Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 2000 Proceedings

Americas Conference on Information Systems (AMCIS)

2000

Computational Freedom in EISs

Bel G. Raggard
Pace University - New York, braggard@pace.edu

Montaceur Zaghdoud
ENSI de Tunis, mzaghdoud@ensi.tn

Computational Freedom in EISs

Bel G. Raggad, School of Comp. Sc. And IS, Pace University, braggad@pace.edu
Montaceur Zaghdoud, ENSI de Tunis, mzaghdoud@ensi.tn

Abstract

This article introduces computational freedom as a EIS development methodology that enforces independence among the EIS engine, its input processes, and output processes. Efficiency of the system stems from its capability of adopting the most cost-effective computing method available in the EIS engine.

Introduction

This article examines current configurations of the executive information system (EIS) technology, redefines its roles, and proposes a new design that adds value to executives' strategic decision processes. The new design will enforce value creation to executives by granting them adequate but feasible computational freedom.

Computational freedom permits executives to select any input, process, or output computing method that pleases them. The article however ensures a feasible computational freedom by controlling computing costs throughout all tasks constituting executive's computing sessions.

Input computing methods considered in this article concern three conceptual resources: noise, data, and knowledge. This study limits the input computing methods to 1) noise import, for the noise resource; 2) formatted data entry, 3) data import, for the data resource; and 4) membership functions, 5) belief functions, 6) frames, 7) If-Then structures, 8) random sets, and 9) training input patterns for the knowledge resource.

Computing methods, employed in output, are limited to 1) formatted screens; 2) reports; 3) graphic display; 4) confident recommendations; and 5) Neural Network classifications.

The process computing methods considered in this article are limited to 1) fuzzy sets; 2) Belief functions; 3) possibilistic reasoning; 4) Bayesian computing; 5) neural computing; 6) and genetic algorithms.

This article demonstrates the proposed design using a real-world application borrowed from an International Olive Oil production company.

Methodology

This section simply announces the methodologies employed in this article. Business Process Engineering (BPR) is adopted as the central methodology used to redesign the executive information system technology. The study also employs several computing methodologies within the executive information system's engine to enhance systems efficiency. Detailed presentations of individual methodologies will not be provided in this article, due to space limitation.

This article uses the BPR methodology to first examine executives' dissatisfaction with current configurations of executive information systems. It then examines those executive's strategic decision processes which are supported with an executive information system, and identify those system's engine components that cannot produce significant value to executives. These components will be redesigned so that the overall system enhances executive's satisfaction.

In redesigning EIS' processes, this article employs 5 main decision making methodologies, namely, 1) Bayesianism, 2) Belief theory, 3) possibility theory, 4) genetic computing, and 5) neural computing.

This study does not however intend to deal with the executive's mental model where the executive's problem is formulated, as in a Simon's decision process (Raggad 1996). In contrast to the work of decision support system environments where decision support is provided to the intelligence, design, choice, and review phases, the executive support system environment is, in fact, an executive information system environment where independent decision models are added to the personalized analyzes of the executive. That is, the executive support system environment does not necessarily contain a decision support system, but provides for independent decision models as required by the executive's personalized analyzes.

An overview on uncertainty management

Uncertainty is the lack of adequate information to make a decision. Unless reduced to an acceptable level, uncertainty can ruin any decision making process. This study organizes uncertainty management into three main areas: 1) Bayesian reasoning, 2) Dempster-Shafer theory, and 3) Zadeh's fuzzy theory.

Bayesian reasoning

Bayesian reasoning is a continuation of the classical probability theory which started as early as the 17th Century. Names that emerged with this probability theory, like Pascal, Fermat, and others, are still around. Probability applications touched almost every area of life, for example, business, economics, sciences, and engineering.

While probability, for some people, means simply a quantitative process that treats uncertainty, for others, it is a theory of games of chances that the real-world cannot be modeled with out it.

The most important invention that came from the classical probability theory, and the shining star that still guides the probabilistic world is Bayes' theorem, invented in the 18 Century by a British clergyman. This theorem is still useful in analyzing decision trees and modeling uncertain situations in all areas of decision support. Bayesian reasoning, known also as Bayesian decision making is still the most common approach employed in uncertainty management.

Of special interest to this literature review is the indifference principle frequently employed in Bayesian decision making. This principle, when called for, assigns equal probabilities for all possible outcomes when there is no evidence of support. This principle has been established in desperation for significant evidence.

Imagine, here I am watching television and suddenly get frightened by the breaking news of an earthquake in the far east. After looking around me, I immediately asked myself about the probability of the same happening in my neighborhood. In the absence of any evidence useful in discerning the possible outcomes, I have to assign .5 to the outcome of an neighborhood earthquake, and .5 to its negation. Anybody in this situation will soon realize that a .5 probability that an earthquake would take place in the neighborhood is a high one, and immediate evacuation is necessary.

This desperation has been at the origin of rethinking the classical probability theory and Bayesian decision making. This rethinking lead to the Dempster-Shafer theory.

Dempster-Shafer theory

This theory started by Dempster on his own attempting to model uncertainty by a range of probabilities instead of a single. Shafer then extended Dempster's work in a book of his own entitled "A Mathematical Theory of Evidence," (Shafer 1976).

Zadeh's fuzzy set theory

The first paper in fuzzy set theory dates back to 1965 with Zadeh's first paper. Fuzzy modeling, fuzzy logic, linguistic modeling, and some times, even approximative reasoning, inexact reasoning, or qualitative modeling may also refer to fuzzy theory.

Approximative reasoning related to decision support maybe implemented using (Zadeh 1978; 1979) inference procedures based on fuzzy production rules that are usually presented each as a two-part construct: antecedent and consequent of the rule. For example, if X1 and Y1 are fuzzy sets of the linguistic variables x and y, then "If x is X1, then y is Y1" is a production rule where "x is X1" is its antecedent and "y is Y1" is its consequent. For example, the fuzzy linguistic variable, software deficiency has the linguistic terms "high", "fair", and "low". These conditional statements describe inexact relationships among linguistic variables. Also, the inference process is characterized by fuzzy modus ponens since subjective judgment associated with IS effectiveness knowledge can only produce imprecise conclusions.

Let x_i, i=1,4 denote respectively the deficiency, cost, productivity, and documentation of an information system. Also, let a fifth input variable, x_5, represent the trend analysis, aggregating all base variables selected by senior developers. The output variable, y, is system effectiveness. All variables are assumed to have three linguistic terms, "low", "fair", and "high". Linguistic terms, "low", "fair", and "high", for x_i are denoted L_, F_, and H_, where the underscore character is replaced by the first letter of the fuzzy variable name.

The universes of discourse, U_i,i=1,5, and V, in which the fuzzy sets x_i,i=1,5 and y will have inexact boundaries are defined first. These fuzzy sets are associated with two properties: vagueness, characterized by the grade membership, and ambiguity, interpreted by the fact that two elements can belong to two different sets with two different grades of membership. Each fuzzy set x_i, in a universe U_i, is coupled with a membership or compatibility function m_{xi}: U_i -->[0,1]. That is, a fuzzy set x_i is the set of ordered pairs, {(u,m_{xi}(u)) in U_i}. An element u shows a partial membership in x_i when $0<m_{xi}(u)<1$ or nonmembership (full membership) when $m_{xi}(u)=0$ ($m_{xi}(u)=1$).

As in traditional set theory, fuzzy set operations are expressed in terms of compatibility functions. In particular, if C1 is a fuzzy set in U, the complement ¬C1 of C1, is the subset {u U u¬ C1) with a compatibility function $m_{\neg C1}(u)=1-m_{C1}(u)$. If C2 is also a fuzzy set in U, then the fuzzy set C1*C2 (* denotes fuzzy set intersection) will have as a compatibility function $m_{C1*C2}(u)=\mathrm{Min}\{m_{C1}(u),m_{C2}(u)\}$. The fuzzy set C1+C2 (+ denotes fuzzy set union) will have as a compatibility function $m_{C1+C2}(u)=\mathrm{Max}\{m_{C1}(u),m_{C2}(u)\}$.

Compatibility functions are constructed with two objectives in mind: (1) represent subjectivity of fuzziness as faithfully as possible, and (2) promote computational ease. As in Levy et al., 1991, and Bonissone, 1982, it is easier to use trapezoidal functions to represent all fuzzy sets of the problem.

Bayesian Environment

Probability is calculated based on frequency realization of an event A, where $A \subset 2^X$ and X = {x1, x2, …, xn} is the universe of discourse. This concept leads to relate the use of probability to the existence of historical data about a decision problem. This means that the executive , in case of using probability environment, should be more near to results and data base contents, and also he/she should be knowledgeable of frequency realization of each input event.

Possibility theory

Possibility theory introduced by Zadeh (1978), it tries to quantify uncertainty of events. It supposes that we can associate a possibility measure Π of uncertain events, where $\Pi : 2^X \rightarrow [0,1]$, such that:

X={x1, x2,…,xn} is universe of discourse,

$\Pi(\varnothing)=0$ and $\Pi(\Omega)=1$.

The dual of the possibility measure is the necessity measure N that: $N(A)= 1- \Pi(\overline{A})$ which can be interpreted as a degree of certainty of an event A.

These two set-functions can be compared to a probability function, as suggested by Zadeh himself, saying that an event should be probable prior to being possible. It leads to the following consistency condition (Dubois 1992):

$$\forall A, N(A) \le P(A) \le \Pi(A).$$

In case of a lack of information about a specific subject, rather than using probability measure, many executives prefer to use these two probability bounds: Necessity and Possibility. EIS should give an executive the freedom to chose his/her own preferred manner to enter or extract information.

Design of the possibilistic executive information system

This section presents the full design of the proposed executive information system. Specifications for various components of the system are discussed in greater details.

Design of the PEIS

The EIS is designed as a computer-based information system with 6 subsystems, as follows:

1. a dialog generation management subsystem (DGMS)
2. a task manager (TM)
3. a model base subsystem (MBS)
4. a computing shuttle (CS)
5. off-line benchmark analyzer (OBA)
6. high-level granularity data warehouse (DWH)

Dialog generation management subsystem

The dialog generation management subsystem is a double interface subsystem. The executive can initiate consulting sessions when strategic support is needed, or 2) he or she can, when desired, add a favorite computing method or delete a resented one.

The executive information system administrator (EISA) is in charge of managing the EIS. Executive's needs should be promptly added to the EIS.

EIS Environment

When the executive activates the system, the system displays a menu containing all available options, namely, 1) Inquiry, 2) Analysis, or 3) Decision. The executive makes a selection. The selection will activate a dialog generation management system (GDMS), as in Sprague and Carlson's DSS generators and as in Raggad (1996).

Even though, the DGMS is an interactive subsystem, the PEIS treats all activities as sequential tasks of three types: 1) Input, 2) Process, or 3) Output.

Model base subsystem

All computing methods, whether they are input, output, or process methods, and whether they are passage or reasoning computing methods, reside in the model base subsystem.

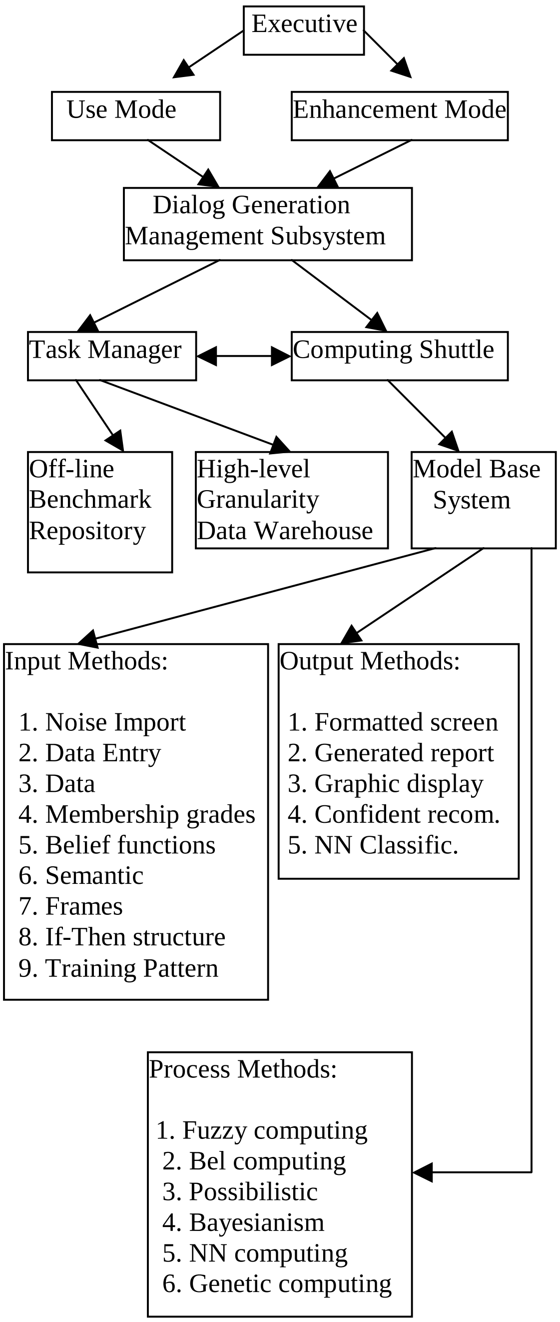

Figure 1: New EIS Environment

Task manager

The task manager 1) receives an executive's input, process, or output computing method; 2) interpret it; and 3) processes it. The task manager requires the cooperation of two system components, namely, the model base, and the computing shuttle. The computing shuttle carries executives' computing activities, defining the task being processed, from one model of the model base subsystem to another, until executive's desired outputs are fully satisfied.

Computing shuttle

The computing shuttle is a computer program responsible of system efficiency. The shuttle buys computing power from the computing method that offers a lower price and a better quality process, as requested by the executive's task. The price is controlled using an off-line benchmark repository.

The selection of a computing method is controlled using two main attributes, 1) executive's familiarity with the computing method, and 2) computing cost.

High-level granularity data warehouse

The level of details or summarization held in the units of data in a data warehouse is called granularity. The more summarized the information is, the higher the level of granularity.

The most efficient fashion an executive can understand internal matters, affecting strategic planning, is not by accessing all corporate databases, as most current executive information systems do, but by accessing high granularity levels of the corporate data warehouse. Our new EIS environment borrows this high granularity level feature, a very important design aspect, from the data warehouse technology.

The EISA is responsible of contacting the corporate data warehouse administrator to arrange for a periodic off-line processing that produces all strategic information needs pertaining to the firm's operational system.

Benchmark repository

The purpose of the off-line benchmark repository is to provide for testing and validating system computing methods and the efficiency of traveling through them.

Test cases are defined using past situations in the internal EIS environment or borrowed externally from similar business sectors. The benchmark repository should be periodically tested for completeness.

A set of common problems are solved using all methods for which results are compared and stored. Results consist of three variables, namely, the name of the method, it response time, and its cost. Cost is modelled as a function of time and data requirements. Table 1 which shows an example of a benchmark table is omitted due to space limitation. The reader who is interested in obtaining the table may request it from the author(s).

The benchmark repository includes data concerning computing methods, and passages among them. Table 2 which provides benchmark table for passage methods evaluation may be obtained for the author(s).

A benchmark is a comparison test performed on all computing methods that are used to solve a sufficiently large sample of similar problems. Results may be stored on-line and consulted by the computing shuttle prior to invoking the most appropriate computing methods.

Computational freedom

Computational freedom uses the computing shuttle to provide profitable computing features accommodating the executive management style. The study attempts to grant feasible computational freedom at all operations defining the executive information system's input, process, and output.

Executives' verbal and written interactions are of the linguistic type. That is, all reports produced by the management committee or various functional units directed by an executive are of the linguistic type. In fact, whether your are looking at management with Fayol's eyes or Mintzberg's eyes, upper management deal with a great deal of qualitative judgment. Fayol's managerial functions, and Mintzberg's managerial roles are fully qualitative. This means that, not only their interactions are of the linguistic type, but also their reasoning.

Executives' processes may also employ other conceptual resources like noise, data, or information obtained from their company databases and data warehouses or other external sources.

They may also initiate personalized analyzes that call for mathematical, statistical, and simulation models. Or they may employ intelligent agents, genetic, or neural programs. Computational freedom has to provide for all these computing methods.

Feasibility is enforced 1) by advising the executive on what input and output tasks should be initiated, 2) by guiding the executive information system's engine in invoking cost-effective computing models available in the system's model base, and 3) by providing, when possible, useful passages among computing methods in order to hide from executives any complex processes the system's engine may activate.

Input Management

Input is modeled as in Raggad (1997) to be a conceptual resource that is processed by the computer-based information system to produce information in support of managerial decision processes. A conceptual resource can be noise, data, information, or knowledge. Noise consists of those environmental raw facts that do not show a known code system. This conceptual resource requires intensive filtering, as in Wang and Turban (1991), before it can be useful for strategic decision support.

The data resource consists of those raw facts that belong to known code systems that provide useful information when processed. Knowledge represents natural forms that can be immediately elicited or interpreted by executives and human experts without the need of major transformation. These natural forms are associated with little human processing and null Bayesian update as showed in Raggad (1996) and Dewan 1992.

Computing methods in input management depend on the type of the conceptual resource. Table 4.3 which provides input computing methods in terms of input types may be obtained from the author(s).

The computing shuttle uses a decision table that identifies the most efficient input computing method in terms of 1) attributes characterizing the executive's task and 2) current benchmark results. Attributes considered in the selection process consist of the following:

1. executive's familiarity with the computing method
2. availability of the data required by the input method
3. size of input

Input elicitation of all input methods available in the EIS environment are presented in a later section, that we intentionally leave out due to space limitation.

Process management

The computational methods considered in this study are limited to the following:

1. Fuzzy computing
2. Bel computing
3. Possibilistic computing
4. Bayesian computing
5. NN computing
6. Genetic computing

The computing shuttle uses a decision table that identifies the most efficient computing method in terms of 1) attributes characterizing the executive's task and 2) current benchmark results. Attributes considered in the selection process consist of the following:

1. availability of the input
2. cost assigned by the benchmark program

At the time of selection, the passage between a computing method and another should not be taken into account when the two computing methods are compared in terms of efficiency. Otherwise, the output-process independence principle will be violated.

Output management

Computing methods employed in output management are limited to the following:

1. Formatted screen
2. Generated report
3. Graphic display
4. Confident recommendations
5. NN classification

The computing shuttle uses a decision table that identifies the most efficient output method in terms of 1) attributes characterizing the executive's task and 2) current benchmark results. Attributes considered in the selection process consist of the following:

1. executive's familiarity with the output method
2. cost assigned by the benchmark program

The passage between an output method and another should be taken into account when the two output methods are compared in terms of efficiency, so a selection decision can be made.

Does this new EIS environment solve problems of current configurations of EISs

The reported cost of an EIS in the private sector varies between $1 million and $2 million dollars (Kuehn and Fleck Jr. 1991). Small, medium and public sector organizations with fixed budgets are very cost-sensitive and cannot therefore invest in a costly project such as developing an EIS.

This problem cannot take place in our EIS environment due to the use of the computing shuttle. Also, because most incurred costs originate at the direct access of on-line corporate databases, not allowed in our new EIS environment, the development cost considerably goes down.

Functional units, in one organization, are organized differently and hence have different strategic necessities. Because functional executives operate in different environments, their information requirements demand different strategic support. Replicating the project of an EIS designed for top management or a different functional unit may not produce an efficient EIS for the very functional executive. That is, an EIS may be designed distinctly for the executives of marketing and manufacturing.

This problem cannot take place in our new EIS environment.

Organization strategic activities can only be partially computerized because of their unstructuredness, complexity or non-profitability. Furthermore, computer-based support for strategic decision making is only effective when generated messages demonstrate relevance, validity, accuracy and timeliness.

This problem cannot take place in our new EIS environment, because it enforces computational freedom.

Executives obtain intelligence by informally scanning the external environment to gather raw data transformable into interpretable data that is useful to top management. Only those data that resist sequential data filtering of the EIS will be considered in playing a role in strategic decision making. Current EISs do not provide full support, especially for cost-sensitive organizations. EISs also cause prolonged delays in providing useful information to top management. Delays generally occur when data pass through EIS filters. Wang and Turban (1991) studied five filters: the organized scanning, perceptual, power, interpretation and communication filters.

These deficiencies cannot take place because of the flexibility in input, process, and output.

The new EIS environment allows executives to select their own input methods, output methods, and process computing methods without disrupting troublesome adversity effects . Executives will choose input, process, and output methods that they trust and are familiar with. They will be better equipped with all the methods they need. They will have the freedom to move from one method to another seeking ease, confidence, and efficiency. The new EIS environment, as show in Figure 1, is characterized by an open system where the current executive or a successor can add their favorite computing methods and delete the disliked ones.

Innovative Features characterizing the proposed EIS environment:

1. Input-process independence
2. Input-output independence
3. Output-process independence
4. Portability
5. Computational freedom

Computational support for the environment translator

This section presents all heuristic and mathematical procedures used by the computing shuttle for the purpose of allowing computational freedom among various input, process, and output computing methods available in the system's model base.

Unfortunately, due to space limitation we are unable to present all the passages from one computing method to another. The full paper that contains the details of the study may be obtained from the author(s).

Conclusion

This article examined current configurations of the executive information system (EIS) technology, redefined its roles, and proposed a new design that can add value to executives' strategic decision processes. The new design will enforce value creation to executives by granting them adequate but feasible computational freedom.

Computational freedom permits executives to select any input, process, or output computing method that pleases them. The article enforced the feasibility of computational freedom by controlling computing costs throughout all tasks constituting executive's computing sessions.

References

Dewan, R.M. An Economics Perspective on the Usefulness of Decision Support Systems, Decision Support Systems, (8), pp. 293-923, 1977.

Dubois, D., Fuzzy Set Connections as Combinations of Belief Structures, Information Sciences, (66), pp. 245-275, 1992.

Frederiks, P. and N. Venkatraman, The Rise of Strategy Support, Sloan Management Review, (47), Spring 1988.

Kuehn, R. and R.A. Fleck Jr., Implementing an EIS in a Large Insurance Company, Journal of Systems Management, January 1991.

Kotter, J. What Effective General managers Really Do, Harvard Business Review, (60), Nov-Dec 1982, pp. 156-167.

Mintzberg, H., The Nature of Managerial Work, New York: Harper & Row, 1973, p.47.

McLeod Jr., R. and J.W. Jones, Making Executive Information Systems More Effective, Business Horizons, (5:29), Sep/Oct 1986.

Milkovich, G.T. and J.W. Boudreau, Human Resource Management, Six Edition, Irwin, 1991.

Mohan, L., Holstein, W.K. and R.B. Adams, EIS: It Can Work in the Public Sector, (14:2), MIS Quarterly, Dec 1990.

Mockler R.J., Computer Information Systems and Strategic Corporate Planning, Business Horizons, (5:30), May/June 1987.

Raggad, B. and L. Boumediane, Functional Executive Information Systems, Proceedings of IABD, 1996.

Shafer, G.A., A Mathematical Theory of Evidence, Princeton University Press, N.J. (1979).

Wallis, L., Power Computing at the Top, Across the Board, Jan/Feb 1989.

Wang, P. and E. Turban, Filtering Strategic Environmental Information Processing Using an EIS, IEEE, pp. 123-129, 1991.

Zadeh, L.A., Fuzzy Sets as a Basis for a Theory of Possibility, Fuzzy Sets and Systems, (1), pp. 3-28, 1978.

Zadeh, L.A., A Theory of Approximate Reasoning, Machine Intelligence, (9), pp. 149-194, 1979.