December 1998

Extending DSS with Partial-Order Planning and Software Agents

Traci Hess
Virginia Tech

Extending DSS with Partial-Order Planning and Software Agents

Traci J. Hess
Department of Management Science and Information Technology
Virginia Tech

Introduction

The purpose of this research is to explore the use of partial-order planning (POP) and software agents as enhancements to Decision Support Systems (DSS). Current limitations in many DSS include the inability to integrate and manipulate models and a lack of automated maintenance. POP and software agents are proposed as solutions to these limitations in decision support.

Model management (MM) is a complex and relatively new research stream. As a result, a foundation for representing models has been laid, but a formal notion of model manipulation and integration does not exist (Dolk & Kottemann, 1993). Researchers have found that the lack of an integrated modeling environment makes models less accessible and comprehensible to organizations and impedes support for organization-wide planning, as models have a functional instead of a strategic orientation (Dolk & Kottemann, 1993).

The volume and type of information sources used by decision-makers has changed drastically over the past two decades. The DSS environment has consequently become quite complex, with many DSS encompassing distributed information sources and multiple stand-alone applications. The complex environment along with the rapidly changing dynamics of organizational decision-making result in extensive, time-consuming maintenance requirements.

Integrating Models with Partial-Order Planning

Research in MM over the past twenty years has primarily focused on a definitional form for models and a model base structure (Blanning, 1993; Dolk & Konsynski, 1984). First order logic, originating in the artificial intelligence (AI) community, has been largely accepted by MM researchers as a flexible representation for both model definition and manipulation. AI Planning, using first-order logic and state-space search techniques, provides a general tool for organizing and integrating models in response to decision-maker needs.

Recent advances in AI Planning resolve problematic issues with previous planners such as completeness of solution and intractable search spaces. POP as set forth in McAllester and Rosenblitt's Systematic Nonlinear Planner (1990) provides an effective means of selecting actions to formulate a plan in response to a stated goal. Within the MM component of a DSS, there exists a comparable process of selecting, sequencing and interfacing various models in response to the user's decision-making needs. POP appears to be well suited for the integration of models and the generation of decision-making alternatives needed within the MM subsystem.

The Role of Software Agents in DSS

The recent surge of interest in software agents has helped to renew interest in artificial intelligence applications but has yet to clarify the concept of agency. As a first step in exploring the role of software agents in DSS, a conceptual model and a tutorial-based explanation of agents are described. The conceptual model extends the views of agency set forth by Wooldridge and Jennings (1996), Franklin and Graesser (1996), and Gilbert (1997). Autonomy is defined and identified as the fundamental agent feature, while intelligence, mobility and interactivity are identified as enabling agent features (Hess, Rees and Rakes, 1998). The primary benefit of utilizing agents is recognized as the level of abstraction provided in the areas of interoperability and user interfaces (Bradshaw, 1997).

DSS provide an ideal environment for a business implementation of software agents (Hess, Rakes and Rees, 1998). The increasingly distributed nature of DSS and the heterogeneous applications that frequently exist or interact within a DSS result in a complex environment for both the DSS builder and user. Through encapsulation and decomposition, agents provide an abstraction for the complex DSS environment. Software agents utilizing intelligence, mobility and interaction abilities can provide responsive, proactive decision support.

Architecture and Implementation

In order to explore and generalize the usefulness of employing software agents in DSS, an architecture for an agent-integrated DSS has been designed. While DSS researchers have been quick to utilize agents within their implementations (Whinston, 1997), a general architecture for employing agents within the three subsystems of DSS has not been developed. The proposed architecture identifies the various types of agents used within the DSS components and is applicable to the DSS frameworks of both Sprague and Carlson (1982) and Bonczek, Holsapple and Whinston (1980).

An example DSS application will be implemented to further demonstrate the usefulness of agents in DSS and to provide an environment in which to study the effectiveness of POP as a model integration tool. The planning will be instantiated through software agents, and meta-planning agents will monitor and evaluate the overall planning effort. The implementation will document the outcome of using POP to support a flexible model management system and provide a test-bed for investigating various search-space pruning heuristics within the domain of DSS.

References

References are available upon request (thess@vt.edu).