFolder(strDirAnal & "\" & strAnalName & "Tables")
    Set filAnal = fso.CreateTextFile(strDirAnal & "\" & strAnalName & ".txt", True)
    filAnal.WriteLine (strAnalName)              ' Write info to text file
    filAnal.WriteLine (pAnal.Description.Comments)
    filAnal.WriteLine (pAnal.Description.CreatedBy)
    filAnal.WriteLine (strPathCode)

    strAnalInTables = pAnal.Results.Location.PathName ' Copy INN, OUT, INFSUM
'tables
    strAnalOutTables = strDirAnal & "\" & strAnalName & "Tables"
    Call GetTables(strAnalInTables, strAnalOutTables)

    strAnalInGraphs = pAnal.Results.Location.PathName   ' Copy graphs
    strAnalOutGraphs = strDirAnal & "\" & strAnalName & "Graphs"
    Call GetGraphs(strAnalInGraphs, strAnalOutGraphs)

    ' Create menu entry for analysis
    'Set pSubAnal = pSubAsmt.CreateMenu(strAnalName)
    'pSubAnal.CreateMacroItem strAnalName & "Shapefile", 0, "ProjMacro"
    'pSubAnal.CreateMacroItem strAnalName & "Results", 0, "ProjMacro"
    'pSubAnal.CreateMacroItem "Scenarios:", 0, "ProjMacro"

    ' Process each scenario within an analysis
    vScen = 0
    While (pAnal.ScenarioCount > vScen)
     vScen = vScen + 1
     Set pScen = pAnal.Scenario(vScen)
     strScenName = pScen.Description.Name
     strDirScen = strDirAnal & "\" & strScenName
     Set fld = fso.CreateFolder(strDirScen)  'Create scenario folders and text file
     Set fld = fso.CreateFolder(strDirScen & "\" & strScenName & "Graphs")
     Set fld = fso.CreateFolder(strDirScen & "\" & strScenName & "Tables")
     Set filScen = fso.CreateTextFile(strDirScen & "\" & strScenName & ".txt", True)

     filScen.WriteLine (strScenName)              ' Write text data
     filScen.WriteLine (pScen.Description.Comments)
     filScen.WriteLine (pScen.Description.CreatedBy)

     strScenInTables = pScen.Results.Location.PathName  ' Copy INN, OUT, INFSUM
tables
     strScenOutTables = strDirScen & "\" & strScenName & "Tables"
     Call GetTables(strScenInTables, strScenOutTables)

     strScenInGraphs = pScen.Results.Location.PathName  ' Copy graphs to new folder
     strScenOutGraphs = strDirScen & "\" & strScenName & "Graphs"
     Call GetGraphs(strScenInGraphs, strScenOutGraphs)

     ' Create menu entry for scenario
     'Set pSubScen = pSubAnal.CreateMenu(strScenName)
     'pSubScen.CreateMacroItem strScenName & "Results", 0, "ProjMacro"


    Wend
    vScen = 0
   Wend
   vAnal = 0
  Wend

  filProj.Close
  filAsmt.Close
  filAnal.Close
  filScen.Close

  Dim intProceed
```

```
   intProceed = MsgBox("Finished creating folder hierarchy.  " _
          & "Proceed to store maps?", vbOKCancel, "Write maps")

   ' Call LayerToShape to traverse dataframe structure and save layers
   ' as Shapefiles.
   If (intProceed = 1) Then
   Call LayerToShape(strDirProj)
   Else
      MsgBox "Exiting; no maps stored"
   End If


End Sub
Private Sub GetTables(strTreePath As String, strTablesFolder As String)

   ' Look for designated .dbf tables (in DBFname) and Excel files and
   ' copy them into the proper Tables folder in the output directory

   Dim strDBFname As Variant
   Dim strTable As String
   strDBFname = Array("OUT", "INFSUM")

   Dim i As Integer
   For i = 0 To 1
      strTable = Dir(strTreePath & strDBFname(i) & ".DBF")
      While (strTable <> "")
         FileCopy strTreePath & strTable, strTablesFolder & "\" & strTable
         strTable = Dir()     ' Repeat with last specification
      Wend
   Next i

   ' Copy any Excel files

   strTable = Dir(strTreePath & "*.xls")
   While (strTable <> "")
         FileCopy strTreePath & strTable, strTablesFolder & "\" & strTable
         strTable = Dir()     ' Repeat with last specification
   Wend
End Sub

Private Sub GetGraphs(strTreePath As String, strGraphsFolder As String)

   ' Look for images of the image types indicated in Suffix that are
   ' assumed to contain graphs created in EMDS.  Copy the image files
   ' to the Graphs output folder.
```

```
    Dim strGraph As String
    Dim strSuffix As Variant
    strSuffix = Array("jpg", "bmp", "gif") 'Image types of graphs


    ' Look for any files with image extension and copy to graph folder
    Dim i As Integer
    For i = 0 To 2
        strGraph = Dir(strTreePath & "*." & strSuffix(i))
        While (strGraph <> "")
            FileCopy strTreePath & strGraph, strGraphsFolder & "\" & strGraph
            strGraph = Dir()    ' Repeat with last specification
        Wend
    Next i

End Sub
Private Sub LayerToShape(strDirProj As String)

    ' Traverse the Data Frames in the ArcMap Table of Contents and
    ' convert each feature layer into a Shapefile to be saved in each
    ' Data Frame map folder.  A Data Frame folder contains maps for
    ' input layers, study area boundaries, and a separate subfolder for
    ' maps created dynamically by EMDS.

    Dim fso As New FileSystemObject
    Dim drv As Drive
    Dim fld As Folder

    Dim pMap As esriCore.IMap
    Dim pLayer As esriCore.ILayer
    Dim pFeatureClass As esriCore.IFeatureClass
    Dim strDirDataFrame As String
    Dim strDirPrefix As String
    Dim strLayerString As String
    Dim strPrefix As String
    Dim intSegments As Integer
    Dim i As Long
    Dim j As Long

     ' Iterate through map layers

    Dim pMxDoc As IMxDocument
    Set pMxDoc = m_pApp.Document
    Dim pMaps As esriCore.IMaps
    Set pMaps = pMxDoc.Maps
    If pMaps.Count = 0 Then
```

```
   MsgBox "No maps present", vbOKOnly, "EmdsCore:clsEmdsLayers"
Else
   For i = 0 To pMaps.Count - 1
      Set pMap = pMaps.Item(i)

      ' Create dataframe folder
      strDirDataFrame = strDirProj & "\Maps\" & pMap.Name & "\"
      Set fld = fso.CreateFolder(strDirDataFrame)
      For j = 0 To pMap.LayerCount - 1
         Set pLayer = pMap.Layer(j)
         If TypeOf pLayer Is esriCore.IFeatureLayer Then
            strLayerString = pLayer.Name
            strPrefix = ""
            intSegments = 0
            ' Parse layer name to get segments
            Call ParseLayerName(strLayerString, strPrefix, intSegments)
            Select Case intSegments
               Case 1, 2
                  Call MakeShape(pLayer, strDirDataFrame, strLayerString)
               Case 3
                  strDirPrefix = strDirDataFrame & strPrefix & "\"
                     If Not (fso.FolderExists(strDirPrefix)) Then
                     Set fld = fso.CreateFolder(strDirPrefix)
                  End If
                  Call MakeShape(pLayer, strDirPrefix, strLayerString)
               Case Else
                  MsgBox "Parse failed; check layer name"
                  Exit Sub
            End Select


         Else
            MsgBox "Not a feature layer"
         End If
      Next j
   Next i
End If


' Clean up
Set pMxDoc = Nothing
Set pMaps = Nothing
Set pMap = Nothing
Set pLayer = Nothing
Set pFeatureClass = Nothing
```

```
End Sub

Private Sub MakeShape(ByVal Layer As esriCore.ILayer, strDirectory As String, _
            strShapefileName As String)

 ' Given a map layer, create a Shapefile in the specified directory from
 ' a join using the Export Operation interface

On Error GoTo eh

Dim pGFLayer As IGeoFeatureLayer
Set pGFLayer = Layer

Dim pDT As IDisplayTable
Set pDT = pGFLayer

Dim pFClass As IFeatureClass
'Set pFClass = pDT.DisplayTable
Set pFClass = pGFLayer.DisplayFeatureClass

'Get the Input DataSet Name
Dim pDataSet As IDataset
Set pDataSet = pFClass
Dim pInDsname As IDatasetName
Set pInDsname = pDataSet.FullName

Dim pOutWSFact As IWorkspaceFactory
Dim pOutWs As IWorkspace
Dim pDData As IDataset
Dim pWSName As IWorkspaceName
Dim pOutDSName As IDatasetName

'Set up the OutPut Workspace
Set pOutWSFact = New ShapefileWorkspaceFactory
Set pOutWs = pOutWSFact.OpenFromFile(strDirectory, 0)
Set pDData = pOutWs

Set pWSName = pDData.FullName
Set pOutDSName = New FeatureClassName
pOutDSName.Name = strShapefileName
Set pOutDSName.WorkspaceName = pWSName

'Export the Feature Class
Dim pExp As IExportOperation
```

```
Set pExp = New ExportOperation

pExp.ExportFeatureClass pInDsname, Nothing, Nothing, Nothing, pOutDSName, 0


 ' Cleanup
 Set pGFLayer = Nothing
 Set pDT = Nothing
 Set pFClass = Nothing
 Set pDataSet = Nothing
 Set pInDsname = Nothing
 Set pOutWSFact = Nothing
 Set pDData = Nothing
 Set pOutWs = Nothing
 Set pWSName = Nothing
 Set pOutDSName = Nothing
 Exit Sub  'Exit to avoid error handler
eh:
 MsgBox "Error in CreateShapeFile" & vbNewLine & _
     Err.Description, vbOKOnly, "DEBUG"
End Sub


Private Sub ParseLayerName(strLayerString As String, strPrefix As String, intSegments
As Integer)
  ' Given the name of a map layer, parse it to determine the number of
  ' segments separated by dashes.  A name generated for a dynamic map
  ' created during an EMDS analysis has the form:
  '   analysis/scenario name - type of indicator - attribute
  ' The first segment of a three-part name is taken as the analysis or
  ' scenario name for a subfolder within a Data Frame folder.

  Dim lFirstDash As Long
  Dim lSecondDash As Long

  lFirstDash = 0
  lSecondDash = 0

  lFirstDash = InStr(1, strLayerString, " - ")
  If lFirstDash = 0 Then
     intSegments = 1
     Exit Sub
  End If

  lSecondDash = InStr(lFirstDash + 3, strLayerString, " - ")
  If lSecondDash = 0 Then
```

```
        intSegments = 2
        Else
           intSegments = 3
           strPrefix = Left(strLayerString, lFirstDash - 1)
     End If
  End Sub

  Public Sub ProjMacro()
     MsgBox "Select project"
  End Sub
  Private Sub ICommand_OnCreate(ByVal hook As Object)
     Set m_pApp = hook  ' Trick to gaining access to application from VB
  End Sub

  Private Property Get ICommand_Tooltip() As String
     ICommand_Tooltip = "Display description"
  End Property
```