Innovation management context strategies systems and processes pdf

Advances in computer-based information technology in recent years have led to a wide variety of systems that managers now use to make and implement decisions. To a large extent, these systems have been developed from scratch for specific purposes and differ significantly from standard electronic data processing systems. Too often, unfortunately, managers have little say in developing these sysems support decisions; at the same time, non-directors who develop them have a limited view of how they can be used. Despite these disadvantages, the author found that some of the 56 systems he studied are successful. And the difference between success and failure is the extent to which managers can use the system to increase their effectiveness within their organizations. Thus, the author suggests that this is the criterion designers and directors should attach together to exploiting the potential of current technologies. What can managers realistically expect from computers other than a pile of reports a foot deep dumped in their offices every other week? Everyone knows, for example, that computers are great at registering requirements. But what about all the promises and all the speculation over the last few decades about the role of the computer in management? While there have been advances in key information retrieval, processing, and demonstration technologies, my recent study of 56 computerized decision support systems confirms the common wisdom that very few management functions have actually been automated to date and all indications are that most may not be. Instead, my findings show what other researchers have reported: the applications are being developed and used to support the administrator responsible for making and implementing the decisions, rather than replacing it. In other words, people in a growing number of organisations are using what are often called decision support systems to improve their management effectiveness.1 Unfortunately, my research has also caused the fact that while more and more practical applications are being developed to use decision-makers, three major obstacles continue to hinder others who could benefit from them. First, computer managers and users in many organizations are familiar with only some of the types of systems that are now in use. As a result, different types of innovative systems have often been designed and cultivated by internal or external entrepreneurs, not by system users or their superiors. Secondly, and closely related my first finding, these entrepreneurs tend to focus on technical characteristics. Very often, this myopia means that they fail to predict the ways in which these systems can be used to increase the effectiveness of individuals in organisms. Finally, highly innovative systems — management itself should find it more useful — are at high risk of never being implemented, especially when the impetus for change comes from a source other than capabilities Quite simply, my purpose in this article is to discuss, without getting into the technology involved, the high potential of a variety of decision support systems, the challenges and risks they pose to administrators and implementers, as well as a wide range of strategies to address these challenges and risks. Types of decision support systems While there are many ways to categorize computer systems, a handy one is to compare them in terms of what the user does with them: Retrieves individual data items. Used as a mechanism for ad hoc analysis of data files. Receives predefined data aggregations in the form of standard reports. It assesses the consequences of the proposed decisions. As exhibit I states, EDP reference systems typically perform only the third operation in this list of functions, which I have organized along a dimension from data orientation to model orientation. Therefore, unlike the EDP user who receives standard reports on a periodic basis, the user of the decision support system usually starts each case of use of the system, either directly or through a personnel intermediary. Exhibit I Comparison of uses, purposes and characteristics of EDP systems against decision support systems Although decision-oriented reference systems are often developed by standard EDP systems, I will focus on seven different types, briefly describing an example of each type. It is important to note that external consultants developed the systems mentioned in my second, fifth and seventh examples, while those of the first, third and sixth were the creations of people who acted as internal entrepreneurs through staff roles; only the fourth system was developed in direct allocation by the user. This same pattern of launching innovative systems by people other than users was present in many of the 56 systems. 1. Retrieve single-one store floor information system. In order to help production precursors improve percentage performance in a newly developed 50-stage process for microcircing construction, the management of a company has installed an online, store floor information system. Operators submit daily track reports, which include performance, release date, identity of the person doing the work, and so on. Contractors then juggle this information to obtain productivity data by operation, operator, machine and batch. Thus, they are able to use the system in various ways. They can monitor workflow, identify performance problems, and settle daily queries, who worked on which lot when and which operators are ahead or behind schedule, or below the standards. Traitors have 13 standard commands with which they can retrieve the data stored in the system and display it in a end tube terminal. Commands allow them to customize reports to their needs. 2. Recovery and analysis—a portfolio analysis system. Before advising clients or making approved trading decisions, portfolio managers at a bank I studied use an on-line system to analyze Portfolios. Administrators can bypass time-consuming manual methods and obtain up-to-date and clearly organized portfolio information in either graphical or table format. Depending on the situation, an administrator can inspect both individual portfolios and groups of portfolios from different points of view— for example, sort them in different ways, take industry-specific or risk-based analyses, and so on. , but they were only accessible through tedious manual analysis. 3. Multiple databases plus analysis—sales information systems. Greater flexibility was also the reason why two consumer products companies and one construction company looked at developed sales information systems, which are quite similar. EDP's typical operations were too rigid to compile ad hoc sales analysis reports in a timely and cost-effective manner for businesses in the marketing and design sectors of companies. In any case, the information extracted from EDP systems is now kept separate in order to be useful and, in two cases, to be able to analyse it in conjunction with externally purchased proprietary databases and models. Basically, each system is a vehicle by which an employee or team tries to help decision-makers. How they work is incremental: identifying a problem; bring the current system and existing expertise to it; develop a solution in the form of an analysis or an additional system module; and integrate the results into an expanded version of the system. 4. Evaluation of decisions using an accounting model— a source budget and the implementation of funds. To speed up business decision-making and financial planning over a two-year horizon, an insurance company uses an online budget, source and capital implementation system. Inflows are forecasts of future business levels in various insurance and investment sectors, as well as assumptions about significant numbers, such as future money market interest rates. Production is a projected total cash flow per month. An investment committee uses the model to allocate capital to investment sectors and to minimise the amount of cash that remains dormant in banks. The Committee shall compare the projected cash flows on the basis of different allocation decisions; the decisions actually taken are those which generate sufficient projected cash flow and are acceptable to the various groups of the company. Actually system is an accounting definition of the company. There is no doubt about the accuracy of the relationships in the model, so the only way the projected results may be accidental is if estimates of business levels or money market interest rates are incorrect. 5. Evaluation of decisions using a simulation model— a marketing decision system. In order to provide a more rational basis for repetitive marketing decisions, a consumer products company uses a model that advertising, promotions and pricing at sales levels for a particular brand. The model was developed into a group that defines by reconciling an analysis of historical brand information with a person's subjective feelings about the impact on sales of various levels and types of advertising and other marketing actions. The model was validated by monitoring its accuracy in forecasting sales based on the competitive actions taken. Unlike the accounting model I just mentioned, this is a simulation model in which some of the most important relationships are estimates at best. For example, there is simply no rule by which sales can be predicted with certainty based on ad levels. In fact, this was the heart of the issue in the development of the model. Even though it has proven to be useful for forecasting, much of the value of the model lies in the company's improved understanding of the market environment. 6. Proposing decisions—optimising the use of raw materials. Another consumer products company, faced with short-term supply problems for many of its raw materials, has developed an optimization model to solve the mathematical puzzle of choice and balance between different product recipes. Inputs to the model include a range of different recipes for many products, short-term supply levels for raw materials and production requirements for finished products. Production is the choice of recipes that maximizes production using existing supplies. When the short-term supply situation shifts, the model can be revised and a new set of recipes selected. The system has had a significant impact on how administrators view allocation policy. Initially, they considered the distribution of rare raw materials in products by setting priorities among products. The model showed that it was more advantageous to start with production requirements and then allocate scarce resources by optimizing the mix of product recipes. 7. Decision-making— a system of insurance renewal rates. As a result of the overhaul of the group insurance information system, an insurance company has developed a system to eliminate some of the office load associated with underwriting renewal and to help ensure that interest rate calculations are consistent and accurate. Instead of calculating renewal rates manually, insurers fill out coded entry sheets for the system, which calculates a renewal rate based on a series of standard statistics and actuarial assumptions. Since these assumptions may or may not apply to a policy, insurers review the documents accompanying the policies and decide whether the standard calculations apply. If not, the encoding sheet shall be amended in an appropriate manner and re-signed. In fact, the system makes the decision in completely standardised situations, while the insurer decides whether the situation is formal and, if not, what adjustments are required. As a result, insurers can focus on their jobs and not the relevant office work. Range of capabilities These seven systems represent a wide range of approaches to decision support. The first helps the eminent production by simply providing rapid access to historical information, such as who worked on which batch, and when the work was done. But the products have to decide what to do once they have the information. At the other end, the system that supports insurers essentially makes the decision in some cases. Between the two extremes, analysis systems and model-oriented systems help people organise information and also facilitate and formalise the evaluation of proposed decisions. Although administrators in most large companies have used budget or design systems similar to the source-and-application of the capital model I mentioned, the range of possibilities for other types of decision support systems is surprisingly wide. Obviously, some of these systems do not have much use in many settings. However, their variety suggests that most companies should have a number of real opportunities to implement the concept of computer-based support for decision-making. Managers' Incentives What do decision support systems do that really helps their users? What's their real impact? In my research, the answers to these questions proved elusive in many cases as users appreciated the systems for reasons that were completely different from the original ideas of what the systems were going to achieve. In fact, there is a wide range of purposes for these systems. While many decision support systems share the objectives of standard EDP systems, they go further and address other management concerns, such as improving interpersonal communication, facilitating problem solving, promoting individual learning and increasing organisational control. These systems can influence interpersonal communication in two ways: by providing individuals with tools of persuasion and by providing organizations with a vocabulary and discipline that facilitates negotiations beyond the boundaries of the sub-unit. Persuasion Tools Standard texts for system analysis completely overlook the personal use of decision support systems as persuasion tools. But consider the following offensive (persuading someone else to do something) uses to which various companies have set these systems: The manager of a chemical plant was trying to achieve the production targets (quantities per product) set by a marketing team. Unfortunately, the group sets targets without taking much account of the shortages of raw materials in the of which the factory operated. The factory used a model to calculate production mixtures. At one point, it happened to the plant manager that he could use this model to investigate whether marketing paced the goals that led to poor plant utilization and made him look ineffective. As he ran the model under a number of different production mix targets, it became clear that this happened, and he used results to convince marketing to change the factory's production mix. A data recovery and manipulation system first received widespread exposure to a transportation company when some of the company's top executives used it to develop a good quantitative logic for a proposed merger. With the system, it was possible to investigate and manipulate a large database of information about the industry. Although the merger was not approved, management felt the system helped fight a good battle. The management of a shipping company found that a system it used to consolidate and improve strategic investment plans also helped it negotiate with banks. Banks and other sources of finance appear to be uniformly impressed by the careful computer analysis on which management based its funding requests. The credibility advantage that emerged was small, but, in the administration's view, remarkable. Now that we've seen the depictions of aggressive tools of persuasion, let's turn to examples of defensive (convincing someone that the user has done a good job) using these systems: When asked if he ever made direct use of a case monitoring system, the head of an arbitration panel at a government regulatory agency said he remembered only one case. That was when he spent an hour of lunch creating a report to make his team's recent performance seem as favorable as possible, despite some unfortunate delays and problems that made the typical report look bad. The new chairman of a large conglomerate used a one-year budgeting model to tell the budget choices that existed, as well as to help him reject what people in different sectors claimed about their own budgetary needs. Scheduling the classes of a training school for a company's service staff had found his job frustrating because it was always difficult to justify the budget for clear reasons. With a model that created optimal training programs, the schedule could protect itself very easily by saying: Using these assumptions about wear and tear, acceptable peak hour shortages, and other considerations, this is the best budget. If (the budget cutter) would like to change these assumptions, I would be happy to create a new budget. What level of lack do you recommend? So the system not only helped the timing make decisions, but also helped him defend them. Many people suspected that a new product company in a consumer business might not be worth it, but no one knew exactly why. a risk analysis was carried out with a model, the reason became clear: the company had a very significant risk of downward liability. In addition to sealing the decision, the analysis provided an understandable response to the people who had proposed the project. A cynic could argue that people in these situations exploited or abused systems. A more practical conclusion is that these systems merely contribute to improving the organisations by helping them communicate with other people. What I want to say is that much of the benefit of many of the decision support systems in my sample was of this kind. Communication aids Decision support systems also help administrators negotiate in all organizational units, standardizing process mechanisms and providing a common conceptual basis for decision-making. During my research, managers often commented that consistent definitions and forms are important aids in communication, especially among people in different organizational units, such as departments or departments. In some cases, the development of these definitions and forms was a long and sometimes arduous task that was gradually achieved over several years, but which was also considered one of the main contributions of the systems. For example, one of the purposes of some of the standard systems in my sample was to assess in advance the overall outcome of decisions that different people looked at separately, filtering these decisions through a single model. In these cases, the system became an indirect arbiter between the different objectives of the different departments. Instead of arguing from their own divergent views, marketing, production and financial people could use the model to demonstrate the impact of a group's proposals on another group's actions and overall outcome. As a result, the issues were clarified and negotiations accelerated. The production products I mentioned earlier have experienced the same kind of facilitation. It helped them in job-planning discussions and problem surveys by providing direct access to objective information about who did what, when, and how well in any batch of production in the store. The administration usually decided to proceed on the basis that the proposed system seemed to make sense and would likely have a beneficial impact on the way people interact and/or make decisions. Monetary savings are obviously a very important and worthwhile logic for the development of IT systems, but it should be clear at this point that the CASE-type assumption that systems should always be justified by these conditions is not sufficient in the area of decision support systems. Equally obvious, there is a clear risk in the development of a system of simply only because someone thinks it makes sense, especially if that someone is not the direct user of the system. In fact, the systems I mentioned as my first, second and fifth examples started this way and faced resistance until they were repositioned as something that users would like to become more effective. Again, the general problem here is a common audience for technical people to focus on the technical beauty of a system or an idea and assume that non-technical people will somehow see the light and will be able to figure out how to use the system to solve business problems. This kind of hypereodoxia was present in the history of almost every unsuccessful system in the sample. The message is clear: try to take advantage of the creativity of technical experts, but be sure that it is channeled into real problems. The challenge, of course, is how to achieve both of these goals. There are several ways I will discuss them now. Development Standards Despite the common wisdom that user needs should be taken into account in system development and that users should be actively involved in their application, users did not start 31 of the 56 systems I studied and were not actively involved in the development of 38 out of 56. The results, presented in Exhibit II, are not surprising. Intended users did not initiate or play an active role in the implementation of 11 of the 15 systems that suffered significant implementation problems. In contrast, there were relatively few such problems in 27 of the 31 systems in which users participated in the launch and/or played an active role in the application. Exhibit II systems resisted by users But it would be wrong to inferred from these findings that systems should be completely avoided if intended users do not initiate them or play an active role in their implementation. First, 14 of the 25 systems I studied in which this pattern was was ultimately successful. Most importantly, many of the truly innovative systems in my sample, including 5 of the 7 I described earlier, presented this standard. On the other hand, many of the systems started by users do more than mechanize existing practices. While such mechanization can be very beneficial, and while I'm certainly not suggesting that great innovations have to come from external sources, the real challenge is to be able to use knowledge regardless of their source. One way to do this is to design an application strategy to encourage user participation and participation throughout system development, regardless of who initiated the concept. Here are examples of successful strategies. Impose gracefully: Marketing and production managers at a decentralized company do not enjoy the extra work (format changes and data submission requirements) required for an annual budgeting system, which top management was installing. At first, they were particularly impatient because they thought the system wouldn't really help them. Thus, at each stage the designers a point of subsystem development to provide these middle managers with sales and material usage information that was never available. Did this exchange work well? instead of seeing the system as a total imposition, the administrator saw it as an opportunity to take part in something that would be to them. Run a Dog and Pony Show: Central design staff at two companies that have designed systems for budgeting and financial analysis. In one company, the system never worked despite long training demonstrations for departmental staff and other potential users. These individuals seemed enthusiastic about the capabilities of the system, but never really used it unless corporate programming people did all the work for themselves. On the contrary, the training program for the system in the other company encouraged direct and active participation. To attend the workshops, people had to bring their own problems of economic analysis. They learned to use the system by working on these problems. When the workshops were finished, many users were enthusiastic: not only did they know how to use the system, but they had also proven to themselves that it could help them. Use a prototype: Two constantly present risks in developing a system create a big, expensive one that solves the wrong problem or creates one that some people in the organization can't live with. Either it can happen, not only when the system is designed without consulting the user and the affected parties, but also when no one has enough experience with this type of system under consideration to clearly reflect its strengths and weaknesses before it is built. The implementers of certain systems in my sample avoided these pitfalls by building small prototypes, which gave users something specific to react to. As a result, the large-scale version could be developed in a realistic sense of both what was needed and what would fly in the organization. A similar approach, also successful, was simply to build systems in small pieces that could be easily used, changed or discarded. Hook the user with responsibility: Each new module or application developed as an outgrowth of one of the three sales information systems I mentioned previously goes through three stages. The first stage consists of general, unbound discussions about any current problem areas with which these are user groups. After research by administrative science staff, the second stage is a brief official problem statement written in conjunction with the user group. In addition to describing the problem, this statement examines the methodology and resources that will be required to respond to it. The third stage is a formal request for the approval of out-of-pocket expenses. Sell the system: At one of the companies I studied, a marketing analysis team used a direct selling process to convince people of the advantages of a system sales. The pitch was very simple: they compared manual monthly forecasts for a year with system forecasts. The system's predictions proved more accurate in ten months out of 12, with fewer errors overall than manual errors. The system has been approved. In another company, management had installed a real-time system to monitor the largely automatic automatic an inexpensive consumer item in order to minimize material loss due to the resuspending maladjustments in machine settings. During the initial installation, the implementation team discovered suspected, but previously unfounded, deception by piecework employees; more pieces left many machines than were entering them. Subtle hints fell that the screen needed to be checked because it recorded weak results. The workers were sold in the new system: they knew very well that it worked. Fundamental changes Despite extensive experience with EDP, many organizations have not used more than one or two of the seven types of decision support systems I have shown here. One reason for this is that justifying such systems can be difficult: quantifying the impact of replacing ten employees with one computer is one thing, while quantifying the impact of improved individual efficiency of line staff is a completely different thing. Another reason is that the application can be difficult: many of the ideas come from people other than users. However, the development of a decision support system makes sense when it becomes clear that a fundamental change in the way decisions are taken and implemented may be needed. Often, the process of determining the system is just as valuable as the system produced. My last point is that the very concept of decision support systems can help administrators understand the role of computers in their organizations. As the name suggests, data processing systems systematize and accelerate business mechanisms by automatically processing data masses. On the other hand, extensions of these decision-driven systems help people to make and communicate decisions about administrative and/or competitive tactics and strategies. The decision support systems I have discussed go one step further. Instead of starting out as extensions of existing data processing systems, many decision support systems are built from scratch for the sole purpose of improving or speeding up a decision-making process. The underlying philosophy is that using computers to help people make and communicate decisions is just as legitimate and worth it using computers to process data masses. There are indications that this view has caught on to some extent and is becoming more widely accepted. The consequence is not that all organizations should get in vogue, but rather that administrators should be aware of the opportunities and challenges in this area and should try to assess whether their organizations will to move in this direction. 1. Steven Alter, A Study of Computer Decision-Making Assistance in Organizations, Ph.D. Thesis, Sloan School of Management, MIT, 1978. A version of this article appeared in the November 1976 issue of the Harvard Business Review. Review. Review.