Characterizing IS Development Projects

Kees van Slooten and Bert Hodes
School of Management Studies, University of Twente
P.O. Box 217, 7500 AE Enschede, The Netherlands
E-mail: cvs@sms.utwente.nl

Abstract

The relationship between project context and project situation is described by defining a number of contingency factors and components of a project approach. The applied contingency model is based on existing literature about situated method engineering. Relationships between contingency factors and the components of the project approach are analyzed for nine non-standard projects of the systems development department of a bank organization. The conclusion is that the choices of project managers concerning the project approach can be related to the project situation. The result of this research is a starting point for a contingency approach of information systems development projects in a bank.

Keywords

Systems development, situated methods, method engineering

1 INTRODUCTION

During a field-study, the organization of the information systems development process of a major bank in the Netherlands was investigated. Up to now, centralization of computer-based data processing was the main approach, including one standard approach to information systems development. However, more specific requests from clients and an increasing dynamic environment require more flexibility and variety from the applied approaches to information systems development. New trends in technology like client/server, relational database, fourth generation tools, end-user computing, object orientation, office automation, groupware and multimedia will influence information systems development. Consequently, one standard approach to information systems development will not suffice and more situation-specific approaches will be necessary. The need for situation-specific approaches has also been emphasized by: Kumar and Welke (1992), Van Slooten and Brinkkemper (1993), Vessey and Glass (1994).

   The concept of Methodology Engineering has been an attempt of Kumar and Welke to define the next level of evolution of methodologies. They discuss the need to customize

methodologies to meet the requirements of the development context. Van Slooten and Brinkkemper prefer the term Method Engineering instead of Methodology Engineering. Subsequently, we follow the terminology of Van Slooten and Brinkkemper, and especially of Van Slooten (1995).

Method Engineering is performed by configuring a project approach or situated method for systems development, utilizing existing method fragments to serve the project in context. Figure 1 is a simplified representation of situated method engineering. Method fragments are coherent components of existing methods. The project context includes the existing systems development organization, the customer organization, the supplier organization, the area of application, information and computerization policies, etc. Contextual or contingency factors, derived from the project context, are important for the entire method engineering process (arrow 1). However, it may sometimes be desirable to change the project context as a result of the method engineering process (arrow 2).

The configuration process comprises characterizing the project and selecting or constructing a situated method. The most important project contingency factors are determined during project characterization as a result of interviewing, brain storming sessions, questionnaires or other knowledge acquisition techniques. The prevalent contingency factors are utilized for the selection or construction of a situated method (arrow 5). This is supported by a method engineering information system, consisting of formalized rules and a method base. The components of the method base are method fragments and route maps. Route maps are plans associated with development strategies, including the activities to be performed and the products to be delivered. The method engineering information system can be considered as a knowledge-based information system supporting the configuration process. It contains method fragments and route maps for the construction of a situated method (arrow 6).

A systems development project is initially started, using the situated method determined during the configuration process (arrow 3).
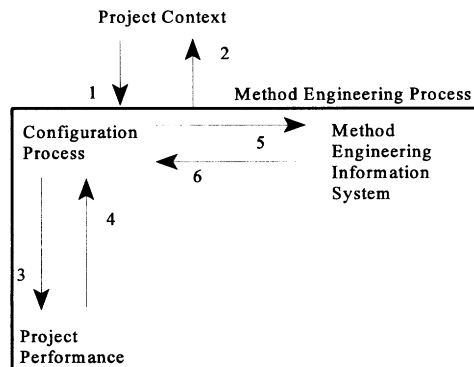

Figure 1. Situated Method Engineering

Unforeseen contingency factors may arise during project performance necessitating im provements and/or clarifications of the project characterization and an adjustment of the situated method (arrow 4). Evaluations during and after project performance may yield new knowledge about situated systems development, which is stored by the method engineering information system (included by arrow 4 and 5).

The bank organization is in the middle of a process of developing a new architecture for information systems development in which the contingency approach takes a central position. This means that various approaches must be available from which the best fitting is chosen depending on the project context. Also Necco (1987) already said that guidelines should be developed within the organization to provide direction for various approaches, which the organization selects for its systems development process. However, before formulating guidelines, it is necessary to know more about possible choice alternatives, prevalent contingency factors, and the relationship between the contingency factors (section 2) and the choice alternatives (section 3). The contingency model (section 2.1) is based on a situated method engineering approach (figure 1, after Van Slooten 1995). The choice alternatives have been made explicit after analyzing the existing practice of information systems development projects within the bank organization. The analysis of actual projects in practice also made some relationships between contingency factors and project approaches available (section 4). Projects selected for this research, are non-standard projects. We avoided standard projects, because standard approaches are linked to standard projects, which will not reveal much information about the relationship between contingency factors and choice alternatives. Furthermore, the research was focused on choices that can be made by project managers, and not on decision making by higher or lower levels of agents in the organization, which is outside the scope of this research.

2   CONTINGENCIES AND CONSTRAINTS

2.1 The contingency model

The contingency model of this research is based on the situated method engineering approach of figure 1. But the focus of this field study is on determining contingency factors as components of the project context, eliciting route maps and method fragments as components of possible project approaches, and relating contingency factors and project approaches (figure 2). The components of approaches may consist of methods, techniques, and tools for information systems development as well as for project management. This means an extension of the definition of method fragments. Contingency factors are variables from the project context with a certain value between Low (L) and High (H) that affect the project approach. Constraints can be considered as a specific kind of contingency factors causing limitations for the approach.
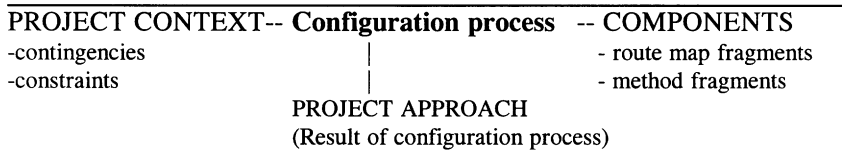
PROJECT CONTEXT-- **Configuration process** -- COMPONENTS		
-contingencies	|	- route map fragments
-constraints	|	- method fragments
	PROJECT APPROACH	
	(Result of configuration process)	

Figure 2   Contingency model

2.2 Contingency factors

The explicitation of contingency factors is based on the work of Van Der Hoef et al. (1995). They composed a list of contingency factors and constraints, which is the product of collecting and integrating existing lists from various sources. However, we removed some inconsistencies from this list and selected the most important factors for the field study according to experts of the bank. Some factors, which have the same value for each project (e.g. the quality of information planning), are outside the scope of the field study. Other factors are a generalization or a specialization of the factors of Van Der Hoef et al. Finally, the list of contingency factors is:

- *Management commitment*. To what extent management supports the project.
- *Importance*. To what extent the project or information system is important for the organization.
- *Impact*. To what extent the information system will change business operation after implementation.
- *Resistance and conflict*. To what extent stakeholders have different or conflicting interests.
- *Time pressure*. To what extent the available time for the project is experienced as insufficient.
- *Shortage of human resources*. To what extent the number of people available for the project is experienced as insufficient.
- *Shortage of means*. To what extent the means available for the project are experienced as insufficient.
- *Formality*. To what extent there are lasting rules, procedures, and standards for the business processes and supporting information.
- *Knowledge and experience*. To what extent the users possess enough knowledge and experience to develop the required information system.
- *Skills*. To what extent the members of the project-team possess enough knowledge and experience to develop the required information system.
- *Size*. The number of people being a member of the project-team.
- *Relationships*. To what extent there are relationships between the new information system and other information systems.
- *Dependency*. To what extent the project depends on activities and conditions outside the project.
- *Clarity*. To what extent the goals, needs, and desires of the users are clear and coherent enabling a sound specification of the functional requirements.
- *Stability*. To what extent the goals, needs, and desires of the users will not change over

time enabling a stable specification of the functional requirements.
*   *Complexity*. To what extent the functional components of the information system are complex.
*   *Level of innovation*. To what extent the applied technology and/or the applied methods, techniques, and tools are new to the organization.

2.3 Constraints

Constraints are specific contingency factors without a relative value between Low (L) and High (H), but they definitely affect the project approach. Constraints are specific circumstances restricting the number of choice alternatives and affecting the relationships between contingency factors and project approach. The influence of constraints on these relationships is outside the scope of this field study. One may distinguish five kinds of constraints: Contracts, Type of information system, Standards, Technical constraints, and External factors. We do not want to go into detail, because it is not part of this research.

3   CHOICE ALTERNATIVES

3.1   Definitions

The situated method engineering approach contains two kinds of building blocks: route maps and method fragments. To describe the situation of the bank more precisely, we shall define the concepts *route map fragment* and *method fragment* as follows.
A *route map fragment* is a coherent part of the complete route map of a systems development project. A route map fragment may refer to strategies, activities, and products concerning systems development as well as project management.
A *method fragment* is a coherent part of a method(ology) for systems development or project management. Method fragments may be linked to a route map, which may establish a complete project approach or a situated method.

3.2   Route map fragments

Tracing and dividing
One of the first activities of a project manager is to determine the scope of the project in co-operation with the users. We distinguish two possibilities for tracing the business functions for the project: *tight (T)* and *wide (W)*. Tight tracing means that the functionality required will partly be realized outside the project. Wide tracing means that the functionality required will be completely realized during the project. Related to tracing the functionality is dividing the functionality into subsystems, which will be developed separately. We distinguish: *one system (o)* and *subsystems (s)*.

Delivery strategy
The delivery strategy is the way of delivering and introducing the information system in the organization. We distinguish three options: *at once (o)*, *incremental (i)*, and

evolutionary (e).
- (o). Delivery at once means that the entire system is delivered at once.
- (i). Incremental delivery means that the system is delivered by a serial delivery of subsystems, each containing a part of the functionality.
- (e). Evolutionary delivery means that the system is delivered by successive versions of the entire system partly containing the entire functionality. Functional requirements may change between two versions.

The delivery strategy deals with subsystems and not with subprojects. The distinction between subsystems and subprojects is important throughout this paper. Different stages of developing one (sub)system may be realized by different subprojects.

Realization strategy

The realization strategy is the way of realizing the various subsystems with respect to sequence and concurrence. We distinguish four options: *at once (a)*, *concurrent (c)*, *overlapping (o)*, and *incremental (i)*.
- (a). Realization at once means that the entire information system is developed at once.
- (c). Concurrent realization means that all subsystems are concurrently developed.
- (o). Overlapping realization means that some subsystems are concurrently developed and other subsystems consecutively.
- (i). Incremental realization means that all subsystems are developed one after another.

Establishing subprojects

There are several ways to divide information systems development into subprojects. We distinguish four options: *one project (o)*, *process-oriented (p)*, *system-oriented (s)*, and *hybrid (h)*.
- (o). One project means no division into subprojects.
- (p). Process-oriented means division into subprojects based on information system development subprocesses.
- (s). System-oriented means division into subprojects based on subsystems.
- (h). Hybrid means division into subprojects partly based on subprocesses and partly on subsystems.

Project organization

Of course, one needs a project organization to run the project. Decisions have to be taken about who is involved and who is responsible for what takes place. A communication structure is provided describing on which levels communication is necessary and its frequency. The project manager may choose a *standard (s)* or an *adapted (a)* structure for the project organization.

Project management products and activities

Other activities of project management are for example: estimating risks, determining the required means, investigating the consequences of the project. These activities, plans and reports concern the performance of the project. These project management products may be *standard (s)* or *adapted (a)*. In the first case the format and the contents are well defined. In the second case we have to deal with more informal project control.

Development strategy

The development strategy is a generic strategy for the sequence and the selection of activities supporting the development of a system (that is) not further divided into subsystems. Also based on Van Slooten and Schoonhoven (1994) we distinguish five options: *phase-wise (p)*, *tile-wise (t)*, *prototyping (g)*, *iterative (i)*, *outsourcing (o)*.

- (p). Phase-wise is strict linear development without prototyping.
- (t). Tile-wise is linear development with partly overlapping phases.
- (g). Prototyping is linear development including prototyping, so-called throwaway prototyping. During functional design a prototype is built to improve the functional requirements or to show the feasibility of a certain technology.
- (i). Iterative or keep-it prototyping. The cycle of analysis, design, implementation and evaluation is reiterated several times. After each iteration the system may be adapted until there are no additional requirements.
- (o). Outsourcing or software package selection means that the system is not developed by the bank organization. Before outsourcing the functional requirements are determined by the bank organization. Required modifications of a software package are realized by the supplier.

System development products and activities

Project management determines which system development products must be delivered. There is a standard list of products, but the project manager may construct his own list if he has good reasons to do so. The products may describe different aspects of the business system and/or the information system. System development activities must be determined to develop the products. Possible options are *standard (s)* and *adapted (a)*.

- (s). Standard if the standard list is used.
- (a). Adapted if the standard list is not completely used.

3.3 Method fragments

Method fragments may come from methods, techniques, and tools for project management as well as systems development. There is a standard way of working for project management, which is described in a manual. The manual contains descriptions of activities for project management and techniques and tools that should be applied. This means that the project manager has two options: *standard (s)* and *adapted (a)*.

- (s). Standard means that the project manager follows the manual to the letter.
- (a). Adapted means that he changes the standard.

Methods, techniques, and tools for systems development deal with the contents of the information system that must be developed. The standard method for the bank organization is Method/1 and a few other tools for specific tasks. This means that here too there are two options: *standard (s)* and *adapted (a)*.

4   RESEARCH APPROACH AND RESULTS

4.1 Projects

The following nine projects, with deviations from the standard approach to systems development within the bank organization, have been selected for this field study:

- Developing an information system establishing data administration to enter the exchange market of shares for a major telecommunication organization in Holland.
- Realizing some changes in information systems supporting business in stocks, which is necessary for maintaining a certain service-level and realizing some changes.
- Developing a new information system dealing with information supporting questions and complaints concerning foreign promotion activities.
- Developing an information system for processing guaranteed means of payment such as cheques, utilizing imaging technology.
- Enhancing a voice-response application with functionality for transactions by phone.
- Re-designing the back-office for business in stocks by developing an information system based on a software package for storing stock transactions.
- Modifying a number of heavy applications to decrease the workload of the mainframe computers.
- Developing an information system that is capable of collecting, enriching, storing, and distributing data from various central databases, supporting various accounting information systems.
- Developing a pilot information system in a client/server environment, supporting the communication between advisers and clients.

Table 1 represents the contingency factors related to the fragments of approaches to systems development for the nine projects. Deviations from the standard approach are printed in bold type. The contingency factors may have the following values: 1 (low), n (normal), or h (high). These values are determined by interviewing the project managers and by sending them a questionnaire to respond. Some contingency factors did not cause a deviation from the standard, e.g. the factor 'resistance and conflict'. Sometimes, the standard approach may allow more than one value of a contingency factor or approach fragment. The standard approach is defined as follows: (Tracing = tight, Dividing = one system, Delivery Strategy = at once, Realization Strategy = at once, Establishing Subprojects = one project, Project Organization = standard, Project Management Products and Activities = standard, Development Strategy = phase-wise, Systems development Products and Activities = standard, Project Management Method Fragments = standard, Systems Development Method Fragments = standard).

Table 1   Contingency factors and approaches for the nine projects

Contingency factors	1	2	3	4	5	6	7	8	9
Management commitment	h	h	n	n	n	h	n	n	h
Importance	h	h	h	h	h	h	h	h	h
Impact	n	h	h	h	n	h	n	h	n
Resistance and Conflict	l	n	n	n	n	n	n	n	n
Time pressure	h	h	n	n	n	n	h	n	h
Shortage of human resources	h	h	l	l	l	l	n	n	l
Shortage of means	l	l	n	n	n	n	n	n	l
Formality	h	h	h	h	h	h	h	l	l
Knowledge and experience	h	h	h	h	h	h	h	h	h
Skills	h	h	n	n	n	l	h	n	n
Size	h	h	n	n	h	n	h	h	n
Relationships	n	h	h	h	h	h	h	h	n
Dependency	n	h	l	l	h	h	n	l	l
Clarity	h	n	h	h	h	l	h	l	l
Stability	n	l	n	l	l	l	h	l	l
Complexity	l	h	n	n	h	h	h	h	h
Level of innovation	l	l	h	h	h	h	l	h	h

Approach

Approach	1	2	3	4	5	6	7	8	9
Tracing (*t = tight, w = wide*)	t	t	t	w	w	w	w	w	t
Dividing (*o = one system, s = subsystems*)	s	s	s	s	s	o	s	s	o
Delivery Strategy (*o = at once, i = incremental, e = evolutionary*)	o	e	i	o	o	o	i	i	o
Realization Strategy (*a = at once, c = concurrent, o = overlapping, i = incremental*)	c	o	o	c	c	a	o	o	a
Establishing Subprojects (*o = one project, p = process-oriented, s = system-oriented, h = hybrid*)	h	h	s	o	o	p	h	h	o
Project Organization (*s = standard, a = adapted*)	a	a	a	s	s	a	a	a	a
Project Management Products and Activities (*s = standard, a = adapted*)	a	s	s	s	s	a	a	s	s
Development Strategy (*p = phase-wise, t = tile-wise, g = prototyping, i = iterative, o = outsourcing*)	to	t	po	po	to	o	t	tgi	i
Systems Development Products and Activities (*s = standard, a = adapted*)	a	s	a	s	s	a	a	s	a
Project Management Method Fragments (*s = standard, a = adapted*)	s	s	s	s	s	a	s	s	s
Systems Development Method Fragments (*s = standard, a = adapted*)	s	a	s	s	s	a	a	a	a

4.2 Discussion of relationships

Management commitment and importance
The factors *management commitment* and *importance* are considered as one factor, because it was difficult to deal with these factors separately. The importance of the project and commitment of management affect the project a lot. Cooperation and flexibility of groups of specialists in the organization will increase considerably if one can rely upon strong interest of management, implying a project of high priority. However, the influence on the actual approach is limited, only the project organization was adapted for three projects that were of great importance.
Project organization. A first consequence was to involve senior employees as a kind of sponsor of the projects, which means decisions will be taken at a higher level. During one project, a board of managers of various business units was available to take important decisions. A second consequence, if we have to deal with high time pressure as well, was including people from the department for computer and network facilities responsible for the technical operation of systems.

Impact
The influence of developing an information system on the users organization depends on to what extent business operation will change because of implementing the system. Important aspects are the number of people for whom work will change and to what extent the work itself will change. The impact of the information system hardly affects the approach to the project. Five of the nine projects had to deal with high impact, one of which had to change the delivery strategy.
Delivery strategy. The delivery strategy for standard projects is delivery of the whole system at once. It was already mentioned that for only one project, impact of the information system was a reason for choosing another delivery strategy, namely an evolutionary strategy. Changing the users organization at once should not be acceptable.

Time pressure
We have to deal with projects that have a deadline. Time pressure increases if the available time becomes much less than the time needed. The high time pressure for four of the nine projects affected a number of approach fragments: tracing, realization strategy, project organization, project management products and activities, development strategy, and systems development products and activities.
Tracing. High time pressure was a reason to limit the functionality of the information system for the time being. A small and simple application with limited functionality can be realized in a shorter time.
Realization Strategy. For three projects, high time pressure was a reason for choosing a concurrent or overlapping realization. Concurrent development of all subsystems should occur as much as possible to decrease the time elapsed for the whole project. Sometimes, a concurrent realization strategy was not possible because of a lack of human resources. In that case, an overlapping realization strategy was chosen.
Project Organization. High time pressure affected the project organization in different ways:

- To tune activities, oral communication was emphasized instead of written documents saving a lot of time.
- Utilizing external workers if nobody else is available. Generally, external workers lack knowledge about the existing systems. Consequently, projects were organized enabling cooperation between internal and external workers.
- Keeping people from specific departments like quality assurance, system management, or computer and network facilities (production management) outside the project organization, if their contribution can be missed.
- Including people from the production management department (computer and network facilities) into the project, if one may expect problems during the transfer of the system, ready for actual operation, to the production management department.

Project Management Products and Activities. Due to time pressure it was decided for one project to deliver only a limited number of project management products like plans for quality assurance, risk management, approach to the project, documentation management, etc.

Development Strategy. Due to time pressure the following development strategies were selected:

- A tile-wise development strategy means that the next phase will start before the current phase is finished. Formal approval of one or more phases was postponed, because this takes time. Sometimes two or more phases were turned into one phase to save time.
- Because of time pressure or shortage of human resources an outsourcing strategy was frequently chosen. The functional requirements were determined by the bank, after which the remaining phases were established by an external organization. However, accepting and introducing the system were again internal activities.

Systems Development Products and Activities. Because of time pressure sometimes only systems development products and connected activities were selected which were absolutely necessary. Sometimes other products were delivered instead of the products, prescribed, if it speeded up the process. It is easier to obtain approval for delivering less or other products if the importance of the project is high.

Shortage of human resources

In only two of the nine projects the shortage of human resources was high. This contingency factor affected the project organization and the delivery strategy.

Project Organization. The shortage of human resources was resolved by hiring external workers if the budget for the project was sufficient.

Delivery Strategy. In one project, shortage of resources was a reason for partly outsourcing systems development. Because of time pressure it was not possible to postpone systems development.

Formality

In seven out of nine projects the formality was high as in standard projects. This contingency factor only affected the systems development products and activities.

Systems Development Products and Activities. In two projects with a high value for formality less products were made, because some products to be made were similar to already existing system documents, e.g. the data model, which means that in such cases it was possible to use existing products.

Size

The size of five projects was high (more than ten persons). The following approach fragments were affected by the size of the project: tracing, dividing, realization strategy, and establishing subprojects.

Tracing. Problems with the management of a project will arise if many people are working for the project at the same time. Therefore, it was tried to trace the project as tight as possible, which means postponing or deleting functionality, if not absolutely necessary. Another way of limiting the size of a project is lengthening the time for the project. However, this was not possible because of a fixed deadline.

Dividing. The size of the project was mentioned most often as the reason to divide the system into subsystems. Therefore, the functionality was divided into coherent subsystems enabling independent development of these subsystems.

Realization Strategy. In a number of projects was chosen for an overlapping strategy instead of realization at once, because of the size of the project. Through realizing the subsystems partly in sequence instead of all subsystems concurrently, it was possible to limit the size of the project. Further decreasing the size of the project by using an incremental realization strategy was often not possible because of high time pressure.

Establishing subprojects. Establishing subprojects was affected for some projects by the size of the project. Generally, subprojects are established for recognized subsystems. However, size was the reason for a number of projects to choose a hybrid approach for establishing subprojects, which means that also for certain phases of the systems development process different subprojects are established.

Dependency

Dependency was high for three projects, but the approach of these projects was hardly affected, only dividing into subsystems.

Dividing. Dividing into subsystems was affected by the dependency of other activities in only one project. There was a strong dependency of a system in the middle of a development process. Therefore, it was decided to consider the functionality that was dependent on another project as a separate subsystem. This subsystem was developed after the other project had been finished.

Clarity and stability

The reason for joining clarity and stability of the functional requirements is that the approach was only affected by instability of the functional requirements if, at the same time, clarity of the functional requirements was low. There was also a relationship between the formality of the business processes, and the clarity and stability of the functional requirements. If the formality was low, then the clarity and stability of the functional requirements were also low. Unclear functional requirements affected the tracing of the functionality and the development strategy. Instability of the functional requirements only affected the development strategy.

Tracing. Unclear requirements of users were the reason for one project to limit initially the functionality of the system. A large number of interest groups put forward their own specific and often conflicting ideas about the application. Consequently, functionality was restricted to common requirements.

Development Strategy. An iterative development strategy was chosen for two projects

instead of the usual phase-wise development strategy, because of unclear and unstable user requirements. The requirements were determined and realized during a first development cycle, after which the user could improve his requirements by using the application developed. A precondition for choosing such a strategy is the availability of CASE-tools that facilitate a rapid application development.

Complexity

For six projects complexity was high. The complexity of the functional components of the system affected the way of tracing and dividing functionality, establishing subprojects, project organization, the development strategy, and the systems development method fragments.

Tracing. High complexity of the required functionality was for two projects the reason for limiting functionality with consequently fewer problems during systems development.

Dividing. High complexity of the system was during one project one of the reasons for dividing the functionality into subsystems. The functionality was divided into two subsystems. Different kinds of expertise were required for developing these subsystems.

Establishing Subprojects. In a number of projects the complexity was a reason for choosing a process-oriented or hybrid approach to establish subprojects for certain processes or phases of systems development. In two cases it was decided to test the application in a separate subproject. In another project with many modifications of existing systems the analysis phase of these systems was established in a separate subproject because of the complexity and the different kind of expertise that was needed for the various systems. Finally, in one project complexity was the reason for realizing a data model in a separate subproject, because specific knowledge was necessary, which was not available in the project-team of the systems development department.

Project Organization. In one project, complexity was the reason for involving people with specific expertise from various departments. One dealt with a technical migration project including technical improvements of existing systems. Because of the complexity it was decided to add experts in databases, hard systems software, etc. to the project-team.

Development Strategy. In one project the complexity was the reason for choosing a tile-wise development strategy. A badly documented system had to be modified. It was decided to start a functional and technical design as partly overlapping phases enabling a clear specification of what should be modified. In another project complexity was the reason for choosing an outsourcing strategy, because an existing software package was more appropriate than internal development of a new application.

Systems Development Method Fragments. In two projects, high complexity was the reason for applying other tools than standardly available. In one project monitoring tools supporting the analysis of complex systems were necessary. In another project a tool for testing programs was necessary, because of the complexity of the interaction of many interacting subsystems.

Level of innovation

In principle, the applied technology is part of the approach in correspondence with functionality. However, during the nine projects the technology was mostly supplied by the users organization. Up to now the central mainframe was the standard platform for running the applications, which means that other environments like LAN, WAN, or PC

are new. The level of innovation was high in non-mainframe environments (six projects). The level of innovation affected the division of the system, the project organization, the development strategy, the systems development products and activities, and the systems development method fragments.

Dividing. In three projects the level of innovation was a reason to divide the system into subsystems. The functionality was divided into two subsystems. One was realized on the mainframe (communication with central systems and processing of central data) and the other on the decentralized environment (the actual application).

Project Organization. In two projects the level of innovation was the reason to involve external workers in the project organization. They participated in project- and working groups with the intention of knowledge transfer from the external workers to the workers of the bank. In other projects of a high level of innovation outsourcing the innovative part of the application was preferred (see also effect on the development strategy). Mutual adjustment of the functional specifications was necessary in this case, enabling the co-operation between the internal and external parts of the application.

Development Strategy. The high level of innovation did affect the development strategy of various projects. It depends on the question whether the organization likes to acquire more knowledge of the new technology or not. In two projects an innovative system was developed in cooperation with an external supplier. For one of these systems, the high level of innovation was the reason for choosing a prototyping strategy. A prototype was constructed to estimate the feasibility of the new technology. An iterative development strategy was used for the development of the prototype. The final specifications were determined by evaluating and modifying the prototype. In the other project (pilot) the prototype was experimental, which meant in this case that the desirability and feasibility of a new kind of application was investigated. An iterative development strategy was chosen again. In three other projects with no intention of transferring knowledge, an outsourcing strategy was chosen for the development of the innovative part of the application. Outsourcing of the development of a subsystem also affects the project management products and activities. In this case, one has to deal with a contract with the external supplier, but such activities are mostly the responsibility of the user's organization with some assistance of a special department. The systems development department is responsible for the control of the contents of the activities of the external workers.

Systems Development Products and Activities. In one of the projects the high level of innovation was the reason to perform other activities and deliver other products than usual. The alternative environment and the tools available enabled another way of developing information systems. The usual systems development products did not fit here, because these usual products were based on the development of mainframe applications.

Systems Development Method Fragments. In one project the high level of innovation was the reason for not applying the standard methodology Method/1. The systems development activities and their sequence was determined by common sense. In two innovative projects the presence of new technology was the reason for applying new tools like: fourth generation environments, object-oriented programming languages, and tools for developing graphical user interfaces. These kinds of tools have until now not been applied in the mainframe environment of the bank, but only in client/server environments.

5 CONCLUSIONS AND FURTHER RESEARCH

This research shows that the project approach is affected by the project context, in spite of the present standards, procedures, and uniform way of working. Generally, a contingency approach to systems development was not supported by the bank organization. A project manager may construct a project approach by choosing various components of an approach as described by this paper. There are several options available for each component of an approach. In the former section it was described how the choices were affected by the contingency factors of the project situation. Some contingency factors did not affect the project approach at all (table 1). It was possible to explain the choices made by the project manager using the current set of contingency factors. However, this does not mean that the current set is the ultimate set of contingency factors for this organization. It is a starting point for further research. We have already seen that sometimes two contingency factors can be handled as one factor, e.g. *management commitment* and *importance*, *clarity* and *stability*. Of course, this research has some limitations:

- The projects have not been evaluated.
- Too few projects have been analyzed in order to support this research with quantitative results.

This means that the found relationships between contingency factors and project approach do not necessarily guarantee a 'best' approach. The relationships found are based on choices made by senior project managers.

This research aims to contribute to the development of a contingency model for systems development projects. Further research must be focused on the determination of successful relationships between the project context and the project approach by evaluating chosen project approaches depending on the situational factors, i.e. contingency factors including constraints.

6 REFERENCES

Hoef, R. van der, et al. (1995), Situatie, Scenario en Succes (Dutch), Memoranda Informatica, Internal Technical Report, University of Twente.

Kumar, K., R.J.Welke (1992), Methodology Engineering: A Proposal for Situation-Specific Methodology Construction. In: *Challenges and Strategies for Research in Systems Development*, Wiley and Sons Ltd.

Necco, C.R., et al. (1987), Systems Analysis and Design: Current Practices. In: *MIS Quarterly*, 11, 4.

Slooten, C. van, S. Brinkkemper (1993), A Method Engineering Approach to Information Systems Development. In: *Information Systems Development Process*, Proceedings IFIP WG 8.1, Elsevier Science Publishers (North-Holland).

Slooten, C. van (1995), Situated Methods for Systems Development, Thesis, University of Twente.

Slooten, C. van, B. Schoonhoven (1994), Towards Contingent Information Systems Development Approaches, In: *Methods and Tools, Theory and Practice*, Proceedings of ISD'94, Bled.

Vessey, I., R.L. Glass (1994), Application-Based Methodologies: Development by Application Domain. In: *Information Systems Management*, Fall 1994.