Tilburg University

Characteristics and implementation of decision support systems

Casimir, R.J.

Publication date:
1987

[Link to publication in Tilburg University Research Portal](#)

Citation for published version (APA):

Casimir, R. J. (1987). *Characteristics and implementation of decision support systems*. (Ter discussie FEW; Vol. 87.06). Unknown Publisher.

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

CBM
R

7627
1987
6

POSTBUS 90153
5000 LE TILBURG

FACULTEIT DER ECONOMISCHE WETENSCHAPPEN
REEKS "TER DISCUSSIE"

**CHARACTERISTICS AND IMPLEMENTATION
OF DECISION SUPPORT SYSTEMS**

Rommert J. Casimir

No. 87.06

R 8

T decision support system

Characteristics and Implementation
of Decision Support Systems *

Rommert J. Casimir

May 1987

* Working Conference on Decision Support Systems
Lisse 3/4 June 1987

Tilburg University (Katholieke Universiteit Brabant)
P.O. Box 90153
5000LE Tilburg The Netherlands
R974CASI@HTIKUB5.BITNET

Summary

Three types of information systems related to DSS's, which in this paper are named genuine DSS's, are defined: The traditionally built DSS, the MIS prototype and the spurious DSS. The latter uses DSS tools and techniques for other purposes than decision support, especially for control. It is shown that a number of features that facilitate DSS building can be used profitably in building genuine DSS's only.

1 Introduction

In the literature, Decision Support Systems (DSS's) are distinguished from other information systems in three ways [2]. First, the purpose of a DSS is to help managers in decision making. Originally, the application of DSS's to strategic, unstructured or semi-structured decisions was stressed [13], but recently, this emphasis has been given less attention [29,19]. Second, a DSS uses sophisticated modeling techniques. Third, a DSS is a flexible system which can be designed and changed at short notice and at low cost. To this end specialized DSS tools are used. DSS texts tend to compare DSS's with "traditional EDP systems", lacking all three characteristics. Such comparisons often make two apparently conflicting claims: On the one hand, DSS's contain more advanced features and consequently give more relevant information to management than traditional EDP systems; on the other hand, DSS's can be built with much less effort. Sometimes these claims are pushed so far that the reader gets the impression that only a conspiracy between hardware vendors and the COBOL programmers union is holding back the total supersession of traditional EDP by DSS's [26]. To put such comparisons into proper perspective, in section two of this paper, the three characteristics mentioned will be used as a base for a classification of information systems. Subsequently, DSS's will be compared with information systems lacking only one characteristic. In section three, it will be shown that the characteristics of a DSS are intimately related. Specifically, it will be indicated why DSS techniques will not result in the same productivity gains when employed outside the realm of genuine DSS's.

2 Classification of information systems

For our classification we introduce some new terminology. An information system aimed purely at helping to make decisions is called a user DSS. An information system using sophisticated modeling techniques is called a logical DSS. An information system that is built with the aid of specific DSS tools is called a DSS implementation. This results in the classification given in table 1:

Table 1: DSS and related systems.

	User DSS	No user DSS
Logical DSS and DSS implementation	Genuine DSS	Spurious DSS
Logical DSS	Traditionally built DSS	Traditional modeling system
DSS Implementation	MIS prototype	Information system prototype
Neither logical DSS nor DSS implementation	Traditional MIS	Traditional EDP system

To better understand the characteristics of a genuine DSS, we compare it with the information systems nearest to it, i.e. the traditionally built DSS, the MIS prototype and the spurious DSS.

2.1 Genuine DSS versus traditionally built DSS

"Traditionally built" encompasses design and programming in a general purpose programming language such as Fortran or Pascal by a single analyst as well as design and implementation by a group of professionals according to a generally accepted system development method such as ISAC [22]. In contrast, a genuine DSS is built with specific DSS tools and generators. However, the dividing line is blurred, because APL, which certainly is a general-purpose language, is often cited as a powerful DSS tool [24]. Language and system development method choice belong to the discipline of software engineering [5].

There is some empirical and experimental research showing large productivity from fourth generation languages [12,14], which, however, does not pertain explicitly to DSS problems, but in general, the study of factors influencing programmer productivity is still in its infancy [18], which leaves this field to prophets of diverse faith such as Dijkstra [11] and Martin [23], whose sermons reach beyond the DSS community. As a consequence, we see no reason to exclude traditionally built DSS's from the DSS field. Of course, DSS practitioners and students should have a firm command of DSS tools, but they should also recognize that some circumstances, such as low cost of programming or scarcity of computer resources, will favor building DSS's with other techniques.

2.2 Genuine DSS versus MIS prototype

Traditional financial statements have long been the most important sources of information for managers. DSS tools, such as System W, are well suited to produce such statements.

The possible superiority of genuine DSS's over information systems producing financial statements as a base for decision making is part of a well-established subject of research, especially experimental research involving management games [4,6,8,10,16,17]. As the superiority of advanced techniques as a base for decision making has not yet been proven, there is no reason to exclude MIS prototypes from the DSS field.

2.3 Genuine DSS versus spurious DSS

The main purposes of information systems other than helping to make decisions are establishing rights and obligations of the organization and exercising control over subordinates. The first purpose is overwhelmingly present in salary systems, that involve payments to government and social security agencies to establish rights that sometimes will be exercised only after decades.

The need for control information stems from limited responsibility. A subordinate is not responsible for errors caused by executing his superior's orders, but he has to prove by punctilious accounting that he did indeed execute those orders. For example, a buyer who is instructed to buy from a specified supplier, must prove he bought from that supplier, but it is not his concern if this supplier is more expensive than competitors.

We define a spurious DSS as an information system with the logical and implementation characteristics of a genuine DSS, that is used for other purposes than helping the manager make the decisions he used to make before. We distinguish three such purposes: Establishing rights or obligations of the organization, increasing control over the manager's actions, and delegation of authority to subordinates.

2.3.1 Spurious DSS's establishing rights and obligations

Typically, a new organization has a large degree of freedom in making decisions. However, as clients become more sophisticated in adapting to organizational decisions, they ask for better predictions. As knowledge of the decision process and the information used in it is the best base for predicting a decision, clients will tend to push for publication of Decision Support Systems. A typical example is student grading, where teachers presently have to apply strict averaging rules. As a result, a score awarded to a student is an obligation and not an item to be used freely in a later decision. This change of character diminishes the value of the information system for decision support because the ability to improve decision rules during the data collection phase is forfeited. Moreover, system costs will soar because of the increased need for security, and clients will adapt to the rules defined in the information system rather than the intentions of the organization. The threat of this evolution, which was named "Migration" by Moore and Chang [25], may dominate the value of a genuine DSS, thus stalling DSS activity in such fields as personnel.

Similarly, many industries face increasing government regulation with a corresponding increase in the amount of data that has to be supplied. A DSS adding to or interpreting those data is always in danger of being appropriated by the regulatory agency.

2.3.2 Spurious DSS's for increased control

The control view on organizations differs from the management science view adopted in the DSS literature in its treatment of deviations from the average. In the management science view, deviations are considered noise without significance for averages; in the control view, they are aberrations from a norm that have to be corrected lest the average should be influenced.

Moreover, it is assumed that deviations will grow when control is loosened. The control view has a strong influence on management practice, as demonstrated by large outlays for control systems, the high esteem for auditors and lawyers, and occasional attempts to use strict controls on petty expenses to encourage a general atmosphere of sparsimony, but it is systematically underrated in the information systems literature. With regard to taxation, the management and control view may be called the macroreconomic view and the IRS view, respectively; it is significant that political managers adhere to the IRS view when they cite tax fraud as an explanation for budget deficits. The conflicting views are also apparent in the census debates in some European countries where opponents of the census implicitly assume that with regard to the census, authorities adhere to the control view rather than the management science view.

The large gains that are possible from increased control favor the use of DSS's for this purpose. As a first step toward this end, use of the DSS is made mandatory, possibly after publishing some phony research results on the effectivity of the DSS. Next, management control may be increased in four ways: First, the DSS may restrict the choices a manager may make. Second, the DSS may operate on a prescribed input set, thus focusing the manager's attention on a distinct set of variables that is not of his own choice. Third, the DSS may monitor input and output of the DSS as well as deviations of DSS advice by the manager. Fourth, the DSS may monitor variables entirely unrelated to the decisions at hand, e.g. attendance. In this vein, the DSS input module might even be used as a lie-detector by registering answers to such questions as "Do you think the applicant is charming?". However, it is to be expected that managers will quickly adapt to the use of a DSS for control by supplying inputs that produce the desired results. This can be countered by higher level management by auditing inputs, but then the notion that such a system is a DSS should be shed altogether.

Just as DSS's introduced by higher level management will be viewed with suspicion, managers may refrain from developing DSS's they need for fear of appropriation by higher level management. In this respect, the attitude of managers to superiors is the same as the attitude of top management towards customers and government: It may be preferable not to collect information that could do harm if accessible to the opponent.

2.3.3 Expert systems as spurious DSS's

We define an Expert System (ES) as a computer program that advises a decision maker on a decision that would not normally be made by him. This definition abstracts from the internal architecture of an ES that is normally put forward as a characteristic feature of Expert Systems, but it is consistent with the intent of an ES as defined in the literature [30]. If the decision making rules of a manager are codified in a DSS and operation of this DSS is entrusted to a subordinate, this DSS is an ES. It is also a spurious DSS, because it changes the decision making process. Especially, codification diminishes the flexibility of decision rules. However, in contrast to spurious DSS's forced onto the manager who previously made his own decisions, ES will be welcomed by users because their responsibilities are widened, at least apparently.

3 Characteristics of genuine DSS's

When comparing genuine and spurious DSS's, it should be noted that the value of information, which is a well-defined concept in the context of decision making, e.g. in [20] and in decision analysis [21], lacks a similar analysis in the context of control. Even for the generally accepted assertion that business applications need high accuracy in number representation [7], no explicit theoretical support can be found in the literature. Consequently, it is assumed that information systems used for control, which will henceforth be called Control Information Systems (CIS), always need full, detailed and accurate data, and it will be argued that genuine DSS's need not.

3.1 Accuracy

3.1.1 Impact of accuracy

As a rule, a better decision may be made when more accurate information is available. An illustrative example is found in the "newsboy problem" [20,21], where a newsboy has to decide on the number of copies of a newspaper he should buy, given his expectations on the number of copies he will sell. It is clear that, in this case, where unsold copies cause an immediate loss, accurate information has a high value. However, the problem is not representative, because in many circumstances, it is possible to compensate for the lack of information by other means, e.g. by carrying stock or by reserving spare capacity. In that case, the final result will be far less influenced by the accuracy of information. A pertinent example, also cited in [20], is the computation of the economic order quantity (EOQ), where a 100% error in the estimate for order costs entails only a 6% error in total costs.

In the design of a DSS the fact that less accuracy is needed than in a CIS can be used in various ways:

3.1.2 Data types

All numeric variables in a DSS may be represented by a single data type, real or floating point numbers. This diminishes program size, both because of the absence of data type definitions and because of the smaller number of data type conversions needed, and the size of the programming language, because of the small number of data types and the absence of duplicate functions. As a corollary, the use of single numeric data type has held DSS languages with a large number of functions within reasonable limits.

3.1.3 Data structures

The use of samples instead of full data, which has been studied extensively in statistical theory, entails a lower data volume, which implies that complicated data structures, intended to save memory space and retrieval time, such as trees, can be replaced by simple data structures, such as arrays. An array is a simple data structure because simple procedures may be used for common operations, such as extracting a row or a column. Moreover, arrays can be transported from external to internal memory and from memory to screen without conversion. It is significant that many tools that have been successfully used in DSS design, such as APL, spreadsheets, and model building languages such as System W, use arrays as its principal data structure. An additional advantage of the use of arrays is that mathematical models are often defined in matrix format.

3.1.4 Procedures

As a consequence of the low data volume, procedures need not to be designed for maximum speed. The trade-off between program size and execution speed is a well known subject in elementary programming texts, but there are few experimental or empirical data on the relationship, even for such well-studied programs as optimizing compilers [1].

3.1.5 Organization

The low priority of efficiency in DSS design implies there is less need to divide the design work between a designer who concentrates on the model and a programmer who concentrates on implementation efficiency. Such a division of labor entails an increase in the total time involved in the design because of communication and waiting costs, which will not be fully compensated by the lower salary and the higher productivity of the programmer.

3.1.6 Reliability

An information system is reliable if there is no danger of accidental loss or corruption of data. Reliability is provided by redundancy, e.g. by duplication or by the use of error-correcting codes. In a genuine DSS, a data item may be considered an item in a sample, so its loss merely reduces sample size, with a concomitant loss of accuracy. This diminishes the demand for reliability, with a corresponding decrease in cost.

3.2 Security

3.2.1 Impact of security

Security of an information system implies reliability, but it also entails protection against illegal data modification and protection against unauthorized reading. Illegal actions of the first type threaten information systems that record rights and obligations and information systems used for control, rather than genuine DSS's. In an information system that records rights and obligations, an intruder may directly profit from a modification in information system data, e.g. when he changes the balance of his bank account. In an information system used for control, a modification may cover up an illegal transaction that otherwise would be detected by comparing data on the physical state with information system data. A DSS offers less scope for fraud because the impact of a modification in DSS data on decisions is not known. Generally, results from a DSS will be compared with data from other sources before a decision is actually implemented.

On the other hand, the danger of unauthorized reading pertains equally to DSS data and data in other information systems. It is acute in competitive industries such as oil prospecting and contracting.

3.2.3 Costs of security

Papers on security tend to present methods available to attain security, rather than risk assessment [15,27]. Moreover, writers on this subject often are of the opinion that in EDP operations not enough attention is given to security [9]. This makes it difficult to assess the gains that can be reaped from lowering security standards for DSS's. However, in view of the length of the lists of security measures recommended for secure systems, they should be considerable.

3.3 Data collection

A DSS may use data extracted from EDP systems. This simplifies design by excluding the data collection phase. Use of generalized Data Base Management Systems simplifies data extraction still further. However, appropriate data for a DSS will not always be found in a database maintained for other purposes. For example, EDP systems for utilities center on usage data for individual consumers, which do not supply information on peak loads needed for investment decisions. In such a case, there are two choices for the DSS designer: either the appropriate data are included in the general database, or a separate data collection scheme is started. Contrary to what might be expected, the first alternative may be more expensive, as data included in the general database will have to satisfy much higher requirements with regard to completeness and reliability. Moreover, once the data have been collected, an opportunity for using them in a CIS may be seen, and this may even get priority over DSS building. On the other hand, various techniques may be used to capture DSS data. One is sampling, another may be called analog aggregation. It is used when a variable which is functionally dependent on the sum of a number of variables, can be measured in nature. Examples are measuring total output of a power plant, determining the number of supermarket visitors by aerial photography of parking lots and estimating working hours from coffee consumption. From the last example it is clear that analog aggregation may be useless for control: many a worker would gladly drink an extra cup of coffee for his absent colleague.

3.4 Models

Models have such a wide use in DSS that modeling systems are sometimes thought to be synonymous with DSS. A model adds an interpretation to a set of data. Thus both continuous and discrete simulation models map states over time. Models contain rules that define relations between variables. A simple rule is a triple $\langle x, y, f \rangle$, where x and y are variables and f is a mapping of x into y . Variables x and y may be measured; mapping f is derived from logic or operations research. In a DSS it suffices to measure x and compute $y=f(x)$. In a control system $y'=f(x)$ is compared to the recorded value of y . A difference signals the necessity of specific control actions. The equation $y=f(x)$ is called an invariant in the theory of program correctness [3], and an integrity constraint in database theory [28]. As the difficulty of a programming task increases with the number of invariants, programming a genuine DSS is easier than programming a CIS using the same model.

Example

The following rule is valid for any organization:

$$\text{cash}_i = \text{cash}_{i-1} + \text{receipts}_i - \text{expenses}_i$$

As the value of cash_{i-1} is supposedly known, application of this rule in a DSS implies that inputs have to be given for only two of the three variables cash_i , receipts_i and expenses_i . Of course, this method has been followed by shopkeepers since times immemorable. In a CIS, all three variables have to be recorded to detect theft as well as counting errors. Cash registers and point of sales systems (POS) diminish the cost of recording the additional variable, which makes sense in a CIS, rather than a DSS context.

The notion that extensive use of models, facilitates, rather than impedes DSS design may come as a surprise to some practitioners, but it is in line with a long tradition in statistics and OR. Once this is recognized, the details of the actual implementation are less important. Users of DSS tools and generators may rely on built-in functions, and designers of traditionally built DSS's may use generally available subroutine libraries.

4 Conclusions

The views on genuine and spurious Decision Support Systems presented in this paper have different implications for DSS research, DSS education and DSS practice.

DSS research should center on determining how information systems affect decision making. This implies it should have solid foundations in management science, an opinion also voiced in [19]. Alongside with DSS's, other types of information systems should be studied in their own right, as this field deserves more fundamental research, rather than contempt from DSS researchers.

For education in DSS, the direction is clear. A DSS curriculum should be largely made up from Management Science and Computer Science courses, crowned by assignments in DSS design.

DSS practitioners give more attention to the real purpose of the systems they are building and the impact of that purpose on design. Some will find themselves in the business of building spurious DSS's. They will be excused if they continue to call their work DSS design, but this should have no influence on DSS research and education.

References

- 1 Aho, A.V., R.Sethi and J.D.Ullman: Compilers: principles, techniques and tools. Addison-Wesley, Reading Mass., 1986.
- 2 Alter, S.L.: Decision support systems: current practices and continuing challenges. Addison-Wesley, Reading Mass., 1980.
- 3 Apt, K.R.: Ten years of Hoare's logic. ACM Trans. on Progr. Lang. and Syst. Vol. 3 No 4 (Oct 1981), pp 431-484.
- 4 Bell, J.: The effect of presentation form on the use of information in annual reports. Management science Vol 30 no 2 (feb 1984), pp 169-185.
- 5 Boehm, B.W.: Software engineering economics. Prentice-Hall, Englewood Cliffs, 1981.
- 6 Benbasat, I. and R.G. Schroeder: An experimental investigation into some MIS design variables. MIS quarterly Vol 1 No 1 (march 1977), pp 37-50.
- 7 Borland International: Turbo Pascal reference manual version 3.0. Borland International, Scotts Valley Calif., 1985.
- 8 Courtney, J.F., G. DeSanctis and G.M. Kasper: Continuity in MIS/DSS laboratory research: the case for a common gaming simulator. Decision sciences Vol 14 No 3 (july 1983), pp 419-439.
- 9 Courtney, R.H. and M.A. Todd: Problem quantification - its importance to cost-effective computer security. in: J.B. Grimson and H.J. Kugler (eds): Computer security: the practical issues in a troubled world. pp 55-63. Amsterdam, North Holland, 1985.
- 10 Dickson, G.W., J.A. Senn and N.L. Chervany: Research in management information systems: the Minnesota experiments. Management science Vol 23 No 9 (may 1977) pp 913-923.
- 11 Dijkstra, E.W.: The humble programmer. CACM Vol 15 No 10 (October 1972), pp. 859-866.
- 12 Forage, G.: Fourth generation languages and advanced software development aids. Data Processing Vol 27 No 9 (Nov 1985), pp 6-8.

- 13 G.A. Gorry and M.S. Scott Morton: A framework for management information systems. Sloan management review, Vol 13 No 1 (fall 1971), pp 56-70.
- 14 Harel, E.C and E.R. McLean: The effects of using a nonprocedural computer language on programmer productivity. MIS Quarterly Vol 9 No 2 (June 1985), pp 109-120.
- 15 Hsiao, D.K, D.S. Kerr and S.E. Madnick: Computer security. Academic press, New York, 1979.
- 16 Jarvenpaa, S.L., G.W. Dickson and G. DeSanctis: Methodological issues in experimental IS research: experiences and recommendations. MIS quarterly Vol 9 No 2 (June 1985), pp 141-156.
- 17 Kasper, G.M. and R.P. Cervený: A laboratory study of user characteristics and decision-making performance in end-user computing. Information and management 9 (1985) pp 87-96.
- 18 Kearney, J.K., R.L. Sedlmeyer, W.B. Thompson, M.A. Gray and M.A. Adler: Software complexity measurement. CACM Vol 29 No 11 (November 1986), pp 1044-1050.
- 19 Keen, P.W.: Decision support systems: the next decade. in: McLean, E.R. and H.G. Sol (eds): Decision support systems, a decade in perspective. North Holland, Amsterdam, 1986.
- 20 Kleijnen, J.P.C.: Computers and profits. Addison-Wesley, Reading Mass., 1980.
- 21 LaValle, I.H.: Fundamentals of decision analysis. Holt, Rinehart and Winston, New York, 1978.
- 22 Lundeberg, M., G. Goldkuhl and A. Nilsson: Information systems development, a systematic approach. Prentice-Hall, Englewood Cliffs, 1981.
- 23 Martin, J.: An information systems manifesto. Prentice Hall, Englewood Cliffs, 1984.
- 24 McLean, E.R. and T.F. Reasing: Installing a decision support system: implications for research. in: Fick, G. and R.H. Sprague (eds): Decision support systems: issues and challenges. Pergamon press, Oxford, 1980.

- 25 Moore, J.H. and M.G. Chang: Meta design considerations in building DSS. in Bennet, J.L. (ed): Building decision support systems. Addison-Wesley, Reading Mass., 1983.
- 26 Thierauf, R.J.: Decision support systems for effective planning and control.
Prentice-Hall, Englewood Cliffs, 1982.
- 27 Tompkins, F.G. and R. Rice: Integrating security in the software development life cycle. in: J.B. Grimson and H.J. Kugler (eds): Computer security: the practical issues in a troubled world. Amsterdam, North Holland, 1985, pp 65-106.
- 28 Ullman, J.D.: Principles of database systems.
Pitman, London, 1980.
- 29 Van Groenendaal, W.J.H.: Towards a workable definition of DSS. 8th European conference on operations research, Lisbon, Portugal, sept 1986.
- 30 Waterman, D.A.: A guide to expert systems.
Addison-Wesley, Reading Mass, 1986.

IN 1986 REEDS VERSCHENEN

- 01 F. van der Ploeg
Monopoly Unions, Investment and Employment: Benefits of Contingent Wage Contracts
- 02 J. van Mier
Gewone differentievergelijkingen met niet-constante coëfficiënten en partiële differentievergelijkingen (vervolg R.T.D. no. 84.32)
- 03 J.J.A. Moors
Het Bayesiaanse Cox-Snell-model by accountantscontroles
- 04 G.J. van den Berg
Nonstationarity in job search theory
- 05 G.J. van den Berg
Small-sample properties of estimators of the autocorrelation coefficient
- 06 P. Kooreman
Huishoudproductie en de analyse van tijdsbesteding
- 07 R.J. Casimir
DSS, Information systems and Management Games
- 08 A.J. van Reeken
De ontwikkeling van de informatiesysteemontwikkeling
- 09 E. Berns
Filosofie, economie en macht
- 10 Anna Harańczyk
The Comparative Analysis of the Social Development of Cracow, Bratislava, and Leipzig, in the period 1960-1985
- 11 A.J. van Reeken
Over de relatie tussen de begrippen: offer, resultaat, efficiëntie, effectiviteit, produktiviteit, rendement en kwaliteit
- 12 A.J. van Reeken
Groeiende Index van Informatiesysteemontwikkelmethoden
- 13 A.J. van Reeken
A note on Types of Information Systems
- 14 A.J. van Reeken
Het probleem van de Componentenanalyse in ISAC
- 15 A. Kapteyn, P. Kooreman, R.J.M. Willems
Some methodological issues in the implementation of subjective poverty definitions
- 16 I. Woittiez
Preference Interdependence and Habit Formation in Family Labor Supply

- 17 A.J. van Reeken
A new concept for allocation of joint costs: Stepwise reduction of costs proportional to joint savings
- 18 A.J. van Reeken
Naar een andere aanpak in de systemering
- 19 J.G. de Boer, N.J.W. Greveling
Informatieplanning met behulp van referentie-informatiemodellen 1. Totstandkoming bedrijfsinformatiemodellen
- 20 J.G. de Boer, N.J.W. Greveling
Informatieplanning met behulp van referentie-informatiemodellen 2. Een methode voor informatieplanning
- 21 W. Reijnders
Direct Marketing: "Van tactiek naar strategie"
- 22 H. Gremmen
A four economy computer simulation game

IN 1987 REEDS VERSCHENEN

- 01 J.J.A. Moors
Analytical Properties of Bayesian Cox-Snell Bounds in Auditing
- 02 H.P.A. Mulders, A.J. van Reeken
DATAAL - een hulpmiddel voor onderhoud van gegevensverzamelingen
- 03 Drs. A.J. van Reeken
Informatisering en de beloning van arbeid
- 04 J. Kriens, P.C. van Batenburg
Bayesian Discovery Sampling: a simple model of Bayesian Inference in Auditing.
- 05 Prof.Dr. J.P.C. Kleijnen
Simulatie

Bibliotheek K. U. Brabant

17 000 01059239 3