Association for Information Systems
AIS Electronic Library (AISeL)

ICIS 1986 Proceedings

International Conference on Information Systems (ICIS)

1986

IMPLEMENTABILITY OF IN-HOUSE DEVELOPED VS. APPLICATION PACKAGE BASED INFORMATION SYSTEMS

Juhani Iivari
University of Oulu

IMPLEMENTABILITY OF IN-HOUSE DEVELOPED VS. APPLICATION PACKAGE BASED INFORMATION SYSTEMS

Juhani Iivari
Institute of Information Processing Science
University of Oulu

ABSTRACT

Even though in-house development and the use of application packages form two major strategies for developing information systems, there is practically no research into the comparative implementability of information systems developed according to these strategies. The paper suggests that the level of originality of information systems can be used as a theoretical concept underlying the dichotomy of in-house development vs. application packages, and puts forward a set of hypotheses regarding the influence of the level of originality on the implementability of information systems. The hypotheses form the theoretical basis for a comparison of the two types of information system development in terms of implementability, and clearly support Lynch's finding (Lynch, 1984) that the implementation of application packages is more difficult than one might initially expect.

INTRODUCTION

There has been a growing trend towards the use of generalized application packages in IS development, largely due to the fact that in-house development has turned out to be costly and risky in terms of development costs and schedules and the quality of the resulting systems. Even though application generators and fourth generation languages can be expected to alter the balance between in-house development and the use of application packages, it is obvious that the above trend will continue in the near future.

Irrespective of this, there is, to the author's knowledge, practically no comparative research available concerning the implementability of in-house developed vs. application package based information systems. The monograph of Lucas (1981), for instance, does not explicitly recognize this dimension as a potential factor explaining the implementability of information systems. The question has been clearly recognized by Lynch, however, in his article on hidden costs and new challenges of implementing package application software (Lynch, 1984), his study being based on the experience of implementing basic financial software packages in two client organizations. He concludes that even though "...implementation of an application package may be cheaper, easier and faster than custom development of the same application; however it will probably still be slower, more difficult and more expensive than one might initially expect" (p. 234).

More indirectly, there are a number of studies, case studies in particular, dealing with the implementation of either in-house developed or application package based information systems, studies which may highlight features potentially specific to these two types of information system. Kole (1983), for example, suggests a non-developmental strategy for implementing stan-

[1]This work was supported by the Academy of Finland

dardized application package based information systems and illustrates the potential value of the proposed strategy in the case of three companies. Company A followed the proposed non-developmental strategy, Company B did so partially, and Company C decided to develop its own software without any well-defined implementation strategy. The results indicated that Company A achieved more rapid implementation and earlier use of its systems than B or C, and that the progress of company C was the slowest.

We have the impression that there is a common-sense belief that application package based information systems are easier to implement than in-house developed systems. It should be observed, however, that the results are not very conclusive in this respect. Lynch's study (1984) itself does not include any comparative data on this score, and Kole (1983) regards his own results as examples rather than scientific tests. Furthermore, Kole's main interest lies in the non-developmental implementation strategy, and since Company C did not apply any well-defined implementation strategy it is difficult to conclude whether the implementation problems were because of the lack of an implementation strategy or due to the policy of in-house development.

There are also certain contradictions, at least between the lines, in the concrete recommendations concerning the implementation of application package based information systems. Kole emphasizes prompt transition to productive use of the standardized systems with minimal tailoring, whereas Lynch seems to be quite skeptical about quick implementation due to the steep learning curve required. Furthermore, he emphasizes that it was not possible to implement all aspects of the packages in the required time and is worried by how much was lost in overall organizational efficiency due to the tight implementation schedule.

Keen, *et al.*, (1982), in their study on implementing a common system in an international bank, also came to quite a different conclusion from Kole. They distinguish a *crash strategy* in which "the priority was to get the technical system up and running as quickly as possible and then to deal with training and organizational problems," and the 'filter strategy' which "adjusts the pace of development to the organization's ability to assimilate the change," and also remark in the context of the latter strategy that it "delays the installation of the system but significantly eases its institutionalization." They also emphasize that in the case of common systems there are often unrealistic expectations about the necessary degree of local adaptation.

There are, however, situational factors which may explain the differences in the conclusions reached in these studies. First, Keen's study concerns a common system developed centrally for worldwide implementation within one banking organization. It may be that this system cannot be regarded as a typical application package, and due to the international scope of the implementation cultural, legislative differences, etc., are obviously more important than in implementation within one country, as in Kole's study. Furthermore, Kole's study is explicitly limited to small organizations, whereas in that of Keen the local units obviously varied considerably in size. In fact, Keen, *et al.*, remark that the "parachuting strategy" associated with the crash pace had been successful in the case of four countries with relatively small local units and with little or no earlier use of computers.

Even these studies, and particularly the cases of obvious implementation failures with application package based information systems, indicate that the implementation of the application packages is by no means unproblematical. Lynch (1984) reports problems such as confusion as to which functions the application packages can support and how the packages carry out the functions they support, limitations on changing the packages, problems in the technical quality of the packages, and problems with vendor support. It should be observed, however, that his cases were based on "off the shelf" implementation of the packages. We see that this idea is fallacious (cf. Part 2), and consequently we suspect that the problems encountered were partly because of that misleading concept rather than inherent in the characteristics of an application package.

Markus' excellent case of implementing a financial accounting package is a good example of the problems in a more severe sense leading to an implementation process which obviously cannot be considered successful (Markus, 1979, 1983). Dutton and Kraemer (1983), reporting on the implementation of packages for fiscal impact analysis in local governments, also suggest that even though the computerized models were appropriate "technical" solutions to the problems of fiscal impact analysis in local government,

few had adopted such models and even fewer had incorporated them into their ongoing planning processes. Finally, Gross and Ginzberg (1984) give a ranked list of 38 problems viewed as barriers to the adoption (purchase and implementation) of application software packages. The ranking was based on interviews with 55 respondents. The paper of Gross and Ginzberg does not deal specifically with the implementation of packages in the sense of their institutionalization, however, and it is significant that practically none of the 38 problems is specifically related to this aspect.

The purpose of this paper is to open the discussion in more theoretical terms on the important theme of the implementability of in-house developed vs. application package based information systems and to put forward a set of hypotheses. A descriptive model for the implementability of information systems based on the author's earlier work (Iivari, 1986b) is first briefly introduced. Concrete hypotheses concerning the comparative implementability of these two types of information system are then put forward, and summarized.

DESCRIPTIVE MODEL FOR THE IMPLEMENTABILITY OF INFORMATION SYSTEMS

In accordance with the author's earlier article (Iivari, 1985) IS implementation is interpreted in this paper in the sense of organizational implementation or institutionalization, and as a phase succeeding the design phase. IS implementation success is defined as a decreasing function of the costs of producing a state in which a model component for the information system is transformed into a compatible real system which is institutionalized in the host organization. The term 'implementability' is used to describe the ease or difficulty of implementation.

The descriptive model for the implementability of information systems to be employed in this paper is based on an innovation research perspective on information systems (Iivari, 1986b). Information systems are interpreted as special types of innovations in the adopting host organizations. The term 'innovation' is used here in the general sense of "something new to the adopting organization" (Pelz and Munson, 1982) or "any idea, practice, or material artifact perceived to be new by the relevant unit of adoption" (Zaltman, Duncan and Holbek, 1973). An innovation is not stipulated to be something new to the field, but its newness is the perception of the social unit adopting it (Zaltman, *et al,* 1973). This generality means that many, if not most, IS development processes can be regarded as innovation processes, and consequently results concerning the implementation of innovations may be of value within IS implementation research, too.

The descriptive model developed and explained by Iivari (1986b) recognizes four characteristics of information systems, complexity, radicalness, originality and divisibility. These concepts are explained in more detail below, but before that we put forward in the next section an hierarchial metamodel for an information system as a necessary conceptual prerequisite for defining the characteristics of information systems mentioned above. We then concentrate on the dimension 'originality' which can be regarded as a theoretical construct covering the dimension of in-house development vs. application packages. Finally, a "causal" model for the impact of the four characteristics of information systems upon the implementability of information systems is briefly introduced.

A Hierarchical Metamodel for an Information System

There is increasing agreement about the usefulness of distinguishing three major levels of modeling in IS development (cf. Falkenberg, *et. al.,* 1983; Iivari, 1986a).

- *Level A* defines the organizational context of the information system to be developed

- *Level B* defines the conceptual/infological specification of the infor-

mation system

- *Level C* defines the technical/datalogical structure of the information system

The terms infological and datalogical are used in the sense of Langefors (1974), and the levels correspond quite closely to the systeological, infological and datalogical perspectives as defined by Welke (1977). In the PIOCO model (Iivari, 1983) these levels correspond to the pragmatic (P), input-output (I/O) and constructive-operative (C/O) metamodels for an information system. The P model is defined as a restricted planned change in the host organization. The change may concern both systemized (formal) and unsystemized (informal) information systems, as well as other structural factors such as staffing, organizational arrangements, working procedures, technological arrangements, etc.

The I/O model defines the primary information and its processing rules at the infological level and the user-information system interaction, and consists of four major components: the object system model defining the conceptual model for the UoD underlying the information system, the information model specifying the information types to be included in the system, the information process model defining the derivation rules, and the interaction model specifying the interaction between the information system and its input and output users.

The C/O model determines the primary data and their processing rules at the datalogical level, the secondary data and their processing rules related to the control and supporting activities included in the information system, and the technical organization of the system. It similarly consists of four major components: the data model defining the files and databases and their technical structure, the data process model defining the programs and their structure, a control and supporting action model specifying the security controls to be included in the system, for example, and the (technical) organization model specifying the human, software and hardware resources to be used in the technical implementation of the information system.

In the following section we apply these characterizations of the levels A, B and C to the task of defining the complexity, radicalness, originality and divisibility of information systems.

Four Characteristics of an Information System

The textbook on innovations by Zaltman, *et al.*, (1973) includes an extensive discussion of the characteristics of innovations "believed to influence their acceptance by organizations." We suggested earlier that these characteristics can be organized into six categories: characteristics of the innovation (e.g. complexity), characteristics of the adopting organization (e.g. degree of commitment), characteristics related to the innovation/adopting organization fit (e.g. compatibility), characteristics related to the actual impact and costs of innovations (e.g. efficiency), characteristics related to the assessability of innovations (e.g. demonstrability) and characteristics related to the expected impact and costs of innovations (e.g. risk and uncertainty). In the descriptive causal model developed by the author (Iivari, 1986b), we concentrated on the immediate characteristics of innovations, including their complexity, radicalness, originality and divisibility, since these characteristics are the most controllable and basic from the viewpoint of IS implementation management, and at the same time quite general, providing good prospects for transferring the results achieved within innovation research into our area of IS implementation. The remaining categories are still taken into account, however, even though not as central constituents of the model as are these four factors.

Table 1 characterizes the complexity, radicalness and originality of information systems at the three levels of modeling. The divisibility dimension is omitted, since it does not affect IS implementability "directly," but only through reduced complexity, radicalness and originality.

Complexity is interpreted in Table 1 in terms of systems theory to refer to the number and variety of elements and their interconnections in a system. Applying this general concept, the three complexity concepts are outlined in terms of the corresponding (systems) models for an information system. The purpose of using these three concepts is merely to express their main ideas without suggesting any exact measures, and their explanations are therefore allowed to include references to more detailed complexity measures, e.g. the complexity of the derivation rules (see level B). In the case of these detailed complexity measures we refer the reader to the existing literature on software complexity (see Curtis, 1981; Belady, 1981; for summaries).

Table 1: Levels of Modeling vs. Characteristics of Information Systems

	Complexity	Radicalness	Originality
Level A —organizational level	How complex is the organizational change? Number of organizational levels, units and members directly affected by the change	How different are the new organizational structure and practice perceived to be?	Are the new organizational structure and practice borrowed, adapted or originated? - imitation of existing systems as models and ideas
Level B —infological/ conceptual level	How complex is the infological/ conceptual model? Number of entity and association types, event and transaction types, reports and query types, and number and complexity of derivation rules and dialogues?	How different is the infological/ conceptual model perceived to be?	Is the infological/ conceptual model borrowed, adapted or originated? - imitation of existing systems as models and ideas - re-use of software
Level C —datalogical/ technical level	How complex is the datalogical/technical model? Number and complexity of programs, databases (files), access control, and the extent and heterogeneity of the software and hardware environment (operating systems, DBMSs, computer systems, communication networks, etc.)	How different is the datalogical/technical model perceived to be?	Is the datalogical/ technical model borrowed, adapted or originated? - imitation of existing systems as models and ideas - re-use of software

Even though there are obviously no naturally correct measures for the complexity of an information system at the three levels of abstraction, the complexity can still be interpreted in quite formal terms. In contrast, radicalness is largely a perceptual measure which describes the perceived amount of change in the system at each level of abstraction (cf. Zaltman, *et al.*, 1973). At the organizational level the information system may imply or be related to a change in working conditions which is regarded as a radical change (by the workers, for instance), while

at the infological/conceptual level the change from an off-line mode of interaction to online interactive use may be perceived as a radical change by the users, and at the datalogical/-technical level the change from a centralized to a distributed database may also be considered a radical change.

The third characteristic, originality, is based on the proposal of Pelz and Munson, who define it as a more-or-less continuous scale which can be described by three values. When an organization develops its own first-time solution without any precedent for doing so, the innovation takes place at the level of origination. When there are precedents which are modified to fit the situation of the applying organization, the innovation takes place at the level of adaptation. And finally, when the organization copies earlier well developed solutions, there is borrowing (Pelz and Munson, 1982). It should be observed, however, that some parts in a given innovation may be borrowed, some adopted and some originated.

Originality forms an important characteristic in the case of information systems in particular, since pre-existing solutions may be used in two senses. Firstly, existing solutions may provide models and ideas which may be imitated at all three levels of abstraction either directly (through borrowing) or tailoring to the specific needs of the adopting organization (adaptation). Secondly, originality also covers the potential re-use of existing systems or software in a concrete sense. Consequently, originality also includes the problem of whether to use existing application packages or whether to develop the system in-house (Davis and Olson, 1984). We shall return to this subject in the following section.

In-house Development, Application Packages and the Originality of Information Systems

We have used the phrases 'in-house developed' and 'application package based' as if there were a dichotomous situation. In practice, a continuum tends to exist, since application packages usually require some tailoring, which was found by Gross and Ginzberg to be the most critical problem area in application software adoption (Gross and Ginzberg, 1984). In the technical sense one can distinguish different forms of tailoring on a continuum describing its difficulty. In the simplest case the tailoring can be done by fixing a set of control parameters; in a more demanding case some additional software is required for the application package, which otherwise can be installed without any changes; and in the most problematic cases the application package and its data definitions and programs must all be changed (Kole, 1983; Keen, *et al.*, 1982). We are not interested in these technical options here, but we do wish to point out that the tailoring can concern purely technical aspects of the system taking place at level C (aiming at converting the package into one appropriate for a specific computer environment or tuning the performance of the package) or at level B (implying changes in reports, derivation rules, interaction techniques). Tailoring at level B naturally implies a certain tailoring at the level C.

The hierarchical metamodel for an information system introduced above also suggests that information systems always have a specific organizational role and context in the sense of level A, irrespective of whether this is consciously designed or is implicitly taken for granted by meshing the system into existing organizational structures (Robey, 1983). There are good reasons for believing that this role should be consciously designed and considered, even though such consideration might lead to preservation of the existing structures (Bostrom and Heinen, 1977; Robey, 1983). One of these reasons is the fact that information systems are not solely or primarily technical systems (level C), but are also organizational/social (level A) and closely related linguistic/communication systems (level B). This means that information systems should not be regarded as commodities which can be bought or acquired from outside markets, like application packages, but rather as organic parts of the host organization requiring conscious design at the organizational level (level A), even in the case of application package based information systems. Organizational redesigning can also be used to reduce the need for package tailoring, as Gross and Ginzberg suggest (1984).

We have thus far put forward two reasons for insisting that the everyday way of opposing in-house developed systems to application packages is misleading due to the implicit dichotomy involved which suggests that they must be mutually exclusive alternatives. The first reason, the need for tailoring application packages, is generally recognized, but the second, the

need for conscious design of the organizational role and context of the information system, is often neglected, obviously because of the narrow image of the concept of information system, covering only levels B and C.

Nevertheless, we have used these misleading everyday expressions throughout our paper and even in its title. Our reason is that they are very common and established phrases in practice, and may be quite useful in everyday conversation. But in a more scientific discussion they should obviously be replaced by a more theoretical concept. We suggest that this could be phrased in terms of the level of originality, interpreted as a continuum, rather than as a discrete three point scale as above (Pelz and Munson, 1982).

The originality dimension characterized in Table 1 is a more general concept than the continuum underlying the dichotomy of in-house development vs. application packages, but it recognizes the two forms of utilization of application packages in IS development as sources of models and ideas and as reusable software--forms of utilization which can take place quite independently. An application package can be used as a source of ideas and models without using it as a software product in the realization of the selected model, conversely IS models may be genuinely originated, and if the originated model turns out to be sufficiently similar to an application package, the latter can be used in its implementation. We shall assume a combined use in the following, however, in order to emphasize the difference between application package based and in-house developed information systems.

In principle, in-house developed information systems are not necessarily very original. The models may be borrowed and implementation may be based on the effective utilization of various types of reusable software. It is obvious, however, that the level of originality of in-house developed information systems must be statistically higher than that of application package based information systems. This can partly be explained by economic factors, in that in-house development is used when a unique system is required to fit the specific needs of an organization, whereas if a more standardized system is acceptable, there are considerable economic reasons for using application packages.

THE CAUSAL MODEL

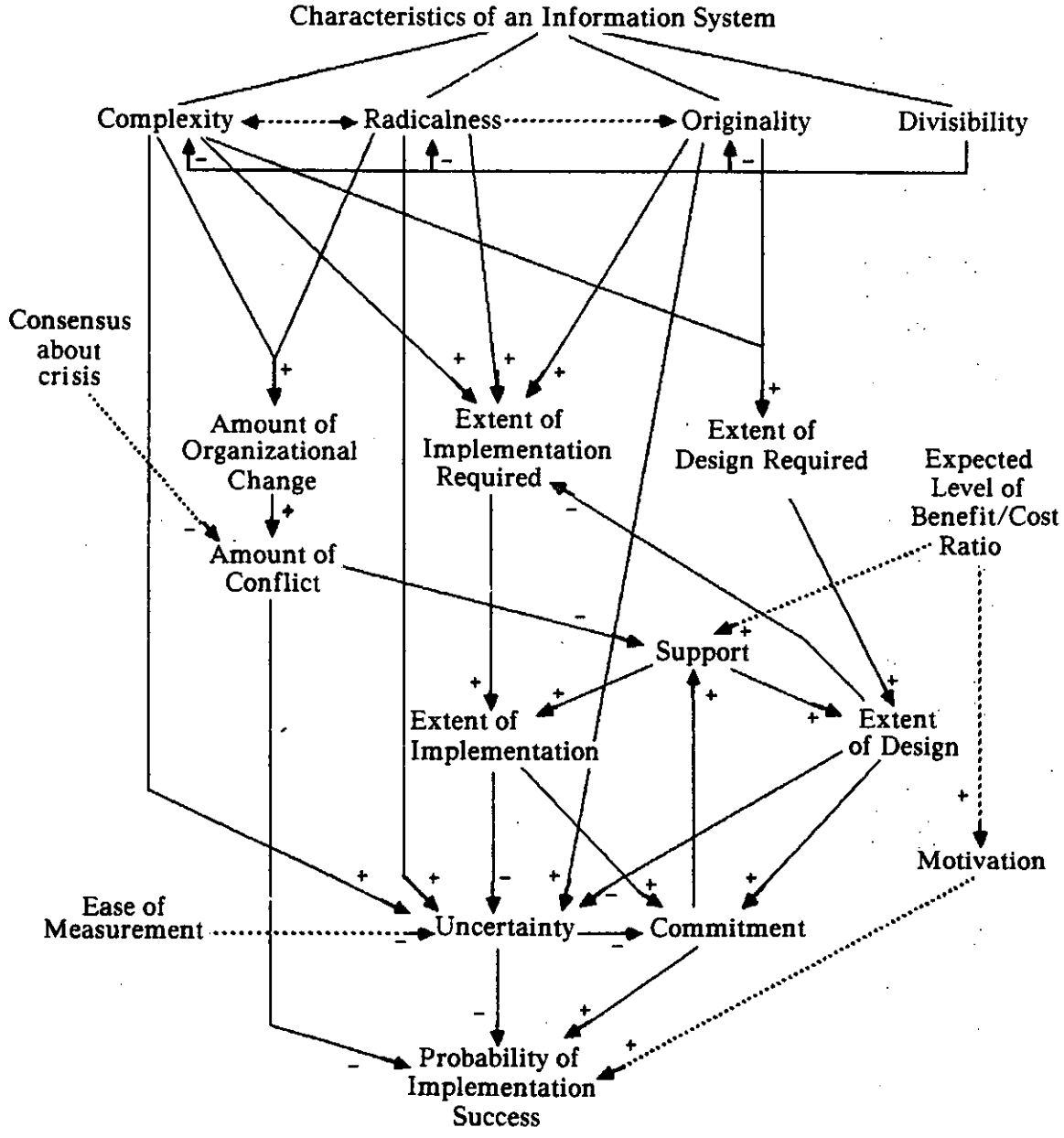
The causal model expressing a set of hypotheses concerning the implementability of information systems, as developed by the author (Iivari, 1986b), is based on two major references, Hage's theory of radical innovations (Hage, 1980) and on Peltz and Munson's ideas concerning the originality of innovations (Pelz and Munson, 1982). The model is depicted in Figure 1.

The dotted arrow between complexity and radicalness reminds us of the close relationship between these two concepts. And the arrow from radicalness to originality denotes that the originality level of highly radical information systems is by necessity high. The negative arrows from divisibility to complexity, radicalness and originality reflect the fact that divisibility makes it possible to reduce these three qualities by decomposing the information system.

We also depict certain external factors such as consensus about crises, ease of measurement, and expected level of benefit/cost ratio (Hage, 1980), which are not assumed to be dependent on the four characteristics identified at the top of the figure, but are taken into account in order to clarify our reasoning.

The reasoning behind the influence of complexity and radicalness upon the implementability of information systems is based on Hage's theory. His interest is in production-oriented radical innovations, which imply something new for the field and are quite rare events (Hage, 1980). Furthermore, he assumes radical innovations to be complex. This "extremest" orientation naturally reduces the degree to which his results can be generalized to our area of information systems. We have used his theory as a source for our hypotheses, however, taking the position that this is a matter of degree, and that Hage's theory is based on an extreme value on the continuous scale of radicalness.

When transferred to the area of IS development, Hage's theory includes the coexistence of political and rational aspects of IS design and implementation (Franz and Robey, 1984; Kling and Iacono, 1984), both aspects being operative simultaneously. The political side is described on the left and the rational side on the right. It also includes certain features of the cultural-systems perspective (Markus, 1979) in its emphasis on the experimentation required during



3-Stage Learning Model of Cognitive Skills

Stage 1: Cognitive

↓

Stage 2: Associative

↓

Stage 3: Autonomous

Figure 1. Causal Model

implementation to change people's perceptions of the proper way of doing things, whether because of psychological factors, prior training or organizational experience.

The main change compared with Hage's theory is the incorporation of the level of originality in the framework. Generalizing from the discussion of Pelz and Munson (1982), we assume that for original information systems of substantial complexity, requiring more extensive design, it is important that there should be sufficient organizational support to ensure the survival of the system to the implementation phase. The design process also increases commitment to the system within the organization. From that point onwards the reasoning largely follows Hage.

There is a need for more extensive experimentation during implementation in the case of complex, radical and original information systems than for less complex, radical or original ones. In Figure 1 the design is assumed to reduce the uncertainty related to the IS development concerning the system, its consequences, benefits and costs, etc. The increased knowledge obtained can be expected to be beneficial from the implementation viewpoint as well as committally (Brunson, 1982). An existing consensus and organizational support will facilitate the actual experimentation, which usually increases the commitment due to the greater costs sunk in the IS development project.

The Impact of Originality upon the Implementability of Information Systems

Figure 1 suggests that the level of originality influences implementability through its cognitive impact (uncertainty) and committal impact. Figure 1 also emphasizes that both IS design and implementation require organizational support. If this support is inadequate, the IS development may be aborted during the design or implementation phase. From the viewpoint of IS implementation the former implies a filter effect suggesting that the most risky IS development projects are aborted in their existing form, i.e. totally terminated or modified to reduce the risk before implementation, and the latter a risk that the information system to be implemented may be abandoned during implementation, possibly without a fair trial.

Cognitive and committal influence and the extent of implementation

This impact is expressed in the form of five hypotheses:

Hypothesis 1. The level of originality has a direct positive influence on the uncertainty related to the IS development

This hypothesis is based on the obvious assumption that the assessability of an information system of low originality is greater than for one of higher originality. Earlier experience regarding the costs and impact of the system is usually available, and particularly in the case of application packages there are good opportunities for experimenting with the system during the design phase.

Hypothesis 2. The level of originality has a direct negative influence on commitment to the information system.

This hypothesis is a direct corollary of Hypothesis 1 and our assumption that uncertainty is negatively related to commitment. Hypotheses 1 and 2 together point to one of the major reasons for believing that information systems based on application packages are easier to institutionalize than in-house developed systems. There are also more indirect relationships, however, which will be reviewed next.

Hypothesis 3. The level of originality has an indirect negative influence on the uncertainty related to the IS development

This hypothesis is based on the assumption that increased originality means more extensive design, which can be assumed to reduce uncertainty.

Hypothesis 4. The level of originality has an indirect positive influence on the commitment to the information system.

Greater originality increases the need for design, and this design creates commitment both in the financial sense (investments) and in the psychological sense. This is particularly true in the case

of those information systems which are able to survive the design phase (cf. Hypothesis 6).

In the case of application packages it should be observed that the design phase may include financial obligations (e.g. the purchase of the package) which means a considerable financial commitment to the system. Assuming that choice of the level of originality is based on a financial analysis, it seems safe to assume that these financial obligations do not exceed the costs of design required in in-house development.

Hypotheses 3 and 4 potentially form one of the major explanations for the difficulties encountered in the implementation of application package intensive information systems. Due to the short period of design there is a lot of uncertainty related to the system, and its organizational implications in particular, regarding the relative costs and benefits, etc. There is not enough time for people to become accustomed and committed to the change implied by the IS development. Both these explanations can be identified in the paper of Lynch (1984), who mentions as problems the lack of system analyst, programmer and user understanding of the application package upon the commencement of implementation, and the difficulty of establishing a relationship between the users and the technical support staff in the short space of time required for application package implementation. The work of Keen, *et al.*, (1982) partially supports these explanations in that they note in the case of common systems that a crash strategy, even though leading to rapid installation of a system, makes its institutionalization more difficult.

Hypotheses 1-4 describe the influence of the level of originality upon the implementability of information systems in the sense that the implementation may turn out to be very difficult and problematic, requiring time and energy. This can be summarized in Hypothesis 5.

Hypothesis 5. The level of originality has a mixed positive and negative influence on the extent of implementation required

This hypothesis is described in Figure 1 as a direct positive relationship between the level of originality and the extent of implementation required; and indirectly as a negative arrow from the extent of design to the extent of implementation required.

Hypothesis 5 proposes that, irrespective of the problems encountered, the implementation of a given system may continue even though excessive time and resources may be required; that the implementation cannot be regarded as successful. The theory underlying Figure 1 also makes it possible to predict situations in which the IS development project is aborted (in its present form) during either the design or the implementation phase.

The abortion influence

The abortion influence includes cases in which the IS development project is either totally terminated or is aborted in its present form. The latter means that the governing idea of the information system to be developed is discarded during IS design and offset by a modified system, usually of reduced complexity, radicalness and originality; or that the information system selected for implementation is similarly modified at this stage.

Hypothesis 6. The level of originality has a positive influence on the number of IS development projects aborted or modified during the design phase.

The central assumption underlying this hypothesis is the positive relationship between the extent of design and the organizational support required. Even though the design itself increases commitment directly and indirectly via increased certainty and in that way increases organizational support, we do not expect this positive feedback to offset the increased support required initially. This is true especially at the early stages of design, when the uncertainty is greatest and the commitment lowest, while in the later stages the reinforcing influence of the design process becomes more important. This leads us to the corollary 1.

Corollary 1. The rate of abortion of IS development projects due to the level of originality is greater in the earlier stages of the design phase than in the later stages.

Hypothesis 6 is important from our viewpoint of the comparative implementability of in-house developed vs. application package based information systems, since it suggests that there is a filter effect in the design of original information systems. Information systems which are able to survive the design phase must have a considerable commitment, enabling the extensive experimentation required for a fair trial of the system to take place.

In more concrete terms, Hypothesis 6 implies that in the case of application package based information systems, where the level of originality is low and there is no need for extensive design, and especially when even that minor need is ignored because of the "parachuting mentality" (Keen, Bronsema and Zubott, 1982), all the problems are encountered during the implementation phase. Consequently, the implementation phase is extremely critical in these systems.

Hypothesis 7a. The level of originality has a mixed positive and negative influence on the number of IS development projects aborted or modified during the implementation phase.

The positive effect is based on Hypothesis 5, that the level of originality directly increases the extent of implementation required. This need is partially offset by the more extensive design. The more extensive design also creates the commitment and support required for the fair trial of the system, but according to Hypothesis 2 the level of originality has also a negative impact upon commitment. Finally, there is the filter effect of Hypothesis 6.

The theory underlying Figure 1 does not suggest any answers concerning the net effect of the level of originality upon the number of IS development projects aborted or modified during the implementation phase. We are, however, inclined to conclude that the net effect is negative.

Hypothesis 7b. The level of originality has a negative influence upon the number of IS development projects aborted or modified during the implementation phase.

The Modifying Influence of Complexity, Radicalness and Expected Level of the Benefit/Cost Ratio

The analysis in the two previous sections was based on the omission of other factors potentially influencing the impact of the level of originality upon the implementability of information systems. In this section the ceteris paribus assumption is partially relaxed by taking into account the potential modifying effect of complexity, radicalness and the expected level of the benefit/cost ratio.

Figure 1 assumes that the extent of design required depends on complexity and originality in a multiplicative rather than an additive manner. This reinforces the positive influence of the level of originality upon the implementability of information systems according to Hypotheses 3 and 4.

Hypothesis 8. The complexity of an information system increases the importance of the indirect negative influence of the level of originality on the uncertainty related to the IS development

Hypothesis 9. The complexity of an information system increases the importance of the indirect positive influence of the level of originality on commitment to the information system.

Since both these hypotheses reduce the extent of implementation required, they suggest that rapid implementation of an application-package based information system is a more feasible strategy in relative terms in the case of small systems than in the case of more complex systems, as we anticipated in our introductory part (Keen, *et al.*, 1982; Kole, 1983).

The increased complexity also reinforces the filter effect:

Hypothesis 10. The complexity of an information system increases the positive influence of the level of originality on the number of IS development projects

aborted or modified during the design phase.

Hypotheses 8-10 combined imply that the implementability of complex application package based information systems of a low degree of originality is difficult compared with complex systems developed in-house. Complexity furthermore directly increases the uncertainty and thus the need for extensive implementation. We have not made any assumptions as to whether this influence is related to originality in a multiplicative or additive manner, and therefore we do not have any specific hypotheses on whether or not complexity reinforces the influence of the level of originality as expressed in Hypotheses 1, 2, 5 and 7.

Complexity has its own impact relationships with the implementability of information systems, of course, but it is beyond the scope of this paper to discuss them in any more detail (see Iivari, 1986b).

Like complexity, radicalness is assumed to interact with originality by increasing uncertainty and reducing commitment, both of which increase the extent of implementation. As in the case of complexity, we do not have any specific assumptions about the interaction relationship between radicalness and originality, and therefore we cannot draw any specific conclusions regarding the potential reinforcing influence of radicalness in the case of Hypotheses 1-7.

When radicalness is combined with complexity there is an additional factor, the higher possibility that IS development will turn out to be a political process. In such a case it may be dominated by political considerations (Markus, 1979, 1983; Markus and Pfeffer, 1983), so that other factors may have quite a marginal role. But particularly in this case of complex and radical information systems, the filter effect of Hypothesis 6 can be expected to be highly influential, eliminating the most difficult IS development projects during the design phase.

Finally, there is the expected benefit/cost ratio. We have not made any assumption about the impact of the level of originality upon this ratio, since the influence obviously varies from case to case, neither do we have any specific assumptions about the interaction between the level of originality and the level of the benefit/cost ratio in the causal model in Figure 1. It is clear, however, that when the ratio is expected to be high, there is motivation to implement the system, the high motivation making it possible to reduce the extent and duration of the implementation required. Keen, *et al.*, (1982) mention this as a major reason why the parachuting strategy succeeded in four local units of an international bank.

FINAL COMMENTS

We have put forward ten hypotheses regarding the impact of the level of originality upon the implementability and probability of implementation of information systems. The hypotheses are theoretical, being derived from innovation research in particular, but we have tried to illustrate them by means of the few case studies available.

One cannot reach any absolute conclusions on this basis regarding the relative implementability of in-house development vs. package based information systems, but the results clearly support Lynch's finding that application packages have hidden costs and that their implementation is more difficult than one might initially expect (Lynch, 1984).

The theoretical model proposed in this paper attributes the difficulties to the short period of design in the case of application packages, implying great uncertainty and a low level of commitment upon the commencement of implementation. Also, there is no filter effect during the design. These adverse consequences are in practice accentuated by the prevalent "off the shelf" image of application packages and the "parachuting mentality" entailed in their implementation. This leads to the most evident practical conclusion of this paper, that more attention should be paid to the design phase in the case of application package based IS development. More extensive design, including an analysis of the organizational setting, the requirements placed upon the system, and the available packages obviously has many positive effects as to the selection and acquisition of the package, but it also has positive side-effects on implementation. As Lynch puts it: "When considering purchasing a package one must realize that some of what is gained by not doing the development (or design in terms of this paper) is given back in increased implementation costs," (p. 234).

Acknowledgements

The author is grateful to his colleague Erkki Koskela for his cooperation in the development of the PIOCO model and to the two anonymous referees for their valuable comments which helped to clarify some important points in the paper.

REFERENCES

Belady, L.A. "Complexity of Large Systems," in *Software Metrics,* A. J. Perlis, F. G. Sayward, and M. Shaw (eds.), The MIT Press, Cambridge, Massachusetts, 1981.

Bostrom, R.P. and Heinen, J.S. "MIS Problems and Failures: A Socio-technical Perspective, Part I: The Causes," *MIS Quarterly,* Volume 1, Number 3, September 1977.

Brunson, N. "The Irrationality of Action and Action Rationality: Decisions, Ideologies and Organizational Actions," *Journal of Management Studies,* Volume 19, Number 1, 1982.

Curtis, B. "The Measurement of Software Quality and Complexity," in *Software Metrics,* A. J. Perlis, F. G. Sayward, M. Shaw (eds.), The MIT Press, Cambridge, Massachusetts, 1981.

Davis, G.B. and Olson, M.H. *Management Information Systems, Conceptual Foundations, Structure and Development,* McGraw-Hill, New York, New York, 1984.

Dutton, W.H. and Kraemer, K.L. "The Politics of Modelling: Computer Models in the Policy Process," *Systems, Objectives, Solutions,* Volume 3, Number 1, 1983.

Falkenberg, E., Nijssen, G.M., Adams, A., Bradley, L., Bugeia, P., Campbell, A.L., Carkeet, M., Lehmann, G. and Shoesmith, A. "Feature Analysis of ACM/PCM, CIAM, ISAC and NIAM," in *Information Systems Design Methodologies: A Feature Analysis,* T. W. Olle, H. G. Sol, and C. J. Tully, (eds.), North-Holland, Amsterdam, Holland 1983.

Franz, C.R. and Robey, D. "An Investigation of User-led System Design: Rational and Political Perspectives," *Communications of the ACM,* Volume 27, Number 12, 1984.

Gross, P.H.B. and Ginzberg, M.J. "Barriers to the Adaption of Application Software Packages," *Systems, Objectives, Solutions,* Volume 4, Number 4, 1984.

Hage, J. *Theories of Organizations,* John Wiley & Sons, New York, New York, 1980.

Iivari, J. and Koskela, E. "PIOCO Model for a Data System," in *Report of the Third Scandinavian Research Seminar on Systemeering Models,* K. Lyytinen and E. Peltola (eds.), Jyv skyl , Finland, 1980.

Iivari, J. and Koskela, E. "An Extended EAR Approach for Information System Specification," in *Entity-Relationship Approach to Software Engineering,* C. G. Davis, S. Jajodia, P. A. Ng, R. T. Yeh (eds.), North-Holland, Amsterdam, Holland, 1983.

Iivari, J *Contributions to the Theoretical Foundations of Systemeering Research and the PIOCO Model,* Acta Universitas Ouluensis, A150, Oulu, Finland, 1983.

Iivari, J. "A Planning Theory Perspective on Information System Implementation," in *Proceedings of the Sixth International Conference on Information Systems,* L. Gallegos, R. Welke, and J. Wetherbe (eds.), Indianapolis, Indiana, 1985

Iivari, J. "Dimensions of Information Systems Design: A Framework for a Long-Range Research Program," *Information Systems,* Volume 11, Number 2, 1986.

Iivari, J. "An Innovation Research Perspective an Information Implementation," forthcoming in *International Journal of Information Management,* 1986.

Keen, P.W.G., Bronsema, G.A. and Zubott, S. "Implementing Common Systems: One Organization's Experience," *Systems, Objectives, Solutions,* Volume 2, 1982.

Kling, R. and Iacono, S. "The Control of Information Systems Developments After Implementation," *Communications of the ACM,* Volume 27, Number 2, 1984.

Kole, M.A. "A Non-Developmental MIS Strategy for Small Organizations," *Systems, Objectives, Solutions,* Volume 3, Number 1, 1983.

Langefors, B. "Information Systems," *Information Processing 74,* North-Holland, Amsterdam, Holland, 1974.

Lucas, H.C. Jr *Implementation, The Key to Successful Information Systems,* Columbia University Press, New York, New York, 1981.

Lynch, R.K. "Implementing Packaged Application Software: Hidden Costs and New Challenges," *Systems, Objectives, Solutions,* Volume 4, Number 4, 1984.

Markus, M.L. and Pfeffer, J. "Power and the Design and Implementation of Accounting and Control Systems," *Accounting, Organiz-*

ations and Society, Volume 8, Number 2/3, 1983.

Markus, M.L. *Understanding Information System Use in Organizations: A Theoretical Explanation,* Department of Organizational Behavior, Case Western Reserve University, 1979.

Markus, M.L. "Power, Politics and MIS Implementation," *Communications of the ACM,* Volume 26, Number 6, 1983.

Pelz, D.C. and Munson, F.C. "Originality Level and the Innovating Process in Organizations," *Human Systems Management,* Volume 3, Number 3, 1982.

Robey, D. "Information Systems and Organizational Change: A Comparative Case Study," *Systems, Objectives, Solutions,* Volume 3, Number 3, 1983.

Zaltman, G., Duncan, R. and Holbek, J. *Innovations and Organizations,* John Wiley & Sons, New York, New York, 1973.