Intelligent Model Management in a Forest Ecosystem Management Decision Support System

Donald Nute[a] (dnute@uga.edu), Walter D. Potter[a], Frederick Maier[a], Jin Wang[a], Mark Twery[b], H. Michael Rauscher[c], Peter Knopp[b], Scott Thomasma[b], Mayukh Dass[a], and Hajime Uchiyama[a]

[a] *Artificial Intelligence Center, The University of Georgia, Athens, GA 30605, USA*

[b] *Northeastern Research Station, USDA Forest Service, P.O. Box 968, Burlington, VT 05402-0968, USA*

[c] *Bent Creek Experimental Forest, Southern Research Station, USDA Forests Service, Asheville, NC, USA*

Abstract: Decision making for forest ecosystem management can include the use of a wide variety of modeling tools. These tools include vegetation growth models, wildlife models, silvicultural models, GIS, and visualization tools. NED-2 is a robust, intelligent, goal-driven decision support system that integrates tools in each of these categories. NED-2 uses a blackboard architecture and a set of semi-autonomous agents to manage these tools for the user. The blackboard integrates a Microsoft Access database and Prolog clauses, and the agents are implemented in Prolog. A graphical user interface written in Visual C++ provides powerful inventory analysis tools; dialogs for selecting timber, water, ecological, wildlife, and visual goals; and dialogs for defining treatments and building prescriptive management plans. Users can simulate management plans and perform goal analysis on different views of the management unit, where a view is determined by a management plan and a point in time. Prolog agents use growth and yield models to simulate management plans, perform goal analyses on user-specified views of the management unit, display results of plan simulation using GIS and visualization tools, and generate hypertext documents containing the results of such analysis. Individual agents use meta-knowledge to set up and run external simulation models, to load rule-based models and perform inference, to set up and execute external GIS and visualization systems, and to generate hypertext reports as needed, relieving the user from performing all these tasks. The NED-2 interface can present basic results from the simulation of alternative management plans in side-by-side displays

Keywords: ecosystem management, model management, decision support system, knowledge based system, blackboard architecture, Prolog.

1. INTRODUCTION

NED-2 is an Intelligent Information System designed to provide decision support for forest ecological system management in the eastern United States. The latest product of the Northeast Decision Model project and the successor of NED-1[1] [Twery et al. 2000], NED-2 integrates databases, knowledge bases, simulations, geographic information systems, visualization tools, and other knowledge sources and knowledge tools. NED-2 has an open agent-based architecture that facilitates future integration of additional knowledge sources and tools. The U.S.D.A. Forest Service developed NED-2 in collaboration with researchers from several universities.

We will describe a session with NED-2 to illustrate the NED decision process. Then we will describe the NED-2 architecture in detail. Finally, we will summarize the current status of the NED-2 prototype and the work that remains to be done to complete a full implementation of the NED-2 design.

[1] NED-1 and other software from the NED project is available free of charge from http://www.fs.fed.us/ne/burlington.ned.

2. A SESSION WITH NED-2

The NED-2 decision process is driven by information about the management unit, but it is also goal driven [Rauscher et al. 2000]. The first steps in using NED-2 are entry of inventory data and selection of management goals. Then treatment parameters are set and management plans are developed. Next, a baseline year is established and treatment plans are constructed. Treatment plans are simulated using growth and yield models. Goal satisfaction analysis can be performed for any management plan at any point in time covered by the plan. Simulated data and the results of goal satisfaction analysis for a management plan at a point in time can be displayed using a combination of on-screen displays, hypertext reports, geographical information systems, and other visualization tools. Based on his review of these analyses, the user is then in a position to make an informed decision in determining the final management plan to be adopted for the management unit

In case studies conducted by the NED team, landowners showed strong interest in incorporating wildlife, recreational, ecological, and other non-timber goals when they realized there were effective, economical ways to pursue these goals. NED-2 provides the user with a tree of goals in these various. A goal can only be included if knowledge bases are available for evaluating whether the goal has been satisfied. The user can define multiple goal sets and evaluate management plans using any of these goal sets.

Managers typically think of the land they manage as a collection of distinct *stands* where each stand occupies an area of uniform composition, age, and condition. NED-2 requires the user to supply the species and size of each overstory tree on at least one sample plot on each forested stand. NED-2 will calculate basal areas, timber volumes, and other stand statistics that will drive many of the management. The user can enter data in a spreadsheet within NED-2 or NED-2 will import overstory data from several standard formats.

To manage for recreational value, wildlife, and other non-timber goals, a manager must also enter information about the shrub and ground layers and about other physical characteristics of each stand such as the presence of ponds or streams, the amount of coarse woody debris available, and the presence of snags or hollow trees.

A user may not have current inventory on all stands, and the most recent inventory for different stands may not have been taken in the same year. So NED-2 will generate a *baseline* by simulating stand growth and treatment on stands up to some common year selected by the user. All management plans constructed by the user will then begin with this baseline year. The first growth and yield simulators integrated into NED-2 are the northeastern and southern variants of the Forest Vegetation Simulator. [Teck et al. 1997]. After a baseline has been generated, this simulated data is displayed to the user using the same NED-2 spreadsheet tool that displays inventory data.

Management happens through treatments applied to the land. The development of management plans is guided by the goals that have been selected. Silvicultural treatments fall into categories such as fertilizing, thinning, and harvesting. But within these categories, specific treatments can be designed in many ways. A particular set of parameter settings for a kind of treatment might be used by a manager repeatedly. The NED-2 user specifies parameters for the treatments he intends to use in his management plans through special NED-2 dialogs.

After treatments have been defined, the user can construct one or several management plans and examine the consequences of each plan. Management plans are designed on a scheduling grid. Treatments can be applied to any stand within a given year displayed on the grid.

Once a plan has been constructed, it can be simulated using any available growth and yield simulator. The simulated data can be displayed using the NED-2 spreadsheet tool. This tool also allows side-by-side display of information for different years in a management plan or for different management plans for a single year.

The experience of the NED team suggests that managers often do not include a particular goal in their management plans, not because the goal is not desirable, but because they do not know how to *operationalize* it. For every goal included in NED-2, NED-2 also includes knowledge bases to evaluate data to determine whether or not the goal has been satisfied. These knowledge bases reduce goals to logical combinations of desirable future conditions (dfcs.) A dfc is a measurable or at least observable variable together with a target value or range of target values for the variable.

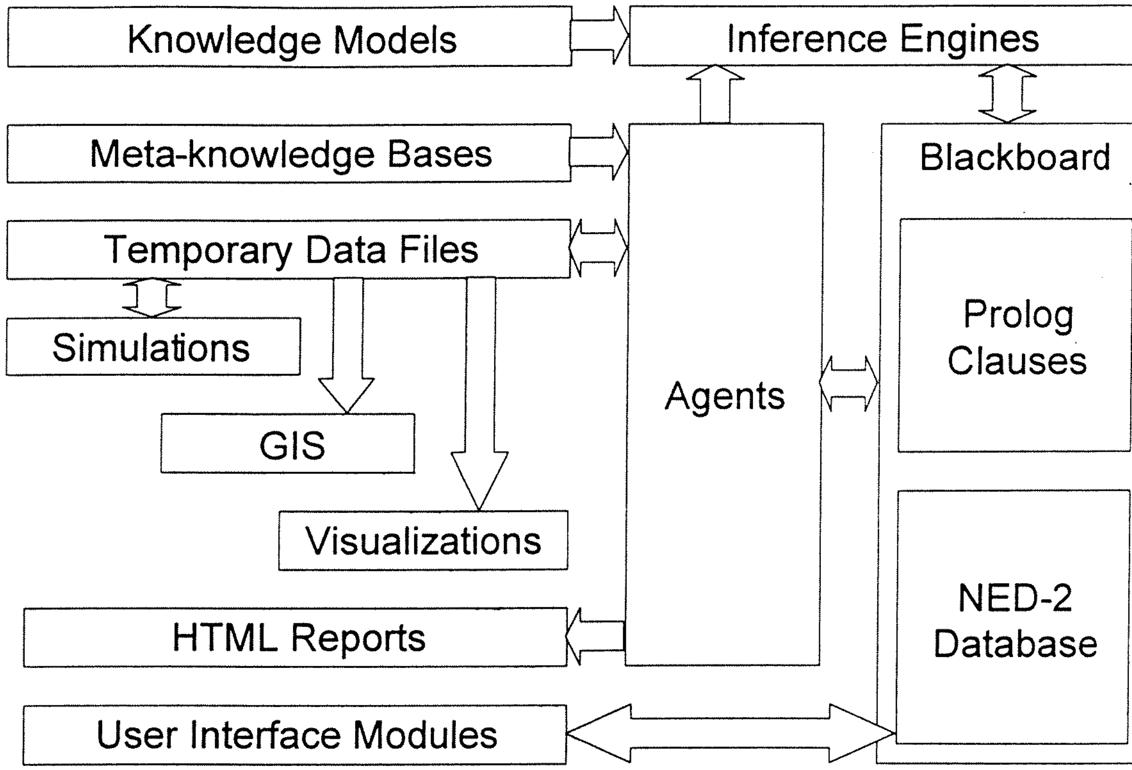

Figure 1: The NED-2 Architecture

NED-1 could perform goal satisfaction analysis only on inventory data. But NED-2 goes far beyond that since it allows the user to design management plans and simulate them.

The user can request goal analysis on the baseline for the management unit or on the simulated data for any year based on any management plan he has constructed. The user can also specify different goal sets for the goal satisfaction analysis.

NED-2 displays the results of simulations and analyses in a spreadsheet tool, but several other methods are also available. The most detailed method is to construct reports in the form of html files that can be viewed using a Web browser. These hypertext documents can be generated for any combination of goal set, management plan, and simulated data for a particular year. They include summaries of the simulated data and of the goal satisfaction analysis performed. If a shape file is available for the management unit, the user can also select data or goal evaluation results and send them to a geographical information system (ArcView.) NED-2 will configure the information into a temporary data table that can be merged with the shape file and displayed on a map. Finally, NED-2 can configure simulated stand information for any year under any management plan and send it to the Stand Visualization System [McGaughy 1997], or configure information for the entire management unit and send it to EnVision [McGaughy 1998].

3. THE NED-2 ARCHITECTURE

NED-2 integrates a sophisticated user interface, databases, simulations, knowledge bases, hypertext documents, geographical information systems, and visualization tools into a single decision support system. We wanted an open system that would allow us to incorporate additional simulations, knowledge bases, and other decision support tools easily. This would not be possible if integration of each component required extensive procedural programming. Instead, we decided to build intelligent agents each of which knew how to use a class of decision support tools. These agents are developed in Prolog, a high-level logic programming language. As an example, the NED-2 simulation agent knows that a growth and yield simulation requires input in a given format, requires control codes to simulate treatments and set stop conditions, and writes output in a specific format.

The central organizing principle for NED-2 is the blackboard [Ni 1989] (Figure 3). Unlike object oriented or mediator architectures, agents do not directly invoke each other in a blackboard system. Instead, tasks that need to be done are posted to a blackboard. Agents also post results of their activities to the blackboard where they are accessible by all other agents.

Agents watch the blackboard continually. A particular set of facts and/or tasks listed on the blackboard can prompt an agent to do some work. If an agent completes a task on the blackboard, it erases that task from the "to do" list. An agent may also begin a task, then discover that something needs to be done that is beyond its capability before it can complete the task. It can put the new task on the blackboard and wait until another agent performs it before completing the original task.

Blackboard architectures have certain advantages over other agent architectures. In object oriented or mediator architectures, an agent (or object) that needs help in performing some subtask must address the request for help to a specific agent that can perform that subtask. It must also provide that assistant with the specific information it will need to perform the subtask. So each agent must know what other agents can do and what they need to do it. With a blackboard architecture, an agent simply posts a request on the blackboard. The request is visible to every other agent and all information the assisting agent needs is also on the blackboard. This reduces the knowledge requirements for agents.

Blackboard architectures are more flexible than rigid hierarchical architectures. Organizing agents into hierarchical trees prevent agents from accepting assignments from any agents except their immediate supervisors. In object oriented programming, an object can respond to the request of any other object that knows how to address it. Using a blackboard architecture, an agent can respond to any agent that can write a request on the blackboard.

The NED-2 blackboard has three components: a set of Prolog clauses, a Microsoft Access database, and a set of Prolog procedures.

The simplest part of the NED-2 blackboard is represented by the two Prolog predicates `fact/4` and `request/1`. Facts are stored as AOV (attribute-object-value) triples. One of these is the first argument in a clause for the predicate `fact/4`. The other three arguments are the source for the fact, a fact characteristic index (usually the atom `true` but might in some cases be a confidence factor, fuzzy number, or some other value,) and a time stamp. A request can be the name of a task to perform or a list of tasks to be performed. AOV triples and tasks may be fully described given the expressive power provided by Prolog. For example,

```
fact(trees_per_acre([stand(17),
species(oak),  size_class(9)],
22),true,user,2345235).
```

Here the object is abstract: it is the oak trees of 9" diameter on stand 17. The attribute is the count per acre for these trees, and the value is 22. Requests can have a simple form like

```
request([analysis]).
```

Requests can also contain complex plans; these will be discussed below in the section on NED-2 planning agents.

NED-2 uses a Microsoft Access database to store inventory and other information. This database has been integrated into the NED-2 blackboard. When a NED-2 agent wants to know something, it calls the Prolog predicate `known/1`. The argument provided to `known/1` is an AOV triple where the attribute and the object are specified but the value is an uninstantiated variable. `known/1` matches this incomplete AOV triple against triples contained in clauses for `fact/4`. If it doesn't find a corresponding fact there, it converts the triple into an SQL query and seeks the information in the NED-2 database.

Agents place facts on the blackboard as Prolog clauses by asserting them. Agents formulate SQL queries to place data in the database.

4. NED-2 AGENTS

A NED-2 user interface agent consists of a small piece of Prolog code and a larger C++ module. The Prolog code reads the blackboard and determines when a particular user interface function is required. It then calls the appropriate C++ module.

Simulation agents are responsible for simulating the management plans that the user constructs. Both inventory, i.e., field measured, and simulated data are stored as "snapshots" in the NED-2 database. Each simulated snapshot represents data for a stand under a treatment regime for a single year. Snapshots include information about the overstory, understory, and ground layers. Growth and yield models generate overstory information; knowledge-based systems generate understory and ground layer information. Interface agents display snapshots and side-by-side comparisons of data in different snapshots.

Meta-knowledge is knowledge about knowledge. Simulations agents will need meta-knowledge to know when and how to use a knowledge source or a knowledge tool. For a particular simulation program, we develop a meta-knowledge base that provides the simulation agent with the tables needed to translate data between the NED-2 format and the format of the simulation. This meta-knowledge base also tells the simulation agent what control codes the simulation understands. So all procedural knowledge for running simulations is written into the simulation agent and specific knowledge needed to run individual simulations is stored in these declarative meta-knowledge bases.

The first growth and yield models integrated with NED-2 are the Northeastern and Southern variants of FVS [Crookston 1997]. Simulation agents set up and call these growth and yield models as needed. The user selects the simulator he wants to use or NED-2 will use meta-knowledge to recommend a simulator based on information about the management unit. Minimally, NED-2 evaluates the user's choice of simulator and notifies the user if it detects a problem. For example, NED-2 will alert the user that the Northeastern variant of FVS may not be appropriate for use in Georgia.

NED-2 knowledge based models are rule sets written in Prolog and used in conjunction with one of the NED-2 inference engines. Standard forward and backward chaining inference engines and a fuzzy backward chaining inference engine are used in NED-2. Other inference engines can be added easily.

Goal analysis agents use fuzzy rule sets to evaluate how well a stand or management unit satisfies management goals. These rules analyze a goal into desirable future conditions. Four fuzzy categories indicate how well a goal is met: fails, nearly passes, barely passes, and passes. NED-2 agents can perform goal satisfaction analysis on any set of snapshots representing the management unit at a point in time under a management plan.

NED-2 includes a knowledge model for wildlife habitat requirements based on research for the northeastern United States. Alternative wildlife models are being developed for the Great Lakes region and for the southeastern United States. Goal analysis agents will use a meta-knowledge base to select the appropriate wildlife model for a user.

Rule-based knowledge models will be used to predict changes in the understory and ground layer of stands over time. Causal models are not available for these layers at present, but it is essential to provide usable data for these layers as input to wildlife and other models. Members of the NED Core team are developing rules that will predict what these layers look like based on historical data and the properties of the overstory layer.

Many other knowledge models ranging in size from quite small to medium sized will be used in NED-2 in many places. For example, knowledge models will be used to evaluate the biological diversity and the "patchiness" of management units. Different agents in the system will know when and how to employ these knowledge models to complete tasks requested by other agents or by the user.

The output from NED-2 analyses can be displayed using a geographical information system. The GIS agent creates a temporary data file containing the information the user wants to view. Then the agent calls a Microsoft Visual Basic wrapper that starts the GIS (ArcView) and tells it to join the temporary data with a shape file the user provides. In the GIS, the user selects stand level variables (forest type, size class, satisfaction of some stand-level goal, etc.) to display on the map.

A user can examine a stand using the Stand Visualization System or examine the management unit using EnVision. The NED-2 visualization agent converts the data into the format required by the tool, evokes the external tool, and directs it to create visual displays representing the selected data.

NED-2 planning agents develop plans for completing complex tasks involving many subtasks. For example, a report-planner responds to the user's request that all selected reports be generated. The report-planner determines whether a baseline has been generated and whether any management plans have been simulated. It opens a dialog box where the user can select from the available management plans and years that can be used to build reports. Once a "view" (the baseline, or a management plan and a year) has been identified, the report-planner formulates a plan to perform any required goal analysis and calculations and to put the results into hypertext reports. This plan is placed on the blackboard and other NED-2 agents perform the steps in this plan.

5. CONCLUSIONS

The intended users of NED include all who are interested in management of forest land, principally those responsible for the individual management decisions on specific units of land. Case studies using the NED decision process on management units ranging from 85 to 15,000 acres have been initiated and some of these have been completed. In these studies, NED-1 is used and the functions of NED-2 not available in NED-1 are performed by an experienced NED-1 user. The approach in NED-2 is to provide independent agents using a blackboard architecture that can perform the integrating tasks of the NED-1 user in these case studies. This approach allows great flexibility while keeping the development effort efficient. This allows us to emphasize analyzing and displaying options for the decisions of the landowner or manager rather than ask NED to come up with "the right answer". Instead, users of the software gain a general understanding of their situation while using data collected from their woods to help analyze specific questions. The key is that through this approach we are able to help people consider multiple benefits and the tradeoffs among them.

NED-2 ver. 0.2 is now complete. It includes all user interface components except those for setting treatment parameters. This version can simulate management plans using the northeastern and southern variants of FVS, and it can perform goal analysis on simulated data for future years. Hypertext reports have been enhanced. GIS and visualization tools are not yet integrated. The entire blackboard architecture is complete. The target date for a beta version of NED-2 that performs all the functions described in this paper is September 2002.

The evaluation of NED-2 will use the case study method. Rauscher [2001] has completed an initial study using NED-1 for inventory analysis and for goal analysis. In this study, simulation and visualization were performed independently of the NED-1 system. Rauscher is engaged in additional case studies for management units ranging from a few hundred to a few thousand acres. He is currently moving the software base for these studies from NED-1 to NED-2. Results will appear in future publications.

6. ACKNOWLEDGEMENTS

Many more than 100 people have contributed to the development of NED software through participation on committees, testing of preliminary versions, or providing financial or moral support. Suffice it to say that without their hard work and dedication, NED and NED-2 would not have been possible.

7. REFERENCES

Crookston, Nicholas L. 1997. Suppose: An Interface to the Forest Vegetation Simulator. In: Teck, Richard; Moeur, Melinda; Adams, Judy. 1997. Proceeding: Forest vegetation simulator conference. 3-7 February 1997, Fort Collins, CO. Gen. Tech. Rep. INT-GTR-373. Ogden, UT: U.S.D.A., Forest Service, Intermountain Research Station.

McGaughey, R. 1997. Visualizing forest stand dynamics using the stand visualization system. In *Proceedings*, of the 1997 ACSM/ASPRS Ann. Conv. and Exp., April 1997, Seattle, WA, vol. 4, 248-257, Bethesda, MD; Am. Soc. For Photogrammetry and Remote Sensing.

McGaughey, R. 2000. EnVision – Environmental Visualization System, http://forsys.cfr.washington.edu/svs.html.

Ni, H. Penny. 1989. Blackboard Systems. In Avron Barr, Paul R. Cohen, and Edward A. Feigenbaum (eds.), *Handbook of Artificial Intelligence, Vol. IV*. Reading, MA: Addison-Wesley.

Potter, W. D., Liu, S., X. Deng, Rauscher, H. M. 2000. Using DCOM to support interoperability in forest ecosystem management decision support systems. *Computers and Electronics in Agriculture* 27: 335-354.

Rauscher, H. Michael, Lloyd, F. Thomas, Loftis, David L., and Twery, Mark J. 2000. A practical decision-analysis process for forest ecosystem management, *Computers and Electronics in Agriculture* 27: 195-226.

Rauscher, H. Michael. 2001. The NED Decision Process for Sustainable Forest Management. The Deer Hill Tree Farm Case Study. Final Report. Asheville, NC: U.S.D.A., Forest Service, Bent Creek Experiment Station.

Twery, Mark J., Rauscher, H. Michael, Bennett, Deborah J., Thomasma, Scott A., Stout, Susan L., Palmer, James F., Hoffman, Robin E., DeCalesta, David S., Gustafson, Eric, Cleveland, Helene, Grove, J. Morgan, Nute, Donald, Kim, Geneho, and Kollasch, R. Peter. 2000. NED-1: integrated analyses of forest stewardship decisions. *Computers and Electronics in Agriculture* 27: 167-193.

iEMSs

INTERNATIONAL
Environmental
Modelling
and Software
SOCIETY