

STELLINGEN

behorende bij het proefschrift

Facilitating Organizational Change The Participative Application of Dynamic Modelling

Gert-Jan de Vreede
15 december 1995

I

Het ondersteunen van de participatie van belanghebbenden in een reorganisatieproces met behulp van een Group Support Systeem verhoogt de effectiviteit van dit proces.

(dit proefschrift)

II

Bestaande coördinatiemechanismen in een organisatie kunnen met interactie-, proces- en actormodellen *voor* en *door* de betrokkenen uit de organisatie inzichtelijk worden weergegeven.

(dit proefschrift)

III

Een prototype van een informatiesysteem kan pas werkelijk worden geëvalueerd als de toekomstige organisatorische context van het systeem eveneens wordt meegenomen.

(dit proefschrift)

IV

De bij het toepassen van reductie-operatoren gebruikte modelleurservaring maakt dat het zinloos is om de volledig automatische transformatie van conceptuele modellen in simulatiemodellen na te streven.

V

In tegenstelling tot reguliere elektronische vergaderingen zullen bij het participatief bouwen van conceptuele modellen kleine groepen efficiënter werken en meer tevreden zijn dan grote groepen.

VI

Ook in de modewereld geldt dat een model een beperkte weergave van de werkelijkheid is.

VII

Internet is geen 'information highway', maar een 'digital swamp'.

VIII

Het veelvuldig gebruik van Franse woorden en zinssneden als '*soit*', '*du moment*' en '*coûte que coûte*', is een overdreven poging het geschrevene of gesprokene meer *cachet* te geven.

IX

Bij het afsluiten van een promotie-onderzoek is er een regenboog in het hart: een lach omdat het werk is geklaard en een traan omdat een mooie periode wordt afgesloten.
(naar Kuifje, De Blauwe Lotus)

X

Door de opgang van het tele-thuiswerken doet zich het nieuwe verschijnsel tele-thuis-overwerken voor.

Facilitating Organizational Change

The Participative Application of Dynamic Modelling

Dit proefschrift is goedgekeurd door de promotor:

Prof.dr. H.G. Sol

Samenstelling promotiecommissie:

Rector Magnificus, voorzitter

Prof.dr. H.G. Sol, Technische Universiteit Delft, promotor

Prof.dr. G.W. Dickson, North Carolina State University

Prof.dr.ir. J.L.G. Dietz, Technische Universiteit Delft

Prof.mr.dr. E.F. ten Heuvelhof, Technische Universiteit Delft

Prof.dr. J.J. Ramondt, Erasmus Universiteit Rotterdam

prof.dr.ir. W.A.H. Thissen, Technische Universiteit Delft

To my father, who set the example and brought me to where I am now

To Sajda, whose love and support give it all a meaning

Colofon

Published and distributed by:

G.J. de Vreede
Westvest 283
2611 BX Delft
The Netherlands
tel: +31.15.212.4669

Delft University of Technology
School of Systems Engineering, Policy Analysis and Management
Jaffalaan 5
P.O. Box 5015
2600 GA Delft
tel: +31.15.278.7170
fax: +31.15.278.3429

The author acknowledges support by:

- The Technology, Labour and Organisation Programme (T.A.O.)
- The Amsterdam Municipal Police Force

Cover design by G.M.A. van der Harst

CIP-GEGEVENS ROYAL LIBRARY, THE HAGUE

Vreede, Gerardus Johannes de
Facilitating Organizational Change
The Participative Application of Dynamic Modelling
Doctoral dissertation Delft University of Technology
With literature list, author and subject index, and summary in Dutch
ISBN: 90-5638-006-0
Key words: management and organization / information systems

© Copyright 1995 G.J. de Vreede

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form by any means, electronic, mechanical, photocopying, recording, or otherwise, without the written permission of the author.

Contents

PREFACE XI

1 ORGANIZATIONS, INFORMATION TECHNOLOGY, AND COORDINATION

 1.1 Introduction I

 1.2 Organizational coordination 4

 1.3 Information technology to support organizational coordination 12

 1.4 Research question and research objective 18

2 RESEARCH APPROACH

 2.1 Introduction 21

 2.2 Research philosophy 21

 2.3 Research strategy 22

 2.4 Research instruments 25

 2.5 Research outline 32

3 DYNAMIC MODELLING FOR ORGANIZATIONAL CHANGE

 3.1 Introduction 35

 3.2 Way of thinking 36

 3.3 Way of working 37

 3.4 Way of modelling 39

4 CASE 1: THE DEPARTMENT OF UNEMPLOYMENT BENEFITS

 4.1 Introduction 63

 4.2 Problem situation 64

 4.3 The conceptual model 66

 4.4 The empirical model 76

 4.5 Problem diagnosis 82

4.6 Alternatives for change	84
4.7 Reflection	87
 5 CASE 2: THE NEUROLOGY OUTPATIENTS' DEPARTMENT	
5.1 Introduction	89
5.2 Problem situation	89
5.3 The conceptual model	92
5.4 The empirical model	101
5.5 Problem diagnosis	111
5.6 Alternatives for change	113
5.7 Reflection	118
 6 GROUP SUPPORT FOR PARTICIPATIVE DESIGN	
6.1 Introduction	121
6.2 Group Support Systems	123
6.3 Way of thinking	136
6.4 Way of working	137
6.5 Way of modelling	139
6.6 Mapping GSS to group design activities	144
 7 CASE 3: THE CRIMINAL INVESTIGATIONS DEPARTMENT	
7.1 Introduction	147
7.2 Problem situation	147
7.3 The conceptual model	154
7.4 The empirical model	163
7.5 Problem diagnosis	168
7.6 Alternatives for change	170
7.7 Reflection	182
 8 A DESIGN APPROACH TO FACILITATE ORGANIZATIONAL CHANGE	
8.1 Introduction	189
8.2 Conceptualize problem situation	190
8.3 Construct empirical model	195
8.4 Diagnose problem situation	202
8.5 Construct models for change	205
8.6 Choose change alternative	210
8.7 Implement change alternative	211
8.8 Concluding remarks	214
 9 CASE 4: THE CRIMINAL INVESTIGATIONS DEPARTMENT REVISITED	
9.1 Introduction	219
9.2 Problem situation	219

9.3 The conceptual model	223
9.4 The empirical model	233
9.5 Problem diagnosis	239
9.6 Alternatives for change	241
9.7 Reflection	254
10 EPILOGUE	
10.1 Introduction	267
10.2 The application area	267
10.3 The research approach	272
10.4 The design approach	273
10.5 Future research	283
REFERENCES	289
AUTHOR INDEX	311
SUBJECT INDEX	317
SAMENVATTING	323
CURRICULUM VITAE	335

Preface

'The story is one that you and I will construct together in your memory. If the story means anything to you at all, then when you remember it afterward, think of it, not as something I created, but rather as something that we made together.'

- Orson Scott Card

Change is one of the key words of our times. Faced with ever fluctuating internal and external demands, modern organizations are continuously forced to reconsider the way in which they structure and coordinate the execution of their processes. In this dissertation, we examined ways to facilitate such organizational change. We integrated techniques and tools such as 'dynamic modelling', 'simulation', 'animation', 'Group Support Systems', and 'prototyping' into a design approach that focuses on the participative analysis and design of improved coordination mechanisms and supporting Information Technology. Experiences from four case studies demonstrate the feasibility and applicability of this approach.

It would be paradoxical if a dissertation that addresses participative design were presented as an individual accomplishment. Surely, this is not the case. There are many people that I am indebted to. First of all, I was very fortunate to have Henk Sol as an inspiring supervisor. I thank him for his never ceasing motivation and coordination of my research efforts. Most of all I am grateful for his constructive criticism and for his support during my period of illness. I was also fortunate to meet Gary Dickson. He helped me to get a grip on my research by infecting me with the 'GSS-virus'. His advice and suggestions during my research I value extremely much.

The organizations that I visited for my research treated me very kindly. First, I like to thank the employees of the Neurology Outpatients' Department for their time and cooperation. In particular, I like to thank Foekje van Dijk for her enjoyable companionship during this study. Second, I want to express sincere gratitude for the research opportunities offered by the Criminal Investigations Department of the Amsterdam Municipal Police Force. I like to thank the CID's staff for their willingness to participate throughout the studies even if it meant spending one and half hours in traffic jams. Especially, I want to mention Ton Leeuwrik who introduced me to and

guided me through the inspiring world of combating organized crime. His enthusiasm during our 'participative design sessions' I found truly inspiring. I thank Bernard Welten for his ability to self-reflect, for our stimulating discussions, and for making research at the CID very gratifying.

Also, I want to thank my colleagues at the School of Systems Engineering, Policy Analysis and Management. In particular, I want to thank Alexander Verbraeck for our animated discussions, for being an infinite source of ideas and suggestions, and for his help with last minute 'graphical' emergencies. Jeroen van Meel showed me the way to Amsterdam where I greatly enjoyed working with him and learning from him. I admire his ability to give instant structural advice and his politeness to system managers. Finally, I thank Daan van Eijck, my dear roommate and partner in the 'far-away-and-beyond' conspiracy. Our joint brainstorming sessions and coordinated publishing policy are among the highlights of the past four years.

During my research, I had the pleasure to work with a couple of bright, hardworking students. I am very grateful to Stephan den Hengst for his enthusiastic work on the first CID study and for his excellence as a 'chauffeur'. Jan-Willem van Aalst, Michiel Bouwman en Bob Noorduin I thank for creating the ACOST prototype and for their 'Rotterdam work mentality'. To Gerard Kuijlaars I owe a lot for his inspired efforts in the second CID study; his contribution to my research was invaluable in making the design approach work.

Doing dissertation research is impossible without the unconditional support from the 'home-front'. I owe my parents greatly for providing me with their love and care and for bearing with me through the last hectic months. Most of all, I wish to express my deepest gratitude to my wife Sajda. Meeting her I owe to my research, but finishing the research I owe to her love and inspiration.

Delft
November 1995

Gert-Jan de Vreede

1 Organizations, information technology, and coordination

'Observe always that everything is the result of a change, and get used to thinking that there is nothing Nature loves so well as to change existing forms and to make new ones like them.'

- Marcus Aurelius

1.1 Introduction

In recent years, organizations have found themselves in a dynamic environment. Technological developments and breakthroughs often give rise to varying productivity rates and standards of living [Toffler 1985]. Organizations witness developments such as intensified competition, government regulations and interventions (e.g. environmental regulations), changing demands from consumers and employees, and fluctuations in financial markets. At the same time, there are drastic social and political changes affecting the lives and attitudes of the organization's workers [Francescato 1992; Morton 1992]. The industrial society, shaped by the ideas of early organizational scientists such as Fayol, Taylor, and Weber, is transforming into a new society labelled with terms as 'super-industrial society' [Toffler 1970], 'post-industrial society' [Huber 1984], 'Third Wave society' [Toffler 1985], or 'globally networked society' [Malone and Rockart 1991; Sol 1992a].

This post-industrial society can be characterized with expressions such as 'complex', 'turbulent', and 'hostile', see e.g. [Huber and Daft 1987; Morton 1992]. Organizations are forced to make more decisions, make them faster [Huber 1984], and make them in an increasing number of exceptional situations for which a routine response does not exist [Lundquist and Huston 1990]. At the same time, organizational decisions are politically and technically getting more and more complex [Huber and McDaniel 1986]. In fact the very nature of organizational work is changing [Drucker 1991]. The manufacturing industry is no longer dominant. Studies show that in 1990 approximately 75 percent of all jobs in the USA were held in the service industry [Morton 1992; Roach 1991].

Organizations can be seen as purposeful, even purposive systems [Checkland 1981a; Sol 1982]. An organization is a *system*, because it may be considered to be a structured set of interrelated actors or activities that perform certain organizational functions [Leeuw 1990; Lundquist and Huston 1990]. It is a *purposeful* or *purposive* system because the organization functions to fulfill the demands of its environment [Ackoff and Emery 1972; Leeuw 1990].

The dynamic character of organizations' environments, brings about constant adjustments in the demands that are imposed on them. In other words, the purpose of an organization and the function it performs, is prone to change. According to Ashby's Law of Requisite Variety, organizations in a changing environment need a variety of responses which can match a variety of environmental demands [Ashby 1956]. However, Ashby states that organizational change does not only follow from changing environments. Organizations also strive to anticipate changes in their environment.

The responses that organizations demonstrate with respect to new and expected demands from their environment involve a constant adjustment of their structures, processes, and technologies, see e.g. [Burns and Stalker 1961; Ghoshal and Bartlett 1995; Lawrence and Lorsch 1967; Morton 1992]. Fueled by decreasing costs and increasing capabilities [Benjamin 1982; McKenney and McFarlan 1982], applying Information Technology (IT) has become an especially popular response to changing demands on organizations. However, this response is not always a successful one. On the contrary, new IT does not always have the intended impact after introduction, see e.g. [Angell and Smithson 1991; Bakos and Brynjolfsson 1993; Davenport 1994; DeSanctis and Poole 1994; Land 1990; Lucas and Olson 1994]. It has not yet been possible to correlate any concrete productivity gains with the application of IT, see e.g. [Baily and Chakrabarti 1988; Loveman 1988; Roach 1991].

There are indications that the absence of significant productivity gains is a result of a customary inclination toward applying Information Technology (IT) to *automate or support their current way of working*, see e.g. [Chen and Nunamaker 1989; Davenport and Short 1990; Drucker 1988; Hammer 1990; Robey 1983; Trauth et al. 1993]. Hammer [1990] claims that more substantial gains and improvements can be achieved by applying IT as support for a *changed or reorganized way of working*. Evidence which supports Hammer's assertion is given by [Davenport and Short 1990; Davis and Botkin 1994; Drucker 1988; Dur 1992; Hammer 1990; Hammer and Champy 1993; Meel 1994; Sol 1992b].

Reorganizing, or 'organizing' in the sense of structuring, includes the division of organizational processes into tasks, the allocation of resources needed to perform tasks, the definition of areas of responsibility, and the distribution of tasks and responsibilities among one or more individuals or groups [Hall 1972; March and Simon

1958]. In order to guide and monitor organizational processes as a whole, their constituent tasks, often carried out in parallel, have to be *coordinated* [Mintzberg 1979]. Hence, a need to reorganize the current way of working implies a need to reorganize the way in which an organization achieves the *coordination* of its processes and constituent tasks [Child 1984; Mintzberg 1979].

Bots and Sol differentiate between three perspectives with respect to reorganizing organizational processes and their coordination [Bots and Sol 1988; Sol 1990, 1992a]:

- The *micro* perspective, focusing on the primary tasks of the individual work places in the organization and the way in which these tasks are supported.
- The *meso* perspective, focusing on the work places in an organization that cooperate in order to perform an organizational process.
- The *macro* perspective, focusing on the interorganizational relationships and cooperation in a 'chain' or network of organizations.

In remaining consistent with the above perspectives, we discriminate between three different forms of coordination: interorganizational coordination (macro), organizational coordination (meso), and work place coordination (micro). *Interorganizational coordination* deals with seeking partnerships [Bakos and Brynjolfsson 1993; Drucker 1995], and establishing interorganizational relationships, sometimes called 'policy issue networks' [Kirst et al. 1984], that are profitable for all parties involved [Alexander 1993; Clemons 1992; Malone and Rockart 1991; Rogers and Whetten 1982; Sol 1992a; Womack and Jones 1994]. It focuses on attuning the linking processes in the interorganizational chain where the output (products or information) of one organization represents the input for one or more other organizations [Alexander 1993; Eijck et al. 1993; Streng 1993; Wierda 1991; Womack and Jones 1994]. The notion of interorganizational coordination is closely related to interunit coordination in geographically dispersed organizations [Ghoshal and Bartlett 1990; Ghoshal et al. 1994]. For a more elaborate discussion of interorganizational coordination and coordination in geographically dispersed organizations, we refer to [Eijck 1996; Straatman 1995].

Organizational coordination deals with orchestrating the joint performance of workers and the constituent tasks within the organization's processes that they carry out, see e.g. [Dur 1992; Ghoshal and Bartlett 1995; Meel 1994]. *Work place coordination* deals with planning and attuning the tasks that are carried out in a single work place. It focuses on individual workers putting tasks on their agenda and managing this agenda. Work place coordination has been discussed in more detail in [Bots 1989].

In summary, we see that modern organizations are forced to reconsider the coordination of their processes and constituent tasks on an ongoing basis. In the next

section, we elaborate on organizational coordination in greater detail. The ensuing section discusses IT and how it can be applied to support organizational coordination.

1.2 Organizational coordination

The subject of organizational coordination is extensively discussed in the literature on organizational theory. In this section we first describe organizational coordination, then we explore a number of ways or mechanisms with which tasks in organizations can be coordinated and appropriate coordination mechanisms chosen. Finally, we examine the relationship between organizational coordination and communication. Note that some of the concepts and coordination mechanisms discussed in this section, may also be applicable for interorganizational coordination and work place coordination.

1.2.1 What is organizational coordination?

According to the Oxford Dictionary [1982], coordination means '*to bring (parts, movements, etc.) into proper relation, cause to function together or in proper order.*' In the literature on organization theory a vast range of descriptions and definitions for coordination can be found from several perspectives and on different levels of abstraction.

Generally, coordination is often linked to the notion of 'organized behavior' by organizational workers. Mintzberg phrases this as follows [1979, p. 2]: '*Every organized human activity ... gives rise to two fundamental and opposing requirements: the division of labor into various tasks to be performed and the coordination of these tasks to accomplish the activity.*' This view is omnipresent, see e.g. [Gulick 1937; Lawrence and Lorsch 1967; March and Simon 1958; Mintzberg 1983; Mooney 1937; Urwick 1937]. Galbraith [1974, 1977] also subscribes to this view. He makes an explicit link between organizational coordination and organizational goals in the context of organization design. He describes organization design as the search for coherence between the goals or purposes for which an organization exists, the people that do the work, and the patterns of division of labor and interunit coordination.

From a decision making perspective, Simon relates coordination to the *decision making* behavior of individuals within an organization [Simon 1947]. According to Simon (a group of) individuals need(s) to be coordinated because they are not capable of applying the right strategies for their actions and decisions, due to a lack of insight about the strategies of other individuals. Therefore, he defines coordination as providing everybody with insight into everybody else's decision making behavior, so that every individual is capable of making the right decisions.

The concept of coordinating behavior is sometimes seen as an issue of *control* [Kickert 1979; Parker Follett 1937]. Kickert defines coordination as control over a system (organization), consisting of subsystems (parts of the organization, departments, or employees) between which relationships exist. Like Bosman and Sol [1984] and in line with Simon, he states that the ultimate goal of this control is that the system will display the behavior that is desired by the controller.

In the literature exhibiting a systems science perspective on organization theory, coordination is often discussed in terms of *interdependencies* between organizational activities. Malone and Crowston [1994 p. 90] define coordination as managing dependencies between activities. They mention *actors* and *interdependent activities* as key concepts. Every organizational activity can be considered to be a task that requires an actor or 'processor' (man or machine) and a certain amount of time to be carried out [Malone 1987]. Coordination is needed because tasks usually have to be assigned to a limited set of processors that have a limited capacity available.

Well known is Thompson's [1967] discussion of the concept of interdependency describing organizational coordination. Thompson classifies organizational activities in terms of their dependency to other activities. Independent activities can be performed without interacting with other co-workers. Activities that require coordination between the people performing them, are considered to be interdependent. According to Thompson, there are three types of interdependencies:

1. *Pooled interdependency* occurs when each person gives a discrete contribution to the overall outcome. Pooled interdependent activities can be performed in parallel.
2. Activities are *sequentially interdependent* when the completion of one activity is imperative for performing the next.
3. *Reciprocal interdependency* occurs when a group of people work together and the outcome of each group member's activity continuously renders input for the other group members. This is, for example, the case in a group discussion.

Malone and Crowston [1994] formulate a different and slightly broader taxonomy than Thompson on interdependencies between organizational activities. According to them there are four types of interdependency. Note that in their taxonomy Thompson's pooled interdependency is not uniquely represented by either one of the four types:

1. *Shared resources*. This interdependency occurs when a number of activities share a limited resource, such as time, money or capacity.
2. *Producer/consumer relationships*. In this type of interdependency the result or output of one activity is used by another activity. This is similar to Thompson's sequential interdependency.

3. *Simultaneity constraints* apply to activities that have to be performed at the same time, such as meetings. This is comparable to Thompson's reciprocal interdependency.
4. *Task/subtask relationships* describe the interdependency between a group of 'sub-activities' that together have to contribute for the main activity to achieve its goal.

Apart from the organization theory perspective, there are other perspectives according to which we may consider organizational coordination as well. As Malone and Crowston [1994, p. 90] state: '*... coordination ... can occur in many kinds of systems: human, computational, biological and others*'. This implies that other disciplines may also offer valuable insights into organizational coordination. In the remainder of this section we shortly discuss three of these disciplines: economics, computer science, and sociology.

From an *economical perspective* coordination is usually discussed in the context of multiple organizations operating on a market, see e.g. [Douma and Schreuder 1991]. However, as some authors show, economic concepts can also be applied to study organizational coordination [Ching et al. 1992; Ledyard 1993; Malone 1987; Malone and Smith 1988]. Coordination is often seen as a process of allocating scarce resources among actors [Douma and Schreuder 1991; Malone and Crowston 1994]. Special attention is paid to how incentives and information flows affect the allocation of resources. Well known theories for studying coordination from an economical perspective include transaction cost theory, see e.g. [Williamson 1975], and agency theory, see e.g. [Arrow 1985; Ross 1973].

The *computer science perspective* on coordination appears to demonstrate some similarities with the economical perspective. In this field much attention is paid to coordination as an allocation problem of shared hardware and software resources, such as processors, memory, peripherals, and databases [Lister 1984; Malone and Crowston 1994; Tanenbaum 1984]. Holt [1988], for example, studies coordination in the context of cooperative work on heterogeneous computer networks. He sees coordination as a '*dynamic glue that binds tasks together into larger meaningful wholes*' [1988, p. 109].

From a *social science perspective* Thompson et al. [1991] discuss the coordination of social life. The term 'social life' embodies various arrangements of actors, e.g. in organizations, in interorganizational chains, or in international relationships between states. Thompson et al. consider social coordination as the bringing into relationship otherwise disparate activities. They describe the goal of coordination as to attune the potentially different and conflicting objectives of the actors performing activities in order to avoid possible chaos and inefficiency.

Although the different perspectives we discussed above highlight different aspects with respect to organizational coordination, we observe some commonalities. It appears to be relatively non-controversial that coordination at least involves *actors* that have to perform a number of distinct organizational *activities* in concert. We adopt a systems science perspective on organizations and organizational coordination as this perspective is *neutral* and *descriptive* [Leeuw 1990]. It represents a set of concepts and methods that are not theory loaded, but allow for the construction of theories and models of organizations. With this in mind, we now define organizational coordination as follows:

Organizational coordination is the way in which interdependent organizational activities of two or more actors mutually influence each other in order to attain a certain objective.

This definition is in line with the definition of Malone and Crowston [1994], but as we see organizations as purposeful systems, we refer to mutual influencing with a certain objective instead of just managing. In our definition, three concepts are important:

- *Multiple actors*, where actors can be seen as individuals, as groups or as organizational units.
- *Mutual influence*. We use 'mutual influence' instead of 'mutual adjustment' because the latter term is often used to describe one specific coordination mechanism, see [Mintzberg 1979].
- *Attaining an objective*. This is of importance because coordination problems are often caused by actors attaining different objectives or goals.

Having defined organizational coordination and having discussed different perspectives on it, it is time to focus on ways in which coordination can be achieved: coordination mechanisms.

1.2.2 Mechanisms for organizational coordination

Both in the literature and in practice one can find a myriad of coordination mechanisms. Scientists and practitioners have often attempted to map organizational situations with the most appropriate (combination of) coordination mechanisms for these situations. However, in reality every organization's situation is unique and requires a blend of appropriate coordination mechanisms [Leeuw 1990]. In this section we provide a structured overview of a number of the most common organizational coordination mechanisms. We propose a taxonomy of coordination mechanisms, which differentiates between mechanisms that focus on *organizational structures*, and mechanisms that focus on *coordination processes* as a means of achieving coordinated

organizational behavior. An overview of this taxonomy of coordination mechanisms is given in table 1.1.

Perspective	Mechanisms for organizational coordination
Structure as a whole	Markets (decentralized/centralized) Hierarchies (product/functional) Networks
Structure as interdependence between activities	Standardization (processes/outputs/skills/norms) Mutual adjustment Direct supervision
Coordination processes	Anticipatory coordination Adaptive coordination Review

table 1.1 Taxonomy of coordination mechanisms.

Many organizational theorists see the structure of an organization as the primary mechanism for coordinating organizational activities [Jablin 1987]. We look at organizational structure as a coordination mechanism from two levels of abstraction: the structure of the organization as a whole, and the structure of the interdependencies between individual organizational activities.

Structure as a whole

In the context of organizational coordination, the structure of the *organization as a whole* aims to dictate the desired behavior of the organization. There are many different types of organizational structures that are employed to this end, see e.g. [Leeuw 1990] for an overview. These structures can roughly be categorized in three basic types: markets, hierarchies, and networks [Malone et al. 1987; Thompson et al. 1991].

The concept of a *market* as a coordination mechanism reflects an economics perspective, where it describes coordination *among* organizations, and not within organizations. However, a market structure can also be found within organizations [Ching et al. 1992; Leonard-Barton et al. 1994; Malone 1987]. The market as an organizational structure coordinates more or less automatically and unseen via bidding and pricing systems [Thompson et al. 1991]. Markets can be classified into decentralized markets and centralized markets [Malone 1987; Malone and Smith 1988]: In a *decentralized market* all buyers are able to contact all suppliers to negotiate transactions, whereas in a *centralized market* transactions between buyers and suppliers are negotiated indirectly through a 'broker'.

In contrast to the market mechanism, the *hierarchy* as organizational structure coordinates very consciously. In a hierarchical organization there are objective qualifications of office, fixed rules of behavior, and clear authority relationships

[Thompson et al. 1991]. The hierarchy concept is often identified with Weber's bureaucracy concept. There are many variations of the hierarchical structure [Leeuw 1990]. The two most common variations are the product hierarchy and the functional hierarchy [Malone 1987; Malone and Smith 1988]: In a *product hierarchy* there is a separate division for each type of good or service, while every division has its own production unit, marketing unit, maintenance unit etc. In a *functional hierarchy* resources are pooled in separate functional units and are shared among the products.

A more recent type of organizational structure is the *network* form. Networks focus on the informal coordinating mechanisms within a group of (relatively independent) elements [Thompson et al. 1991]. Network organizations are made up largely of temporary teams materializing informal relationships across functional departments [Drucker 1988; Qureshi 1995]. A network organization empowers teams with members from different functional departments to form a '*small company inside the large company*' [Charan 1991, p. 105]. Other organizational structures closely related to the network form include the matrix organization, project organization, and the adhocracy [Leeuw 1990; Mintzberg 1979; Toffler 1970].

Structure as interdependence between activities

As may be apparent from the above discussion, organizational structures only describe how the whole organization is supposed to operate. However, it does not deal with the way in which the actual coordination of interdependent activities takes place [Ching et al. 1992]. Interdependencies between single activities or groups of activities can be coordinated by a variety of mechanisms, of which the most predominant ones are standardization, direct supervision, and mutual adjustment [March and Simon 1958; Galbraith 1974; Mintzberg 1979].

Coordination by *standardization* involves defining rules for the performance of each activity in order to ensure that the result of a certain activity renders the correct input for a next activity. There are four types of standardization [Mintzberg 1979; Quinn et al. 1988]: standardization of work processes, standardization of work outputs, standardization of worker skills, and standardization of norms. Coordination by *mutual adjustment* involves actors making on-going adjustments with respect to the performance of their activities and those of other actors. Coordination by *direct supervision* involves one actor responsible for managing a number of interdependent activities. This supervising actor also has the power to adjust the performance of these activities in order to attune them.

Coordination processes

The basic assumption of achieving organizational coordination by designing structure, both in the organization and for the individual activities, is that the people in the organization display the prescribed behavior intended by its structure. Though this may

be the case in many situations, *formal* structures and coordination mechanisms are sometimes overruled by *informal* ones. Especially the Human Relations School in organization theory, which focuses on the functioning of individuals and groups in organizations, has shown that there are other, less tangible, coordination mechanisms. Examples include leadership and power [Likert 1961], self-fulfillment [McGregor 1960], and individual needs and goals [Argyris 1973].

It may therefore in some cases be more appropriate to look at the actual *processes* of coordination instead of the defined *structures* for coordination. Taking a process view on coordination, March and Simon [1958] show that there are two different types of coordination processes to achieve organized behavior. These two types can be distinguished on a temporal basis:

1. *Coordination by plan or Anticipatory coordination.* This involves the coordination of activities that have yet to be executed through planning.
2. *Coordination by feedback or Adaptive coordination.* This involves coordination during the actual execution of organizational activities through monitoring, feedback, and control.

The same distinction is put forward by Kieser and Kubicek [1977] in terms of future coordination and feedback coordination. In addition to these two types, Ching et al. [1992] define a third type of coordination process:

3. *Review coordination.* This involves performance evaluation for the benefit of organizational learning and for coordinating future activities.

From a discussion of the coordination mechanisms described above, it appears that choosing the most appropriate combination of coordination mechanisms for particular situations is not always straightforward. In the next section we discuss this issue in more detail.

1.2.3 Choosing appropriate coordination mechanisms

In organizational theory, the contingency school has attempted to identify a number of organizational variables to characterize organizations. For each type of organization, the most appropriate coordination mechanism needs to be determined. Thompson [1967], for example, claims that pooled interdependencies have to be coordinated by rules and standards, sequential interdependencies by planning, and reciprocal interdependencies by mutual adjustment. Others identify a correlation between task uncertainty and the most preferred coordination mechanism [March and Simon 1958; Randolph 1978; Van de Ven et al. 1976]: Organizational workers seem to prefer direct personal contact (mutual adjustment, direct supervision) for non-routine tasks,

involving uncertain and equivocal task-related information. They prefer written rules and procedures (standardization) for routine tasks involving objective and certain task-related information. However, it has also been pointed out that the relationships between contingencies are more complicated than some researchers suggest [Bird Schoonhoven 1981; Mintzberg 1979]. This is illustrated by the fact that different researchers have presented conflicting results [Mintzberg 1979]. In addition, as the number of seemingly significant contingencies increases, the number of possible contingency combinations for which appropriate coordination mechanisms have to be determined, increases exponentially.

The process of selecting appropriate coordination mechanisms is further complicated by the instable circumstances which organizations operate in. As we explained in section 1.1, organizations are forced to adapt their structures and processes constantly. Mintzberg argues that when the complexity of the activities that have to be coordinated increases, the favored means of coordination appears to shift from mutual adjustment to direct supervision to standardization, preferably of work processes, otherwise of outputs, or else of skills, finally reverting back to mutual adjustment [1979, p. 7].

The object of improving organizational coordination is to achieve better coordinated processes. But what is meant by *better or improved coordination*? Certainly coordination is not a goal in itself [Holt 1988]. Coordination is a requirement for carrying out activities to enable a greater goal to be achieved. Some coordination mechanisms may be more expensive in terms of resources, but may bring about a significant improvement of the extent to which the goals of the coordinated activities are achieved. It follows that the extent to which the coordination of an organization has improved should be measured in terms of *indicators* (or 'evaluation criteria' [Malone and Crowston 1994]) that are derived from the goals of the organization. Examples of such indicators are delivery time, service level or overall throughput time. When changing an organization's coordination mechanism, these indicators show changes in the extent to which the organization is 'better' coordinated. As Malone and Crowston phrase it [1990, p. 359]: *'When we analyze coordination...we must (at least implicitly) evaluate the actors' collective behavior in terms of how well it achieves some overall goals.'*

In summary, the selection of coordination mechanisms seems to be affected by three factors. First, there is no standard optimal coordination mechanism for certain organizational structures. Second, the selection of coordination mechanisms is a continuous problem as organizations have to adapt constantly. Finally, the appropriateness of a coordination mechanism can only be determined indirectly by looking at indicators.

1.2.4 Coordination and communication

In the last part of this section, we look at the close relationship between organizational coordination and communication. March and Simon [1958] state that the greater the efficiency of communication, the greater the tolerance for interdependency. As Malone and Crowston point out [1994, p. 99]: *'One obvious way of generating new coordination processes is by considering alternative forms of communication.'* Without communication, organizational workers cannot influence the behavior of other workers and therefore cannot coordinate their activities [Malone and Rockart 1991; Simon 1947]. In order to synchronize activities and processes, organizational workers may have to inform each other *explicitly* about the starting and ending times of their respective activities, or about the status of these activities. They may have to issue instructions to other colleagues to have activities completed. Coordinating information between two organizational workers can also be *implicitly* communicated. If the result of an activity of a worker is conveyed to a colleague, this colleague may know what to do next by the nature and appearance of the item he receives. The above can be summarized with the words of Lundquist and Huston, [1990, p. 80]: *'Organised workers must both manage their own actions ... and manage the actions of others through communicating informing information.'*

The notion of organizational communication is of great importance to the study of organizational coordination. As we discussed in the previous section, the formal structure of an organization can be considered a means to achieving coordinated behavior in organizations. However, often the formal structure of an organization does not fully explain the way in which organizational workers coordinate their activities in reality [March and Simon 1958]. Formal organizational structure merely dictates the *constraints* within which organizational members can communicate and hence coordinate their activities [Jablin 1987]. It is only by looking at the communication processes itself that the actual patterns and mechanisms of coordination, both formal and informal, within organizations can be identified [Monge and Eisenberg 1987; Qureshi 1995].

In essence, communication implies exchanging information. The growing possibilities of information technology to support different forms of information exchange make it an important means of supporting organizational coordination. This is elaborated in the next section.

1.3 Information technology to support organizational coordination

Fueled by a combination of decreasing costs and increasing capabilities, information technology (IT) has now penetrated almost every work place in the post industrial

organization [Cash and Konsynski 1985; Holt 1991]. Organizations have long focused on supporting the individual needs of their workers by offering stand alone support such as word processors, spread sheets, and database applications. However, with the advent of cost-effective communication technologies, individual workplaces are becoming increasingly interconnected. As a consequence, the use of IT is now increasingly shifting from supporting pure computational activities towards supporting coordination activities as well [Malone and Rockart 1991]. Apart from processing information, IT is increasingly being employed to mediate interpersonal and group communication. The nature and appearance of IT is changing accordingly. As Sol and Bots observe: *'Inevitably, information systems will take the shape of networks of workplaces that are designed with the single purpose of making information workers more effective in performing their problem solving and decision making tasks'* [1992, p. 27].

In this section we discuss IT that is aimed at supporting organizational coordination. In the process of organizational change, IT for coordination support may perform two roles: It may be a required technology for organizational changes to take effect (organizational pull), or it may be an enabling technology to bring about organizational change (IT push). The next subsection elaborates on these two roles and describes examples of IT to support organizational coordination. The ensuing two subsections deal with the effects of IT on various organizational characteristics and with the duality between organizational structure and IT. We conclude with an assessment of what is currently known about IT for coordination support and which questions remain yet to be answered.

1.3.1 *The double role of IT*

New forms of IT for coordination support offer more than just the ability to connect people and send, receive, store, and retrieve information. This type of IT overcomes traditional constraints such as time, space, and hierarchy and is seen to thereby *enable* organizations to explore new possibilities for adaptation [Lewin and Stephens 1993]. It offers the ability to initiate the restructuring of organizational processes and the coordination of these processes; it may change the patterns of organizational communication and coordination, whether anticipated or not [Culnan and Markus 1987; Flores et al. 1988; Lewin and Stephens 1993]. On a different level of abstraction, IT can also be used to enable actual design activities [Davenport and Short 1990; Dennis et al. 1994; Kyng 1991; Meel 1994; Vogel et al. 1993; Vreede et al. 1995e]. Examples of such IT employment include simulation environments, case tools, and application generators for prototypes.

Although IT is often seen as an enabling technology, it can also be considered as a *requirement* for effective organizational coordination. As organizational coordination

involves communication and hence information exchange, one resulting problem associated with organizational change is that a requirement is created for changing an organization's information system in order to effectively support the modified organization. A number of cases in which IT is not the cause of organizational change, but merely the *sine qua non* for restructuring an organization to be successful, are described by [Crowston et al. 1987; Robey 1981 and 1983]. In addition, modern organizations recognize that well coordinated interdependencies between their actors are crucial, that organizational work is increasingly becoming a group effort, and that flexibility and adaptiveness are a necessary condition upon which internal and external demands are met [Huber 1984; Huber and McDaniel 1986; Malone and Crowston 1994]. To deal with these challenges, organizations require IT that is capable of disseminating information faster, more cheaply, and more selectively [Malone and Crowston 1994; Simon 1973].

Whether as an enabling or as a required technology, modern organizations exhibit various types of new IT to support organizational coordination and communication. These types of IT are often labeled as *groupware*, indicating information technologies that mediate electronic interpersonal collaboration [Coleman 1994; Ellis et al. 1991; Johansen 1988; Oppen and Fersko-Weiss 1992]. There are a myriad of different groupware applications, see e.g. [Coleman and Huckle 1994], most of which can be divided into the following four categories:

1. *E-mail facilities*. In recent years, this has become one of the most ubiquitous technologies used to support asynchronous communication [Keen 1994; Sproull and Kiesler 1986]. In its simplest form, email facilities support one-to-one message exchange. Other types of email facilities, such as listservers and newsgroups, specifically support asynchronous group conferencing activities.
2. *Documentary Information Systems (DocIS)*, or *Information Retrieval Systems (IRS)*, support the indexing, imaging, retrieval, and exchange of (multimedia) documents [Hooegeveen 1994; Hooegeveen and Meer 1994; Meer 1994]. Examples of DocIS include library catalogue systems, full text retrieval systems, and the hypertext based World Wide Web (WWW).
3. *Workflow systems* support the execution and management of administrative processes [Harst 1994]. A workflow system usually comprises of facilities to formally define administrative processes and to automatically route forms and documents in the form of email messages between the actors involved in these processes. Workflow systems predominantly support Type I office activities [Panko 1984], that are pooled or sequentially interdependent [Vreede 1994].
4. *(Distributed) Group Support Systems ((D)GSS)* support meeting activities of dispersed or face-to-face groups, see e.g. [Bostrom et al. 1992a; Jessup and Valacich 1993]. GSS support Type II group activities [Panko 1984]: creative, intellectual activities with a unique character. These activities are in essence

reciprocally interdependent [Vreede 1994; Watson et al. 1994]. GSS are discussed in more detail in Chapter 6.

Limited media richness is often argued to be the main weakness of these types of groupware. Electronic communication is considered to represent a poorer medium in terms of information interpretation than, for example, face-to-face communication [Flores et al. 1988; Daft and Lengel 1986]. However, not every group task requires the richest possible medium [Daft and Lengel 1986; McGrath and Hollingshead 1993]. In addition, electronic communication processes should not just be assessed by comparing them to the characteristics of other, more ordinary types of communication. Unique capabilities of electronic communication support should also be taken into account, such as new ways to address communication, new modes of communication storage and retrieval, and enhanced capabilities to control access to and participation in communication [Culnan and Markus 1987].

1.3.2 *Effects of IT*

From a cost perspective, Malone and colleagues argue that the increasing application of IT will significantly reduce the costs of coordinating interdependent activities [Malone 1992; Malone and Crowston 1994; Malone and Rockart 1991]. The expected effect of this cost reduction is an increase in the overall amount of coordination within and between organizations, and a greater tendency towards more coordination intensive organizational structures, such as the adhocracy, the network structure, decentralized markets, and matrix organizations.

Apart from cost effectiveness, it is often claimed that IT supporting coordination and communication will improve the productivity of groups. There are unmistakable indications that contribute to this view. First, use of time studies show that managers and professionals respectively spend about 85 percent and 60 percent of their time on communication related activities, see e.g. [Mintzberg 1973; Poppel 1982; Panko 1992]. Second, in the service industry, representing 75 percent of all US jobs [Roach 1991], products have a high information content [Dur 1992]. These products are rarely made by individuals. Hence, adequate support for communicating content related information and coordination information is *likely* to have a significant impact in these environments.

In 1958, Leavitt and Whisler predicted that the introduction of IT would reduce the number of middle managers, thus flattening the organization. Since then, many researchers have investigated hypothetical effects of IT on organizational variables such as size, centrality of decision making, and span of control. However, these studies did not find conclusive and unambiguous evidence of the exact nature of these effects [Attewell and Rule 1984]. Several studies have shown IT either to increase or decrease

an organization's size, and to stimulate both centralization and decentralization of decision making, see e.g. [Bakos and Brynjolfsson 1993; Crowston et al. 1987; George and King 1991; Gurbaxani and Shi 1992; Gurbaxani and Whang 1991; Malone et al. 1987]. Furthermore, IT has been shown to reinforce existing decision and authority structures [George and King 1991; Robey 1983].

Apart from organizational variables, IT effects are sometimes described in terms of perceived benefits, but these benefits are very difficult to determine accurately [Morton 1992]. First, benefits of IT investment often end up in the consumers' hands. Second, IT is often used to respond to certain new demands, such as government regulations, that do not directly influence economic returns. Benefits of IT are therefore often sought in the enhanced *quality* and *availability* of information needed to make (coordinating) decisions. A study by Foster and Flynn [1984] shows that the introduction of IT for communication support seems to increase the level of organizational communication and hence the amount and availability of information. It is a common misconception, however, to assume that *more* information will yield better decision making, better decision quality, and improved coordination abilities, as has been pointed out by [Jong 1992; Reuyl 1982; Simon 1973; Sol 1982].

Finally, IT for coordination support does not always render the effect that was anticipated. As (adaptive) structuration theorists point out, organizations may appropriate new technologies in ways that were originally not intended [Orlikowski 1992; DeSanctis and Poole 1994]. The introduction of IT to support organizational coordination may indeed sometimes have unwelcome effects. Lucas and Olson [1994] show how the implementation of IT may actually create an unexpected *greater* need to coordinate. First, IT to support coordination allows people to work more at home, making them more difficult to manage, control, and coordinate. Second, increasing the pace with which people perform their work may result in greater stress on the coordinators of these people. On the same line of argument, Kiesler observes that implementing IT for communication support may not exactly facilitate coordination: '*If you just add technology to the office, you may wind up having more communications to monitor, more things to type, and more projects initiated that don't get completed; you may not improve performance,*' [1986, p. 48].

1.3.3 A reciprocal interdependency

Above we discussed the effects of IT on an organization. This is but one side of the coin. Many authors who have studied the relationship of IT and the organization, have observed a dual effect. This dual effect can be described as a *reciprocal interdependency* between organizational structure and information technology. Transformations in either one element incur transformations in the other.

Leavitt [1965] was one of the first to describe the reciprocal relationship between organization and (information) technology. He sees organizations as consisting of four components: People, structure, task, and technology, which are strongly interrelated and mutually adjusting. When technology is changed, the other components subsequently adjust themselves or need adjusting. To put it more strongly: IT will not impact an organization's performance, unless corresponding investments are made in the people, tasks, and structure of the organization [Morton 1992].

Taking Leavitt's work and similar work of others as a starting point, Orlikowski [1992] uses structuration theory to describe a new model that tries to explain the equivocal results of empirical studies into relationships between these organizational components. Orlikowski's model postulates continuous and simultaneous influences between people in organizations, the (information) technology in organizations, and organizations' institutional properties, such as strategies, structures, processes, culture, and environmental pressures. These elements influence each other as follows:

- IT imposes formal and informal structures on the people in the organization.
- People determine the actual, possibly unintended, use and appearance of IT, by appropriating and modifying it.
- The organization's institutional properties influence how people interact with each other and with IT.
- IT influences the institutional properties, transforming or reinforcing them.

In short, the exact relationship between organizations and IT supporting organizational coordination greatly depends on the particular organizational context: it depends on institutional properties and on the people using the technology [Orlikowski 1992; Orlikowski and Robey 1991]. Therefore, exact rules to predict IT impact on organizational coordination will be difficult if not impossible to conceive.

As a consequence, in supporting organizational coordination through IT the real issue is no longer designing IT and predicting its impact. The focus should be on designing the *destined synergy* between IT and organizational coordination such that it will be most effective [Sol 1992a]. If organizational workers are supported by networked information systems, the interaction between the organization's structure, the coordination of its processes, and its supporting technology becomes so intense that joint design is not only necessary but also inevitable [Chen and Nunamaker 1989; Child 1977; Holt 1991; Huber and McDaniel 1986; Meel 1994; Specht and Robey 1988]. Such an integral design effort will focus on '*prototypes*', both organizational and technological, and on '*participation*' [Sol 1982, 1992a; Tyre and Orlikowski 1994].

1.3.4 Concluding remarks

In this section we discussed the nature and impact of IT to support organizational coordination. Empirical studies into the impact of IT on organizations and organizational coordination have found equivocal and sometimes conflicting results. Some authors explain this by stating that most of these empirical studies are oversimplified and too fragmented, see e.g. [Crowston et al. 1987; Orlikowski 1992; Robey 1983]. There is, however, consensus on two issues. First, IT does offer opportunities for supporting coordination and communication within organizations. The question is not whether or not to employ IT, but how to design and implement IT most effectively. Second, designing IT and organizational coordination cannot be addressed separately. In order to handle their intrinsic interdependency they have to be designed in an integral way. Given these opportunities and the need for integral design, we are faced with a number of problems:

1. There is scant literature on designing IT for coordination support, or groupware. Much research in this area is descriptive, is performed in artificial settings, is performed at one point in time, and ignores some of the new capabilities of this type of IT [Culnan and Markus 1987].
2. Most IT design literature does not address the organizational context of IT, see e.g. [Holt 1991; Huber and McDaniel 1986].
3. Much of the organizational design literature does not address IT to support organizational coordination, see e.g. [Meel 1994].

1.4 Research question and research objective

In this chapter we have described the problem area of our dissertation. Modern organizations are on the one hand forced to reconsider the coordination of their processes and constituent tasks. On the other hand, they are offered new forms of IT to support current and new coordination mechanisms. Apart from the problems with respect to integral design that we described in the previous section, there are various reasons why organizations have great difficulty in adequately dealing with these simultaneous challenges:

1. Except for social-technical systems design theory [Eijnatten 1993], organization design theory merely presents design guidelines on a high level of abstraction, which cannot easily be translated into appropriate change programmes for organizational practice.
2. Notwithstanding the increasing attention for improving organizational coordination, this attention is mostly of a descriptive nature. Few field studies have been reported on the improvement of existing coordination processes. Exceptions include

[Eijck 1996; Gurbaxani and Shi 1992; Kling 1992; Malone et al. 1987b and 1987c; Meel 1994].

3. Efforts on integral design of organizational processes and coordinating IT are expected to focus on 'participation'. However, there have been few substantial inquiries into the possibilities of bringing about organizational change in a participative fashion. Exceptions include [Eijnatten 1993; Mumford 1983; Vogel and George 1992].

In light of the above, the need for methods and techniques to help organizations improve their coordination and adapt their information technologies accordingly, is well supported. Therefore, we formulate the following research question:

RESEARCH QUESTION *How can we support organizations in the process of improving their coordination and their supporting information technology?*

To answer our research question, we pursue the following research objective:

RESEARCH OBJECTIVE *Formulate a design approach for facilitating the process of improving an organization's coordination and its supporting information technology.*

Such a design approach can be seen as an inquiry system as defined by Sol: 'A structured set of instruments which can be used as a context in the problem-solving activities. It expresses a methodology in view of a problem area' [1982, p. 13]. Hence, an inquiry system can be thought of as a set of instruments embedded in a way of thinking, a way of working, a way of modelling, and a way of controlling [Sol 1990]. In this sense, our design approach should:

- Make a worldview explicit with respect to organizations and organizational coordination.
- Outline the steps, or working method, necessary to analyze, diagnose and change organizational situations in which coordination problems occur.
- Be capable of modelling organizational activities and their coordination in current and future situations.
- Offer guidelines and techniques to manage and control the application of the modelling techniques according to the working method.

The research approach we employed to answer the research question and design the required set of instruments is described in Chapter 2.

2 Research approach

'If you want to swim, you must enter the water.'

- Frank Herbert

2.1 Introduction

The research approach of a scientific inquiry may be defined as following a certain research *strategy* in which a set of research *instruments* are employed in order to collect and analyze data on the phenomenon studied, guided by a certain research *philosophy*. In the following sections we discuss the research philosophy, strategy, and instruments which we apply to address our research question and pursue our research objective as described in the previous chapter. This chapter concludes by outlining the research to be pursued in this dissertation.

2.2 Research philosophy

A research philosophy underlines the way in which the data on the phenomenon studied is collected and analyzed. In general terms, a research philosophy is seen to determine the epistemology guiding a study. It determines what kind of knowledge can be obtained and what are the limits to that knowledge.

In studying organizational phenomena, including the application of information technology within organizations, we can distinguish between two major research philosophies, or 'schools of thought' [Alavi and Carsol 1992; Burrell and Morgan 1979; Galliers 1991; Orlikowski and Baroudi 1991; Pervan 1994]:

- *Positivism*. Positivists claim that reality can be observed and described objectively without interfering with the phenomenon being studied. They focus on isolating phenomena (reductionism) and on the repeatability of observations. Positivism

seeks to explain what happens by searching for regularities and causal relationships between the constituent elements of reality.

- *Interpretivism.* Interpretivists, sometimes called anti-positivists, claim that reality can only be understood by subjectively interpreting observations of reality. They focus on studying phenomena in their natural environment, and maintain that scientists have an impact on the phenomena they study. While pointing out that multiple interpretations of reality are possible, they emphasize human interpretation and understanding as constituents of scientific knowledge.

Both schools form the basis of many studies in organization research and information system research. Although in *organization research*, the bulk of early studies was of a descriptive, interpretive nature, positivism has since long been the dominant philosophy [Burrell and Morgan 1979; March and Simon 1993; Mintzberg 1979; Susman and Evered 1978]. *Information system research* has shown a similar shift in emphasis. In the early years (until 1978) descriptive, interpretivist studies were predominant. The last two decades have been characterized with a strong focus on positivist research [Dickson 1981; Dickson and DeSanctis 1990]. According to Orlikowski and Baroudi [1991], 96.8 percent of research in the leading American information system journals follows the positivist tradition. Other sources confirm this bias, see e.g. [Alavi and Carlson 1992; Galliers and Land 1987; Kaplan and Duchon 1988].

As we will demonstrate in Section 2.3, the preference for either one of these schools guides the choice of research instruments with which data on the research subject is collected. Ontologically, our research follows an interpretive philosophy. We propose, that this philosophy is more appropriate for the purpose of investigating *how* we can support organizations in improving their coordination and in designing supporting information systems. It will be very hard, if not impossible, to answer our research question and pursue our research objective without entering the arena of organizational reality ourselves. We will elaborate on this in more detail in the next two sections.

2.3 Research strategy

A research strategy is seen to concern the steps are carried out to execute the inquiry into the phenomenon studied. It outlines the sequence of data acquisition and analysis. Taking a problem solving perspective on scientific research, Mitroff et al. [1974] demonstrate that in theory there are 3555 different strategies to conduct a scientific inquiry. Churchman [1971] distinguishes between five strategies, or *inquiring systems*:

1. *Leibnitzian*: expanding scientific knowledge by formal deduction from elementary forms of knowledge.

2. *Lockean*: expanding scientific knowledge by induction from sensing experiences, endowing them with properties, and combining them with previous experiences.
3. *Kantian*: expanding scientific knowledge by formal deduction as well as by informal experiencing through a set of '*a priori sciences*'; a blend of Leibnizian and Lockean.
4. *Hegelian*: expanding scientific knowledge '*objectively*' by identifying conflicting interpretations of observations, and going beyond this conflict through synthesis.
5. *Singerian*: expanding scientific knowledge by adapting it '*endlessly*', inductively, and multidisciplinary based on new observations.

The choice of a research strategy or inquiring system is based on the *nature* of the research problem, and on the status of *theory development* in the research field. In general, we can distinguish between three phases of theory development, see e.g. [Galliers 1991; Jarvenpaa 1988; Pervan 1994]: theory building, theory testing, and theory expanding. Within most research areas, there often is an advancement from theory building to theory testing and then theory expanding.

With respect to the *nature* of our research problem we argue that our research problem represents an ill-structured problem [Sol 1982]. It is difficult, if not impossible to solve the problem of improving organizational coordination and supporting information technology in a purely deductive way. This is reinforced by the fact that there is no appropriate *theory* available on designing organizational coordination nor on integrated organization and information system design. Therefore, we argue that it is imperative to inductively identify and further explore the requirements of our design approach in reality and develop our theory accordingly.

We propose that in order to meet this requirement, it is most appropriate to apply an adapted version of the inductive-hypothetic model cycle as research strategy. This model cycle is based on Churchman's Singerian inquiring system, see [Churchman 1971; Bosman 1977; Sol 1982]. The main benefits of the inductive-hypothetic strategy can be derived from the nature of the Singerian inquiring system [Sol 1982, p.4]:

1. It stresses the inductive specification, testing, *and* expanding of a theory.
2. It offers possibilities for interdisciplinary research.
3. It makes space for the generation of various solutions for a problem situation.
4. It emphasizes learning by considering analysis and synthesis as interdependent activities.

Our inductive-hypothetic research strategy consists of five activities, see figure 2.1 and [Jong 1992; Meel 1994; Sol 1982]:

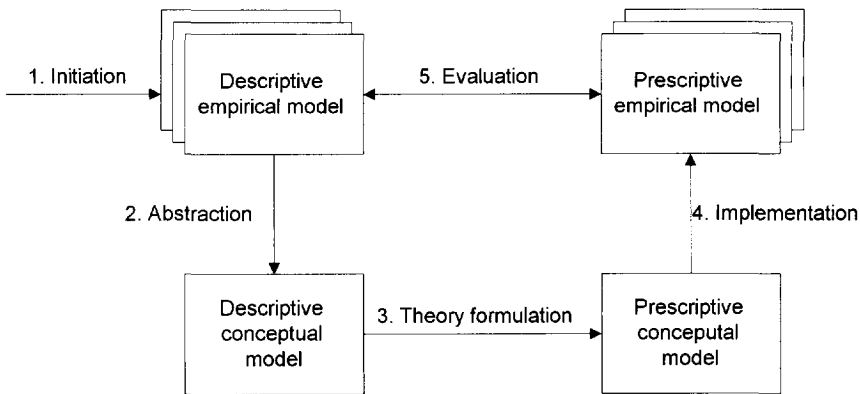


figure 2.1 The inductive-hypothetic research strategy.

1. First, a number of initial theories, regardless how rudimentary, are identified. Using these theories, a number of empirical situations are investigated and described in one or more empirical models.
2. Second, the essential aspects, especially perceived problems, from the empirical situations are abstracted into a descriptive conceptual model.
3. Third, from the descriptive conceptual model a prescriptive, or normative, conceptual model is derived in the form of a theory. This theory should be capable of solving the problems observed.
4. Fourth, to test and validate the prescriptive conceptual model, the model is implemented in one or more prescriptive empirical situations.
5. Finally, the results of the prescriptive empirical situations are evaluated. Additional requirements for improving the prescriptive conceptual model may be identified. In order to further test and expand the theory inductively, it may be used (in combination with other theories) as an initial theory in a number of new empirical situations.

Note that in the context of the inductive-hypothetic strategy, the term ‘theory’ is used in a broad sense. A theory constitutes a proposed solution for a problem situation, for example, in terms of (a combination of) a set of problem handling guidelines and modelling concepts [Bots 1989; Dur 1992; Wierda 1991], modelling support [Jong 1992; Verbraeck 1991], or an inquiry system [Meel 1994; Sol 1982; this research].

Although our strategy clearly describes the order and interdependence of the modelling steps, it does not offer guidelines to carry out the individual steps. To this end, we describe the instruments we used in our research in the next section.

2.4 Research instruments

The instruments of a given research concern the means with which data on the phenomenon studied is collected and subsequently analyzed. By describing the research instruments, it is defined how the steps from the research strategy are carried out. In this section we first explain why we choose case study research and action research as our research instruments. We subsequently describe the characteristics of these research instruments and elaborate on the way in which we employed them.

2.4.1 Choosing the research instruments

There are a variety of instruments available to conduct research on organizational phenomena and information systems. We distinguished the most commonly employed instruments in table 2.1, using two classifications. First, the instruments were classified according to the setting in which they are employed: instruments can be employed in natural settings, in constructed settings, or they can be setting independent [Benbasat 1984]. Second, instruments were classified according to their predominant underlying philosophy: positivism and interpretivism [Burrell and Morgan 1979; Galliers 1991]. For a detailed discussion of the nature and methodological issues involving these instruments, we refer to [Benbasat 1984, 1989a; Cash and Lawrence 1989; Galliers 1991; Galliers and Land 1987; Iivari 1991; Kraemer 1989; Orlikowski and Baroudi 1991; Van Horn 1973; Wynekoop and Conger 1991].

Research instrument	Type of setting			Underlying philosophy	
	Natural	Constructed	Independent	Positivistic	Interpretivistic
Lab experiment	-	X	-	X	-
Field experiment	X	-	-	X	-
Case study ¹	X	-	-	-	X ²
Action research	X	-	-	-	X
Survey	-	-	X	X	-
Gaming ³	-	X	-	-	X
Simulation ³	-	X	-	X	-
Prototyping/Systems development	-	-	X	X	-
Review	-	-	X	-	X

¹ Sometimes, a *field study* is identified as an ex-post, short period type of case study research, [Iivari 1991; Wynekoop and Conger 1991].

² Some authors claim that a case study is positivistic [Galliers 1991; Lee 1989; Yin 1988]. However, we agree with Smithson who states that case studies are often rich of contextual detail as an aid to understanding phenomena [1991, p.366].

³ Gaming may also be referred to as *organizational simulation* and simulation as *programmatic simulation* [Wynekoop and Conger 1991].

table 2.1 Instruments in organization research and information system research.

In organization and information systems research there is strong preference for objective, deterministic instruments, like experimentation [Burrell and Morgan 1979; Orlikowski and Baroudi 1991; Vogel and Wetherbe 1984]. However, studies with these

instruments are often criticized for their lack of external validity, for their lack of an adequate theoretical basis describing the nature of the variables and relationships to be quantified, for the questionable reliability and validity of measuring instruments, for the diversity of experimental tasks making outcomes of different experiments incomparable, and for their lack of focus on process issues [Benbasat 1989b; Clark and Ackoff 1959; DeSanctis 1989; Dickson 1989; Galliers and Land 1987; Jarvenpaa et al. 1985; Orlikowski and Baroudi 1991; Palys 1989; Rohrbaugh 1989; Todd and Benbasat 1987; Weill and Olson 1989; Zigurs 1993]. However, studies with subjective, interpretivist instruments are often criticized for their limited generalizability, for their lack of 'scientific rigor', for the possibility of researcher bias, for possible different interpretations by different researchers, and for often ignoring historical data as explanation for current situations [Benbasat et al. 1987; Clark and Ackoff 1959; Galliers 1991; Lee 1989; Orlikowski and Baroudi 1991; Wynekoop and Conger 1991].

The selection of a particular instrument depends, among other things, on the amount of existing knowledge (theory), on the nature of the object of research, on the aspects of the research object that the researcher wants to focus on, and on the resources available to the researcher [Benbasat 1984; Benbasat et al. 1987, 1992; Orlikowski and Baroudi 1991]. It even appears that the choice of an instrument is more often based on the institute's tradition and on publishing guidelines of academic journals rather than on rational grounds [Galliers 1991; Orlikowski and Baroudi 1991; Vogel and Wetherbe 1984].

When studying organizations and information systems, employing multiple research instruments or *pluralism* is often encouraged [Dennis et al. 1991; Denzin 1978; Jarvenpaa et al. 1988; Leeuw 1990; Morgan 1983; Nunamaker et al. 1990; Orlikowski and Baroudi 1991; Vogel and Nunamaker 1990]. This enables the researcher to compare and contrast (triangulate) the different data sets, and create a richer and more revealing picture of the phenomenon studied. In addition, the combination of instruments may counterbalance their respective weaknesses and strengths. However, a pluralistic approach may generate problems when the instruments used have different underlying philosophies with opposing assumptions [Burrell and Morgan 1979; Smithson 1991].

The research instruments we employ in our research are *case study research* and *action research*. We contend that these instruments fit our inductive-hypothetic research strategy very well, see also [Sol 1982]. Benbasat et al. [1987, p. 370] give three arguments to employ case study research. We propose that these arguments are also valid for action research:

1. The phenomenon needs to be studied in its *natural setting*. We find that the phenomenon we are studying, organizational coordination and supporting

information systems, is too complex to emulate in a constructed setting and to study with objective instruments, see also [Yin 1989b; Zmud et al. 1989].

2. The focus should be on the process, namely 'how' and 'why' questions. Our research question relates to supporting the *change process* within organizations. Therefore, emphasis is laid on process aspects.
3. *Few previous studies* have been carried out in the research area. As far as we know there have been few similar studies on developing a design approach for improving organizational coordination.

The characteristics of both instruments and the way in which we design our research, are discussed in the following two subsections.

2.4.2 Case study research and action research

Case study research is intended to examine a contemporary phenomenon in its natural setting where the boundaries between the phenomenon and its context are unclear. Employing multiple methods of data collection [Benbasat et al. 1987; Eisenhardt 1989; Hartley 1994; Yin 1989a and 1989b], case study research can be explanatory, exploratory, or descriptive, focusing on natural phenomena in order to build or test theories [Benbasat et al. 1987; Galliers 1991; Hartley 1994; Yin 1989a]. It can be characterized as qualitative and observatory, using predefined research questions [Yin 1989a and 1989b]. In carrying out case study research, a researcher goes through the phases of design, collection, analysis, and report [Eisenhardt 1989; Yin 1989a]. This process is mainly sequential in nature [Yin 1989a].

Action research is intended to have the dual outcomes of action (or change) and research (understanding) [Dick 1993]. It is an inquiry into how people design and implement action in relation to each other [Argyris et al. 1982]. Action researchers participate or intervene in the phenomenon studied in order to apply a theory to practice and evaluate its worth [Argyris and Schön 1989; Benbasat et al. 1987; Checkland 1981a and 1991; Clark 1972; Dick 1993]. Action research can be characterized as cyclic, participative, qualitative, and reflexive [Argyris et al. 1982; Dick 1993]. It can be used for theory building, as well as for theory testing and theory expanding [Galliers 1991]. In carrying out action research, a researcher goes through various phases: planning, execution, observation, reflection, and back to planning [Checkland 1991; Dick 1993; Zuber-Skerritt 1991]. This process is not sequential or discrete, the researcher may be cycling between the phases many times [Zuber-Skerritt 1991].

Sometimes, action research is seen as a subset of case study research [Galliers 1991]. In table 2.2, we summarize the most significant differences between case study research and action research.

Case study	Action research
Researcher is observer	Researcher is active participant
Exploratory, explanatory, or descriptive	Prescriptive, intervening
Focus on 'how?' and 'why?'	Additional focus on 'how to?'
Possibly positivistic	Interpretivist

table 2.2 Differences between case study research and action research.

Case study research may tend to be more objective than action research, focusing on describing the processes in the situation studied. Action research focuses on intervention and actually designing the processes in the situation studied. Notwithstanding the differences between case study research and action research, they are not often practiced in their 'pure form'. Looking at the level of participation by the researcher, a continuum between case study research and action research can be visualized, see figure 2.2. In the middle is *participant observation*, where the researcher interacts socially with the organizational subjects, but does not intervene [Waddington 1994].

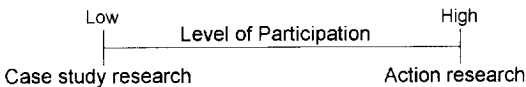


figure 2.2 Level of participation in case study research and action research.

The major strengths of case study research and action research are that they involve the most direct form of observation, that they capture reality in greater detail, and that subjects forget that they are indeed the subject of the research [Checkland 1981a; Clark 1972; Galliers 1991; Wynekoop and Conger 1991]. In particular action, research allows for a good, close relation with the subjects, and permits theory application and evaluation concurrently [Checkland 1981a; Clark 1972; Galliers 1991]. The main critique on case study research and action research related to criticisms on interpretivist research instruments, see section 2.3.1. In addition, action research is sometimes seen as a mere excuse for consultancy. However, contrary to consultancy, action research studies are aimed at carrying out scientific reflection on the theory applied in order to adapt and improve them. In addition, the results of an action research study are usually public [Checkland 1991; Clark 1972; Jönsson 1991].

2.4.3 Design of our research

A way to deal with the weaknesses and criticisms of case study research and action research is by consciously designing the research in which these instruments are applied. To this end we design our research using the case study design guidelines as defined by [Benbasat et al. 1987; Yin 1989a] that concern the exact design of the research, site selection, data sources, and the method of analysis. In addition, we

discuss the initial theoretical framework with which we entered the research. To facilitate the discussion, we refer to each visit to a particular site as a 'case study'.

Exact design of the research

The exact design of our research concerns the goal of the research, the unit of analysis, and the question of whether to use single or multiple cases. As we describe in Chapter 1, the *goal of our research* is to formulate a design approach for facilitating the process of improving an organization's coordination and its supporting information technology. In order to get a rich picture of the coordination and change processes at hand, data in each case is collected with respect to three *units of analysis*, or organizational levels: the macro, the meso, and the micro level. However, as the central issue of our research is organizational coordination, the focus of our analysis efforts is on the *meso level*.

An important issue in the design of a case or action research study concerns the *number of sites* to be visited. Limited resources and available sites may restrict a study to a single site. Single case studies are often seen as inferior to multiple case studies with respect to generalizability. However, when selected with care, a single case study may be very successful in terms of theory formulation and theory testing [Eisenhardt 1989; Markus 1989; Yin 1989b; Walsham and Waema 1994]. In our research, we visited four sites, see table 2.3. Visiting multiple sites allowed us to contrast and compare the results of individual sites and to *build* our theory irrespective of a particular business sector. The first two case studies were descriptive. The third case study was both prescriptive and descriptive. Originally, the third case was selected as a 'critical case' [Markus 1989; Yin 1989b], to *test* the theory that we built after the first two studies. As our theory needed further enhancement, we simultaneously used the third study descriptively to *extend and further build* our theory. We used our last case study to test our extended theory prescriptively. As depicted in table 2.3, in the first case study, our research instrument was case study research, the other three cases involved action research.

No.	Case	Report	Nature		Instrument	
			Descriptive	Prescriptive	Case study	Action research
1	DUB	Chapter 4	x	-	x	-
2	NOD	Chapter 5	x	-	-	x
3	CID-Preparation	Chapter 7	x	x	-	x
4	CID-Execution	Chapter 9	-	x	-	x

table 2.3 Sites visited in our study.

Criteria for site selection

Though the selection of research sites often is a matter of '*planned opportunism*' [Pettigrew 1989, p.16], it should be a conscious and formalized step in the design of

a case study. Based on our research question, our research objective, and the description of our research field in Chapter 1, we formulated four criteria to select organizations for our case studies. First, the organization must be in the process of changing the coordination of its processes, or must require to enter such a change process. Second, as part of this change process, the organization should recognize the necessity to develop or improve its information technology to support the coordination of its processes. Third, the organization must be willing to cooperate in our research. It must provide access and information that is required to conduct our research. Finally, each organization should preferably represent a different business sector. These criteria led us to select the following four cases:

1. *The Department of Unemployment Benefits (DUB)*. The DUB is part of the Dutch Ministry of the Interior. It is responsible for the monthly provision of compensation payments for former civil servants that involuntarily lost their job.
2. *The Neurology Outpatients' Department (NOD)*. The NOD is part of Leyenburg Hospital in The Hague. It handles patients that require treatment for neurologic symptoms and disorders without being admissioned in the hospital.
3. *The Criminal Investigations Department (CID)*. The CID is part of the Amsterdam Municipal Police Force. It is responsible for combating organized crime. In this case study, we focused on the CID departments that are involved in the *preparation* of investigations into criminal organizations.
4. *Again, the Criminal Investigations Department (CID)*. In the last case study, we revisited the CID, but this time we focused on the CID departments that are involved in the *execution* of investigations into criminal organizations.

Determination of data sources

During the case studies we used both qualitative and quantitative data sources to create a rich representation and to permit comparison and contrast of the collected data (triangulation). *Qualitative data sources* included semi-structured and open interviews, attendance at meetings, our own observations, reviews of reports with respect to the participating organizations, and reviews of relevant literature on the work domain of the organizations. *Quantitative data sources* included simulation models, questionnaires completed by members of the organizations, and meeting logs of computer supported meetings that were held with members of the organizations. The way in which we made use of the data sources is elaborated upon in the chapters that report on the case studies, see table 2.3.

Method of analysis

Following Meel [1994] each case study was set up and planned as a project. The interpretation and analysis of the data from our data sources was documented in several reports. After each project phase, a report was written and discussed with the participating organization. After the completion of each case study, a final report was

written and presented to the relevant members of the organization, see [Meel et al. 1994; Vreede 1992a, 1992b, and 1993a; Vreede et al. 1995d].

Reflection on the theory that we were developing, was not only carried out by writing this dissertation and the case study reports, but also by writing a number of papers. These papers covered different aspects of our theory, such as 'problem solving' [Verbraeck et al. 1995], 'dynamic modelling' [Vreede 1993b; Vreede et al. 1993a, 1994, 1995a, 1996], 'modelling organizational coordination' [Eijck and Vreede 1994 and 1995; Vreede and Eijck 1994, 1996], 'animating organizational processes' [Meinsma et al. 1995; Verbraeck and Vreede 1993 and 1994; Vreede and Verbraeck 1996], and 'group support for participative design' [Vreede 1993b, 1996; Vreede et al. 1995b, 1995c, and 1995e; Vreede and Sol 1994a, 1994b, and 1994c].

As stated in the previous subsection, case study research and action research are often criticized for relying too much on subjective interpretations of collected data. One way to counterbalance the researcher's bias is by using various data sources, as we did in our studies. Another way is by using *research teams* to collect and analyze the data [Hartley 1994]. Objectivity can then be sought in combining the subjective interpretations of different observers [Churchman 1971]. In our case studies, several additional researchers were involved. They included colleagues from our school (all cases), colleagues from the School of Social Sciences of Erasmus University Rotterdam (Case 2), a colleague from the University of Minnesota (Case 3), and students pursuing their masters degree by participating in parts of a study (Case 3 and 4).

Initial theoretical framework

In addition to complying with the above design guidelines for case studies, we also need to make the initial theoretical framework explicit. This framework, also referred to as an intellectual framework [Checkland 1985], offers a *rough* outline of the theories with which we enter our *descriptive* case studies. Its main purpose is to facilitate and guide the data collection process, to confine the possible level of generalizability, and to avoid ending up with 'story telling' rather than theory building and theory testing [Checkland 1985 and 1991; Glaser and Strauss 1967; Hartley 1994; Markus 1989; Yin 1989a and 1989b].

Our initial theoretical framework is twofold. The first part involves a *simulation based inquiry system*, embodying a problem solving perspective on organizational change. This inquiry system is referred to as the 'dynamic modelling approach' and is discussed in Chapter 3. The second part involves a *group support based inquiry system*, embodying a set of techniques and support for the participative design of organizational processes and information systems. This inquiry system is referred to as 'group support for participative design' and will be discussed in Chapter 6.

2.5 Research outline

In essence, every research project should comprise of an investigation into a particular application area 'A', using a particular intellectual framework 'F' according to a particular methodology 'M' [Checkland 1985 and 1991]. Using these constructs, we can define our research as follows: Our application area 'A' involves organizations that are forced to reconsider the coordination of their processes and the information technology that they employ to support this coordination. Our intellectual framework 'F' involves dynamic modelling, group support, and techniques for participative design as our initial theories. Finally, our methodology 'M' involves a combination of an interpretivist research philosophy, an inductive-hypothetic research strategy, and case study and action research as research instruments.

In figure 2.3 we depict the outline of our research according to our inductive-hypothetic research strategy as described in Section 2.3. As visualized, we completed two inductive-hypothetic cycles in our research. (Note that the 'abstraction' and 'theory formulation' activities are combined). The *first cycle* is initiated by describing the application area of our research in Chapter 1 and our research approach in Chapter 2. An initial theory on dynamic modelling was applied in the first two case studies, described in Chapter 4 and 5. Reflection during these cases suggested two major problems. First, the initial *way of working* and *way of modelling* appeared to pay little structural attention to support stakeholder involvement. Therefore, we explored and reviewed how animation modelling techniques can be used to improve stakeholder involvement in problem diagnosis, solution finding, and design. Second, the initial *way of modelling* also appeared to be insufficient with respect to modelling organizational coordination. In response, we defined a body of concepts to build conceptual models of organizational coordination, and we outlined the transformation of conceptual models into simulation models. Chapter 3 describes the synergy between the initial theory on dynamic modelling and the enhancements that were brought forth from the first two case studies. Although the new theory on dynamic modelling was developed in parallel with the two case studies, we describe it before discussing the case studies. This allows us to illustrate the new theory's applicability as a form of 'proof by demonstration'.

The new theory on dynamic modelling was employed *prescriptively* in the third case, the results of which are described in Chapter 7. Reflection on this case suggests two major problems. First, it appeared to be *less appropriate to use simulation techniques* for diagnosing the current situation and identifying alternatives for organizational change. This was believed to result from the fact that (1) the case involved too many ill-structured, unique activities, (2) the current situation had been too unstable during the last few years to serve as a yardstick, and (3) there was hardly any empirical data available. Second, notwithstanding the merits of the dynamic modelling approach to

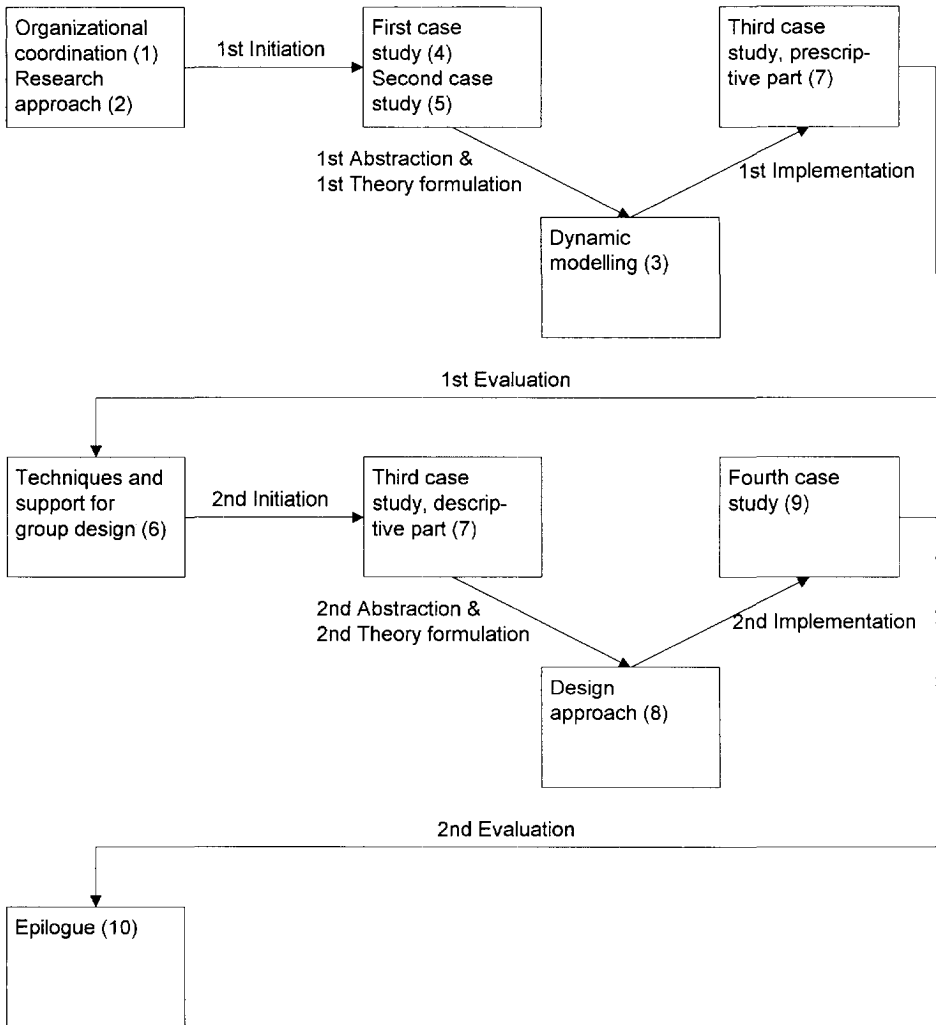


figure 2.3 Outline of the dissertation.

elicit information requirements, the approach requires additional techniques and support for the *subsequent design and development of the supporting IT*.

To compensate for these two weaknesses, we postulated that more attention should be paid to involving a group of stakeholders in the change process. Therefore, we initiated our *second cycle* by describing techniques and support for group design to support stakeholder involvement in the diagnosis, analysis, and design of organizational change and related information technology. The third case study was now further used *descriptively* to explore the merits of these techniques and support. The techniques and support for group design are described in Chapter 6, before the actual case study in

which they were applied (Chapter 7), for the same reasons Chapter 3 describes dynamic modelling before the associated case studies. Based on the reflection on the *descriptive* part of the third case study, we enhanced the dynamic modelling theory with the group design techniques and support. The result is a prescriptive design theory, i.e. the design approach constituting our research objective, that is further enriched with graphical group modelling support, see Chapter 8. A final *prescriptive* case study employing the design approach is described in Chapter 9. Finally, Chapter 10 reflects on the research as a whole and an agenda is defined for future research.

To conclude this chapter we want to point out that readers only interested in the results and the lessons from the case studies, can restrict themselves to the introduction, problem description, and reflection sections of each case study chapter.

3 Dynamic Modelling for Organizational Change

'A process cannot be understood by stopping it. Understanding must move with the flow of the process, must join it and flow with it.'

- Frank Herbert

3.1 Introduction

In this chapter we propose dynamic modelling as an approach to facilitate the process of organizational change. The dynamic modelling approach has been applied in the first two case studies. There have been various previous experiences with dynamic modelling in several research projects [Sol 1992], which represent our initial theory on dynamic modelling. Examples include designing interorganizational information systems for international goods transport by rail [Wierda 1991], developing an information system to support treasury management and in-house banking [Motshagen 1991], analyzing schedules in a production environment [Verbraeck 1991], evaluating information systems and diagnosing processes in information intensive organizations [Dur 1992], using fleet-management systems with mobile data communication [Schrijver 1993], supporting decisions on investment for the use of electronic data interchange in the port of Rotterdam [Streng 1993], and analyzing organizational and information change alternatives for the Amsterdam Municipal Police Force [Meel 1994].

The experiences in our first two case studies combined with the experiences from the above studies led us to formulate the dynamic modelling approach discussed in this chapter. Following the framework by Sol [1990], this approach is discussed in terms of its way of thinking, its way of modelling, and its way of working. The main focus of this chapter is on the way of modelling and to a lesser extent on the way of working as these are the result of the experiences of the first two case studies and the reflection on these. For an elaborate discussion of the way of controlling of the dynamic modelling approach we refer to [Meel 1994; Sol 1992a and 1992b].

3.2 Way of thinking

The way of thinking concerns the Weltanschauung of the dynamic modelling approach. It delineates how we observe the application domain in which the dynamic modelling approach is employed.

We adopt a systems perspective on organizations and organizational processes. According to this perspective, an organization can be seen as a system, being a set of related objects [Ackoff 1971; Checkland 1981a; Leeuw 1990; Sol 1982]. Organizational processes can be seen as structured sets of subsequent transformations [Leeuw 1990]. These transformations are carried out by a set of related actors (objects), that have their own goals and their own subjective perception of their working environment [Leeuw 1990].

Improving coordination of organizational processes and constituent activities can be considered as a complex activity. First, there are often many actors involved, whose own goals may conflict with the goals of the organization or the goals of the processes that they are part of. Second, there are many possible courses of action to change organizational coordination. It is not clear which course of action will be most efficient and effective beforehand. Third, there are usually many options or alternatives for achieving organizational coordination. As the effects of these change alternatives are often dangerous or impossible to test in reality, the effects of these changes are seldom known in advance. Finally, apart from 'rationale', there may be other (conflicting) factors, such as political, cultural, and social factors, that influence the actual change process.

Following Sol [1982], we argue that it is appropriate to take a problem solving perspective on the process of designing improved organizational coordination. This complex process can be seen as solving an ill-structured problem. An ill-structured problem is defined as a problem that does not meet the requirements of a well-structured problem [Sol 1982]. We follow Sol in defining the requirements of a well-structured problem as follows [1982, p. 5]:

- 1. The set of alternative courses of actions or solutions is finite and limited.*
- 2. The solutions are consistently derived from a model system that shows a good correspondence.*
- 3. The effectiveness or the efficiency of the courses of action can be numerically evaluated.'*

There are different ways to try and solve ill-structured problems. Simon [1981] distinguishes between substantive rationality and procedural rationality. Substantive rationality assumes perfect conditions for the problem solving process so that

eventually the optimal solution is identified. However, in ill-structured real life problem situations, these perfect conditions hardly ever exist. In these situations, the focus shifts from finding the optimal solution to finding a *process that will yield a satisfying solution*. This perspective is referred to as procedural rationality. Procedural rationality is a particular form of *bounded rationality* [Bots 1989; Sol 1982]. Following Sol [1982], we adopt a bounded rationality perspective on solving the problem of designing organizational coordination. In the next section we describe how this perspective on improving organizational coordination can be translated into a problem solving process that represents the way of working of the dynamic modelling approach.

3.3 Way of working

The way of working describes the steps that need to be taken to address the problem of improving organizational coordination following the dynamic modelling approach. As designing organizational change can be considered as solving an ill-structured problem, we need a problem solving strategy that addresses the intricate attributes of such a process. In the problem solving literature several strategies are proposed to tackle ill-structured problems, see e.g. [Ackoff 1978; Bots 1989; Mitroff et al. 1974; Simon 1947, 1973, and 1981; Sol 1982; VanGundy 1988]. Based on this literature and on the experiences gained during the first two case studies of our research, we propose the problem solving process depicted in figure 3.1 to handle the problem of designing organizational change [Sol 1982; Verbraeck et al. 1995].

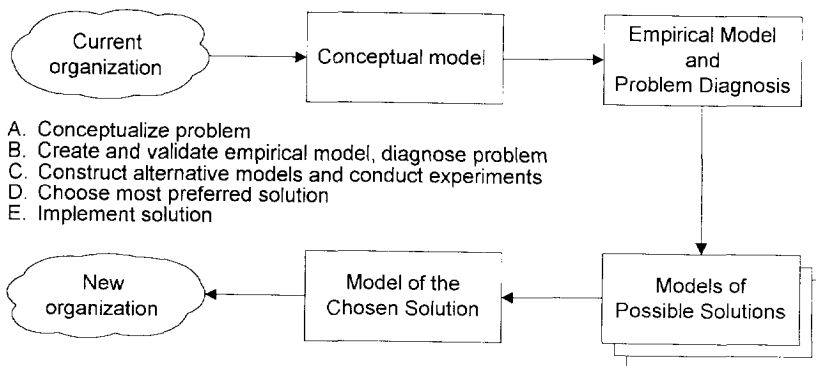


figure 3.1 The problem solving process in dynamic modelling.

This process of problem solving consists of five activities. The first activity concerns the *conceptualization* of the problem situation. The resulting conceptual model describes the structure of the problem situation in such a way that the efforts for detailed, low-level data gathering for creating the empirical model in the next activity can be focused and minimized. In the dynamic modelling approach the *object-*

attribute-action worldview has shown to be a valuable way of describing a problem situation, see [Sol 1982; Bots 1989; Dur 1992]. Using this worldview, the structure of the problem situation is modelled by identifying its discrete objects and describing their properties in terms of attributes and actions.

The main purpose of the second activity is to construct a *descriptive empirical model* of the problem situation by translating the conceptual model into a simulation model. In addition, empirical data is gathered and incorporated into the simulation model. Before this model can be used to diagnose the problem situation, it has to be tested for whether it is a sound representation of the problem situation. For this purpose, the simulation model is first *verified* to insure that the model behaves as the experimenter intends it to [Shannon 1975, p. 210]. Second, the model is *validated* to test the statistical correspondence between the model and the problem situation. A distinction can be made between replicative and structural validation, [Kleijnen 1992; Sol 1982]. In *replicative validation* the values of the endogenous variables are compared with sample data from the problem situation that was not included in the empirical model. In *structural validation* the hypothesized behavior of the simulation model is investigated, for example by giving impulses to the model and comparing the resulting values of the endogenous variables with the values found in the problem situation. After verifying and validating the simulation model, the problem situation can be *diagnosed* by experimenting with the model, e.g. by testing it under extreme conditions. Subsequently, a set of solution requirements are defined in terms of endogenous variables of the model.

The third activity concerns *seeking solutions* for the problem situation. Based upon the results from the problem diagnosis activity, several alternative solutions may be generated. The alternative solutions are worked out in detail in a number of *prescriptive* empirical models. In order to study the effects of the alternatives, a simulation experiment is conducted. An experiment embodies a number of simulation runs under the same treatment. A treatment consists of the following elements: a specification and a collection of exogenous data, initialization conditions, run control conditions, and a specification of endogenous data [Sol 1982]. The results of the experiments are analyzed by conducting statistical tests to compare them with the as-is simulation model and the solution requirements.

Following an analysis of the experiments with the *prescriptive empirical models*, the solution to be implemented in the problem situation has to be *chosen*. This solution may be a combination of alternatives that were defined in the previous activity or it may be to leave the situation as it is.

Finally, in the last activity of the problem solving process the solution has to be *implemented* in the organization to actually solve the problem situation. In this stage,

the simulation model of the solution can be used to support training or post-evaluation. For the latter purpose, organizational data defined by the simulation model's endogenous variables, has to be collected and statistically compared with the model output. The implementation of the solution in the organization may bring about a new problem situation. Hence, the problem solving process may be cyclical in nature.

During the problem solving process described above, a number of models of the organization are constructed and analyzed. It is important to notice that these models are not built within a number of discrete steps. At any time, earlier models may be modified according to the requirements and insights that surface at a later stage. It is therefore more accurate to think of the problem solving process in terms of a *model sequence* than in terms of linear activities. These models will be closely examined in the next section.

3.4 Way of modelling

The way of modelling describes the different types of models that are constructed during a dynamic modelling study. In the dynamic modelling approach a clear distinction is made between *conceptual* models which define problem structure, and *empirical* models which enable analysis and diagnosis of the problem or possible solutions. In this section, we describe the construction of both these models. However, we first discuss the outline of this section on the basis of figure 3.2, depicting the relationship between the conceptual model, the empirical model, and several aspect models of these models.

The conceptual model is constructed using the object-attribute-action worldview as a conceptualization filter. Hence, this model consists of a number of object class definitions, which will be described in Section 3.4.1. Based on these object classes, we define a number of *aspect models*. An aspect model is a representation of certain model elements that aims to highlight specific characteristics of a model. The three aspect models of the conceptual model are the interaction model (Section 3.4.2), the process model (Section 3.4.3), and the actor model (Section 3.4.4).

The empirical model is constructed using a specification filter in which a simulation language is used as a model construction language. This filter supports the translation of the object definitions from the conceptual model into simulation code and the collection of empirical data from the problem situation (that first passes through the conceptualization filter) to be combined with the simulation code. The specification filter also includes several reduction transformations like leaving out or aggregating concepts or data. The resulting empirical model consists of a set of empirical data, simulation code, and an experimental design, as defined by the model builder, to carry

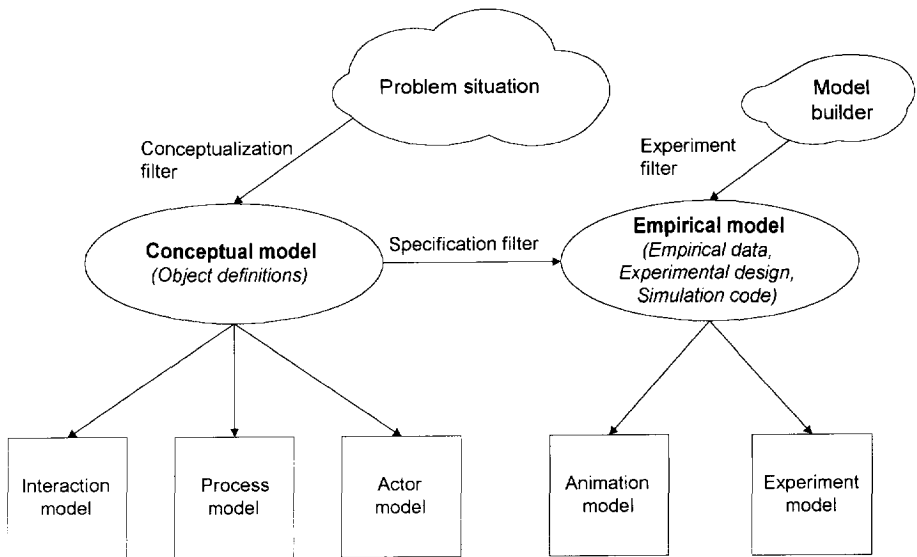


figure 3.2 The conceptual and empirical model and the different aspect models.

out the analysis of the modelled situation. In Section 3.4.5 we describe the transformation of the conceptual model elements into simulation code. Also, we define two aspect models of the empirical model: the animation model (Section 3.4.6) and the experiment model (Section 3.4.7).

3.4.1 The conceptual model

In this subsection we first briefly discuss a number of modelling techniques that previously have been employed to conceptualize organizational coordination. Building on the work of Bots [1989] and Dur [1992], we then introduce a set of basic concepts for modelling organizational processes and their coordination. Together these concepts, object definitions, constitute the conceptual model of a problem situation.

Modelling techniques for conceptualizing organizational coordination

In the literature several modelling approaches can be found that support modelling organizational coordination. The *Petri net formalism* [Petri 1980; Reisig 1985] is one of the first approaches to modelling concurrency and synchronization. Typical applications areas are communication protocols, computer systems, and distributed systems, [Aalst 1992]. Based on the Petri net formalism, a substantial number of modelling approaches have been formulated, see e.g. [Sol and Crosslin 1992; Sol and Hee 1991]. However, two reasons make Petri nets less suitable for modelling organizational coordination. First, they are difficult to communicate to people who are not trained in reading them. Second, Petri nets tend to grow very large and incomprehensible, if the modeled situation gets slightly complex.

A mechanistic view on organizational coordination is offered by *diplans*, a formal graphical language for expressing plans [Holt 1988]. The strength of diplans models is that they make the coordination effort in a certain situation explicit. Their weaknesses are similar to those of Petri nets: they are difficult to communicate and quickly grow very complex and incomprehensible.

Flores and Winograd approach modelling of coordination from a *linguistic perspective* [Flores et al. 1988; Winograd and Flores 1986]. By looking at how communication brings about coordination, they model coordination actions as different types of *speech acts*, such as requests, promises, assertions, and declarations. This modeling approach has been proven especially useful for modelling office information systems that support group activities. However, we argue that the speech act formalism is less suitable or at least unnatural for modelling coordination mechanisms that do not involve explicit communication between the actors, such as situations where working procedures are standardized.

Task structures as proposed by [Bots 1989] offer a means of modelling the activities (tasks) and their relationships of individual actors. However, coordination that involves two or more actors at the same time cannot be modeled in a straightforward manner, even though the formalism offers a construct for modelling situations where coordination with other actors is required. Task structures have proven to be easy to communicate [Bots 1989]. Dur [1992] proposes *task structures* from a different perspective: in his task structures, tasks can be performed by different actors, but all involve the same item, e.g. a form or product being processed. Because of this focus on items, it is not possible to describe coordination tasks in which no item is being processed. Furthermore, it is not possible to use single task structures to describe the coordination of processes that involve more than one class of items. Like Bots' task structures, Dur's task structures have proven to be easy to communicate [Dur 1992; Meel 1994].

Based on these experiences, we will build on the work of Bots and Dur and propose a set of object classes as basic modelling constructs for the conceptualization of a problem situation. With these object classes we will define three aspect models that focus on different characteristics of the conceptual model: the interaction model, the process model, and the actor model. We postulate that the conceptual model together with the three aspect models is easy to communicate and enables us to model the coordination of organizational processes that involve zero, one, or more items.

Building blocks: Object classes

As stated previously, we see organizations as systems, consisting of networks of communicating objects. We adopt an object-attribute-action worldview to conceptualize organizational situations by describing the attributes and actions of the perceived

objects. The attribute part contains data elements describing the state of an object. The action part contains the tasks and decisions that reflect the possible behavior of an object. Our object-attribute-action worldview is close to the object-oriented view, where objects have attributes and methods, see e.g. [Kim and Lochovsky 1989; Meyer 1988]. However, the object-oriented view usually includes more aspects, like inheritance, encapsulation, and polymorphism, see e.g. [Booch 1994; Rumbaugh et al. 1991].

We follow Bots [1989] and Dur [1992] in stating that the main advantages of the object-attribute-action worldview are that:

1. The notion of an object quite naturally describes both what it *is* (attributes) and what it *does* (actions).
2. The distinction between object class and object instance naturally represents the difference between the conceptual and empirical level of abstraction.
3. Autonomous behavior by organizational actors can be adequately represented.
4. The close relationship between the phenomena of the problem situation and their object equivalents increases the comprehensibility of the model and facilitates its communication to others.

In order to conceptualize organizational processes and organizational coordination, we distinguish between eight 'leaf' object classes that inherit certain properties from a number of superclasses. This is depicted in the object hierarchy in figure 3.3, see also [Eijck et al. 1995].

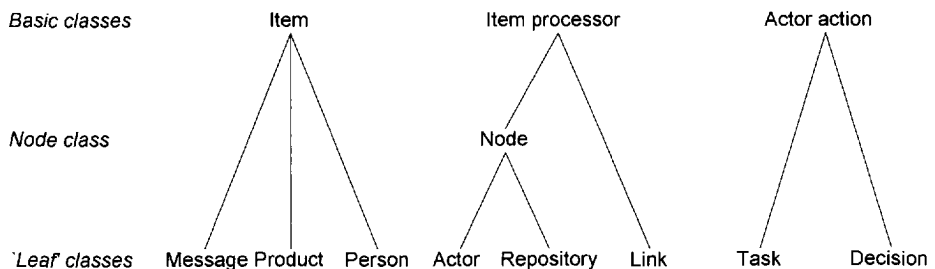


figure 3.3 Object hierarchy for the conceptual model.

There are three *basic classes*: Items, item processors, and actor actions. An *item* is a *passive* object that can be communicated or exchanged between item processors. It does not carry out any actions itself, it only contains information in its attribute values. This information may trigger the receiving item processor to select a particular course of action [Ackoff and Emery 1972; Checkland 1981b]. We use the term 'item' instead of 'message' because an item may not only represent the information itself but also its

physical carrier. The arrival of a physical carrier at an item processor may also trigger actions.

Therefore, we distinguish between three types of items: *Messages*, *products*, and *persons*. With messages, the item information is represented directly by the message contents. With products and persons, the item information is represented indirectly by the product's or person's properties. Note that people are only modelled as person items if they are 'passive' in the sense that they do not exhibit autonomous behavior. The formal object definitions of the item class and the three item types are depicted below.

object class ITEM	object class MESSAGE inherits from ITEM	object class PRODUCT inherits from ITEM	object class PERSON inherits from ITEM
attributes list of Identifying attributes	attributes Information contents	attributes Product properties	attributes Person properties

An *item processor* is an *active* object that can perform actions. Its nature, status, and potential are respectively described by a list of identifying attributes, a list of status attributes, and a list of the items that the item processor can handle. An item processor's actions involve sending and receiving items to and from other item processors, altering its own status attribute values, and altering the attribute values of the items that it processes.

Item processors can be further divided into *nodes* and *links*, that both inherit from the item processor object class. A *node* stores, retrieves, and processes items. A *link* connects nodes in such a way that they can exchange items. A link is specified by its attributes. It routes items from node to node, and possibly vice versa, in a way that is determined by the technology it represents. The formal object definitions of an item processor, a node, and a link are depicted below.

object class ITEM PROCESSOR	object class NODE inherits from ITEM PRO- CESSOR	object class LINK inherits from ITEM PROCESSOR
attributes list of Identifying attributes list of Status attributes	attributes	attributes starting node ending node uni- or bi-directional list of Other attributes
actions receive item send item	actions store item retrieve item	actions route item

A *node* is a superclass for two subclasses: *actors* and *repositories*. Both repositories and actors inherit the properties of a node. When a node is an *actor*, for example a

physician, its action part, apart from inherited actions, is not empty. An actor can modify its own attribute values of those of the items it holds. When a node is a *repository*, for example a database, its action part is empty, apart from inherited actions. (However, a database that performs other actions on items besides sending, receiving, storing, and retrieving them, is modelled as an actor.) A repository contains data (items) that can be examined or changed by actors. A repositories may serve as a coordination mechanism: a job database, for example, may be the node between dispatchers and service engineers that serves as mechanism to coordinate the work of the engineers and the dispatchers. The formal object definitions of actors and repositories are depicted below.

object class ACTOR inherits from NODE	object class REPOSITORY inherits from NODE
attributes <i>list of Other attributes</i>	attributes <i>list of Other attributes</i>
actions <i>list of Actor actions</i>	actions

Finally, we distinguish between two types of actions performed by actors, or *actor action classes*: tasks and decisions. A decision constitutes the choice between two or more *courses of action*. Every task and decision should be defined in the action part of one of the actors' object definitions. The formal object definitions of the superclass 'actor action' and its two subclasses are depicted below.

object class ACTOR ACTION	object class TASK inherits from ACTOR ACTION	object class DECISION inherits from ACTOR ACTION
attributes <i>list of identifying attributes</i> <i>list of required items</i> <i>list of preconditions</i> <i>list of postconditions</i> <i>list of next actor actions</i>	attributes delay <i>list of attribute assignments</i> <i>list of subactions</i>	attributes <i>list of decision rules</i> <i>list of subdecisions</i>

Actor actions may *require items* in order to be carried out. The actor executing the actions, receives these items from repositories or from other actors. An actor only executes an action if the *preconditions* are met. As a result of this execution, the *postconditions* should be met. The transition between pre- and postconditions implicitly models the goal that is attained by executing the actor action. Pre- and postconditions may refer to the attributes of any of the defined objects. Postconditions can also refer to the conveyance of items to repositories or other actors. The list of *next actor actions* describes which actions will be carried out when the execution of the current action is completed.

Tasks take time to be executed. In order to capture the dynamics of the modelled situation, the *time delay* has to be specified for each task. This time delay may be deterministic or probabilistic. In addition, the *attribute assignments* have to be determined for each task. The attribute assignments refer to modifications in the values of the actor's attributes or those of the items it holds. Finally, a task may be compound, which means that it can be described in a number of *subactions* (subtasks or subdecisions), on a lower level of abstraction.

For each decision, the list of *decision rules* expresses the rationale with which an actor carries out the decision. A decision rule can be a probability that is expressed with a stochastic variable. It can have the form of an 'IF condition THEN decision'. Like a task, a decision can be modelled as a compound decision if it is very complex in terms of decision rules. Note that a compound decision can only consist of *subdecisions*.

The eight 'leaf' classes of the object hierarchy we now have described, (messages, products, persons, links, actors, repositories, tasks, and decisions), form the basic building blocks with which we conceptualize organizations and the way in which they execute and coordinate their processes. However, organizational processes and their coordination are often more complex than a single model can depict, see e.g. [Checkland 1988].

The same goes for the conceptual model in terms of object definitions that we just described. The object definitions themselves cannot point out clearly all the intrinsic characteristics of a modelled situation. Based on our object classes, we therefore define three models that each highlight different aspects of the way in which an organization achieves the coordination of its processes and activities. These aspect models thus enable a multi-perspective analysis of the modelled situation. The aspect models include the interaction model, the process model, and the actor model. Naturally, one may define other models that highlight different aspects. However, we propose that these three aspect models form a good basis for the analysis of a problem situation. The three models are discussed in the following subsections.

3.4.2 The interaction model

As a number of authors point out, one way to understand the structure of organizational relationships is by describing the formal and informal patterns of communication or interaction, see e.g. [Lundquist and Huston 1990; March and Simon 1958; Qureshi 1995]. The *interaction model* aims to facilitate understanding of the coordination relationships within an organization. It does so by highlighting the exchange of items between nodes. These items convey stimuli and data required to evoke and execute processes. The icons for the graphical representation of the interaction model are given in figure 3.4.

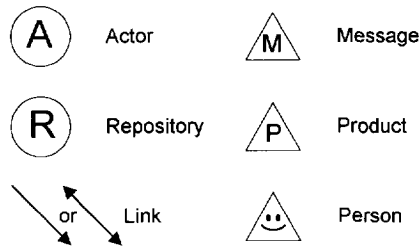


figure 3.4 Icons for the interaction model.

In the representation of an interaction model, the actor and repository icons are connected by single and double arrows, representing uni-directional and bi-directional links. The icons of the items that flow over a link are placed next to the associated arrow.

Note that an interaction model does not describe how the interaction of items takes place in time; it only illustrates which interaction relationships exist between the actors and repositories in the modelled situation. An example of an interaction model is given in figure 3.5. This model illustrates the item flows between the actors and repositories that are involved in the preparations for organizing and attending an academic conference.

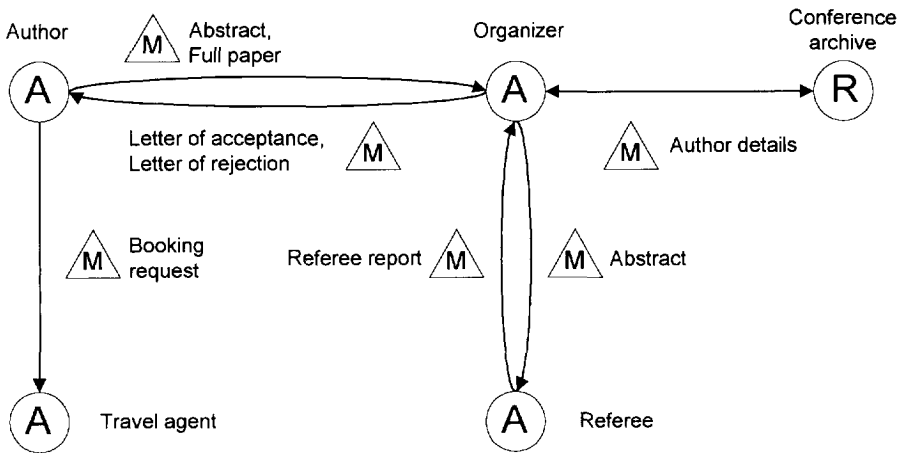


figure 3.5 Example of an interaction model.

3.4.3 The process model

The interaction model describes the exchange of items between nodes. It does, however, not describe the way in which the actors in the organization handle these items. In order to investigate this, we introduce the *process model*. We consider a process as a sequence of actions carried out by interacting nodes, to obtain a certain result or to achieve a certain goal, e.g. making a product or serving a customer. Hence, a process model depicts the interdependencies of actors' and repositories' actions (possibly involving items) that together are aimed at achieving a certain goal. Building on the work of Dur [1992], we use the symbols as depicted in figure 3.6 to make a graphical representation of a process model.

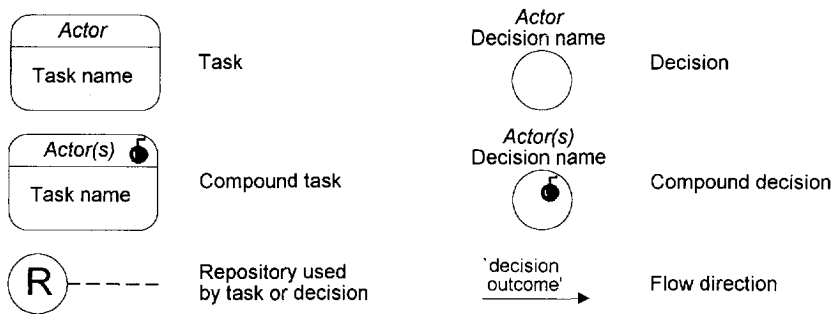


figure 3.6 Icons of the process model.

Compound tasks and decisions are depicted by a small bomb in the upper right corner of the icon, meaning that the task or decision is able to 'explode' into one or more subtasks or subdecisions. All actors involved in compound tasks or decisions are identified in the icon. The flow direction icon may be labelled with the possible outcomes of a decision. The tasks and decisions may be linked with a dotted line to a repository icon to denote the location of the items that are used while executing a task or decision. *Initial* tasks or decisions in process models are emphasized in bold. Using these icons, the process model of submitting of a paper to an academic conference can be represented as depicted in figure 3.7.

We like to point out the main differences between our process model and the same model as defined by Dur [1992]. Dur requires exactly *one* item class to flow through each process. We allow any number of item classes, including zero, to flow over the flow direction icon through the process model. In addition, we define a specific icon for decisions and require that the actor making the decision is specified as well.

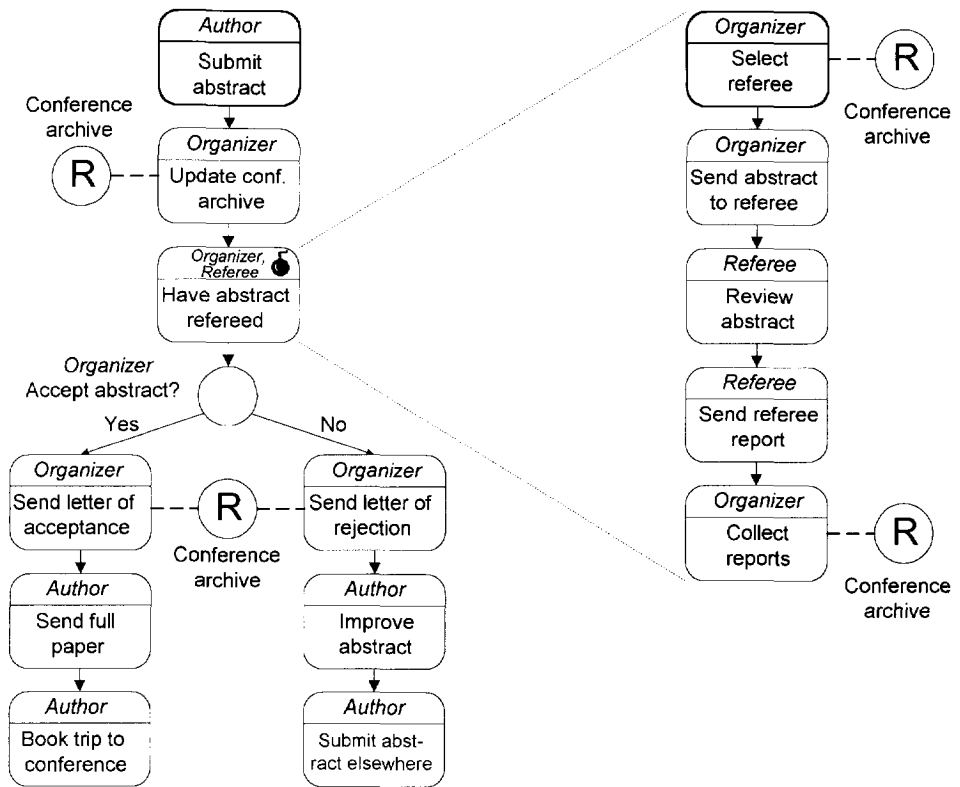


figure 3.7 Example of a process model.

3.4.4 The actor model

Based on [Bots 1989], the *actor model* is the third and final aspect model within the conceptual model of an organization's processes and their coordination. It aims to facilitate the analysis of the behavior of an individual actor by highlighting the interdependencies of the actions this actor has to perform and coordinate in order to achieve its goals.

An actor model consists of the same elements as a process model. However, the actor model represents all the actions of individual actors, whereas the process model represents the actions of all actors in the context of a specific process. We use the icons as depicted in figure 3.8 to build a graphical representation of an actor model. As in the process model, a bomb denotes a compound task or decision, and the flow direction icon may be labelled with the outcomes of a decision. *Initial* tasks or decisions in actor models are emphasized in bold. Using these icons, the actor model of the conference organizer in our example can be represented as depicted in figure 3.9.

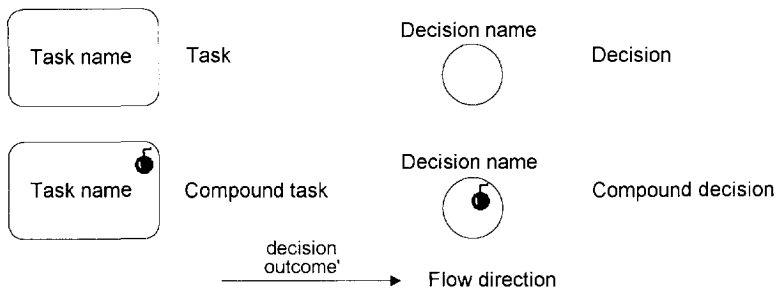


figure 3.8 Icons of the actor model.

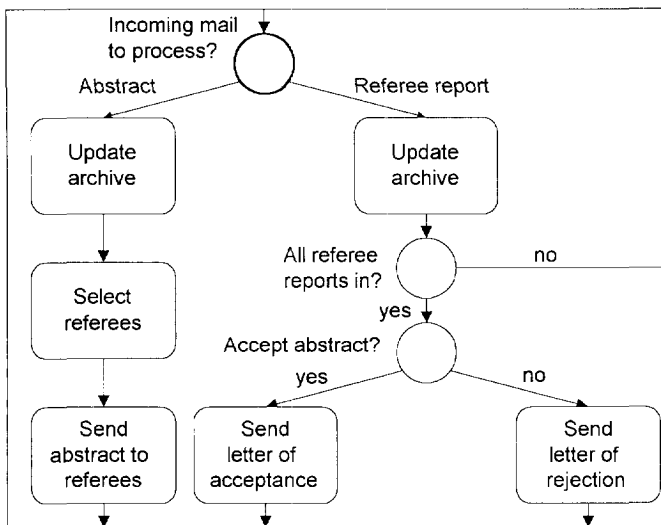


figure 3.9 Example of an actor model.

We point out that the main differences between our actor model and task structures as defined by Bots [1989] are that we allow compound decisions and that we do not differentiate between structured and unstructured tasks.

3.4.5 The empirical model

In this section we discuss the empirical model of a problem situation. While the conceptual model only describes the structure of the problem situation, the empirical model gives detailed specification of this situation. In the dynamic modelling approach, an empirical model consists of three elements:

1. *Simulation code.* We use a simulation language as a model construction language for the empirical model. A simulation model is constructed by translating the elements of the conceptual model into pieces of simulation code which can then be linked together. This translation of the object concepts from the conceptual model is discussed in this section.
2. *Empirical data.* In order to use the simulation model for a quantitative analysis of the modelled situation, empirical data such as process times, capacities of actors, and interarrival times of items, have to be incorporated into the simulation model. We will not elaborate any further on this data collection activity. It suffices to state that data collection can be guided by the definition of the object classes that are translated into the empirical model.
3. *Experimental design.* For the quantitative analysis of the modelled situation, the simulation model has to be used in a so-called experimental design. An experimental design consists of a number of treatments to which the simulation model is exposed [Sol 1982]. The exact design of the experiment depends on the hypotheses that have to be tested with the simulation model. We will come back to issues concerning the experimental design part of the empirical model in Section 3.4.7.

During the construction of (the parts of) the empirical model, several *reduction activities* may be carried out. Certain object classes can, for example, be left out or aggregated. Empirical data can be aggregated into stochastic distributions rather than be described fully deterministically. The time dimension of the model can be limited to certain intervals, see [Sol 1982].

Based on the empirical model, we define two aspect models. These models are the animation model, portraying the dynamic behavior of the empirical model (Section 3.4.6) and the experiment model, featuring the results of the treatments that the empirical model was subjected to (Section 3.4.7).

However, we will first demonstrate in this section how the simulation code of the empirical model can be constructed by translating the modelling concepts of the previous section into simulation code. Taking the three aspect models of the conceptual model as a starting point, the simulation code is constructed in two steps: First, the interaction model is translated. Then, the process model and actor model are translated. We start this section with a brief description of the simulation language we used.

Concepts of the simulation language

For our case studies we used SIMAN, a process oriented simulation language that allows processes to be executed in parallel [Pegden et al. 1990]. A simulation model in SIMAN consists of two parts: an experiment frame and a model frame. A *model frame* describes the physical elements of the modelled situation and their logical

interrelationships. An *experiment frame* describes the experimental conditions under which the model is to run and the model builder's specification of the elements that are to be manipulated by the model. A SIMAN simulation model can be thought of as a collection of items (called 'entities') flowing through a piece of simulation code. Items can be created and disposed of. Items can be sent to other pieces of code by conditional branch statements. Code that is restricted to one physical location can be assigned to a station. Items may be routed between stations. Stations may contain queue/resource combinations that act on items. Strong features of SIMAN are that it is possible to separate empirical data from the structure of the simulation model, that it supports statistical analysis, and that it has animation facilities including the possibility to switch between different animation views [Streng 1993; Wierda 1991].

Element	Description
ATTRIBUTE	Defines the attributes of an entity
QUEUE	Waiting space for entity, e.g. before seizing a resource
RESOURCE	Represents men or machines; when seized, holds the entities seizing it
STATION	Specifies locations in simulation model where entities can be sent

Block	Operands	Description
ASSIGN	Variable = Value	Assigns values to variables and attributes
BRANCH	Maximum, Randomstream, IF, Condition, Label, Primary, WITH, Probability, Label, Primary, ELSE, Label, Primary, ALWAYS, Label, Primary	Controls flow of entities through code
CREATE	Batchsize, Offset, Interval, Maxbatches	Creates entities
DELAY	Duration, StorageID	For time delays
DISPOSE		Disposes an entity
QUEUE	QueueID, Capacity, Blocklabel	Queues entities up for a resource
RELEASE	Resourcenam, Quantity	Releases a resource from an entity
REMOVE	QueueID, Label	Removes an entity from a queue
ROUTE	Duration, Destination	Transfers entities between stations
SEARCH	QueueID, StartPos, EndPos, Condition	Searches a queue for an entity
SEIZE	Priority, Resourcenam, Quantity	Lets an entity capture a resource
SIGNAL	Code, TotalLimit	Signals WAIT to release entities
STATION	StationID	Implements a location
WAIT	Code, WaitLimit	Holds entities until signalled

table 3.1 SIMAN elements and blocks.

For the construction of the simulation code of the empirical model derived from our modelling concepts, we use the SIMAN elements and blocks given in table 3.1. Elements are SIMAN concepts that specify the logical constructs to be used by the model. Elements are defined in the experiment frame. Blocks represent statements in the model frame that operate on entities and elements. There are many more elements and blocks but only the ones stated in table 3.1 are necessary for our purposes. For a more elaborate discussion of the SIMAN language we refer to [Pegden et al. 1990].

Basic structure: the interaction model

In this section we propose a method for converting each 'leaf' object in the interaction model into a piece of SIMAN code. With these pieces of code it is possible to simulate the basic structure of the interaction model. We do not claim that the description in this section represents the only or best method to convert a conceptual model into SIMAN code. Nor is this our objective as we simply intend to demonstrate that it is possible to perform the required conversion in a structured way.

Items can be directly translated to SIMAN entities. The attributes of an item can be specified in the experiment frame as the attributes of an entity. Since SIMAN only allows one type of entity to flow through the model, the SIMAN entity definition should be able to represent all possible items. One way to achieve this is by introducing an extra attribute 'item_type' and defining as much entity attributes as the items that contains the most attributes. This means that if, for example, there are two items defined, one with *three* and one with *five* attributes, the SIMAN attributes definition will specify *six* attributes (five plus one to denote the 'item_type').

We propose that *actors* are modeled with the station block and a queue/resource combination. An actor is seized by an incoming product, message or person and performs a number of actions. Since an actor has limited capacity, items will be queued up if the actor is busy. This results in the following code for an actor:

```

                                STATION,      Actorname;
                                QUEUE,        ActornameQ;
                                SEIZE:        ActornameR;
                                BRANCH, 1:    ALWAYS, ActornameAL;
ActornameRA  RELEASE:        Actorname;
                                BRANCH, 1:    ALWAYS, ActornameLI;
```

In this code, Actorname stands for the name of the actor. This name is extended with a Q to denote the queue, with an R to denote the resource, with AL to denote the address where the code for the actor's list of actions is located, with RA to denote the address where the item returns after the work process has been accomplished, and with LI to denote the address of the link interface where the actor sends the item to after performing its actions. When an actor shows autonomous behavior while there is no item to work with, the conversion becomes more complex. In this case, the resource representing the actor should be kept 'alive' by a so-called 'trigger' entity [Eijck and Vreede 1994].

We propose that *repositories* are also modeled with a station. The only (inherited) tasks of a repository are to send, receive, store and retrieve items. This can be done by the following code:

```

                STATION,      Repname;
                BRANCH, 1:    IF, Store, RepnameSTO
                                IF, Retrieve, RepnameRTR;
RepnameSTO      QUEUE,      RepnameQ: DETACH;
RepnameRTR      BRANCH, 1:  IF, NQ(RepnameQ) == 0, RepnameLI;
                                ELSE, RepnameRTRA;
RepnameRTRA     SEARCH,     RepnameQ: Searchcondition;
                REMOVE, J,   RepnameQ, RepnameLI: DISPOSE;

```

An item is received automatically when it arrives at the station block. Because items are not placed in any order in a queue, the queue must be searched first to find the item that is requested. The RepnameLI address points to the link interface where the repository sends a retrieved item.

After having completed a list of actions, the actor may send items away to link objects. An actor, however, may have more than one link, one for each actor or repository it is connected to. Hence, the destination of the item should be determined. The easiest way is to reserve an attribute in the item where its destination is stored. This may be altered during the execution of an action. It may also be stored in a global variable or determined instantly when sending the item. The following code can be used for sending an item to a *link* object:

```
ObjectnameLI    ROUTE:        0, Destinationexpression;
```

The duration is zero, because any time delay in transport is arranged for by the link object. The destination can be determined from the item, from a variable or local expression, or can have a standard value. The destination must be an actor or a repository. To model the *link* object we propose the following code:

```

LinknameST      STATION,      Linkname;
                BRANCH,1:    IF, Active, LinknameGO;
                BRANCH,1:    IF, NOT Active, LinknameNOGO;
LinknameGO      ROUTE:        LinknameDelay, Actor/Repname;
LinknameNOGO    DELAY,        LinknameDelay2;
                BRANCH,1:    ALWAYS, LinknameST;

```

This construct simulates the possible working or failure of the link object. The LinknameDelay variable stands for the time it takes to send an item to its destination. If a link is inactive, the item waits (LinknameDelay2) and tries again later. This construct can be replaced by other mechanisms, e.g. the item can be disposed of if it is not sent. The failure of links can be simulated by stochastic generators.

Process and coordination: the process and actor model

With the conversion of the interaction model into SIMAN code we can simulate the flows of items between the nodes. Now we turn to the way in which these items are processed and coordinated. This is reflected in the actions that are carried out by

actors. This means that these actions should be specified, i.e. modelled in SIMAN code. Upon arrival at the part of the code for an actor, the item is sent to the initial task or decision of the list of actions by the following code:

```
ActornameAL      BRANCH,1:      ALWAYS, Initial_task_or_decisionPR;
```

This may also be a conditional branch, determining the initial task from the attribute values of the incoming item. It might even be that items have to be copied to accomplish a work process.

Next, we have to specify the pre- and postconditions of the task and decision objects. Postconditions should be met automatically if the code for the execution of the task or decision is specified correctly. Preconditions may be expressions that operate on global variables or attributes. We propose two ways of implementing this. The first is by assigning values to attributes and variables that may be tested at a later stage (the conditions are implicitly specified by the values of attributes and variables). In this case, the simulation code for the task and decision object needs to be extended with code for precondition testing. This is done with a simple conditional branch block like the one below.

```
Task/DecnamePR  BRANCH,1:      IF, Preconditions met, Task/Dec.name:
                                      IF, NOT Precond. met, Task/Dec.nameWT;
Task/DecnameWT  DELAY:          Duration of test cycle;
                                      BRANCH,1:      ALWAYS, Task/Dec.namePR;
Task/Decname    .....          Rest of code for task or decision
                                      (see below)
```

The second way of implementing preconditions is by using the wait and signal construct. The above simulation code can then be replaced by the following:

```
Task/Dec.namePR WAIT:          Precondition expression (integer);
Task/Dec.name    .....          Rest of code for task or decision
                                      (see below)
```

Any time a condition is met, the wait block should be signalled by the following block:

```
Somewhereincode SIGNAL:      Precondition expression (integer);
```

We prefer the first construct because in this case the code for testing a precondition is only located at one position in the simulation model. The signal block, on the contrary, may be far away from the wait block, making the model quite complex and difficult to debug.

Finally, we turn to the simulation code for the task and the decision objects themselves. Tasks have two properties: They take time and they may influence attribute and

variable values. Decisions do not take time, although they may influence the sequence of tasks to be performed. In the code below, tasks and decisions are specified on the level of 'simple' tasks and decisions. Compound tasks and decisions are constructs that cannot be translated directly when the simulation model is constructed; they have to be expanded first.

We propose that a *task* is modeled by the following piece of simulation code:

```
Taskname      ASSIGN:      Variable1 = Value1:
                        :
                        VariableN = ValueN:
                        Attribute1 = Value1:
                        :
                        AttributeM = ValueM;
                DELAY:      TasknameDUR;
                BRANCH,1:    ALWAYS, Next_task_or_decisionPR;
```

The assignment of attributes may also include a destination attribute that determines the flow of the item after the task is completed. If a task has successor actions, the item is sent to the code for the precondition test of the next action. Otherwise, the last line of code is replaced by:

```
                BRANCH,1:    ALWAYS, ActornameRA;
```

A *decision* may be modeled by the following piece of simulation code:

```
Decisionname   BRANCH,N:    IF, WITH and/or ALWAYS constructs;
```

In the branch block, items may jump to the precondition test part of other tasks and decisions according to the decision rules or probabilities specified in the decision object. The decision rules are specified by a structure of conditional branch blocks. N represents the maximum number of items that exit the decision for one incoming item. This means that an item can be copied by the branch block.

3.4.6 The animation model

At present, animation is often a standard feature of simulation environments. A survey of simulation software by Swain [1991] shows that around 80% of the simulation packages have animation capabilities. The availability of animation tools, combined with the fact that animated simulations appear to make simulation models more 'use(r)-friendly', has driven the use of animation technology in more and more application domains. In this section we will elaborate on the first aspect model of the empirical model, the animation model. This model is aimed specifically at highlighting the *dynamics* of the problem situation as they are included in the empirical model. We propose that using animation *facilitates the different types of communication between*

the parties involved in a dynamic modelling study on organizational change. In this section, we first discuss the rationale for using animation to support communication. Then, we define what animation models are and discuss their implementation. Finally, we conclude with a summary of the potential benefits and disadvantages of animation.

Why use animation?

In the previous sections we described a set of modelling concepts and their transformation into simulation code. These concepts and the resulting simulation model enable us to analyze the (dynamic) characteristics of (the coordination of) organizational processes and to evaluate alternatives for organizational change. However, experience suggests that modelling or diagramming techniques and simulation languages may have some limitations, see e.g. [Anton et al. 1993; Kramer and Ng 1988; Meinsma et al. 1995; Spek et al. 1994]:

- Diagramming techniques may be able to model the dynamics of a situation, but the resulting diagrams are static in themselves. This makes it difficult to thoroughly understand and analyze the dynamic characteristics of the modelled processes.
- Diagramming techniques are often difficult to communicate to problem owners. Problem owners usually lack experience in reading and understanding diagrams. In addition, diagrams seldom resemble the problem situation visually.
- A simulation model can still be considered to be nothing more than a black box hiding the dynamics of the problem situation. It is difficult to investigate the modelled processes by only going through large sets of exogenous and endogenous data.

These limitations have in common the fact that they all concern *communicating and interpreting* a model of a problem situation. Every model, whether it is expressed in natural language or in highly formalized diagrams, is a translation of a model builder's mental model of a problem situation. Developing a model of a problem situation is not a goal in itself. Its main purpose is to *communicate knowledge about the relevant aspects of a problem situation to others*. In addition, models are needed (1) to accumulate knowledge of a problem situation for later use, (2) to record knowledge because the problem situation is too complex to consider as a whole, and (3) to get a better understanding of the problem situation by studying it from different perspectives.

In a setting where dynamic modelling is used to facilitate organizational change one can distinguish between three *roles* (or parties) that communicate knowledge about a problem situation. First there are one or more *problem owners*, who are confronted with the problem. Often the problem owners do not have the power to decide on changes with respect to the problem situation. This is a task of the *decision makers*. One or more *problem solvers* carry out the dynamic modelling study in order to solve the problem situation of the problem owners. Between each of the parties involved,

knowledge of the problem situation and aspects of the related models have to be communicated. In each of the problem solving activities, several activities can be identified in which communication is important:

1. *Problem conceptualization.* During the first activity, the problem owners communicate knowledge of the problem situation to the problem solvers. The problem solvers give feedback on their interpretation of the problem to the problem owners using different versions of a conceptual model.
2. *Empirical modelling and diagnosis.* For the construction of the empirical model, the problem owners supply empirical information of the problem situation to the problem solvers. The problem solvers use the empirical model output to create an error-free, verified simulation model. The output of the simulation model is shown to the problem owners for validation and diagnosis. The problem owners, decision makers, and (possibly) the problem solvers discuss the solution requirements.
3. *Construct alternative models and conduct experiments.* Based on the experiences with the simulation model, the problem solvers, problem owners, and decision makers suggest possible solutions for the diagnosed problems. These are implemented into the simulation model. The problem solvers report the results of the experiments with the simulation models of possible solutions to the problem owners and decision makers. They show the effects of the proposed changes.
4. *Choose solution.* The decision makers and problem owners discuss the results of the previous activity and the implications of certain solutions.
5. *Implement solution.* The problem solvers can use the simulation model to train the problem owners or others affected by the chosen solution.

In a dynamic modelling study, the models of the current and possible future situations are used as the vehicle of communication between the problem solver on the one hand, and the problem owners and decision makers on the other hand. Therefore, it is important that these models are easy to communicate. In this context, the cardinal requirement of a model is that *it closely resembles the mental model of the person it is communicated to* [Anton et al. 1993; Checkland 1981; Wierda 1991]. As was stated above, this appears to be difficult when models are expressed in diagrams or "bare" simulation models. We propose that animation models have the potential to resemble closely the mental models of both problem owners and problem solvers. Therefore, animation techniques are particularly suited to facilitating the communication processes between the parties involved in a dynamic modelling study. In the subsequent subsections we first discuss what animation models are and how we implement them. Then, we describe how dynamic modelling studies may benefit from the use of animation, and the drawbacks which may occur.

What is animation?

A compact description of an animation model is given by ElMaraghy [1982]. She describes an animation model as dynamical graphical output. Pegden et al. [1990] give a more distinct definition in which an animation model is required to show some resemblance to the modelled system. They define an animation model as a dynamic display of graphical objects that change position, shape or color on a static background. The changes correspond to alterations in the state of the related simulation model. The static background represents the environment in which the simulated system exists.

We follow Pegden et al. in stating that an animation model consists of a *static* and a *dynamic* part. The dynamic part depicts the dynamics of the empirical model. Apart from a representation of the environment of the empirical model, the static part may also include (abstractions of) elements of the empirical model itself, e.g. text or static artifacts of the simulation code, like the values of the exogenous variables. Therefore, we define an animation model as follows:

An animation model is a graphical model of an empirical model of a problem situation. An animation model's static part consists of (a) representations of the relevant objects in the problem situation and in the empirical model and (b) of a static background. Its dynamic part depicts the status changes over time of the problem situation's object instances that have been included in the empirical model.

Our definition of an animation model implies that we do not consider a visualization of *just* the dynamic aspects of an *empirical* model to be an animation model. Examples of this kind of visualization include graphical representations of Petri-net models, dynamic plots and histograms, and debugging tools for simulation models, see e.g. [Kramer and Ng 1988; Tsalgatidou et al. 1994]. In order to call a graphical model an animation model, we require at least some kind of representation of the problem situation to be present to enhance recognizability. These type of animation models we call *portrayal animation models*. Bell and O'Keefe [1987] distinguish between 'Visual Interactive Simulation' (VIS) and portrayal animation. However, in our opinion, there is no basis for this distinction. Interaction takes place with the simulation model, whereas portrayal animation is only used to represent the state of the model in a dynamic and recognizable way, interactive or not.

Implementing animation models

There are several applications or 'engines' that support the construction of an animation model. Most of these are part of a specific simulation environment, others are stand alone, see [Swain 1991]. There are two types of animation engines: postprocessors and concurrent processors. Postprocessors work with a file of output data that is generated by a simulation model. This data presents the status of the

objects in the simulation model over time. Concurrent processors display the behavior of the associated model in 'real time'.

Postprocessors have two advantages over concurrent processors. First, a postprocessed animation is usually faster as the execution of the simulation model itself is not claiming extra hardware resources. Second, postprocessed animations can be more flexible when studying a model because it may allow the modeller to 'rewind' and to 'replay' certain fragments of the animation. The main strengths of concurrent processors is twofold. First, it is not necessary to build an extensive file with animation data prior to studying the animation model. This is especially an advantage during model construction and debugging. Second, when executing an interactive simulation model, the results of interactions may be immediately visible allowing prompt follow-up interactions.

As we use SIMAN within the dynamic modelling approach as a construction language for building the simulation part of an empirical model, it is natural that we use Cinema to implement animation models in. Cinema is a concurrent, pixel based animation processor [Pegden et al. 1990]. In Cinema, it is possible to define various animation views, so-called 'layouts', that focus on different parts or behavioral aspects of the empirical model. Cinema supports free hand drawing to build the static part of an animation model. The dynamic part consists of representations of the status changes of the objects that are included in the empirical model.

The four SIMAN elements that we use to build the simulation code from the conceptual model, see table 3.1, can be represented as follows: Entities (as defined by the 'attributes' element) are depicted by user defined icons. The same goes for resources, but different icons can be used to represent the different states that a resource can be in, e.g. idle, busy, or preempted. Queues and stations can be positioned on the screen, but are not directly visible during the execution of the empirical model. Their presence is only visible when entities have to wait in a queue or pass by a station. Finally, the model builder can define which path an entity icon must follow when it is routed from one station to another.

Using animation models

In addition to increasing insight into the problem situation, the main goal of an animation model is to support and simplify communication between the parties involved in a dynamic modelling study. Below, we summarize the various ways in which this communication may be supported.

When a problem solver communicates with a *colleague problem solver*, animation models may stimulate the second problem solver to build a mental model of his own

that closely resembles that of the first problem solver. Therefore, the second problem solver may have a faster and better insight into the problem situation.

For a *single problem solver*, the use of animation models usually decreases the development time of the simulation code because of the enhanced debugging possibilities. With animation it may be quicker to find and localize mistakes than by going through output of traces or using a debugger. This is especially the case with mistakes caused by unexpected behavior of the model. In other words, animation aids verification and therefore enhances the quality of the model [Gipps 1986; Grant and Weiner 1986; Smith and Platt 1987].

An important strength of an animation model is its potential to bridge the gap between the mental model of the *problem owner* and that of the *problem solver* [Grant and Weiner 1986]. When a problem solver translates his mental model into an animation model, the result will be less abstract than diagrams or simulation code. The more an animation model visually corresponds to the problem situation, the easier the problem owner may communicate with the problem solver about the problem situation. This communication process may converge the mental models of both actors.

Facilitated communication is seen to increase understanding. As a result, problem owners may tend to participate more actively in the problem solving process. Their attention is drawn because the empirical model ceases to be a black box [Gipps 1986]. This may lead to improvements of the empirical model and to higher received benefits by the user [Smith and Platt 1987].

During the construction of a descriptive empirical model, using animation offers unique possibilities for *face-validity tests* [Grant and Weiner 1986]. Structural mistakes in the model or deviant behavior can be pointed out by the problem owners. This has proven to be very worthwhile, see e.g. [Dur 1992; Meel 1994; Streng 1993; Wierda 1991].

In the solution seeking activity, animation may help to point out *inadequacies of possible solutions* which the problem owners or decision makers want to test [Smith and Platt 1987]. With animation these shortcomings may be presented in such a way that the decision makers can understand and accept them.

In addition to the above, problem solvers and problem owners no longer have to go through large amounts of printed, numerical data [ElMaraghy 1982]. Animation models help focus on particular aspects of the system under investigation [Gipps 1986]. This may prove especially useful when presenting the results of a dynamic modelling study to a *decision maker*. In this situation, an animation model can illustrate the statistical results of such a study in an easy accessible manner.

In spite of the advantages of using animation models, there may be two drawbacks in using animation. First, some argue that there is a potential danger of basing decisions upon snap-shots instead of statistically valid results, see e.g. [Grant and Weiner 1986]. However, animation should never be seen as a replacement for statistical sound output data. Second, using animation models may result in a 'pacman effect' [Meel 1994]. This occurs when the attention is only focused on the animation model rather than on the empirical model or the results of the simulation experiments. To overcome this problem, simulation results can be presented separately, e.g. on slides, and questionnaires can be handed out to draw the attention of the audience to relevant issues [Meel 1994].

3.4.7 The experiment model

The second aspect model of the empirical model, the experiment model, specifically supports the quantitative analysis of a modelled situation. In order to perform such an analysis, the empirical model needs to be executed and placed within an experimental design. An experimental design consists of a set of treatments to which the empirical model is exposed [Sol 1982]. A *treatment* consists of [Sol 1982; Ören and Zeigler 1979]:

- The specification of input data.
- The collection of input data.
- The initialization conditions.
- The run control conditions.
- The specification of output data.

An experiment with an empirical model consists of a set of replications. Each replication consists of a collection of output data regarding a certain treatment [Sol 1982]. The experiment model may facilitate the understanding and the analysis of the modelled situation by presenting the collection of output data in a convenient way. For this purpose, the experiment model includes two parts. The first part contains the results of all replications with respect to the experimental design to which the empirical model was exposed. The second part contains the outcomes of the analysis of these results.

The experiment model may depict the results of each of its constituent replications, both individually and in aggregated form. The results of the replication consist of the collection of output data, using graphs, tables, or charts. These can, if possible, already be constructed during the execution of the empirical model. In this respect it may be worth combining part of the experiment model with the animation model to support analysis and communication. By depicting the progress of a replication, the experiment model's aim is threefold. First, the model may be used as a debugging aid to recognize

and localize modelling errors. Second, the model may be instrumental in identifying critical situations that occur in the modelled situation. Third, it may facilitate the understanding of the functioning of the model and the processes it represents.

The complete collection of output data in the experiment model can be analyzed using spreadsheets or statistical data analysis applications, such as SAS, SPSS, or SIMAN's Output Processor [Pegden et al. 1990]. The analysis results may be presented in various ways, such as tables, graphs, charts, or in the form of recommendations derived directly from the results.

In the case studies presented in the following chapters, we combine the presentation of the collection of output data with the animation models to create an integrated view of the execution of the empirical models. The results of the analyses of the output data are presented using a combination of various presentation forms. The analyses themselves are performed using spreadsheet applications.

4 Case 1: The Department of Unemployment Benefits

'All the planning charts and demolition orders have been on display in your local planning department in Alpha Centauri for fifty of your Earth years, so you've had plenty of time to lodge any formal complaint and it's far too late to start making a fuss about it now.'

- Douglas Adams

4.1 Introduction

In this chapter we describe the first case study with the dynamic modelling approach. This case study was carried out in a constructed, simplified case setting, the Department of Unemployment Benefits (DUB), based upon a real situation as described in [Dur 1992]. The aim of this case study was to develop, apply, and evaluate some of the dynamic modelling concepts that have been presented in the previous chapter.

The main reason for carrying out the first study in a constructed setting is that such a setting represents a comparatively safe and efficient way to explore, develop, and evaluate the possibilities for modelling and analyzing organizational coordination, see also [Sol 1982]. As this study represents the first case study of our research, we preferred investigating dynamic modelling in a 'controllable' setting before moving into a real life situation. A potential drawback of a constructed setting is that it lacks the richness of a real situation. The real life situation as described in [Dur 1992] is indeed very rich in terms of coordination issues: The main product of the DUB concerns services with respect to compensation payments. These services are the result of combining separate efforts and contributions of the various departments and employees. In other words, the coordination of activities within the DUB is of vital importance to the quality of the final product. The richness of the DUB case was maintained in the constructed setting by limiting the simplification process to the following points:

- Contacts of the DUB with other departments with respect to social security and tax payment issues were not taken into account.
- Contacts with former employers were not taken into account.
- Different skill levels of the DUB's clerks were not taken into account.

- The Investigation office was excluded from the case.

We now turn to the description of the problem situation at the DUB in more detail. Then we describe the conceptual model (Section 4.3), the empirical model (Section 4.4), the problem diagnosis (Section 4.5), and the alternatives for change (Section 4.6). We conclude the chapter by a reflection on this case study.

4.2 Problem situation

The Department of Unemployment Benefits (DUB) of the Dutch Ministry of the Interior administers regulations set up to provide monthly compensation payments for people who involuntarily lost their job as a civil servant. The DUB's main activities are:

1. *To judge new applications for unemployment benefits.* Depending on the documents provided by the applicant, a regulation clerk may decide to accept or reject the application straight away, or to ask for additional information. In most cases, the clerk will check with the government agency the applicant claims to have worked for. If the benefit claim is in order, the clerk sets up a scheme according to which the payments will be made. After verification and authorization, this scheme and other relevant data are entered into the DUB's computer system. All documents are stored in a client file in the DUB's archives.
2. *To determine the correct amount for each monthly payment to registered clients.* This is by no means a trivial activity. The payment scheme is not the only determinant, as the amount may be affected by changes in for instance a client's personal situation (illness, divorce etc.). More important is the fact that clients have to send in a monthly *income statement* indicating the exact amount of income derived from other sources than the DUB. Such additional income is deducted from the amount indicated by the payment scheme. If no income statement has been received before the 15th of the month, the client's record is locked. On the 18th, the computer system processes all unlocked client records in order to produce a tape for the bank with payment data. The DUB is obliged to process all income statements arriving between the 15th and 18th. This involves time consuming manual calculations otherwise done by the computer system.

To meet its work load, the DUB employs 80 people, distributed among three offices (*Regulation*, *Special Cases*, and *Verification*) and two supporting offices (the *Mail Room* and the *Archives*), see figure 4.1.

In principle, the Regulation office processes all applications, mutations and income statements. However, some applications require specific judicial expertise and are

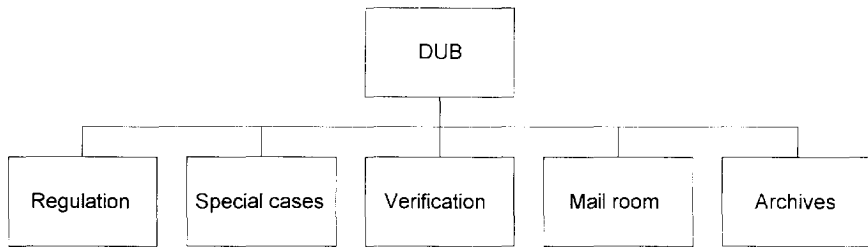


figure 4.1 Organization chart of the DUB.

forwarded to a clerk of the Special Cases office. General requests for information are answered by the Regulation office. Requests for information on specific clients are processed by the Special Cases office. As there are reasonably large sums of money involved (the DUB has an annual turnover of 400 million Dutch guilders), all of the decisions made by Regulation clerks or Special Cases clerks are checked by one of the Verification clerks. The Verification office is also responsible for producing and sending the tape for the bank with payment data.

All written correspondence passes through the Mail Room: incoming letters are opened, sorted, and subsequently delivered to the appropriate office, while outbound mail is collected, stamped and sent. Most of the vast flow of paperwork entering the organization ends up in client files stored in the DUB's extensive archives. The Archives office is responsible for the storage and distribution (upon request) of client files. In addition, archive employees make regular 'rounds' through the DUB offices, picking up and delivering files as they go.

The way the DUB currently operates is not satisfactory. The main perceived problems in the DUB are summarized in table 4.1.

Problem description

Current service level is unacceptable:

- Many income statements are processed too late
- Many mutations are processed too late

Increasing number of clients

Risk of vicious circle

Reduced work force

table 4.1 Perceived problems in the DUB.

Even when accounting for late client response, a substantial number of the income statements can not be processed in time and therefore result in late payments. In addition, it appears that many mutations are processed after the deadline. This either results in incorrect amounts or in payments that bounce due to inaccurate client data.

On the whole, the management of the DUB fears that the *service level* of the DUB is unacceptable. The service level is defined as the percentage of correct and timely benefit payments for which the information needed was available to the DUB at the proper time. The DUB management aims for a service level of at least 95 percent.

Due to the reorganizations within the Dutch government service, the DUB must seriously reckon with a disproportionate *increase* in the number of benefit applications, in the number of resulting income statements, and in the number of potential mutations. Therefore, the DUB may expect an increasing number of income statements and mutations processed too late, requiring manual processing. Given that manual processing puts an extra burden on the available capacity, the DUB risks entering a *vicious circle*: more capacity loss → more income statements and mutations processed too late → even more capacity loss → etc.

Finally, governmental regulations and restrictions prevent the DUB from solving the capacity shortage by hiring more personnel. Furthermore, the ongoing reorganizations within the government service will likely require the DUB to reduce its personnel force by about 30 percent the coming five years.

The DUB management is, therefore, very much interested in exploring reorganizational options and possibilities of modern IT to enhance the productivity of the department. In order to analyze and handle the problems mentioned above, we carried out a dynamic modelling study into the DUB situation. In the next section, we describe the conceptual model of the DUB. The subsequent sections respectively address the empirical model and the problem diagnosis, the alternatives to change the current situation, and the reflection on this case study.

4.3 The conceptual model

In this section we describe the conceptual model of the DUB case. First, we discuss some of the object definitions that together form the conceptual model. Then, we focus on specific (coordination) aspects by presenting the interaction, process, and actor model of the conceptual model. During the discussion of these three models, we will make use of the taxonomy of coordination mechanisms that was presented in table 1.1.

4.3.1 Object definitions

There are a substantial number of documents, forms, and other paperwork that find their way to and through the DUB. These are modelled as items, most in the form of messages. As will be shown, the item definitions below do not contain any attributes concerning information content. This is due to the fact that the actual processing of

items will not be modelled in detail. In addition, the actors' decision rules with respect to which task to perform on items will not be modelled in terms of items' attribute values, but in terms of probabilities. The only attribute included in the definitions refers to the office in the DUB where the item has to be sent to after a certain actor has dealt with it.

To begin with, there are *applications* from potential clients for an unemployment benefit. Applications may be retroactive if they concern past periods. The applicant receives an *application outcome* which may be a 'grant' or a 'reject'. If the DUB does not have sufficient information to decide on granting an application, a *request for extra information* is issued. As a consequence, the client will have to send in a new application.

object class APPLICATION
is a MESSAGE

attributes
next office

object class APPLICATION
OUTCOME
is a MESSAGE

attributes
next office

object class REQUEST FOR
EXTRA INFORM.
is a MESSAGE

attributes
next office

A granted application results in a *transaction form* on which information is put to update the automated client database. The granted application together with the transaction form is verified before the client database is actually updated. This verification may result in an *error form*, implying that either the application or the transaction form have to be corrected. Both have to be resubmitted for verification.

Clients who's personal situation changes, send in a *mutation form*. If a mutation influences a client's payment schema, the mutation information is put on the same sort of transaction form as with granted applications. Again, the transaction form is verified, which may result in an error form.

object class TRANSACTION
FORM
is a MESSAGE

attributes
next office

object class ERROR FORM
is a MESSAGE

attributes
next office

object class MUTATION FORM
is a MESSAGE

attributes
next office

Each month, clients have to send an *income statement* to the DUB. This is used to determine the exact amount of the benefit they will receive that month. If the income statement is received in time and results in a change in the payment schema, a transaction form is created and send off for verification. As usual, this may result in an error form. If the income statement is received too late, a *payment form* is created manually and send to be verified. This too, may result in an error form.

object class INCOME STATEMENT is a MESSAGE	object class PAYMENT FORM is a MESSAGE
attributes next office	attributes next office

Clients or potential clients often ask the DUB for information. This is done by sending in forms, called *information requests*, or by making a *telephone call* in person. In reply to these information requests, the DUB sends out *information packages*.

object class INFORMATION REQUEST is a MESSAGE	object class TELEPHONE CALL is a PERSON	object class INFORMATION PACKAGE is a MESSAGE
attributes next office	attributes next office	attributes next office

Every month each client receives a *financial statement*, stating the gross and net amount of the benefit he or she receives.

All personal data of the client and a history of its mutations are stored in *client files*. These files are consulted during the various activities with respect to applications, income statements, and requests for information. Clerks have to send a *client file request* in order to have a client file delivered on their desk.

object class FINANCIAL STATEMENT is a MESSAGE	object class CLIENT FILE is a MESSAGE	object class CLIENT FILE REQUEST is a MESSAGE
attributes next office	attributes next office	attributes next office

The items we just described are handled by a number of item processors. We first describe the actors in the DUB and then the links. There are no repositories in the DUB's current situation. (The genders used to illustrate the actors are random). Note that we do not define identifying attributes for the actors, as we will not need to differentiate between an office's individual actors in the empirical model.

The first actor we define is a DUB *client*. A client sends applications, mutation forms, income statements, and information requests to the DUB's Mail Room. Also, a client may want to make inquiries by telephone. From the DUB, a client receives application outcomes, information packages, financial statements, and requests for extra information with respect to the applications the client sends in.

object class CLIENT
is an ACTOR

attributes

actions

receive document
send application to Mail Room
send mutation form to Mail Room
send income statement to Mail Room
send information request to Mail Room
make telephone call

object class MAIL ROOM CLERK
is an ACTOR

attributes

actions

receive mail
sort mail
send application to Regulation office
send mutation form to Regulation office
send income statement to Regulation office
send information request to Special Cases office
send application outcome to client
send request for extra information to client
send financial statement to client
send information package to client

object class REGULATION CLERK
is an ACTOR

attributes

actions

receive document
what type of document?
is application retroactive?
is income statement on time?
new client?
need extra information?
process application
process mutation form
process income statement
create client file
create transaction form
create payment form
correct application
correct transaction form
correct payment form
send application to Special Cases office
send client file to Archives
send client file request to Archives
send application to Verification office
send mutation form to Verification office
send income statement to Verification office
send transaction form to Verification office
send payment form to Verification office
send application outcome to Mail Room
send request for extra information to Mail Room

A Mail Room clerk receives mail from outside the DUB, consisting of applications, mutation forms, income statements, and requests for information. He sorts this mail and sends it to the appropriate offices. The Mail Room clerk also sorts mail from the other offices and conveys this to (potential) clients.

A Regulation clerk processes applications, mutation forms, and income statements. He sends retroactive applications to the Special Cases office. While processing items, he may request client file from the Archives, or create client files himself and send them to the Archives. Regulation clerks also send transaction forms and payment forms to the Verification office. As a consequence, they may have to correct these forms. Finally, a Regulation clerk sends application outcomes and requests for extra information to the Mail Room, that will convey them to the clients concerned.

A clerk at the Special Cases office processes retroactive applications. As a result, she may send transaction forms to the Verification office and new client files to the Archives. She also corrects incorrect transaction forms sent back by the Verification

office. For processing requests for information and telephone calls, a Special Cases clerk may request client files from the Archives. Application outcomes and information packages are sent to the clients via the Mail Room.

object class SPECIAL CASES CLERK
is an ACTOR

attributes

actions

receive document
what type of document?
general or personal information request?
process application
process request for information
handle telephone call
create client file
create transaction form
correct application
correct transaction form
send transaction form to Verification office
send application to Verification office
send client file to Archives
send client file request to Archives
send application outcome to Mail Room
send information package to Mail Room

object class VERIFICATION CLERK
is an ACTOR

attributes

actions

receive document
what type of document?
inconsistencies found?
check application with transaction form
check mutation form with transaction form
check income statement with transaction form
check income statement with payment form
process financial statement
send application to Regulation office
send application to Special Cases office
send mutation form to Regulation office
send income statement to Regulation office
send error form to Regulation office
send error form to Special Cases office
send financial statement to Mail Room
send transaction form to Mail Room

Verification clerks have to compare applications, mutations, and income statements with the associated transaction forms and payment forms. After doing so, they send these forms back to their originator, accompanied by an error form in case inconsistencies were found. If no inconsistencies are found, the transaction forms are kept and, via the Mail Room, are sent to the Computer Center at the 15th of the month. Finally, Verification clerks process financial statements and send these to the mail room to be passed on to the clients.

The *Archives clerk* spends most of her time storing and retrieving client files, after receiving client files and client file requests respectively. In addition, the Archives clerk is responsible for making 'rounds' through the DUB, collecting and delivering forms, files, and other mail as she passes through the various offices. In other words, when an actor from one of the previously discussed four offices sends an item to a colleague, this item is placed so that the Archive clerk may pick it up.

The last actor in the DUB case is the *Computer Center*. Although this actor is not part of the DUB itself, it plays an important role in the functioning of the department. Its main task is to process incoming transaction forms and provide the Verification office with the financial statements of the payments that will be done.

object class ARCHIVES CLERK
is an ACTOR

attributes

actions

received document?
what type of document?
time for round?
more offices to visit?
store client file
retrieve client file
collect appropriate document
deliver appropriate document
go to next office

object class COMPUTER CENTER
is an ACTOR

attributes

actions

receive transaction form
process transaction form
create financial statement
send financial statement to Mail Room

There are three classes of links in the DUB. The first involves the links between the offices and the Archives clerk. As may be obvious from the previous descriptions, actors from different offices do not interact directly, but via an Archive clerk. The second link involves the normal mail by which the DUB interacts with the Computer Center and its clients. The third and final link is the telephone, that the clients may use to request information from a Special Cases clerk. Note that this is a uni-directional link; only a client calls the Special Cases clerk, not the reverse.

object class OFFICE_LINK
is a LINK

attributes

starting node (Regulation, Verification, Special Cases, or Mail Room clerk)
ending node (Archives clerk)
bi-directional

actions

route all defined messages

object class MAIL
is a LINK

attributes

starting node (Mail Room)
ending node (Computer Center, Client)
bi-directional

actions

route transaction form
route financial statement

object class TELEPHONE
is a LINK

attributes

starting node (Client)
ending node (Special Cases clerk)
uni-directional

actions

route telephone call

The actions of the actors can be further described with 'task' and 'decision' object definitions. To conclude this section, we give an example of each type of object definition. These example include the decision whether an application is retroactive by a Regulation clerk and the processing of an (retroactive) application by a Special Cases clerk.

object class	IS APPLICATION RETROACTIVE?	object class	PROCESS RETROACTIVE APPLICATION
is a	DECISION	is a	TASK
attributes		attributes	
required items:	application	required items:	application
preconditions:	-	preconditions:	-
postconditions:	nature of application and next action known	postconditions:	next office of application = Verification office
next actions:	process application ('normal'), send application to Special Cases office ('retroactive')	next actions:	create transaction form delay
decision rules:	normal (90%) retroactive (10%)	att. assignm.:	next office: =Verification office

4.3.2 The interaction model

Looking at the structure of the DUB as a whole, we see that the department coordinates its activities according to a *functional hierarchy* [Malone 1987], where there is a specialization of functions, objective qualifications and qualities of office, acting only according to a fixed set of rules, and a hierarchy of authority, see [Thompson et al. 1991]. As a consequence, the DUB's (functional) offices continuously have to exchange pieces of work that have to be further processed by other offices. This situation becomes clear when we look at the interaction of items between the actors in the DUB, see figure 4.2. (Note that the links in this model are not labelled with the class name that defines them in order to minimize the information overload).

The interaction model of the DUB clearly shows the central role of the Archives clerk in the interaction of items between actors that represent the other offices. There is no direct contact between clerks from the Mail Room, the Regulations office, the Verification office, or the Special Cases office. In fact, the Archives clerks provide the interaction medium by which coordination takes place. The only direct personal contact is when a client calls a Special Cases clerk to request information. Hence, the actors in the DUB coordinate their efforts by sending each other work that must be self explaining. How the actors further deal with this work is described in the next section.

4.3.3 The process model

When we look at the way the DUB's actors coordinate their activities in more detail, we have to examine the interdependencies between these activities. In this sense, it appears that *standardization* is the predominant coordination mechanism in the DUB. Rules for the performance of each activity are defined, so that the result of an activity yields the correct input for the next activity. Various types of standardization are applied: standardization of *work processes* (the Regulation office, the Mail Room, and the Archives strictly adhere to predefined operating procedures), standardization of

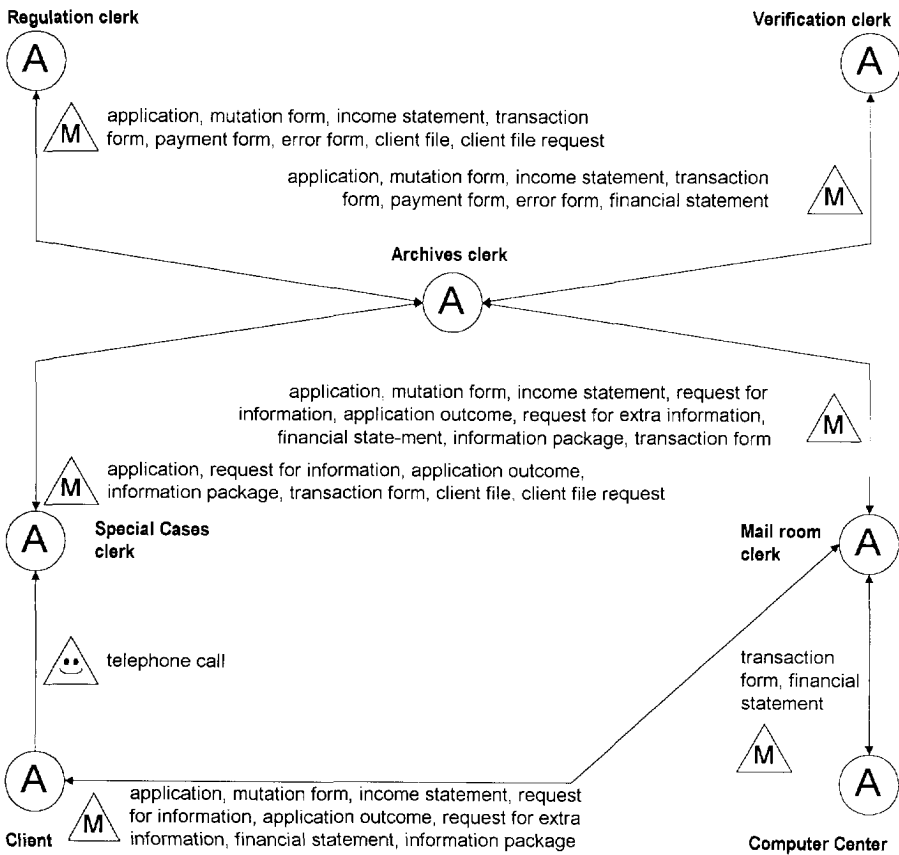


figure 4.2 The interaction model of the DUB.

work outputs (the DUB makes heavy use of standard forms for file requests, payment schemes, etcetera), and standardization of *worker skills* (the Verification and Special Cases offices employ professional clerks). The standardized work process of a Special Cases clerk with respect to providing information to clients is depicted in figure 4.3.

Taking a closer look at the coordination *processes* in the DUB, we see that there is no anticipatory coordination. The DUB does not *plan* the processing of items. On the contrary, the clerks perform their activities whenever they have an item that they can process. In other words, the coordination processes at the DUB are *adaptive*. Specific coordination activities only take place when exceptional events happen, such as items arriving late, being incorrect, or lacking sufficient information to perform the required activity. This is illustrated in the process model for handling applications by the DUB, see figure 4.4.

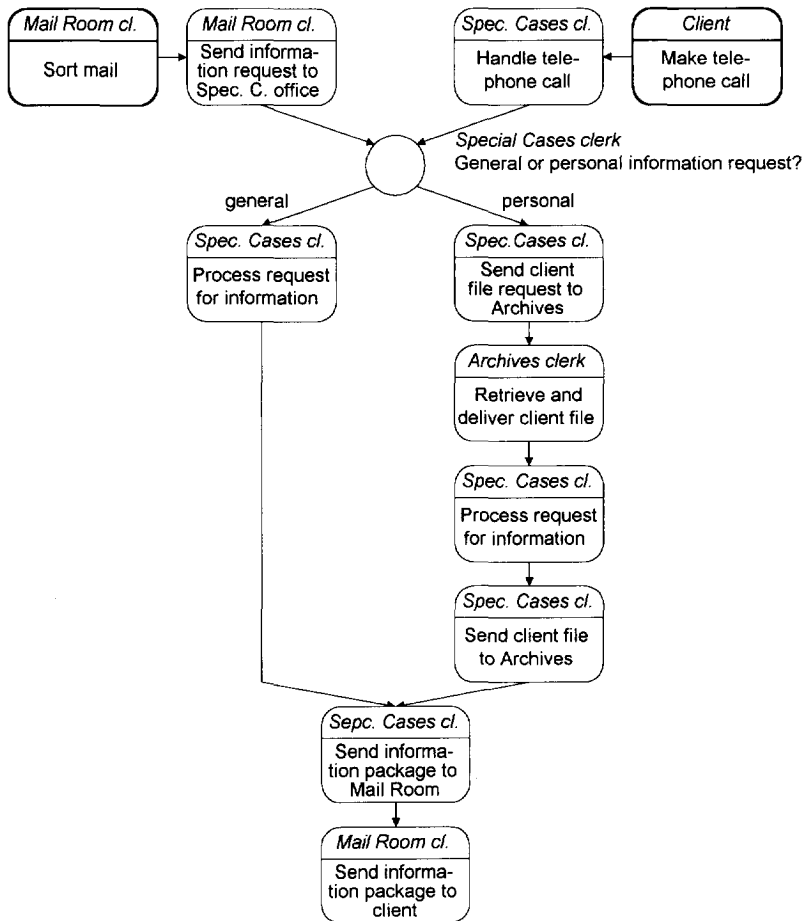


figure 4.3 Process model for providing information to clients.

Finally, there are no *review* coordination processes in the DUB. The DUB has no learning mechanisms implemented to evaluate the way the work in the department is carried out and use this knowledge to improve the way of working.

4.3.4 The actor model

The third and final model highlights the way in which individual actors carry out their activities. Like the process models in the previous section, the actor model of the Archives clerk in figure 4.5 exemplifies the standardization of work processes in the DUB. The Archives clerk performs its required activities without having to consult with other actors. Similar models can be drawn up for the other clerks in the DUB.

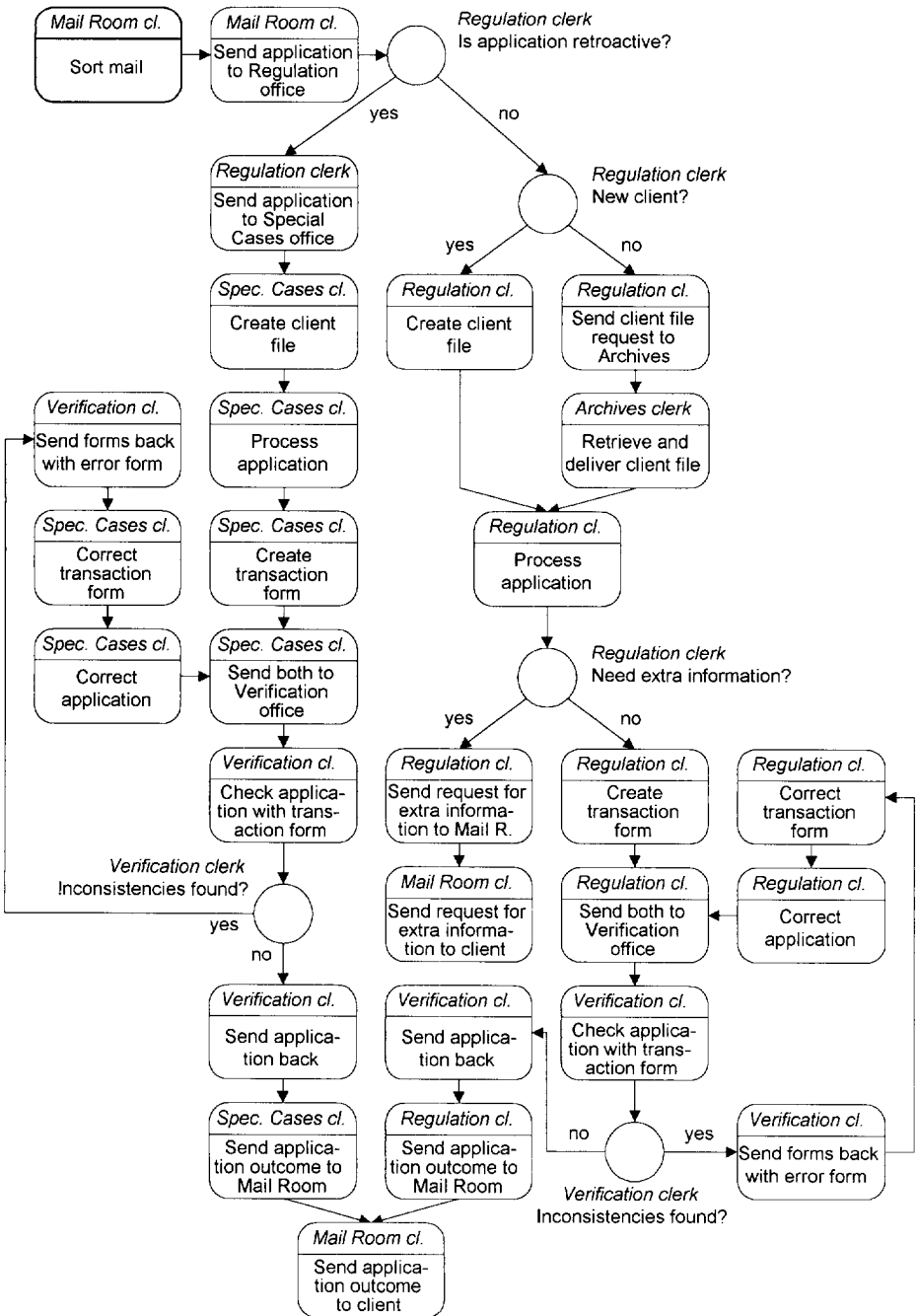


figure 4.4 Process model for handling applications.

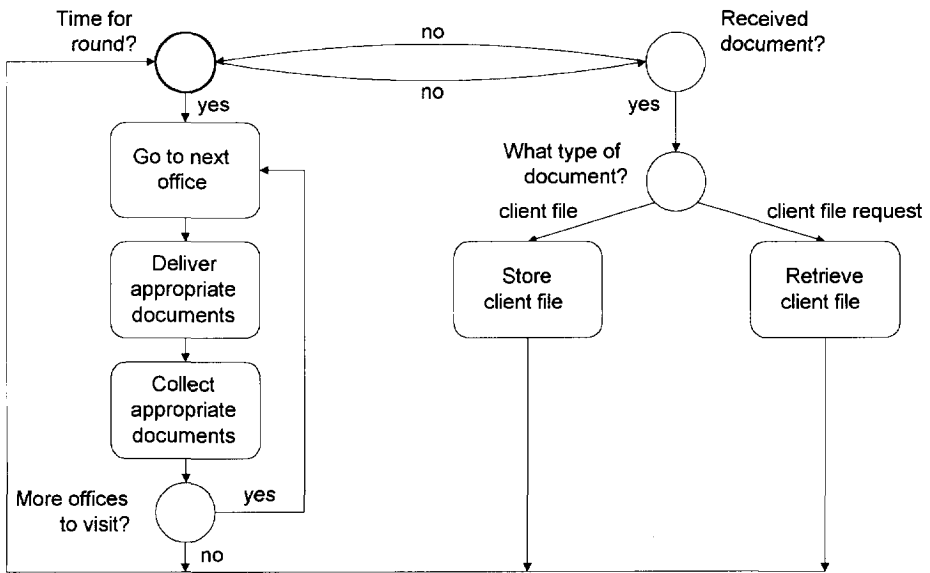


figure 4.5 Actor model for the Archives clerk.

4.4 The empirical model

Based on the conceptual model, we further specify the DUB situation by building an empirical model. This model will be used to analyze and diagnose the DUB situation. As stated in Chapter 3, the empirical model consists of three parts: simulation code, empirical data, and an experimental design. The SIMAN simulation code was not constructed according to the transformation procedure as this procedure was not developed until the second case study. The empirical data and experimental design are the subject of Section 4.4.1 and 4.4.2 respectively. Next, the animation model of the empirical model is described in Section 4.4.3. We conclude this section with a discussion of the verification and validation of the empirical model. The experiment model is discussed in combination with the diagnosis of the problem situation in Section 4.5.

Before turning to the description of the various elements of the empirical model, we first point out which reductions have been made during the construction of this model:

- In the simulation code, clients are modelled as document and telephone call generators instead of actors with autonomous behavior.
- The Computer Center is, like clients, modelled as a generator of a batch of financial statements.

- It is assumed that every day an equal number of items is generated, except for income statements.
- Instead of creating separate simulation elements for each instance of an actor in the DUB, all actor classes have been translated into single resources with a number of units equal to the number of actor instances in the respective offices.
- The capacity utilization of all actors in the DUB is set at 75 percent. This percentage is used to calculate the process times of the DUB's various activities and to calibrate the empirical model.
- The DUB is only simulated between 8:15 and 17:15.

4.4.1 Empirical data

The empirical data included in the empirical model is ordered in the following categories: data on the arrival of items at the DUB, capacity of the item processors, process times of the various items, and probabilities with respect to types of items and courses of action. Each of these categories is discussed below.

Item	Arrivals per month	Arrival distribution
Applications	1,100	Exponential(10.64)
Mutations	3,750	Exponential(3.12)
Income statements	7,000	see figure 4.6
Written requests for information	1,000	Exponential(10.80)
Phone calls	7,600	Exponential(1.29)

table 4.2 Data on the arrival of items at the DUB.

Each month, the DUB receives over 20,000 items that have to be processed. The number of arrivals of each item per month is given in table 4.2. Based on these numbers, the arrival of the various items at the DUB is modelled as follows:

- Applications, mutations, and written requests for information are 'posted' and then delivered to the DUB by mail, which arrives at the Mail Room at about 8:45 and at 13:15 each working day. The interarrival times of these items at the *mail box* are modelled with exponential distributions (mean interarrival time in minutes in brackets).
- Telephone calls arrive at the Special Cases office according to an exponential distribution with a mean of 1.29 minutes.
- As income statements need to be 'in' by the 15th of each month, they do not arrive regularly: 70 percent arrives before the 10th, 15 percent between the 10th and the 15th, and other 15 percent during the remainder of the month. During these periods, the arrival behavior can be modelled with an exponential distribution. This leads to the distribution of the arrival of income statements as depicted in figure 4.6.

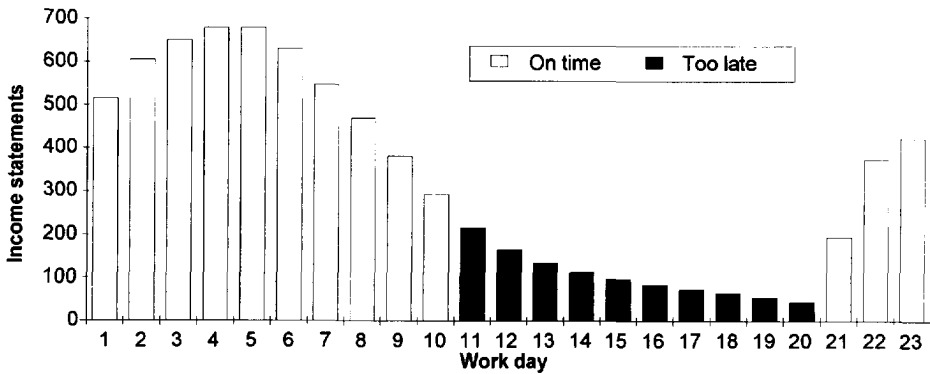


figure 4.6 Number of income statements arriving per work day.

The clerks in the DUB work from Monday to Friday, from 8:30 to 12:30 and from 13:00 to 17:00. Although clerks do not accept new tasks after 12:15 and 16:45, they do finish the task they are working on before leaving the office (for lunch). The DUB employs 80 people which are distributed over the various offices as follows: the biggest office, the Regulation office, has 43 employees. The Special Cases office and the Verification office have 12 and 9 clerks respectively. The two supporting offices, the Mail Room and the Archives, employ 4 and 12 clerks respectively.

The last set of empirical data concerns the processing times of the various tasks and the probabilities associated with the decisions the actors in the DUB have to take. With reference to the process models in figure 4.3 and figure 4.4, a number of examples of these processing times and decision probabilities are given in table 4.3.

Actor	Task	Mean time to complete
Mail Room clerk	Sort mail	1.8 min
	Send application to Regulation office clerk	0.6 min
Regulation clerk	Create client file	10.0 min
	Process application	40.0 min
	Correct transaction form	5.0 min
Special Cases clerk	Process information request	7.5 min
	Send client file request to Archives	0.6 min
Verification clerk	Check application with transaction form	7.4 min
	Send forms back with error form	5.6 min
Actor	Decision	Probabilities of outcomes
Regulation clerk	Is application retroactive?	yes 10%, no 90%
	Need extra information?	yes 5%, no 95%
Special Cases clerk	General or personal information request?	general 83%, personal 17%
Verification clerk	Inconsistencies found?	yes 5%, no 95%

table 4.3 Example processing times and probabilities of actions of DUB actors.

The decision probabilities were based on the data from the case as reported by Dur [1992]. The processing times were calculated such that the capacity utilization of the various clerks was 75 percent. This implies that the clerks in general are quite occupied, but do not have to work under extreme pressure. All processing times were implemented using triangular distributions, with a minimum and maximum of 50 percent and 150 percent of the mean respectively. A triangular distribution was preferred over a normal distribution as the latter one may introduce negative processing times.

4.4.2 Experimental design

The experimental design part of the empirical model consists of the treatment that will be used to analyze the current situation at the DUB. As stated in Chapter 3, a treatment consists of a specification and collection of input data, and of a specification of initialization conditions, of output data, and of run control conditions. The specification and collection of input data have been discussed in the previous section. The specification of the other treatment elements is discussed below.

For each run of the empirical model, the DUB is considered to be empty; there are no items that need to be processed. To determine the *start up time* for each run of the empirical model, a graph was plotted of the number of financial statements that arrive at the DUB each month when the Computer Center has processed them, see figure 4.7.

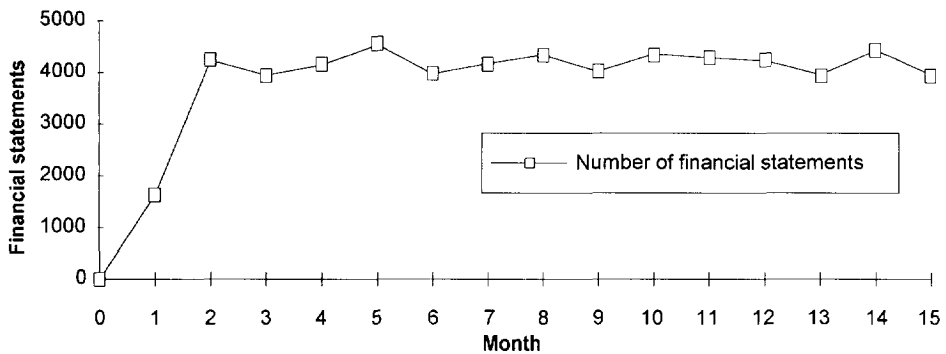


figure 4.7 Start up time.

Fluctuations in the DUB's workload are caused by the deviant arrival behavior of income statements. The workload in the DUB determines the amount of income statements that are processed before the 15th, which, in turn, determines the amount of transaction forms created each month. As each transaction form results in a financial statement, the number of financial statements can be used to assess the initial fluctuations of the simulation. The start up time is set equal to the month where major fluctuations have disappeared: 3 months.

The run control conditions are twofold: the run length and the number of replications. The *run length* can heuristically be taken equal to three times the maximal cycle time [Sol 1982]. The *maximal cycle time* in the DUB is one month. The *run length* is set to 12 months, one year, to be on the safe side.

In order to determine the number of replications, we first have to define the output data. Two types of *output data* are specified. The first type concerns the throughput times of the various items that are sent to the DUB by its clients: applications, mutations, income statements, information requests, and telephone calls. The second type relates to the service of the DUB, i.e. making correct compensation payments to its clients. For this purpose, we define the output variable 'service level' which is the percentage of income statements and mutations that are received before the 18th and are processed in time.

Finally, the *number of replications* was determined by performing a number of initial replications. Based on the outcome of these replications, the number of additional replications is calculated for each of the (dependent) output variables to achieve an acceptable confidence level. An elaborate description of the procedure to determine the minimal number of replications can be found in [Meel 1994, p.209]. The results of the procedure are described in table 4.4. The desired confidence interval is subjectively taken as half the difference between the highest and lowest value of the output variable. Based on these results, the number of replications is set to 7.

Output variable	Desired confidence interval	Minimal number of replications ($\alpha=0.05$)
Application throughput time	0.90 hours	7
Retroactive application throughput time	0.90 hours	7
Mutation throughput time	0.60 hours	6
Income statement on time throughput time	0.40 hours	7
Income statement too late throughput time	0.80 hours	7
Request for information throughput time	0.40 hours	7
Telephone call throughput time	0.05 hours	7
Service level	0.30 percent	6

table 4.4 Determination of the number of replications.

4.4.3 The animation model

In order to portray the dynamic characteristics of the DUB situation, an animation model was designed. This model was based on the simulation code of the empirical model and implemented in Cinema. The animation model of the DUB consists of seven animation screens: an opening screen, an overview screen, and a screen for each DUB office. Each screen shows the status and activities of the actors as well as their workloads and output in detail. Screenshot of the overview screen and of the Special Cases office are depicted in figure 4.8 and figure 4.9.

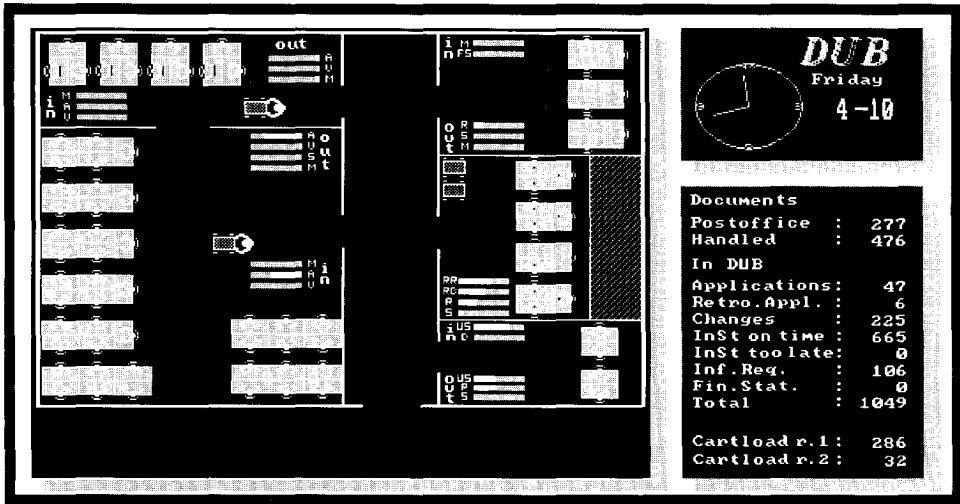


figure 4.8 Animation screen giving an overview of the DUB.

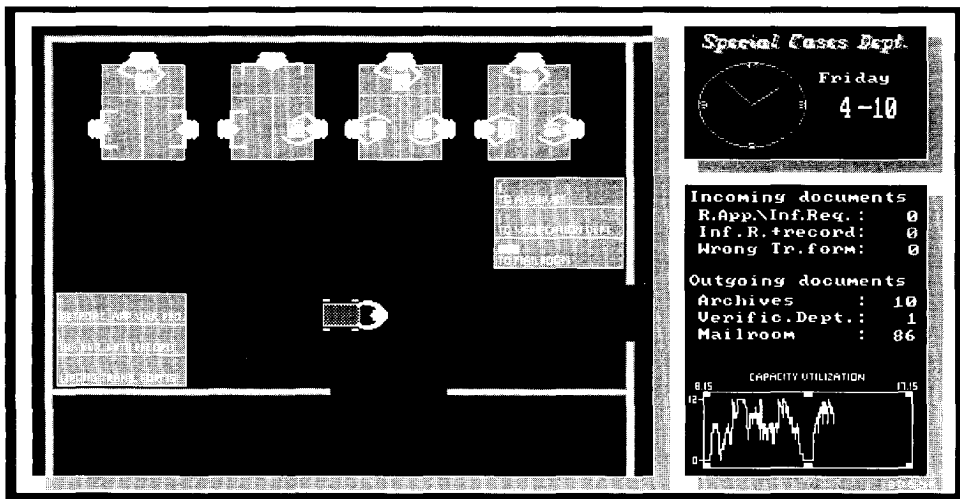


figure 4.9 Animation screen depicting the Special Cases office.

As the DUB involves a constructed case situation, the animation model could not be used to support communication activities between problem situation stakeholders and model builders. However, the animation model was instrumental in the verification and structural validation of the empirical model, as well as in the diagnosis of the current situation at the DUB. This is discussed in more detail in the next sections.

4.4.4 Model verification and validation

The *correctness* of the empirical model was *verified* using 'tracing' techniques [Meel 1994; Pegden et al. 1990]: individual items were generated and their flow through the model was traced. This allowed us to check whether the items underwent the required transformations and whether they underwent these in the correct order. The animation model facilitated the tracing process by clearly visualizing the flow of each item. During the verification of the model, a number of modelling errors were corrected.

As the empirical model is a representation of a constructed case situation, it was not possible to perform a *replicative validation* for lack of empirical data to compare to the model output. The *structure* of the model was *validated* by presenting the animation model to a number of people that had either been involved in the original DUB study [Dur 1992], or had expertise in the area of administrative processes in governmental organizations. These presentations did not result in any structural critique that required modifications in the model.

4.5 Problem diagnosis

The experiment with the empirical model of the current situation at the DUB consists of a set of seven replications under the treatment described in Section 4.4.2. The result of the replications in terms of the output variables is depicted in table 4.5.

Output variable	Mean	Standard deviation
Application throughput time	29.80 hours	0.60 hours
Retroactive application throughput time	34.56 hours	0.62 hours
Mutation throughput time	31.42 hours	0.57 hours
Income statement on time throughput time	17.95 hours	0.29 hours
Income statement too late throughput time	39.10 hours	0.62 hours
Request for information throughput time	13.81 hours	0.25 hours
Telephone call throughput time	3.35 hours	0.03 hours
Service level	86.04 percent	0.19 percent

table 4.5 Experiment results of the model of the current situation.

From the results it follows that the way the DUB currently operates cannot be considered satisfactory. The desired service level of 95 percent can not be achieved. Applications, retroactive applications, and mutations take on the average about four days to be processed. Income statements take over two days. When an income statement arrives after the 15th, it is almost certain not to meet the deadline of the 18th, as it takes in general almost 5 days to be processed. As a result, an average of almost 800 income statements is not being processed before the 18th of the month. In addition, over 600 mutations are processed after the deadline, which results in either

incorrect amounts or payments that bounce due to inaccurate client data. This results in a service level of just over 86 percent.

In order to find the bottlenecks in the DUB's processes and their coordination that cause the unsatisfactory performance, we studied the animation of the model of the current situation and the experimental data in more detail. At first, the indirect coordination of the various clerks' activities via the Archive clerks did not seem to cause any difficulties. However, after a closer examination it turned out to be a major problem: The amount of work in the DUB varies considerably during the day. A detailed analysis reveals a dependency between the amount of work and the schedule of the Archive clerks that distribute the items between the different offices. The animation model gave some clear examples of the inconveniences that resulted. At certain times during the day, more than 50 percent of the clerks of the Regulation office does not have jobs to process. On the other hand, vast amounts of work for the Regulation office are present at other offices. It only has to be distributed. A second example is the fact that the carts are driven by employees of the Archives, independent of the amount of work that still has to be processed there. A third example involves the workload difference between the different offices during the day. It is peculiar that the DUB, when viewed as a whole, has a large workload, while during certain times of the day employees have no work to do at their own office.

These observations are supported by the experimental data. This data shows that, for example, a mutation has an average throughput time of about 31 hours. However, if the mean times of the separate activities with respect to process of handling a mutation are added together, it turns out that a little under an hour is required to complete this process. This means that each mutation is delayed for about 30 hours because the actors involved in processing it are waiting for items that are waiting to be transported to them in some other office. Similar observations can be made for the other items that the DUB handles.

In response to the observed problems, the DUB needs to improve the coordination of its processes in order to increase its performance. As stated before, the DUB aims at a service level of at least 95 percent, especially given the expected increase of the number of clients and the expected decrease of the number of clerks. To bring about the improved coordination that is required to augment the DUB's service level, a number of alternative ways to coordinate the DUB's activities are proposed in the next section. The effects of these alternatives will be evaluated by implementing them into the empirical model and comparing the output variables, or indicators, with the ones of the current situation.

4.6 Alternatives for change

There are a myriad of potential change alternatives to improve the coordination of the DUB's processes, see [Dur 1992; Vreede 1992a]. These alternatives fall in one or more of three categories. First, alternatives may be aimed at facilitating the coordination in the DUB by changing the arrival behavior of items, e.g. by changing the client behavior. Second, the DUB organization itself can be changed in terms of the interdependencies of related activities and the coordination processes that currently take place. Third, the means by which the coordination in the DUB is carried out, in other words the coordination medium, can be improved. A number of examples of each type of change is given in table 4.6.

Client behavior change
Send early reminder for income statements to clients.
Introduce phone periods during the day.
Penalize late arrival of income statements.
Organization change
Filter out income statements and mutations that contain relevant changes.
Appoint a central work coordinator.
Assign clerks to document types and not to office, based on their skills.
Station Verification clerks in the Special Cases and Regulations offices.
Medium change
Increase the frequency of the Archives clerks' rounds.
Have rounds made by clerks from the least busiest office.
Introduce an Electronic Document Handling system.
Introduce a workflow management system.

table 4.6 Potential change alternatives for the DUB.

In the original case situation, Dur [1992] investigated some alternatives of the first and second type. The results showed that changes in the work process of income statements caused the highest improvement. For our, simplified, case situation we will explore alternatives of the third type. The DUB case offers a good possibility to study the potential impact of new forms of IT that support organizational coordination. The two alternatives we investigate represent 'low-tech' coordination support, an increase of the frequency of the Archives clerks' rounds, and 'high-tech' coordination support, an Electronic Document Handling system. Both alternatives are discussed in more detail below.

4.6.1 Extra Archives clerks' rounds

The first change alternative aims to speed up the distribution process of documents between the different offices by creating extra rounds of the Archives clerks. Within the schedule of these rounds there is space for two extra rounds, one in the morning,

and one in the afternoon. In order to keep the capacity utilization of the Archives at an acceptable level, an extra Archives clerk is added to the model to take care of the extra work.

To evaluate the effects of this change alternative on the DUB's performance, a similar experiment was carried out as with the model of the current situation: there were 7 replications in each experiment. The results are given in table 4.7. A *t*-test was performed with a significance level $\alpha=0.05$ to check whether the improvements could be considered significant.

Output variable (time in minutes, level in percentage)	Current situation		Extra rounds		Significant? ¹
	Mean	St.dev.	Mean	St.dev.	
Application throughput time	29.80	0.60	27.00	1.06	yes
Retroactive application throughput time	34.56	0.62	30.89	0.96	yes
Mutation throughput time	31.42	0.57	28.75	0.95	yes
Income statement on time throughput time	17.95	0.29	16.41	0.26	yes
Income statement too late throughput time	39.10	0.62	36.36	1.14	yes
Request for information throughput time	13.81	0.25	11.85	0.25	yes
Telephone call throughput time	3.35	0.03	2.91	0.05	yes
Service level	86.04	0.19	86.82	0.27	yes

¹ $t_{0.025}(12) = 2.179$

table 4.7 Results of the first change alternative.

Although the results represent a statistically significant improvement in the DUB's throughput times and service, the results cannot be considered satisfactory. A mean service level of well under 90 percent is clearly unacceptable; it does not meet the solution requirement of 95 percent. It seems that the frequency with which the items between the DUB's offices are distributed, needs to be increased.

4.6.2 Electronic Document Handling

The second change alternative can be seen as a further extrapolation of the first alternative. The introduction of an Electronic Document Handling (EDH) system replaces the rounds by the Archives clerks for a fully automatic transport of documents between the DUB's offices. The proposed EDH system is a documentary information system with the following functionality:

- Scanning documents at the Mail Room and storing them in a central database.
- Distributing all documents that need to go to other offices every 10 minutes.
- Delivering client files upon request automatically to the clerks of the Regulation and Special Cases office.
- Printing outgoing documents in the Mail Room.

Because everything is stored electronically, the waiting times for retrieving dossiers from the Archives and the distribution times between the offices should be eliminated. In fact, the EDH system pushes the DUB in the direction of a paperless office. The introduction of this change alternative has a dramatic impact on the Archives. After a period of transition, the Archives clerks seem to be out of work. However, the Mail Room will require extra capacity to handle the extra workload (scanning and mailing of printed documents). In the implementation of the EDH alternative in the empirical model, the available capacity of the Archives has therefore been transferred to the Mail Room. The results of the experimental evaluation of the EDH alternative are presented in table 4.8. Again, there were 7 replications in each experiment.

Output variable (time in minutes, level in percentage)	Current situation		EDH system		Significant? ¹
	Mean	St.dev.	Mean	St.dev.	
Application throughput time	29.80	0.60	3.74	0.06	yes
Retroactive application throughput time	34.56	0.62	4.65	0.10	yes
Mutation throughput time	31.42	0.57	6.34	0.06	yes
Income statement on time throughput time	17.95	0.29	4.98	0.12	yes
Income statement too late throughput time	39.10	0.62	5.44	0.09	yes
Request for information throughput time	13.81	0.25	2.64	0.08	yes
Telephone call throughput time	3.35	0.03	0.57	0.00	yes
Service level (percent)	86.04	0.19	98.62	0.05	yes

¹ $t_{0.025}(12) = 2.179$

table 4.8 Results of the second change alternative.

The results clearly show a great improvement in the DUB's throughput times and service level. The required service level of 95 percent is achieved. A further analysis of the experiment data shows that the income statements and mutations that are not handled in time for the deadline, mostly arrive in the DUB at the day of this deadline. The DUB can therefore not be held fully responsible for the late payments that remain.

An additional advantage of the EDH alternative is that it may broaden the type of coordination processes in the DUB. Both in the current and in the alternative situations, the DUB only uses adaptive coordination processes. The ability of an EDH system to electronically store various performance data opens up possibilities for *review* processes that may help the DUB to learn from its functioning and hence improve future coordination efforts. For example, by keeping track of the type and frequency of certain types of errors in transactions forms, the DUB may decide to automate part of the transaction form creation or to assign Verification clerks directly to a group of Regulation or Special Cases clerks in order to provide direct control and continuous training.

4.7 Reflection

The DUB case represents an organization where coordination between the actors is a prerequisite for successful performance. Although it is a constructed case, we considered it to be 'rich' enough to explore the merits of dynamic modelling for coordination improvement, especially since it is firmly based on a real life situation. In addition, according to our own experiences and that of others to whom the DUB was presented, this case represents a substantial class of service organizations. This suggests that our findings with respect to the approach we employed may be of greater general significance.

We further reflect on this case study with a number of observations, that either address the DUB case itself or the dynamic modelling approach that was employed.

4.7.1 The DUB

In this chapter we only investigated changes in the coordination *medium* of the DUB. Possibilities of changing the coordination mechanisms itself have not been addressed (see [Dur 1992] for these type of alternatives). At this point, we want to point out one change alternative that is particularly relevant in case the DUB will need to further develop in order to improve its performance. This alternative concerns the introduction of special task groups, consisting of clerk of the different functional office. A task group could handle the processing of a particular item completely before starting on the next. As Meel [1994] has demonstrated, groupwise production may greatly enhance performance in information-intensive organizations.

A task group organizational structure would facilitate coordination efforts between different functional clerks that are partly involved in processing certain items. Another advantage would be that the workload could be spread more evenly over the available clerks. In spite of these advantages, the coordination costs of this type of organization often is high [Malone and Crowston 1994]. There has to be insight in the workload of each task group. Timeliness and availability of items is important for the smooth functioning of each task group. To meet these requirements, the application of IT supporting group coordination, such as workflow systems or the EDH solution discussed earlier, has to be considered.

4.7.2 The dynamic modelling approach

The dynamic modelling approach that we employed in the DUB case enabled us to effectively diagnose the DUB's current situation and helped us to identify and quantitatively evaluate a number of change alternatives. There are, however, a number of problems that occurred during this case. First, the way of modelling of our initial

theory on dynamic modelling did not offer adequate *modelling concepts for conceptualizing organizational coordination*. During the case study, we therefore developed a first version of the modelling concepts that were introduced in Chapter 3 and used in this chapter. The modelling concepts turned out be especially useful in highlighting the interdependencies between the DUB's various activities, in focusing on coordination in term of the interaction between the DUB's clerks, and in identifying the present coordination structures and processes.

Second, the way of working did not offer guidelines for the *transformation of a conceptual model into an empirical model*. In the DUB case, the empirical model, especially the simulation code part, was constructed using a number of modelling heuristics and tips from (simulation) modelling experts. We felt that a more structured approach to this transformation could both increase the efficiency of the process as well as the quality of the resulting model.

Third, if the DUB case had to be taken further in terms of implementing the proposed EDH alternative, the dynamic modelling approach would not have been sufficient to support the *development of this information system*. It has been shown that dynamic modelling supports the requirement specification of a proposed information system [Meel 1994; Sol 1982]. In the words of Sol [1982, p. 190]: '*The emphasis lies on the solution of the systelological problem and on the transition from the systelological to the infological one*'. Hence, the approach requires additional methods and techniques for the subsequent translation of these requirements into function and data specifications, prototypes, and working information systems. Especially in real life situations where the design of organizational change and IT to support the changed organization are intertwined, the dynamic modelling approach seems to require an extension or alliance with an appropriate approach for information system design and development.

5 Case 2: The Neurology Outpatients' Department

'I had the uneasy feeling there was too much flux, too much instability in the system. I felt a little like an interviewer who has come upon his subject at a bad time. Only later did I realize that there would never be a "good" time, and that change is a constant feature of the hospital environment. The true figurehead of modern medicine is not Hippocrates but Heraclitus.'

- Michael Crichton

5.1 Introduction

Our second case study with the dynamic modelling approach was carried out at the Neurology Outpatients' Department of Leyenburg Hospital in The Hague, the Netherlands. The aim of this case study was threefold. First, we wanted to further develop the modelling concepts for conceptualizing a problem situation. Second, we developed and tested a procedure to transform the conceptual model into the simulation code part of the empirical model. Third, we examined to what extent animation could be used to enhance the communication processes in a dynamic modelling study, especially with respect to the involvement of the problem owners.

In this chapter we first describe the problem situation at Leyenburg Hospital in more detail. We then subsequently discuss the conceptual model (Section 5.3), the empirical model (Section 5.4), the problem diagnosis (Section 5.5), and the alternatives for change (Section 5.6). The chapter is concluded with a reflection on this case study.

5.2 Problem situation

Leyenburg Hospital, in short Leyenburg, is one of the municipal hospitals in The Hague. Founded in 1815, it moved into its current location and facility in 1971. The hospital houses about 800 beds, 110 specialists, and 700 nurses. Specialists are employed by the hospital, but are also allowed to have their own practice in the hospital's outpatients' departments. Annually, there are about 300,000 patient visits to the outpatients' departments, while the clinic handles 20,000 admissions.

Health care in the Netherlands is going through a very hectic phase. There occur structural changes in the insurance scheme, the "graying" of the population is on the increase, the number of people working in health care is diminishing, patients are assuming a more critical attitude, and the government requires the number of hospital beds to be reduced [CABLA 1990]. Consequently, hospitals' means threaten to be insufficient to satisfy the growing demand for care, both in terms of quantity and quality. As a result, medical considerations will no longer be the exclusive criteria that determine the level of care that the hospital will provide. Economic considerations will also have to be taken into account.

In response to these problems, hospitals have focused on exploiting the available means as efficiently as possible. An example of this is the increasing number of patient visits to outpatients' departments compared to clinical admissions. Expensive admissions for diagnostic purposes have become exceptional.

It was not until a few years ago that Leyenburg recognized that it also had to take appropriate actions to face the increasing complex problems. Hence, Leyenburg's management initiated a large scale reorganization. The goal of this reorganization was to become a hospital that provides top-clinical care in a patient-friendly and efficient way [Leyenburg Management 1991]. 'Top-clinical' care means that the hospital is able to provide and train a great diversity of specialist care.

A premise of the reorganization efforts at Leyenburg is that, in order to be patient-friendly, the hospital's structure should be based on current and future flows of patients [Hogenwind 1988; Leyenburg Management 1991; Ramondt and Rhébergen 1991a and 1991b]. 'Patient-friendliness' is related to the time that individual patients have to spend in the hospital. This not only concerns the duration of a patient's individual visits, but also the number of visits that the patient has to make for the whole treatment he or she is receiving and to the length of this treatment. It is assumed that the length of a treatment or the number of required visits is mainly dependent on *medical* grounds. The length of an individual visit is mainly determined by the *efficiency* with which the hospital handles such a visit.

During a patient's visit, various actors in the hospital have to coordinate their efforts to make the visit as efficient and effective as possible. The orchestration of these activities is a point of consideration for the hospital's management. Following the national trend, the number of patient visits to Leyenburg's outpatients' departments has increased considerably over the last years compared to the number of clinical admissions. Since patients often have to pass by more than one part of the outpatients' departments during a visit, some outpatients' departments hold a great number of patient during the day for considerable lengths of time. It is believed that this situation

can be improved if the actors that are involved in a patient's visit could better coordinate their efforts with respect to this patient.

For a number of reasons, the intended reorganization of the hospital has so far not been able to take up the challenge presented by the above problems [Vreede 1993a]. First, it was unclear how to quantify the care process in such a way that this process could be evaluated. The hospital had never before felt a need to measure its performance in terms of quantity and/or quality and use this to make structural changes. Second, there were two influential parties in the reorganization process: the medical staff and the managerial staff. Although both parties recognized the necessity to reorganize, they were, unfortunately, not always in agreement on how this should be done. Third, the financial information of the hospital turned out to be unsuitable for performance evaluation [Dekker 1987; Vreede 1992b]: Leyenburg's budgeting method did not relate budget, costs, and revenues.

Problem description

Pressures from the hospital's environment:

- Increasing demand for care.
- More critical attitude to care.
- Less means for care in terms of personnel, equipment, and time.

Increasing costs of providing care.

Little consensus between medical and hospital staff on direction of reorganization.

No suitable process information available for performance evaluation.

Financial information not suitable for performance evaluation.

Increasing claim on capacity of outpatients' departments making visits long and crowded.

table 5.1 Problems faced by Leyenburg.

The various problems that Leyenburg is facing have been summarized in table 5.1. Given these problems, the object of our study from the point of view of the organization was defined as to:

1. Demonstrate how and which quantitative information about flows of patients could be used to assess the performance of the hospital in terms of patient-friendliness.
2. Examine how the patient-friendliness could be improved by enhancing the coordination between the relevant actors in the hospital.

In view of the size of Leyenburg, the study was limited to Outpatients' Department of Neurology. According to the hospital management, this can be considered to be an 'average', representative department. The Neurology Outpatients' Department deals with patients that suffer from, for example, a disturbance of equilibrium, epilepsy, dementia, or psychiatric disturbances. Patients are usually referred to the department by their family doctor. Occasionally patients are referred by Leyenburg's emergency room. As part of their treatment by the Neurology Outpatients' Department, patients

may be sent to functional departments to undergo various tests that help the physicians to diagnose and treat the patients' problems.

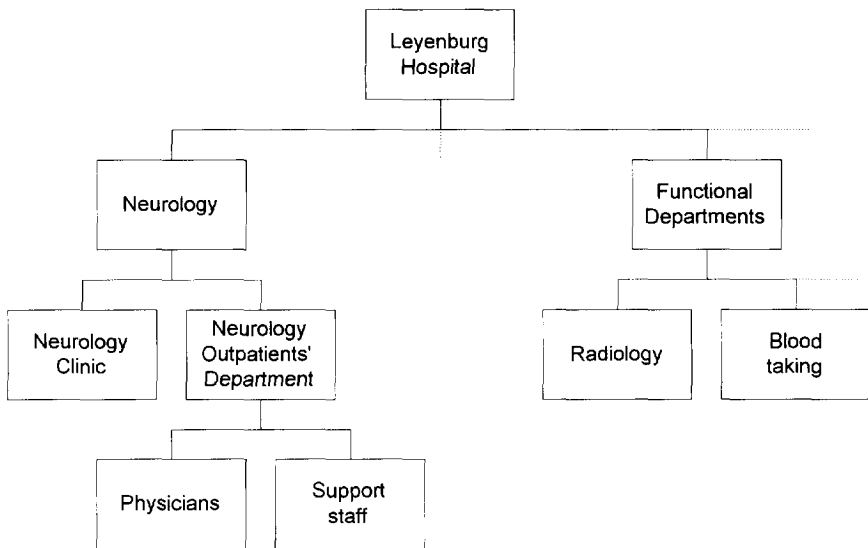


figure 5.1 Organization chart of the relevant parts of Leyenburg.

The relevant organizational parts of Leyenburg for our study are outlined in figure 5.1. As depicted, the Neurology Outpatients' Department employs, apart from physicians, a number of supporting staff employees that make appointments and carry out secretarial activities. Through the remainder of this chapter we will refer to the Neurology Outpatients' Department together with the relevant functional departments as the *NOD*. In the next section we will describe the structure of the situation at the *NOD* in more detail. The situation will subsequently be specified in an empirical model that can be used for diagnosis purposes.

5.3 The conceptual model

In this section we describe the conceptual model of the *NOD* case. First, we discuss the object class definitions that together form the conceptual model. The genders used to illustrate the object classes are random where appropriate. Then, we highlight specific (coordination) aspects by presenting the interaction, process, and actor model of the conceptual model. During the discussion of these three models, we will use the taxonomy of coordination mechanisms from table 1.1.

5.3.1 Object definitions

In the NOD we can discriminate between three different item classes. The predominant one is the *patient*, which is modelled in the form of a person item class. One may argue that patients should be modelled as actors instead of passive items since, in reality, they may display autonomous, unpredictable behavior. However, a *flow of patients* may be well predictable and can be planned because of the large numbers involved [Hogenwind 1988]. As we are not interested in individual patients but in the way patient flows are handled by the NOD, we decided to model patients as items.

There are two types of patients. The first is a patient that visits the NOD. We refer to an NOD patient as just 'patient'. A patient has to see a certain physician, for which she has made an appointment in advance. After being seen by the physician, a patient may need to undergo functional examinations or may need to be admitted. A patient may have poor mobility. In that case, a registration employee makes a personal *telephone call* to the registration employee of the appropriate functional department with whom the patient has to make an appointment for the examination. The second type of patients are other patients visiting the functional departments that NOD patients also visit. These patients are referred to as '*functional patients*'.

A patient's medical history is maintained in a *patient file*. Patient files are created by a registration employee when the patient visits the NOD for the first time. The patient files are used by the physicians to prepare for the patient's visits and to record what is discussed during such a visit.

object class PATIENT is a PERSON	object class TELEPHONE CALL is a PERSON	object class FUNCTIONAL PATIENT is a PERSON
attributes physician appointment information first visit? seen physician? examination information admission information mobility	attributes functional department object class PATIENT FILE is a MESSAGE attributes -	attributes -

The three item classes are handled by various item processors. We first describe the actors. During a visit to the NOD, a patient is handled by at least two actors: a registration employee and a physician. A *registration employee's* main actions include receiving patients, directing them to the waiting room and to the functional departments if the physician has ordered some examinations. The registration employee also makes the follow-up appointment for a patient, after the patient has made the necessary examination appointments.

In addition to dealing with the patients themselves, a registration employee also each morning gets the files of that day's patients from the patient file archive. At the beginning of a physician's shift, the registration employee hands the files over to him. The registration employee also creates files for new patients, which he sends together with the new patient to the waiting room.

Physicians do not have full time visit hours at the NOD. Each physician has his own schedule. After preparing the patient's visits, the physician calls patients from the waiting room according to their appointment time. He then 'sees' the patient, meaning that he checks the patient's condition and decides what future course of action has to be taken. This may include (1) sending the patient home, (2) wanting to see the patient again, possibly after having some examinations carried out to support the diagnosis process, or (3) having the patient admitted if her condition gives reason to do so. The physician gives the patient the required information for the future course of action and sends her home or to the appropriate actor. After the patient has left, the physician updates the patient's file, which he gives to his secretary at the end of his visit hours.

object class REGISTRATION EMPLOYEE
is an ACTOR

actions

patient seen physician already?
patient needs functional examination
appointment?
how is patient's mobility?
receive patient
receive patient file
send patient (with patient file) to waiting room
send patient to functional department
registration employee
send patient home
send patient files to physician
make next appointment
make phone call to functional department
registration employee

object class PHYSICIAN
is an ACTOR

attributes

physician_id
schedule

actions

visit hours over?
waiting room empty?
patient needs to be admitted?
patient needs to come back?
patient needs functional examinations?
receive patient
receive patient files
send patient home
send patient to registration employee
send patient (file) to secretary
prepare patient visits
see patient
call patient from waiting room
check patient's condition
give patient admission information
give patient appointment information
give patient functional examination information
update patient file

Patients that have to undergo examinations are sent to the appropriate functional department. A patient is first received by the *registration employee* of the functional department. For some examinations, such as blood taking, a patient can be dealt with

immediately by a *medical employee*. After the examination, the medical employee sends the patient back to the neurology registration employee. However, for most examinations an appointment has to be made first. This is done by the functional department's registration employee. After booking the appointment, the registration employee sends the patient back to neurology registration employee. Some appointments are made by phone if the patient has poor mobility. Functional patients are handled the same way as NOD patients.

object class FUNCTIONAL DEPARTMENT
REGISTRATION EMPLOYEE
is an ACTOR

attributes
functional department

actions
receive (functional) patient
receive telephone call
(functional) patient needs immediate
examination?
make examination appointment
send (functional) patient to functional
department medical employee
send patient to registration employee
send functional patient to origin

object class FUNCTIONAL DEPARTMENT
MEDICAL EMPLOYEE
is an ACTOR

attributes
functional department

actions
receive (functional) patient
perform functional examination
send patient to neurology registration employee
send functional patient to origin

A *secretary* of a neurology physician receives patient files from the physician after each shift of visiting hours. She is mainly occupied with handling the physician's correspondence that results from the patient visits. Further, a physician sends a patient to his secretary when she has to be admitted in the hospital's neurology clinic. The secretary gives the patient some additional admission information and sends her to an employee at the admission department. The *admission employee* makes the admission appointment for the patient and sends her home.

object class SECRETARY
is an ACTOR

actions
receive patient file
receive patient
give patient additional admission information
send patient to admission employee

object class ADMISSION EMPLOYEE
is an ACTOR

actions
receive patient
make admission appointment
send patient home

There are two repositories in the NOD, that are used to store and retrieve items in and from. The first repository concerns the *waiting room*. This clearly functions as a means of coordinating the flow of patients between the registration employee and the

physician: the registration employee sends patients to the waiting room where they are called from when the physician is available. The other repository concerns the *patient file archive*. This is both used by the secretary and the registration employee to hold the file of any patient that has visited the NOD.

object class WAITING ROOM
is a REPOSITORY

actions

receive patient
send patient to physician

object class PATIENT FILE ARCHIVE
is a REPOSITORY

actions

receive patient file
send patient file to secretary
send patient file to registration employee

The actors and repositories in the NOD are connected by various classes of links over which they exchange items. The bi-directional *archive link* connects the registration employees and secretaries to the patient file archive so that they can send and receive files to and from the archive. The function of the *telephone* link is self explanatory. The most predominant class of links in the NOD are the ones that route patients, such as the *examination link* defined below. These links classes actually represent the different walking routes that patients may have to follow during a visit to the NOD.

object class ARCHIVE LINK
is a LINK

attributes

starting node (Registration employee, Secretary)
ending node (Patient file archives)
bi-directional

actions

route patient file

object class EXAMINATION LINK
is a LINK

attributes

starting node (Reg. employee)
ending node (Functional department reg. employee)
bi-directional

actions

route patient

object class TELEPHONE
is a LINK

attributes

starting node (Registration employee)
ending node (Functional department reg. employee)
uni-directional

actions

route telephone call

Task and decision classes can be defined to model the actions of the actors in more detail. Below, we give an example of both a decision definition and a task definition. The decision represents the interpretation of a patient's mobility by a registration employee. The task definition represents the physician's compound task 'see patient'. This task and its constituents are graphically depicted in Section 5.3.4.

object class is a DECISION	HOW IS PATIENT'S MOBILITY?	object class is a TASK	SEE PATIENT
attributes		attributes	
required items: patient		required items: patient	
preconditions: -		preconditions: -	
postconditions: nature of patient's mobility known		postconditions: appointment information, examination information, and admission information are specified	
next actions: send patient to functional dept. reg. employee ('good'), make phone call to functional dept. reg. employee ('poor')		next actions: patient needs admission?	
decision rules: IF mobility = 'good' THEN send patient to functional department registration employee		delay: -	
IF mobility = 'poor' THEN make phone call to functional department employee		att. assignm.: -	
		subactions: call patient, check patient's condition, patient needs to be admitted?, patient needs funct. exam.?, give patient admission inf., give patient appointment inf., give patient funct. exam. inf., send patient home, send patient to reg. employee, send patient to secretary, update patient file	

5.3.2 The interaction model

Looking at the NOD as a whole, we see that the department coordinates the handling of patient visits according to a *hierarchy structure*. There are fixed rules of behavior, objective qualifications and qualities of office, and, for a hospital environment not unusual, a clear hierarchy of authority. The hierarchy structure of the NOD situation is partly functional and partly product oriented in nature. Within the NOD situation one can distinguish between several *product* groups: the neurology group (registration employee, physician, and secretary), the functional examination group (functional department registration and medical employee), and the admission group (admission employee). Each product group offers a specific type of service (care) to a patient.

Within each product group, there is a *functional* hierarchy between the medical staff, the registration staff, and the secretarial staff. The actors in each group perform certain (discrete) activities that together form the service that their group offers to a patient. Note that the separate functional actors have not all been modelled except for the neurology group, as they are not all equally relevant with respect to the handling of neurology patients.

The consequences of the mixed structure with which the NOD coordinates their efforts can be observed in the interaction model of the NOD situation, see figure 5.2. Patients are moving back and forth through the hospital. The more types of care service they need, the more products groups they are sent to. Within each group, the patient is sent from actor to actor to have the appropriate actions carried out. The reason that patients are sent to each separate actor, is that the patient herself is carrying the information that the actors need to perform their appropriate actions. Hence, in addition to

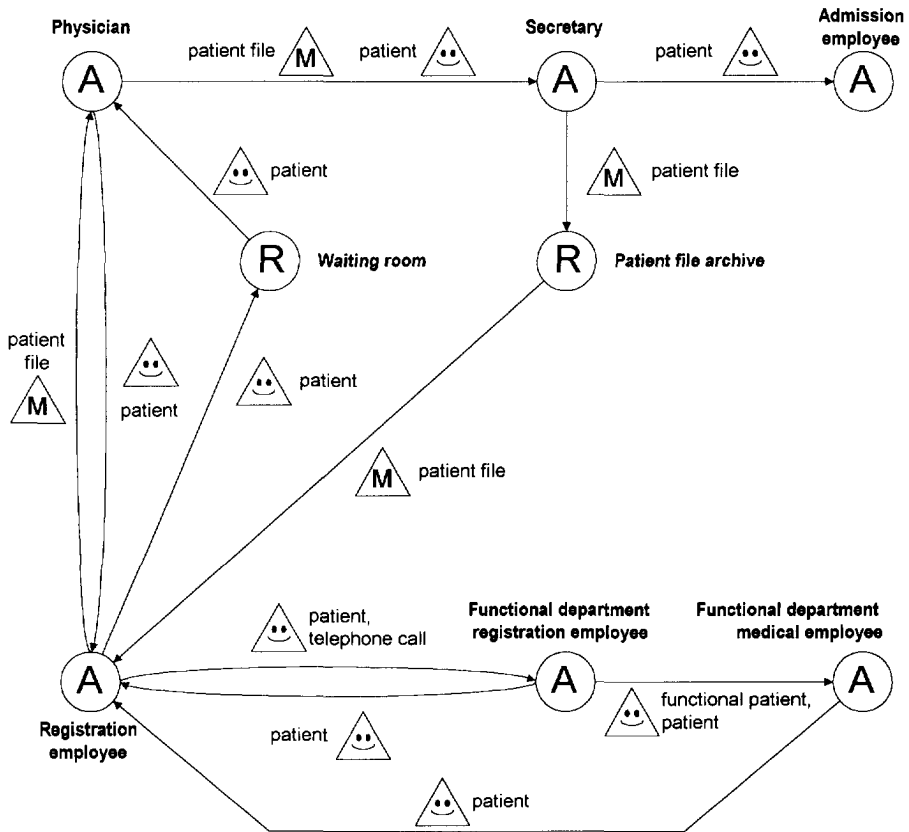


figure 5.2 Interaction model of the NOD situation.

receiving medical care, patients are used as a sort of medium with which the actors in the NOD situation coordinate their actions. How these actions are interrelated is the subject of the next section.

5.3.3 The process model

Taking a closer look at the processes in the NOD and the interdependencies between the constituent actions, we see that *standardization* is the predominant coordination mechanism. There is hardly any explicit communication (necessary) between the actors in the NOD situation; they perform their actions in a predefined way that is triggered by the information that the patient they are dealing with represents. This is visible in the process model of handling patients in the NOD, see figure 5.3.

There are two types of standardization in the NOD. The leading type is standardization of *work processes*. All registration and secretarial staff perform their actions according to a set of predefined procedures. The second type is standardization of *worker skills*.

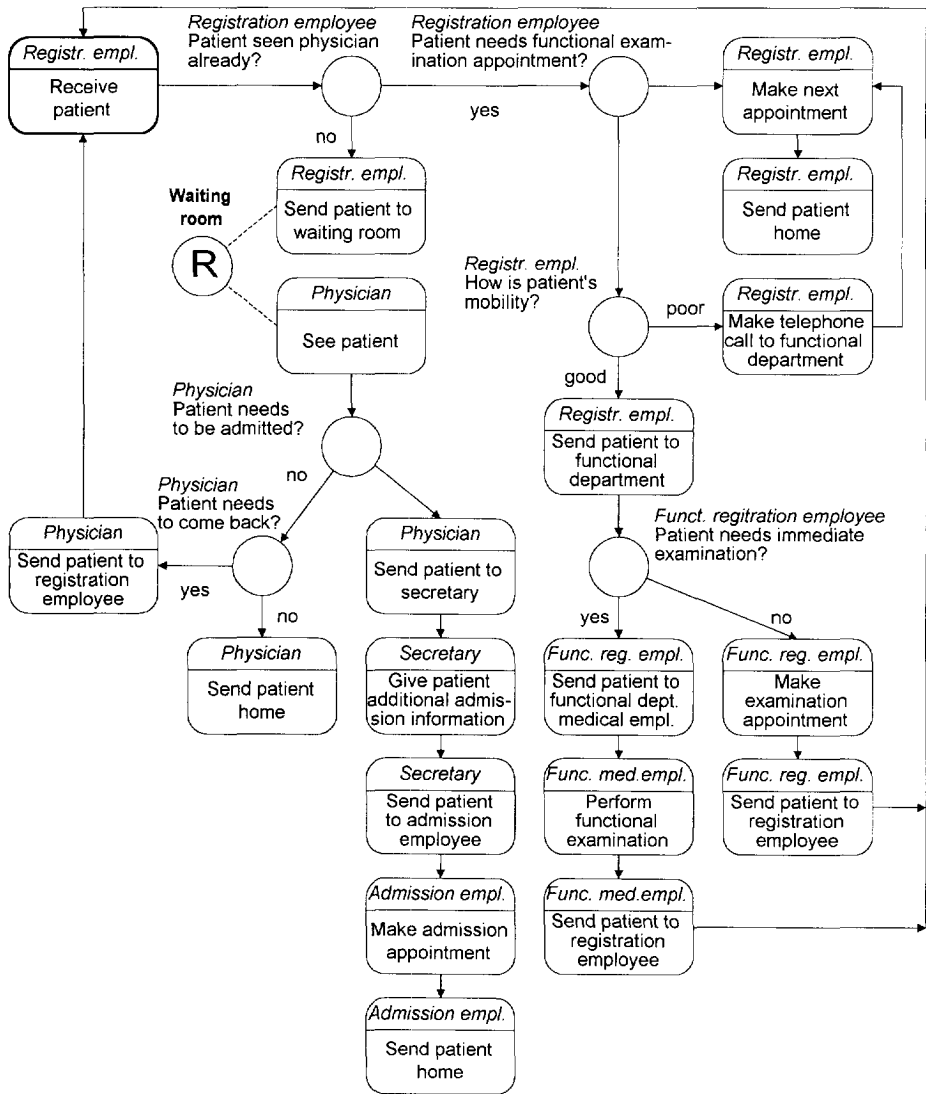


figure 5.3 Process model for handling patient visits.

The coordination among the medical staff in the NOD, the neurology physicians and the medical employees from the functional departments, is based on the assumption that the other medical staff actors will perform the required activities satisfactory because of their medical training and competence.

If we look at the coordination *processes* in the NOD, we see that *anticipatory* coordination processes are omnipresent. Patients only arrive at the NOD when they have an appointment made in advance. If it is decided during a visit that a patient

needs additional medical care, she cannot receive it before additional appointments have been set, except for a few types of examinations.

In addition to the anticipatory processes, there are also a few *adaptive* coordination processes. Physicians decide not before they see their patients what the rest of their treatment will consist of and what other actors need to be involved in it. Depending on a patient's mobility, a registration employee may decide to make the patient's appointment for a functional examination himself by telephone instead of sending the patient to the functional department involved.

Finally, there are no *review* processes in the NOD. The NOD does not keep record of patient's visits in such a way and on such a level of detail that this information is suitable to reflect on the performance of the NOD and to suggest future improvements.

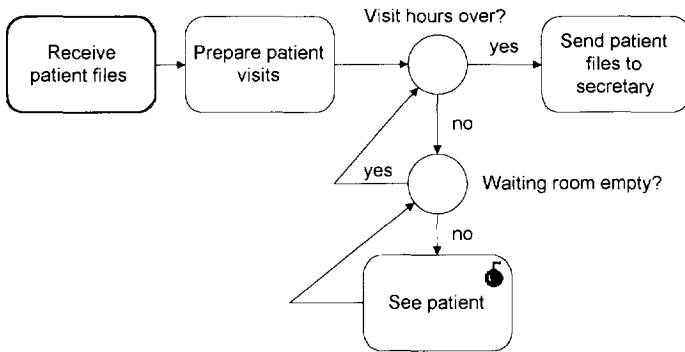


figure 5.4 Actor model for the physician.

5.3.4 The actor model

The actor model represents the interdependencies between the activities of individual actors. The actor model of a physician is given in figure 5.4. The task 'See patient' is further expanded in figure 5.5. This model also reflects the standardization discussed in the previous section. During a patient's visit, the physician does not need to communicate explicitly with any other actor in the NOD. He only receives the appropriate patient files from the registration employee before the visit hours begin and hands the modified files over to his secretary when his shift is over. In the mean time, he performs his part of the chain of activities that a patient is subjected to and can trust that the other actors involved will carry out their part of the chain as well. Similar models and observations can be made for the other actors in the NOD.

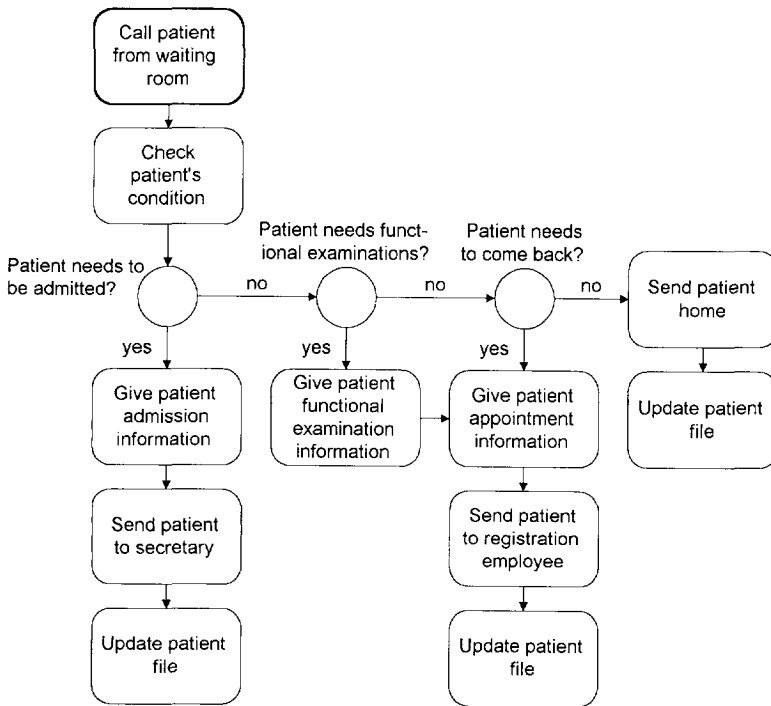


figure 5.5 Actor model of the task 'See patient'.

5.4 The empirical model

The conceptual model as described in the previous section was used as a basis for an empirical model to analyze the NOD situation. In this section we discuss the three parts of this empirical model. First we shortly demonstrate some parts of the simulation code that was constructed using the transformation procedure described in Chapter 3. The empirical data and the experimental design are the subject of Sections 5.4.2 and 5.4.3 respectively. Next, the animation model of the NOD empirical model is described. This section is concluded with a discussion of the verification and validation of the empirical model. In Section 5.5 we will discuss the experiment model of the empirical model together with the problem diagnosis of the NOD situation.

To conclude this introduction, we briefly indicate which reductions we made during the transformation of the conceptual model into the empirical model:

- In the real NOD situation, a small percentage of patient does not show up for their appointment. This percentage was considered small enough by the NOD employees to justify exclusion from the empirical model.

- The flow of patient files has not been included in the empirical model either. The object of our study was to investigate (the handling of) flows of patients. From the conceptual model it appeared that the flow of files does not have a great effect on the flow of patients as it predominantly takes place *before* and *after* the visit hours. Feedback from those involved with the flow of patient files, registration employees, physicians, and secretaries, convinced us that these items could be excluded from the empirical model. Moreover, the NOD employees could not identify any problems with respect to patient files that affect patient visits, such as missing files delaying a visit. The only aspect that was included in the empirical model is the required time to create a patient file when a new patient presents herself since this does effect the flow of patients directly.
- There are three qualified neurology physicians working in the NOD together with two assistants physicians that are in training. However, there are only four consultation rooms in the NOD, which therefore have to be shared among the physicians. To reduce the complexity of the simulation code, the five physicians have been implemented as four resources that operate according to a schedule that reflects the occupancy of each consultation room.
- There are many different types of examinations that neurology patients may have to undergo. These examinations are performed in various functional departments throughout the hospital. Only three of these departments have been included in the empirical model: the Radiology department, among other things responsible for CT-Scans, the Clinical Neurophysiology (CNPh) department, among other things responsible for electroencephalograms (EEGs), and the Blood taking department. These three departments jointly handle 92 percent of all examinations of the NOD patients [Dijk-Veenenga Kingma 1992].
- For the Radiology and CNPh department, no waiting queue and no other patients than NOD patients were included in the empirical model. We observed that these departments were always very quiet in terms of patients to handle. NOD patients sent over to make appointments could be dealt with immediately.
- Telephone calls by the registration employee to the registration employee of a functional department have been implemented as an extra delay for the registration employee's task 'make next appointment'. For the analysis of the NOD situation it was not relevant to simulate the behavior of the functional department registration employee answering a telephone call.
- The NOD is only simulated between 7:00 and 18:00.

5.4.1 Simulation code

In this section we illustrate the transformation of some of the objects of the conceptual model into SIMAN simulation code. The definition of a patient in terms of SIMAN's attributes element is given below. Exclamation marks indicate comments which are not part of the actual code.

```

ATTRIBUTES: Physician,0:      !Value 1-4
              Appointment,0:  !Appointment time
              New,0:          !0=return visit, 1=first visit
              Examinations,0: !Value 0-7 (Blood=1,Radiology=2,CNPh=4)
              Admission,0:    !0=no admission, 1=admission
              Mobility,0;      !0=poor, 1=good
    
```

The examinations attribute represents the sum of the values of the examination departments that the patient has to visit. Patients visiting the Blood taking department were defined the same way as neurology patients because SIMAN only supports the definition of one entity (item).

The following piece of SIMAN code describes the implementation of the Radiology department registration employee. This employee receives patients, makes examination appointments, and sends patients to the next actor that they have to visit afterwards. Patients are never examined immediately at the Radiology department. As there are no preconditions for the Radiology employee's task 'make examination appointment', there is no simulation code representing this.

```

              STATION, Radio;                !Rad. empl. receives patient
              QUEUE,   RadioQ;                !Patient enters queue
              SEIZE:   RadioR;                !Employee handles patient
              BRANCH,  1:ALWAYS,RadioAL;      !Branch to list of actions
RadioRA       RELEASE: RadioR;                !Employee done with patient
              BRANCH,  1:ALWAYS,RadioLI      !Patient to link interface
...
RadioAL       BRANCH,  1:ALWAYS,AppntmntRAD;  !Jump to 1st task
AppntmntRAD   ASSIGN:  Examinations=Examinations-4; !Appointment made
              DELAY:   ED(19);                !Duration of task
              BRANCH,  1:ALWAYS,RadioRA      !Last task done
    
```

After handling a patient, the Radiology department's registration employee either sends the patient to a registration employee from the CNPh department, to one from the Blood taking department, or to one from the neurology physician. For this purpose, the entity representing the patient is first sent to a link interface (RadioLI). This interface determines to which link it has to be sent. After passing through the interface, the proper link routes the patient-entity to the station that receives entities for the appropriate next actor.

The simulation code for the link interface and link objects involved is given below. The delay distributions, ED(n), represent the 'walking' times from the begin station to the end station of the link. As the walking routes are always available, there is no code included to check whether the link is 'active' or not.

```

RadioLI       STATION, FromRad                !Link interface
              BRANCH,  1:                    !Take proper link
              IF, Examinations>1,ToCNPhLI:
              IF, Examinations=1,ToBloodLI:
    
```



```

ELSE, ToPhysRegLI;
ToCNPhLI    ROUTE:    0, ToCNPh;          !To link to CNPh
ToBloodLI   ROUTE:    0, ToBlood;         !To link to Blood
ToPhysRegLI ROUTE:    0, ToPhysReg;       !To link to phys. reg. empl.
...
STATION,    ToCNPh;          !Link to CNPh employee
ROUTE:      ED(8), CNPhST;    !Patient sent to CNPh empl.
...
STATION,    ToBlood;         !Link to Blood employee
ROUTE:      ED(6), BloodST;   !Patient sent to Blood empl.
...
STATION,    ToPhysReg;       !Link to Phys. Reg. employee
ROUTE:      ED(4), PhysRegST; !Patient sent to Reg. empl.

```

Similar pieces of simulation code as the ones we just described, have been constructed to implement the other actors, repositories, links, and actions that were included in the empirical model. In addition, SIMAN blocks and elements were added (1) to collect statistical data during the execution of the empirical model, and (2) to facilitate and enhance the animation of the empirical model. To facilitate the maintenance and verification of the simulation code, we constructed a submodel for each actor in the NOD. All link objects were represented in one submodel called 'walking.mod' as they represented the various walking routes through the NOD.

5.4.2 Empirical data

Collecting the empirical data for the empirical model turned out to be a major problem in the NOD case study. Much of the required information was either not recorded or recorded in an almost useless form, see also [Dijk-Veenenga Kingma 1992]. A number of examples illustrate this. First, processing times for the various tasks were unknown. Second, the percentage of patients that had to visit the functional departments to undergo examinations was not known either. The hospital's administration only recorded the number of actual examinations per department. Since patients had to undergo between one and sometimes up to ten different examinations per functional department, the number of departments the patient went to *per visit* could not be retrieved. Third, some data could have been recovered from the hospital's central information system. However, this was not possible because the hospital's IS department found it technically too complex, too demanding on their available capacity, and in violation with a patient's privacy. Fourth, another source of data involved the patient files, which were made available to the researchers. Unfortunately, it turned out to be too difficult to understand these files well enough without any medical background. Besides, it was too time consuming to go through each file manually.

To overcome these problems we had to resort to doing a great deal of personal observations for a couple of days, recording arrival times, task durations etcetera. Also, the physicians and registration employees cooperated by giving a number of expert

estimations, triggered by prototypes of the animation model. Below we describe some of the empirical data that was incorporated in the empirical model. This data falls into the following categories: arrival and type of patients, the capacity of the actors, and the processing times of the tasks. Decision were not expressed in terms of probabilities as in the DUB case, but in terms of the attribute values of the patient items that were determined at the (simulated) arrival at the NOD.

Patients are supposed to arrive at the NOD registration desk in time for their appointment. Visit hours are normally scheduled in such a way that a physician sees a number of batches. Each batch consists of one new patient and four patients with a follow-up appointment. For new patients, 30 minutes are scheduled, for 'follow-up' patients 7.5 minutes. Hence a physician is supposed to see 5 patients per hour on the average. Obviously, patients often arrive well before the appointment is due. Their arrival is, therefore, modelled as a process that generates a patient item every 7.5 or 30 minutes and sends it to the registration desk a number of minutes before the actual appointment time. The time that a patient arrives too early is modelled with a triangular distribution with a mean of 18, a minimum of minus 7 (patient is too late), and a maximum of 50 minutes. We choose a triangular distribution as this is the best option in situations where there is little empirical data available [Pegden et al. 1990].

The way in which patients are handled and flow through the NOD depends on their attribute values. Based on a number of observations, expert estimations, and data from the hospital administration, we built the 'chance-tree' depicted in figure 5.6 to determine the attribute values of each patient entering the NOD model.

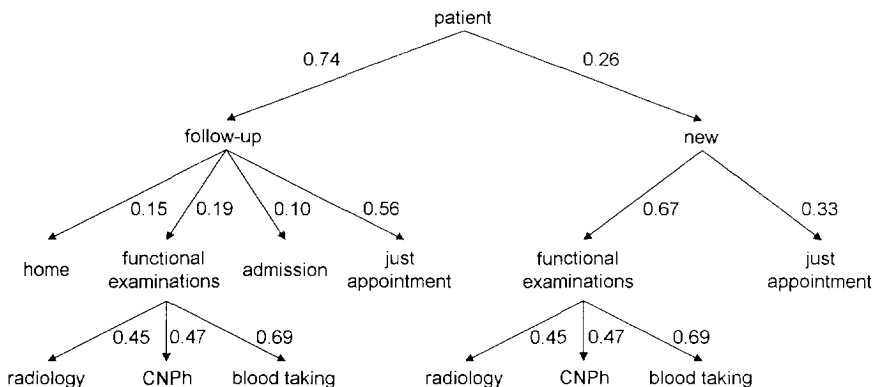


figure 5.6 Chance tree to determine the attribute values of a patient.

The number on each branch of the chance-tree represents the chance that a patient is of a certain type. The different types that we distinguish are new patients, and patients coming for follow-up appointments. New patients either have to make a follow-up appointment only, or they have to undergo some examinations as well. 'Follow-up'

patients may be sent home, be admitted, or have to make a new follow-up appointment, possibly with some examinations. Note that the chances on the branches of the examinations do not add up to one. This is because patients may have to undergo more than one examination.

Physician	Monday		Tuesday		Wednesday		Thursday		Friday	
	am	pm	am	pm	am	pm	am	pm	am	pm
1	-	x	-	x	x	x	-	x	x	-
2	x	x	-	x	x	-	-	x	x	-
3	x	x	-	-	x	x	-	-	x	-
4	x	-	-	x	x	x	-	x	x	-

- = no visit hours, x = visit hours

table 5.9 Visit hours schedule of the neurology physicians.

The NOD's physicians work according to the schedule given in table 5.9. The other actors in the NOD work regular hours. There are two neurology registration employees working every day, from 8:00 till 16:45 with a 45 minutes lunch break. They take this break in turn to keep the registration desk manned. The secretaries and the registration employees at the Radiology, the CNPh, and the Blood taking department are present from 8:00 till 16:30. At the Blood taking department there are three medical employees that are also present from 8:00 till 16:30. However, if the number of patients in the waiting room becomes too large, one or two extra employees are called for from the laboratory upstairs.

With respect to the processing times of the tasks in the NOD, a number of examples are given in table 5.10. These tasks are depicted in figure 5.3. The processing times were all implemented using a gamma distribution (which is a generalization of the Erlang distribution). The gamma distribution is especially suitable to model the time to complete tasks like customer service [Pegden et al. 1990].

Actor	Task	Mean time to complete
Physician	See new patient	28.19 min
	See follow-up patient	7.61 min
Secretary	Give patient additional admission information	3.41 min
Registration employee	Make telephone call to functional department	3.39 min
	Make appointment	2.27 min
Fun. registration employee	Make appointment	2.27 min
Fun. medical employee	Perform functional examination (take blood)	4.44 min

table 5.10 Example processing times of actions of NOD actors.

5.4.3 Experimental design

We now turn to describing the treatment to which the NOD's empirical model was exposed. The specification and collection of the input data has been discussed in the previous section. Below we discuss the specification of the initialization conditions, of the output data, and of the run control conditions.

The NOD represents a *closed system*. This means that at the beginning and at the end of each simulated day, there are no items (patients) in the model. Hence, it is not necessary to determine a start up time for the model.

The run control conditions consist of the run length and the number of replications. The physicians work according to a weekly schedule that differs per day, see table 5.9. This implies that the workload of the actors in the NOD is not the same each day. In order to compensate for the varying workload, the cycle time of the NOD empirical model is set to one week instead of one single day. Therefore, the run length should at least be set to three weeks. Furthermore, the run length is set to 6 months to compensate for the substantial variances in some of the distributions. These variances were due to the, in the statistical sense, limited number of observations.

In order to determine the number of replications, we first have to specify the *output data*. There are two types of output data we want to record. The first concerns the *type of patient*. Looking at the way in which a patient is sent back and forth through the NOD after seeing her physician, we can identify four types of patients, see also figure 5.6. A physician may send a patient home, may have her admitted, or may order a follow-up appointment, possibly with a number of examinations. These patient types result in a four different *flows of patients* that can be observed in the NOD. The second type of output data concerns the *process phase* a patient is in during a visit. We distinguish between four phases. A patient can be at the registration desk, she can be in the waiting room, with the physician, or at the functional departments. For both types of output data we measure the *throughput time*. Thus, for each type of patient we keep track of the amount of time she spends in each phase and in total.

With the output data specified, we performed a number of initial replications. Based on the outcome of these replications, we chose an acceptable confidence interval. We used the total throughput times of each type of patient to determine the number of replications. This way we, indirectly, also took the different phases into account. We calculated the minimal number of replications for the selected output variables to achieve a significance level of $\alpha=0.05$ using the procedure described in [Meel 1994]. Based on the outcomes of the initial replications, the desired confidence interval was subjectively set to 0.5 minutes. The results are given in table 5.11. Based on these results, the number of replications is set to 10.

Output variable	Desired confidence interval	Minimal number of replications ($\alpha=0.05$)
Home	0.5 minutes	10
Admission	0.5 minutes	9
Just follow-up appointment	0.5 minutes	7
Functional examinations	0.5 minutes	9
Total	0.5 minutes	8

table 5.11 Determination of the number of replications.

5.4.4 The animation model

The dynamic characteristics of the NOD situation were highlighted by an animation model that was implemented in Cinema. The animation model consisted of three animation screens: one screen showing the physicians, the registration employees, and the secretary, another screen showing the three functional departments, and a final screen displaying statistical information, such as the mean throughput time of patient, the length of the queues in front of registration employees, and the occupancy of waiting rooms. The three screens are depicted in figure 5.7, figure 5.8, and figure 5.9 respectively.

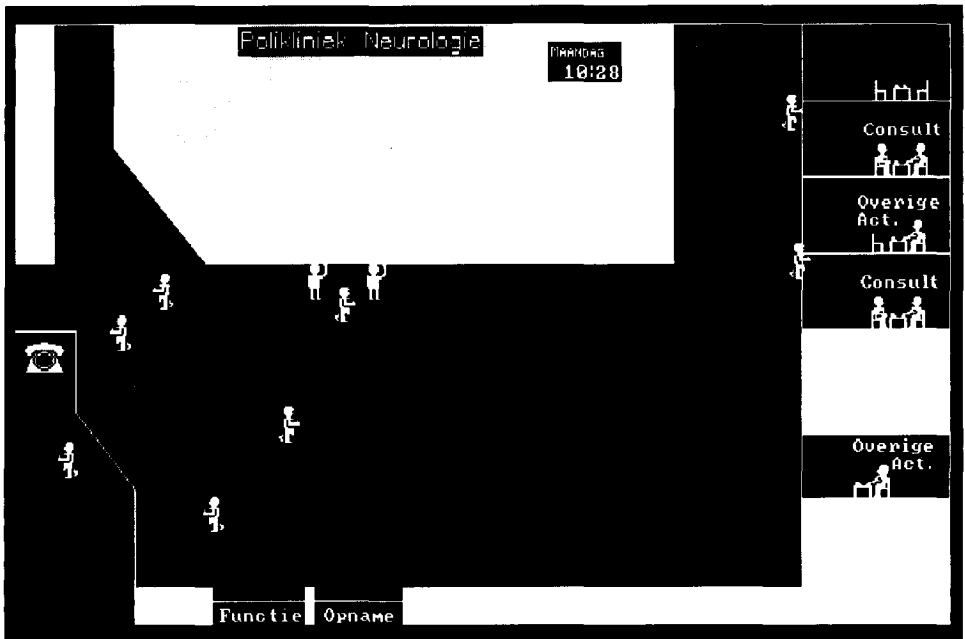


figure 5.7 Animation screen of the neurology actors.

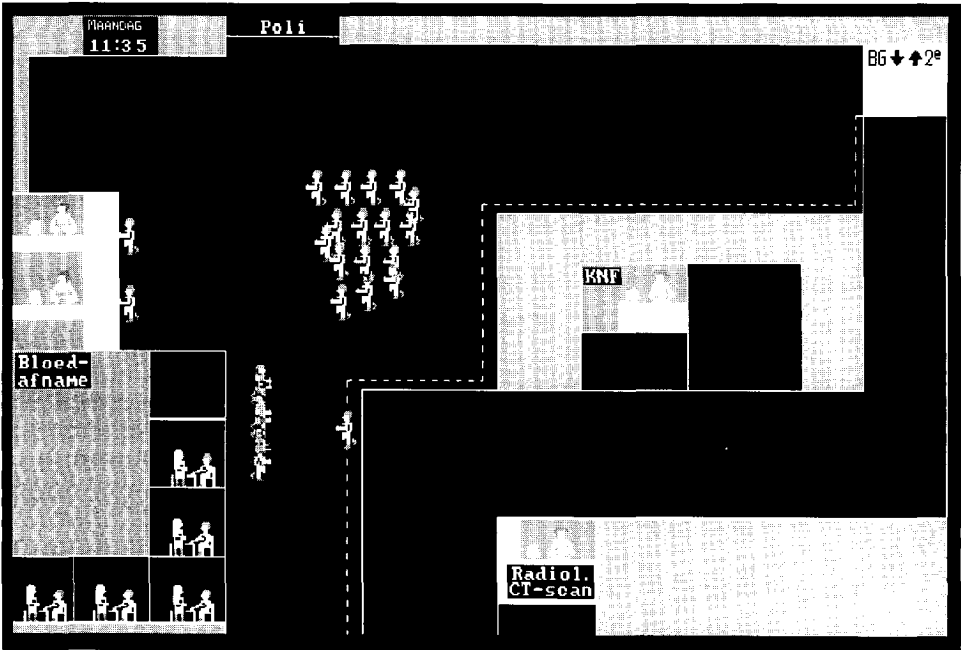


figure 5.8 Animation screen of the functional examination departments.

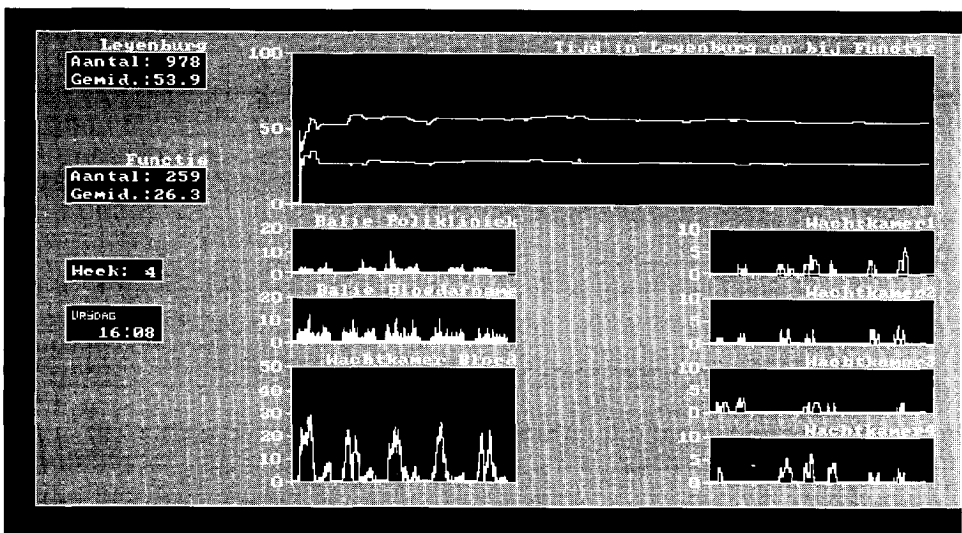


figure 5.9 Animation screen with statistical data.

To identify the phase of the handling process that a patient is in, we modelled each patient with a different color, depending on the phase she was in. Patients entering the NOD and arriving at the registration desk for the first time are dark blue. Patients arriving at the registration desk for the second time are light blue. Patients arriving at the registration desk for the third time are green. Other phases (being in the waiting room or with the physician) could be identified through the location of the patient. In order to discriminate between neurology patients and other patients visiting the Blood taking department, the latter ones are colored red.

The animation model was used for various purposes. We used it ourselves to support the verification of the empirical model and the debugging of the simulation code. In addition, we put the animation model into action in a number of sessions together with several employees in the NOD. The first session was held early in the study. A prototype of the animated empirical model was shown to the registration employees. This prototype version did not yet include the functional departments. The session was used to address a number of conceptual issues that were uncertain at that point and to perform a first structural validation of the part of the model that was finished.

The full version of the model was shown in a second session to the physicians to help them making a number of expert estimations. Also, the physician were asked to look for structural inconsistencies. The third and final session involved all neurology employees. The purpose of this session was twofold. First, we wanted to achieve structural validation. Second, a number of alternatives for changing the handling of patients had to be identified. For this purpose, an obvious change alternative had already been included in the model.

The reactions of the NOD employees to the sessions were positive. They found the animation model recognizable and easy to comprehend. They were able to quickly identify a number of inconsistencies. Furthermore, it increased their involvement in the study at that time. Before the last two sessions, the employees were merely involved passively. They answered any questions that we had and provided other information. During the second and third session, they were triggered to consciously think about their work processes and identify ways of doing things differently. We felt that an additional advantage of the third session was the increased insight in each other's work situation that arose. It became visible in what way the actors formed a chain of service providers to the patients and how they influenced each other's work load.

5.4.5 Model verification and validation

The empirical model's *correctness* was *verified* using 'tracing' techniques [Meel 1994; Pegden et al. 1990]. Individual patients of each type were generated and fed into the model. Traces of the flow of these individual patients did not yield any incorrectnesses

in terms of the required transformations and their order. The animation model was instrumental in visualizing the individual flows.

The data necessary to perform a *replicative validation* had to be collected separately. During a whole day, the total throughput time of the different types of patients was measured manually. The collected data was compared to the experimental results of the empirical model for ten replications using a *t*-test and a 95 percent confidence level. the results of the replicative validation showed no significant differences as all difference intervals contained the value zero, see table 5.12.

Patient type	Model	Observation	t-value ($\alpha=0.05$)	Significant?
Home	31.06 min	29.83 min	2.086	no
Admission	38.21 min	34.25 min	2.120	no
Just follow-up appointment	38.60 min	40.54 min	2.030	no
Functional examinations	71.22 min	66.57 min	2.025	no
Total	46.98 min	47.97 min	1.993	no

table 5.12 Outcome of the replicative validation.

As described in the previous section, the *structure* of the empirical model was validated by presenting the animation model to various NOD employees in a number of sessions. During the first and second session a number of minor structural modifications were suggested. These were implemented in the final version of the model, which was considered to be structurally valid in the third and final session.

5.5 Problem diagnosis

We performed an experiment consisting of 10 replications with the empirical model of the NOD's current situation under the treatment described in Section 5.4.3. The results of this experiment are given in table 5.13. (All times are in minutes).

Patient type	At Registration desk	At Waiting room	At Physician Mean	At Functional department Mean	Total time in NOD		Percentage
	Mean	Mean	Mean	Mean	Mean	StDev	
Home	4.94	15.92	7.56	0.00	31.06	0.50	11.38 %
Admission	4.94	15.95	7.57	0.00	38.21	0.48	7.45 %
Just follow-up	8.89	15.93	10.65	0.00	38.60	0.39	52.78 %
Examinations	13.14	15.82	16.80	20.97	71.22	0.46	28.40 %
Total	9.35	15.93	11.84	5.96	46.98	0.37	100.00 %

table 5.13 Results of the model of the current situation.

From the results it follows that the total time a patient spends in the NOD during a visit varies considerably with the type of patient. Patients that may go home, leave the NOD in 31 minutes, whereas patients requiring examinations have to spend more than twice that time on the average. Of course, patients requiring examinations have to make extra appointments and possibly have to have their blood taken. However, these activities together take a maximal average of less than 13 minutes. If we take into account that patients requiring examinations spend more time with the physician, it still means they are waiting in queues and moving through the NOD the rest of time, being 20 minutes.

A closer examination of both the experiment results and the animation model gives an even more negative picture with respect to examination patients. Visit durations of up to two hours are no exception. This is especially the case when such a patient has to pass by the Blood taking department in the morning. As a lot of blood examinations require patients to have been fasting before the blood is taken, the Blood taking department is very crowded in the morning.

The purpose of a patient's visit to the NOD is to see her physician and undergo any necessary examinations. Taking the mean time that patients arrive too early for their appointment into account, patients spent *less than half of their time* on seeing their physician or having examinations. The rest of the time they mainly spend in waiting queues, making appointments, and telling employees what appointments they made with other employees. To put it bluntly, *over 50 percent* of the time that an average patient visits the NOD is '*overhead time*': it is spent on *supporting the coordination* of the actors in the NOD. Especially examination patients or admission patients have a lot of overhead time: more than 60 percent of their visit duration.

Confronted with these results, the employees in the NOD were surprised to learn the amount of time that a patient has to spend in the NOD during a visit. In seeking for potential improvements, it was not considered appropriate to define hard quantitative solution requirements in terms of the maximum time a patient's visit may last. The NOD employees were afraid that too much emphasis on cutting down on time might have a negative impact on the (patient's perception of the) care the NOD offers. On the other hand, the employees realized that the current percentage of overhead time was out of proportion. It was, therefore, decided to look for a number of change alternatives that cut down a patient's overhead time without intervening in the overall (medical) quality of the care process. The change alternatives that we investigated are discussed in the next section.

5.6 Alternatives for change

As described above, the change alternatives in the NOD study were aimed to minimize a patient's overhead time during a visit. As this overhead time actually is spent on supporting the coordination between the NOD employees, the alternatives had to address ways in which to facilitate or improve current coordination practices. Together with the NOD employees a number of potential alternatives were identified that involved both organizational alternatives, IT alternatives, and combinations of both. They included the realization of an examination appointment facility at the neurology registration desk, a decentralization of the blood taking department, a change in the physician's schedules, and the introduction of automated appointment facilities at every workplace together with a reassignment of responsibilities.

The effects of the proposed alternatives were investigated by implementing them in the empirical model of the current situation and performing a number of experiments. The alternatives and their evaluation are discussed in the following subsections.

5.6.1 *A decentralized examination appointment facility*

The first alternative involved a decentralization of the appointment facility for functional examinations. It was recognized by the NOD employees that in the current situation there is a certain overkill: patients who need examinations, first have to go to the appropriate departments to make an appointment and then have to come back to the registration employee to make a follow-up appointment with the physician. The rationale behind this procedure is that (1) a follow-up appointment cannot be done before the examination results are available, and (2) the appointment facility of the functional departments is not accessible for the neurology registration employees. Both the neurology registration employees and those of the functional departments found it not acceptable to have all examination appointments made by phone as is done for patients with poor mobility. They were afraid this would clog up the phone line too much for other internal and external calls.

Therefore, we proposed an alternative where all functional appointment facilities would be linked in such a way that the neurology registration employees could make examination appointments as well. The effects of this alternative were evaluated by performing a similar experiment as with the model of the current situation. There were 10 replications in each experiment. The results are given in table 5.14 (all times in minutes). We left out the waiting room phase and the physician phase as the times in these phases did not change notably. A *t*-test was performed with a significance level of $\alpha=0.05$ to check whether the changes in the patient's visit duration could be considered significant.

Patient type	At Registration desk		At Functional department		Total time in NOD				Significant? ¹
	C.S.		C.S.		C.S.		Alt.1		
	Mean	Mean	Mean	Mean	Mean	StDev	Mean	StDev	
Home	4.94	5.30	0.00	0.00	31.06	0.50	31.42	0.91	no
Admission	4.94	5.31	0.00	0.00	38.21	0.48	38.31	0.79	no
Just follow-up	8.89	9.64	0.00	0.00	38.60	0.39	38.96	0.60	no
Examinations	13.14	15.03	20.97	15.67	71.22	0.46	63.72	0.93	yes
Total	9.35	10.38	5.96	4.50	46.98	0.37	45.18	0.68	yes

C.S. = Current Situation, Alt.1 = 1st Alternative

¹t_{0.025}(18) = 2.101*table 5.14 Results of the first change alternative.*

The results show that there is a statistically significant decrease in the duration of a patient's visit. However, this decrease can only be considered considerable with respect to patients requiring examinations. Yet, these patients still spend a considerable time in the functional department phase. This is because a lot of patients need to have their blood taken for examination. For this, they do not need an appointment; they have to go there immediately.

Another noteworthy observation is that although the number of patient visits to the registration employee decreases because of this alternative, the time spent with the registration employee goes up (significantly) for all patients. Apparently, the queue in front of the registration employee grows considerably because of the extra time the registration employee needs to make examination appointments for patients.

5.6.2 Decentralized Blood taking

In order to further reduce the overhead time of examination patients, we proposed an organizational alternative on top of the first one. This alternative comprised of a decentralization of the Blood taking department. The registration employee would be medically trained to take a patient's blood. The blood laboratory could have tubes picked up at convenient times during the day. As a result, examination patients would no longer have to go to any functional department; they would be handled in one location. This alternative is not as revolutionary as it may seem. In a lot of hospitals it is, for example, not uncommon to have medically trained staff at a registration desk. The results of the second alternative are given in table 5.15. Again, there were 10 replications in each experiment.

The results show a dramatic improvement for examination patients. On the average, they only have to spend about 6 minutes to have their blood taken, compared to 21 minutes in the current situation. As a result, the average visit duration of all patients also drops significantly with about 13 percent. However, notwithstanding the overall improvement, this alternatives has some negative side-effects. It appears that the

Patient type	At Registration desk		At Functional department		Total time in NOD				Significant? ¹
	C.S.		C.S.		C.S.		Alt.2		
	Mean	Mean	Mean	Mean	Mean	StDev	Mean	StDev	
Home	4.94	5.25	0.00	0.00	31.06	0.50	31.45	0.61	no
Admission	4.94	5.26	0.00	0.00	38.21	0.48	38.38	0.83	no
Just follow-up	8.89	9.55	0.00	0.00	38.60	0.39	39.07	0.40	yes
Examinations	13.14	12.24	20.97	5.94	71.22	0.46	47.94	0.47	yes
Total	9.35	9.50	5.96	1.69	46.98	0.37	40.68	0.42	yes

C.S. = Current Situation, Alt.2 = 2nd Alternative

¹t_{0.025}(18) = 2.101*table 5.15 Results of the second change alternative.*

average visit duration of the other patient types is slightly increasing. The increase for patients requiring just a follow-up appointment is even statistically significant. The experiment results suggest that this is because the time patients spend in front of the registration desk is increasing. A closer look at the animation screens confirms this: several times during the day, the queue in front of the registration desk is growing rather long, sometimes up to 16 people. It seems that if further efforts to bring down the visit duration put extra strain on the registration employees, the negative side-effects will be greater than the benefits.

5.6.3 A smoother physicians' schedule

The above observations led to a third alternative: a rearrangement of the physicians' visit hours. The animation model clearly showed the relationship between long queues in front of the registration desk and the physicians' visit hours: when all four physicians were working, the registration employees had a difficult time preventing long waiting times in front of their desk.

In this respect, the schedules of the physicians can be improved. Some parts of a day all four physicians have visit hours, while on other parts, there are no visit hours at all, see table 5.9. The third alternative, therefore, proposed a rearrangement of the schedule in such a way that both in the morning and in the afternoon, the minimum and maximum number of physicians having visit hours would be 2 and 3 respectively. The impact of this alternative can be deduced from the results in table 5.16. Again, there were 10 replications in each experiment.

Although compared to the current situation the results show a significant improvement for the average patient, the alternative does not yield the results that were expected compared to the second alternative. The visit duration of patients that require just a follow-up appointment does decrease significantly compared to both the current situation and the second alternative, but the improvement cannot be considered

Patient type	At Registration desk		At Functional department		Total time in NOD				Significant? ¹
	C.S.		C.S.		C.S.		Alt.3		
	Mean	Mean	Mean	Mean	Mean	StDev	Mean	StDev	
Home	4.94	4.56	0.00	0.00	31.06	0.50	31.19	0.56	no
Admission	4.94	4.60	0.00	0.00	38.21	0.48	38.61	0.80	no
Just follow-up	8.89	8.04	0.00	0.00	38.60	0.39	38.11	0.36	yes
Examinations	13.14	10.92	20.97	5.95	71.22	0.46	47.26	0.51	yes
Total	9.35	8.23	5.96	1.73	46.98	0.37	40.05	0.38	yes

C.S. = Current Situation, Alt.3 = 3rd Alternative

¹t_{0.025}(18) = 2.101

table 5.16 Results of the third change alternative.

substantial. A closer examination of the results showed that the improvement at the registration desk just made patients arrive in the waiting room sooner and subsequently spend more time there. Hence, the total visit duration does not change substantially.

Looking at the three alternatives we investigated so far, it appears that improving parts of the chain of care activities in the NOD only brings about improvements for certain patient types or certain phases. At the same time, it may deteriorate the situation for other patient types or introduce bottlenecks at other phases in the process. For this reason a final, more radical alternative was proposed, which is discussed in the next section.

5.6.4 Reassignment of responsibilities and an integrated appointment facility

The final alternative seeks to decrease the average visit duration by taking all patients types into account. The alternative builds on the second and third alternative in the sense that it is assumed that the registration employees are trained to take blood. The fourth alternative has a number of components. First, there is the IT component. There will be an automated appointment facility, that is accessible for all actors in the NOD. It will allow every actor to make appointments for follow-up consults, for examinations, and for admissions.

Second, there is an organizational component. The tasks and responsibilities of the actors will be adapted. With new appointment facility, *physicians* will be able to easily make any appointment they want the patient to have. Instead of writing all information about required appointments down on various notes and in patient files, a physician will now enter the appointment requirements in the appointment facility. In return, the facility suggests a few options, one of which is selected and printed in duplicate: one (adhesive) printout for the patient's appointment card, and one for the patient's file. This implies that only patients needing to have their blood taken, have to return to the registration desk. All other patients are done after seeing the physician. There will also be changes for the *secretary*. She will join the employees at the registration desk. This

way she can help out at times there are a lot of patients to handle. Of course, she will no longer have to handle patients that need to be admitted.

The rationale behind this alternative is that information to coordinate the remaining activities with respect to a patient after she is seen by the physician, should not be carried around through the hospital. It should be acted upon as soon as possible. By providing the right type of automated support, acting on this information does not need to take more time than writing it down in the current situation. The results with respect to this alternative are presented in table 5.17. Again, there were 10 replications in each experiment.

Patient type	At Registration desk		At Functional department		Total time in NOD				Significant? ¹
	C.S.		C.S.		C.S.		Alt.4		
	Mean	Mean	Mean	Mean	Mean	StDev	Mean	StDev	
Home	4.94	4.30	0.00	0.00	31.06	0.50	30.41	0.61	yes
Admission	4.94	4.31	0.00	0.00	38.21	0.48	29.90	0.71	yes
Just follow-up	8.89	4.12	0.00	0.00	38.60	0.39	32.92	0.53	yes
Examinations	13.14	8.58	20.97	3.81	71.22	0.46	43.74	0.40	yes
Total	9.35	5.44	5.96	1.09	46.98	0.37	35.52	0.37	yes

C.S. = Current Situation, Alt.4 = 4th Alternative

¹ $t_{0.025}(18) = 2.101$

table 5.17 Results of the fourth change alternative.

The results for the fourth alternative are encouraging. The average visit duration is significantly less for all types of patient compared to the current situation. The overall visit duration drops with almost 25 percent to 35.5 minutes. Only the improvement for patients that may go home is not substantial. This was, however, expected as this type of patients does not need to carry around appointment information.

Taking the average time patients arrive too early for their appointment into account, one can see the overhead times going down considerably as well. In the current situation, patients may have on the average between 15 and 60 minutes of overhead time. With the fourth alternative this is never more than 5.5 minutes. Relatively, patients requiring the most attention, examination and follow-up patients, would never spend more than 30 percent of their time waiting in queues or moving around. In the current situation this is 60 percent.

An additional advantage of the implementation of an integrated appointment facility would be that it enables the recording of the data that is needed to further evaluate the handling of the various patient flows in the NOD. In the current situation there is hardly any data available to make such an evaluation possible. An important aspect would be recording the overall treatment of a patient, not just the individual visits. In

this study, only individual visits could be taken into account as there was no way to retrieve data over multiple visits of the same patient. There might well be ways to improve the patient-friendliness of the NOD from this perspective. In summary, an integrated appointment facility and the use of it as described above, would enable the NOD to have *review* coordination processes in addition to the current anticipatory ones; as such, it would facilitate the further investigation of possibilities to enhance patient-friendliness and to handle different types of patient flows appropriately.

5.7 Reflection

The NOD proved to be a dynamic environment for testing and enhancing the dynamic modelling approach. Our reflections on this case study address the NOD situation itself and the approach that we employed.

5.7.1 *The NOD*

In our study, we took a predominantly efficiency angle on patient-friendliness. Obviously, there are other aspects as well. These include, for example, the layout and appearance of the interior of the hospital, the way in which staff members approach patients, the quality and appropriateness of the medical treatment, and the general atmosphere in the hospital. As these aspects are difficult to quantify, we did not take them into account in this study. These aspect also make patient-friendliness a subjective issue as well, especially from the point of view of the patients. If Leyenburg is to seriously address the issue of patient-friendliness any further, it will have to take both quantitative and qualitative aspects into account and, if possible, directly involve a number of patients in this process as well.

Unfortunately, the study at the NOD ended with the identification and evaluation of the four change alternatives. During the study, the hospital management changed both in terms of composition and in terms of buy-in for our study and the overall project it was part of [Ramondt and Rhébergen 1991a and 1991b]. As a result, the outcome of the study was never presented to the hospital management. This is unfortunate in the sense that we felt a demonstration of the empirical model of both the current situation and of the alternatives, would have been a valuable contribution to the ongoing discussion on improving patient-friendliness by visualizing, quantifying, and evaluating flows of patients.

5.7.2 *The dynamic modelling approach*

Like the DUB case, the main purpose of the NOD case was to apply, evaluate, and enhance the dynamic modelling approach. We further developed the concepts for

modelling organizational processes and their coordination. In addition, we developed a transformation procedure to translate the conceptual model elements into pieces of simulation code. With this procedure, the simulation code was quickly written, and the various alternatives were easily incorporated, even if it required structural changes of the model.

There are, however, two issues that turned out to be a problem and required further attention. First, with respect to the way of working, it appeared that the *availability of empirical data* was a critical success factor for the application of the approach. For lack of available and useful data in this study, the construction of the empirical model of the current situation became extremely time consuming. During this period it proved to be a challenge keeping the NOD employees interested and motivated to provide us with various information without promptly getting tangible results back in return.

The second issue relates to the way in which dynamic modelling's way of working and way of modelling support the involvement of stakeholders in a study. Contrary to the DUB case, the NOD represented an excellent opportunity to investigate the *support of stakeholder participation with animation models*. Our experiences in this respect are twofold. On the one hand, the animation model clearly facilitated the construction and validation of the empirical model. Moreover, it created an awareness with the NOD employees about their current situation and stimulated the generation of potential alternatives for change.

On the other hand, the animation model *on its own* proved to be incapable of efficiently involving the stakeholders and surfacing all their relevant tacit knowledge. Partly due to the data availability problem, the animation model could only be used to support the stakeholders' involvement at a late stage in the project and for a short period of time. Especially during the earlier stages of the study, most of the stakeholders' tacit knowledge was required to build the conceptual and empirical model of the problem situation. Furthermore, the animation model only *triggered* the discussion on the NOD situation. Extra support is required to actually manage and structure the subsequent group interaction processes that would make the study a real participative effort.

6 Group Support for Participative Design

'Wallenquist made a big thing of the personal touch. He deplored communication through electronics. It was too impersonal, too aloof. "Barriers go up then, man! Wires, waves, screens! You can't really know a person or like him or get him to know you and like you if you're talking through machines. You're ghosts then. What we need is flesh and blood, man. Touch and smell."'
- Philip José Farmer

6.1 Introduction

In the previous two chapters we discussed the application of the dynamic modelling approach within two case studies. The experiences in these studies were used to extend and modify the initial dynamic modelling approach. The result of this exercise was described in Chapter 3.

Notwithstanding the fact that some points for future consideration remained, we decided to apply the new theory on dynamic modelling prescriptively in a third case study, which was carried out at the Criminal Investigations Department of the Amsterdam Municipal Police Force. During this study it became apparent that the dynamic modelling approach was not appropriate to handle all aspects of the intricate organizational change process in this particular case situation. The reasons why this was the case, are summarized below. For a more elaborate discussion of the third case study and our experiences, we refer to Chapter 7:

1. The organization was in a constant flux. There was no stable 'current situation' that represented a yardstick to evaluate change alternatives.
2. It was practically impossible to collect empirical data for a simulation model.
3. Additional techniques were required to develop IT to support coordination in the new organization.

With this in mind, we decided during the study to introduce a *new theory* to support the remainder of the organizational change process in this case. Hence, the second part of the case study was used *descriptively* to investigate the usefulness of this theory. The new theory not only had to address the three issues that we mentioned above, but

it also had to take into account the points of consideration that were raised with respect to the dynamic modelling approach (see also Chapters 4 and 5):

1. The elicitation of problem owners' tacit knowledge and opinions can be a very time consuming activity, especially in situations where a great number of stakeholder groups are involved.
2. Although the use of animation models is likely to enhance and facilitate problem owner participation in a study, this participation seems to be limited to a few occasions during the later stages of a study, namely during the validation of the empirical model and the identification of potential change alternatives. In addition, we identified a need to manage the group processes triggered by an animation model.
3. There is a requirement for additional techniques and methods to further support the design and development of IT solutions as part of options to change organizational processes and their coordination.

In order to address these issues and handle the problems we encountered during the third case study, we decided to employ techniques and support for group design. We embedded these techniques and support in the way of thinking and way of working of the dynamic modelling approach. For sake of simplicity, we refer to this combination as *the participative design approach*. This approach represents the second initial theory that we used, evaluated, and extended during our dissertation research. It focuses on the issue of stakeholder participation in design activities to facilitate the process of organizational change and information systems development. The involvement of groups in such design processes is often considered to be a prerequisite for the success of such attempts, see e.g. [August 1991; Emery and Emery 1974; Engelstad and Gustavsen 1993; Mumford 1983; Vitalari and Dickson 1985]. Initially, the participative design techniques and support were based on the GRIPPPER approach as proposed by Wanninger and Dickson [1992a and 1992b], which they developed to support the incremental, participative development of management information systems. During the case study, we modified and extended the techniques and support to handle the specific problems that we encountered. In particular, the resulting participative design approach aims to:

- Enhance and facilitate the participation of problem owners during the majority of the design activities within a process of organizational change.
- Support the construction of empirical models and diagnose organizational problem situations with techniques that do not exclusively rely on the collection of statistical empirical data.
- Address the design and development of IT to support the coordination of new and modified work processes in the organization.

In the remainder of this chapter, we describe the techniques and support for group design in more detail. The material in this chapter does not only reflect the basics of the GRIPPER approach and the previous experiences with it, it also represents the modifications and extensions that were brought about by evaluating the participative design approach during the second part of the third case study. In the next section, we first address the type of group support technology that serves as a cornerstone in the approach to facilitate problem owner participation in the design process. Then, we elaborate on the techniques in the approach in the context of its way of thinking, its way of working, and its way of modelling in Section 6.3, Section 6.4, and Section 6.5 respectively. The chapter concludes with a discussion on how we propose to employ group support technology for the various design activities.

6.2 Group Support Systems

Over the past decade, teams or groups are becoming a predominant form in which organizational activities take place [Drucker 1989]. Organizations find themselves in an environment where they have to take an increasing amount of decisions, in a faster way [Huber 1984]. At the same time, the increasing political and technical complexity of organizational decisions sets the requirement for participative decision making [Huber and McDaniel 1986]. This also brings about new requirements for technologies supporting groups in an organizational setting. The increasing occurrence of networked computers and communication technology has provided the opportunity for new technologies that support groups with a multitude of organizational tasks. Often the term *groupware* is used to denote such technologies [Ellis et al. 1991; Johansen 1988; Oppen and Fersko-Weiss 1992], ranging from electronic mail, messaging, and bulletin boards to group project management, conferencing, and workflow management systems, see also [Coleman 1994].

A subset of groupware concerns information technology that aims to make group meetings and group decision making more productive. Originally, this particular instance of groupware was regarded as a special type of a DSS, a so-called Group Decision Support System (GDSS) [Gray et al. 1981]. In recent years the term Group Support System (GSS) is most often used to denote such systems, because by definition this term does not limit the application of the technology to a specific group task, such as decision making. Other terms that are often used include EMS (Electronic Meeting System), DGSS (Distributed Group Support Systems), CMCS (Computer Mediated Communication Support), Group Negotiation Support Systems (GNSS), GCSS (Group Communication Support System), CAC (Computer Assisted Communication), and GIS (Group Information System).

6.2.1 The nature of GSS

We follow Jessup and Valacich in defining GSS as ‘computer-based information systems used to support intellectual collaborative work’ [1993, p.5]. By intellectual collaborative work they mean work that takes place in a meeting. The roots of GSS can be traced back to the late 1960s and early 1970s when the first electronic GSS were developed [Beer 1973; Johansen et al. 1979; Johansen 1988; Wagner et al. 1993]. Over the years a great deal of research effort has been devoted to the development of GSS. At this moment, there are at least 20 true GSS software products available [Bostrom et al. 1992a]. These can be categorized in three types of GSS [Bostrom et al. 1992b; Watson et al. 1994]:

1. *Chauffeured GSS*. A chauffeured GSS features a single workstation and equipment to project information on a public screen. The workstation is operated by the meeting facilitator or by a dedicated chauffeur. The chauffeur types in everything that is said and projects selected information on the public screen. Sometimes, chauffeured GSS are also referred to as facilitated GSS.
2. *Keypad GSS*. This type of GSS resembles a remote control. Participants have a keypad with which they can type in numeric responses to questions that are displayed on a public screen. Their responses are radiographically transmitted to a central workstation, that is able to display the voting results on the public screen. Keypad GSS are only suitable for voting sessions. They are especially useful in large group settings aimed at surfacing opinions.
3. *Workstation GSS*. With workstation GSS, each member of a meeting uses his or her personal workstation that is linked to the other workstations through a network. Workstation GSS usually support a broad variety of meeting activities electronically, such as exchanging ideas, voting, and brainstorming. Workstation GSS are sometimes also referred to as interactive GSS.

Chauffeured GSS		Keypad GSS	
COPE Resolver	University of Strathclyde Tandem Corporation	OptionFinder Vision Net	Option Technology Leadership 2000
Workstation GSS			
EIES2 GroupSystems V and GroupSystems for Windows SAMM SAGE VisionQuest	New Jersey Institute of Technology Ventana Corporation University of Minnesota National University of Singapore Collaborative Technologies		

table 6.1 Examples of the three GSS types.

A number of examples of each type of GSS and its manufacturer is listed in table 6.1. In the remainder of this chapter, the discussion will be limited to workstation GSS. This is because in *participative design* settings, keypad GSS and chauffeured GSS have a number of disadvantages compared to workstation GSS:

1. Keypad GSS do not offer support for generating and organizing information. These activities can be expected to represent a significant part of participative design meetings. In addition, not all voting techniques can be supported with keypad GSS, e.g. point allocation or multiple choice.
2. Chauffeured GSS do not allow group members to participate anonymously. Field studies suggest that anonymity increases both the quality of the meeting results and the satisfaction of the participants with the meeting itself [Dennis and Gallupe 1993].
3. Chauffeured GSS does not support parallel information gathering. Experiences with workstation GSS indicate that parallelism yields increased group productivity and satisfaction [Dennis and Gallupe 1993].

A workstation GSS is usually part of a Group Support Facility (GSF) consisting of the following components [DeSanctis and Gallupe 1985; Vogel et al. 1987]: furnishings, audiovisuals, software, hardware, and problem solving techniques and strategies. In our research, we used the GSF of the School of Systems Engineering, Policy Analysis and Management (SEPA). This facility is depicted in figure 6.1.

SEPA has outfitted two GSFs. The layout of each GSF is identical and based on the 'classical' scheme developed and tested at Claremont Graduate School and at the University of Arizona: a meeting room with a U-shaped table facing a public screen, flanked by a number of break-out rooms, see e.g. [Gray 1986; Nunamaker et al. 1991]. The table consists of 15 slightly wedge-shaped segments. Each segment has a sunk-in CRT and a IBM compatible Personal Computer fixed to one of its supports. The 15 Pcs are connected to a server running under Novell 3.11. The server is dedicated to the combined GSF in such a way that a joint session involving both facilities can be conducted. The public screen has been implemented using a beam projector placed overhead, powerful enough to generate a crisp image even in daylight. The GSF infrastructure serves as a platform for several applications that are used, both for research and teaching purposes. The specific GSS currently in use are Ventana's GroupSystems V, extended with a multi-criteria analysis tool developed by SEPA, GroupSystems for Windows, and TeamGraphics. In addition, various stand-alone applications are available such as text editors, spreadsheet, and database applications.

With the appropriate (tele)communication network connections, workstation GSS can be used to support four types of meeting settings in terms of member proximity and duration [DeSanctis and Gallupe 1985, 1987; Johansen 1988, 1992]:



figure 6.1 SEPA's Group Support Facility.

1. *Same time, same place.* These types of meetings, better known as face-to-face meetings, are predominant in most organizations.
2. *Same time, different place.* These types of meetings are becoming increasingly popular with the advent of cheaper and more powerful communication technology, such as conference calling and video technology.
3. *Different time, same place.* These types of meetings usually take place in a dedicated environment, or team room, used by a team working together on a joint task or project.
4. *Different time, different place.* These types of meetings occur when a group of people have to perform a joint task (e.g. writing a joint report or joint software development), but do not have the possibility or the need to have joint meetings.

The ongoing developments in communication technology are expected to make the boundaries between the above four types of meeting faint and artificial. Future meeting environments are likely to be based on portable machines that communicate through wireless technology. Having access to the same tools, data, and displays being used in traditional meetings, meetings will be 'any time, any place' [Nunamaker et al. 1994].

6.2.2 Experiences with GSS

Since its appearance in the early 1980s, GSS have grown to become a popular area for researchers in a variety of disciplines. Most of this research is carried out as experimental research. A review by Pervan [1994] reveals that of most studies into GSS, 35 percent, are lab experiments. This type of research was especially popular in the earlier of years of GSS research: A review by Pinsonneault and Kraemer [1989] shows that up to 1988 about two thirds of all GSS studies were lab experiments. Of the more recent studies (1989-1990) about one third are experimental research [Zigurs 1993]. The positive findings in the earlier experimental studies appear to be paving the way for commercial applications of GSS. Over the past few years, GSS have been moving out of the laboratory settings into a variety of real world application areas, some examples of which are listed in table 6.2.

Application area	References
business process redesign	[Nunamaker and Briggs 1994; Vreede et al. 1995]
case analysis	[Krcmar et al. 1994]
education	[Alavi 1994; Bostrom et al. 1992; Briggs et al. 1993; Dickson et al. 1992; Gallupe 1992; Gopal et al. 1992; Hiltz and Turoff 1992; Nunamaker et al. 1992]
gaming and training	[Gray and Borovits 1986]
information planning, policy formulation	[Bots et al. 1994; Herik 1995]
information system development	[Bostrom et al. 1992b; Davison 1995; Dennis et al. 1995; Lewis and Newton 1995; Vreede et al. 1995b and 1995e; Wanninger and Dickson 1993a, 1993b]
process quality assessment and improvement	[DeSanctis et al. 1991; Gallupe 1982; Krcmar et al. 1994; Nunamaker and Briggs 1994; Vician et al. 1992]
process modelling	[Dennis et al. 1994; Vogel et al. 1994]
product development	[Muller and Vreede 1995a and 1995b]
project design and evaluation	[Krcmar et al. 1994; Qureshi and Cornford 1994]
resource allocation	[Volpato 1989]
strategy planning and strategy creation	[Atkinson and Marshall 1995; Bostrom and Anson 1992; Eden and Ackermann 1992; Gallupe 1982; Gray 1992; Lewis and Lewis 1994; Tyran et al. 1992; Volpato 1989]

table 6.2 Examples of GSS application areas.

As a result of field studies and a large number of experiments, a substantial body of knowledge has emerged about the effects of GSS on group processes and outcomes, especially with respect to GSS used in same time, same place settings. Compared to 'traditional' group settings, these types of GSS meetings appear to be more efficient and more effective, while the participants are more satisfied with the results and the process, see e.g. [Tyran et al. 1992; Wanninger and Dickson 1992a]. Several attributes have been identified that appear to bring about differences between GSS meetings and non-GSS meetings. These attributes can be classified into four categories [Nunamaker

et al. 1993]: ‘group’ (e.g. group size or group proximity), ‘task’ (e.g. idea generation or decision choice), ‘context’ (e.g. culture or reward structure), and ‘GSS’ (e.g. anonymity or media types). Together these categories constitute the basis of a research model (see figure 6.2) that, in different variations, has been used in many GSS research programs, see e.g. [Bostrom et al. 1992a].

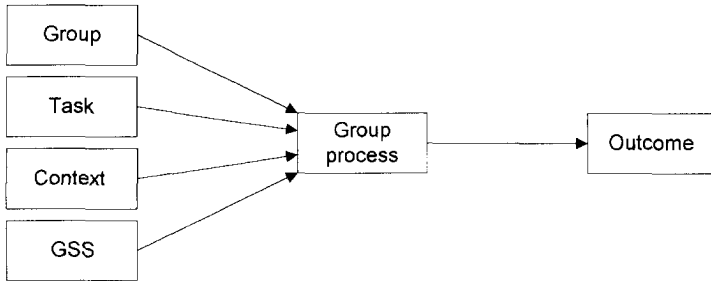


figure 6.2 GSS research model [Nunamaker et al. 1993].

Below we discuss a number of attributes that have been found to have a significant impact on group process and process outcome. For reasons of simplicity, we classify these attributes in two categories: attributes of the *GSS itself*, and attributes of the *setting* in which the GSS is employed.

GSS attributes

With respect to the GSS itself, we can identify four attributes that are said to be responsible for bringing about the beneficial results that are reported [Dennis and Gallupe 1993; Dennis 1994; Nunamaker et al. 1991]: Anonymity, parallelism, group memory, and media richness. However, we will argue that in some situations these attributes may actually have no effect or even have a negative effect on the group process or the outcome of this process.

For student groups in laboratory settings *anonymity* did not have much of an impact. However, in field settings anonymity has proven to be very useful, especially for nonpeer groups. By being able to enter ideas, comments, and votes anonymously, silent or shy participants are more encouraged to enter their ideas. Group members are hindered from dominating by position or personality. Ideas appear to be judged on their merit, not on the personality of the participant that entered it. Creative, non-conformist ideas are more easily entered. On the other hand, anonymity may sometimes turn out to be an obstacle if generated ideas or voting results need clarification. Those who submit information may be reluctant to reveal their identity, which may result in information remaining unclear or even being deleted [Lewis and Newton 1995; Vreede and Sol 1994b].

In electronic group meetings, participants can generate ideas and communicate them in *parallel*. First, this implies that participants get equal 'air time' because they do not have to listen to others before they can submit their own views. Second, parallel communication prevents production blocking because participants do not have to remember their ideas and criticisms until they have the floor and can therefore spend more time on generating new ideas. Diehl and Stroebe [1987] showed that in non-GSS groups, production blocking accounts for most of a group's productivity loss. They argue that brainstorming, as introduced by Osborn [1957], is more effectively done *individually* before discussing the results with the group as a whole. Third, it has been found that because of the ability to work in parallel, GSS groups generate more ideas in a given amount of time. In this respect, it is important to note that the number of ideas of high quality appears to be directly proportional to the total number of ideas [Valacich et al. 1994].

In spite of these advantages, the increase in productivity that follows from parallel communication, may imply information overload, see [Dennis 1994]. In this sense, parallelism may sooner prove to hinder the group process than to move it forward. In order to reduce the initial information overload in GSS meetings, special brainstorming techniques can be advocated, for example Nominal Group Technique (NGT) [Van de Ven and Delbecq 1974]. Participants initially brainstorm individually and then submit their ideas to the group one at a time. Note that this technique seems to anticipate Diehl and Stroebe's findings.

All ideas, comments, and votes that are entered during a meeting are stored electronically, thereby creating a *group memory*. A benefit of group memory is that the results of previous meetings are readily available when a group continues from where itself or another group left off. Another benefit is the significant decrease in the amount of time needed to produce the meeting minutes and/or meeting results. At this point it is important to note that electronic group memory differs from 'traditional' minutes in several important aspects [Culnan and Markus 1987]. First, as minutes are traditionally written by a single individual, they are likely to reflect that individual's understanding of the meeting outcome. The electronically generated group memory gives an objective history. Second, minutes do not often describe the participant's initial positions or their interim postures. Group memory depicts the evolution of the group's position over time. Finally, group memory makes each comment during a meeting public, to which others can refer at a later stage.

With respect to group memory, the other side of the coin is that, people may hesitate to contribute if everything is stored immediately. It may be the case that the more sensitive brainstorming and decision making processes are impaired if *all* of the output is documented. We suggest that people's positions on certain topics may either become

more rigid or less outspoken for fear of not having the possibility of changing one's point of view later on in the process.

In terms of *media richness* GSS are often said to represent a poor medium, since information is only communicated electronically [Daft et al. 1987]. With a lack of *social cues*, the subtle nuances of behavior that affect group activity no longer play a crucial role on the group process and certain types of information, such as sarcastic remarks, may be interpreted out of context. This may jeopardize the level of buy-in or the extent to which participants feel comfortable with the resulting decisions.

However, this need not necessarily be a problem, as it depends on the task a group has to perform: generative tasks can be adequately supported by poor media such as GSS, whereas judgement tasks or negotiation processes normally require richer media, such as face-to-face communication [McGrath and Hollingshead 1993]. One way of overcoming this potential problem is by not employing GSS as a replacement for a richer medium, but as an addition to it. In this way, settings where GSS is combined with other media can, as a whole, be considered to represent a rich medium [Jarvenpaa et al. 1988]. Another way is to spend a significant part of the meeting on non-GSS supported activities, such as the oral discussion of intermediary results [Vreede et al. 1995b].

GSS setting attributes

Apart from the effects brought about by the characteristics of GSS technology, the setting in which the technology is used, influences meeting outcomes as well [Dennis and Gallupe 1993]. Characteristics of a GSS setting that are said to have a significant impact on meeting outcomes and group productivity include facilitation, group size, group experience with GSS, and culture.

Facilitation is defined as a set of functions or activities carried out before, during, and after a meeting to help the group achieve its outcomes [Bostrom et al. 1993]. Frequently, facilitation is identified as a significant attribute of the setting in which GSS technology is used [Dennis et al. 1991; Grohowski et al. 1990; McGoff and Ambrose 1991; Kraemer and King 1988]. An important goal of facilitation is to provide *structure* in meetings where messy problems are defined and handled. However, studies have shown that facilitation *per se* does not guarantee improved meeting process and outcome [George et al. 1992]. *Rigid* facilitation, for example, does not seem to fit judgmental task situations, i.e. decision making tasks without right answers, such as multi-criteria decision making sessions [Dickson et al. 1993]. In a later study, Dickson et al. [in press] have shown that facilitation has a significant positive effect on meeting outcomes when (1) facilitation is *harmonized* with the task the group is performing, and (2) facilitation is *flexible*, i.e. the behavior and actions of the facilitator depend on what happens during the meeting. Other studies support these

findings [Anson and Heminger 1991; Anson et al. in press]. A particular drawback of facilitation may be that it can dictate rigid structure to an extent that some groups may feel uncomfortable working under such an imposed structure [DeSanctis and Poole 1994]. Opposed to human facilitation, Limayen et al. [1993] evaluated *automated* facilitation. They found that a simple form of automated facilitation that resembles flexible human facilitation, is as effective as its human counterpart.

Another significant attribute of the setting in which GSS are applied concerns *group size*. Findings consistently have shown that GSS have a greater positive impact on larger groups in terms of productivity and satisfaction of the members, see e.g. [Dennis 1994; Dennis et al. 1990; Nunamaker et al. 1988]. Small groups usually have up to at most 8 members, large groups consist of as many as 30 members. A practical consideration with respect to large groups concerns meeting organization. In real life settings, meetings with large groups may involve serious agenda problems in terms of bringing all group members into a meeting, see e.g. [Vreede et al. 1995b and 1995c].

Users' *GSS-experience* may be of significant impact on meeting outcomes. As with most technologies, users have to go through a learning process when using GSS. Participants' comfort with the technology seems to increase over time [Vician et al. 1992]. This may motivate groups to change the way in which they adopt the GSS technology over time to better meet their requirements [DeSanctis and Poole 1994]. Also, participants' attitude toward the technology tends to become more positive over time [Zigurs et al. 1990]. A study by Chidambaram and Bostrom [1993] over a limited period of time illustrates that a GSS supported group became more productive over time compared to a manual group. However, apart from these initial findings, little is yet known about whether more experienced groups perceive better meeting outcomes and processes than first-time users.

Finally, an important attribute that until recently has not received much GSS research attention, concerns the *culture* of the group. Most research on GSS groups has focused on American subjects or situations [Pinsonneault and Kraemer 1989; Raman and Wei 1992]. Yet, research in other disciplines, such as psychology, sociology, and group behavior, have shown that groups from different cultural backgrounds display different forms of group behavior, see e.g. [Hofstede 1980]. Especially when looking at American/Western, Japanese, Taiwanese, and Singaporean groups, one recognizes different procedures according to which these groups meet and reach consensus or take decisions [Raman and Wei 1992]. This implies (1) that GSS research findings need not necessarily be applicable to non-American groups, and (2) that GSS functionality should be carefully matched to the behavior of a group from a specific cultural background. For example, anonymity may not be desirable in settings in which it is not acceptable to express openly negative opinions [Raman and Wei 1992].

Given the above discussion on GSS (setting) attributes, it is not surprising that the nature of GSS meetings has been seen to differ from traditional meetings. Instead of being occasions in which work is planned to be carried out, GSS meetings appear to be occasions where the work is *actually done* [Nunamaker and Briggs 1994]. This brings about the most important critical success factor of conducting successful electronic meetings: meeting planning [Dennis 1994; Vreede and Sol 1994b]. In this respect, the key issue is not what the GSS technology *can* do. The key issue is how a *meeting process can be designed* in such a way that it takes maximum advantage of the potential benefits of using GSS while at it recognizes and anticipates potential risks [Kumar et al. 1995; Ven 1989]. The next section presents a model of meetings in terms of a classification of tasks that are performed during meetings. By mapping the functionality of different GSS to this taxonomy of meeting activities, we have an instrument to design and plan GSS supported meetings.

6.2.3 A general description of GSS functionality

We define a meeting as a sequence of tasks or activities performed by two or more people. Although, meeting processes cannot be defined or generalized [Morrison and Vogel 1991], they can be described in terms of task taxonomies by defining a set of atomic tasks that take place in group meetings. Several taxonomies have been proposed in the literature. Within the GSS field, the two most widely supported are those of McGrath [1984; McGrath and Hollingshead 1993], and of Bostrom and his colleagues [Bostrom and Anson 1992; Bostrom et al. 1993]. Both taxonomies represent a number of basic activities that may occur in random order several times during a meeting. McGrath distinguishes between four different types of group tasks, each divided into two subtypes:

1. *Generate*. Groups often have to come up with ideas, suggestions or viewpoints. This type of activity can be seen as 'information generation' and is a very common task performed by groups in organizations. According to McGrath's taxonomy, generating information is a *creativity* task, resulting in ideas, or a *planning* task, resulting in action plans.
2. *Choose*. Another common activity during group meetings concerns choosing between alternatives (ideas, plans etc.). Alternatives can be chosen in a myriad of ways, ranging for simple Yes/No voting methods to complex multi criteria decision making techniques. Choosing alternatives is an *intellective* task when there is a correct alternative to choose. If there is no objective criterion of correctness, choosing is a *decision making* task, also known as a judgmental or preference task [DeSanctis and Gallupe 1987].
3. *Negotiate*. As groups often represent different parties, they may find themselves negotiating a (set of) solution(s). During this activity, groups perform *cognitive*

conflict tasks to resolve conflicting viewpoints, or *mixed-motive tasks* to resolve conflicting interests.

4. *Execute*. Groups can come together to execute certain tasks. Executive tasks take the form of *contest* tasks competing for victory over another group, or of *performance* tasks striving to meet external standards of performance.

According to Bostrom and his colleagues, meetings can be seen as a sequence of information processing activities. They distinguish between four basic information processing activities:

1. *Generate*. This activity can be considered to be the same as in McGrath's taxonomy.
2. *Organize*. When a group has generated information, e.g. a list of ideas, this information may have to be organized. Consequently, duplicate information items are removed, non-relevant items are deleted, and items showing some common characteristics are grouped or categorized.
3. *Evaluate*. This activity consists of two stages. First, a group subjectively judges a number of ideas or alternatives or applies objective norms. Second, an alternative or a combination of alternatives is selected.
4. *Communicate*. This activity usually takes place during or right after each of the first three activities. Group members, for example, discuss intermediary results or decide how to proceed with the agenda.

For the purpose of deriving a classification of GSS functionality, we prefer the taxonomy of Bostrom and his colleagues over that of McGrath. First, Bostrom and his colleagues make an explicit distinction between information *generation*, which is seen to be divergent in nature, and information *organization*, which is seen to be convergent in nature. The different natures of these tasks set different requirements in terms of GSS support and meeting facilitation. Second, we consider McGrath's negotiate task to be from a higher level of abstraction than the generate and choose tasks. A negotiation process goes through several stages or phases, in which, amongst other tasks, a number of generate and choose tasks are exercised [Carmel et al. 1993]. Third, the same line of argument can be presented with respect to the execute tasks from McGrath's taxonomy.

In table 6.3 we depict which functions from three of the most widely applied workstation GSS, GroupSystems V, SAMM, and VisionQuest, support the four basic group meeting activities from Bostrom and his colleagues' taxonomy. Information to compose this table is taken from [Anson et al. 1992; DeSanctis et al. 1988; Dickson et al. 1992; Ventana Corporation 1993a and 1993b].

Activity	GroupSystems V	SAMM	VisionQuest
Generate	Electronic Brainstorming Group Dictionary Group Outliner Group Writer Idea Organization Policy Formation Questionnaire Stakeholder Identification Topic Commenter	Allocate Model Idea Gathering/Brainstorming Problem Formulation Stakeholder Analysis	Brainwriting Comment Cards
Organize	Categorizer Electronic Brainstorming Group Outliner Idea Organization	Clustering Problem Formulation	Brainwriting Comment Cards Compactor/Categorizer
Evaluate	Alternative Evaluation Group Matrix Questionnaire Stakeholder Identification Survey Vote	Allocate Model Idea Evaluation Contingency Model Stakeholder Analysis	Compactor/Categorizer Point Allocation Ranking Rating Scoring Subgroup Selection Voting
Communicate	Mood meter	Mood meter SAMM talk Send-a-note	-

table 6.3 GSS support for basic meeting activities.

With respect to the information presented in table 6.3 we want to add the following comments:

- VisionQuest's evaluation functions directly refer to certain voting techniques, which are all supported by GroupSystems' Vote and SAMM's Idea Evaluation.
- Only GroupSystems V and SAMM support multi criteria evaluation. Apart from the voting technique itself, SAMM's Allocate Model also provides process support by leading the participants through the preceding steps of generating ideas and criteria, and the assessment of weighted criteria.
- With respect to the fourth activity, communicate, we simply included the support offered for *oral* discussion. All three GSS support the electronic exchange of information for the first three main activities.

It is apparent from table 6.3 that there is little difference between the three GSS products in terms of the general functionality: all three GSS provide support for the first three meeting activities. Oral communication is only supported by GroupSystems V and SAMM. Functional differences do occur at a lower level of abstraction, that is in terms of the group techniques they offer, such as special brainstorming techniques or voting methods. However, the most important difference between the three GSS is not so much in terms of functionality, but in the *underlying*

philosophies that affect the way in which the functionality is offered to groups. The underlying assumption in GroupSystems V and VisionQuest is that group performance can be improved by imposing an efficient structure on the group through specific processes and technologies. Both these GSS are non-adaptive and facilitator driven. In contrast, the underlying assumption in SAMM is that structure should not be imposed on a group by the technology, but the technology should be offered in such a way that the group can selectively adopt the structure that is suggested by the technology [DeSanctis and Poole 1994]. Therefore, SAMM is designed to be adaptive, user driven. These different philosophies may determine which GSS a group or facilitator chooses to use for a meeting. Other factors include the availability of the system, and the experience a facilitator or a group has with it.

With the above meeting task taxonomy and GSS functionality mapping, meetings can be prepared in three steps. First, the meeting has to be split up into the tasks from the taxonomy. Second, the appropriate functions of the GSS to be used have to be selected. Finally, the meeting organizer or facilitator has to decide which of the tools to use, that support the *same* basic meeting activity. This choice largely depends on the nature of the information or the technique with which the information is processed. For example, choosing a preferred alternative from a list of over 50 items in VisionQuest is usually not done with Ranking, but with Subgroup Selection. For generating information, GroupSystems' Idea Organization can be used when it is expected that a one line description of each idea can be given or the Nominal Group Technique (NGT [Van de Ven and Delbecq 1974]) has to be used. For free text idea generation, GroupSystems' Topic Commenter is more appropriate, while Electronic Brainstorming can be used if one is afraid that the group discussion will have a tendency to follow a single theme.

In [Kumar et al. 1995] we propose how the above combination of a meeting taxonomy and the GSS functionality mapping forms the basis for a framework to direct the *application of GSS within a particular design approach*. In this framework, additional attention is paid to the actors involved in each of the approach's activity and to the potential effects of the GSS (setting) attributes. We further introduce and apply this framework in the last section of this chapter where we match the GSS technology to the activities of our participative design approach. In the following sections, we elaborate on the way of thinking, the way of working, and the way of modelling of the participative design approach. At this point, it should be recalled that this approach consists of techniques and support for group design that are embedded in the dynamic modelling approach's way of thinking and way of working. Furthermore, the participative design approach inherits some of the elements of the GRIPPPER approach which stands for Group Requirement Identification Process, Prototype, Pilot, Evaluation, and Rollout [Wanninger and Dickson 1992a and 1992b].

6.3 Way of thinking

The way of thinking of the participative design approach covers the same aspects as the way of thinking of the dynamic modelling approach. Organizations are seen as systems consisting of interrelated objects. The process of designing organizational change is carried out from a problem solving perspective while recognizing the concept of bounded rationality. However, as our approach, and the original GRIPPER approach in particular, aims to address the design of information systems as well, we also have to make our way of thinking explicit in this respect.

First, we need to define how we *perceive an information system*. We regard an information system as a combination of hardware, software, data, procedures, and people [Sol 1982]. We stress the importance of the latter two elements. An information system is never a goal in itself, it is used for supporting and controlling real systems, in particular organizations [Brussaard and Tas 1980]. In this sense, it is fruitful to think of information systems as *social systems*: a complex web of interpersonal relations that produces, uses, and communicates information [Land and Hirschheim 1983]. In other words, information systems can be seen as networks of communicating inquiry systems [Sol and Bots 1992]. An information system need not necessarily have a technical component, but when it does, the social interactions of the people play a crucial part in determining the way in which the technology is used [Checkland and Scholes 1990].

Second, we need to describe a philosophy behind *designing information systems*. It is apparent that the above ontological perspective on information systems comes close to our perspective on organizations. This similarity offers a basis for the integral design of organizational processes and their supporting information systems. Analogous to designing organizational processes, we adopt a problem solving perspective on designing information systems. Sol [1982] argues that designing an information system can also be seen as solving an ill-structured problem. Moreover, he demonstrates how dynamic modelling techniques used in the context of a problem solving process, support information systems design.

Following the GRIPPER approach, the problem solving process of our participative design approach has the following three basic assumptions and attributes:

1. The new information system should be designed in response to a need that is both related to the organization's current (problem) situation and to the organization's strategy.
2. The design activities should be carried out by a *team* in which all required skills are represented.
3. The design process should allow for feedback and iteration loops with different versions of the information system.

In the next two sections we describe the way of working and way of modelling that result from the above perspectives on designing organizational change and information systems.

6.4 Way of working

As stated in the previous section, we adopt a problem solving perspective on the process of designing organizational change and information systems. Therefore, the way of working constitutes the process of problem solving that we describe in Chapter 3, see also figure 3.1. However, the way in which the various activities are carried out and the models are constructed differs. Below, we discuss in more detail how the problem solving process is carried out. Before moving on, we want to stress that the process does not represent a number of linear model building steps that have to be carried out as if it were a cookbook. At any time, earlier models may be modified according to the requirements and insights that surface in a later activity.

The problem solving process starts with the *conceptualization* of the organizational problem situation. Using the object-attribute-action worldview of the dynamic modelling approach, the structure of the problem situation is described in terms of objects.

In the participative design approach the *empirical model* is not constructed using simulation and animation techniques. Instead, to further specify the organizational problem situation, a group of stakeholders from the organization first identifies and prioritizes a number of organizational requirements that have to be met for the organization to adequately perform its function in its environment. The construction of the empirical model is further supported by having the group of stakeholders identify and prioritize the problems associated with the processes in the conceptual model that frustrate the attainment of organizational requirements.

The identification of organizational requirements supports the construction of the empirical model in two ways. First, it helps to reassess the boundaries of the organizational problem situation and hence enables a reduction operation: The empirical model only needs to focus on problems with respect to processes that are related to the organizational requirements. Second, it may direct the identification of (potential) problems in the organization, if certain requirements are not adequately addressed in the current situation.

The empirical model is validated in an implicit and explicit way. Implicitly, because a group of stakeholders is involved in the identification and prioritization of the problems. Explicitly, because the identification of organizational requirements helps

to focus the empirical model on the right issues. Also, the empirical model is presented to another group of stakeholders than the one that was involved in its construction. The model is subsequently modified from their feedback.

After validating the empirical model, the problem situation is *diagnosed* by attempting to identify the root causes of the problems in terms of the way in which the organizational processes are organized and coordinated. This activity is carried out together with a group of stakeholders using brainstorming techniques, structured walkthroughs through the conceptual and empirical model, and common sense.

The third activity concerns *seeking solutions* for the problem situation. Based on the list of prioritized problems in the empirical model, a group of stakeholders together with the analyst have to identify organizational changes that are required to handle the identified problems. The propositions for organizational change may include new working procedures, new mechanisms to coordinate activities, and new information systems to provide the necessary information to perform these activities. In case of an information system, a specification is given of the functions that the system has to perform and of the information that it has to provide. In addition, the system is checked against other existing information systems that already perform these functions or provide the necessary information. Then, a first prototype of the system is built. This is done in three steps:

1. The prototype is programmed after selecting suitable hardware and software environments.
2. A limited set of realistic data are collected or generated in order to test the prototype.
3. The prototype (with data) is presented to the future users. Based on their comments, the specification of the information system may be modified and a new version of the prototype may be constructed.

When the prototype is considered mature enough, a pilot can be initiated in which the prototype is used over an appropriate period of time in a realistic environment. At the start of the pilot the users have to be trained. It may also be necessary to convert and import data from other systems. The purpose of a pilot is twofold [Tyre and Orlikowski 1994]. First, it may help to shape the further development of the system, by observing and evaluating the users' familiarization process with the system and paying attention to system aspects such as the magnitude of the data files, the system capacity, the response time, and the system's functionality [Wanninger and Dickson 1992b]. Second, a pilot may also reveal the (unforeseen) impact of the system on its organizational environment, and help to analyze and to anticipate these impacts.

The fourth activity concerns *selecting* a combination of change alternatives. This activity is supported by evaluating to what extent each of the alternatives addresses the identified problems. The actual selection is made by the group of stakeholders. It is most likely to be influenced by other aspects as well, such as costs, organizational culture, and policies concerning the conditions of employment in the organization. With respect to the information system's prototype, this fourth activity involves a formal evaluation, that takes place after the pilot period. Based on the experiences from the pilot, it is decided whether the prototype has to undergo a number of modifications or whether it can be implemented within the organization.

Finally, the selected change alternatives have to be *implemented* in the organization. With respect to the information system, this means that the prototype may be used as a basis and upgraded into the definitive version of the information system. In the GRIPPPER approach this is called Rollout. In general, the following steps may have to be taken [Wanninger and Dickson 1992b]: training the future users, further developing some parts of the prototype, and converting and importing a certain amount of data from other systems.

6.5 Way of modelling

We now turn to a description and discussion of the different types of models that are constructed during a study. As is apparent from the way of working, we make a distinction between conceptual models, that define the structure of a problem situation, and empirical models, that represent a further specification of the problem situation and facilitate analysis and diagnosis. The relationship between these models is depicted in figure 6.3.

The conceptual model is constructed with the object-attribute-action worldview of the dynamic modelling approach as a conceptualization filter. Based on the object definition part, we can define three aspect models: the interaction model, the process model, and the actor model. For a more elaborate discussion of the object part of the conceptual model and its three aspect models, we refer to Section 3.4.1.

The empirical model is constructed using a specification filter in which natural language is used as a model construction language. This filter supports the identification of organizational requirements and problems. These requirements and problems are related to the object definitions of the conceptual model as the information to describe them first passes through the conceptualization filter. The specification filter also includes several reduction transformations, like leaving out or aggregating concepts. The empirical model is discussed in the next subsection.

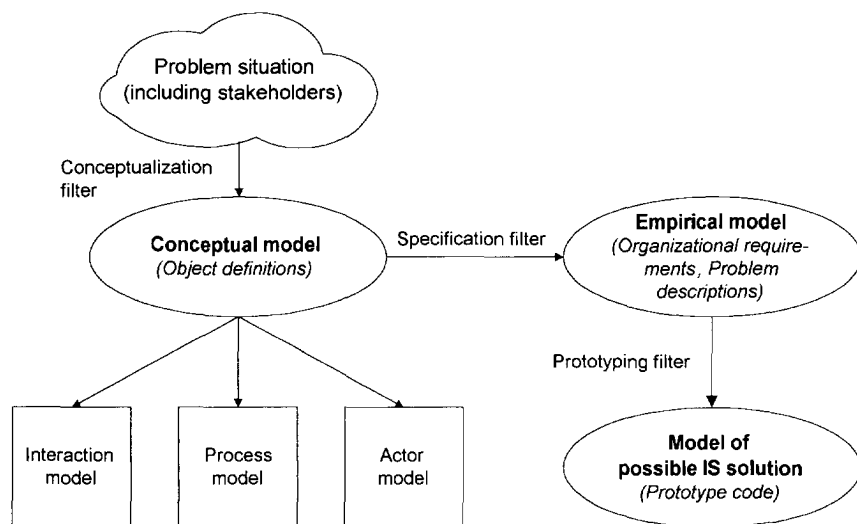


figure 6.3 The conceptual and empirical model in the participative design approach.

Finally, from the empirical model, a model of a possible solution in terms of an information system may be derived in the form of a prototype. For this transition, a prototyping filter is used. The model of a possible information system solution is discussed in Section 6.5.2.

6.5.1 The empirical model

Organizational requirements

The first part of the empirical model consists of a *list of organizational requirements*. These organizational requirements resemble the Critical Success Factor (CSF) concept that was first introduced by Rockart [1979]. The CSF concept was proposed as an aid to define *managerial information requirements* during the design of management information systems. CSFs represent the key areas where ‘things must go right’ for an organization to ensure attainment of its goals [Rockart 1979; Rockart and Bullen 1986]. CSFs are not related to strategic planning information, but to information that is required for managerial control of existing processes.

Although the CSF concept has proven to be a useful one for its purposes, see e.g. [Rockart 1979; Wanninger and Dickson 1992a; Wetherbe 1991], there are some points of consideration for using it in the context of our approach. First, by definition the CSF concept is focused on eliciting requirements for information system support, not on requirements for organizational processes and their coordination. Second, traditionally, CSFs are determined in a number of individual interviews with managers. This may be a time consuming activity, not in the least because in the end a set of

CSFs has to be constructed that sufficiently reflects everybody's point of view [Wetherbe 1991]. Finally, managers sometimes feel uncomfortable expressing their thoughts in terms of CSFs instead of in terms of activities or processes [Hengst 1994; Meel et al. 1994].

In order to address the above points, we deviate from the original CSF technique. First, we do not focus on information systems design. Instead, we elicit requirements with respect to the organization's functioning. We refer to these requirements as organizational requirements. By associating the organizational requirements to the organization's processes as described in the conceptual model, it is determined which processes are essential for which requirements and vice versa. This provides a basis for the analysis of an organization's situation with the empirical model which is grounded in the conceptual model.

Second, we determine and prioritize the list of organizational requirements for a particular organization in a *group setting*. Apart from the belief that this will prove to be more efficient, we expect the immediate exchange of ideas on organizational requirements to enhance the quality of the final list of these requirements. In a group setting, the organizational requirements list can be generated using various brainstorming techniques, such as NGT or Ideawriting [Moore 1994; Van de Ven and Delbecq 1974]. The prioritization of the requirements list may be supported with voting methods such as yes/no voting, rank order, allocation, multiple choice, or Interpretive Structure Modelling (ISM) [Moore 1994; Nunamaker and Briggs 1994].

As an example, we turn to the conference example used throughout Chapter 3. In table 6.4, we give a number of OCSFs from the conference organizer's perspective, and link these to a number of processes that the organizer has to carry out during the organization of a conference. It can be seen that there are two types of organizational requirements. The first type of requirement is attained when the related processes are carried out satisfactory. The second type of requirement needs to be fulfilled for the successful execution of a process.

Problem descriptions

The second part of the empirical model consists of a *list of problems* that are perceived by the stakeholders and the analyst with respect to the current situation in the organization. The problem list is expressed in natural language. Graphics may be used to further clarify a problem statement. Where possible, quantitative data is provided in order to further specify the problem statement. The stakeholders prioritize the list of problems with respect to being solved as soon as possible. In a group setting, the creation of the problem list can be supported with the same brainstorming and voting techniques that were mentioned with respect to the construction of the list of organizational requirements.

Organizational requirements				
	Attractive location	Low conference fee	High quality program	Timely submissions
Enough paying participants				
Processes				
Select conference location	x	x		
Form program committee			x	
Distribute call for papers			x	x
Have submissions reviewed			x	x
Distribute call for participation				x
Handle registrations				x
Seek sponsoring	x	x	y	y

x = process should support the attainment of the organizational requirement
y = organizational requirement should support the successful execution of the process

table 6.4 Linking organizational requirements to processes in the conference example.

In order to validate and, if necessary, extend the list of problems, the individual problem statements are linked to the ‘critical processes’ that are described in the conceptual model. This is done by building tables such as table 6.5 which depicts a number of problems with respect to the conference organizer’s example. Note that there currently are no activities or processes employed that are related to the problem that a number of speakers do not show up at the conference.

Problems	Processes
Many on site/late payments	Handle registrations
30% too late paper submissions	Distribute call for papers
	Have submissions reviewed
Conference fee perceived too high	Select conference location
	Seek sponsoring
Circa 5% speakers does not show up	-
Administrative work load too high	Have submissions reviewed
	Distribute call for papers
	Distribute call for participation
	Handle registrations

table 6.5 Problems in the current situation, related the processes.

6.5.2 *The model of a possible IS solution*

The model of a possible solution may refer to an information system that is to be employed in the future situation. This model is expressed in terms of a *prototype*. A prototype is a simplified version or model of the information system. It may contain an implementation of only few functions of the information system (vertical prototype) or of all functions but with less detail (horizontal prototype). Prototyping can be seen as a special form of simulation [Sol 1984].

Prototyping may be used for various purposes [Sol 1984]. A prototype can support the identification of system requirements. It can also be used to verify the appropriateness of a system in fulfilling these requirements. In addition, prototypes are often employed to gradually develop certain parts of the information system, such as the user interface. Finally, a prototype can be implemented in the organization during a pilot period to pre-test the final information system.

Different versions of a prototype are presented to a group of stakeholders from the organization, representing the future users. Based on their feedback, modifications may be made in the specifications of the information system and may result in a new version of the prototype. In the end, the information system may be built out of the final version of the prototype ('prototyping for evolution' [Floyd 1984] or 'upgrading' [Vonk 1984]), or the final prototype may only be used as a further specification of the information system (a 'throw-away prototype' [Sol 1984]).

Often, prototyping efforts are characterized by a lack of attention to the information requirements definition. There is a tendency to start with building a prototype right away, rather than a thorough requirements analysis [Boehm 1988; Floyd 1984]. In our approach, this is not a point of consideration. As we described previously, our approach pays extensive attention to (joint) requirements definition. Another risk related to using prototypes is an early freezing of design constraints and choices. The organization or analyst may be reluctant to throw away (parts of) a prototype since money and time was invested to create it [Boehm 1988; Sol 1984]. In our approach we attempt to avoid this risk by (1) inviting and stimulating the stakeholders to give a very critical evaluation of each prototype that is developed, and by (2) not involving the stakeholders in the actual development of the prototype itself to avoid a feeling of 'ownership'.

The exact way in which prototypes are built during our approach depends on the specific problem situation in terms of the available hardware and software and of the type of information systems that needs to be prototyped. In addition, functional and data specification techniques may be used to facilitate the transition from the expressed requirements of the future information system into prototype code.

If we assume in the conference example that (1) the stakeholder group agrees on the development of a conference information system to handle administration and manage conference information, and that (2) the conference organizer has an MS-Dos environment with Microsoft Windows available, we might consider using Clipper or Microsoft Access to build the prototype. This would blend well with the nature of the conference system as it would probably represent a number of databases on which various functions are performed.

6.6 Mapping GSS to group design activities

From the description of the way of working and the way of modelling, it follows that during the various model building activities there is frequent interaction between the analyst and a number of stakeholders in the organization. This interaction, we feel, is of vital importance for the quality of the design process and its outcome. Amongst other reasons, stakeholders' opinions and perspectives with respect to the functioning of the organization or a future information system may change during the course of the process, thereby resulting in changes in the organizational or system requirements. Also, it is believed that stakeholder participation enhances their acceptance of the eventual information system.

The GRIPPPER approach employs GSS to support stakeholder involvement. It is considered to be a key factor in the process of facilitating stakeholder participation [Wanninger and Dickson 1992a and 1992b]. Originally, the use of GSS within the GRIPPPER approach is limited to the GRIP phase. In this phase, a GSS is used to brainstorm about CSFs, function specifications, and data specifications. We propose to extend the application of GSS in the participative design approach. It is apparent from the description in Section 6.2, that GSS may have the potential of supporting the design activities where participation is opportune.

Based on the framework we proposed in [Kumar et al. 1995], table 6.6 identifies the specific activities that may be supported with GSS with respect to the participative design process. The basic activities in the problem solving process are split into a number of subactivities. For each subactivity, the relevant actors that are involved are identified. They may involve the analyst, the stakeholders, or both. In order to identify the type of group support needed, the nature of each separate subactivity is determined. Using the meeting task taxonomy we describe in Section 6.2.3, each subactivity is classified as *generate*, *organize*, *evaluate*, *communicate*, or *other*. Note that the activities in the table do not represent a particular order.

In table 6.6, we assume that there is one analyst. However, it is not unlikely that a group of analysts is involved in either one of the design processes. In this case, the

Activity	Actors	Nature
<i>Construct conceptual model</i>		
• Identify objects	Analyst	Generate
• Create object definitions	Analyst	Other
• Create or modify aspect models	Analyst	Other
• Present aspect models to stakeholders	Analyst	Communicate
• Give feedback on aspect models	Stakeholders	Generate + Evaluate
<i>Construct empirical model</i>		
• Identify organizational requirements	Both	Generate + Organize
• Prioritize organizational requirements	Both	Evaluate
• Identify problems	Both	Generate + Organize
• Prioritize problems	Both	Evaluate
• Link problems to processes	Both	Organize
• Determine which problems to address	Both	Evaluate
<i>Construct models of possible solutions</i>		
• Identify possible solutions	Both	Generate + Organize
• Build prototype	Analyst	Other
• Present prototype	Analyst	Communicate
• Give feedback on prototype	Stakeholders	Generate + Evaluate
• Determine necessary modifications	Both	Generate + Organize
• Prioritize necessary modifications	Both	Evaluate
• Collect (realistic) data to test prototype	Both	Generate
<i>Choose most preferred solution</i>		
• Select modifications to include in next prototype	Both	Evaluate
• Determine next step: pilot, new prototype, or implementation?	Both	Evaluate
• Choose other solution(s) to be implemented	Both	Evaluate
<i>Implement solution</i>		
• Present final information system	Analyst	Communicate
• Give feedback	Stakeholders	Generate + Organize
• Start using information system	Stakeholders	Other
• Implement other chosen solutions	Both	Other

table 6.6 *A classification of the activities of the participative design approach.*

group of analysts may also benefit from the use of GSS during some of the activities for which they are identified as the single actor, e.g. identify objects.

When the nature of the various design activities is classified, the next step is to determine which type of GSS technology has to be employed. This has to be done both in terms of the GSS product to be used and in terms of the GSS' function(s) or module(s). To this end, we can use the information depicted in table 6.3. However, the fact that in theory GSS can be employed for each of the activities that are classified as joint activities, is no guarantee for the success of this application. The successful application of a GSS requires the judicious appraisal of the potential benefits and risks of using GSS. In each particular situation, the potential effects of the attributes of the GSS technology and the GSS setting have to be ascertained. This concerns anonymity,

parallelism, group memory, media richness, facilitation, group size, group experience, and culture. For each attribute, the analyst has to determine whether the attribute will be likely to ameliorate or harm the group process, or will be likely to have no effect. In case an attribute may appear to be beneficial, the analyst has to make sure to take maximum advantage of this. In case of a potential detrimental effect, he or she has to decide to abstain from using GSS support or has to take preventive measures.

7 Case 3: The Criminal Investigations Department

*'And as a single leaf turns not yellow but with the silent knowledge of
the whole tree, so the wrongdoer cannot do wrong without the hidden
will of you all.'*

- Kahlil Gibran

7.1 Introduction

The third case study in our research was carried out at the Criminal Investigations Department of the Amsterdam Municipal Police Force. The initial goal of this case study was to test in a prescriptive way the dynamic modelling approach that was developed in the first two case studies. As it turned out during the study, the dynamic modelling approach required additional methods and techniques to handle this particular problem situation. The required extensions and modifications of the dynamic modelling approach led to the second initial theory of our dissertation research: techniques and support for participative design, which we described in the previous chapter. Hence, the additional goal of our third case study was to develop, apply, and evaluate these techniques and support.

In this chapter we first describe the problem situation at the Criminal Investigations Department in more detail. We then subsequently discuss the conceptual model (Section 7.3), the empirical model (Section 7.4), the problem diagnosis (Section 7.5), and the alternatives for change (Section 7.6). The chapter is concluded with a reflection on this case study.

7.2 Problem situation

In this section we describe the environment of the CID, organized crime, and the way the CID is organized to work in this environment. We also state the practical assignment of this case study. Part of the material in this section was earlier described in [Meel et al. 1994; Meel 1994].

7.2.1 *The environment*

The main mission of the Criminal Investigations Department (CID) of the Amsterdam Municipal Police Force is to combat organized crime in the Amsterdam area in order to protect the integrity and democratic quality of society [Riessen 1991; Welten 1992]. Organized crime no longer is an exclusive phenomenon in Italy and the United States [Middelburg 1992]. Today, criminal organizations operate globally. In 1993, their annual profits were estimated at \$1 trillion, almost the same size as the US federal budget [Newsweek 1993]. A survey of criminal records maintained by the police and the law, revealed that there are about 600 criminal organizations in the Netherlands [National Criminal Intelligence Service 1988; Task Force Crime Analysis 1991]. The combined annual turnover of these organizations was estimated to be 10 billion Dutch guilders.

The phenomenon 'organized crime' appears to be difficult to define in a concise and unambiguous way [Baarle 1991; Fijnaut 1989b]. Intuitively, criminal organizations are often perceived as strong hierarchical organizations led by a 'Godfather' that are involved in narcotics or racketeering [Duyne 1992b]. However, in reality, there are a myriad of organizational forms which have various (other) criminal core businesses [Duyne et al. 1990; Fijnaut 1989b; Reuter 1983]. Examples include fraud, illegal fire arms trade, car theft, women trade, illegal gambling, and prostitution [Lensing 1989; Ministry of Justice 1985; National Criminal Intelligence Service 1992]. Regardless of their organizational form or core business, criminal organizations do have one common characteristic. Their criminal activities are not a goal in themselves, they are aimed at obtaining financial means and power [Küster 1991].

A complicating factor in the combat against organized crime is that criminal organizations seldom limit themselves to the illegal world. They try to expand their business into the legal world. This not only helps them to carry out their criminal activities. In time, after laundering and investing their criminal money into legal businesses, it will also transform them into perfectly legal and respected enterprises. At the same time, it is sometimes very interesting (financially) for legal organizations, such as banks and law firms, to offer their services to illegal clients [Baarle 1990; Duyne 1991; Roer 1989]. Unfortunately, it is very difficult to make a strict distinction between the legal world and the illegal world. Consequently, it is not clear to what extent criminal organizations have infiltrated the legal world. There seems to be a 'grey twilight zone' between the two worlds for which the police and the law lack adequate authority and means to operate in [Baarle 1991; Welten 1992]. Recent developments in Italy and Colombia show that this twilight zone is a very dangerous and fertile breeding ground for organized crime.

Traditionally, the police and the law have fought a combat against organized crime that was based on two assumptions. First, it was assumed that the employment of more men, means, and tactics could solve the problem of organized crime [Hogenbrink and Teerds 1989]. However, the failures of the 'war on crime' and 'war on drugs' programs under the Reagan and Bush administrations in the United States have shown that this strategy of 'more' does not result in the solution of the problem [Duyne 1992a]. Moreover, historically organized crime has been around for several centuries. It seems that in principle, organized crime cannot be wiped out, but only controlled to a certain extent [Egmond 1989; Fijnaut 1989a]. Second, it was assumed that each criminal offense has to be prosecuted on its own. As a result, the broader scope in which organized crime takes place was never exposed. Experiences show that there are better ways to combat organized crime [Barton 1988; Duyne 1991; Jacobs 1991]. By looking at individual offenses, the best result one seems to achieve is harassing a criminal organization for a short time [Baarle 1991].

Ever since the legal authorities in the Netherlands have been aware of the ineffectiveness of these assumptions, a number of strategic and operational changes in the combat against organized crime are occurring. The task of the police and the law has shifted toward *controlling* organized crime [Korthals-Altes 1989]. A requirement of this shift is that the police need to have good intelligence and adopt a 'holistic' approach to follow and act on new trends and developments in society [Baarle 1989; Riessen 1991]. Also, there has been a growing interest in new corrective and preventive police tactics and strategies, such as observation, infiltration, crime analysis, seclusion of areas in the legal world, and the 'fox hole approach' [Duyne 1992a; Fijnaut 1989c; Hogenbrink and Teerds 1989]. Finally, it is recognized that organized crime is no longer an exclusive problem for the police to handle. The government determines to a large extent the potential business areas for criminal organizations [Duyne et al. 1990; Korf 1992; Lissenberg 1992]. Laws cannot be taken in isolation from the social developments to which they are a reaction. Hence, organized crime has become a *societal problem* which requires the close and interdisciplinary cooperation between the police and a great number of social actors and institutions.

7.2.2 The organization

Since the sixties, the CID has transformed itself a number of times in order to better deal with the ever changing nature of the criminality it had to handle. For a description of the early years of the CID, we refer to [Meel et al. 1994]. In the mid eighties, the CID roughly consisted of two divisions: an operational division for the execution of actual projects, and a division for technical and tactical support [Task Force 'Kit in de lift' 1986]. Each division consisted of a number of relatively autonomous departments that were highly specialized according to criminal offense type or to suspect nationality. Over the years, this structure increasingly obstructed combating organized

crime effectively. First, there was a lack of interdepartmental coordination, both in terms of daily operations and in terms of the exchange of criminal information. This coordination was a prerequisite, since criminal organizations seldom restrict themselves to one type of offence, and, therefore, the activities from multiple specialized departments are required. Second, at that time, the CID still focused on individual offenses and not on the broader, organized context in which they were committed. Third, the departments and their detectives were very autonomous with respect to starting their own projects. In other words, the overall direction of the CID was not determined by a political or societal agenda, but by the interests and opportunities of the individual detectives and their departments. As a result, projects were difficult to manage, could last a long time, and sometimes could even be expected not to lead to success.

In order to overcome these problems, a major change operation was begun in 1990. Not only a new strategy for combating organized crime was introduced, but also the structure of the CID was changed. After promising experimental results, the force's top management decided that the case driven 'individual' strategy was to be replaced by a new market driven 'holistic' strategy [Riessen 1991]. The CID started operating with project teams, that included all disciplines necessary to conduct a project. Part of the market driven strategy was to distinguish between two 'markets', known as Market I and Market II. *Market I* refers to 'safety', 'livability', and 'care' and can be seen as equivalent to the 'local community'. Crime in this market is characterized by its local nature, low degree of organization, and its daily visibility to citizens. This market is the responsibility of the decentralized departments of the police force. *Market II* refers to 'social integrity'. Crime in this market is organized, of a regional, of a national, and even of an international nature, and far less visible to the citizens. This market is the responsibility of the CID.

Initially, the two divisions of the CID, the operational and support divisions, began to reorganize drastically to implement the new market driven strategy [OCID 1992; SCID 1992]. Evaluation of the reorganization plans of the two divisions showed that it would be fruitful to integrate both divisions into a new central department. Hence, in 1992 it was decided to create the new CID [CID 1992, 1993]. This decision also marks the beginning of our case study in the CID. The new CID focuses its activities on Market II and in particular on the 'grey twilight zone' between the legal and illegal world [Welten 1992]. These activities take shape in projects that have both repressive and preventive objectives. With respect to Market I, the new CID only provides technical and tactical support as it is more efficient to organize these centrally. The new CID only supports projects and does not initiate its own projects in this market. Exceptions are made for a number of special crimes such as kidnapping, sexual crimes, and physical and sexual forms of child abuse.

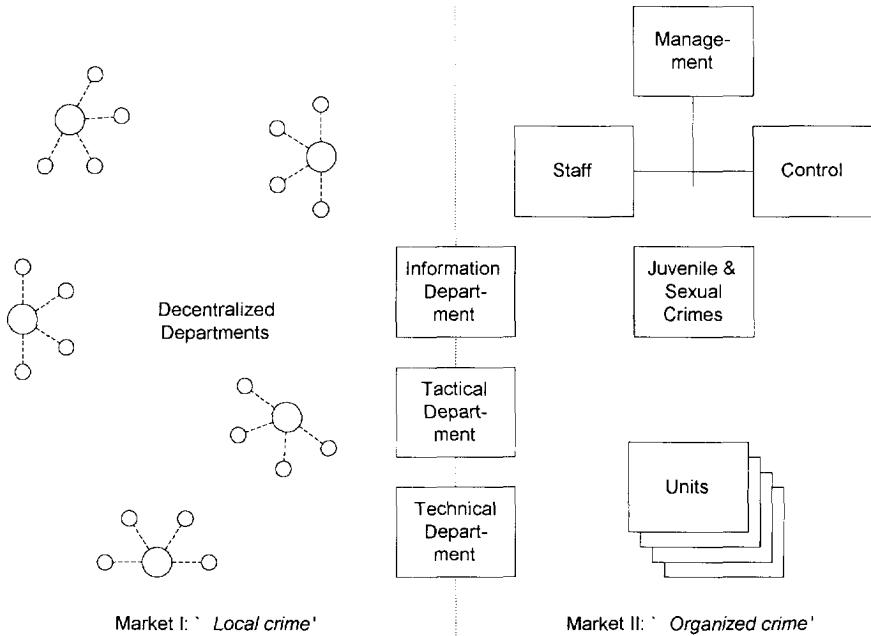


figure 7.1 New organizational structure of the CID [CID 1992].

The organizational structure of the new CID is given in figure 7.1. The new heart of the organization is formed by the *units*: multidisciplinary project teams that are autonomous in their daily operations. A unit consists of a base team of ten to fifteen police detectives who have worked with each other on various projects. This base team is extended with experts from other internal departments and external institutions depending on the type, size, and course of the project. Each unit has a unit chief supported by a coordinator. The management of the new CID has final responsibility for all projects.

The department of the old divisions that are not merged into the units, are combined into three new supporting departments: the Information Department, the Tactical Department, and the Technical Department. These departments are designed to support the units of the new CID (Market II) and the decentralized departments of the force (Market I). As a first norm, 50 percent of the support capacity should be used for Market I and 50 percent for Market II [CID 1992, 1993; Task Force Objectives 1992].

The major task of the *Information Department* is to build up an intelligence about criminal activities in Market I and Market II. This information is used to prepare new projects. The Information Department must provide the management of the CID with advice on new projects to be started on a regular basis. The *Tactical Department* can be seen as a specialized deployment agency for areas such as observation, infiltration,

arrests, and fraud. Examples of tactical support are short assistances, part time or full time detachments of employees, or consultancy. The *Technical Department* employs technical specialists in the area of technical projects (fingerprints, laboratory research) and technical equipment (scanners, videos). This department is purely supportive and does not detach its employees.

With respect to the transition of the CID to its new strategy and organizational structure, a number of problems can be identified, see also [Meel et al. 1994]. These problems are categorized into four categories. A summary of the problems is given in table 7.1.

Transition process:

- Autonomy, position and task content
- Information and system management
- Fast results versus complex projects?

Preparation of projects:

- Design and functioning of Information Department
- Selection of projects
- Preventive projects, twilight zone, societal trends

Execution of projects:

- Unit structure
- Automated support for project administration
- Project management, management information

Tactical support:

- Deployment agency?
 - 50%-50% support norm
-

table 7.1 Problems faced by the CID.

Problems with respect to the transition process

In the new CID, detectives from different former departments must cooperate together. Detectives will no longer be autonomous. Cultural differences are expected to cause some start up problems in projects in which detectives from different backgrounds participate. Also, due to the blending of various departments, the number of middle managers is reduced drastically.

Because of the CID's new strategy, the task content of detectives will become more information intensive. However, information resource management in the CID is not yet prepared for these new demands. Issues such as file organization and integration, technical management, security, and standardization require more attention.

The throughput times of projects are increasing as organized crime is getting more complex. Moreover, ever since organized crime has been put high on the political agenda, there is a growing pressure to 'score' and get results.

Problems with respect to the preparation of projects

The Information Department will play an important role in the new CID. This department will have to be the central node where all information flows come together. This information the Information Department will have to use to prepare projects. However, in the current situation it is only clear which of the former departments are going to be blended into the Information Department. A lot of questions still remain, such as Which are the information flows? How does the Information Department have to be organized in order to manage and coordinate these flows? What technical support does the department need for this? How does the Information Department have to coordinate their activities with the rest of the CID? and How are projects going to be prioritized, especially if the department has to advise the CID management which projects to initiate?

Another problem is that there is as yet little experience with preparing other types of projects than those on drug trafficking organizations. It is not clear how preventive objectives have to be pursued, nor how projects aimed at societal trends and the 'grey twilight zone' have to be prepared.

Problems with respect to the execution of projects

A difficult issue with respect to the execution of projects concerns the structure of a unit. Initially, the idea was to determine the exact composition of a unit at the start of each project. However, this idea received considerable opposition. It was argued that this way, projects would suffer from long start-up times because the unit members would have to get used to each other each time. Therefore, it was decided to work with so called 'base units' to which internal and external experts could be added during later stages depending on the size, type, and course of a project. How this method is to be implemented in practice is not clear. Extra capacity is not always available and should, therefore, be planned. Due to the long duration of projects, the great fluctuations in the work load of a unit, and the large amounts of over time, this is easier said than done.

Another problem concerns information management during a project. As projects become more and more information-intensive, there is a high demand for automated support. Unfortunately, little support is currently available in this respect. A solution may be a new system called Octopus, aimed at recording and managing large amounts of paper work that is created during large scale projects [Hoogeveen et al. 1992; Wouw 1992]. An experiment with Octopus was just underway at the beginning of the case study. However, the organizational and technical consequences of the system are still unclear.

Information on the execution of projects also needs to be maintained in order to facilitate project management by unit chiefs and the CID management. The progress of projects needs to be monitored constantly. How this management process should be

structured and carried out is unclear. It is not clear either exactly which management information will be needed.

Problems with respect to tactical support

The Tactical Department will have to function as a specialized deployment agency. Similar to the composition of units, it is unclear how this concept should be implemented. It is also unclear to what extent the 50%-50% support norm for Market I and Market II is achievable or sensible.

7.2.3 Assignment of the case study

The general assignment for this case study was to design and evaluate possibilities for organizational change so that the CID could effectively implement their new market driven strategy [Meel et al. 1994]. Based on this assignment, two studies were initiated. The first study looked into improving the way in which projects are executed within the CID. An account of this study can be found in [Meel 1994]. The second study had to look into the issues with respect to the preparation of projects and the design and support of the Information Department. The problem assignment for this second study was twofold:

1. To design the Information Department's structure, processes, and their coordination.
2. To design support for the Information Department for coordinating its information flows.

The remainder of this chapter reports on the results of this second study.

7.3 The conceptual model

In this section we describe the conceptual model of the CID case. First, we discuss the object class definitions that together form the conceptual model. Then, we highlight specific coordination aspects by presenting the interaction, process, and actor model of the conceptual model using the taxonomy of coordination mechanisms from table 1.1.

Before turning to the object definitions, we make two comments. First, given the scope of our problem assignment, we will only focus on the objects that are involved in the preparation of projects. Furthermore, for the sake of simplicity, we will not describe the whole conceptual model, but focus on a number of key objects. For a more elaborate description, we refer to [Hengst 1994; Meel et al. 1993 and 1994]. Second, the 'current situation' that we conceptualize is the situation that existed right after the

Information Department had been officially founded by the CID management. This means, that the employees of the Information Department, originating from various specialized departments, did not yet have any experience working together except for in a number of experimental projects. Hence, the activities and processes described in this section are to a large extent based on these projects.

7.3.1 Object definitions

In the CID we can distinguish between six different item classes that are relevant to the preparation of projects. Projects in the CID are born out of information on particular subjects such as people, organizations, or trends. This information arrives at the Information Department in the form of *tips*. A tip originates from an active or a passive source. Active information sources are, for example, (anonymous) contacts, passive information sources may represent newspapers, databases or other (electronic) media.

object class TIP
is a MESSAGE

attributes
origin

object class PROJECT IDEA
is a MESSAGE

attributes
-

object class PROJECT
PROPOSAL
is a MESSAGE

attributes
-

object class PROJECT
INFORMATION
is a MESSAGE

attributes
-

object class PROJECT IDEA
DECISION
is a MESSAGE

attributes
decision result

object class PROJECT PRO-
POSAL DECISION
is a MESSAGE

attributes
decision result

A tip or a combination of tips may give rise to a *project idea*. This is a rough description of a potential problem area for which a project could be prepared. Before an employee of the Information Department can proceed with a project idea, the chief of the department has to issue a positive *project idea decision*. If this is the case, various additional tips are gathered and combined into *project information*. The final result is a *project proposal* that describes the project's problem area, its goal, and a rough outline of the desired strategy to execute the project. The project proposal is presented to the CID management, that, together with the public prosecutor, has to issue a positive *project proposal decision* before the proposal can be forwarded to a unit to be executed.

There are several actors that handle the mentioned items while preparing projects to combat organized crime. The key actor is the *Information Department employee* (ID employee). An ID employee is specialized with respect to the type of information he

handles: 'hard' information (proven facts), or 'soft' information (not yet proven or unconfirmed facts). The ID employee is responsible for handling all tips and information with respect to Market II problem areas. He builds and maintains the information position of the CID. He is in contact with passive and active *information sources* from which he gets tips. Performing desk research, such as crime analysis, he combines and processes tips. The results consist of project ideas and project proposals. When the *ID chief* and the CID management approve of the material he prepared, he hands the project proposal over to the *unit chief* responsible for executing the project.

object class ID EMPLOYEE
is an ACTOR

attributes
specialty

actions
receive tip from information source
receive tip from information storage
receive project inf. from information storage
receive project idea decision from ID chief
receive project prop. dec. from CID managem.
send tip to information storage
send project idea to ID chief
present project proposal to CID management
send project information to information storage
send project proposal to unit chief
build information position
prepare project
process tip
perform desk research
present project proposal to CID management
keep project proposal up-to-date

object class INFORMATION SOURCE
is an ACTOR

attributes
reliability

actions
send tip to ID employee

object class ID CHIEF
is an ACTOR

actions
approve of project idea?
receive project idea from ID employee
send project idea decision to ID employee

object class UNIT CHIEF
is an ACTOR

actions
receive project proposal from ID employee

The *CID management* has the final responsibility for the preparation and execution of projects. It has to approve of a project proposal, before the project is appointed to a unit. For this approval, the CID management discusses the proposal with the *public prosecutor* that will probably have the final responsibility for the project. In case the CID management and the public prosecutor decide in favor of a project but there is no unit available, the ID employee has to store the proposal and keep it up-to-date until there is a unit available. In addition, the CID management has to determine the overall policy of the department.

object class CID MANAGEMENT
is an ACTOR

actions

approve of project proposal?
unit available?
receive project proposal from ID employee
send project proposal to public prosecutor
send project proposal decision to ID employee
discuss project proposal with public prosecutor
appoint project to unit
determine CID policy

object class PUBLIC PROSECUTOR
is an ACTOR

actions

approve of project proposal?
receive project proposal from CID management
send project prop. decision to CID management
discuss project proposal with CID management

There are two repository classes in the CID. The first represents passive *information sources* such as the media. The ID employee uses these sources to get a feel of what is going on in society, of which trends are developing. The second repository class represents all systems in which tips and project information are stored: the *information storage*. The Information Department makes use of a variety of systems and applications, both manual and automated. Examples include card-trays, RapidFile, WordPerfect, AskSam, and Octopussy (the PC version of Octopus). The reason that so many different systems and applications are used is that they originate from the constituent departments of the Information Department and there are no organization-wide standards.

object class INFORMATION SOURCE
is a REPOSITORY

attributes

type of source

actions

receive information
send tips to ID employee

object class INFORMATION STORAGE
is a REPOSITORY

actions

receive tip
receive project information
send tip to ID employee
send project information to ID employee

The actors and repositories in the CID exchange information through various links. Two examples are the *ID chief link* and the *storage link*. The ID chief link connects the ID employee with his chief. This link supports the oral or written communication of a project idea from the employee to his chief. The storage link defines the interaction of the ID employee with the information storage. This link may either represent the electronic exchange of project information and tips or the analog exchange in case the information is stored on indexed cards or in paper files.

object class ID CHIEF LINK
is a LINK

attributes

starting node (ID employee)
ending node (ID chief)
uni-directional

actions

route project idea

object class STORAGE LINK
is a LINK

attributes

starting node (ID employee)
ending node (Information storage)
bi-directional

actions

route project information
route tip

The actions of the actors in the CID can be modelled in more detail with decision and task classes. An example of each class is given below. The decision concerns the approval of a project idea by the ID chief. As can be seen, there are no 'hard' decision rules for this decision. The outcome of the decision is mainly based on the ID chief's experience and intuition. The task represents the discussion of a project proposal by the CID management and the public prosecutor. An important precondition is that both actors meet each other at the same time (not necessarily at the same place).

object class APPROVE OF PROJECT IDEA?
is a DECISION

attributes

required items: project idea
preconditions: -
postconditions: ID chief's opinion on project idea
is known
next actions: send project idea to ID chief
('no'), perform desk research
('yes')
decision rules: IF ID chief considers idea fruitful delay
THEN perform desk research
IF ID chief considers idea not
fruitful THEN send project idea
to ID chief

object class DISCUSS PROJECT PROPOSAL
is a TASK

attributes

required items: project proposal
preconditions: CID management and public
prosecutor meet at same time
postconditions: CID management and public
prosecutor have joint opinion
on project proposal
next actions: approve of project proposal?
att. assignm.: -

7.3.2 Interaction model

The CID as whole can be seen as a *functional hierarchy*. There is a clear hierarchy of authority: the CID management has the final responsibility, the ID chief and the unit chiefs have responsibility over the efforts concerning the preparation and execution of projects respectively. The Information Department, the Tactical Department, the Technical Department, and the units each have a well defined function in the combat against organized crime.

Looking at the Information Department in particular, we also see a *product hierarchy*. As said, the ID chief has the final responsibility over the output of this department. There is one major subdivision in the Information Department: ID employees either deal with 'soft' or with 'hard' information. In the situation before the Information Department was founded, this product division was also present, but more detailed. There used to be several departments that dealt with different types of soft information.

There are also some (potential) *networking relations* present in the Information Department. The ID employees have very informal relationship with the active information sources. In addition, the ultimate desire of the Information Department is that each employee is capable of performing all functions with respect to either 'soft' or 'hard' information. This would allow flexible cooperation between employees from different functional backgrounds during the preparation of a project.

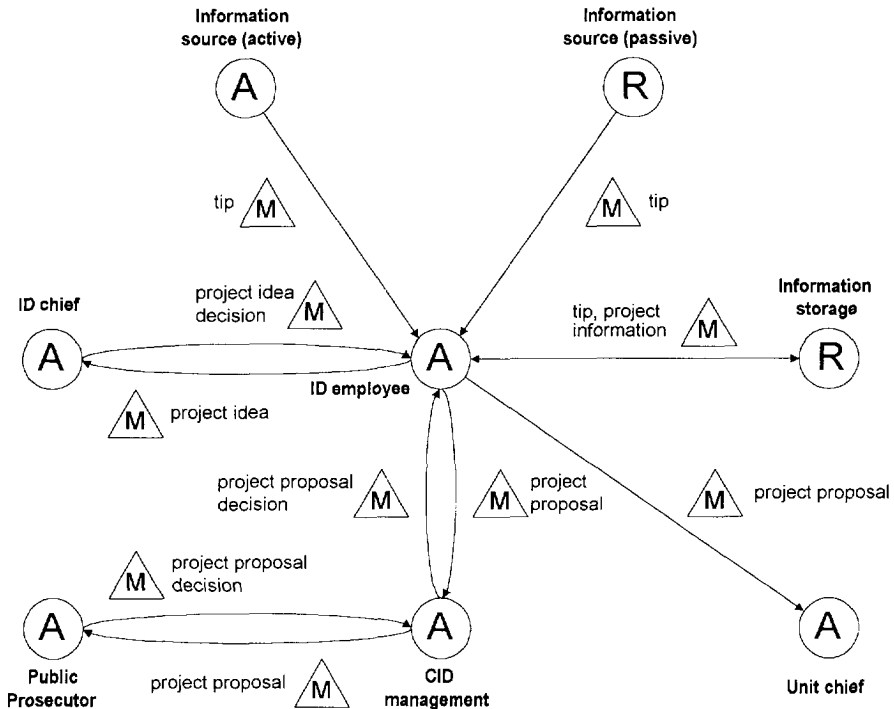


figure 7.2 Interaction model for the Information Department situation.

Looking at the interaction model, it is clear that the ID employee has a very central position, see figure 7.2. Apart from two 'approval' relationships with his chief and the CID management, the ID employee is largely autonomous with respect to preparing projects. He does not have to attune his activities with other actors. This is clearly represented by the absence of other (bi-directional) links between the ID employee and

other actors. The employee only receives tips from information sources and forwards an approved proposal to a unit chief.

7.3.3 Process model

If we look at the processes in the CID and the Information Department, we see that *standardization* is the predominant coordination mechanism. Although there are a few approval tasks and decisions, in preparing projects there is little explicit communication between the actors for coordinating the constituent activities of this process. For example, the ID employees perform their activities and then pass a project proposal on to a unit chief who knows exactly how to take it further, see figure 7.3 (CIDman = CID management, PP = public prosecutor).

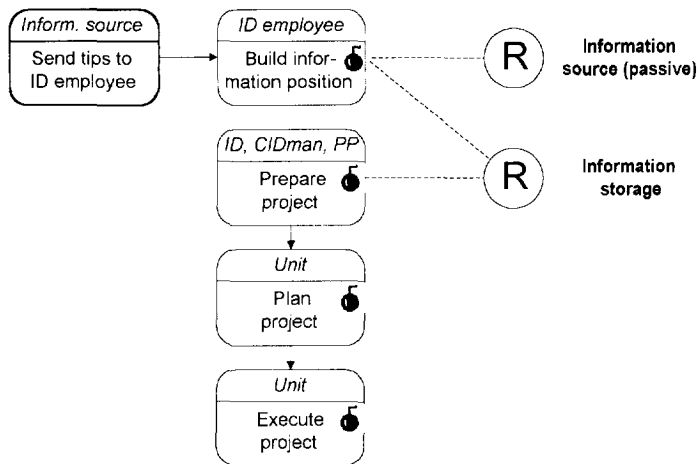


figure 7.3 Process model for combating organized crime.

The leading type of standardization on the process level is standardization of *worker skills*. The quality and talents of the individual actors are considered to be a guarantee for the outcome of the processes. For example, an ID employee is expected to produce a project proposal with which a unit can successfully start a project. Hence, on the level of a process as a whole, there is no standardization of *work processes*, although certain individual actors have to work according to predefined rules and procedures (see next subsection).

It can be said that to a certain extent, there is also standardization of *work outputs* in the CID and the Information Department. Project proposals have to contain a number of predefined elements. Nevertheless, proposals normally turn out to differ substantially in terms of detail and accuracy with respect to the information they present. In the

same spirit, tips from active information sources have to be registered in a predefined format. This allows for assessing the reliability and credibility of the tip.

Notwithstanding the predominance of standardization mechanisms, there is also some *direct supervision* present in the CID. At several points during the execution of the CID's processes, there are actors that instruct other actors (on a lower level in the hierarchy) what they should do. This, for example, is the case during the preparation of a project, where the CID management appoints a project to a unit, see figure 7.4.

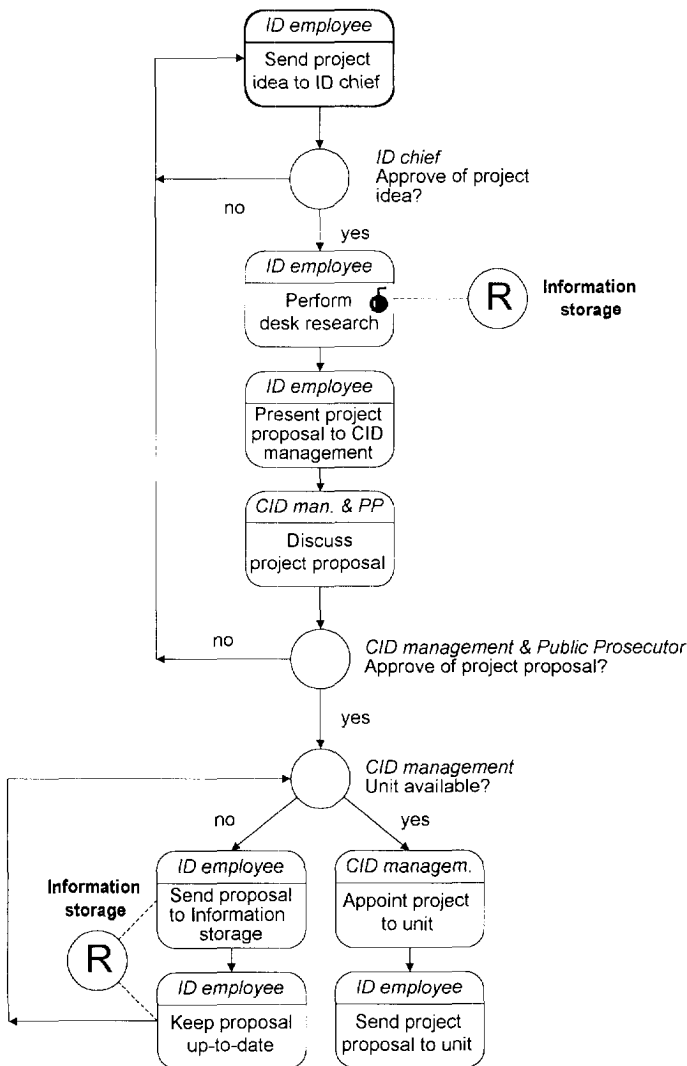


figure 7.4 Process model for preparing projects.

Looking at the *coordination processes* in the CID, we see *adaptive* processes. These processes are represented by the approval decisions that the CID management and department chiefs take at certain points in the processes. Hence, the actors that have a higher authority only participate in the work processes when it is required. This is for example visible in the process model preparing projects, see figure 7.4. Both the ID chief and the CID management are only adaptively involved in the actual preparation of projects: when it is required, they approve or disapprove of project ideas or proposals.

There are no *anticipatory* or *planning coordination processes* with respect to the preparation of projects. Project proposals are expected to be ready when a unit finishes a project and become available for a next one. It is also argued, that there are far more potential projects than that there are sufficient units available to take them up. There are no *review processes* either. The way in which projects are prepared and appointed to units is not recorded and evaluated.

7.3.4 Actor model

If we look at the interdependencies between the activities of the individual actors, we also see that *standardization* is the predominant coordination mechanism. However, on the level of individual actors, the type of standardization is not only worker skills, but also *work processes*. For a number of activities that the actors in the CID and in the Information Department have to carry out, there are written procedures and (legal) regulations that have to be followed. An example is given in figure 7.5, that depicts the actor model of an ID employee performing a crime analysis. Crime analysis is one of several types of desk research that ID employees carry out.

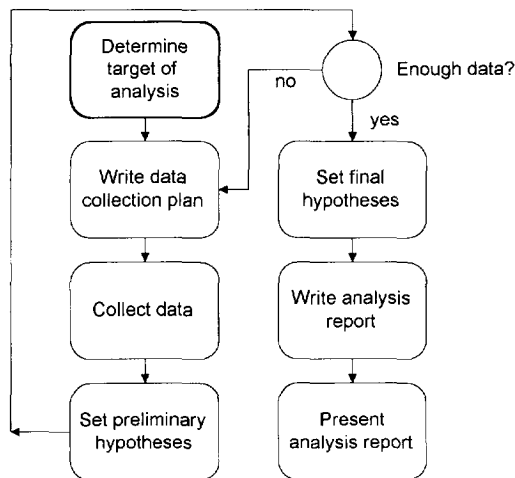


figure 7.5 Actor model for an ID employee doing crime analysis desk research.

7.4 The empirical model

In this section, we turn to the empirical model of the current situation with respect to the preparation of projects. The empirical model consists of two parts. We first describe a number of organizational requirements that were identified in cooperation with a group of stakeholders from the organization (Section 7.4.1). Next, we discuss the problem list that the same group of stakeholders constructed (Section 7.4.2). We conclude this section with a discussion of the structural validation of the empirical model.

To conclude this introduction, we briefly point out the two reductions we made during the transition from the conceptual model to the empirical model:

1. We did not include the public prosecutor in the empirical model. Although this actor plays an important role in the preparation of projects, his involvement is limited to some discussion with the CID management. Moreover, this involvement does not give rise to any problems as far as the CID is concerned.
2. The unit chief is not included in the empirical model either. He only performs a passive function with respect to the preparation of projects: receiving approved project proposals.

7.4.1 Organizational requirements

The first part of the empirical model consists of a set of organizational requirements. These requirements have to be met because the CID and the Information Department to perform their function adequately and achieve its goals. The set of organizational requirements was identified as a part of a 7 hour group session with ten participants from the CID [Hengst and Vreede 1993]. This group not only comprised the CID management and the ID chief, but also the key people in the constituent former departments of the Information Department. During the session the group used GroupSystems V, in particular the modules Idea Organization, for brainstorming and categorizing requirements, and Vote, for identifying the most important requirements.

For an extensive discussion of the organizational requirements, we refer to [Meel et al. 1994]. In this section, we restrict ourselves to an outline of the most important requirements:

1. *Insight into the work domain of the CID.* For the CID's market driven strategy to be successful, there has to be insight into (1) known criminal organizations and relevant societal trends, and (2) the 'clients' of the CID that request or require its service.

2. *Clearly stated goals and policy.* The policy of the CID is determined by the law, the top management of the Amsterdam Municipal Police Force, the political agenda, and the clients of the CID. The goals that are derived from the department's policy should be clearly stated and operationalized into non-overlapping assignments for all departments and subdepartments of the CID.
3. *Coordination between departments.* The CID 'produces' a complex product: the combat against organized crime. This is most often the result of the joint effort of the various (specialized) actors of the department. Hence, optimal orchestration of these efforts is a critical success factor for the successful operation of the CID.
4. *Manageable Information Department.* In achieving the previous requirement, the activities and processes in the Information Department should be organized in such a way that they are controllable and can be coordinated explicitly.
5. *Skilled personnel.* The right man or woman at the right place. There should be possibilities for extra training and education, especially with respect to information management and analysis of the ID employees.
6. *Adequate IT support.* Especially for the Information Department, the availability of automated support for the storage, processing, and retrieval of information is perceived as being critical. Information from various sources needs to be managed and represented in a well organized and surveyable way.

In table 7.2 we summarize how each of the organizational requirements relates to one or more of the main processes in the CID. This helps us to delineate the part of the conceptual model for which we are going to identify the main problems. In other words, it helps us to focus on the right issues. The relation between a process and an organizational requirement can be of two forms. The first form concerns attaining requirements by successfully executing a process. An example of this form is processing tips that help to achieve insight into the CID's work domain. The second form concerns successfully executing a process by attaining a requirement. An example of this form is the coordination between departments required for the execution of a successful project.

It is clear from table 7.2, that the Information Department has to render the key contribution of attaining the most important organizational requirement: insight into the work domain. Furthermore, the apt processing of information requests should enhance coordination between the Information Department and other departments in the CID. In order to attain the most important organizational requirement, the other requirements have to be met as well. The processes in the Information Department are not expected to yield the desired result without adequate IT support, skilled personnel, clearly stated goals and policy, and if the department's processes are not controllable.

Organizational Requirements						
	1. Insight into CID's work domain	2. Clearly stated goals and policy	3. Coordination between departments	4. Manageable Information Department	5. Skilled personnel	6. Adequate IT support
Processes						
1. Build information position						
a. Process tips	x	y	-	y	y	y
b. Process information requests	x	-	x	-	y	y
2. Prepare project	x	y	y	y	y	y
3. Perform desk research	x	-	y	-	y	y
4. Plan project	-	y	y	-	y	-
5. Execute project	-	-	y	-	y	y
6. Determine CID policy	y	x	-	-	y	-

x = process should support the attainment of the organization requirement

y = organizational requirement should support the successful execution of the process

table 7.2 The relation between the CID's organizational requirements and processes.

The importance of the process which determines the CID policy is also clearly recognizable. This process, obviously, supports the formulation of goals and policy, which in turn is required for processing tips and preparing projects. In effect, the determination of the CID's policy indirectly enables the attainment of insight into the CID's work domain. The catch is that this insight is also required to formulate the CID's policy.

In the next section, we examine to what extent problems occur that have a negative impact on the relevant processes.

7.4.2 Problem list

The second part of the empirical model consists of a list of problems with respect to the current situation. Following the aforementioned results of the organizational requirements, the problem list focuses on the Information Department. As we described earlier, the Information Department did not have its final organizational form at the beginning of our study. At this point, the Information Department only was on 'paper' a blend of the former specialized departments, whereby a single department was

created by merging two subdepartments for 'hard' and 'soft' information. The Information Department only had limited experience with the new market driven strategy in a number of experimental projects. Hence, the problems that the Information Department currently faced, partly refer to the experiences during these first projects, and partly to the situation just prior to the creation of the new department. Some of these problems we described in general terms in Section 7.1. However, to actually design the processes and coordination in the Information Department, we needed a more thorough insight into the problems of the situation prior to the establishment of the department.

The problems in the 'current' situation were identified by the same group of stakeholders that identified the CID's organizational requirements. Again, Groupsystems V was used to support the brainstorming (Idea Organization), organizing (Idea Organization), and prioritizing (Vote) activities. The original list of problem statements that were identified by the participants during the session, included more than 30 items, some of which were overlapping. In this section, we summarize the most important problems. These are organized in three categories: work process, project information (the product of the Information Department), and 'public relations'.

Work process

- *Autonomy.* Almost all of the former departments of the Information Department create their own workload, based on personal interest and specialization, with no consultation. This lack of communication creates a situation where certain information is sometimes collected more than once. In fact, the same type of activities are being performed in more than one department for the same type of information.
- *Late guidance of project preparations.* The way in which projects are prepared is mainly based on intuition, experience, and personal interests. Guiding decisions of the actors that have the final responsibility, are at too late a stage in the process. As a result, project proposals sometimes cannot be forwarded to a unit, get out-dated, or even lack the required quality. In addition, the nature of the resulting project proposal sometimes does not match the expertise and activities of the unit that has to execute it. This has serious consequences for motivating the unit personnel and the potential success of a project.
- *Difficult to focus.* It is difficult to focus the efforts with respect to information collection and project preparation as there is no or insufficient knowledge about which are the current problem areas in society and which should have the highest priority.
- *Transition preparation → execution.* It appears to be difficult to define exactly where the preparation of a project stops, and where its execution begins. The preparation phase is often extended as this has certain advantages when the

outcomes of the project are finally presented in a court of law. However, the consequence is that this requires extra effort from the Information Department.

- *Lack of direct control.* The majority of effort of one of the former departments, the Political Intelligence Department (PID), is requested directly by external parties. This makes it difficult for the CID management and the ID chief to coordinate the efforts of the actors in this department.

Project information

- *Information quality.* Because of incomplete information arriving at the Information Department and the limited capacity that is available in some of the former departments, information is often only *registered*. Its quality is not always improved ('enriched') by combining and analyzing multiple pieces of information.
- *Property of information.* The information that the PID collects for external clients, is the property of these clients. Hence, it cannot be used by the CID. In other words, a great deal of the PID effort is not beneficial for the Information Department or the CID.
- *Information security.* The security measures with respect to 'hard' information are considered to be insufficient.

Public relations

- *Services unknown internally.* The exact services of specific former departments are not well known to other departments. This does not encourage a sense of cooperation and integration.
- *Services unknown externally.* The exact services of the Information Department as a whole are not yet clear to the other parts of the CID and the other main departments of the Amsterdam Municipal Police Force. Consequently, the department's potential with respect to granting information requests is not fully utilized.

It is clear, given the above problems, that the organizational requirements for the CID are not met in the current situation. We will further discuss this in Section 7.5. First, we conclude this section by describing how the empirical model was validated.

7.4.3 Model validation

There were no objective criteria available to test if the empirical model was a correct representation of the CID's current situation. Therefore, we had to depend on the opinion of the CID's employees. Obviously, we had achieved a type of 'implicit' verification and validation because we involved several employees in the modelling process. In addition, we took two explicit steps to further validate the empirical model.

First, we distributed a questionnaire to the participants of the session in which the model was built. Amongst other things, we inquired whether the session results confirmed their personal views of the current situation. On a scale of 1 to 5, 1 being extremely negative, they responded an average of 3.9 (standard deviation 0.4). Second, we presented the empirical model to a substantial number of other CID employees than those who had participated in the session. This presentation did not yield any corrections or extensions to the model. Considering this, we found that the empirical model could be regarded as a correct and valid representation of the CID's current situation.

7.5 Problem diagnosis

In this section, we will diagnose the problem situation at the CID by taking a closer look at the processes that were identified as part of the conceptual model, and the problems that were included in the empirical model. We do not address the problems in the 'public relations' category. This category does not directly relate to the assignment of our case study, namely the organizational design of the Information Department. However, the ID chief and the CID management recognized the above mentioned problems in this category and initiated a number of informing presentations to the potential clients of the Information Department.

In table 7.3 we depict the problems with respect to the Information Department's work processes. The table further identifies which processes of the CID in general are affected by these problems.

Problem	Processes affected
Autonomy	Process tips, Prepare project, Perform desk research
Late guidance of project preparation	Prepare project, Perform desk research, Plan project, Execute project
Difficult to focus	Process tips, Prepare project
Transition preparation → execution	Prepare project, Perform desk research, Plan project, Execute project
Lack of direct control	Process tips, Process information requests

table 7.3 The processes affected by the identified work process problems.

From table 7.3 it follows that there are few problems with respect to processing information requests. Only the fact that some of the PID activities cannot be influenced by the ID chief or the CID management sometimes causes some problems. Another problem concerns the transition from the preparation phase to the execution phase. It is clear that this problem affects almost entirely, the operation of the CID. Still, we

will not further address this problem as it mainly involves legal aspects that are beyond the scope of our study.

Notwithstanding the importance and relevance of these two problems, far more important are the problems that together seem to jeopardize the coordination of the Information Department's effort and thus have a very negative impact on fulfilling the most important requirement: insight into the CID's work domain. These problems concern the autonomy of actors, the late guidance of project preparations, and the difficulty to focus. In order to deal with these problems, there is especially a need for clearly stated goals and policy, and for an organization of the Information Department that is easy and flexible to manage. In other words, alternatives for organizational change in the department should focus on these two issues:

1. The process with which the goals and policy of the CID is determined and the role of the Information Department in this process.
2. The more explicit coordination of the processes in the Information Department, which, for example, relates to decisions that determine the selection of a project idea and the course of a project proposal. Also, the capability of the actors to work in synergy should receive attention.

The problems with respect to project information and the processes they affect are depicted in table 7.4.

Problem	Processes affected
Information quality	Process tips, Process information requests, Prepare project, Perform desk research, Plan project, Execute project
Property of information	Process tips, Prepare project, Perform desk research
Information security	Process information requests

table 7.4 The processes affected by the identified project information problems.

The security problem relates to 'hard' information. Currently, adequate measures have been taken to deal with it. The problem with respect to the property of PID information may affect all major processes of the Information Department. However, legal regulations prevent the CID from solving this problem itself.

Obviously, the most important problem with respect to project information is the quality of this information. This problem not only affects the processes involved with the preparation of projects, but also those with respect to executing projects. The main reason for this problem is that pieces of information are not combined and 'enriched'. This is partly due to the fact that there is as yet insufficient coordination between actors dealing with different information of which the blend could be worthwhile.

Also, the technical support for the collection and analysis of information is too diverse to allow for merging information, analyzing it, and presenting it in a meaningful way. Finally, there is as yet scant experience with methods and techniques to combine and present information in such a way that a clear picture of a criminal organization or a societal trend unfolds.

It is evident that the information quality problem also has a very negative impact on the attainment of insight into the CID's work domain. In order to deal with this problem, it is suggested that attention be focused on two issues:

1. Changing the Information Department's work processes in such a way that actors that naturally have to cooperate in order to combine and enrich information, are more or less automatically forced to do so.
2. Developing IT to support the collection and enrichment of information and to allow the employees of the Information Department to build and represent their knowledge about criminal organizations and societal trends in a comprehensive and useful way.

With the four above issues in mind, a number of change alternatives were developed in cooperation with the various stakeholders from the CID and external parties. These alternatives are introduced and elaborated on in the following section.

7.6 Alternatives for change

As stated in the previous section, change alternatives should be aimed at improving the CID's policy making process, at designing flexible work processes in the Information Department that are coordinated explicitly, and at developing IT support for handling project information. Together with various stakeholders, four alternatives for change were identified and worked out in more detail. These alternatives include a multi criteria decision making process to define the department's policy in terms of a project agenda, a functional structure for the Information Department, a new product structure for the same department, and an information system that supports both the handling of project information and the policy making process of the first alternative. Each alternative is discussed and evaluated in the following subsections.

7.6.1 A multi-criteria policy making process

The CID's policy making process aims to identify which problem areas are perceived as being the most important in the sense that projects into these areas have to be initiated. A problem area may be a specific type of criminal organization or societal

trend, but it may also refer to a more general area of crime, such as fraud or eastern European organizations.

An important characteristic of the CID's policy making process is that there may be a lot of actors involved. These include not only the Information Department representatives and the CID management, but also the so-called Triangle (the mayor of Amsterdam, the chief commissioner of the Amsterdam Municipal Policy Force, and the chief public prosecutor in the Amsterdam area), and various top officials from the Ministry of Justice and other judicial organizations. These actors may contribute to the overall picture of problem areas that have to be considered. They may also have a role with respect to deciding which areas should receive the highest priority.

Together with a number of CID stakeholders, we discussed the way in which the policy making process could be organized. A significant point of consideration was that it is often not easy to compare two problem areas directly. It was, therefore, recognized that for a more detailed and more conscious comparison, the process should use a set of criteria to compare problem areas with respect to different aspects. Since a large group of actors are involved in the process, it takes the shape of a group multi-criteria decision making (MCDM) process. This process is depicted in figure 7.6.

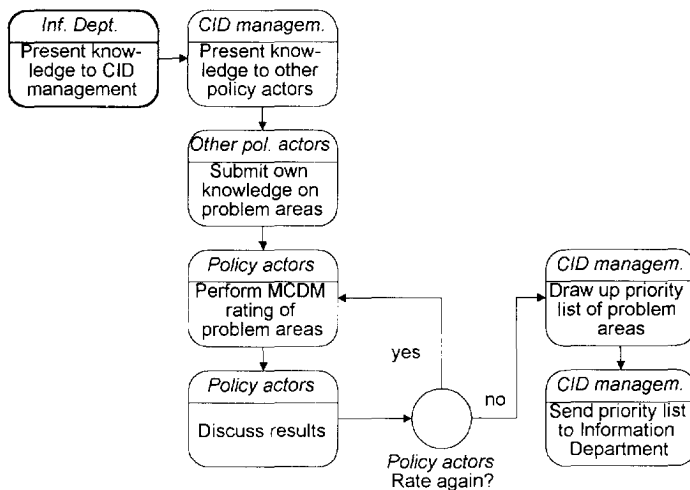


figure 7.6 Process model of the policy making process.

In this process, the Information Department keeps the CID management up-to-date with respect to their knowledge of problem areas. The CID management presents this knowledge to the other actors in the policy making process. In return, the other actors submit their knowledge about these and other problem areas. The next task concerns the MCDM activity itself. Before the actual rating of the problem areas can take place,

a number of decisions have to be made first. It has to be decided which problem areas are included in the rating process, which criteria are going to be used, and which weights these criteria have. Also, it may occasionally be decided to invite other actors, such as external experts in certain areas, to participate in the rating process.

The outcome of the MCDM rating of the problem areas is then discussed. During this discussion, a need to rate the problem areas again may unfold, for example, because it appears that certain actors had different (and contradicting) conceptions of certain problem areas. If there is no need to redo the rating, the CID management draws up a priority list of problem areas. It should be noted that this priority list is not necessarily a direct copy of the rating results, but may be an interpretation of these results. Finally, the CID management passes this priority list on to the Information Department. This enables the Information Department to prioritize the collection and analysis of tips. It will be also valuable for the approval of project ideas and proposals by the ID chief and the CID management.

In order to test the value and appropriateness of this process, three trial policy making processes were conducted. These processes all took place in the Group Decision Room, using GroupSystems V to support the identification and prioritization of problem areas and criteria. Apart from the Idea Organization and Vote modules, that were used for brainstorming and simple voting procedures, Alternative Evaluation was used to support a multi criteria assessment of problems areas. Each of the three sessions involved a group on a different level of policy making. This was done in order to gradually increase the complexity of the group setting. The three levels included:

- The *local* level consisting of representatives of the Amsterdam Municipal Police Force [Vreede et al. 1993b].
- The *regional* level consisting of representatives of the Amsterdam Municipal Police Force, of the Mayor of Amsterdam, and of Amsterdam judicial authorities (the Triangle) [Vreede et al. 1993c].
- The *national* level consisting of representatives of National Organized Crime Task Forces, of Specialized Investigations Organizations (for example the Dutch IRS), and of the Ministry of Justice [Vreede et al. 1993d].

During these sessions, the satisfaction of the participants with the process and its outcome was very high. The process was accepted as adequate to support the determination of the CID policy with respect to problem areas. However, one major point of consideration remained. During the three sessions the groups were not yet able to use a mature and well tested set of criteria. Although the session participants did find that the sessions' results gave a truthful reflection of the group's perception, they felt that a more refined set of criteria could improve the quality of the process results. It was also expected to increase the necessary buy-in into the results by other actors

that were not involved in the process. Therefore, an extra session was organized in which a group of experts from within and outside of the policy group used GroupSystems V to define a balanced set of criteria [Hengst et al. 1993a]. Initially, over 80 criteria were identified. Two ‘organizing’ activities (see Chapter 6) brought this number down from 36 to 10. The 10 final criteria were adopted by the policy group for a trial period.

7.6.2 A function oriented Information Department

The second change alternative addresses the coordination of the work processes in the Information Department. It was acknowledged that the current specialization in terms of information type, caused the majority of coordination problems in the Information Department’s work processes. With this in mind, a group of stakeholders from the Information Department and the CID management decided to transform the organization of the Information Department into a *functional hierarchy* [Hengst 1994; Leeuwrik and Otte 1993]. It was agreed that concentrating on information *functions* rather than on information *types* could enhance the coordination between the employees working with the same information and could thereby increase the information quality. The resulting subdepartments (and consequently the main functions) in the reorganized Information Department are listed below:

- *Collection.* This subdepartment is responsible for the collection of both ‘hard’ and ‘soft’ information with respect to traditional police information and information on societal trends. The subdepartment also processes this information in such a way, that it can be used by the other subdepartments of the Information Department.
- *Advice.* This subdepartment composes an overview of (known) problem areas, i.e. the work domain of the CID. The overview, together with an evaluation, is then presented to the CID management along with advice of which project to prepare.
- *Preparation.* After receiving an assignment from the CID management, this subdepartment collects information on a target, which may be a criminal organization or a societal trend. This results in a project proposal, which after the customary approval procedure, is passed on to a unit.
- *Distribution.* This subdepartment is responsible for handling information requests from other CID departments and from other police units. In other words, this subdepartment serves as a front desk to the ‘clients’ of the Information Department.

The work processes of the first three subdepartments are depicted in figure 7.7. We will not elaborate on these process models, as they are self explanatory. For a further discussion of the work process of the distribution subdepartment, we refer to [Hengst 1994; Leeuwrik and Otte 1993].

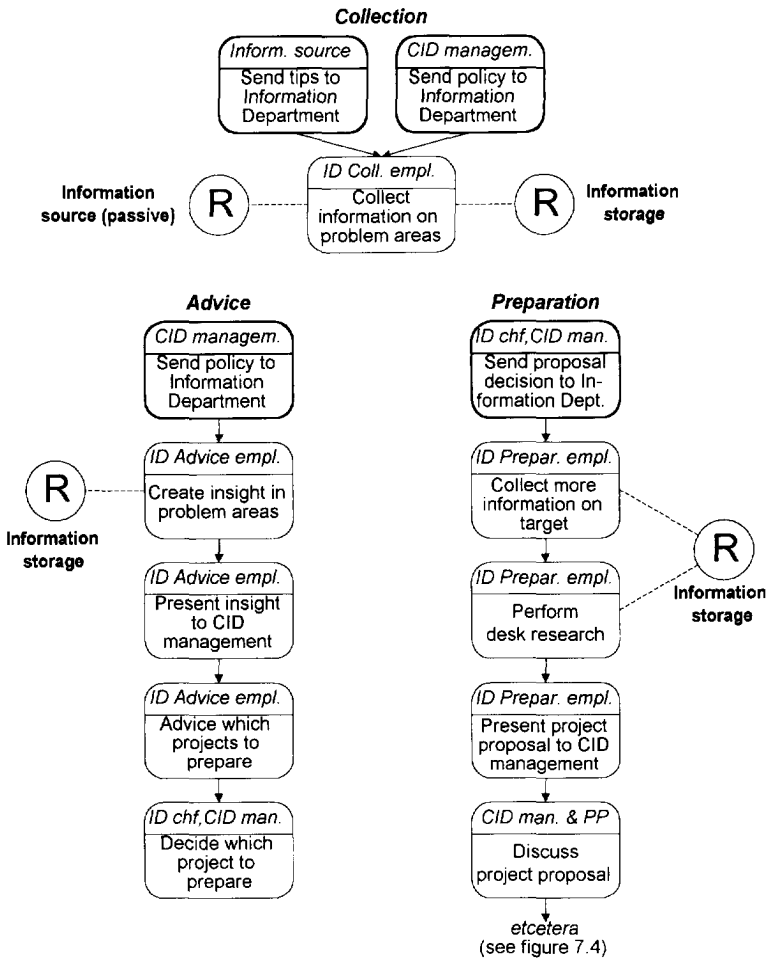


figure 7.7 Process models for three of the functional subdepartments.

The functional organization of the Information Department has several advantages compared to the current situation. First, the Advice subdepartment forces the ID chief and, in particular, the CID management to take an active part in directing the preparations of projects. In this way, the CID's policy process is significantly enhanced. Second, clustering all types of information functions into separate subdepartments facilitates coordination *within* these functions. The risk that certain activities are unnecessarily carried out more than once is decreased. Third, instead of several subdepartments each performing all the required functions, there is now only one subdepartment for each function. This decreases the complexity of coordinating the Information Department by the ID chief and the CID management.

However, there are also a few points of consideration. First, there remains a legal obligation for the Collection subdepartment to separate criminal information from political and societal information. In this sense, some coordination issues in this subdepartment will have to receive explicit consideration. Second, the final product of the Information Department, an approved project proposal, is always the joint result of the first three functions. This implies that in the new situation, the coordination *between* the functions in the Information Department will require additional attention. With respect to this last point of consideration, we propose a second change alternative for the organization of the Information Department.

7.6.3 A product oriented Information Department

In order to facilitate coordination between the functional subdepartments in the Information Department, it was proposed to further develop the organizational structure of the department into a *product hierarchy*. Although the current situation also has the form of a product hierarchy, here the characteristic product is the *type* of information that a subdepartment processes. In the proposed new situation, the characteristic product will be the *output* of the Information Department, which are its project proposals. For each project proposal, an *information team* is formed that includes the required expertise from all functions in the department. Hence, information teams operate in a *market driven* way *within* the functional hierarchy of the Information Department.

The decision to form an information team is made by the ID chief in cooperation with the CID management, see figure 7.8. Although an information team has the final responsibility for the quality of its product, it is not allowed to operate entirely autonomously. During the preparation of a project, the CID management and the ID chief have to make an explicit decision regarding the continuation of the preparation of a project, see figure 7.8.

An information team is always headed by the chief of one of the functional subdepartments. Together, the chiefs of all information teams are directly accountable to the ID chief. The focal point of the team chiefs' responsibility is to prevent an information team from generating its own input and thus become uncontrollable. To this end, there is a weekly meeting between all team chiefs and the ID chief.

After an initial period of working according to the functional alternative described in the previous section, the Information Department and the CID management decided to adopt the information team alternative. An important consideration in this respect was the upcoming implementation of the ACOST systems that specifically supports the activities of an information team. The ACOST system is discussed in the next section.

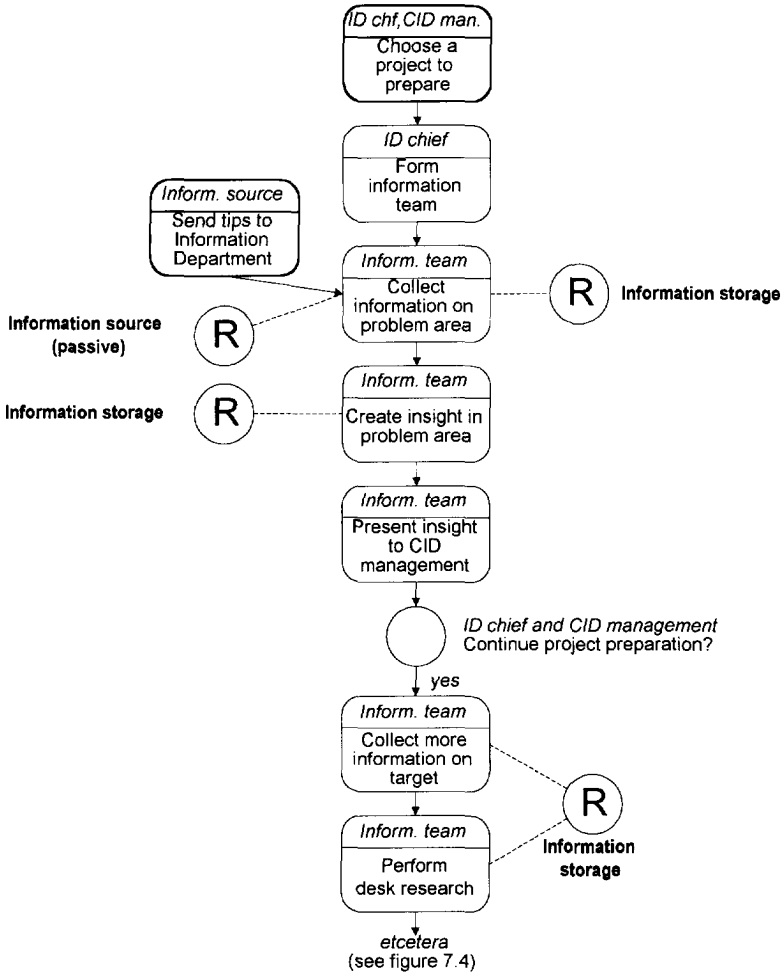


figure 7.8 Process model of an information team creating a project proposal.

7.6.4 The Atlas of Criminal Organizations and Societal Trends

As is apparent from the discussion of the previous three change alternatives, the actors in the Information Department have to process and exchange a lot of information to perform their work processes. This information mainly concerns criminal organizations and societal trends. In order to support these actors and the work processes they participate in, it was decided that an information system that would facilitate the processing and communication of relevant information in the Information Department and the CID be developed. More specifically, the goal of this system was twofold:

1. To support the policy making process described in Section 7.6.1.
2. To support the management and communication of information that is used during the preparation of CID projects into criminal organizations and societal trends.

In order to achieve these goals, the information system had to support the creation and maintenance of a thorough and up-to-date insight in the work domain of the CID, i.e. the CID's most important organizational requirement. With this in mind, the system was named the Atlas of Criminal Organizations and Societal Trends (ACOST). The ACOST would not only fulfill the role of the 'Information storage' repository, but also support a number of other activities, for example the first four activities in figure 7.6 and 'Present insight to CID management' in figure 7.8. In this section, we first discuss the way in which the ACOST represents both criminal and societal information. Next, we sketch the main functions of the system. We then discuss the development of a prototype of the ACOST and conclude with an account of the pilot period in which the ACOST prototype was employed in the Information Department.

In the ACOST, criminal organizations and trends in society that may give rise to organized crime, are both referred to as *problem areas*. The ACOST represents information on problem areas at three levels of abstraction:

1. The lowest level deals with specific problem areas, so-called *Fields of Attention*. At this level, the people and organizations involved are identified by name. An example of a Field of Attention is the drugs organization Fata Morgana, that launders money via various banks in Luxembourg and Switzerland.
2. The middle level deals with problem areas in a more abstract manner. This level describes so-called *Areas of Attention* which represent the conceptual structures according to which criminal activities or societal trends take place. Areas of Attention do not refer to organizations and persons by their own name, but by the name of the type or class that they represent. An example of an Area of Attention is a class of drug organizations, resembling Fata Morgana, that launder money according to a general construction involving various banks.
3. The highest level of abstraction is called the *Topics* level. On this level a number of Areas of Attention are grouped together according to some common characteristics. These characteristics may, for example, refer to the policy of the CID. If the CID's policy states that fraud should be high on the department's agenda, a Topic 'fraud' could include the following Areas of Attention: money laundering, cover firms, and letterbox companies.

The importance of the distinction between Areas of Attention and Fields of Attention was brought out during the three policy making sessions with GroupSystems V in which various problem areas were identified, see Section 7.6.1. Specific descriptions in a Field of Attention support the preparation of projects that focus on particular

persons or organizations and are often of a corrective nature. Conceptual descriptions in an Area of Attention support the preparation of projects that focus on general procedures or developments that give rise to organized crime. A result of such a project is often of a more preventive nature. An example is that the CID advises certain regulations to be issued in order to frustrate the laundering of money.

In order to identify the required functionality of the ACOST, the results from the three policy making sessions were combined with those of the first session. In addition to the identification of the CID's organizational requirements and problems, the participants of the first session brainstormed about what type of information a system like the ACOST should be able to handle, and what type of activities should be supported. The combined outcomes of the four mentioned sessions resulted in the specification of the following five main functions that the ACOST should offer:

1. *Collection.* The ACOST must support collecting two information types: non-related information and related information. *Non-related information* refers to information that might belong to a certain problem area which at the time of collection is not known well. The ACOST must store non-related information as 'scans'. The collection of *related information* is triggered by a desire to learn more about a certain problem area that is already known to the ACOST. This type of information may refer to certain actors (people or organizations) in the problem area, or to relationships between these actors. Related information must be stored as 'documents' which are linked to problem areas.
2. *Insight.* The ACOST should visualize, both textually and graphically, criminal organizations and societal trends in terms of actors and relationships on the three levels of abstraction that were introduced above.
3. *Selection.* The ACOST must allow criteria to be defined so that users can evaluate Fields of Attention and Areas of Attention. Based on this evaluation, it must be possible to generate single criterion or multi-criteria prioritized lists. When criteria are defined and rated according to the CID's goals, the management can use the ACOST for deliberately selecting which investigations to prepare or initiate.
4. *Project definition.* Within the ACOST, projects must be definable with respect to Fields of Attention or Areas of Attention. During additional investigations, documents with related information on elements of the problem area are added. When the preparation is considered to be sufficient, information relating to the project is handed over to the appropriate unit.
5. *Reports.* The ACOST must be able to print or save reports of documents, scans, Topics, Areas of Attention, and Fields of Attention using various selection criteria.

For each of the ACOST's main functions a number of subfunctions were defined as well, for example enter, delete, move, or edit a scan. For a detailed description of the ACOST's functional specification, we refer to [Hengst 1994; Meel et al. 1994]. Based

upon the ACOST's specifications, a first prototype of the system was built [Aalst et al. 1993a, 1993b, and 1993c]. As the Information Department had just installed an MS-Dos PC network, it was decided that the prototype should be constructed in CA-Clipper, including the graphical library DGE. Apart from being suitable for developing database applications, CA-Clipper allows stand-alone network applications to be built. This means that CA-Clipper is not required to execute the application.

The first version of the ACOST prototype covered about two thirds of the specified functions. Functions that were not included addressed more complex manipulations on the databases and a number of complex reports. To be able to evaluate the prototype in a meaningful way, a set of realistic information on problem areas was included in the system. This information was taken from the results of the regional policy making session, see Section 7.6.1. In addition, a number of criteria from the results of the criteria session were included to facilitate a sensible evaluation of the problem areas.

<u>Scans</u>	<u>Problem areas</u>	<u>Policy</u>	<u>Find</u>
	<i>Topic</i>	Top 10 lists	Area of Attention
	- document	Criteria	Field of Attention
	- <i>Area of Attention</i>	Scan management	
	- document	Reports	
	- evaluation	User management	
	- project		
	- area actor		
	- document		
	- relation		
	- document		
	- <i>Field of Attention</i>		
	- document		
	- evaluation		
	- project		
	- field actor		
	- document		
	- relation		
	- document		

figure 7.9 The menu structure of the ACOST prototype.

The menu structure of the prototype is given in figure 7.9. As can be seen, the functional specification has not been directly mapped onto the menu structure of the ACOST. Figure 7.10 and figure 7.11 depict how problem areas are represented textually and graphically. Figure 7.10 illustrates that the Topic 'witwassen' (money laundering) includes three Areas of Attention. In figure 7.11, the actors and their relations are represented graphically with respect to the Area of Attention 'Exploitatatie gokverslaving' (exploitation gambling addiction). Figure 7.12 and figure 7.13 depict parts of the prototype that refer to the selection of problem areas. Figure 7.12 gives a screendump of the way in which particular criteria are rated over time for a certain problem area. In figure 7.13, the result of the multi-criteria selection of Areas of Attention is represented.

Categorisatie	
Superuser	
(Topic) Nederlandse crim. netwerken Milieucriminaliteit Witwassen Horeca 4e wereld conjunctie Internationale geweldpleging Italiaanse mafia	(Aandachtsgebied) Panamense firma's Kunsthandel Bankwezen
(a)anmaken (b)ekijken (w)ijzigen (v)erwijderen (<Esc>) (<F12>)	

figure 7.10 Screenshot of an overview of Topics and Areas of Attention.

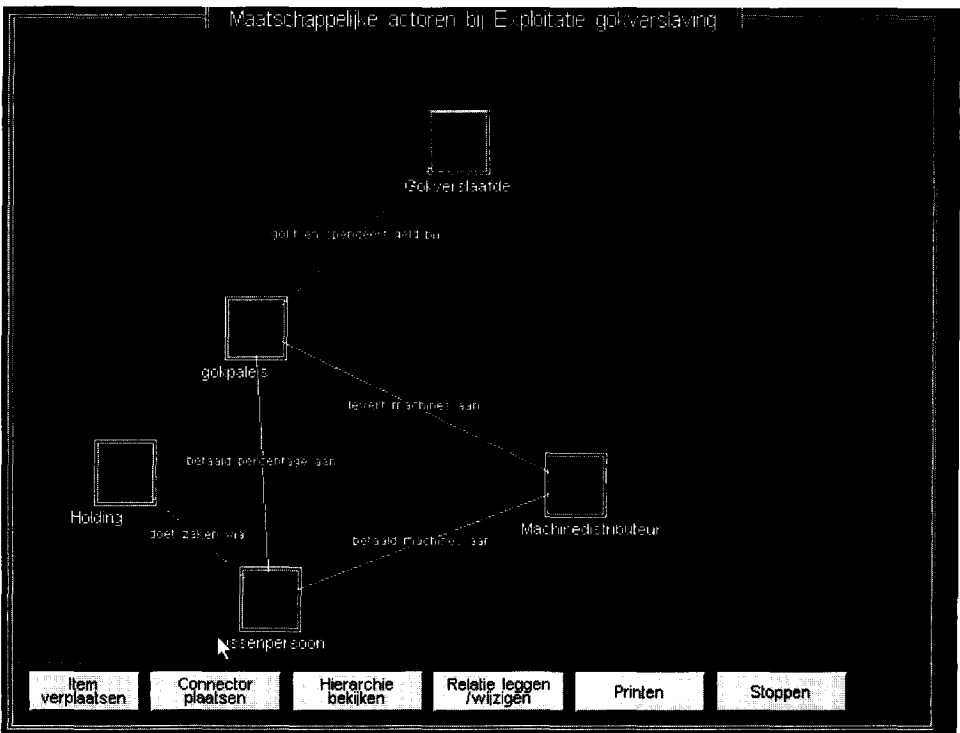


figure 7.11 Screenshot of a representation of an Area of Attention's structure.

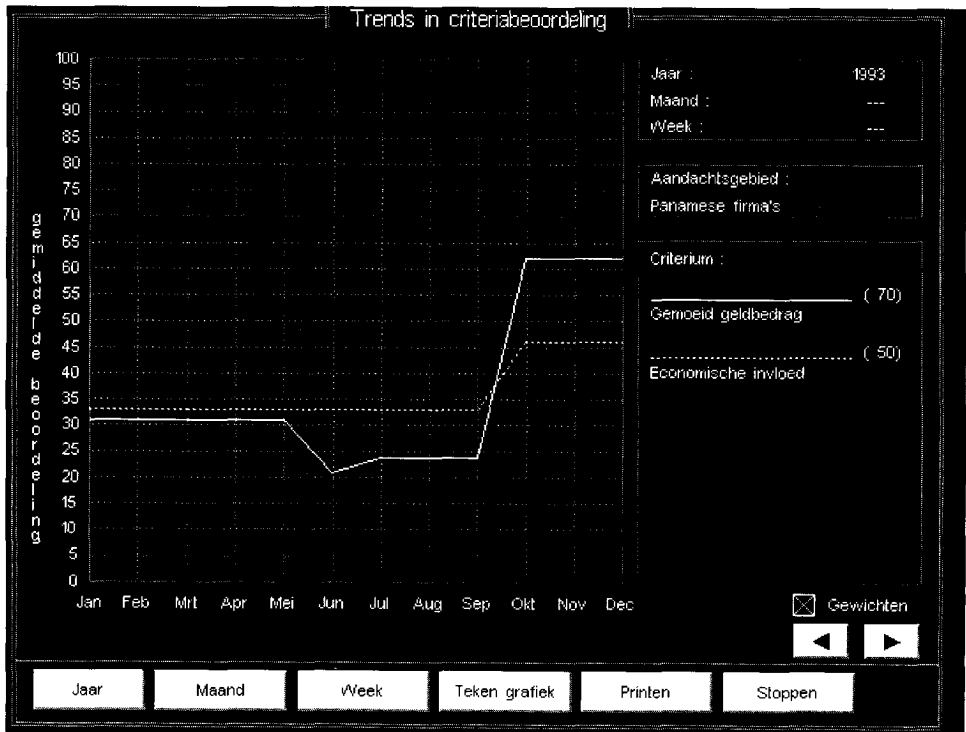


figure 7.12 Screenshot of the development of criteria over time.

Sorteren lijsten	
Superuser	
aandachtsgebied	gewogen indexcijfer
Bankwezen	84
Kunsthandel	80
Panamese firma's	64
Afvalverwerkende bedrijven	38
(<Esc>) in(f)ormatie (p)rinten	

figure 7.13 Screenshot of the results of evaluating certain Areas of Attention.

The actual evaluation of the ACOST prototype took place in a GroupSystems V supported session with nine stakeholders participating [Hengst et al. 1993b]. After an introduction of the prototype, the session participants received a number of informal assignments that forced them to use all the functions included in the prototype. While using the prototype, GroupSystems V was running simultaneously. Participants were sufficiently motivated to enter their ideas, problems, questions, and wishes module with respect to the prototype in the Topic Commenter. In total, the participants made 65 contributions. After using the prototype, the participants numerically evaluated the main parts of the system with respect to criteria such as functionality, user friendliness, and presentation. The results of this evaluation showed that the 'Scans' and 'Problem areas' parts were liked slightly more than the other functions: 7.4 against 7.0 (standard deviation between 0.6 and 1.2) on a scale from 1 to 10. From these results it can also be deduced that the overall opinion of the participants with respect to the prototype was altogether positive.

Given the positive feedback of the participants, it was decided that a three month pilot period with the ACOST prototype in the Information Department be initiated. Before this pilot was started, a few bugs were fixed and a number of very much desired modifications were carried out. During the pilot period, the ACOST prototype running within the Information Department was further enhanced in terms of reliability and security. Also, the remaining functions were programmed, but not yet included in the prototype in use. This was done to present the ACOST to the users as a 'stable' system in the sense that it did not change every week.

At the end of the pilot period, a final GroupSystems V session was organized with the users of the prototype. In this session both the prototype in use and the implementation of the remaining functions were evaluated. The feedback of the participants during this session was positive to the extent that it was decided that the ACOST be implemented as soon as possible. To this end, the prototype was upgraded into the ACOST version 1.0 by including the remaining functions, and by determining and incorporating levels of authorization for the different users of the ACOST. Version 1.0 was officially put into use one week after the last session.

7.7 Reflection

The CID represented an interesting and valuable environment for employing both the dynamic modelling approach and the techniques and support for participative design. In the study we encountered a clear need to restructure the processes of the organization and their coordination as well as design coordination support for the new processes. In this concluding section, we present our reflections on this case study with

respect to the case situation itself, to the dynamic modelling approach, and to the techniques and support for participative design.

7.7.1 The CID

As of this writing, the restructuring of the Information Department is completed. Contrary to the initial plans, the PID has not been merged into the Information Department. Regulations and practical experience learned that it is necessary to keep political information separated from other types of information. Apart from this exception, the Information Department is currently working according to the product hierarchy alternative described in Section 7.6.3. Information teams are continuously formed and employed to prepare projects. The actual change from the situation at the beginning of our study to the new product oriented situation was facilitated by a major renovation of the CID building. According to the employees, this stimulated the instalment of information teams and both the intra- and inter-functional cooperation.

An important issue remaining for the Information Department is cooperation with external experts with respect to certain problem areas. Organized crime is not a problem that the police can handle alone. During scanning activities and the collection of more information on a problem area, the CID often requires the assistance of external actors. The Information Department is currently looking into methods and technical support to facilitate to *joint* attainment of insight in organized crime with experts from outside the department. Experiences show that GSS can effectively support this effort [Hengst et al. 1995].

A network version of the ACOST has been in use for over a year. We observed a 'bond' between the users and the system. The ACOST is accepted as a joint effort instead of something imposed upon the organization by system developers. The following quotation from one participant during the final GroupSystems session is illustrative of overall attitudes: *'The ACOST is a system that we built together. And it's something that we should be proud of and proud to present to others in the force.'*

Since the introduction of the ACOST, the type of projects that are prepared and executed, has changed. The actors responsible indicated that there are significantly more projects of a preventive nature than in the old situation. Also, projects focus on a larger variety of organized crime rather than mainly on drug trafficking. It is not likely that this is only the result of the implementation of the ACOST and the new working processes in the Information Department. The change in the CID's strategy can also be held responsible. However, with respect to the CID's annual plan for 1995, the ACOST's contribution is clear. To the satisfaction of all parties involved, this plan was composed using information from the ACOST.

There are plans to implement a distributed version of the ACOST in the district police stations of the force. Although these police stations focus their efforts on Market I crime, they are able to provide the CID with a lot of information about local instances of organized crime or about particular actors that are part of a criminal organization. Apart from regional interest in the ACOST, there are currently also a number of CIDs in other police forces in the Netherlands looking into the applicability of the ACOST to their work situation.

Apart from these positive outcomes, there is also a serious point of consideration for the future employment of the ACOST. The ID chief and the ID employee responsible for the ACOST indicated that the focus of the Information Department has been predominantly on collecting information with respect to societal trends on the level of Areas of Attention. As a result, units sometimes complained that they did not get clearly defined project proposals and project goals. In order to deal with this problem, the Information Department is planning to shift its focus more towards criminal organizations on the level of Fields of Attention. Follow-up projects could then concentrate on Areas of Attention.

7.7.2 *The dynamic modelling approach*

The CID case study was initially used to apply and evaluate the dynamic modelling approach. This resulted in three points of consideration. First, we tested the modelling concepts that were developed in the first two case studies for the *conceptualization* of the problem situation. This exercise turned out to be a rather time consuming exercise, which, given the complexity of the problem situation, could have been expected. In order to get an understanding of the situation, a substantial number of interviews were held and the conceptual model went through several iterations. In retrospect, we feel that this process may also have benefitted from the use of group support technology. Building the conceptual model together with a group of stakeholders may have been more economical in time than having to combine the results of individual interviews and feeding versions of the model back to the CID. To this end, it may be worthwhile to explore the applicability of the group modelling tool TeamGraphics. TeamGraphics allows different types of diagrams to be drawn by (distributed) groups, such as entity relation diagrams, data flow diagrams, and task structures. The facilitation of stakeholder involvement in the design of organizational processes and information systems is a seemingly fertile application area for this new group technology.

Second, it turned out to be practically unfeasible to *transform the conceptual model* into a simulation based empirical model. There are two reasons for this:

1. Over the past few years, the organization had been changing constantly in terms of structure and processes. It did not represent a stable environment. Hence, it was

difficult to build a simulation based empirical model that could be used as a yardstick to diagnose the current situation and evaluate change alternatives.

2. In our second case study in the NOD, we found that the availability of empirical data was a critical success factor for the successful application of the approach. This finding was confirmed in the CID case. First, there was an almost total absence of (performance) data sources. Second, there were predominantly Type II activities carried out [Panko 1984]. As these activities are difficult to measure and represent accurately with statistical distributions, the resulting model may not have displayed the required statistical correspondence to the real situation. Third, expert estimations could rarely be used. The instability of the case situation made it difficult for the people involved to become 'experts' and give meaningful estimations.

The final point of consideration concerns the *communication of organizational change alternatives* to the members in the organization. During the design phase of the study, it sometimes appeared to be difficult for the participants to grasp the intentions and consequences of change alternatives if these were represented with process models. Although it was easy to *understand* these models, they seemed to be insufficient to depict a future situation in a for the stakeholder *insightful* way. In this respect, the absence of a dynamic model of the alternatives was felt.

7.7.3 Techniques and support for participative design

In response to the problems with the dynamic modelling approach during the construction of the empirical model, we employed techniques and support for participative design as described in Chapter 6. Using these techniques and support, we were able to (1) identify the most critical problems as they were perceived by those involved in the case situation, and to (2) direct and facilitate the joint design and evaluation of alternatives for change. Below we reflect on the participative design techniques and support by first looking at the setting in which it was employed: group sessions. Then, we discuss the experiences with the GSS technology that we used throughout the process.

Group sessions

Apart from the fact that planning sessions with 8-14 participants sometimes proved to be cumbersome, the group sessions left an overall positive impression. This is expressed by the results of the first questionnaire question, see table 7.5.

We want to point out with respect to table 7.5, that in the original (Dutch) questionnaire, not all questions were formulated with a 'positive' orientation. For the sake of simplicity, we translated all questions in table 7.5 (and table 7.6) into positive oriented questions.

Question	Mean*	Std
1 Are you satisfied with the session?	4.2	0.7
2 Did the session results confirm your ideas on the session topic?	3.8	0.9
3 Did the session help to gain new insights on the session topic?	3.5	1.1
4 Has enough agreement been achieved among the participants?	3.8	1.0
5 Did you enjoy the collaboration with the other group members?	4.5	0.7
6 Do you approve of the composition of the group?	4.2	0.9
7 Are you willing to participate in a follow-up session or activity?	4.9	0.4

* On a five point scale, 1 being very negative and 5 being very positive.

table 7.5 Questionnaire results, part I.

It appears that the individual group members could identify themselves with the session results, see question 2. Also, the participants felt that the sessions to some extent helped them to increase their insights, see question 3, and that enough consensus among the participants was reached with respect to the session results, see question 4. The identification with the session results most probably increased the participants' acceptance of these results. We often noticed that seeing their individual demands and objectives from a group perspective made it easier for the participants to accept situations in which individual demands had to be sacrificed to group demands.

Finally, it appeared that there was a sense of unity within the group. Participants enjoyed cooperating very much, see question 5. On the questionnaires several participants expressed that the atmosphere of the sessions was open and friendly and that they could freely discuss all their questions and problems. In addition, the participants felt that all people significant to the study took part in the sessions, see question 6, although some noted that also the higher management of the police force should have participated in some of the sessions.

The participants' positive perception of the sessions not only facilitated priority setting with respect to involvement in the study and follow-up activities, but also with respect to the information and knowledge elicitation during the process. The remarkable positive results of question 7 illustrate this.

GSS technology

The use of GSS technology is a significant ingredient of the participative design approach. As is obvious from the case study, a GSS was used to support groups in different problem activities. With reference to table 6.6, GSS technology was predominantly employed during the construction of an empirical model and the models of possible solutions. More specifically, we used the GSS for the development and evaluation of an MCDM policy making process, for the elicitation of organizational requirement, problems, and functional specifications of the ACOST, for the collection of data for the ACOST pilot, and for the presentation and evaluation of the ACOST prototypes.

With respect to the GSS technology our overall experiences are positive. The technology, for example, was instrumental in clearly pointing out the amount of consensus within a group. This enabled the group to focus the discussion on relevant issues and saved a lot of time not discussing other issues. As some of the participants remarked: *'There were no unnecessary endless discussions'* and *'Not having to discuss everyone's individual arguments had a liberating effect'*.

These observations are supported by the participants' perceptions, see table 7.6. They felt that the quantity of session results was high, see question 8, and could even increase *'if we would have had more experience with the system'*. The participants overwhelmingly attributed this high productivity to the GSS, see question 9. As one participant said, *'There is little risk to get lengthy discussions with each remark or idea being submitted.'* When asked about the quality of the sessions' results, see question 10, the participants still displayed a positive impression, but a number of concerns were expressed as well. Some felt the session process went too fast. They would have liked to have had more time to process the results themselves (*'Let them sink in'*) and reflect on them.

Question	Mean*	Std
8 How do you perceive the quantity of the session results?	4.1	0.8
9 Do you perceive a GSS meeting's productivity higher than a manual meeting?	4.8	0.4
10 How do you perceive the quality of the session results?	3.8	0.9
11 How well did you understand GroupSystems V?	4.4	0.8
12 Did you find GroupSystems V user friendly?	4.7	0.5
13 Did you find GroupSystems V interesting to use?	4.8	0.4

* On a five point scale, 1 being very negative and 5 being very positive.

table 7.6 Questionnaire results, part II.

The participants expressed great enthusiasm about the particular GSS used, GroupSystems V. They felt they understood the technology very well, see question 11, and found it extremely user friendly and interesting to use, see questions 12 and 13. Nevertheless, several participants felt that the Alternative Evaluation module was difficult to comprehend. For a more elaborate discussion of experiences with GroupSystems V, see [Vreede and Sol 1994a].

8 A Design Approach to Facilitate Organizational Change

'But now we have with us some concepts that greatly alter the whole understanding of things.'

- Robert Pirsig

8.1 Introduction

In the first two case studies, we developed, applied, and evaluated the dynamic modelling approach. In our third study, we employed and evaluated participative design techniques and support. From the results of this third study, it appears that participative design techniques and support can facilitate the problem solving activities of the dynamic modelling approach in several ways:

1. The collection of data for model construction activities can be carried out efficiently using GSS.
2. Stakeholder involvement can be facilitated and supported throughout the process.
3. The use of GSS can structure the group processes during a study.
4. Following the specification of organizational change alternatives, prototyping techniques can be employed to jointly develop supporting IT.

In this chapter, we present a design approach for organizational change that embodies a blend of the concepts, methods, and techniques from the dynamic modelling approach presented in Chapter 3, with the participative design techniques and support presented in Chapter 6. The resulting design approach can be thought of as an *'enriched' dynamic modelling approach* as it offers the strengths of dynamic modelling, while covering a number of issues that were not yet specified in the original approach. With reference to Chapter 2 and figure 2.1, this design approach represents our prescriptive conceptual model for handling the problem of organizational change. The application of the approach will be illustrated and evaluated in Chapter 9.

The way of thinking of the design approach obviously remains the same as discussed in Sections 3.2 and 6.3. In the remainder of this chapter, we use the process of

problem solving as a basis to describe the design approach in more detail. A process model describing the main problem solving activities is given in figure 8.1.

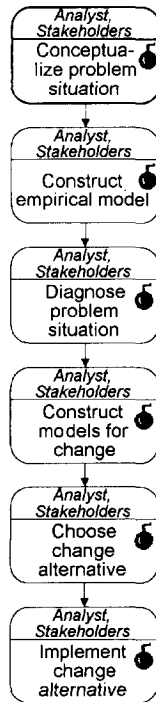


figure 8.1 Process model for the main problem solving activities.

In Sections 8.2 to 8.7, we discuss the main problem solving activities in more detail. We give a process model for each of these activities and describe the techniques that may be used for the various constituent modelling activities. In addition, we identify the ‘group’ nature of the activities and propose how GSS can be employed to support stakeholder participation in the process. It should be noted that for the sake of simplicity, the (constituent) activities are presented and described in a *linear* fashion. The last section of this chapter presents some concluding remarks. The analyst is referred to in the masculine tense.

8.2 Conceptualize problem situation

The first activity in the problem solving process is the conceptualization of the problem situation. The goal of this activity is to model the structure and the boundaries of the problem situation. This will facilitate the subsequent collection of empirical data that is required for the further specification and diagnosis of the problem situation. The

result of the conceptualization activity consists of a set of object definitions, an interaction model, process models, and actor models. The decomposition of the conceptualization activity in a number of subactivities is described by the process model in figure 8.2.

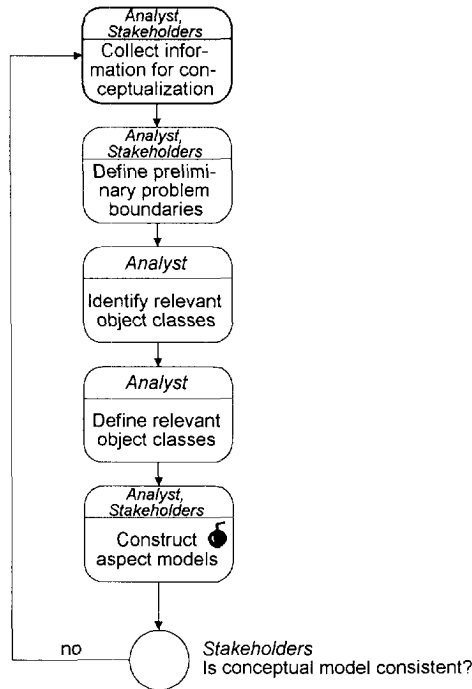


figure 8.2 Process model for 'Conceptualize problem situation'.

The conceptualization of a problem situation starts with the *collection of information*. The analyst observes the organization in order to create a 'rich picture' of what is going on. For this purpose, the analyst may employ various information gathering techniques such as (semi) structured interviews or participative observation. Information can also be collected from other sources, such as internal or external documents and literature studies. Finally, a group of stakeholders may be actively involved in the gathering activity, for example during a GSS session.

Next, the *preliminary boundaries* of the problem situation have to be *defined*. Often, not all observed information is relevant for a description of structure of the problem situation. Also, the problem situation may have to be limited along several axes to be able to handle it in a reasonable amount of time. Examples of axes are: part of the organization, level of detail, processes, organizational items, employees in the organization, and part of the time dimension. The members of the organization may be involved in this activity by indicating which parts of their problem situation they

feel is most relevant or which part they want the remainder of the process to focus on first.

When the preliminary boundaries are set, the analyst can start with the *identification of the relevant object classes* in the problem situation. This identification represents a further specification of the problem boundaries [Sol 1982]. For this activity, the analyst uses the object-attribute-action worldview as a pair of corrective glasses with which the collected information is observed. The object classes are expressed in terms of the modelling concepts we introduced in Chapter 3: actors, repositories, links, messages, products, and persons. Next, the object classes are *defined* in more detail by describing their attributes and actions. It may be useful to actively involve organizational members in these activities. However, we found that thinking in terms of objects, attributes, and actions, is not a straightforward exercise for people that are not used to it. It may, therefore, be wiser to have the object classes identified and defined by an (expert) analyst.

The previous activities actually result in a conceptual model. However, experience has shown that just a set of object definitions does not easily provide real insight to stakeholders and analysts with respect to the interactions and interdependencies between the individual objects. To this end, three *aspect models* can be *constructed* that focus on specific characteristics of the conceptual model. For a detailed discussion of the purpose and composition of these models, we refer to Sections 3.4.2 to 3.4.4. The decomposition of the activity 'construct aspect models' is depicted in figure 8.3.

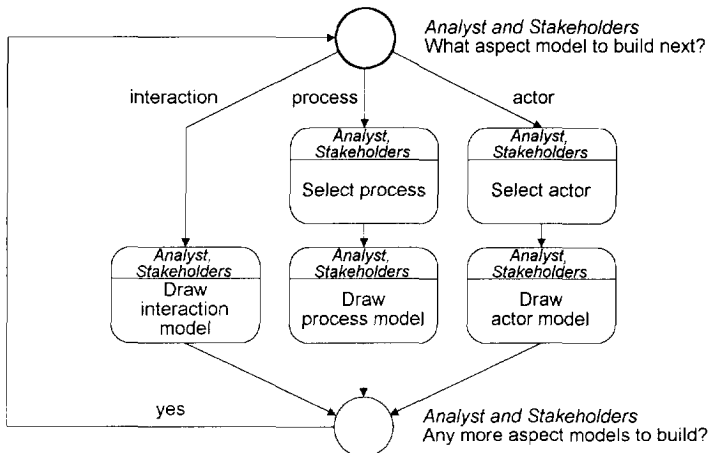


figure 8.3 Process model for 'Construct aspect models'.

The construction of the aspect models starts with a decision about *which aspect model has to be built*. This decision is made by the analyst, if necessary in consultation with

the stakeholders. The construction of the *interaction model* is straightforward. From the definitions of actors, repositories, and links, the items that are exchanged between the item processors can be identified. These elements can be directly depicted graphically with the interaction model icons.

The construction of a *process model* starts with the *selection* of the *process* that will be modelled. Sometimes it is difficult to identify the actual processes in an organization. We propose two methods for identifying processes. Both methods require observations and interviews with actors that take part in the actual situation. The *first method* is to evaluate the flow of one item. Often, a process can be seen as the routing and transformations an item undergoes from the moment it enters (a part of) the organization to the moment it exits the organization. In fact, the modeller 'sits' on an item, watching the tasks and decisions that are carried out during the item's flow [Streng 1993]. The *second method* is to watch the actions of each actor. The tasks and decisions that contribute to a process should be connected to those of other actors. This way, a chain of actions will appear. This chain will either end or cross a boundary of the problem situation. After identifying the constituent tasks and decisions of the chosen process, a graphical representation of the process model can be *drawn* using the appropriate icons.

The construction of an *actor model* resembles that of the process model. First, an *actor* is *selected*. Then, the actor model is *drawn* after identifying the interdependencies between the actor's tasks and decision. When the construction of an aspect model is finished, the analyst may decide in cooperation with the stakeholders whether it is necessary to construct *another aspect model*.

When the construction of the aspect models is concluded, the stakeholders involved in the conceptualization activity have to decide whether the conceptual model is *consistent*. This means that they have to indicate whether the model gives a coherent representation of the structure of their problem situation. If this is not the case, each of the previous activities may have to be carried out again. Otherwise, the conceptualization activity is concluded.

At this point, we want to stress that the conceptualization activity is *not* carried out in the linear fashion in which it is depicted and described above. Actually, the process model in figure 8.2 allows the analyst and stakeholders to follow *various modelling strategies*. The analyst may, for example, decide to construct the whole conceptual model himself and then present it to the stakeholders. He may also decide to involve the stakeholders in the modelling process from the beginning and build the model in a true cooperative fashion. Finally, a 'prototyping' strategy may be followed as well. The analyst constructs a first rough version of some aspect models with one or two

stakeholders. The resulting models are subsequently evaluated and modified in a group session in which various other stakeholders participate.

To conclude this section, we summarize the way in which GSS may be used to support the joint execution of the conceptualization activity. In table 8.1, we identify the actors and nature of each of the constituent activities. We propose which modules of GroupSystems V are most appropriate to support these activities. Furthermore, we indicated GroupSystems V's support for the activities 'identify' and 'define relevant object classes' in which the stakeholders are not likely to participate. It may be the case that there is a *group* of analysts that execute these activities.

Conceptualization activity	Actors	Nature	Group support
Gather conceptual data	Analyst,Stakeholders	Generate	TC
Define problem boundaries	Analyst,Stakeholders	Generate + Organize	TC,IO
Identify relevant object classes	Analyst	Generate + Organize	IO
Define relevant object classes	Analyst	Generate + Organize	IO
Construct aspect models	Analyst,Stakeholders	Generate + Organize	TeamGraphics
Is conceptual model consistent?	Stakeholders	Evaluate	Vote

table 8.1 Group support for the conceptualization activity.

As can be seen in table 8.1, we propose two different modules for generating information: Topic Commenter (TC) and Idea Organization (IO). We feel that Topic Commenter is more appropriate for free format, unrestricted brainstorming. Idea Organization is more practical to identify issues that can be summarized in one short statement or to organize information that was generated with Topic Commenter. This also holds for the other problem solving activities that we discuss in the remainder of this chapter.

The activity 'construct aspects model' can be carried out in close cooperation with a group of stakeholders using group support technology such as TeamGraphics, a group drawing GSS [Ventana Corporation 1993c]. TeamGraphics supports groups working in a joint drawing space using various predefined icons. In addition, TeamGraphics supports the construction of 'templates'. A template controls the features of TeamGraphics that are available to the group members, such as menu commands, default settings, and the names, colors, and availability of drawing icons. Hence, dedicated templates can be designed for each of the aspect models, thereby reducing the complexity of the modelling tool for the group members. An example of a template in TeamGraphics for drawing interaction models is given in figure 8.4. Note that only circles, triangles, (poly) lines, and text are available as drawing icons, and that they can be selected by clicking on the name of the concept they represent in the 'shapes' menu.

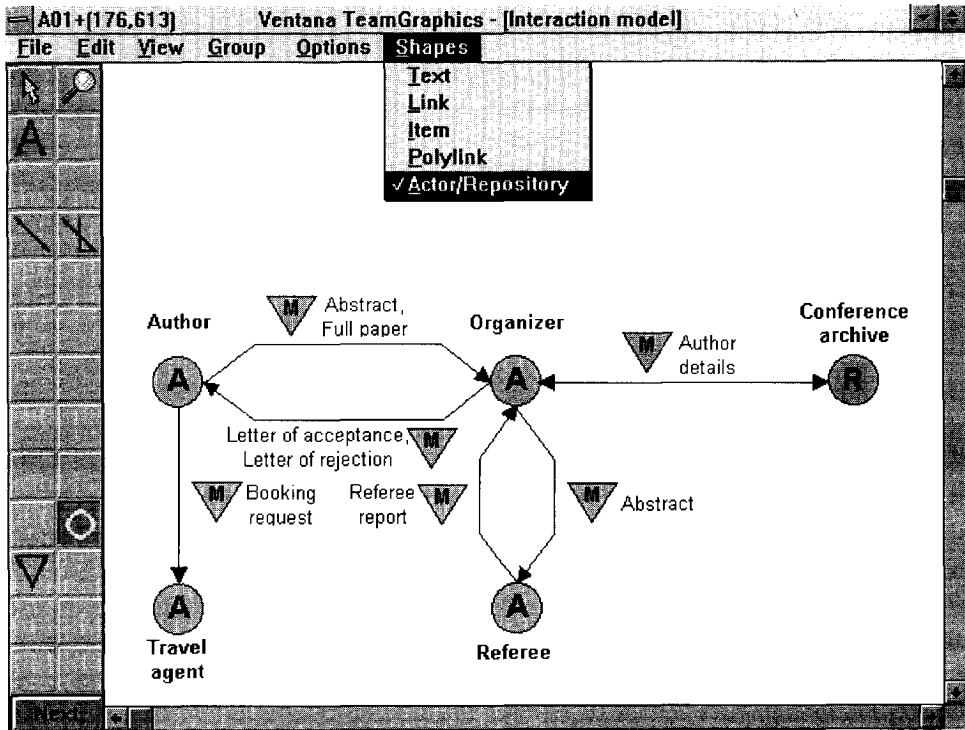


figure 8.4 Screenshot of TeamGraphics using an interaction model template.

8.3 Construct empirical model

The second main activity in the process is the construction of an empirical model of the problem situation. The goal of this activity is to further specify the conceptual model on such a level of abstraction, that the resulting empirical model can be used to analyze and diagnose the problem situation. The result of this activity consists of a verified and validated empirical model. We adopt a combined model construction language for the construction of the empirical model as this model may include different parts: organizational requirements, problem descriptions, empirical data, an experimental design, and simulation code. To express the first four parts we use natural language (restricted to describing the concepts identified in the conceptual model), for the other part we use a simulation language. The decomposition of the activity 'construct empirical model' is represented by the process model in figure 8.5.

The construction of the empirical model starts with the *formulation of organizational requirements*. A group of stakeholders together with the analyst have to determine which are the requirements that the organization must fulfill in order to execute its processes adequately and perform its function in its environment. This activity will

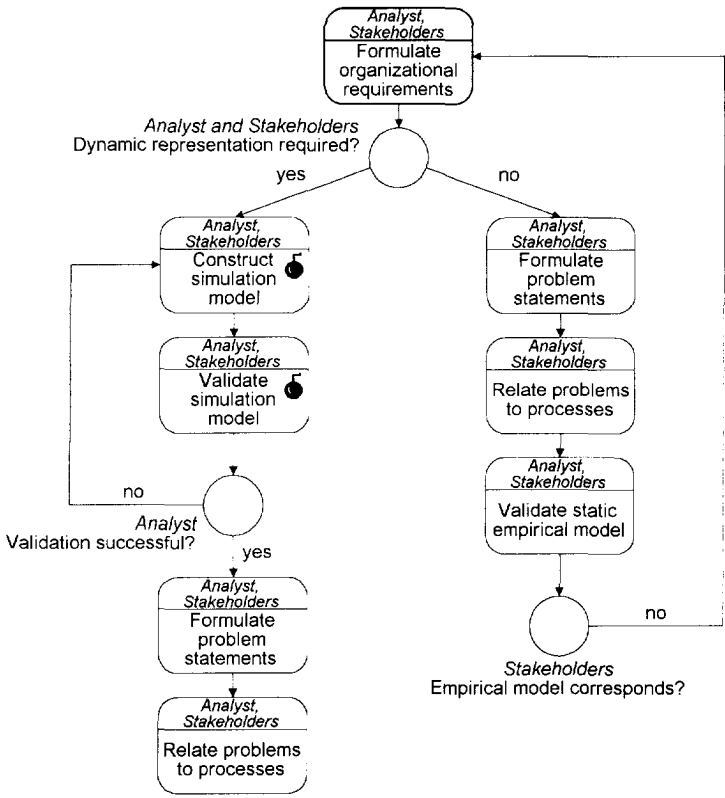


figure 8.5 Process model for 'Construct empirical model'.

most likely follow one or more 'generation-organization-evaluation' cycles before an acceptable set of requirements is established. The group processes during this activity can be supported with techniques that stimulate group creativity and idea structuring, such as NGT, ISM, or Idea Writing to name a few, see also [Moore 1994]. It is paramount that this activity is carried out by a group of stakeholders, for example in a GSS session. Later on in the process, the results of this activity may help to convince those involved that certain organizational changes to their situation are unavoidable.

Next, the analyst and the stakeholders have to decide whether it is *necessary* to build a *dynamic representation* of the problem situation. A reason to do so is, for example, that the interaction between actors and the processes they participate in, is too complex to thoroughly understand from a (static) interaction, process, and actor models. A reason to abstain from building a dynamic model is that such a model is already available. If it is decided that a dynamic representation of the problem situation is required, then the analyst *constructs a simulation model* in cooperation with the stakeholders. The decomposition of this activity is given in figure 8.6.

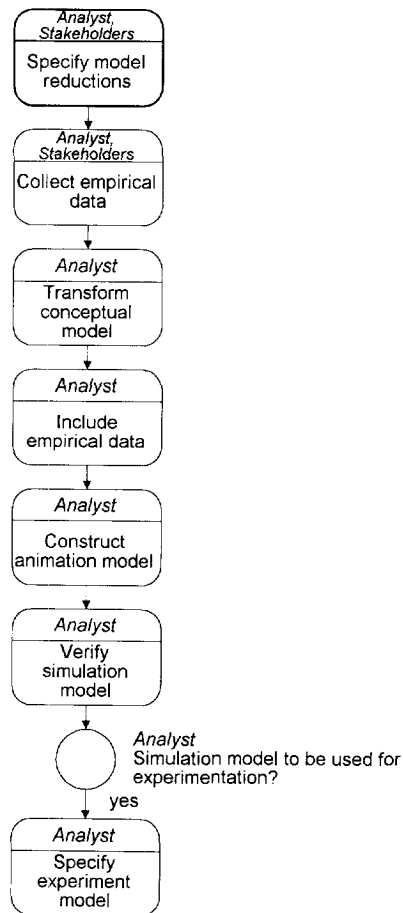


figure 8.6 Process model for 'Construct simulation model'.

The first activity for the construction of a simulation model is to *specify a number of model reductions*. Parts of the conceptual model may either be aggregated or not be included in the empirical model, for example, because they are expected to have a small impact on the model output. The list of organizational requirements may be used to select parts of the conceptual model that need not be included. For example, after consultation with the stakeholders, the analyst may decide to exclude certain objects and processes from the remainder of the process, because they do not contribute to any of the major organizational requirements. Objects' attribute values may be represented by aggregated ranges of values or stochastic distributions taking the availability of empirical data into account [Sol 1982].

After the specification of model reductions, the analyst *collects empirical data* in cooperation with members of the organization. This activity is directed by the

definition of the conceptual model's object classes that will be included in the empirical model. To collect empirical data, the analyst may carry out (semi) structured interviews, handout questionnaires, obtain expert estimations, (statistically) analyze administrative sources and organizational databases, or take a set of measurements himself. Organizational members can also be invited for a GSS session in which they have to answer structured questionnaires or give a set of expert estimations that they all agree on.

Then, the analyst carries out the actual *transformation* of the *conceptual model* into simulation code. If SIMAN [Pegden et al. 1990] is used as the model construction language, then the analyst can use the transformation procedure that we described in Section 3.4.5. After the transformation, the simulation code *together* with gathered empirical data results in an empirical model embodying an executable *simulation model* of the problem situation.

The next activity concerns the *construction of an animation model* that depicts the dynamic behavior of the simulation model. The animation model will facilitate the communication of the simulation model to members of the organization and hence stimulate their involvement in the process. The analyst first decides which animation screens or views on the problem situation, for example micro, meso, and macro, he expects to be most effective in supporting the communication of the dynamic characteristics of the problem situation to the stakeholders. Next, the static background of the animation model has to be designed. The background represents, for example, the layout of the organization. The analyst subsequently defines the icons that represent the item processors (actors and repositories) and items that are included in the empirical model. Finally, he defines the routes that items follow on the screen when they are sent from one item processor to the other.

When the construction of the animation model is completed, the analyst has to *verify the simulation model*. This means that the correctness of input data generation, the model structure, and the calculation of the model output have to be checked [Meel 1994; Sol 1982]. To this end, the analyst may employ various techniques such as 'structured walk throughs', 'module and integration testing', and 'tracing' [Meel 1994]. The animation model may support this activity by clearly visualizing in which part of the simulation model modelling errors were made or unexpected behavior occurs [Vreede and Verbraeck 1996].

The last activity with respect to the construction of a simulation model, is the *specification of the experiment model*. The analyst defines a set of treatments, including an initial treatment to which the model is exposed. An experiment model does *not* need to be specified if the analyst only wants to use the simulation model to depict the dynamic characteristics of the problem situation in a way that is easy to understand for

the stakeholders. In this case, it is also unlikely that the simulation model has the level of statistical correspondence that is required for meaningful experimentation.

Turning back to the main process model for the construction of an empirical model (figure 8.5), we see that the next activity concerns the *validation* of the simulation model. The decomposition of the validation activity is depicted in figure 8.7.

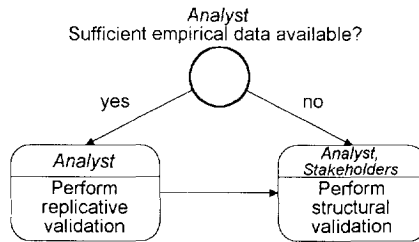


figure 8.7 Process model for 'Validate simulation model'.

If there is sufficient empirical data available, then the analyst performs a *replicative validation*. During this activity, the output data of the empirical model is compared to corresponding sample data from the real problem situation. This is preferably done by statistical testing, but techniques for stability testing may also be employed [Meel 1994; Sol 1982].

Next, the analyst performs a *structural validation* of the empirical model. The purpose of this activity is to check whether the model behaves as would be expected. This can be done by giving the model impulses, while presenting the animation model to a group of stakeholders, and subsequently discussing the model behavior and outcomes, see also [Meel 1994; Wierda 1991; Vreede and Verbraeck 1996]. The structural validation activity can further be supported by using semi-structured questionnaires in which the stakeholders are asked to critically assess the behavior of the model and certain reductions that were made during its construction [Meel 1994].

If the validation of the simulation model is *not* considered to be successful, meaning that significant inconsistencies have been found, then the model has to be modified. Hence, one or more of the construction activities have to be carried out again. If the validation is considered successful, then the analyst and stakeholders proceed to the next activity (see figure 8.5). This activity is also carried out if no dynamic representation of the problem situation was required.

In this next activity, '*formulate problem statements*', the analyst and stakeholders identify a number of problems that they think frustrate the attainment of the

organizational requirements. It is not unlikely that the analyst may already have compiled a list of problem ideas during the collection of conceptual data and the construction of the conceptual model. Together with a dynamic empirical model (if constructed), this list of problem ideas can be used as a kickoff list in a group session to stimulate the stakeholders in identifying problems. The formulation of a list of problem statements may be carried out in a similar way and with similar techniques as with the identification of the organization requirements. It has to be stressed that at this point, the problem statements formulated by the analyst and stakeholders merely represent *problem assumptions*. Experiments with the empirical model have to be carried in order to define exact statements of problems and their causes. This is done during the *diagnosis* activity, which will be discussed in the next section.

After the compilation of a list of problems statements, the problems have to be *related to the processes* in the organization. The purpose of this activity is to identify which of the organization's processes, as represented in the conceptual model, require the most attention. This activity should be carried out together with the stakeholders. It may not only be more efficient to perform this activity in a group setting. It may also create extra awareness for the problems that occur in the processes of the participants themselves of their colleagues'. Problems should be related to processes or the processes' constituent tasks and decisions on the lowest possible level of abstraction in order to consciously direct the diagnosis of the problem situation. This means that, for example, a problem with respect to the 'performance of desk research' (see figure 7.4), is preferably related to one of the constituent tasks and decisions of this task (as depicted in figure 7.5).

Finally, if no validated simulation model was constructed, then the static elements of the empirical model, the list of organizational requirements and of problem statements, have to be *validated*. To this end, the analyst discusses the list of organizational requirements and problems with a group of stakeholders other than the ones that were involved in the compilation of these lists. Obviously, the previous involvement of stakeholders guarantees a certain level of validity. During the discussion, the stakeholders give feedback with respect to whether they feel that the list of requirements and problems gives an accurate account of the current organizational situation. This feedback is, for example, collected during individual interviews or in a GSS session.

Based on the results of the previous validation activity, the stakeholders are either convinced that the empirical model sufficiently *corresponds* to their current situation or not. If not, then one or more of the previous activities have to be carried out again in order to make the necessary modifications in the model. If the model is accepted as a correct representation, then it can be used to diagnose the problem situation.

With respect to the order in which the activities for the construction of an empirical model are carried out, we want to make similar comments as for the construction of the conceptual model. The construction of an empirical model is seldomly performed in a linear fashion. In this respect, we observe that the process models presented in this section represents different modelling strategies that the analyst and stakeholders can follow. Of course, the analyst may choose to employ a linear approach and try to complete every modelling activity before starting the next. But he may also follow a more incremental strategy, constructing initial lists of organizational requirements, a draft version of the simulation model, an initial list of problems, and engaging in a number of 'validation-modification-extension' cycles. The same goes for the involvement of the stakeholders, which can be substantial from the beginning or be built up slowly as the process proceeds.

The way in which GSS may be employed to support the active involvement of stakeholders during the various modelling activities is summarized in table 8.2. Again, based on our experience, we propose which modules of GroupSystems V should be most suitable to support a particular activity. Note that we only include activities in table 8.2 that, we think, will benefit from GSS support.

Construction of empirical model	Actors	Nature	Group support
Formulate organiz. requirements	Analyst, Stakeholders	Gen. + Org. + Eval.	TC, IO, Vote
Dynamic representation required?	Analyst, Stakeholders	Evaluate	Vote
Construct dynamic empirical model:			
• Specify model reductions	Analyst, Stakeholders	Generate	TC, IO
• Gather empirical data	Analyst, Stakeholders	Generate	TC, IO, QE, SV
Perform structural validation	Analyst, Stakeholders	Generate + Evaluate	IO, QE, SV
Formulate problem statements	Analyst, Stakeholders	Gen. + Org. + Eval.	TC, IO, Vote
Relate problems to processes	Analyst, Stakeholders	Organize	GO, GM
Validate static empirical model	Analyst, Stakeholders	Generate + Evaluate	TC, IO, Vote
Empirical model corresponds?	Stakeholders	Evaluate	Vote, MM

table 8.2 *Group support for the construction of the empirical model.*

With respect to table 8.2, we make four comments. First, we point out that the formulation of organizational requirements and initial problem statements consists of the generation, organization, and evaluation of ideas. Therefore, both information handling (TC, IO) and evaluation (Vote) modules have to be used for this activity. Second, Questionnaire (QE) and Survey (SV) are, as their name suggests, two modules that support a group of participants in filling out predefined forms that include free text formats, but also various types of voting techniques. Third, in order to relate problems to processes, we propose either one of two modules, Group Outliner (GO) or Group Matrix (GM). *Group Outliner* enables session participants to build a hierarchical tree of items, in this case processes and constituent activities. Problems can then be associated to a particular part of the hierarchy in the form of 'comments'. *Group*

Matrix presents the participants with a matrix in which the cells may represent a numerical or qualitative relation between the row item and the column item. Processes and problems can be associated with GM by allowing the participants to place an 'x' in a cell if a problem refers to a particular process or activity in that process. Finally, *Mood Meter* (MM) is a communication module with which the opinion of a group can be polled in a quick and very informal way.

8.4 Diagnose problem situation

When the empirical model is considered to be a valid representation of the problem situation, the analyst can use the model as a 'substitute' in order to diagnose the problem situation. The goal of the diagnosis activity is to identify causes underlying the problem situation and to gain insight in the relations between cause and effect. The result of the diagnosis activity is a set of solution directions that will be used to guide the identification and exploration of possible solutions.

It should be noted that there is a strong *reciprocal interdependency* between the construction of an empirical model and the diagnosis of a problem situation. During the former activity, usually ideas for problem causes emerge, for example during validating the simulation model. During the latter activity, indications for other problems or causes present themselves, which, in order to be further examined, require modifications in the empirical model. Therefore, the linear fashion in which we present the two main activities should only be considered to facilitate our discussion of both activities. The process model for the diagnosis activity is given in figure 8.8.

The diagnosis of the problem situation starts with the analyst and stakeholders *formulating* a number of *initial cause hypotheses*. These hypotheses describe the first ideas why according to the group, certain problems may occur. The basis for this activity is laid during the construction of the empirical model, where the organizational requirements and lists of problem statements were compiled and where these problems were related to the organization's processes. Cause hypotheses are expressed in natural language or in terms of the empirical model's input and output variables, if a simulation model was constructed. Hypotheses refer to the way in which the organization executes and coordinates its processes, to external factors or other problems, for example when one problem is believed to be the cause of another one.

It is important to carry out the formulation of initial cause hypotheses together with a group of stakeholders. They have to achieve consensus on a set of cause hypotheses, as this implies that they commit themselves to finding a solution to take the causes away. A way to perform this activity is that the group identifies and prioritizes the *key* problems in the current situation. These are the problems that are believed to frustrate

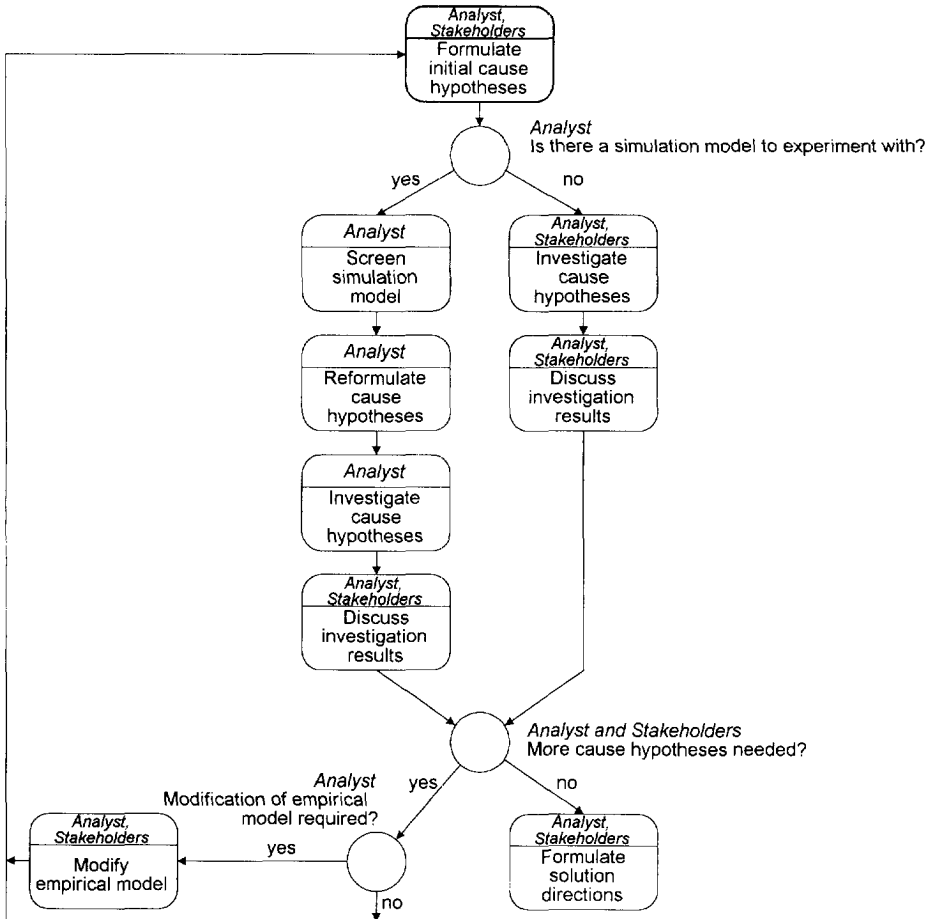


figure 8.8 Process model for 'Diagnose problem situation'.

the execution of the processes that support the attainment of the most important organizational requirements. A group session supported by GroupSystems V can be an effective setting for such an exercise. If a simulation model was built, it can be used to stimulate discussion about the problems and their relative importance.

If the analyst *built* a simulation model that was both replicatively and structurally validated, then it may be used to further examine the cause hypotheses. To this end, the analyst first *screens* the empirical model. This means, that he explores the empirical model for possible relationships between input and output variables. The screening activity is directed by the initial cause hypotheses. However, depending on the results of the screening activity, the analyst may *reformulate* any of the cause hypotheses or add new ones.

Next, the analyst *investigates* the cause hypotheses. If he built a simulation model, then the analyst translates the (reformulated) cause hypotheses into changes in the empirical model and defines an experimental design that maximizes the discriminating power for testing the hypotheses. The results of the experiments enable the analyst to quantitatively assess the impact of the changes he made in the model and thus determine whether or not the cause hypotheses hold. For this activity, the analyst may use tools for sensitivity analysis and statistical analysis. The animation model may also support this activity by depicting how processes proceed over time on different abstraction levels. If there is no simulation model built, then the analyst investigates the cause hypotheses in close cooperation with the stakeholders. To this end, they may employ techniques such as gaming, structured walkthroughs, cognitive mapping, or 'just' (electronic) discussions.

The analyst then *discusses* the investigation results with the stakeholders. At this point, a need to investigate *more cause hypotheses* may emerge, for example because the use of the empirical model and the discussion that follows, yields a greater understanding of the problem situation. If this is the case, the analyst and stakeholders enter a new 'hypotheses formulation-investigation' cycle, possibly after *making a number of modifications to the empirical model* that are required to be able to investigate the new cause hypotheses. For this purpose one or more of the activities described in the previous section may have to be performed again.

However, if the stakeholders and the analyst feel that the cause hypotheses sufficiently explain the identified key problems, or if there is no simulation model to experiment with, then the group *formulates solution directions*. The solution directions consist of two parts. The first part is a set of *solution requirements*, that indicate to what extent causes, effects, and other problem characteristics need to be changed before the problem is considered to be solved. Note that not changing certain characteristics of the problem situation may also be part of the solution requirements. The second part consists of a rough outline of possible *solution characteristics*. For example, are solutions sought in changing the structure of the organization, in adapting the interdependencies between certain activities, or in designing IT that will support the organization's processes? Although the analyst will most likely have more adequate expertise to formulate the solution directions, it is important to nurture the commitment of the stakeholders by actively involving them in this activity.

To conclude this section, we indicated in table 8.3 in which way GSS, and GroupSystems V in particular, can support the diagnosis of the problem situation. As can be seen, we only included activities that, we feel, will benefit from group support.

Diagnosis activity	Actors	Nature	Group support
Formulate initial cause hypotheses	Analyst, Stakeholders	Generate + Organize TC, IO	
Investigate cause hypotheses	Analyst, Stakeholders	Generate + Organize TC, IO	
Discuss investigation results	Analyst, Stakeholders	Generate	TC
More cause hypotheses needed?	Analyst, Stakeholders	Evaluate	Vote
Modify empirical model	Analyst, Stakeholders	see Section 8.3	see Section 8.3
Formulate solution directions	Analyst, Stakeholders	Generate + Organize TC, IO	

table 8.3 Group support for the diagnosis activities.

8.5 Construct models for change

In this section, we discuss the construction of models for change. The goal of this activity is to identify and work out in further detail a number of potential solutions for the key problems. The result of this activity is a number of models for change along with, where possible, both a quantitative and a qualitative evaluation of each. The process model for the construction of models for change is depicted in figure 8.9.

The construction of models for change starts with the *identification of possible solutions* in terms of structural and procedural changes in the way the organization's processes are carried out and coordinated. This activity is guided by the solution directions that were formulated at the end of the diagnosis activity. Although possible solutions will often come to mind during the previous problem solving activities, the identification of solutions is not a straightforward activity. It requires expertise, experience, and creativity. It may also be the case that the organization beforehand indicates a number of solutions that it wants to have investigated. In order to facilitate the solution identification activity, a number of techniques can be employed, such as heuristic searches, model analysis, idealized design, and dialectical debate, see [Meel 1994] for an overview.

It is paramount that stakeholders are involved in this activity. Not only because of the activity's complex nature, but also because the stakeholders are most likely going to be affected by the activity's outcome if one of the solutions is eventually implemented. In this respect one has to bear in mind that stakeholders' involvement may prevent the identification of solutions that seem threatening to their work situation. To deal with this possibility, the group of stakeholders should be composed carefully, and if feasible include some people that are not directly involved in the problem situation. The facilitation of the group process should focus on stimulating creativity and spontaneity, and uninhibited and unrestrained contributions.

After a set of possible solutions is identified, the analyst and stakeholders have to *select* which of these solutions they want to elaborate in more detail. This activity can be guided by intuition, but also by other factors such as the expected costs and benefits

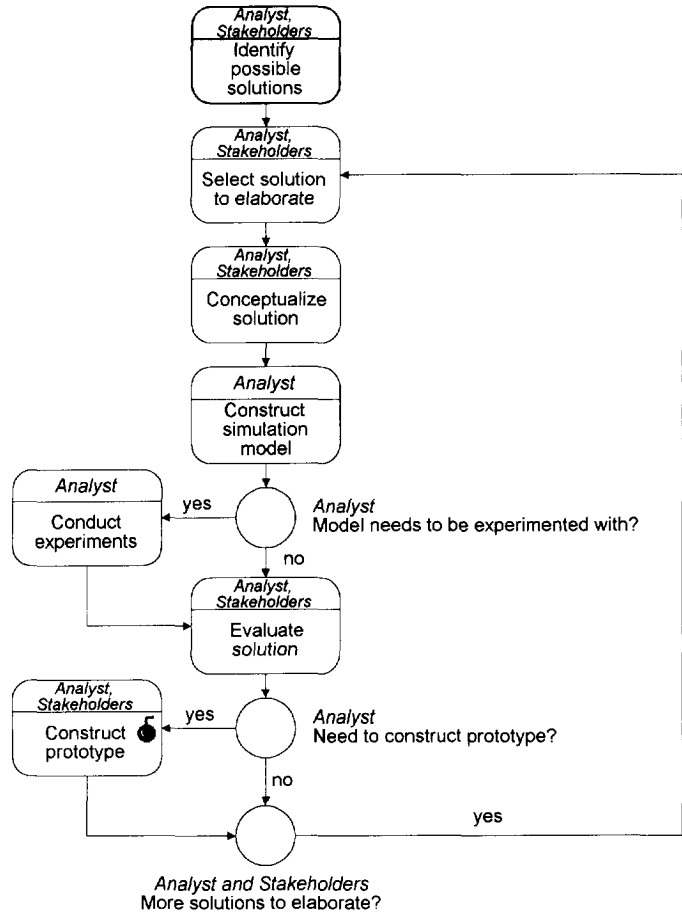


figure 8.9 Process model for 'Construct models for change'.

of a solution, or the expertise of the analyst. In this sense, the selection activity may become a multi criteria decision making (MCDM) activity. The group may decide to use a GSS to structure and execute this activity.

After selecting a solution, the analyst together with the stakeholders first *conceptualizes* the future organizational situation that would come into being if the solution were implemented. This activity can be carried out in the same way and can be supported by the same techniques and group support as we discussed in Section 8.2. However, it is not always necessary to perform an elaborate conceptualization. The solution not necessarily prescribes a complete redesign of the organization's processes and their coordination, but may represent more gradual changes. Therefore, the conceptual model of the current situation may be used as a basis.

Next, the analyst *constructs a simulation model* that represents the future situation after implementing the solution. If the analyst already built a simulation model of the current situation and the solution does not prescribe too drastic changes, then the old model may serve as a basis. Otherwise, the analyst can use the transformation procedure described in Section 3.4.5. It is important that at least a 'logically valid' version of the empirical model is built. Our third case study showed that it is awkward to use, for example, process models to present an organizational solution to members of the organization, especially given the fact that the people evaluating the associated solution are not necessarily the same ones that identified it. A dynamic representation of the new situation is more likely to provide insight about the way in which the processes are executed.

If the constructed empirical model is to be used for *experiments*, then the analyst exposes the model to a treatment. Statistical tests can be used to compare the experiment results with those of the model of the current situation. It also has to be verified whether the solution satisfies the solution requirements.

If the model cannot be used for experiments, or if the experiment results have been analyzed, then the analyst and a group of stakeholders jointly *evaluate* the solution. During this activity the analyst presents the simulation model and, if available, the experiment results. The stakeholders may subsequently enter a discussion on the strengths and weaknesses of the solution. They may also propose extensions to the solution. The evaluation activity can be carried out during a GSS supported session to structure the discussion and stimulate the efficient exchange of ideas and opinions.

If the solution that was evaluated, includes the development of supporting IT for the future organizational situation, then the analyst has to decide whether it is necessary to construct a prototype of this support. The decomposition of the activity '*construct prototype*' is given in figure 8.10.

For the construction of a prototype of the required IT support (an information system), the analyst first *determines the functionality* that has to be included in the prototype. To this end, the analyst first has to look at the process models of the processes in which the information system will be used. Tasks and decisions that exchange information with the system have a repository icon linked to them, representing the system. From the object class definitions of the items that are exchanged and the definitions of the actors' actions, functional requirements of the system can be deduced. Next, the analyst has to determine what functionality will be included in the prototype. In this sense, the analyst makes a *reduction*. He can decide to fully develop a few functions (a vertical prototype), or to implement all functions, but only superficially (a horizontal prototype). The group of stakeholders participates in this activity by identifying what information they require for the execution of the processes

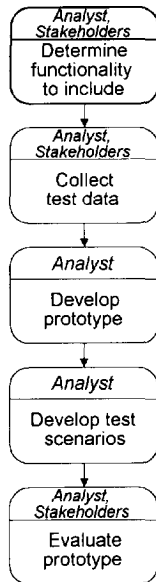


figure 8.10 Process model for 'Construct prototype'.

that the IT solution relates to. To this end, a GSS session can be organized. The analyst has to take the stakeholders' feedback and use it to finetune the functional specifications with.

Next, a set of *test data* has to be *collected*. The purpose of this activity is twofold. First, when a prototype of the system is built, realistic data has to be used to make an evaluation of the prototype worthwhile. Second, by examining the test data, modifications in the functional specifications of the system may turn out to be required. Test data may be collected from existing databases and administrations in the organization. The analyst can also organize a GSS session with a number of stakeholders (see Chapter 7).

After the functional specification and the collection of test data, a *prototype* of the IT solution can be *developed*. For this activity, the analyst has to choose a hardware and software platform. This choice depends on several factors, such as the costs of a particular platform, the availability of hardware and software in the organization, the expertise of the analyst, and the appropriateness of the platform for the realization of the required functionality. If there is a group of analysts involved in this choice activity, they may decide to carry it out as an MCDM activity.

In order to evaluate the prototype, the analyst *develops* a number of *test scenarios*. These scenarios may refer to a single meeting in which the prototype is presented,

used, and evaluated by future users. It may also describe a pilot period in which the prototype is used in (a part of) the organization.

When the scenarios are developed, the *evaluation* of the prototype can start. This evaluation activity is conducted in a group session in which stakeholders participate to test the prototype. The stakeholders give their feedback on different aspects of the prototype, such as its functionality and its user interface. They may also point at required modifications and extensions. As we demonstrated in Chapter 7, this process can be adequately supported with a GSS.

When a solution has been evaluated, possibly including the associated IT support, the analyst and stakeholders have to decide whether or not to *elaborate more solutions*. This can be a new solution, but also an extension or modification of the one that was just evaluated. If there is no need to look into more solutions, the activity 'construct models for change' is completed. Otherwise, the above activities have to be carried out again for another solution. Note that in this sense, a *prototyping strategy* can be applied for the development of solutions by quickly presenting rough outlines of a solution and then iterating toward a solution that is satisfiable to all parties involved.

GSS can be used at several points during the construction of models for change. In table 8.4, we summarize the type of GSS support for the activities that may benefit from it.

Model construction activity	Actors	Nature	Group support
Identify possible solutions	Analyst, Stakeholders	Generate + Organize	EBS, TC, IO
Select solution to elaborate	Analyst, Stakeholders	Evaluate	Vote, AE
Conceptualize solution	Analyst, Stakeholders	Generate + Organize	TeamGraphics
Evaluate solution	Analyst, Stakeholders	Generate + Evaluate	TC, IO, Vote, AE
Construct prototype			
• Determine functionality to include	Analyst, Stakeholders	Generate + Organize	TC, IO
• Collect test data	Analyst, Stakeholders	Generate	TC, IO
• Develop prototype	Analyst	Evaluate	Vote, AE
• Evaluate prototype	Analyst, Stakeholders	Generate + Evaluate	TC, IO, Vote
• More solutions to elaborate?	Analyst, Stakeholders	Evaluate	Vote

table 8.4 Group support for the construction of models for change.

With respect to table 8.4, we want to make three comments. First, Electronic Brainstorming (EBS) is a module of GroupSystems V in which participants enter ideas into separate discussion files that are randomly shared throughout the group. This procedure attempts to stimulate creative and broad idea generation by precluding the group from focussing on one theme. (The reason that we have not suggested the use of EBS for other activities is that the exchange of information with other modules, most notably Vote, is not optimal). Second, Alternative Evaluation (AE) is an MCDM

module that allows the participants to rank a number of alternatives according to a set of criteria. The relative weights of these criteria can also be defined. Finally, the GroupSystems V's support indicated for the activity 'develop prototype' refers to the selection of a hardware and software platform.

8.6 Choose change alternative

The solutions that were explored and evaluated in the previous section, represent the different change alternatives for the organization. In this section, we discuss the activity 'choose change alternative'. The goal of this activity is to select which of the change alternatives is going to be implemented in order to solve the key problems in the organization. Hence, the result of this activity is a description of the (combination of) alternative(s) that is selected. The decomposition of the 'choose change alternative' activity is represented by the process model given in figure 8.11.

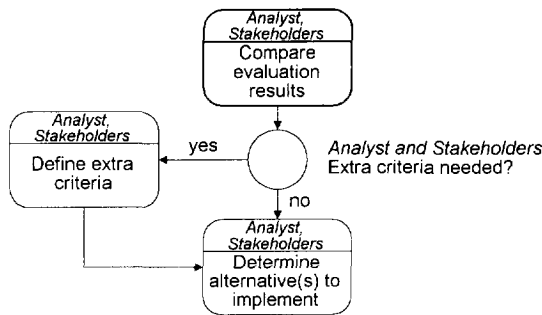


figure 8.11 Process model for 'Choose change alternatives'.

The choice of a change alternative starts with the *comparison* of the *evaluation results*. During the previous activities, a number of solutions, possibly including IT support, were examined and evaluated. At this point, the analyst and stakeholders must find out to what extent these solutions meet the solution requirements that were defined earlier. To support this activity, the group can employ a type of group support that allows them to organize the evaluation results as follows: For each solution that was examined, the evaluation results with respect to each of the solution requirements are first identified and then attributed to this requirement. This way, an overview of all evaluation results emerges that can be used to decide which solution to implement.

Although the above activity may clearly put forward one or more of the proposed solutions as the most favorable, there may be other *criteria* that influence the final choice by the group. Examples of such criteria are political, social, psychological, environmental, economical, financial, and technical constraints. If extra criteria are needed, the analyst has to support the stakeholders to *define* them. To this end, a group

session can be organized in which, apart from the identification of criteria, their relative importance in terms of weights is defined as well. For this latter activity, the group can apply various voting techniques.

Finally, if a set of extra criteria is defined, or if this was not perceived to be necessary, the analyst and stakeholders have to determine which of the alternatives is going to be implemented in the organization. Obviously, this activity is likely to take the shape of an MCDM activity. Note that the alternative chosen can also be the current situation, that is, not change anything. The group choosing among the alternatives can employ different strategies [Ackoff 1978]. The group can try to *optimize* by seeking the best or most effective combination of alternatives. They can also try to *satisfice* by seeking a solution that is good enough. The choice for a particular strategy will depend on the extra criteria, the solution requirements, and their relative weights, that were defined before. For example, if 'cost' is included as a criterion and if it has a high weight, the group is likely to adopt a satisficing strategy if the alternatives best meeting the solution requirements are most expensive.

The GSS support for the activities described in this section is given in table 8.5. As can be seen, the choice activity is highly participative. As a single comment with this overview, we point out that Group Matrix seems to be the only GroupSystems V module that supports the comparison of evaluation results the way we described them.

Model construction activity	Actors	Nature	Group support
Compare evaluation results	Analyst, Stakeholders	Organize	GM
Define extra criteria	Analyst, Stakeholders	Gen. + Org. + Eval.	IO, Vote
Determine alternative(s) to implement	Analyst, Stakeholders	Evaluate	Vote, AE

table 8.5 Group support for the choice of change alternatives.

8.7 Implement change alternative

In the previous activities, a model of a possible future situation has been created and selected by the organization for implementation. This model just indicates that the selected alternative might eliminate the problem or its major causes and/or effects. In most cases, the model does not prescribe *how* the change process should be structured; a translation from the ideas in the model to an actual design, construction, and implementation is still needed. In this section, we describe the implementation of the chosen alternative. The goal of this activity is to effectively transform the current situation in the organization into the situation as prescribed by the model of the chosen solution. The result is a changed organization. The decomposition of the implementation activity is depicted by the process model in figure 8.12.

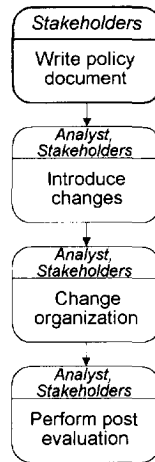


figure 8.12 Process model for 'Implement change alternative'.

To prepare for the implementation of the chosen change alternative(s), the stakeholders write a *policy document*. This document addresses the following three issues:

1. It gives a summary of the key problems for which the change alternative represents a solution.
2. It summarizes the consequences of the alternatives in terms of changes in the work situation of the actors involved. It clearly describes the actors' new job contents.
3. It formulates an implementation strategy, that explains the transition path for the migration of the current situation to the new situation. There are various implementation strategies possible, such as implementing all changes at once ('big bang'), performing a phased introduction, or performing an incremental evolution [Eason 1988; Meel 1994].

It is vital that the policy document is authored by members of the organization. This will stress the fact that the organization created and choose the alternative itself instead of having it imposed on itself by an external analyst. Writing a policy document can be carried out with group support that focuses on the joint creation of documents, such as CSCW technology [Ellis et al. 1991].

The next activity after the creation of the policy document is to *introduce* the solution to all people in the organization that will be affected by it. Given the participation of various stakeholders in the problem solving activities so far, it is unlikely that the people in the organization are unaware of the imminent changes. Nevertheless, it is important to introduce the changes properly in order to create a 'critical mass' for the forthcoming change effort [Herzog 1991]. This activity may actually take the shape of a group session in which the simulation model is used to explain the changing situation

to the participants. A GSS can be used to collect the participants' reactions like questions, observations, and recommendations. The information that is generated during this activity may result in some final modifications in the model of the change alternative.

In case *IT support* for the solution has to be implemented as well, the analyst has to introduce the system to its future users. This introduction comprises of three subsequent activities:

1. If a simulation model is available from the situation in which the system will be used, then it is shown to the users. This will explain the role of the system in their work situation.
2. The information system is presented to the users. If possible, the users get a chance to use the system and ask questions.
3. The analyst has to write user manuals for the system and distribute these among the users.

For the introduction of the system, a GSS session can be organized to collect feedback from the users. The results of this session can be used to make some final modifications in the system.

After the forthcoming changes have been introduced, the actual change effort can be carried out according to the implementation strategy described in the policy document. In order to support the change process and maintain the momentum, the organization may decide to appoint a number of *change agents*. This is, for example, one person per process that is changed (a 'champion'). This person is responsible for the daily management of the change efforts with respect to his or her process. Another possibility is to install a change team, that guides the change process and offers support in the organization where necessary [Duck 1993]. Obvious candidates to become change agents are some of the stakeholders that participated in the previous problem solving activities.

The change effort may also involve the installation of the developed IT support. Apart from the overall implementation strategy described in the policy document, there are some technical issues that may require specific attention. These issues include the possible conversion of old data files, the backup of old systems in case the new one causes unforeseen problems, and the question whether and if so, to what extent the old systems will remain in operation during the first period of the new system's operation. In addition, the analyst has to set up a '*helpdesk*' that is able to deal with unforeseen system problems. The helpdesk should also be continuously available to answer questions that the users may have.

When the solution has been implemented in the organization for a significant period of time, the analyst can perform a *post evaluation*. During this activity, he investigates together with some stakeholders whether the key problems indeed no longer exist. To this end, he may gather empirical data to compare with the solution requirements and the predicted data from the empirical model, if available. He may also organize a GSS session, in which he questions a group of stakeholders about their perception of the current situation compared to the old one. If an IT support was implemented as well, this can be evaluated using various types of evaluation methods, see [Vreede 1993c] for an overview. Some of these methods can benefit from using a GSS, especially when the method prescribes the use of questionnaires and (semi) structured interviews. The same type of information gathering techniques can be used to evaluate an organizational solution.

The potential GSS support for the implementation activities is summarized in table 8.6. *Group Writer* (GW) is a module that lets participants jointly write documents. Apart from outlining documents, it supports participants bringing in and working with pieces of information from other sources, for example from a *Topic Commenter* report. Finally, we point out that due to the multitude of potential methods and techniques to support the post evaluation activity, there is a considerable number of GroupSystems V modules that can be employed as support.

Implementation activity	Actors	Nature	Group support
Write policy document	Stakeholders	Generate + Organize	TC, GW
Introduce changes	Analyst, Stakeholders	Generate	TC
Perform post evaluation	Analyst, Stakeholders	Generate + Evaluate	TC, IO, QE, SV, AE, GM, Vote

table 8.6 *Group support for the implementation of change alternatives.*

8.8 Concluding remarks

In this chapter, we presented a design approach for organizational change that represents the dynamic modelling approach, enriched with participative design techniques and support. It employs simulation techniques to quantitatively analyze and evaluate an organization's current and future situation. It uses animated representations of organizational processes to create insight for the organizational stakeholders in their current and future work situation. The availability of statistically representative empirical data is not decisive for the construction of empirical models with which a problem situation can be diagnosed. The use of group sessions and GSS is included to support the efficient and effective collection of modelling information, and to facilitate and structure the involvement of stakeholders in the process. The design approach simultaneously addresses the design of organizational processes, their coordination, and supporting IT. In literature on organization design and on information system design,

such integrated approaches are rare. Exceptions are Checkland's Soft Systems Methodology [Checkland 1981] and the socio-technical design methodology ETHICS [Mumford 1983].

In this final section, we make a number of concluding remarks and observations with respect to the design approach that we presented. First, we introduced and described the activities in the approach in a *linear* fashion. In reality, the linear application of the approach is very unlikely. Certain modelling activities will be carried out in parallel and on different levels of abstraction, while going back and forth. For example, during the construction of the conceptual model, the analyst may start by building a rough version of the interaction model, deduce a first outline of the object class definitions, subsequently build some process models, and then finalize the object class definitions. Also, during the construction of the empirical model, the analyst and stakeholders may continuously evaluate the modelling outcomes so far and may make reduction decisions accordingly. Finally, during the process of choosing between the models for change, new questions and requirements may emerge for additional diagnosing efforts or for new change solutions that were not considered before. In other words, the description of the design approach should *not* be read as a *cookbook*, but should be seen as a set of design activities that are carried out in a flexible way that is determined by the nature of the problem situation and the style of the analyst.

Second, the presented approach is not *prescriptive* in the sense that all problem solving activities *must* be carried out. In the first place, the design of organizations and information systems is such a complex, ill-structured problem, that a universal design approach is very unlikely to exist [Checkland 1981; Sol 1982]. In the second place, the exact application of the approach depends on the particular situation in which it is applied. In some situations, the problems in the current situation are clear enough to start with the identification of solutions. In other situations, the organization itself may have identified a number of change alternatives beforehand, and just wants to compare these to the current situation in order to select the most appropriate one. Stated differently, our approach consists of a problem solving framework that includes various activities, methods, techniques, and types of group support. This framework is *conceptual*. During the application of the approach, the analyst and stakeholders have to *specify* which of the elements in the framework they want to apply in a particular situation. The way in which the approach is applied is, therefore, a *reduction* of the approach presented in this chapter.

Third, the design approach describes the construction of conceptual models and parts of the empirical models as true participative efforts. Organizational stakeholders are not only involved passively in the sense that they merely provide the information with which the analyst constructs a model. The stakeholders are *actively* involved in the construction of conceptual aspect models and the lists of organizational requirements

and problem statements. Our design approach prescribes this active involvement because the joint development of models helps and forces stakeholders to reflect on the way they work and on the way they think about the overall work situation in their organization. Their involvement teaches them to see problems in their work situation and buy in into changes [Argyris 1994; Senge and Sterman 1992]. In other words, participative modelling helps to create a 'platform for change' [Akkermans et al. 1993; Beer 1973]. Participative modelling is found to stimulate debate between stakeholders with different views and to force them to see their own position in the overall situation [Geurts et al. 1993]. This leads to the creation of awareness and insight into a problem situation, to a shared vision among the people involved, and to commitment with respect to change initiatives [Akkermans et al. 1993; Geurts et al. 1993; Vennix et al. 1993]. Finally, as the conceptual and empirical models of a current problem situation are the basis for change alternatives, commitment to and acceptance of these models is required [Akkermans et al. 1993].

Fourth, the proposed application of GroupSystems V modules to support the various activities is based on our experiences during the third case study, on our experiences during various other projects involving GroupSystems V, and on our personal preferences. However, GroupSystems V is flexible in that most modules can be used to support diverse activities. Hence, the proposed application of GroupSystems is *not excluding* other ways of applying this GSS. The exact application will also depend on the particular problem situation and on the personal preferences and experiences of the analyst and participating stakeholders.

Fifth, the proposed application of GroupSystems V modules only indicated *which* modules to use for certain activities. *How* these modules are used is not addressed in great detail. This depends on the exact *design* of the session in which GroupSystems V is employed. A session design consists of a number of *questions or assignments* for the participants, of a *procedure* in terms of a number of steps to handle the information that is produced, and of a *support description* in term of which part of a GroupSystems V module is used for which steps, for example which brainstorming options (Idea Organization) or which voting techniques (Vote). It also has to be determined for which steps the application of GroupSystems V may actually be harmful for the group process, see Chapter 6. We did not include such a detailed description of the application of GroupSystems V modules. This would either be prescriptive and, therefore, too restrictive, or it would be too unique as each situation may require a different session design.

Finally, our approach looks at organization design in terms of an organization's structure, its processes, and the way in which the processes and their constituent activities are coordinated. However, there are many *other factors* and issues that may need to be considered during an organization design process, such as an organization's

culture and atmosphere, the quality of labor, the working conditions, rewarding systems, or the personnel management. Although explicit attention to such factors is outside the scope of the approach, the analyst and stakeholders can take these factors to a certain extent into consideration during the definition of solutions requirements and selection criteria.

9 Case 4: The Criminal Investigations Department Revisited

'For organized crime, so it would seem, is certainly puzzle-solving par excellence.'

- Paul Feyerabend

9.1 Introduction

Our fourth and final case study was, like the third case study, carried out at the Criminal Investigations Department (CID) of the Amsterdam Municipal Police Force. While the previous study focused on the CID's departments and processes involved in the *preparation* of projects, this case study focused on the parts of the CID involved in the *execution* of projects. The goal of this case study was to apply and evaluate the design approach that we described in the previous chapter.

In this chapter we start with a brief description of the problem situation at the CID. Then, we subsequently discuss the conceptual model (Section 9.3), the empirical model (Section 9.4), the problem diagnosis (Section 9.5), and the alternatives for change (Section 9.6). The chapter is concluded with a reflection on this case study.

9.2 Problem situation

In this section, we briefly describe the CID's environment, its organization, and the assignment of the case study. We only address relevant developments and recent issues that emerged during the period in which the third case study was carried out. For a more elaborate description of the CID's environment and organization, we refer to Section 7.2 and [Meel et al. 1994; Meel 1994].

9.2.1 The environment

Compared to the situation a few years earlier, organized crime has grown to become a more complex problem to handle. Criminal organizations are increasingly operating

on a global scale [Welten and Koningh 1994]. This tendency has been fueled by a number of political developments, such as the opening of the European borders and the joint fall of the Berlin Wall and East-European communism [Newsweek 1993]. Its global character forces governments to closely cooperate to combat organized crime, see for example [Council of Europe 1994]. On the technological site, the expanding network of telecommunication services has created vast opportunities for criminals to engage in 'joint ventures' with foreign partners, thereby creating international chains of criminal activity [Newsweek 1993]. Also, the possibilities of electronic data transfer allow criminal organizations to exchange and transport large sums of money back and forth over international borders.

The Dutch government have realized the severity of these developments and placed the combat against organized crime high on the political agenda. Dedicated National Organized Crime Task Forces were installed to handle large criminal networks. Various regulations and measurements were issued to help the police and other judicial bodies to combat organized crime more effectively [Middelburg and Es 1994]. Examples of such combatting tactics include the use of special microphones and preventive break-ins (so-called 'look-inside operations'). Furthermore, various new laws were passed. Bank employees, for example, are now required to report any suspicious transaction above f 25,000,- [Elsevier 1994]. Criminal capital and property can now be confiscated if it is proven to originate from criminal activities (the so-called 'Pluck'em' law). New laws with respect to computer crime facilitate the way in which the police is dealing with, for example, (organized) software piracy [Tactical Department 1993].

Currently, other CIDs throughout the Netherlands are employing strategies that resemble that of the CID in Amsterdam. Instead of focusing their efforts on criminal organizations (involved in drug related crimes), there is more attention now for societal trends that create a fertile breeding ground for organized crime. This strategy is frequently referred to as the 'phenomena approach' [Kortekaas 1994]. Furthermore, it is increasingly suggested to employ scientifically educated people to assist in the execution of this phenomena approach [Eshof and Vegter 1994; Kortekaas 1994; Tactical Department 1993; Welten and Koningh 1994]. The function of these people would not only be to give advice with respect to certain societal trends that they have expertise about, but also to actively participate in projects, for example as data analysts.

9.2.2 The organization

The organizational structure of the CID changed in two respects during the period in which the previous study was carried out, see figure 7.1. First, the number of units changed. At the beginning of this new CID study, the number of units had doubled to

eight. During the study, two units were merged into a new National Organized Crime Task Force, leaving the CID with six units.

Second, the CID has started to eliminate the centralized 'Control' division, see figure 7.1, and delegate control processes and responsibilities to each of the CID's departments and units [Task Force Z&R 1994a and 1994b]. To this end, the CID is developing standards for project plans and project reports. Also, effort is put into acquiring and implementing a standard application for registering the working hours of CID employees [Paternotte and Ligteringen 1994]. The goal of these changes is to be able to report and to justify the execution of projects and other activities that the CID carries out. Justification of projects and activities has become necessary ever since organized crime became a topic with a high political priority. The change process in the CID coincides with a process of decentralizing control authority in the Amsterdam Municipal Police Force as a whole [Broer 1994].

During our third case study and the study reported in [Meel 1994] that was carried out simultaneously, a number of the problems described in table 7.1 were addressed. However, some problems remained, while new ones emerged. A summary of these problems with respect to three categories is given in table 9.1.

Preparation of projects:

- Involvement of external experts
- Trend recognition
- MCDM process labor-intensive

Execution of projects:

- Unit structure - deployment agency structure
- Project management
- Successful project?
- Coordination of project execution

Tactical support:

- Deployment agency structure
 - 50%-50% support norm
-

table 9.1 Problems faced by the CID.

Problems with respect to the preparation of projects

At the end of our third case study, three problems with respect to the preparation of projects presented themselves. First, it is not yet clear how to involve external experts in projects with respect to specific types of organized crime or specific societal trends. Second, the current version of the ACOST is not capable of analyzing recorded information on criminal organizations and societal trends in such a way that trends become visible. The Information Department has identified this capability as a future requirement. Finally, executing the MCDM policy making process with the ACOST appears to be very labor-intensive.

Problems with respect to the execution of projects

The composition and structure of units remains a difficult issue for the CID. The CID is still working with so-called 'base units', see Section 7.2, but the CID management sees the 'deployment agency structure' as a more desirable alternative [Welten and Koningh 1994]. However, a number of cultural and practical issues still block the implementation of this form of network structure.

The study by van Meel [1994] confirmed and stressed the importance of project management information. It appeared that little management information is available in the CID because (1) units and other departments use incompatible (automated) administrations, and (2) the collected information on the progress of projects is often of a poor quality.

Another problem pointed out in [Meel 1994] is the clear definition of a 'successful project'. Often, a project that does not result in any arrests, can still be regarded as successful, for example because it resulted in an improved information position of the CID. However, given the need to justify its actions, the CID needs to state clearly what the outcomes of its projects are and whether they can be considered successful.

Finally, the CID management had indications that the coordination of its departments involved in the execution of projects was unsatisfactory. From the results presented in [Meel 1994] and daily experience, it appears that there are difficulties attuning the efforts of all parties involved in the execution of projects and that there are possibilities for improvement.

Problems with respect to tactical support

The Tactical Department is confronted with the same problem with respect to the 'deployment agency structure' as the units. Another problem that remains, is the 50%-50% support norm for Market I and Market II. At the beginning of van Meel's study [1994], there were some uncertainties with respect to this norm. It was found during the study that the norm is in fact not applicable. First, the used definitions are ambiguous. Second, the norm was found not to be according to the demand experienced by some departments in reality.

9.2.3 Assignment of the case study

The general assignment for this case study was to design and evaluate possibilities for organizational change so that the CID could employ its resources (men, means, and money) as efficient and effective as possible [Hengst et al. 1994]. Based on this assignment, two studies were initiated. The first study looked at the development of an approach and supporting instruments for the effective preparation of projects. The extension and further development of the ACOST is part of this study. An account of

the results of the study so far can be found in [Hengst 1995]. The second study focused on issues with respect to the execution of projects. The problem assignment of this second study was twofold:

1. To investigate the way in which the CID's departments involved in the execution of projects coordinate their activities and to design improved coordination mechanisms where necessary.
2. To design support for the management of CID projects by the departments involved.

The remainder of this chapter reports on the second study.

9.3 The conceptual model

In this section we describe the conceptual model of our second CID study. First, we discuss the object class definitions that together form the conceptual model. Then, we highlight specific coordination aspects by presenting the interaction, process, and actor model of the conceptual model using the taxonomy of coordination mechanisms from table 1.1. It should be noted that, given the scope of our problem assignment, the focus in this section is on the execution of projects. Moreover, for the sake of simplicity, we do not describe the whole conceptual model, but only address a number of key objects. For a more elaborate description, we refer to [Kuijlaars 1995; Vreede et al. 1995d].

The first version of the conceptual model was built in two 3 hour sessions with three analysts and one CID employee that had a good overview of the operational processes because of his central position in the organization. The first version of the conceptual model was evaluated, modified, and further extended as part of a third 7 hour session, in which eight CID stakeholders participated, representing the key actors in the CID's relevant processes. During all three sessions, TeamGraphics was used to support the modelling process. The group mainly focused on process models and actor models, as these turned out to be the easiest to understand and use for the participants. After the sessions, the analysts deduced the objects definitions and interaction model.

9.3.1 Object definitions

In the CID we can distinguish between thirteen different item classes that are relevant to the preparation of projects. Two of these item classes, a *project proposal* and a *project proposal decision*, we do not discuss as these were already described in Section 7.3.1.

There are several items that can initiate the execution of a project. First, the Triangle (the Mayor of Amsterdam, the Chief public prosecutor, and the Chief Commissioner of the Amsterdam Municipal Police Force) can send a *project assignment* to the CID management. The Triangle usually bases this assignment on information provided by the CID, or on the political agenda at that moment. The CID management may pass the assignment on to the Information Department so that the project will be prepared. Second, the Information Department may get *information on a problem area* from the ACOST and distill a description of a *critical problem area* from it. Such descriptions are regularly sent to the CID management. Third, other departments of the Amsterdam Municipal Police Force or external parties may be faced with a *calamity* that requires the involvement of the CID. Examples of calamities include kidnappings, liquidations, and severe sex crimes. A calamity, by definition, requires immediate attention. Therefore, the CID management directly appoints the calamity to a unit; the Information Department does not have to prepare its execution.

object class PROJECT ASSIGNMENT	object class INFORMATION ON PROBLEM AREA	object class CRITICAL PROBLEM AREA
is a MESSAGE	is a MESSAGE	is a MESSAGE
attributes -	attributes -	attributes -
object class CALAMITY	object class PROJECT PLAN	object class PROJECT PLAN
is a MESSAGE	is a MESSAGE	DECISION
attributes nature	attributes -	is a MESSAGE
		attributes decision result

After the CID management has decided that a project has to be executed, a unit has to write a *project plan*. In this plan, the unit describes the goals of the project, the target of the project, and a global planning. The project plan is sent for approval to the CID management and to the public prosecutor that is responsible for the judicial quality of the project. The execution of the project will not start unless both of these actors send a positive *project plan decision* back to the unit.

During the execution of the project, the unit regularly writes a *status report* which is sent to the CID management and the public prosecutor. Based on these reports, both actors issue *status report decisions*, in which they determine whether the project should continue or be stopped. A reason to stop a project is, for example, that it turns out to be impossible to get enough evidence to bring the target to court.

object class STATUS REPORT is a MESSAGE	object class STATUS REPORT DECISION is a MESSAGE	object class SUPPORT REQUEST is a MESSAGE
attributes -	attributes decision result	attributes type period
object class SUPPORT RE- QUEST DECISION is a MESSAGE	object class PROJECT INFORMATION is a MESSAGE	
attributes decision result	attributes -	

At various points during the execution of a project, a unit may require support from internal or external supporting departments. To this end, the unit submits a *support request* to the appropriate department. In this request, the unit states which type of support it requires for which period. The department concerned comes back to the unit with a *support request decision*, stating whether or not the request is granted. Finally, throughout the execution of a project, the unit has to store a great deal of *project information*. This information not only relates to the target that is investigated, but also to the way in which the project is going, how many hours are made etc. Project information is used to compose the status reports mentioned above.

There are various actors involved in the execution of projects to combat organized crime. We will not define and elaborate on the *Triangle*, the *Amsterdam Municipal Police Force*, and *External parties* as their role is significant but rather limited and has been sufficiently addressed above. The key actor in the execution of projects is the *unit*. A unit is first of all responsible for planning a project. In this respect, the unit defines a global project plan and, after approval by the CID management, presents this to the public prosecutor. If either of them requires it, the unit modifies the plan. Otherwise, it starts up the project.

During the actual execution of a project, the unit determines the subgoals of the project and then starts to gain insight in the project's targets. At certain moments during the project, it is decided whether to conclude the project or not. If this is not the case, the unit either continues or stops the project, depending on a decision from the CID management and the public prosecutor. Otherwise, the unit initiates the final operation, meaning that the targets are arrested, if possible. Then, the project administration is concluded, after which it is handed over to Justice. At the same time, the unit may suggest a number of administrative measures to frustrate the particular type of organized crime that was the issue in the project. Finally, while gradually reducing the size of the unit, some unit members have to continue working on the financial part of the project. This may take several extra months.

Other important actors are the CID management and the public prosecutor. Their role in the preparation of projects is already described in Section 7.3.1. With respect to the execution of projects, their primary role is to monitor the efficiency and effectiveness of the unit. They have to evaluate a project plan as defined by a unit. Together, they are also responsible for decisions whether to conclude or to continue the project. The public prosecutor is also responsible for handing the project over to Justice after its execution is finished and the project administration has been concluded.

object class UNIT
is an ACTOR

attributes

unit number

actions

Conclude project?

Continue project?

receive project proposal from Information Dept.

receive project plan dec. from CID man. and PP

receive status report dec. from CID man. and PP

receive support request decision from Other unit, or Inf/Tac/Tec Dept.

receive project information from project administration

receive calamity from CID management

send project plan to CID man. and PP

send status report to CID man. and PP

send support request to Other unit, or Inf/Tac/Tec Dept.

send project information to project administration

plan project

define global project plan

present project plan to public prosecutor

modify project plan

start up project

execute project

determine subgoals

gain insight in target

stop project

initiate final operation

conclude project administration

suggest administrative measures

gradually reduce unit size

hand project over to Justice

continue financial part

object class CID MANAGEMENT
is an ACTOR

actions

type of project to be executed?

approve of project plan?

conclude project?

continue project?

receive project assignment from Triangle

receive calamity from AMPF or external parties

receive critical problem area from Inf. Dept.

receive project proposal from Information Dept.

receive project plan from unit

receive status report from unit

send project assignment to Information Dept.

send project proposal decision to Inf. Dept.

send project plan decision to unit

send status report decision to unit

send calamity to unit

appoint calamity to unit

appoint project to unit

plan project

evaluate project plan

execute project

object class PUBLIC PROSECUTOR
is an ACTOR

actions

approve of project plan?

conclude project?

continue project?

receive project plan from unit

receive status report from unit

send project plan decision to unit

send status report decision to unit

appoint project to unit

plan project

evaluate project plan

execute project

hand project over to Justice

There are three actors that have a supporting role in the execution of projects. These are the *Information, Tactical and Technical Departments*, and an *Other unit*. To each of these actors, a unit may send a request for support. If a unit requires a specific specialism, it sends to request to the Information, Tactical or Technical Department. If a unit only requires extra men for a certain period of time, then it requests this capacity from another unit. This is, for example, the case when a unit has to observe more suspects than it has unit members.

object class TACTICAL AND TECHNICAL
DEPARTMENT
is an ACTOR

actions

receive support request from unit
send support request decision to unit

object class OTHER UNIT
is an ACTOR

actions

receive support request from unit
send support request decision to unit

object class INFORMATION DEPARTMENT
is an ACTOR

actions

receive inform. on problem area from ACOST
receive project assignment from CID man.
receive project proposal dec. from CID man.
receive support request from unit
send information on problem area to ACOST
send critical problem area to CID management
send project proposal to CID management
send support request decision to unit
build information position
identify critical problem area

There are two repository classes in the CID with respect to the execution of projects. The first concerns the *ACOST*, that we developed during the previous case study. The second repository class represents all systems in which the units store project information: the *project administration*. The units use a variety of systems and applications, both manual and automated for storing and retrieving project information. Examples of applications include RapidFile, WordPerfect, and Octopus. Paper forms are used to write down the number of hours that each unit member works during a week. The diversity of project administration systems is the result of an absence of organization-wide standards.

object class ACOST
is a REPOSITORY

actions

receive inf. on problem area from Inf. Dept.
send inform. on problem area to Inf. Dept.

object class PROJECT ADMINISTRATION
is a REPOSITORY

actions

receive project information from unit
send project information to unit

The actors and repositories that we described above, exchange items through various links. Two examples of such links are the *CID management link* and the *administration link*. The CID management link connects the unit with the CID management. This link supports the written communication of project plans and status reports from the unit to the management. The link also routes the management's oral or written decisions

with respect to project plans and status reports back to the unit. The administration link defines the interaction of the unit with the project administration repository. This link may either represent the electronic exchange of project information or the analog exchange in case the information is stored in paper files.

object class CID MANAGEMENT LINK
is a LINK

attributes

starting node (unit)
ending node (CID management)
bi-directional

actions

route project plan, project plan decision
route status report, status report decision

object class ADMINISTRATION LINK
is a LINK

attributes

starting node (unit)
ending node (project administration)
bi-directional

actions

route project information

Decision and task class definitions can be used to model the actions of the actors in the CID in more detail. An example of each class is given below. The decision concerns the determination of the type of project that has to be executed. This is a rather straightforward decision, as the outcome only depends on which item class representing a need to execute a project presents itself to the CID management and the Triangle. The task represents the initiation of the final operation during a project. This means that the target are (hopefully) arrested and the goals of the project are attained. The precondition for this task is that the public prosecutor, the CID management, and the unit together have decided that the project should be concluded.

object class TYPE OF PROJECT TO BE
EXECUTED?
is a DECISION

attributes

required items: project assignment, calamity, or
critical problem area
preconditions: -
postconditions: nature of project and next action
is known
next actions: send project assignment to
Information Dept. ('regular'),
send calamity to unit ('calamity')
decision rules: IF item is project assignment or
critical problem area THEN
'regular', IF item is calamity
THEN 'calamity'

object class INITIATE FINAL OPERATION
is a TASK

attributes

required items: status report decision
preconditions: decision result = 'conclude
project'
postconditions: -
next actions: conclude project administration
delay
att. assignm.: -

9.3.2 Interaction model

As we described in Section 7.3.2, the CID as a whole can be seen as a *functional hierarchy*. Responsibilities and lines of authority are clearly defined and exercised. The actors depicted in figure 7.2 and figure 9.1 each perform a well-defined function in the combat against organized crime, see [Welten and Koningh 1994].

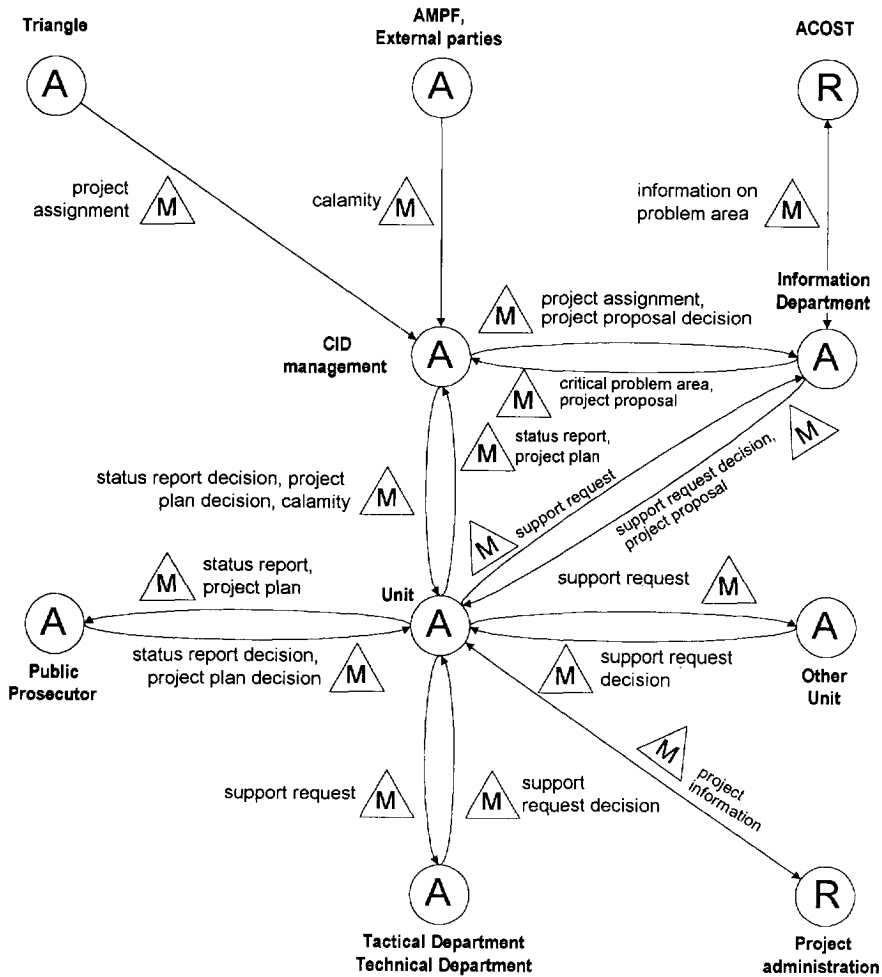


figure 9.1 Interaction model for the execution of projects in the CID.

When we look at the actors *directly* involved in the *execution* of projects in more detail, the public prosecutor, the CID management, the units, and the supporting departments, we see a *mixed functional and product hierarchy*. The public prosecutor and the CID management are both part of the functional hierarchy by definition of

their activities and responsibilities. The Tactical, Technical, and Information Department also represent the functional hierarchy, as they provide a unit with specialized support (function) when the unit requires this. The units themselves represent a product hierarchy. Each project that the CID has to execute, is handled by one unit that performs all necessary functions for this project, except for a few specialized functions that required a supporting department.

From the interaction model in figure 9.1 it appears, not surprisingly, that a unit has the most central position in the CID. It is in direct contact with all other actors that are actively involved in the execution of a project. A unit is to some extent autonomous with respect to its work. Although it has to have project plans and status reports approved of by the public prosecutor and the CID management, it may determine and propose its preferred course of action. The relationship between a unit and the actor that may offer support is a confirmation of a unit's relative autonomy. In general, the unit itself can ask for support without consulting with either the public prosecutor or the CID management.

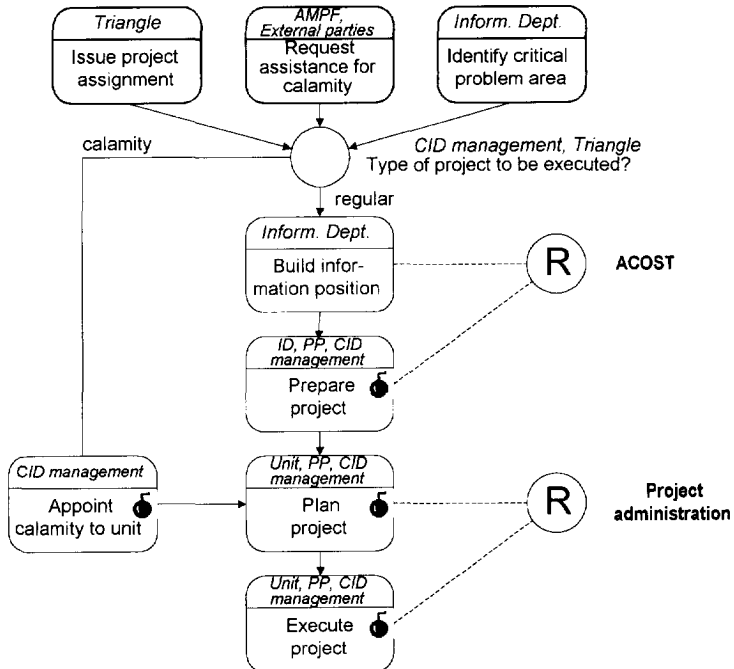


figure 9.2 Process model for combating organized crime.

9.3.3 Process model

Looking at the processes in the CID with respect to the execution of the projects, we see that *direct supervision* is one of the predominant coordination mechanisms. The Triangle orders the execution of a project by issuing a project assignment to the CID management, see figure 9.2. The CID management appoints a project or a calamity to a unit. The public prosecutor and the CID management give directions to the units about project plans and status reports. Hence, the execution of a project is directly supervised, although units may propose courses of action themselves.

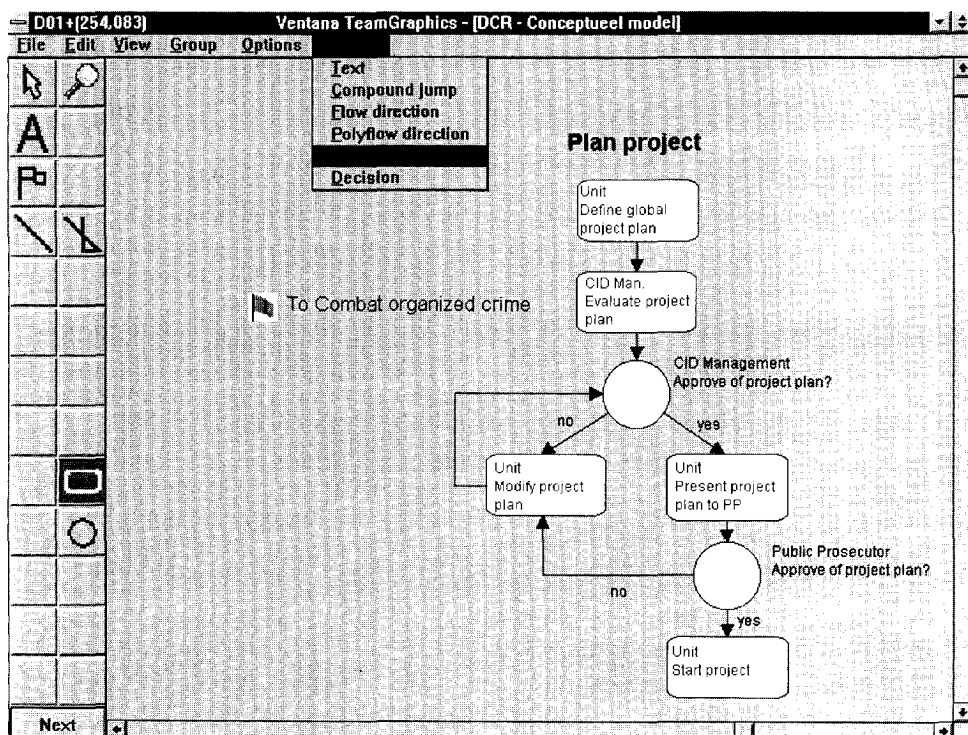


figure 9.3 Process model for planning projects as built in TeamGraphics.

If we look at the processes in further detail, see figure 9.3 and figure 9.4, it is apparent that *standardization mechanisms* are scarcely used. There are only two exceptions that concern standardization of *worker skills*. First, another department of the Amsterdam Municipal Police Force or an external party expects the CID to be skilled enough to handle a calamity once they brought it to the CID's attention. Second, unit members know what to expect from specialized actors from the three supporting departments when these are (temporarily) added to the unit. Apart from the exceptions, the absence of standardization mechanisms becomes apparent from the fact that there are a substantial number of approval tasks and decisions, combined with a considerable

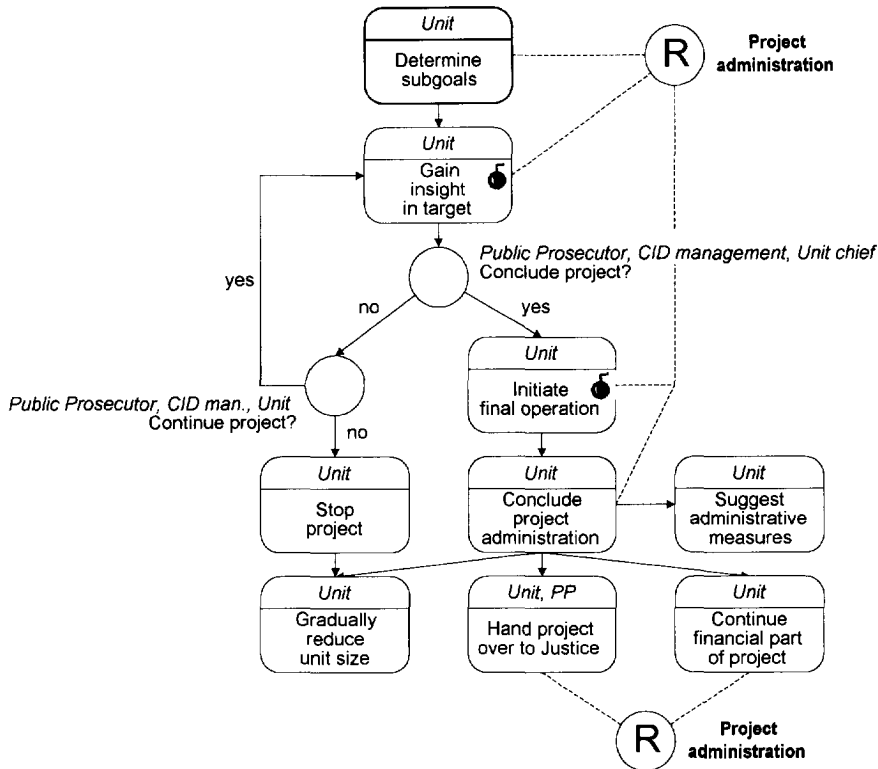


figure 9.4 Process model for executing projects.

amount of explicit communication between the actors to attune their activities. In this sense, *mutual adjustment* is as predominant a coordination mechanism as direct supervision. The different employees working in a unit coordinate their activities very informally. They are capable of standing in for each other whenever necessary. Support by other units or by the three supporting departments is informally coordinated as well. At the moment a unit experiences the need for support, it inquires informally with a 'support request' whether its need can be fulfilled, see also figure 9.5 in the next section and [Kuijlaars 1995; Vreede et al. 1995d].

The predominance of the mutual adjustment mechanism is expressed by the *coordination processes* in the CID. There are mainly *adaptive* coordination processes. The actors involved in the execution of projects coordinate their activities when they think it is necessary, see for example figure 9.4 and figure 9.5 in the next section. This is also illustrated by the considerable number of approval decisions.

Although there is a special planning process, see figure 9.3, one cannot say that there are many *anticipatory* coordination processes. A project plan is only made at the start of a project, but during the actual execution of the project it is hardly used. As can be

seen in figure 9.4 and figure 9.5, any reference to the project plan is missing. The only real exception is the action script which is made to support the execution of the final operation. This script describes in great detail what all actors involved in this activity have to do where and when.

Finally, there are no *review* processes in the CID. The actors involved do not collect information on the execution of projects that is used for the evaluation and reflection on the way these projects were carried out.

9.3.4 Actor model

Looking at the interdependencies between the activities of the individual actors, we see quite some commonalities compared to the process level. For individual actors, the predominant coordination mechanism is *mutual adjustment* as well: the execution of tasks and decisions is based on what is required at that moment. In this respect, the actor model for a unit gaining insight in their target is illustrative, see figure 9.5. The unit continuously collects information to support the hypotheses with respect to its target. Only if support is required or a status report needs to be written, then the unit takes the appropriate action. Similar models can be constructed for other actors.

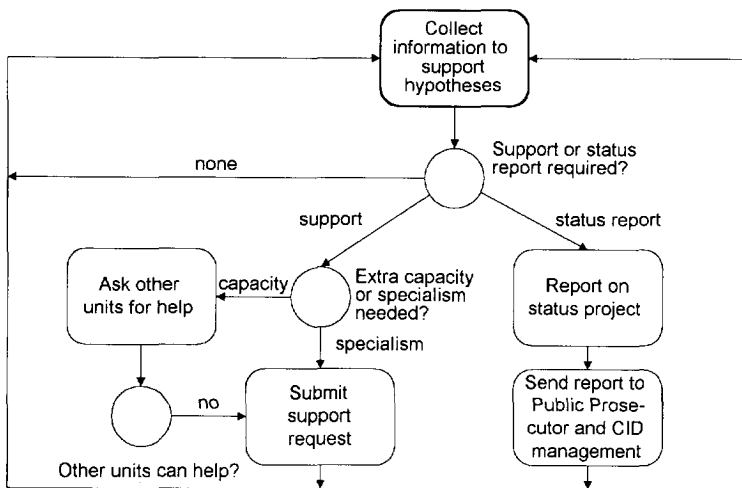


figure 9.5 Actor model for a unit gaining insight in their target.

9.4 The empirical model

This section describes the empirical model with respect to the execution of projects in the CID. The empirical model consists of two parts. First, we describe a number of organizational requirements that were identified and prioritized by a group of

stakeholders from the CID (Section 9.4.1). Second, we elaborate on a list of problem statements that was compiled in cooperation with the same group of stakeholders (Section 9.4.2). This section is concluded with a discussion of the verification and validation of the empirical model.

As we described in Chapter 8, we could have decided to build a dynamic empirical model, or simulation model, of the current situation at the CID. However, there were a number of reasons why we decided not to do this:

1. Previous experiences show the great deal of effort required to construct a (replicatively) validated simulation model. In a previous study by Meel [1994] a simulation model was built to investigate the time spent on project work. This model had a number of limitations due to the (too) extensive amount of time required to collect data and to the lack of some data altogether [Hengst et al. 1994]:
 - Only half of the units were represented by the model, because there was no empirical data available to represent the other half.
 - The supporting departments were not fully represented.
 - The empirical data that was incorporated in the model was of an arguable quality.
 - The coordination between the actors involved in the execution of projects was not explicitly addressed in the model.
2. We first considered reusing and extending Van Meel's simulation model. However, it was expected to be very time consuming to collect the data that was additionally required to extend the model on the issues described above. Moreover, we would probably have to settle for a great deal of expert estimations and significant reductions that may jeopardize the explicative and predictive power of the resulting model.
3. The organization was still in a constant flux. The number of units changed, while the number of employees per unit and per supporting department was not stable either. With this in mind and the experiences from Van Meel's study, there was a significant risk that when the model was finally ready, it would describe a situation that no longer existed. In addition, the stakeholders from the CID felt that based on the results from the previous study as well as from the insight that they gained by actively participating in the construction of the conceptual model, there was a sound basis to specify the current situation immediately in terms of problem statements.

Before turning to the description of the organizational requirements, we conclude this introduction with an account of the reductions we decided to make during the transition from the conceptual model to the empirical model:

1. The Triangle and public prosecutor were not taken into account in the empirical model. Although they play an important role in the execution of CID projects, their role is (1) no cause for any problems as experienced by the CID actors involved, and (2) beyond the influence and control of the CID anyway.
2. The Technical Department was not included in the empirical model either. This supporting department spends most of its time handling support requests from 'Market I clients'. The units do not need this type of support very frequently. Furthermore, the cooperation with the department does not give rise to any problems as far as the actors involved are concerned.

9.4.1 Organizational requirements

The first part of the empirical model represents the organizational requirements that the CID has to meet for the execution of its projects to be considered successful. The organizational requirements presented in this section differ from the ones presented in Section 7.4.1 as they mainly focus on the execution of projects instead of on the preparation of projects.

The set of organizational requirements was identified and prioritized as part of the 7 hour session in which the second version of the conceptual model was constructed. To support this activity, the group of stakeholders used GroupSystems V's Idea Organization module for brainstorming and categorizing requirements, and the Vote module for prioritizing them. During the session, the group identified 14 important organizational requirements. For an elaborate discussion of all requirements, we refer to [Vreede et al. 1995d]. Below, we only describe the six most important requirements:

1. *Insight into the execution of projects.* The CID management and unit chiefs need a thorough insight into the status and progress of projects. This insight should help them manage the execution of projects.
2. *Insight into the required quality and quantity of support.* In order to manage the employment of actors from supporting departments, there has to be insight in what quality and quantity the clients of these departments require. This insight should, amongst other things, also indicate whether the 50%-50% support norm for Market I and Market II is realistic.
3. *Clearly stated goals and policy.* This is the same requirement as described in Section 7.4.1.
4. *Clearly stated project goals.* In order to manage the execution of an individual project, the unit chief and CID management should be working with project goals that clearly define the desired outcome of the project. This should also facilitate the determination whether a project was successful or not.
5. *CID as societal actor.* In order to successfully combat organized crime, the CID has to behave as an actor in a societal network. To this end, the CID should

cooperate with societal partners from various disciplines. This is especially important when the target of a project is not a criminal organization but a societal trend, or when one of the project goals is to give advice to legislative bodies or the public administration.

6. *Handling calamities adequately.* The CID has to be flexible enough to immediately handle calamities without disrupting the execution of regular projects.

We depict the relationships between the organizational requirements and the processes in the CID in table 9.2. These relationships help us to determine which are the most important issues that we have to focus on during the further specification of the conceptual model. As stated before, the relationship between a requirement and a process can be twofold. A process may support the attainment of a requirement. A requirement may support the successful execution of a process.

Organizational Requirements	1. Insight into execution of projects 2. Insight into required support 3. Clearly stated goals and policy 4. Clearly stated project goals 5. CID as societal actor 6. Handling calamities adequately					
Processes						
1. Appoint calamity to unit	y	-	-	-	-	x
2. Plan project	x	x	y	x	x	x
3. Execute project	y	y	y	y	x	x
4. Gain insight in target	x	x	-	y	y	x
5. Initiate final operation	y	y	-	y	-	x
6. Request support	y	y	-	y	x	x

x = process should support the attainment of the organizational requirement

y = organizational requirement should support the successful execution of the process

table 9.2 *The relation between the CID's organizational requirements and processes.*

It is clear from table 9.2 that two processes are of paramount importance to the CID. First, 'Plan project' should ensure that project goals are clearly defined. If this is not the case, then the execution of the four consecutive processes is in danger. Second, during the process 'Gain insight in target', status reports are written that should give insight in both the execution of projects and the demands on the supporting departments by comparing them to information from the related project plans. This dual insight is required to facilitate the execution of the other processes.

Furthermore, it is noteworthy that the fifth organizational requirement, 'CID as a societal actor', can only be attained if explicit attention is paid to planning and actually implementing the involvement of (external) expert support in CID projects. If this requirement is fulfilled, then it is expected that a better insight into a project's target is gained. Finally, it is clear that all processes serve to support the adequate handling of calamities. In this sense, it is clear that this requirement is imposed on the CID from outside the organization. The requirement does not actually support the execution of any of the indicated CID processes itself.

In summary, all processes support the attainment of at least one of the most important organizational requirements. Of these processes, the first, 'Appoint calamity to unit', is of less importance. In the next section, we therefore focus on the problems that occur in the CID's current situation that have a negative impact on the other processes.

9.4.2 Problem list

The problems in the CID's current situation were identified in the same session as the organizational requirements. The Group Outliner module of GroupSystems V was used to support this activity. The outline represented the process hierarchy in the conceptual model. Problem statements were added as comments to the appropriate level (process) in the outline. We discuss the way in which this activity was carried out in more detail in Section 9.7. The original list of problem statements consisted of exactly 100 items, 38 of which were entered beforehand by the analysts to stimulate the brainstorming process. After cleaning up and categorizing the list, 28 problem statements remained which are described in [Vreede et al. 1995d]. In this section, we summarize the most important problems. These are organized in two categories: work process and management information.

Work process

- *Timing between preparation and execution.* The Information Department has insufficient insight in the progress of current projects. As a result, it is difficult to plan the preparation of projects in such a way that a project proposal is available when necessary. Likewise, the units do not know the status of the project preparation and hence cannot anticipate it.
- *Impact planned projects on other projects.* When a project is planned, the impact of this project in terms of available resources with respect to other projects is not evaluated. When the projects are executed, coordination problems occur when different projects require the same resources.
- *Feasibility of planned support.* During the planning of projects, the units do not inquire whether their support requirements can be met in reality. There is, for example, no contact with the supporting departments concerned. As a result, the project plan does not always describe a realistic or possible course of action.

- *Support specialisms unknown.* Units sometimes cannot anticipate which specialisms they will require during a project because they have insufficient insight in the available types of support.
- *Drifting project goals.* Project goals are not always defined in terms of concrete results that have to be achieved. This makes it hard to manage a project. Moreover, there often is a tendency to broaden the goals during the course of project, because new evidence can continuously be collected. The main risk of this phenomenon is that the criminal organization keeps on expanding while there is already sufficient material to initiate a final operation and arrest the key players.
- *Project plan not used during execution.* During the execution of projects, the original plan is not used to monitor and direct the course of a project. This is mainly due to the lack of useful information about the execution of a project (see also the next category).
- *Relation between calamities and project delays.* In the current situation it is not possible to determine whether delays in the execution of a project are the result of calamities. Therefore, neither the unit nor the CID management can give account for experienced delays to their respective superiors.
- *Support requests are late and ad-hoc.* Requests for support often reach the supporting department just before the support is actually required. 'It should have been organized yesterday.' This is partly the result of bad planning, and partly the result of the insufficient communication about the project plan between the unit and the supporting departments concerned. Consequently, situations in which two or more requests for the same type of support are issued, cannot be avoided.

Management information

- *Registration of project management information.* The CID management and unit chiefs lack management information with respect to the execution of projects. First, the registration of working hours in projects is incomplete: only the working hours of unit members is registered, not those of supporting department employees that are involved in the project. Second, working hours are only registered per employee, not per project. With the composition of a unit varying and unit members assisting other units every now and then, working hours per project are difficult to determine. Finally, the accuracy of registered working hour is questionable, see also [Meel et al. 1994].
- *Evaluation of support employment.* It cannot be assessed to what extent the supporting departments meet the requirements of their clients. Requests for support, especially the ones that are turned down, are not registered. The hours that support employees work for certain projects within or outside the CID are not registered either. The chiefs of the supporting departments have no information about the actual demand for and the employment of the employees in the various specialisms that are part of their departments. In other words, there is no

information that supports the CID management and the supporting departments' chief to manage these departments.

- *Different registration systems.* It is not possible to integrate the project information that *is* registered because different systems are used. As a result, the CID management does not have a combined and abstracted overview of the execution of CID projects.

Given the above problems, it is evident that the organizational requirements are not met in the current situation. Based on the material presented in this section, we will diagnose the CID's problem situation in Section 9.5. However, first we describe the validation of the empirical model.

9.4.3 Model validation

There were no objective criteria to validate the empirical model of the CID's current situation. Therefore, we had to rely on the assistance of CID employees to validate the model. Obviously, the empirical model was to some extent validated (and verified) implicitly. A number of CID stakeholders actively participated in the construction of both the conceptual model and the empirical model, with the former model being the basis for the latter one. Similar to our first CID study, we took two explicit validation steps.

First, we distributed a questionnaire to the participants of the session in which both the conceptual and empirical model were constructed. Amongst other questions, we asked the participants whether the outcome of the session coincided with their personal views of the current situation. On a scale from 1 to 5, 1 being extremely negative, the participants responded an average of 4.5 with a standard deviation of 0.5. Second, we presented the results of our joint modelling activities to a large group of other stakeholders that participated in two group sessions that were held to design alternatives for change (see Section 9.6). Although the presentation resulted in an animated discussion, it did not result in any corrections or extensions to the empirical model. Given the results of these two activities, we felt that the empirical model could be regarded as a correct and valid representation of the CID's current situation.

9.5 Problem diagnosis

In this section, we diagnose the CID's problem situation by reviewing the problems that were identified in the previous section and their impact on the processes that are part of the conceptual model. First, we take a closer look at the problems with respect to the CID's work processes. In table 9.3 we depict which processes are affected by these problems.

Problem	Processes affected
Timing between preparation and execution	Plan project
Impact planned projects on other projects	Gain insight in target, Initiate final operation, Request support
Feasibility of planned support	Execute project, Gain insight in target, Initiate final operation, Request support
Support specialisms unknown	Plan project, Gain insight in target, Initiate final operation, Request support
Drifting project goals	Execute project, Gain insight in target
Project plan not used during execution	Execute project, Gain insight in target, Initiate final operation, Request support
Relation between calamity and project delays	Gain insight in target
Support requests late and ad-hoc	Execute project, Gain insight in target, Initiate final operation

table 9.3 The processes affected by the identified work process problems.

Based on the information presented in table 9.3, we can make a number of observations. First, the unfamiliarity with support specialisms appears to affect almost all processes. Second, the actual execution of projects seems to be frustrated by two problems in particular: the facts that the feasibility of planned support is not checked, and that project plans are not used during project execution. Third, there are remarkably few problems that hinder the planning of projects. However, it is even more remarkable that problems related to planning issues (the second, third, sixth, and eighth problem) affect *all* other processes. In other words, the planning of projects itself is a problematic process. In this respect, attention is required to support the attainment of clearly stated goals and policy, something which is not really supported by any of the current processes, see table 9.2. Finally, it turns out that all problems but the first obstruct the process 'Gain insight in target', especially with respect to the quality and usefulness of project status reports. Together with 'Plan project', this process should support the attainment of the two most important organizational requirements, see table 9.2: insight into the execution of projects and into required support.

In summary, alternatives for organizational change in the CID should focus on two issues:

1. The process with which project plans are defined *and* used during the execution of projects.
2. The establishment of overall goals and policy that would facilitate and direct the planning and execution of projects.

The problems with respect to management information and the processes they affect, are depicted in table 9.4.

Problem	Processes affected
Registration of management information	Plan project, Execute project, Gain insight in target
Evaluation of support employment	Plan project, Gain insight in target, Initiate final operation, Request support
Different registration systems	Plan project, Gain insight in target

table 9.4 The processes affected by the identified management information problems.

Obviously, the problems with respect to management information mainly affect two processes: 'Plan project' and 'Gain insight in target'. As stated above, these are the two processes that have to ensure the attainment of the two most important organizational requirements. Especially the problems affecting the planning process seem to be (partly) responsible for the planning problems given in table 9.3. Bearing the above in mind, we suggest that attention be further focused on two issues:

1. The registration of useful project information with respect to the progress of a project and the employment of (human) resources.
2. The design of IT to support the collection and analysis of project management information. The CID management, unit chiefs, and chiefs of supporting departments have to use this information to coordinate the execution of projects.

Together with a large group of stakeholders from the CID, the four issues mentioned in this section were taken as a starting point to develop a number of change alternatives. These alternatives are introduced and discussed in the next section.

9.6 Alternatives for change

From the discussion in the previous sections, it follows that the CID has to extend its repertoire of adaptive coordination processes with *anticipatory* and *review* processes. The CID needs to improve the way in which projects are planned. These plans should be used to monitor the execution of a project. IT support is to be designed to facilitate the collection and analysis of information for coordinating the execution of projects. Together with a group of stakeholders representing all relevant departments and processes in the CID, a number of change alternatives were designed and evaluated. Three alternatives are discussed in this section: the creation of a project agenda, the continuous coordination of projects, and an information system to support this coordination. Other change alternatives include the integral preparation of projects and the way in which support requests are handled. These latter alternatives are not discussed in this section; we will shortly address them in the reflection on this case study (Section 9.7). For a complete and elaborate overview of all change alternatives, we refer to [Kuijlaars 1995; Kuijlaars et al. 1995].

9.6.1 The agenda process

The first alternative concerns the design of a process for the creation of a project agenda. A *project agenda* describes the projects that are considered to be executed during a certain period. In this sense, a project agenda represents the combined planning of a set of projects over a period of, for example, one year. The goals of the creation of a project agenda, in short ‘the agenda process’, is to match the available capacity of the CID to the projects that are considered to be most important. This process naturally follows the policy making process which we discussed in Section 7.6.1. the output of this process, a prioritized list of problem areas, is used as input for the agenda process.

Based on the problems presented earlier, we made a first conceptual draft design of the agenda process in terms of an interaction model and a process model. Next, we transformed these models into an (animated) simulation model, see figure 9.6. For this purpose, we used the simulation language Arena, which is the successor of SIMAN/Cinema [Pegden et al. 1995].

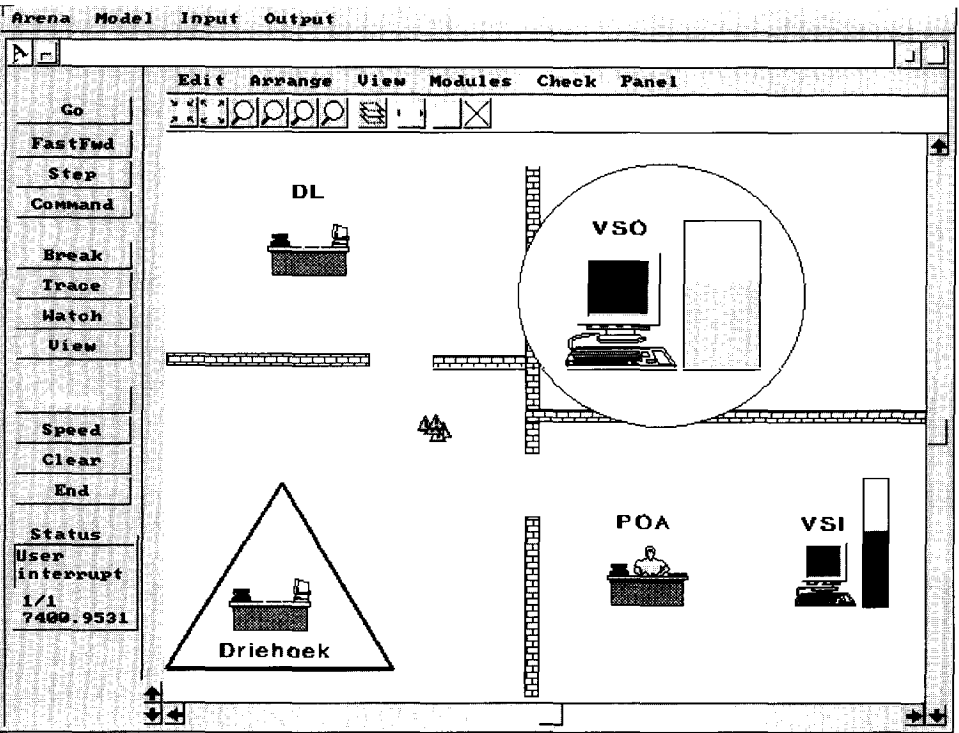


figure 9.6 Animated simulation of the proposed agenda process.

The simulation model was presented as a kick-off model to two groups of stakeholders (20 in total) during two GroupSystems V sessions [Kuijlaars and Vreede 1995a]. During each session the proposed process was introduced and explained to the stakeholders and a demonstration of the simulation model was given. As soon as the model and the process it depicted was clear to the stakeholders, the Topic Commenter module was used to brainstorm about the proposed agenda process. During this process, the stakeholders were encouraged to be as critical and creative as possible. The stakeholders entered comments and held electronic discussions on several topics such as 'expected problems with the proposed process', 'alternative ways to create a project agenda', and 'remaining questions'. In total the two groups entered 82 comments with respect to the proposed process.

Based on the stakeholders' feedback and the subsequent group discussions during the sessions, the definitive design of the agenda process was conceptualized. The process model is depicted in figure 9.7, for the interaction model we refer to [Kuijlaars 1995; Kuijlaars et al. 1995].

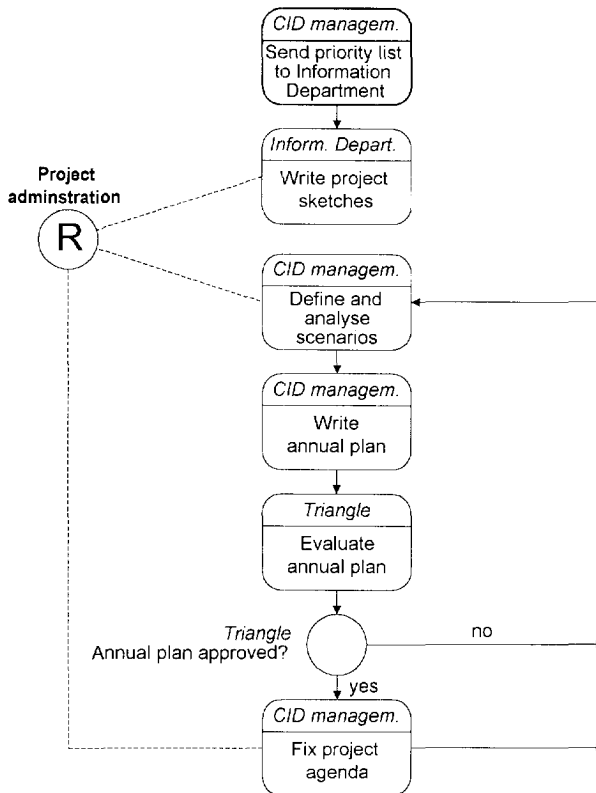


figure 9.7 Process model of the agenda process.

In the agenda process, the CID management first sends a priority list of problem areas to the Information Department. Based on this list, the Information Department writes a project sketch for each problem area. A project sketch gives a rough description of the project, an estimation of the length of the project and of the required capacity. Project sketches are stored in the project administration repository.

At least once a year, the CID management defines a number of scenarios. A scenario consists of a hypothetical sequence of projects in time. The project administration repository is used to analyze these scenarios to see which scenarios consists of an optimal blend of project priorities and available resources in terms of people and means. During this activity, the expected occurrence of calamities is also taken into account. Based on the analysis of the scenarios, the CID management draws up the annual plan of the CID and sends this to the Triangle. The Triangle then evaluates the annual plan. Next, the CID management (1) may fix the project agenda for the coming year in the project administration repository if the Triangle approved of the annual plan, or (2) may evaluate a number of other scenarios and adapt the annual plan if the Triangle did not approve of it.

The final design of the agenda process is approved of by the CID management and the other parties that were involved in the evaluation of the proposal for agenda process. However, the process has not yet been implemented in the organization. As can be seen from the process model, the execution of the agenda process is closely related to the implementation of an adequate project administration repository. This will be addressed in Section 9.6.3.

9.6.2 The continuous coordination of projects

The second change alternative addresses the continuous coordination during the execution of a project. In the current situation, the coordination of projects is cumbersome because project plans are not used to evaluate a project's progress, whereas the status of a project is not adequately monitored. In the second alternative, the planning and evaluation of a project are considered to be cyclic activities, that are continuously carried out *during* the execution of a project. The goal of this alternative is to ensure the ongoing coordination of a project. This is depicted in figure 9.8.

The change alternative presented in this section was designed and evaluated in the same manner as the first alternative. Initially, we designed and conceptualized processes to support the ongoing coordination of projects in terms of an interaction model and process models. A simulation model was subsequently constructed in Arena. This model included several animation views. These views included a single unit, a single unit with supporting departments, and the overview over all units and other relevant actors shown in figure 9.9.

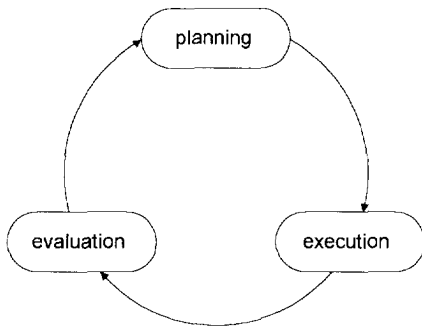


figure 9.8 Ongoing coordination during project execution.

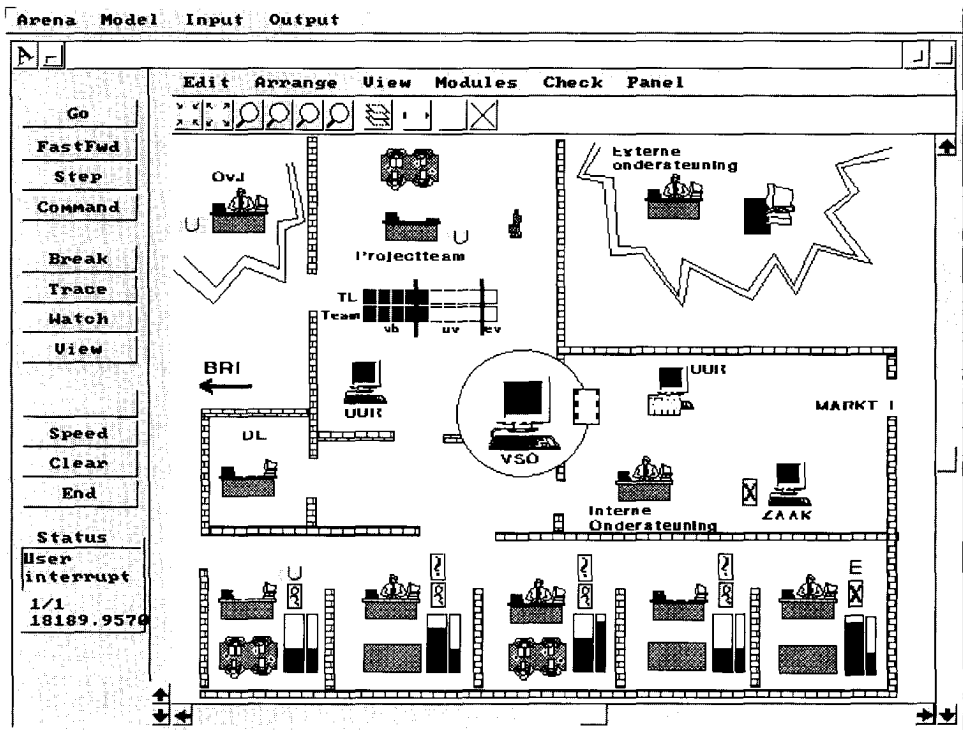


figure 9.9 Animated simulation of the ongoing execution and coordination of projects.

The simulation model was shown to the same group of CID stakeholders during the same sessions as the model of the agenda process [Kuijlaars and Vreede 1995a]. During the presentation of the model, the different views were used to highlight separate parts of the model in turn and hence reduce the complexity of the proposed alternative for the stakeholders. After the presentation, the Topic Commenter module

was used to collect feedback and electronically discuss the proposed processes. In total, the two groups entered 125 comments.

The feedback of the stakeholders and the group discussions during the sessions enabled us to modify the proposed change alternative and conceptualize the definitive design in terms of an interaction model and two process models. For the interaction model, see [Kuijlaars 1995; Kuijlaars et al. 1995]. The first process model describes the continuous planning, execution, and evaluation of a project, see figure 9.10. The planning activity is further decomposed in figure 9.11.

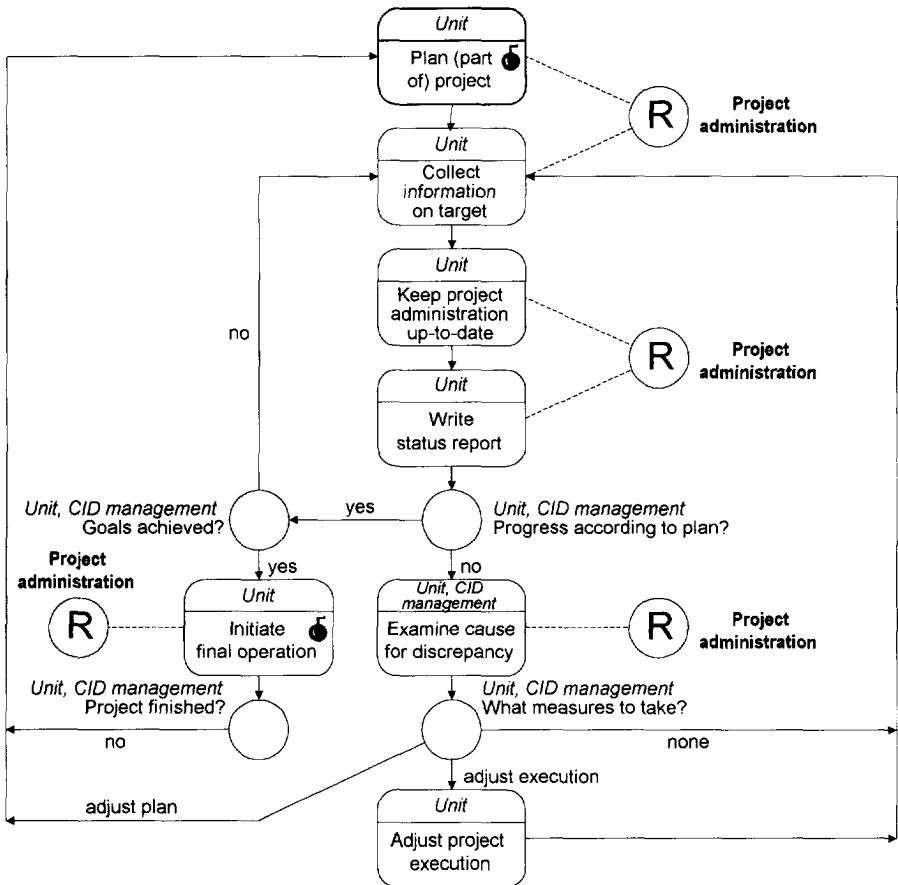


figure 9.10 Process model for continuous planning, execution, and evaluation.

The execution of a project will be carried out in phases, each of which represents a part of the project. The execution starts with the planning of (a part of) a project by a unit. After the project plan has been approved of by the CID management and the public prosecutor, the unit starts collecting information on the target of the project

(part). The unit also keeps the project administration up-to-date with the project administration repository. The working hours of the unit members are registered, as well as the results and status of the project at that moment. With this information the unit writes a status report, which it sends to the CID management for approval. Together with the unit, the CID management checks whether (a part of) the project is still progressing according to plan. If this is the case, then the unit and CID management determine whether all goals of (a part of) the project have been achieved already. If not, then the unit continues to collect information on the target, otherwise, it initiates a final operation. If the final operation concerned a part of the project, the unit critically evaluates the existing planning for the remaining parts and modifies it where necessary.

If (a part of) the project is *not* progressing according to plan, then the unit and CID management together examine the cause of the divergence. It can be, for example, that goals were set too ambitiously, that the planning was too optimistic, or that dependencies on other parties slowed down the execution. Depending on the results of this inquiry, three measures can be taken. First, the unit may have to adapt the project's plan. The new project plan is stored in the project administration repository. Second, the unit may have to adjust the way in which the project is carried out, e.g. by sooner initiating the final operation of project parts. Third, if the discrepancy is not considered to be severe enough, then no measures are taken.

As we described above, the planning of a project is an activity that is continuously carried out during the execution of a project. Units have to plan each phase or part of the project. During the planning process, see figure 9.11, a unit first estimates the amount of resources that it will require for the part of the project it is planning. As a result, the unit submits a number of support requests to the appropriate departments that are stored in the project administration repository as well. This will enable the supporting departments to keep a planning of their resources as well. Next, the unit (re)write the project plan, which will be evaluated by the CID management. If the CID management approves of the plan, the unit conveys it to the public prosecutor. If this actor also approves of it, then the execution of (a part of) the project can start. If either the CID management or the public prosecutor does not approve of the plan, then the unit has to re-execute one or more of the previous activities.

The CID management and other stakeholders that participated in the sessions, approved of the final design for the continuous execution and coordination of projects. The actual implementation of the processes described above, is planned to coincide with the implementation of the project administration repository. This is discussed in more detail in the next subsection.

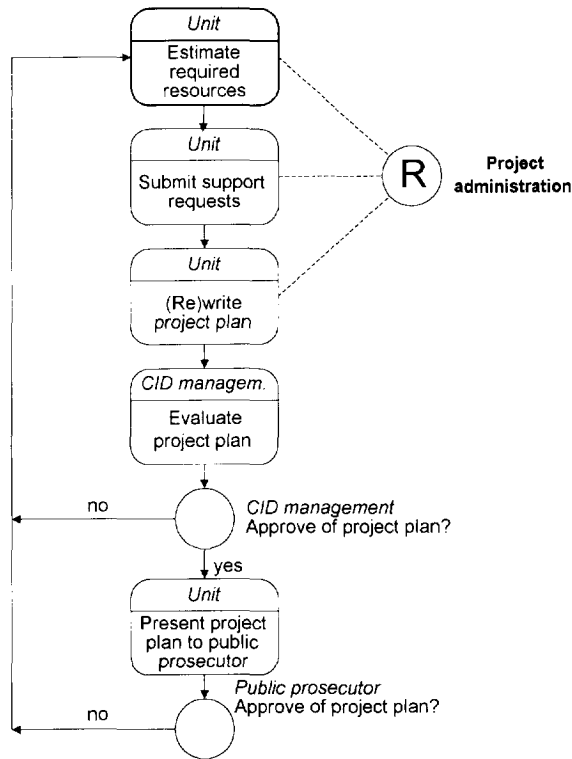


figure 9.11 Decomposition of the planning activity.

9.6.3 The Project Management Information System

As is apparent from the discussion of the previous two change alternatives, the actors involved in the execution of projects have to collect and analyze a lot of information in order to coordinate the execution of projects. This information consists of, for example, working hours of employees, project plans, and project agendas. In order to support these actors and the processes they participate in, it was decided to develop an information system that would facilitate the collection and analysis of project information. More specifically, the goal of this system was threefold:

1. To support the *monitoring* of current projects. In this sense, the system had to facilitate *adaptive* coordination processes.
2. To support the *planning* of current and future projects. In this sense, the system had to facilitate *anticipatory* coordination processes.
3. To support the *evaluation* of concluded projects. In this sense, the system had to facilitate *review* coordination processes.

In order to achieve this goal, the information system had to support the various actors in registering and reviewing their contribution to the execution of projects. The combined overview of these contribution would bring about insight in the CID's execution of projects and the required support, i.e. the CID's most important organizational requirements. This system was named the PROject Management Information System (PROMIS). The PROMIS would fulfil the role of the project administration repository depicted in the preceding subsections. Note that with 'project administration' we do not imply the registration of the actual *content* (target) of a project. This is registered in systems such as Octopus. The PROMIS has to administer information with respect to the *execution* of a project. In the remainder of this section, we first discuss the way in which the PROMIS represents information on projects. Next, we introduce the main functions of the system. We conclude with a description of the development of a prototype of the PROMIS.

The PROMIS distinguishes between projects and cases. A *project* represents the employment of CID resources in order to achieve a goal that directly relates to the main goals of the CID [Kuijlaars 1995]. A *case* represents the employment of resources from the supporting departments (1) to assist in a project, or (2) to assist an external client of the CID [Kuijlaars 1995]. In other words, projects represent work carried out by units, cases work carried out by the supporting departments.

In the PROMIS, projects are defined and represented in terms of a number of standard project phases. The goal of using standard project phases is twofold. First, it can serve as a guideline during the planning of a project. Second, it enables the units and the CID management to compare different projects. The standard project phases include:

1. *Project advice*. In this phase, the Information Department identifies a critical problem area and sends it to the CID managements.
2. *Information phase*. During this phase, an information team prepares a project, see also Chapter 7.
3. *Preliminary investigation*. The goals of the project are determined in this phase. The unit plans the project.
4. *Subject phase*. The goals of the project are set and pursued. The unit starts the actual execution of the project by trying to gain further insight in the target and collect evidence.
5. *Final operation*. During this, usually, short phase, the targets of the project are arrested. If the goal of the project was not to arrest a target, but, for example, to gain insight in an area of attention, then the final operation represents the compilation of a report.
6. *After care*. The unit concludes the project. The administration is brought up-to-date and closed. If necessary, financial and legislative subprojects are continued.

For each project phase, the PROMIS has to handle information on which employees participate, on which activities are planned, and on the working hours of all involved employees. Moreover, within each project phase, one or more *cases* can be carried out, if the unit requires the assistance of a supporting department. Information with respect to the execution of these cases also has to be stored in the PROMIS.

The required functionality of the PROMIS was determined in two steps. First, during the sessions in which the proposed designs for the new processes were presented and evaluated, we asked the participants to brainstorm and identify how they perceived that the PROMIS would support their activities in the new situation. To this end, the PROMIS and the proposed interaction with it, was also depicted in the simulation models, see computer icon labelled 'VSO' in figure 9.6 and figure 9.9. Second, from the final design of the new processes, the requirements of the PROMIS were deduced from the specifications of the activities that it should support, see also [Kuijlaars 1995]. These two steps resulted in the specification of the following four main functions that the PROMIS should offer:

1. *Definition.* With the PROMIS, it must be possible to define projects, project phases, and project plans. It must also be possible to define cases and link these to projects.
2. *Registration.* This function has to ensure that all information is registered with respect to the progress of a project and to the employment of resources. Also, received support in the form of cases should be registered.
3. *Report.* Information on current and concluded projects must be abstracted into management reports. These reports must, for example, be able to give insight in the way in which a single project or a group of projects is planned and in differences between planning and reality. Aggregated historical information with respect to projects, cases, and calamities can support units and CID management during the planning of projects. Every user of the PROMIS should be able to define reports that are accustomed to his or her situation.
4. *Analysis.* First, this function should support the agenda process by analyzing a set of scenarios. Based on a number of criteria and historical information, an optimal match of project sketches should be suggested for a predefined period. Second, this function has to predict what the impact of a calamity will be on the planned progress of one or more projects.

For each of the PROMIS' main functions, a number of subfunctions were defined as well, for example specify, generate, and print report. A detailed description of the PROMIS' functional specification can be found in [Kuijlaars et al. 1995]. Based on the functional specification, a first prototype of the PROMIS was built in Microsoft Access. We chose this data base application because it supports the flexible exchange

of data in a multi-user setting. Moreover, being a Windows application, no significant effort had to be spent on developing a user interface.

The PROMIS prototype covered three of the four main functions. The 'Analysis' function was left out, because (1) there was a risk that the PROMIS prototype would seem too complex to users, and (2) the value of the 'Analysis' function will better show once a realistic set of project and case information has been collected. The Report function was not implemented completely, as we wanted to use the prototype to further determine what types of reports the future users wanted to have. To test the prototype in a realistic way, a set of test data was compiled based on historical information from completed projects.

<u>Project module</u>	<u>Case module</u>
unit composition	product group composition
- attach unit member	- attach product group member
project planning	- define type of working hour
- milestone planning	define specialisms
- phase planning	support requests
- plan employment	- overview requests
- for activity	- plan support employment
- for specialism	- appoint employee
- request support	- plan employment per week
report employment	report total employment
report support requests	

figure 9.12 Menu structure of the PROMIS prototype.

The menu structure of the PROMIS prototype is given in figure 9.12. As can be seen, the prototype consists of two modules: a project module and a case module. Employees from units will use the project module, those from supporting departments the case module.

Figure 9.13 gives an example of an registration screen from the project module. It illustrates the project plan, consisting of five project phases (the 'project advice' phase is not included in the planning of a project). Two examples of reports are shown in figure 9.14 and figure 9.15. The first represents a report from the project module, depicting the difference between the planned and realized amount of work per week for a project over a number of weeks. The second gives an overview of the planned employment of all employees of a particular supporting specialism over a number of weeks.

project

projectnaam: Palixaan, Puerdes

projectcode: 290

projectleider: Gerrie Gruijtes

beginndatum: 01-Jul-94

gerealiseerde beginndatum: 01-Aug-94

einddatum: 01-Jul-95

gerealiseerde einddatum:

openingschem

projectplanning

Fasen:

	fasenaam:	beginndatum:	gerealis.:	einddatum:	gerealis.:
▶	informatiefase	01-Jul-94	01-Aug-94	01-Nov-94	01-Dec-94
▶	vooronderzoeksfase	01-Nov-94	01-Dec-94	01-Jan-95	01-Jan-95
▶	subjectfase	01-Jan-95	01-Jan-95	01-May-95	
▶	afrondding	01-May-95		15-May-95	
▶	nazorg	15-May-95		01-Jul-95	
*					

Mijlpalen:

	mijlpaal:	datum:	gerealis.:
▶	afrondding vorig project	01-Nov-94	1-01-95
▶	aanvraag nieuw project	01-May-95	
*			

figure 9.13 Screendump of the registration of the planning of a project.

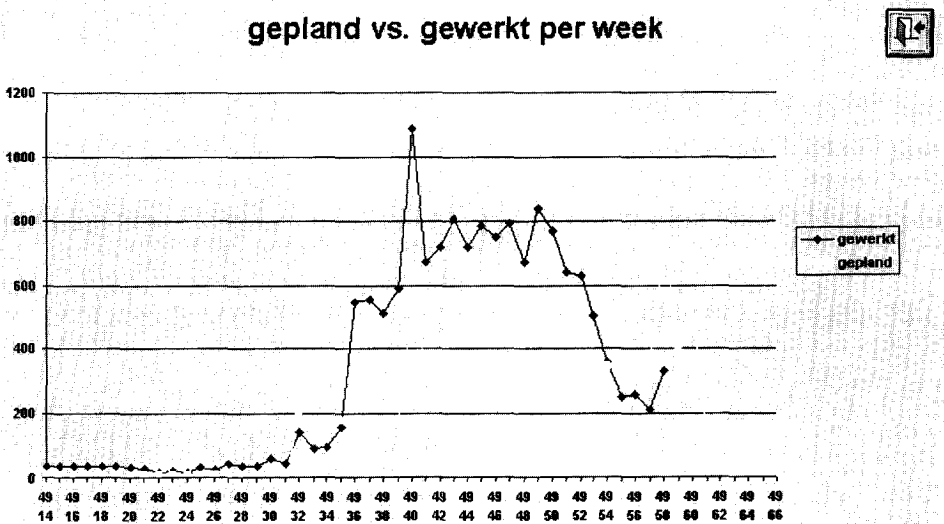


figure 9.14 Screendump of the difference between planned and realized project efforts.

totaalinzet per week

geplande uren ☐
 gewerkte uren ☐
 verschil gepland-gewerkt ☐

Overzicht starten bij week: van het jaar:

(operingscherm)
 (weekinzet op aanvraag)
 totaalinzet per week

stamn	naam	18	19	20	21	22	23	24	25	26	27
311605	Alberts, A.	10	22	23		44					
314793	Dusschoten, P. van	10	22	23		44					
342235	Groot, C. de	25	25	25	15	44	15	15	15	15	15
636348	Kok, W.	25	25	25	15	44	15	15	15	15	15
443233	Maas, J.K.	10	22	23		44					
412729	Poldermans, B.	10	22	23		52					
333004	Soest, D. van	10	10	10		44					
294422	Vis, A.	10	22	23		44					

figure 9.15 Screenshot of the planned assignment of supporting employees.

The evaluation of the PROMIS prototype took place during two subsequent GroupSystems V sessions with 15 stakeholders participating [Kuijlaars and Vreede 1995b]. These stakeholders represented the CID management, the units, and supporting departments. During these sessions, GroupSystems V and the PROMIS prototype were running simultaneously. In order to structure the evaluation and to ensure that the stakeholders experienced all functions of the prototype, a scenario was developed for each module. Each scenario consisted of a number of assignments, see table 9.5.

Project module	Case module
Enter project chief	Define product group
Extend project unit with yourself	Define specialism
Book and plan project phase	Define types of working hours
Define required employment of specialism	Register support request
Define deployment of unit members	Define employment based on request
Define project parts	Define employment based on already scheduled employment
Modify requests according to planning	Generate report of planned personnel per week
Generate report of planned activities and specialisms	Generate report of planned vs realized employment per request
Generate report of project per week	

table 9.5 Scenarios for the prototype evaluation.

After introducing the prototype, the stakeholders received the script. During the execution of their assignments, stakeholders were encouraged to entered comments,

ideas, problems, and questions in GroupSystems V's Topic Commenter module. Participant also had to fill out a number of short evaluation forms after each set of assignments. These questionnaires were collected and entered in GroupSystems V's Vote module. In total, the participants made 95 contributions. The results of the prototype questionnaires revealed that the case module was liked slightly better than the project module: 7.3 against 6.9 (standard deviation between 0.5 and 0.8) on a scale from 1 to 10. The stakeholders also filled in a questionnaire with respect to the session itself. From the results of this questionnaire we found that the participants felt that the PROMIS prototype was adequately introduced to the stakeholders: 4.3 (standard deviation 0.7) on a scale from 1 to 5, 1 being extremely negative. When asked whether they had a good overview of the prototype's function after the session, the stakeholders responded an average of 3.9 (standard deviation 0.7) on the same scale. From these results, we deduced that the overall opinion of the stakeholders with respect to the PROMIS prototype was altogether positive.

Currently, the PROMIS prototype is being adapted according to the feedback of the stakeholders. The goals of this exercise is to build a version of the prototype that can be employed in a pilot period, which is planned to start in November 1995. During the test sessions of the prototype, several units and supporting departments volunteered to participate in a pilot period. For the introduction of the pilot version of the prototype, another couple of GroupSystems V sessions is planned for the stakeholders that will participate in the pilot.

9.7 Reflection

The situation at the CID with respect to the execution of projects represented an excellent opportunity to apply and evaluate the design approach described in Chapter 8. There were a number of serious problems with respect to the coordination of the organization's processes. Furthermore, there was a need to develop IT support to facilitate the coordination processes in the new situation. In this concluding section, we will reflect on our final case study with respect to the case situation itself and to the design approach that we employed.

9.7.1 The CID

As of this writing, the CID is in the process of entering a pilot with the PROMIS and the new processes. Three units and three supporting specialisms of the Tactical Department were found to be volunteering to participate in this pilot.

As we mentioned in the previous section, we also designed change alternatives with respect to the preparation of projects and employment of supporting actors. These will

be part of the pilot period as well. At this point, we briefly summarize the significant characteristics of the two alternatives. These include for the preparation of projects:

- The chief of a unit identifies the moment that the preparation of a project should start, considering the planned conclusion of his unit's current project. He contacts the Information Department.
- The chief of a unit is involved in the project preparation by planning the project execution in cooperation with the information team.
- The chief of a unit determines the goals and subgoals of the project together with the information team.

The characteristics of the alternative concerning the supporting departments include:

- The appropriate subdepartments are contacted as soon as possible when the (planned) need for support has been identified.
- The CID management appoints a priority to a request for support in cooperation with the unit.
- The chief of the supporting department determines whether to grant a request or not based on a number of criteria that are determined by the CID management.
- All requests, granted or not, are registered in the PROMIS.

Besides the new coordination processes that were designed during our study, the CID management is still interested in the possibility to implement the 'deployment agency structure' that we described in Section 9.2. The CID management believes that with this network type organizational structure, the CID would be better capable of to cope with changing workloads during projects. It is, for example, known that the workload during a project varies with the project phase [Meel 1994]. The advantages of the network structure would be that (1) resources can be employed more efficiently, and that (2) the organization is more flexible to adequately and promptly handle calamities. However, a network structure would also introduce some potential disadvantages. First, there is a risk for an abundance of coordination meetings between CID management, unit chiefs, and the chiefs of supporting departments. Second, it is unclear how the fact that people may only work on a project for a short period of time and then move on, will affect the 'team spirit' in the units. Third, every CID employee that was involved in a project with respect to collecting evidence, remains responsible for this part of the project. This may be cumbersome when the project has to be concluded and all participating employees have to be involved again. In this respect it may, therefore, be wise for the CID to first gain experience with the new (coordination) processes and collect information to get a more thorough insight in the execution of projects and the requirements on the supporting departments. Based on the experiences and insights, the possibility of introducing the 'deployment agency' structure can then be evaluated, for example, by constructing a simulation model.

The PROMIS is currently being prepared to enter the pilot period. To this end, a number of modifications are made, based on the results of the stakeholders' evaluation of the system. Furthermore, an interface will be constructed with two other systems that will be implemented in the CID. The first system concerns an hour registration system. The goal of this system is to register *all* working hours of *all* CID employees. The system is directly linked to the salary administration. It should be noted that the goal of this system is fundamentally different from the PROMIS' goal. The PROMIS aims to register all hours spent on a particular project, regardless of which individual makes these hours. The hour registration system aims to register all hours worked by a particular individual in order to determine or check his or her pay check. Nevertheless, it is clear that registered hours can be imported from the hour registration system into the PROMIS as long as a project code is entered for project hours. The other system concerns a financial information system. The goal of this system is to support all CID employees that have responsibility for a budget, such as the CID management and the unit chiefs. The PROMIS will import budget information on particular projects and cases, so that those responsible can also use financial information to coordinate the course of a project.

9.7.2 The design approach

In this section we reflect on the design approach that we applied in our fourth case study. We first look at the design process. Then we discuss the group sessions that were organized during the study. We conclude with an evaluation of the group technology that we used to support the group sessions.

Process

With respect to the design process, there are four issues that we want to highlight. First, with respect to the *conceptualization* of the CID's current situation, we followed an incremental modelling strategy. The conceptual model was constructed in four steps:

1. *Interviews.* First, a number of short interviews were held with some key actors in the CID, especially in the departments that are involved in the execution of projects. During these interviews, information was gathered on the relevant processes in the CID.
2. *First version.* Second, using the information from the interviews, a first version of the conceptual model consisting of process and actor models was build together with a key CID employee. This activity was supported with TeamGraphics.
3. *Group session.* Third, a group of CID employees evaluated the first version during a session using TeamGraphics.
4. *Second version.* Finally, based on the feedback during the group session, the model was modified and extended afterwards with object definitions and an interaction model.

As can be seen, both versions of the conceptual model were constructed in a participative fashion. We had several reasons to believe that constructing the conceptual model participatively would be beneficial:

1. *Implicit verification.* Since there were no objective criteria to test if the model was a correct representation of the CID's processes, we had to depend on the opinion of the CID's employees. By involving several stakeholders of the CID in the modelling process, we assumed that the model resulting from this process would be implicitly verified.
2. *More efficient.* We expected a group modelling process to be more efficient than the process of building a model from information gathered in detailed individual interviews. Interviews may deliver incomplete or contradicting information. This would force us to go back to the actors involved one or more times and attune the different views.
3. *Attuned mental models.* It was important for the CID actors involved to have a shared view about the situation and the problems that they face. In other words, their mental models of the problem situation needed to be in harmony. This is important to arrive at a diagnosis of the problem situation and a proposed solution that everybody agrees upon [Akkermans et al. 1993]. We assumed that by discussing and improving the initial conceptual model as a group, the mental models of the group members would converge.

However, we also anticipated a major drawback with respect to building the conceptual model participatively: we expected the stakeholders to have a tendency to overemphasize details with respect to their processes. Being as much involved in these processes as they are, it is only natural that they would want to see a perfect representation of their work situation. However, experience learns that a model that describes a certain situation perfectly, including every exceptional event that may occur, either is impossible to create or costs a disproportionate amount of modelling effort [Sol 1982]. It was possible to test whether this drawback happened during the session. If it occurred, it would show in the size and focus of the resulting second version of the conceptual model. A full representation of the current situation including several exceptional events, would require an unbalanced or exponentially increased number of modelling elements.

The results of the quantitative analysis of the two modelling efforts are depicted in table 9.6. The number of tasks and decisions in each version of the model have been counted for each of the (sub)processes described by the model. The results of this activity are presented in the columns '1st Version' and '2nd Version'. We also kept record of the way in which the 1st version of the model changed into the second version. The columns 'Tasks' and 'Decisions' depict how many tasks or decisions were newly added (N), how many were deleted (D), and how many were changed (C).

Process	1st Version		2nd Version		Tasks			Decisions		
	Tasks	Dec.	Tasks	Dec.	N	D	C	N	D	C
Combat Organized Crime	8	1	11	2	3	-	-	1	-	-
Build information position	4	2	9	0	5	-	-	-	2	-
Sustain ACOST	4	2	3	2	-	1	-	-	-	-
Process information	6	0	5	0	-	1	-	-	-	-
Prepare project	7	3	6	3	-	1	-	1	1	-
Plan project	6	2	6	2	1	1	2	1	1	-
Execute project	8	2	11	3	3	-	2	1	-	-
Gain insight in target	5	3	7	4	2	-	-	1	-	1
Initiate final operation	3	1	5	2	2	-	-	1	-	-
Appoint project to taskforce	6	0	6	0	-	-	-	-	-	-
Draw up annual plan	4	1	6	1	2	-	-	-	-	-
Request tactical assistance	9	1	9	2	2	2	-	1	-	1
Request observation assistance	7	2	7	2	-	-	1	-	-	1
Request homicide assistance	3	1	4	1	1	-	1	-	-	-
Total	80	21	95	24	21	6	6	7	4	3

table 9.6 Number of tasks and decisions in the two versions of the conceptual model.

When a task or decision was changed, it implied that its description was modified, but its nature remained the same. (Note that this analysis concerns the complete conceptual model and not only the condensed version represented in Section 9.3).

There are a number of observations we can make with respect to the data depicted in table 9.6 and the session process:

1. The total size of the first version of the model in terms of tasks and decisions equalled 101. The second version counted 119 elements. This implies that the model size increased by about 18 percent. We feel that this finding does not support our initial assumption that the model size would increase dramatically and exponentially when the model would be presented to the group of stakeholders. On the contrary, the stakeholders did not refrain from deleting elements from the model. In total, they deleted 10 elements, equalling 10 percent of the total number of elements of the model's first version.
2. The modelling formalism, task structures, proved to be easy to understand and use. After a short introduction, the stakeholders did not have any difficulties understanding the first version of the conceptual model. As is apparent from the results in table 9.6, the stakeholders did not avoid any discussion during the session. Only one subprocess was found correct immediately (Appoint project to taskforce). Moreover, after a short while the stakeholders started using the task structure concepts themselves to draw outlines for improvements of parts of the model.
3. The first version of the conceptual model was discussed and evaluated within three and a half hours. During this period, a great majority of the stakeholders'

suggestions to modify and expand the model had been processed. In this sense, we feel that the session was very efficient, compared to the situation in which a number of individual interviews with the stakeholders have to be conducted to check and improve the model.

4. All stakeholders agreed at the end of the session that the second version of the model was a valid representation of the CID's processes. In addition, the CID management requested that this model would also be used as a basis in other studies related to the reorganization of the CID. This fact stresses the acceptance of the model.

Second, the *empirical model* of the CID's current situation was also constructed in a number of participative steps representing an incremental modelling strategy:

1. *First problem list.* Based on the first version of the conceptual model and on our interviews during the conceptualization activity, we identified a number of problems. These problems were listed and presented to the session's participants during the third step.
2. *Organizational requirements identification and prioritization.* During the same session in which the conceptual model was evaluated and modified, the CID stakeholders also identified and prioritized a number of organizational requirements using GroupSystems V's Idea Organization and Vote modules.
3. *Problem identification.* Next, the group was invited to brainstorm about possible problems that hindered the CID from meeting its organizational requirements. The way in which this was done will be discussed below in more detail.
4. *Problem organization.* Finally, we converted the results of the session (concerning both the conceptual and empirical model) into a report, describing the current situation at the CID and the most important problems [Vreede et al. 1995d].

For the third step, Problem identification, we used GroupSystems V's Group Outliner module in combination with TeamGraphics. Every stakeholder had access to two adjoined terminals, one running TeamGraphics, the other one Group Outliner. The stakeholders could use TeamGraphics to 'walk through' the conceptual model during the brainstorming exercise. In Group Outliner we outlined the hierarchical structure of the conceptual model's process models. The stakeholders were asked to enter the problems they perceived as comments to the most appropriate part of the outline. This implies that very specific problems were entered as comments to the task they related to, whereas the most general problem statements were entered at the top level of the outline. The outline representing the hierarchical structure of the conceptual model's process models is depicted in figure 9.16.

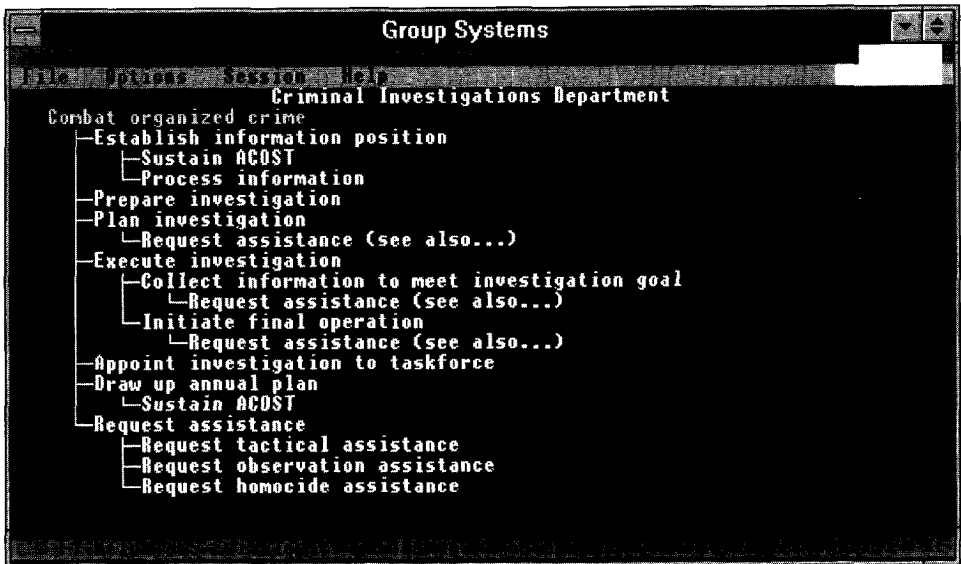


figure 9.16 Outline representing the hierarchical structure of the process models.

In order to stimulate the brainstorming process, we entered our initial list of problems to the outline before the stakeholders could enter their ideas. The brainstorming exercise lasted about half an hour. We expected two types of behavior from the stakeholders during this exercise. First, we considered our initial list of problems to be reasonably accurate and complete. Therefore, we did not expect the stakeholders to enter many new problems. Second, the problems in our initial list were formulated in a very direct manner. We expected that this would invite some of the stakeholders, that did not agree with our perception of the situation, to start a discussion in Group Outliner.

The quantitative results of the brainstorming process in terms of the number of problems generated, are summarized in table 9.7. It has to be noted that the stakeholders entered some problems more than once. We did not exclude 'double' problems from the count, for we feel that they conveyed an important implicit message: the more stakeholders submitted the same problem, the more serious the group seemed to perceive this problem.

Number of problems entered by interviewers	38
Number of problems entered by CID participants	71
Number of problems entered by both	9
Total number of problems	100

table 9.7 Number of problems generated during the session.

There are two observations that we can make with respect to the data presented in table 9.7 and the way in which the session went by:

1. There is only a small overlap between the problems identified by the interviewers and those identified by the stakeholders (9 problems in total). In other words, the stakeholders had a very significant contribution to the outcome of the analysis of the CID's current situation. This outcome did not meet our *first* expectation with respect to the completeness of our initial list of problems. The little overlap may have two explanations. On the one hand, obviously, the stakeholders could notice that certain problems were already identified so there was no need to duplicate them. On the other hand, however, we also noticed a clear distinction in the type of problems identified by the two parties. The analysts tended to focus more on strategy problems that were formulated in more general terms. The problems identified by the CID stakeholders had a strong operational focus and were described very specifically, sometimes including examples. Clearly, these two focuses complemented each other well. With respect to our *second* expectation concerning the amount of stakeholders' agreement with our problem perception, it turned out that the stakeholders actually did *not* question our perception. Stakeholders' feedback after the session revealed that they agreed with our initial list of problems, but they felt it just lacked many of the operational problems that they encountered each day. Hence, their focus during the session.
2. We feel that identifying the problems of CID's current situation in a *group setting* was beneficial to our study. Clearly, when looking at the number and the nature of the problems that were added by the stakeholders to our initial list, we got a much richer picture of the problem situation. The CID stakeholders obviously outperformed the analysts in terms of the number of problems generated. In addition, during the session the stakeholders had a chance to react on other stakeholders' problem statements immediately. This fuelled the discussion about some problems resulting in clearer description and perception on these problems. In addition, this saved the analysts time to validate our perception of the problems with each individual stakeholder.

Third, after being conceptualized, the *models for change* were constructed as simulation models in the simulation environment Arena. Also, the animation facilities of Arena were used to portray the dynamic behavior of the simulation models for change. It was expected that this way, it would be easier to communicate the alternatives to the stakeholders and have them critically evaluate them. However, in a previous study, it was observed that stakeholders seemed to be overwhelmed by animation models and become interested viewers instead of critical evaluators. This is referred to as the 'pacman effect' [Meel 1994]. We anticipated this risk by keeping the models as simple as possible and inviting and stimulating the stakeholders to a critical brainstorming exercise in GroupSystems V.

During the process, we indeed witnessed an initial short period in which the stakeholders seemed a bit awed by the model. However, as soon as they had the opportunity to react to the model and enter an (electronic) discussion, their passiveness made way for a critical and assertive attitude. This is illustrated by the considerable amount of comments that were submitted during the two sessions in which the models were presented, see table 9.8. (1st = first session, 2nd = second session, Σ = total).

Topic	Model for agenda process			Model for preparation process			Model for planning, execution, evaluation			Model for support requests			Total
	1 st	2 nd	Σ	1 st	2 nd	Σ	1 st	2 nd	Σ	1 st	2 nd	Σ	Σ^+
Problems	2	15	17	6	10	16	20	13	33	13	10	23	89
Alternatives	1	7	8	7	17	24	11	8	19	9	5	14	65
Questions	3	2	5	5	5	10	6	5	11	5	1	6	32
PROMIS	-	15	15	7	10	17	13	8	21	9	3	12	65
Other	27	10	37	29	9	38	13	28	41	14	18	32	148
Total	33	49	82	54	51	105	63	62	125	50	37	87	399

table 9.8 Number of contributions during the evaluation of the models for change.

In total, the 20 stakeholders participating generated nearly 60 pages of feedback with respect to the five discussion topics: expected problems with the change alternative, alternative solutions, questions about the change alternative, remarks and requirements regarding the PROMIS, and other comments and ideas. In this sense, we can not conclude that using animated models in a GSS environment harmed the active involvement and evaluation efforts of the stakeholders.

Finally, the fact that we chose to *implement the simulation models of the change alternatives in Arena* did not imply that we could not follow the transformation procedure described in Chapter 3 to translate a conceptual model into simulation code. Being a successor of SIMAN/Cinema, Arena still supports the statements described in Chapter 3. The major difference is that statements are no longer edited in a text file in a word processor. With Arena models are constructed graphically using icons that represent (groups of) statements.

Group sessions

Similar to the previous case study, we handed out questionnaires after each group session. Below, we summarize the questionnaire findings with respect to the group sessions. The next section will focus on the group technology that was used.

On the whole, the group sessions left a positive impression with the participants. The results of the questionnaire are presented in table 9.9. It should be pointed out that in the original (Dutch) questionnaire, not all questions were formulated with a 'positive'

orientation. For the sake of simplicity, we translated all questions in table 9.9 (and table 9.10) into positive oriented questions.

Question	Mean*	Std
1 Are you satisfied with the session?	4.4	0.5
2 Did the session results confirm your ideas on the session topic?	3.9	0.9
3 Did the session help to gain new insights on the session topic?	3.9	0.6
4 Has enough agreement been achieved among the participants?	4.3	0.8
5 Did you enjoy the collaboration with the other group members?	4.4	0.6
6 Do you approve of the composition of the group?	3.8	1.1
7 Are you willing to participate in a follow-up session or activity?	4.7	0.4

* On a five point scale, 1 being very negative and 5 being very positive.

table 9.9 Questionnaire results, part I.

Although the group of CID actors that participated in the sessions differs considerably from the actors participating in the previous case study, the questionnaire results roughly give the same overall picture. From the results in table 9.9 it follows that the stakeholders were very satisfied with the sessions, see question 1. They felt comfortable with the way in which the session was structured and the issues that were addressed. Furthermore, they subscribed to the results of the session, see question 2, while at the same time they thought that the session helped them to gain new insights, see question 3.

A significant difference with the results from the previous case study concerns the amount of agreement. In this second study, the participants felt a lot of consensus was reached, see question 4: 4.3 compared to 3.8 in the previous study. Notwithstanding the fact that in the 'models for change' session there were some energetic discussions of various changes alternatives between individual actors, the participants obviously enjoyed working together during the sessions, see question 5.

The composition of the group was evaluated lower than in the previous study: 3.8 against 4.2. Based on comments from a number of participants, we are convinced that this is due to the fact that during one of the sessions, a lot of people canceled. Consequently, the remaining participants felt that the group was incomplete. Finally, the participants' overall positive attitude toward the sessions is expressed by the results of question 7. They are very willing to attend future sessions.

GSS technology

During the group sessions, we used two types of group support technology. The first type concerns *TeamGraphics*. We did not hand out questionnaires with respect to *TeamGraphics* because the group of people actively using it was too small. Only four people were involved in the creation of the first version of the conceptual model: three project members and one CID representative. After getting used to the technology and

the process of actually working together on the same model at the same time, they felt comfortable using it.

During the evaluation of the first version of the conceptual model, we did not allow the stakeholders to use TeamGraphics themselves for editing the model. The main reason for this was the significant amount of time we expected it to take to train the stakeholders to get sufficiently acquainted with the technology. Instead, we used a dedicated chauffeur to translate all stakeholders comments and suggestions into modifications and additions to the model. In this sense, we used TeamGraphics as a facilitator GSS [Watson et al. 1994]. The stakeholders' reactions during the evaluation of the first version of the conceptual model to this way of working were very positive.

At this moment TeamGraphics only supports drawing the *constructs* of certain diagramming techniques. It would be a significant improvement if TeamGraphics would also support the *semantics* of diagramming techniques. This would allow modellers to check their models for inconsistencies.

The use of *GroupSystems V* was evaluated using questionnaires. The stakeholders involved in the sessions gave the same positive feedback as in the third case study. They liked the ability of *GroupSystems V* to focus the discussion on particular topic and to work simultaneously and anonymously. Furthermore, the productivity that can be achieved using *GroupSystems V* has been illustrated above during the discussion of the design process. The results of the questionnaire question with respect to *GroupSystems V* are presented in table 9.10.

Question	Mean*	Std
8 How do you perceive the quantity of the session results?	4.2	0.9
9 Do you perceive a GSS meeting's productivity higher than a manual meeting?	4.7	0.4
10 How do you perceive the quality of the session results?	3.9	1.0
11 How well did you understand <i>GroupSystems V</i> ?	4.3	0.7
12 Did you find <i>GroupSystems V</i> user friendly?	4.7	0.4
13 Did you find <i>GroupSystems V</i> interesting to use?	4.5	0.4

* On a five point scale, 1 being very negative and 5 being very positive.

table 9.10 Questionnaire results, part II.

The stakeholders felt that the quantity of session results was high, see question 8. The stakeholders clearly attributed this high productivity to *GroupSystems V*, see question 9. One of the participants phrased it as follows: '*It's more productive because there's less yackety-yack, little distraction and no endless discussions. You are forced to get to the point*'. When asked about the quality of the sessions' results, see question 10, the participants still displayed a positive impression. However, this could have been more positive if the prototype sessions had received a better evaluation for this question. The average response for these sessions in terms of quality was 3.6

(standard deviation 0.5), compared to 4.2 and 4.1 (standard deviation 0.4. and 0.6 respectively) for the other sessions. This may be attributed to the fact that the prototyping session left fewer tangible results for the participants. Although we welcomed their evaluation feedback of the prototype very much, this information was not directly useful to the participants themselves.

Like in the previous study, the participants expressed great enthusiasm about the particular GSS used, GroupSystems V. They felt they understood the technology very well, see question 11, and found it very user friendly and interesting to use, see questions 12 and 13.

10 Epilogue

'You see, I went on with this research just the way it led me. That is the only way I ever heard of research going. I asked a question, devised some method of getting an answer, and got - a fresh question.'

- H.G. Wells

10.1 Introduction

In Chapter 1 we formulated the research question of this dissertation: *How can we support organizations in the process of improving their coordination and their supporting information technology?* In order to answer this question we defined the following research objective:

Formulate a design approach for facilitating the process of improving an organization's coordination and its supporting information technology.

During our research, we pursued this objective by developing, applying, and evaluating the intended design approach. In this concluding chapter, we reflect on our research according to Checkland's [1985 and 1991] tripartition of a research project in terms of our application area ('A'), our research methodology or approach ('M'), and our intellectual framework, i.e. our design approach ('F'). We conclude this chapter by presenting an agenda for future research.

10.2 The application area

In this research, the arena in which modern organizations have to survive is seen to be characterized by increasing complexity, hostility, and dynamism. The conditions under which organizations have to operate become both stricter and more unstable. Organizational processes are no longer performed by small departments or divisions. Organizational activities and decisions require the close cooperation and collaboration of many actors, that may well operate from dispersed locations. Internal and external pressures require immediate responses. It is in this environment that achieving

organizational coordination, the orchestration of organizational processes and their constituent activities, has become the issue for organizations to survive. To this end, organizations constantly have to reassess the way in which they coordinate their processes and use IT to support coordination. The organizations that we selected for our case situations find themselves in such an environment.

The Department of Unemployment Benefits (DUB) of the Ministry of the Interior was faced (a) with an unacceptable service level, (b) with the prospect of not being able to deal with the expected increase in the demand for its service, and (c) with a forced reduction of its work force.

The Neurology Outpatients' Department (NOD) of Leyenburg Hospital faced external pressures in terms of (a) increasing demand on care, (c) more critical attitudes towards care, (c) forced reduction in the means of providing care, and (d) increased costs of care. In addition, internal pressures in terms of a lack of consensus between medical staff and management staff frustrated the progress of reorganization.

The Criminal Investigations Department (CID) of the Amsterdam Municipal Police Force is faced with the increasing complexity and hostility of organized crime. Politicians demand a solution for this problem area and are very critical about the way in which the CID implements and executes its new strategy to combat organized crime. Internal pressures concerned (a) the uncertainties during the CID's transition process to its new strategy and structure, and (b) the way in which personnel and other means were rearranged to prepare and execute projects on organized crime.

In order to take a closer look at the coordination issues in our case studies, we take into account the various *interdependencies* between activities (after [Thompson 1967; Malone and Crowston 1994]) and the *mechanisms* to coordinate these interdependencies (after the taxonomy presented in table 1.1), that we introduced in Chapter 1. In table 10.1, we depict which coordination issues were present in each of the case situations that we visited (c.s. = current situation, n.s. = new situation).

Based on the overview in table 10.1, we can make a number of observations. First, in our studies we have come across all types of interdependencies between activities. Second, in terms of coordination mechanisms, we only visited situations that were structured according to the hierarchy model, often a combination of a product and a functional hierarchy. There were some network aspects in the third case situation, but these were insignificant. Third, in each case situation standardization was used as a coordination mechanism to manage the interdependencies between the constituent activities of the organizations' processes. The only exception was the fourth case situation where there was a much stronger presence of mutual adjustment and direct supervision coordination mechanisms. It may be that the omnipresence of

Coordination issue	Case 1 (DUB)		Case 2 (NOD)		Case 3 (CID)		Case 4 (CID)	
	C.S.	N.S.	C.S.	N.S.	C.S.	N.S.	C.S.	N.S.
Interdependencies								
● Pooled	x	x	-	-	x	x	x	x
● Sequential	x	x	x	x	x	x	x	x
● Reciprocal	-	-	x	x	x	x	x	x
● Shared resource	x	x	x	x	x	x	x	x
● Producer/Consumer	x	x	x	x	x	x	x	x
● Simultaneity constraints	-	-	x	x	x	x	x	x
● Task/subtask relationships	x	x	x	x	x	x	x	x
Mechanisms								
● Market	-	-	-	-	-	-	-	-
● Hierarchy	x	x	x	x	x	x	x	x
● Network	-	-	-	-	x	x	-	-
● Standardization	x	x	x	x	x	x	x	x
● Mutual adjustment	-	-	x	x	-	x	x	x
● Direct supervision	-	-	-	-	x	x	x	x
● Anticipatory processes	-	-	x	x	-	x	-	x
● Adaptive processes	x	x	x	x	x	x	x	x
● Review processes	-	-	-	-	-	-	-	x

table 10.1 Coordination issues in the four case situations.

standardization can be explained by the fact that organizations feel that standardization is the easiest way to coordinate interdependencies. It requires the single conscious effort of determining a standard, which can then be integrated into processes. If there are few standardization mechanisms, the actors require a significant number of explicit communication activities to attune their activities, as could be seen in the fourth case situation. Finally, it is worth noting that in the current situations of the selected organizations, we did not find any review processes. The organizations did not reflect on the way in which their processes and constituent activities were coordinated. As part of the change alternative in each case situation, we proposed an information system that could offer support for review processes. Only in the fourth case situation we actually build structural review activities into the organizational processes.

The IT aspect of our case situations can be looked at with respect to two issues that we described in Section 1.3: IT's double role and the effects of IT that are hypothesized in the literature. In each case situation, we investigated the application of IT in a new situation: an EDH system in the DUB, an integrated appointment facility in the NOD, and the ACOST and the PROMIS in the CID. If we look at the role of IT in each case, we see that IT mainly serves as a requirement for the proposed organizational changes, see table 10.2. (Obviously, from a design perspective, each of the case situations included *enabling* technology in the form of simulation tools and GSS).

IT application	Primary role
Case 1: An Electronic Document Handling (EDH) system	Enabler
Case 2: An integrated appointment facility	Requirement
Case 3: The Atlas of Criminal Organizations and Societal Trends (ACOST)	Requirement
Case 4: The PROject Management Information System (PROMIS)	Requirement

table 10.2 The role of the IT applications in the case situations.

Only in the first case, IT predominantly served as an *enabling* technology: the EDH system offered the DUB the possibility to restructure and facilitate communication patterns between the departments and hence achieve the desired service level. In the other three case situations, the IT was in essence a *requirement* for the proposed organizational changes to take effect. The NOD would not be able to achieve the improved way of coordinating the patient flows without the integrated appointment facility. The CID would not be able to effectively implement its new strategy for combating organized crime and preparing and executing projects without the support of the ACOST and the PROMIS.

We now take a look at five hypothetical effects with respect to the introduction of IT to support organizational coordination, see also Section 1.3. In table 10.3, we look at the extent to which these effects can be observed in the *future* situation of each organization. It should be noted that the information in table 10.3 and the ensuing discussion is based on our own observations and interpretations of each situation and, where applicable, on the results of the simulation experiments.

Hypothetical IT effect	Case 1 (DUB)	Case 2 (NOD)	Case 3 (CID)	Case 4 (CID)
Reduced coordination costs	✓	✓	✓	✓
Increased amount of coordination activities	x	x	✓	✓
Improved information quality	✓	-	✓	?
Improved information availability	✓	✓	✓	✓
Increased productivity	✓	✓	✓	?

table 10.3 IT effects in the four case situations.

Based on the information in table 10.3, we make a number of observations. First, if we look at coordination costs in terms of time and effort needed to coordinate activities, we see that in each situation the IT application reduces coordination costs. The actors in the case situations have to make fewer efforts to coordinate their activities with those of their colleagues. For example, in the DUB, they no longer have to request information or send information by the Archives but can assume that their colleagues can get the required information from the EDH system. Second, there are mixed results with respect to the amount of coordination. In the DUB and the NOD, the amount of coordination is not likely to increase; a decrease is more likely as the

increased availability of required information makes a lot of explicit communication to coordinate unnecessary. However, in the third case situation, the introduction of the ACOST clearly results in an increased amount of coordination between the actors involved in the preparation of projects. The same is expected in the fourth situation after the implementation of the PROMIS. This increase in the amount of coordination is probably due to the lack of such coordination activities in the CID's situation at the beginning of the studies. Third, with respect to the information quality, we can observe that in the first and third case study this will likely increase. The DUB can build checks into the EDH system and unclear handwriting is no longer an issue. The ACOST prevents the same information from being collected and processed more than once and links between pieces of information ('enrichment') can be more easily identified. However, in the second case situation, information quality is not an issue, whereas in the fourth situation it has yet to be seen if the CID actors involved will enter high quality information into the system. Fourth, it is clear that all four examples of IT enhance the availability of information with which the actors involved have to coordinate their interdependent activities. Finally, an increase in the productivity of the actors who used the IT applications in Cases 1, 2, and 3 is observed. With respect to the fourth case, a potential increase in productivity largely depends on the quality of the information.

To conclude this section, we want to mention two limitations of our research with respect to the selection of our case situations:

1. All case situations represent the service sector. Although this is a reflection on the increasing significance of this sector, see [Morton 1992; Roach 1991], it remains to be seen whether the results obtained from these sectors are also applicable in a manufacturing environment.
2. Each case situation concerned a relatively small group of stakeholders which made the study manageable as it was easier to involve actors in the study. Larger groups would have resulted in a greater complexity of our case situation. However, there are strong indications that the GSS technology that we employed is capable of effectively supporting very large groups, see e.g. [Lancione et al. 1994; Massman 1994].

The discussion in this section illustrates that our selection of case situations fit the general characteristics of our application area and provided us with a rich cross-section of the aspects of coordination and IT applications. Notwithstanding the limitations mentioned above, we conclude that the selected case situations offer an appropriate representation of this area.

10.3 The research approach

Our application area represents a new and complex setting. There was little knowledge with respect to design approaches that focus on organizational coordination and IT to support coordination at the onset of our research. Hence, the nature of our research had to be exploratory, focusing on building a new body of knowledge with respect to designing organizational coordination supported by IT. Therefore, we pursued our research objective by following an *inductive-hypothetic research strategy*. According to this strategy, we performed a number of empirical (case) studies. The empirical experiences from these studies were abstracted and conceptualized into the intended design approach. To collect the empirical data from our case studies, we used two research instruments. *Case study research* was used in the first (constructed) situation, *action research* was employed in the other three situations. In this section, we briefly reflect on our research approach by identifying some points of consideration and a number of strengths.

The points of consideration are twofold. First, a weakness of case study research and action research concerns the general applicability of research findings. Because the findings of such a research are based on experiences in a limited number of practical situations, it is possible that these findings cannot be generalized beyond the context in which they occur. In this light we want to stress that in our research we did not seek generalizability in the *number* of case studies. The goal was not to find statistical significance, but *representative power*. In other words, the extent to which we allege that our findings have significance beyond the context of the case situations, is determined by their 'fit' with the application area. This issue is discussed in further detail in Section 10.2.

Second, it is typical of action research that during a series of individual studies there occurs a *rethinking* or *extension* of the 'A', 'M', and 'F' of the research [Checkland 1991]. As a result, the body of knowledge that grows out of such a series of studies may shift focus with respect to various aspects or issues related to the research question and objective. It is the explicit declaration of the 'framework of ideas' that makes this dynamic focus, and subsequent integration of research findings possible and manageable. In our research, a synergetic framework bringing together the issues that we focussed on in our case studies, is provided by our notion of a design approach: a set of instruments embedded in a way of thinking, a way of working, a way of modelling, and a way of controlling. The various aspects and issues within this framework are discussed in more detail in Section 10.4.

To conclude this section, we point out what are, in our view, the most important strengths of our research approach. First, our approach promotes a process of *constant learning*. The results from one case study are reflected upon, the lessons are recorded

by making the appropriate modifications in the 'A', 'M', and 'F', and the ensuing enriched design approach is applied and evaluated in a subsequent case study. We further discuss this in Section 10.4. Second, in our role as an interpretivist researcher, we described organizational problem situations while being part of the design processes to bring about the required change. This provided us with a means of extracting *rich detail* and enabled a *realistic understanding* of the situations under investigation. Third, as we described in Chapter 2, case study research and action research are both useful instruments to support theory *building*. As a result, the findings of such a research project can be instrumental in identifying imminent issues within the theory under development and in setting an agenda for further research. In Section 10.5, we discuss the research agenda that we construct upon the findings of our research.

10.4 The design approach

The theme in our dissertation is how to help organizations assess their current situation and enable them to design new (improved) coordination mechanisms and using IT support. Through a process of constant learning in a number of case situations, we built a body of knowledge that resulted in a design approach. We began our research using 'dynamic modelling' as our initial theoretical framework. In our first case study, we found that we required additional modelling techniques for conceptualizing organizational coordination. Therefore, we developed an initial version of the modelling concepts that were discussed in Chapter 3. Furthermore, it appeared that the transformation of the conceptual model into an empirical model was a cumbersome process for non-simulation modelling experts. Therefore, we outlined a procedure to guide such a transformation. With these insights, we entered our second case study. In this study we refined our modelling concepts and our transformation procedure. In addition, we employed animation techniques to stimulate stakeholder involvement in the study. It appeared that the animation model that we built, facilitated and stimulated stakeholder involvement and discussion, but only at a late stage in the study and for a short period of time.

In order to further support stakeholder involvement, we investigated the applicability of group support methods and techniques. Furthermore, in the second case study we also found that statistical empirical data is sometimes difficult, if not impossible, to collect. To this end, we investigated techniques to build an empirical model that does not exclusively rely on the collection of statistical empirical data. These investigations resulted in our second initial framework, with which we performed our third case study. This framework was in effect an extension of the dynamic modelling approach. It constituted of (a) the use of GSS to support interaction with and among stakeholders at various stages during the study, and (b) the concepts of 'organizational requirement' and 'problem description' to construct an empirical model of a current situation. In

addition, during the case study we were faced with the need to develop an IT solution based on the specifications of an organizational change alternative. In response, we employed prototyping techniques and GSS to evaluate the subsequent prototypes.

The third case study taught us four major lessons. First, it appeared to have been more efficient and effective to use the concepts for modelling organizational coordination in a true participative fashion, i.e. together with the stakeholders. Second, the use of 'organizational requirements' and 'problem descriptions' appeared to result in an empirical model that could be used as a basis for diagnosis and for the design of a number of alternatives for organizational change. Third, it turned out that using the modelling concepts alone to communicate change alternatives to stakeholders did not provide true insight. Finally, the experiences with GSS for the various problem solving activities were promising. These lessons encouraged us to introduce joint modelling support for the conceptualization activity and we suggested the mandatory use of simulation and animation techniques for the modelling of change alternatives.

Together, the experiences in the first three case studies helped us to formulate our intended design approach, which we subsequently applied in the fourth case study. This last case study in general illustrated the feasibility of the design approach, both in terms of process and outcome. A number of issues that emerged from the experiences in this study included the design of joint modelling sessions, the update of our transformation procedure, and the evaluation of organizational change alternatives using gaming techniques. These issues are further elaborated upon in the final section of this chapter.

From the experiences in and the reflection on the four case studies, we may conclude that *there is empirical evidence that suggests that our design approach facilitates the process of improving an organization's coordination and using information technology to support it*. In particular, we summarize our main findings as follows:

1. Our design approach enables the analyst to analyze (both quantitatively and qualitatively), diagnose, and design various mechanisms for organizational coordination.
2. Our design approach structures the (group) processes for the collection of data that is required for the construction and evaluation of both models for understanding and design.
3. Our design approach facilitates and encourages the active involvement of stakeholders in the model building and evaluation activities.
4. Our design approach ensures the specification of organizational change alternatives that provide a basis on which prototypes of IT applications to support organizational coordination may be successfully developed.

Evidently, certain elements of our design approach contribute to each of these findings. In the above discussion of our case studies, we indicated that throughout our research, we investigated certain issues. These contributed to the formulation of several elements of our design approach, that mainly related to the way of working and to the way of modelling:

- *Way of working:*
 - The transformation procedure.
 - The process of problem solving.
- *Way of modelling:*
 - Modelling concepts.
 - Concepts 'organizational requirements' and 'problem description'.
 - Animation models.
 - Group Support Systems.
 - Prototyping techniques.

In the remainder of this section, we discuss a number of these elements separately in more detail. We describe the insights that we gathered and, where appropriate, synthesize these into some practical considerations.

10.4.1 The transformation procedure

The transformation procedure we described in Chapter 3 was developed to structure the transition from a conceptual model to a (simulated) empirical model. This transition was not a trivial activity. The nature of our conceptual model (object oriented) was different from that of the SIMAN simulation language (process oriented). This hindered a one to one transformation.

The transformation procedure was first successfully applied in the NOD case. In the fourth case study, we used the successor of SIMAN, Arena. Arena can still handle the SIMAN simulation language, but models are constructed in a different fashion, namely graphically. Consequently, we did not have to execute the transformation procedure to the extent that we had to write textual code. Only the 'logic' of the transformation process needed to be used [Kuijlaars 1995].

In retrospect, we argue that the transformation procedure facilitated the construction of the simulation models. Also, we argue that models built according to this procedure are easy to maintain and can be understood effortlessly and modified by another model builder who is familiar with the concepts and transformation procedure.

10.4.2 The process of problem solving

The problem solving process that we presented in Chapter 8 using a number of process models, represents the backbone of our design approach. Although in general terms it is based on well-known problem solving literature (see Chapter 3), the detailed decomposition of this process is derived from the needs and experiences of our case studies. If we look back at the way in which the problem solving process was carried out, we come to three general conclusions:

1. The problem solving process should not be seen as a cookbook or a recipe that just needs to be followed. The individual activities of the process and the order in which they were carried out, was different in each case situation. In this sense, the problem solving process represents a true '*conceptual approach*'; the exact specification depends on the particular situation in which it is employed. This means, for example, that different strategies can be followed. In the first case, we built the conceptual model in a linear fashion: first object definitions, then each of the aspect models in turn. In the fourth case, we first built a rough version of some process models, had these evaluated and corrected by a number of stakeholders, and finally constructed and evaluated the other aspect models and object definitions.
2. In the description of the process in Chapter 8, we only refer to two actors: the analyst and the stakeholders. In reality, these actors actually represent roles that can be performed by many different individuals. In the last three case studies, almost a 100 people participated during various activities (mainly in the role of stakeholder). In retrospect, we can say that it requires careful planning and selection to get together a group that is both manageable and gives a good representation of their organization.
3. To successfully employ our activities in the problem solving process, we contend that an analyst should not only possess expertise with respect to some basic interviewing techniques and techniques for modelling organizations and IT specifications. In addition, an analyst should be able to handle, direct, and stimulate the group processes that are inextricably bound up with the design activities of this process. Moreover, he or she should be able to appreciate the potential of GSS and wisely pick and choose which GSS technology to use when and for what problem solving activity.

In summary, we conclude that our process of problem solving effectively represents a collection of problem solving activities, that, depending on a particular situation, can be carried out (or not) in a particular order. It depends on the craftsmanship of the analyst(s) to find the optimal blend of problem solving activities, problem solving strategies, actors to be involved, and techniques to be used.

10.4.3 The modelling concepts

In all four case situations, we conceptualized organizational processes and their coordination using a set of modelling concepts consisting of a number of fundamental object classes. We defined three aspect models, different representations of the object classes, that each focus on different coordination aspects. Part of the modelling concepts were based on concepts proposed by Bots [1989] and Dur [1992]. However, we modified their concepts, significantly extended the set of fundamental object classes, introduced the interaction aspect model, and proposed and performed the integrated application of the object classes and aspect models to the problem of designing organizational coordination and supporting IT.

Overviewing the results from the case studies, we contend that using our modelling concepts it was possible to grasp the various coordination issues that we encountered, see also Section 10.2. This is demonstrated by the many examples in the descriptions of our case studies. With respect to the *interdependencies* between activities, we only came across a single difficulty. This concerns Thompson's [1967] reciprocal interdependency, which is comparable to Malone and Crowston's [1994] simultaneity constraint. Our object class definitions assume that an action is completed before an item is passed on. In other words, an item processor cannot send information to another item processor without completing the action that generates this information. Unfortunately, this is exactly what is required for reciprocally interdependent activities: two reciprocally interdependent activities constantly feed each other information. In our case studies, we handled this problem by defining an 'artificial' compound actor that represented all actors performing the actions concerned. In the process model, such actions were depicted with an action icon that included two or more actors. (Note that a compound action was depicted in a similar way).

With respect to the *coordination mechanisms* in table 1.1, the modelling concepts could handle all mechanisms. First, we argue that the concepts could cope with all three structural mechanisms, although we only encountered hierarchy structures. After all, the structure *itself* is not subject to modelling, only the arrangement of the organization's activities that result from the structure. Second, standardization mechanisms do not require any additional modelling efforts. Only the way in which the processes and their constituent activities are actually carried out needs to be modelled. Mutual adjustment and direct supervision can explicitly be modelled as reciprocally or sequentially interdependent activities, which have been discussed above. Finally, anticipatory and review coordination processes are directly modelled by defining planning or review activities in the model. Adaptive coordination processes can be represented by explicit communication tasks and decisions that are performed to attune activities or direct a course of action.

The way in which we introduced and presented the modelling concepts in this dissertation, only illustrates what the concepts represent. It does not give formalized rules about how they should be applied, or, in other words, whether an analyst or stakeholder should model someone, for example, as a person, as an actor, or as a repository. Based on our cumulative exploratory experiences with the concepts in the case situations, we conclude this subsection by outlining a few practical guidelines for using the modelling concepts:

1. An object in the situation under investigation should be modelled as an item if:
 - It does not display behavior. It is passive.
 - The behavior of the object is irrelevant for the analyst. It falls outside the boundaries of the problem situation.
2. An object should be modelled as a repository if its *only* behavior is 'holding' items for a certain period of time. It does not alter any of the item's attributes. It just stores them somewhere until they need to be retrieved.
3. An object should be modelled as a link if:
 - Its *only* behavior is transporting items from one node to the other.
 - A more detailed analysis of the transporting behavior is not of interest to the analyst.
4. In all other cases, an object should be modelled as an actor.

10.4.4 The use of animation models

In our design approach, we suggest the application of animation models for two purposes:

1. To support the analyst in building a correct model of the problem situation.
2. To facilitate the communication between and among stakeholders and analysts in order to involve the stakeholders in the study.

In previous studies on dynamic modelling, the use of animation models was briefly touched upon by stating that it could provide insight into a problem situation [Dur 1992; Meel 1994; Streng 1993; Wierda 1991] but could also distract the stakeholders [Meel 1994]. However, the added value of animation techniques was never the subject of a critical and conscious investigation into a series of case studies.

In our case situations, we investigated the use of animation models in all but the third case study. Based on our experiences, we can make a number of general observations. First, without hesitation, we can argue that the use of animation techniques was instrumental in developing (and debugging) correct simulation models. In all case situations, the animation models helped us to identify a great number of modelling errors that would have been difficult to discover from output data or run traces.

Second, the experiences in the second and fourth study clearly illustrated the potential of animation models as a communication vehicle. The animation models stimulated the stakeholders from different (horizontal and vertical) positions in their organization to take some distance from their own situation, look beyond it, and critically discuss the overall work situation with their colleagues. We did not observe this in the third case study where only aspect models of the current and future situation were presented to the stakeholders. In this light, it is notable that during the fourth study, one of the stakeholders commented during the discussion of a change alternative: *'You know, this animation tells me more than ten of these process models.'* Furthermore, we observed in the fourth study that using animation in combination with group support technology did not lead the attention of stakeholders away, but, on the contrary, greatly encouraged them to seriously react to the animated change alternatives.

Finally, a critical observation can be made as well. The way in which we originally employed the animation model to facilitate stakeholder involvement did not prove to be entirely successful: the animation model was not shown before the empirical model had been completed. Hence, it could not stimulate the involvement of stakeholders during the preceding modelling activities. In addition, the animation models only encouraged stakeholder involvement during a short period of time, namely in the meetings in which the models were presented.

Based on these observations, we can formulate a number of practical guidelines with respect to using animation techniques to facilitate stakeholder involvement. First, we contend that animation models should be built as *early* in the process of creating an empirical model as possible. The collection of empirical data does not have to be completed before an animation model is built and demonstrated. This not only may prove beneficial because stakeholders can point out modelling errors in the conceptual model, but also because it can stimulate stakeholders' commitment to participate during the later phases of the design process. Moreover, it may motivate stakeholders to give (expert) estimations for certain pieces of empirical data that still need to be included in the model. Second, in situations where *little statistical empirical data* is available, we argue that animation models are especially useful. We found that in these situations, an animation model that displays logically correct behavior (structurally valid) is an invaluable instrument. It provides stakeholders with an inviting mirror of their own situation that they can reflect upon, and it creates insight into future situations that are considered for implementation. Finally, an analyst should find the *balance* between designing a schematic '*bare*' animation model and a *fancy* model with lots of detail. Our experience suggests that a simple model with less detail is easier to comprehend and react to than a model that tries to display a one to one correspondence to the modelled situation.

10.4.5 The use of Group Support Systems

The application of GSS in our case studies aimed to facilitate active stakeholder involvement in the analysis, diagnosis, and design activities of our approach. To this end, we successfully used GSS to support various group activities. Examples include but are not restricted to: conceptual model design and evaluation, empirical data collection, organizational requirement identification and prioritization, problem formulation, change alternative design and evaluation, user requirements specification, prototype presentation and evaluation, prototype data collection, and the real life simulation of work procedures. The variety of these activities demonstrates the wide applicability of GSS to support group design activities.

Although various sources stress the importance of conducting more studies in natural settings [DeSanctis and Gallupe 1987; Pervan 1994; Zigurs 1993], not more than 15 percent of the studies into GSS concern case studies or action research [Pervan 1994]. Most of these field studies originate from a limited number of organizations, the most predominant being IBM, see e.g. [Martz et al. 1992], the IRS, see e.g. [DeSanctis et al. 1991], Texaco, see e.g. [Vician et al. 1992], and the US Department of Defense, see e.g. [Dennis et al. 1994]. It is noteworthy that at the beginning of our third case study we found no published field study which dealt with the use of GSS to support organization design and IT development. An exception was the work by Wanninger and Dickson [1992a; 1992b], but they only examined the application of GSS during the information requirements specification phase. This implied that we might have been able to begin theorizing about the application of GSS in real life settings but there was as yet little substantial evidence of their practical value. Furthermore, it meant that we had to find out in what way we could effectively and efficiently support the various group activities of our design approach using GSS.

During our research we collected a great deal of insightful data on the application of GSS to support and facilitate the process of organizational change. By collecting data from various sources, we were able to obtain a rich picture of the application of GSS. It can be said that our findings in general did not contradict the results reported by Wanninger and Dickson and other field studies using GSS which considered anonymity, efficiency, productivity, and user satisfaction, see [Dennis and Gallupe 1993]. Below, we discuss a number of aspects of using GSS in more detail. We structure this discussion according to the 'model' of GSS experiences that we presented in Chapter 6, consisting of GSS attributes and GSS setting attributes. The GSS attributes include:

- *Anonymity.* An advantage of GSS often referenced in the literature concerns anonymity. Although we did not ask participants to rate the usefulness of anonymity, their reactions to open ended questions and interviews were very

supportive. Because ideas and votes could be submitted anonymously, the participants felt that silent colleagues were encouraged to take part in the meetings and generate ideas. It also prevented or hindered certain participants from dominating a meeting due to personality or position. In hierarchical organizations like the police force this is a substantial risk. As one of the participants phrased it, *'I felt that being able to communicate anonymously utilized the potential of silent participants and diminished the polarizing effect of the more aggressive members'*. In this sense, our experiences support the findings in other field studies with respect to anonymity, see [Dennis and Gallupe 1993].

- *Parallelism.* Results from lab experiments and field studies suggest that GSS meetings are more efficient and productive than non-GSS meetings [Dennis and Gallupe 1993]. This advantage to productivity is usually attributed to the fact that participants in an electronic meeting can work in parallel. Our experiences from the case studies support the assumption that GSS meetings are more productive. This is the predominant opinion of the participants in our studies. They replied to the proposition that 'GSS meetings are more productive than manual meetings' with 4.7 on a scale from 1 to 5 (standard deviation 0.4). Second, the analysis of several electronic meeting logs gives some clear examples: A list of over 70 serious problem areas was created in only 20 minutes. 60 pages of feedback (399 comments) were generated in just over three hours of brainstorming.
- *Group memory.* It was not often during the meetings in our studies that groups of stakeholders had to work with the exact results of previous meetings. However, we did experience two clear benefits of having group memory available. First, categorization and modification of ideas could be done effortlessly, because ideas were electronically available and easy to manipulate. Problem areas on the list mentioned above, for example, were discussed, modified and grouped within two hours. Second, meeting reports could be generated quickly. Every meeting during the third and fourth case study was comprehensively documented within a few days.
- *Media richness.* Our observations illustrated that media richness is an issue that should receive careful attention. Certain types of activities require a richer medium than electronic communication alone [McGrath and Hollingshead 1993]. We witnessed this during the creative generation and evaluation of ideas. In our initial meetings, we decided to carry out most activities in a pure electronic fashion. As a result, several participants expressed a concern to allow for sufficient time to discuss intermediate results. We followed these suggestions when planning later meetings. We felt, the participants would tend to be less likely to buy in to the results of the meeting if we did not integrate their suggestions.

We now turn to the GSS setting attributes. We do not discuss the GSS setting attribute 'group culture' as we did not investigate this in any way. (The participants were accustomed to functioning in a Western European work environment):

- *Facilitation.* In retrospect, it appeared that the role of the facilitator was most effective in enabling the participants to focus their discussion and benefit from the GSS technology. This was confirmed by the feedback of the participants. In addition, it can be said that during the *planning* stage of the meetings, the facilitator made a significant contribution, which comprised designing each meeting in terms of activities to be performed and selecting the appropriate GSS tools.
- *Group size.* The size of the groups that participated in the meetings ranged from 6 to 16. If we look at the participants' questionnaire results, we do not see any significant differences between small groups (up to 9 members) and large groups (10 to 16 members) with respect to issues such as meeting satisfaction or the perception of the usefulness, quantity, and quality of the meeting results. In this sense, our experience with respect to group size is not in line with the findings that GSS have a greater positive impact on larger groups in terms of productivity perception and participant satisfaction, see [Dennis 1994]. The only exception that we observed is that small groups, not surprisingly, felt they had less difficulty achieving consensus: on a scale from 1 to 5, 4.1 for small groups (standard deviation 0.9) against 3.3 for large groups (standard deviation 1.2).
- *Group/Individual experience.* With the progress of the studies, the percentage of meeting participants that used a GSS for the first time dropped. As the composition of the groups changed considerably over time, we could not draw conclusions with respect to group experience on a group level. On an individual level, an analysis of the questionnaires does not indicate any significant differences between inexperienced and experienced participants with respect to performance and perceptions of meeting process and outcomes.

To conclude our reflections on the use of GSS, we synthesize three practical guidelines based on our experiences in our last two case situations:

1. *Meeting preparation and design.* Planning for the GSS meetings with identification of specific desired outcomes and mapping the use of the technology into the meetings proved of critical importance. Since the participants spent a considerable amount of time working individually (entering ideas, comments, thoughts, votes etc.), the goal of the meeting's activities and what is expected from them had to be perfectly clear. It is our experience and belief that the meeting task taxonomy and GSS functionality mapping presented in Chapter 6, are of critical value for the preparation of GSS meetings.
2. *Be cautious with voting results.* During the GSS meetings we observed on several occasions a risk of attributing unwarranted accuracy to results, especially during voting activities. For example, when a large number of participants bypassed a voting activity (they felt that they were unqualified to offer an opinion on a particular topic), the results of the vote had to be considered with suspicion. Also, we sometimes encountered situations where the ballot items were not clear to the

participants. The aggregated voting results did not reveal this situation. Hence, careful examination and discussion of the technology's output is critical.

3. *Focused and limited brainstorming.* It is not difficult to generate a great deal of output during brainstorming activities with GSS. However, the real challenge is then to process and organize the output into a useful format, such as a set of categories. To facilitate this activity, we advice a strategy of 'focused and limited brainstorming' consisting of the following steps:
 - Define the brainstorm question as precisely and narrowly as the meeting topic allows it. Try to prevent the participants to wander off in too many directions or give input on too many divergent levels of abstraction.
 - Stop the brainstorming exercise temporarily after each participant has entered one idea. Discuss and organize the list of ideas. This helps the group to focus and adjust to the same abstraction level. It will also disclose misinterpreted brainstorm questions.
 - Use Nominal Group Technique (NGT) if possible. This will allow the facilitator to organize and discuss the brainstorming output in manageable portions.

10.5 Future research

We learned a lot, yet little we know. Our research led us through an interesting and complex application area. Using interpretivist research instruments, we were able to get a rich taste of reality. We developed, applied, evaluated, and modified our framework of ideas to facilitate organizational coordination. In the previous sections we reflected on what we learned. In this concluding section we outline what we still want to know. We identify a number of issues that, we hope, will be subject of future research efforts. The issues are organized according to our 'A', our 'F', and our 'M'.

10.5.1 The application area

It is clear that our application area is both complex and vast. In our research, we could only investigate four examples of organizations within this area. Based on the insights we gathered during the visits to our case studies, we want to raise a few issues that are, in our view, worthy of further investigation.

First, organizations that are structured according to the network or market structure deserve ample research attention. As our world is heading for a 'networked society', the significance of network structures and networking relationships within and between organizations is recognized and will only increase. However, this research area is still in its formative stages [Qureshi 1995]. Issues that emerge are, for example, 'Which social patterns of cooperation and collaboration will surface in the network

organization?', 'How can we design adequate mechanisms to coordinate this cooperation and collaboration?', 'How can we design flexible support for (dispersed) working teams?', and 'How can we support the actors in the network to see beyond their own direct links and understand the overall situation and their position in it?'

Second, it is worth inquiring whether there exist relationships between certain coordination issues and the characteristics of organizations and their environment. For example, is there a correlation between the type of interdependencies that an organization has to coordinate and the mechanisms it employs to this end? Are certain types of standardization more appropriate for particular interdependencies than others? Was it a coincidence that our case situations did not have review processes or can some common set of characteristics be found, also in other organizations, to explain this?

Finally, as stated in Section 10.2, our research did not visit situations in which very large groups of stakeholders had to be involved. Nor did we investigate organizations in the manufacturing sector. These areas are open for further exploration.

10.5.2 The design approach

In our research, we paid attention to various aspects and elements of our design approach. From the experiences with our approach and from the reflections that we made, we can identify a substantial number of future research issues. We restrict the discussion to what are, in our view, the most important and interesting issues.

First, in our last case study we used TeamGraphics to support *joint modelling*. We constructed and evaluated a graphical representation of a conceptual model together with a number of stakeholders. Our experiences in this respect were encouraging. However, there is as yet little practical experience with such tools, while the group dynamics that we experienced in a TeamGraphics session were significantly different from a GroupSystems V session. Research questions that come to mind include:

1. The optimal group size for the type of modelling sessions we described in this paper is worthy of consideration. Previous studies with GSS have shown that larger groups may benefit more from the application of GSS in terms of productivity and satisfaction than smaller groups [Dennis and Gallupe 1993]. However, one might also argue that a joint modelling task becomes too complex to handle when a large number of people are involved at the same time.
2. It is worth investigating if the desire to build a model that is a 100 percent representation of the organization is dependant upon the group setting in which this takes place. What is, for example, the effect of setting attributes such as process moderation, group size, or participants' background?

3. In our fourth study we decided to build the conceptual and empirical models in two steps: first a prototype, that was subsequently improved and completed by the participants. It is worth examining whether this procedure is indeed more efficient or effective than building a model with the whole group from the beginning. Also, what is the effect of this procedure with respect to the '100 percent representation issue'? Will a prototype procedure decrease the desire to build a 100 percent representation?
4. Finally, it is worthwhile to further investigate which modelling techniques and methods can be used in a participative fashion that represent rigor, flexibility, and comprehensiveness for non-modelling experts. We found that task structures are easy to comprehend and use. Other studies have found the same for the IDEF modelling technique [Dennis et al. 1994]. However, not all techniques that can be used to model organizational processes are expected to be equally appropriate for joint modelling sessions, take for example Petri-nets.

Second, further research is needed in the *transformation* of conceptual models into empirical models. In our last case study, we used Arena as a construction language for the empirical models. Although our transformation procedure can be used successfully with Arena, there is another option which seems more efficient. Arena supports the use of *templates*. An Arena template represents a library of user defined modelling constructs. Each of these modelling constructs may consist of a combination of basic Arena constructs. In this sense, a dedicated template may be designed that allows simulation models to be built directly in terms of actors, repositories, messages, products, persons, tasks, and decisions. This template would allow for rapid model generation, while the remaining availability of basic Arena constructs ensures modelling flexibility where required. An operational prototype shows the feasibility of this template, see [Bouwman and Kuijlaars 1995; Eijck 1996].

Third, we introduced and evaluated alternatives for organizational change using animation models. Although these presentations triggered substantial discussion and resulted in a massive amount of feedback, there are two points of consideration. Animation models together with quantitative data may give stakeholders an idea of a future situation, but do not allow them to really experience these situations. In addition, the role and use of new IT applications in the future situation cannot be fully experienced either. Although our prototyping sessions allowed stakeholders to use and evaluate prototypes of IT applications, they lacked the future organizational context. In this light, we suggest that further research is conducted into the use of *gaming techniques* to evaluate change alternatives. During a game, a change alternative may be 'enacted' by stakeholders playing a role. Prototypes of future IT applications can be employed during the game to see them in their proper environment. We propose that a '*game shell*' be developed to facilitate the organization and execution of these

games. Such a game shell can be seen as a special type of group support technology that consists of the following components:

- *Actor base*. This component contains the definitions of all actors or players in the game with respect to their position in the process and their potential behavior. A repository is also seen as an (automated) player.
- *Stimuli/Reactions base*. This component contains a set of stimuli (items) that may be received by the actors in the game. Based on the stimuli, the actors may (automatically) select a number of reactions (also items) from a predefined set.
- *Prototype link*. This component ensures that the prototype of an IT application that will be part of the enacted future situation, can be used during the game in order to issue certain stimuli or to help actors determine their reactions.
- *Game management*. This component manages the execution of the game. It consists of three parts. A *score model* defines the rules of the game and keeps track of the score during the game. An *analysis model* summarizes the (aggregated) behavior of the actors during or after the game. An *evaluation tool* allows the actors to discuss the process and results of the game and the organizational change alternative that it represent.

A game shell consisting of these components could, for example, be implemented in Lotus Notes as this application offers flexible facilities to exchange stimuli and reactions between users (actors) and to keep track of the exchange process. The prototype of the IT application could be linked as an external application. The evaluation tool could be implemented as a link to GroupSystems for Windows. It will be interesting to find out whether this type of group support technology indeed results in a more profound assessment of change alternatives and adequately supports the introduction of the change alternatives during their implementation.

Finally, we made extensive use of *group support systems* (GSS) during our last two case studies. From our discussion of GSS in Chapter 6, it is clear that a lot of research effort has been spent on investigating GSS. Notwithstanding the fact that this research generated a substantial amount of knowledge, we feel that in some cases GSSs represent a *research hype*. The fact that the concept is intuitively attractive, that empirical meeting data is easily collected, and that an almost infinite number of experiments can be designed with variables describing the group, the group process, the group support, and the process outcomes, have brought about a massive amount of GSS research projects. Unfortunately, the majority of these research projects only focus on the application of GSS in controlled laboratory settings with student subjects. We feel that a mature GSS research agenda should pay attention to three different perspectives on GSS:

1. *Application perspective.* From this perspective GSS are seen as a research object. The research objective is to study the impact of GSS on group process and outcomes. As said above, most research projects approach GSS from this perspective. Interesting research issues in this respect refer to longitudinal studies ('How does the impact of GSS change within the same group over time?'), facilitation ('What type of facilitation is most beneficial for which situations?'), and real life validation of experimental findings ('Do real GSS groups and processes display the same behavior as their experimental counterparts?').
2. *Design perspective.* From this perspective GSS are seen as a design object. The research objective is to examine how existing and future organizational processes can be supported with GSS. So far, attention has predominantly been on facilitating existing group processes in organizations. There are few examples where (distributed) GSS are used to actually support new ways in which to engage in cooperative work. Research issues from this perspective include the determination of an organizational paradigm that describes cooperative work and the development of approaches that focus on the design of GSS supported group processes.
3. *Instrumental perspective.* From this perspective GSS are seen as a design instrument. Here, the research objective is to employ GSS as support for participative design efforts. Our last two case studies fall in this category. Research issues within this perspective include the application and evaluation of different research instruments and data collection techniques to measure the added value of GSS and the creation of a framework that facilitates the comparison and contrasting of empirical data from different studies.

10.5.3 The research approach

The issues raised and questions asked in the previous two subsections require the specification of an 'M', or research approach. It is clear that our application area is complex and is growing more complex. Although our body of knowledge is expanding, there are still many issues to investigate that so far have received little attention. To continue research in this area, we argue for *pluralism*.

On the one hand, we suggest continuing along *inductive, exploratory strategies*, using research instruments that allow for a rich description of the phenomenon under investigation. This richness is needed to distill lessons, and identify research questions that require detailed, dedicated attention. Candidate issues are, for example, those mentioned with respect to network organizations, the game shell, and the application of GSS in distributed environments.

On the other hand, as a number of issues become clearer, we suggest to that more *deductive strategies* to extract theories from cumulated experiences be employed. These

theories can subsequently be tested in a number of (controlled) natural or constructed settings, using research instruments such as field experiments, lab experiments, or surveys (which are setting independent). Candidate theories are, for example, theories that relate coordination mechanisms to organizational characteristics and theories related to the research questions with respect to TeamGraphics. It should be noted that previous explorative material that was not used as a source to base such theories on, may now also be employed for testing purposes. We contend that the case material presented in this dissertation has enough richness to be analyzed again, but from other perspectives than we have chosen to do.

In this final section, we outlined the contours of a research programme on designing coordination mechanisms and IT support for modern organizations within a changing environment. Further building and executing this programme represents a challenge for researchers and practitioners alike.

References

'After all is said and done, there's always a lot more said than done.'

- Alfred E. Neuman

- AALST, J.W. VAN, M.R. BOUWMAN, AND B. NOORDUIN, *VSI - Handleiding bij prototype versie 1.0*, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, 1993a. (In Dutch)
- AALST, J.W. VAN, M.R. BOUWMAN, AND B. NOORDUIN, *VSI - Technisch Document bij prototype versie 1.0*, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, 1993b. (In Dutch)
- AALST, J.W. VAN, M.R. BOUWMAN, AND B. NOORDUIN, *VSI - Functionele specificatie bij prototype versie 1.0*, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, 1993c. (In Dutch)
- AALST, W.M.P. VAN DER, *Timed coloured petri nets and their application to logistics*, Doctoral Dissertation, Eindhoven University, Eindhoven, The Netherlands, 1992.
- ACKOFF, R.L., Towards a system of system concepts, *Management Science*, Vol. 17, No. 11, 1971.
- ACKOFF, R.L., *The Art of Problem Solving*. John Wiley & Sons, 1978.
- ACKOFF, R.L., AND F.E. EMERY, *On Purposeful Systems*, Aldine-Atherton, Chicago, 1972.
- AKKERMANS, H., J.A.M. VENNIX, AND E. ROUWETTE, Participative Modelling To Facilitate Organizational Change: A Case Study, in: *Proceedings of the 1993 Systems Dynamics Conference*, Cancun, Mexico, 1993.
- ALAVI, M., Computer-Mediated Collaborative Learning: An Empirical Evaluation, *MIS Quarterly*, Vol. 18, No. 2, pp. 159-174, 1994.
- ALAVI, M., AND P. CARSON, A review of MIS research and disciplinary development, *Journal of Management Information Systems*, Vol. 8, No. 4, pp. 45-62, 1992.
- ALEXANDER, E.R., Interorganizational Coordination: Theory and Practice, *Journal of Planning Literature*, Vol. 7, No. 4, pp. 328-343, 1993.
- ANGELL, I.O., AND S. SMITHSON, *Information Systems Management, Opportunities and Risks*, Mac Millan, London, 1991.
- ANSON, R.G., AND A. HEMINGER, *An Assessment of Process Facilitation in a Group Support System setting*, working paper, Boise State University, Boise, ID, USA, 1991.
- ANSON, R.G., R.P. BOSTROM, J.W. FELLERS, AND L. CHIDAMBARAM, Using Group Support Systems to Facilitate the Research Process, in: Nunamaker, J.F. Jr., and R.H. Sprague (eds), *Proceedings of the 25th Hawaiian International Conference on System Sciences*, IEEE Computer Society Press, Los Alamitos CA, 1992.

- ANSON, R.G., R.P. BOSTROM, AND B. WYNNE, An Experiment Assessing Group Support System and Facilitator Effects on Meeting Outcomes, *Management Science*, in press.
- ANTON, A.I., T.A. GALE, W.M. MCCracken, AND J.J. SHILLING, Object Based Requirements Modeling for Process Continuity, in: Nunamaker, J.F. Jr., and R.H. Sprague (eds), *Proceedings of the 26th Hawaiian International Conference on System Sciences*, pp. 191-202, IEEE Computer Society Press, Los Alamitos CA, 1993.
- ARGYRIS, C., Some Limits of Rational Man Organizational Theory, *Public Administration Review*, pp. 253-267, 1973.
- ARGYRIS, C., Good Communication That Blocks Learning, *Harvard Business Review*, Vol. 71, No. 4, pp. 77-85, 1994.
- ARGYRIS, C., R. PUTNAM, AND D. McLAIN SMITH, *Action Science - Concepts, Methods and Skills for Research and Intervention*, Jossey-Bass, San Francisco, 1982.
- ARGYRIS, C., AND D.A. SCHÖN, Participatory Action Research and Action Science Compared - A Commentary, *American Behavioral Scientist*, Vol. 32, No. 5, pp. 612-623, 1989.
- ARROW, K.J., The Economics of Agency, in: Pratt, J., and R. Zeckhauser (eds.), *Principals and Agents: The Structure of Business*, Harvard Business School Press, Boston, 1985.
- ASHBY, W.R., *An Introduction to Cybernetics*, Chapman and Hall, London 1956.
- ATKINSON, D.J., P.H. MARSHALL, Using the Multi Criteria Decision Making Facilities of a GDSS to Support Strategic Planning in an Organisation, in: G.J. de Vreede, and H.G. Sol (eds), *Proceedings of the INFORMS '95 Sessions on GDSS as Support for Designing Organizations and Information Systems*, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, 1995.
- ATTEWELL, P., AND J. RULE, Computing and Organizations: What We Know and What We Don't Know, *Communications of the ACM*, Vol. 27, pp. 1184-1192, 1984.
- AUGUST, J.H., *Joint Application Design - The Group Session Approach to System Design*, Yourdon Press, Englewood cliffs, New Jersey, 1991.
- BAARLE, B. VAN, Recherchefunctie en -organisatie in grote korpsen van gemeentepolitie, in: C.J.C.F. Fijnaut, A. Heijder (eds), *Recht van Spreken, Twintig jaar Recherche Advies Commissie*, J.B. van den Brink, Lochem, 1989. (In Dutch)
- BAARLE, B. VAN, Meer kansen voor aanpak georganiseerde misdaad, *Algemeen Politieblad*, No. 6, 1990. (In Dutch)
- BAARLE, B. VAN, Some aspects of Investigating Organized Crime in The Netherlands, in: C.J.C.F. Fijnaut, J.B. Jacobs (eds), *Organized Crime and its Containment, A Transatlantic Initiative*, Kluwer Law and Taxation Publishers, Boston, 1991.
- BAILY, M.N., AND A. CHAKRABARTI, *Innovation and the Productivity Crisis*, Brookings Institution, Washington, D.C., 1988.
- BAKOS, J.Y., AND E. BRYNJOLFSSON, Why Information Technology Hasn't Increased the Optimal Number of Suppliers, in: Nunamaker, J.F. Jr., and R.H. Sprague (eds), *Proceedings of the 26th Hawaiian International Conference on System Sciences*, pp. 799-808, IEEE Computer Society Press, Los Alamitos CA, 1993.
- BARTON, The Kansas City Experience: "Crack" Organized Crime Cooperative Task Force, *The Police Chief*, No. 1, 1988.
- BEER, S., *Design for Change*, John Wiley and Sons, New York, 1973.
- BELL, P.C. AND R.M. O'KEEFE, Visual Interactive Simulation - History, recent developments, and major issues, *Simulation*, Vol. 49, No. 3, pp. 109-116, 1987.
- BENBASAT, I., An Analysis of Research Methodologies, in: McFarlan, F.W. (ed.), *The Information Systems Research Challenge*, Harvard Business School Press, pp. 47-85, Boston, Massachusetts, 1984.
- BENBASAT, I. (ED.), *The Information Systems Research Challenge: Experiment Research Methods*, Harvard Business School, Boston, Massachusetts, 1989a.

- BENBASAT, I., Laboratory Experiments in Information Systems Studies with a Focus on Individuals: A Critical Appraisal, in: Benbasat, I. (ed.), *The Information Systems Research Challenge: Experiment Research Methods*, pp. 33-47, Harvard Business School, Boston, Massachusetts, 1989b.
- BENBASAT, I., G. DESANCTIS, AND B.R. NAULT, Empirical Research in Managerial Support Systems: A Review and Assessment, in: Holsapple, C.W. and A. Whinston (eds), *Recent Developments in Decision Support Systems*, pp. 383-437, Springer-Verlag, Berlin, 1992.
- BENBASAT, I., D.K. GOLDSTEIN, AND M. MEAD, The case research strategy in studies of information systems, *MIS Quarterly*, Vol. 11, No. 3, pp. 369-386, 1987.
- BENJAMIN, R.I., Information Technology in the 1990's: A Long Range Planning Scenario, *MIS Quarterly*, Vol. 6, No. 2, pp. 11-13, 1982.
- BIRD SCHOONHOVEN, C., Problems with Contingency Theory: Testing Assumptions Hidden within the Language of Contingency "Theory", *Administrative Science Quarterly*, Vol. 26, No. 3, pp. 349-377, 1981.
- BOEHM, B.W., A Spiral Model of Software Development and Enhancement, *IEEE Computer Journal*, May, 1988.
- BOOCH, G., *Object Oriented Design*, Benjamin/Cummings Publishing Company, 1993.
- BOSMAN, A., *Een metatheorie over het gedrag van organisaties*, Stenfert Kroese, Leiden, 1977. (In Dutch)
- BOSMAN, A., AND H.G. SOL, Knowledge Representation and Information Systems Design, *Proceedings IFIP WG 8.1 Working Conference*, Durham, 1984.
- BOSTROM, R.P. AND R. ANSON, The Face-to-Face Electronic Meeting: A Tutorial, in: Bostrom, R.P., R.T. Watson and S.T. Kinney (eds.), *Computer Augmented Teamwork, A Guided Tour*, Van Nostrand Reinhold, New York, 1992.
- BOSTROM, R.P., R. ANSON, AND V.K. CLAWSON, Group Facilitation and Group Support Systems, in: Jessup, L.M., J.S. Valacich (eds), *Group Support Systems - New Perspectives*, Macmillan Publishing Company, New York, 1993.
- BOSTROM, R.P., R.T. WATSON, AND S.T. KINNEY (EDS.), *Computer Augmented Teamwork, A Guided Tour*, Van Nostrand Reinhold, New York, 1992a.
- BOSTROM, R.P., R.T. WATSON, AND D. VAN OVER, The Computer-Augmented Teamwork Project, in: Bostrom, R.P., R.T. Watson and S.T. Kinney (eds.), *Computer Augmented Teamwork, A Guided Tour*, Van Nostrand Reinhold, New York, 1992b.
- BOTS, P.W.G., *An Environment to Support Problem Solving*, Doctoral Dissertation, Delft University of Technology, The Netherlands, 1989.
- BOTS, P.W.G., AND H.G. SOL, Shaping Organizational Information Systems through Co-ordination Support, in: Lee, R.M., A.M. McCosh and P. Migliarese (eds.), *Organizational Decision Support Systems*, Elsevier Science Publishers, Amsterdam 1988.
- BOTS, P.W.G., G.J. DE VREEDE, AND H.G. SOL, Applications of Group Systems in Systems Engineering, Policy Analysis, and Management, in: *From Shared Visions to Shared Realities - Proceedings of the Fifth Annual Group Systems Users' Conference*, Tucson, Arizona, March 14-16, 1994.
- BOUWMAN, M.R., AND G.F.S.J. KUIJLAARS, *Simulatiebouwstenen voor het modelleren van coördinatie in bedrijfsprocessen*, Internal Report, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, Delft 1995. (In Dutch)
- BRIGGS, R.O., R.J. JOHNSON, AND H.M. BROWN, *EMS in the classroom: New roles for teachers and technologies*, Tucson: MIS Department, University of Arizona manuscript, March 9, 1993.
- BROER, W., De Splitszwam van het Z&R-denken, *Algemeen Politieblad*, No. 3, pp. 14-16, 1994. (In Dutch)
- BRUSSAARD, B.K., AND P.A. TAS, Information and Organization Policies in Public Administration, in: Lavington, S.H. (ed.), *Information Processing 80*, pp. 821-826, North-Holland Publishing Company, Amsterdam, The Netherlands, 1980.
- BURRELL, G., AND G. MORGAN, *Sociological Paradigms and Organizational Analysis*, Heinemann, London, 1979.
- BURNS, T., AND G. STALKER, *The Management of Innovation*, Tavistock Publications, London, 1961.

- CABLA (COLLEGE VAN ADVIES EN BIJSTAND INZAKE LEVENSBESCHOUWELIJKE AANGELEGENHEDE VAN DE NATIONAL ZIEKENHUISRAAD I.S.M. HET UNIVERSITAIR CENTRUM VOOR BIOETHIEK EN GEZONDHEIDSRECHT VAN DE RIJSUNIVERSITEIT UTRECHT), *Selectie van patiënten, wachtlijsten en modellen voor een rechtvaardige verdeling van gezondheidszorg*, working paper, Utrecht, 1990. (In Dutch)
- CARMEL, E., B.C. HERNITER, AND J.F. NUNAMAKER JR., Labor-Management Contract Negotiations in an Electronic Meeting Room: A case Study, *Group Decision and Negotiation*, Vol. 2, No. 1, pp. 27-60, 1993.
- CASH, J.I. JR., AND B.R. KONSYSKI, IS Redraws Competitive Boundaries, *Harvard Business Review*, Vol. 63, No. 2, pp. 134-142, 1985.
- CASH, J.I. JR., AND P.R. LAWRENCE (EDS), *The Information Systems Research Challenge: Qualitative Research Methods*, Harvard Business School, Boston, Massachusetts, 1989.
- CHARAN, R.E.N., Now Networks Reshape Organizations - For Results, *Harvard Business Review*, Vol. 69, No. 5, pp. 104-115, 1991.
- CHECKLAND, P.B., *Systems Thinking, Systems Practice*, John Wiley & Sons, Chichester, 1981a.
- CHECKLAND, P.B., Science and the Systems Movement, in: Beishon, J. and G. Peters (eds). *Systems Behaviour*, Harper & Row, London, 1981b.
- CHECKLAND, P.B., From Optimizing to Learning: A Development of Systems Thinking for the 1990's, *Journal of the Operational Research Society*, Vol. 36, No. 9, pp. 757-767, 1985.
- CHECKLAND, P.B., Soft Systems Methodology: An Overview, *Journal of Applied Systems Analysis*, Vol. 15, pp. 27-30, 1988.
- CHECKLAND, P.B., From Framework through Experience to Learning: The Essential Nature of Action Research, in: Nissen, H.-E., H.K. Klein, and R. Hirschheim (eds), *Information Systems Research: Contemporary Approaches and Emergent Traditions*, Elsevier Science Publishers (North-Holland), pp. 397-403, 1991.
- CHECKLAND, P.B., AND J. SHOLES, *Soft Systems Methodology in Action*, John Wiley & Sons, Chichester, 1990.
- CHEN, M., AND J.F. NUNAMAKER JR., Metaplex: An Integrated Environment for Organization and Information System Development, in: DeGross, J.I., J.C. Henderson, and B.R. Konsynski (eds.), *Proceedings of the 10th International Conference on Information Systems*, pp. 141-151, Boston, MA, 1989.
- CHIDAMBARAM, L., AND R.P. BOSTROM, Evolution of Group Performance Over Time: A Repeated Measures Study of GDSS Effects, *Journal of Organizational Computing*, Vol. 3, No. 4, pp. 443-469, 1993.
- CHILD, J., *Organizations: A Guide to Problems and Practice*, Harper & Row, New York, N.Y. 1977.
- CHING, C., C.W. HOLSAPPLE AND A.B. WHINSTON, Reputation, Learning and Coordination in Distributed Decision-Making Contexts, *Organization Science*, Vol. 3, No. 2, pp. 275-297, 1992.
- CHURCHMAN, C.W., *The Design of Inquiring Systems: Basic Concepts of Systems and Organizations*, Basic Books, New York, NY, 1971.
- CID (CRIMINAL INVESTIGATIONS DEPARTMENT), *Voornemen tot Reorganisatie Dienst Centrale Executieve Rechercheen Dienst Centrale Ondersteunende Recherche & Onderzoeksplan Dienst Centrale Executieve en Ondersteunende Recherche*, Amsterdam Municipal Police Force, 1992. (In Dutch)
- CID (CRIMINAL INVESTIGATIONS DEPARTMENT), *Invoeringsplan Dienst Centrale Executieve en Ondersteunende Recherche*, Amsterdam Municipal Police Force, 1993. (In Dutch)
- CLARK, D.F., AND R.L. ACKOFF, A Report on Some Organizational Experiments, *Operations Research*, Vol. 7, pp. 279-293, 1959.
- CLARK, P.A., *Action Research and Organizational Change*, Harper and Row, London, 1972.
- CLEMONS, E.K., *Information Technology and the Changing Boundary of the Firm: Implications for Industrial Restructuring*, Working paper, Department of Decision Sciences, University of Pennsylvania, Philadelphia, USA, 1992.
- COLEMAN, D., 20 Rules for Success with Groupware, in: Coleman, D. and P. Huckle (eds.), *Proceedings of GroupWare '94 Europe*, pp. 11-38, June 6-8, London, Great Britain, 1994.

- COLEMAN, D. AND P. HUCKLE (EDS.), *Proceedings of GroupWare '94 Europe*, June 6-8, London, Great Britain, 1994.
- COUNCIL OF EUROPE, *Les rapports de 19^e Conférence des ministres européens de la Justice*, La Valette, Malta, 14-15 June, 1994. (In French)
- CROWSTON, K., T.W. MALONE, AND F. LIN, Cognitive Science and Organizational Design: A Case Study of Computer Conferencing, *Human-Computer Interaction*, Vol. 3, pp. 59-85, 1987.
- CULNAN, M.J., AND M.L. MARKUS, Information Technologies, in: Jablin, F.M., L.L. Putnam, K.H. Roberts, and L.W. Porter (eds.), *Handbook of Organizational Communication - An Interdisciplinary Perspective*, pp. 420-443, Sage Publications, London, 1987.
- DAFT, R.L., AND R.H. LENGEL, Organizational Information Requirements, Media Richness, and Structural Design, *Management Science*, Vol. 32, No. 5, pp. 554-571, 1986.
- DAFT, R.L., R.H. LENGEL, AND L.K. TREVINO, Message Equivocality, Media Selection and Manager Performance: Implications for Information System, *MIS Quarterly*, Vol. 11, pp. 355-336, 1987.
- DAVENPORT, T.H., Saving IT's Soul: Human-centered Information Management, *Harvard Business Review*, Vol. 71, No. 2, pp. 119-131, 1994.
- DAVENPORT, T.H., AND J.E. SHORT, The New Industrial Engineering: Information Technology and Business Process Redesign, *Sloan Management Review*, pp. 11-27, Summer 1990.
- DAVIS, S., AND J. BOTKIN, The Coming of Knowledge-Based Business, *Harvard Business Review*, Vol. 71, No. 5, pp. 165-170, 1994.
- DAVISON, R., The Role of GSS in Requirements Specification: A Case Study, in: G.J. de Vreede, and H.G. Sol (eds), *Proceedings of the INFORMS '95 Sessions on GDSS as Support for Designing Organizations and Information Systems*, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, 1995.
- DEAN, D., R. ORWIG, J. LEE, AND D. VOGEL, Modeling with a Group Modeling Tool: Group Support, Model Quality, and Validation, in: Nunamaker, J.F. Jr., and R.H. Sprague (eds), *Proceedings of the 27th Hawaiian International Conference on System Sciences*, Volume IV, pp. 214-223, IEEE Computer Society Press, Los Alamitos CA, 1994.
- DEKKER, W., *Bereidheid tot Verandering*, Commissie Structuur en Financiering Gezondheidszorg, Rijswijk, 1987.
- DENNIS, A.R., Electronic Support for Large Groups, *Journal of Organizational Computing*, Vol. 4, No. 2, pp. 177-197, 1994.
- DENNIS, A.R., AND R.B. GALLUPE, A History of Group Support Systems Empirical Research: Lessons Learned and Future Directions, in: Jessup, L.M., J.S. Valacich (eds), *Group Support Systems - New Perspectives*, Macmillan Publishing Company, New York, 1993.
- DENNIS, A.R., G.S. HAYES, AND R.M. DANIELS, JR., Re-Engineering Business Process Modelling, in: Nunamaker, J.F. Jr., and R.H. Sprague (eds), *Proceedings of the 27th Hawaiian International Conference on System Sciences*, Volume IV, pp. 244-253, IEEE Computer Society Press, Los Alamitos CA, 1994.
- DENNIS, A.R., G.S. HAYES, AND R.M. DANIELS, JR., Accelerating Information Systems Design with GDSS: Lessons from the U.S. Department of Defense, in: G.J. de Vreede, and H.G. Sol (eds), *Proceedings of the INFORMS '95 Sessions on GDSS as Support for Designing Organizations and Information Systems*, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, 1995.
- DENNIS, A.R., J.F. NUNAMAKER JR., AND D.R. VOGEL, A comparison of laboratory and field research in the study of electronic meetings systems, *Journal of Management Information Systems*, Vol. 7, No. 2, pp. 107-135, 1991.
- DENNIS, A.R., J.S. VALACICH, AND J.F. NUNAMAKER JR., An Experimental Investigation of the Effects of Group Size in an Electronic Meeting Environment, *IEEE System, Man and Cybernetics*, Vol. 25, pp. 1049-1057, 1990.
- DENZIN, N.K., *The Research Act*, 2nd Edition, McGraw-Hill, New York, 1978.

- DESANTIS, G., Small Group Research in Information Systems: Theory and Method, in: Benbasat, I. (ed.), *The Information Systems Research Challenge: Experiment Research Methods*, pp. 53-82, Harvard Business School, Boston, Massachusetts, 1989.
- DESANTIS, G., AND R.B. GALLUPE, Group Decision Support Systems: A New Frontier, *Data Base*, Vol. 16, No. 2, pp. 3-10, 1985.
- DESANTIS, G., AND R.B. GALLUPE, A Foundation for the Study of Group Decision Support Systems, *Management Science*, Vol. 33, No. 5, pp. 589-609, 1987.
- DESANTIS, G., AND M.S. POOLE, Capturing the Complexity in Advanced Technology Use: Adaptive Structuration Theory, *Organization Science*, Vol. 5, No. 2, pp. 121-147, 1994.
- DESANTIS, G., M.S. POOLE, G. DESHARNAIS, AND H. LEWIS, Using computing to facilitate the quality improvement process: the IRS-Minnesota project, *TIMS Interfaces*, Vol. 21, No. 6, pp. 23-36, 1991.
- DESANTIS, G., V. SAMBAMURTHY, AND R.T. WATSON, A Software Environment for GDSS Research, in: E.S. Weber (ed.), *DSS-88 Transactions: Eight International Conference on Decision Support Systems*, pp. 3-12, June 6-9, 1988.
- DICK, B., A Beginner's Guide to Action Research, *Arcs Newsletter*, Vol. 1, No. 1, pp. 5-9, 1993.
- DICKSON, G.W., Management Information Systems: Evolutions and Status, *Advances in Computers*, Vol. 20, pp. 1-37, 1981.
- DICKSON, G.W., A Programmatic Appeal to Information Systems Research: An Experimentalist's View, in: Benbasat, I. (ed.), *The Information Systems Research Challenge: Experiment Research Methods*, pp. 147-170, Harvard Business School, Boston, Massachusetts, 1989.
- DICKSON, G.W., AND G. DESANTIS, The Management Of Information Systems: Research Status And Themes, in: Jenkins, A.M., H.S. Siegle, V. Wojtkowski, and G. Wojtkowski (eds), *Research Issues in Information Systems: Agenda for the 1990's*, W.C. Brown, 1990.
- DICKSON, G.W., J.E. LEE-PARTRIDGE, AND L.H. ROBINSON, Exploring Modes of Facilitative Support for GDSS Technology, *MIS Quarterly*, Vol. 17, No. 2, pp. 173-194, 1993.
- DICKSON, G.W., M. LIMAYEM, J.E. LEE PARTRIDGE, AND G. DESANTIS, Facilitating Computer Supported Meetings: A Cumulative Analysis In A Multiple Criteria Task Environment, working paper, *Group Decision and Negotiation*, (in press).
- DICKSON, G.W., M.S. POOLE, AND G. DESANTIS, An Overview of the GDSS Research Project and the SAMM System, in: Bostrom, R.P., R.T. Watson and S.T. Kinney (eds.), *Computer Augmented Teamwork, A Guided Tour*, Van Nostrand Reinhold, New York, 1992.
- DIEHL, M., AND W. STROEBE, Productivity Loss In Brainstorming Groups: Toward the Solution of a Riddle, *Journal of Personality and Social Psychology*, Vol. 53, No. 3, pp. 497-509, 1987.
- DIJK - VEENENGA KINGMA, F., *Sociotechniek en Ziekenhuis*, MSc-Thesis, Faculty of Social Sciences, Erasmus University Rotterdam, Rotterdam, 1992.
- DOUMA, S., AND H. SCHREUDER, *Economic Approaches to Organizations*, Prentice Hall, New York, 1991.
- DRUCKER, P.F., The coming of the new organization, *Harvard Business Review*, Vol. 66, No. 1, pp. 45-53, 1988.
- DRUCKER, P.F., *The New Realities*, Harper and Row, New York, 1989.
- DRUCKER, P.F., The New Productivity Challenge, *Harvard Business Review*, Vol. 69, No. 6, pp. 69-79, 1991.
- DRUCKER, P.F., The Information Executives Truly Need, *Harvard Business Review*, Vol. 72, No. 1, pp. 54-62, 1995.
- DUCK, J.D., Managing Change: The Art of Balancing, *Harvard Business Review*, Vol. 70, No. 6, pp. 109-118, 1993.
- DUR, R.C.J., *Business Reengineering in Information Intensive Organizations*, Doctoral dissertation, Delft University of Technology, Delft, The Netherlands, 1992.
- DUYNE, P.C. VAN, Crime Enterprises and the Legitimate Industry in The Netherlands, in: C.J.C.F. Fijnaut, J.B. Jacobs (eds), *Organized Crime and its Containment, A Transatlantic Initiative*, Kluwer Law and Taxation Publishers, Deventer/Boston, 1991.

- DUYNE, P.C. VAN, Strategieën ter Bestrijding van Georganiseerde Misdaad, *Research Memorandum*, Wetenschappelijk Onderzoek- en Documentatiecentrum (WODC), Ministry of Justice, 1992a. (In Dutch)
- DUYNE, P.C. VAN, Georganiseerde misdaad, in: C. Bouw, H. van de Bunt, H. Franke (eds), *Kernbegrippen in de criminologie*, Gouda Quint BV, Arnhem, 1992b. (In Dutch)
- DUYNE, P.C. VAN, R.F. KOUWENBERG, AND G. ROMEIJN, *Misdaadondernemingen. Ondernemende misdadigers in Nederland*, Arnhem, Gouda Quint, 1990. (In Dutch)
- EASON, K., *Information Technology and Organisational Change*, Taylor and Francis, London, 1988.
- EDEN, C., AND F. ACKERMANN, Strategy Development and Implementation - The Role of a Group Decision Support System, in: Bostrom, R.P., R.T. Watson and S.T. Kinney (eds.), *Computer Augmented Teamwork, A Guided Tour*, Van Nostrand Reinhold, New York, 1992.
- EGMOND, F., Georganiseerde misdaad en de overheid in het verleden: Nederland tijdens de 17e en 18e eeuw, in: C.J.C.F. Fijnaut (ed.), *Georganiseerde misdaad en strafrechtelijk politiebeleid; rechtshistorische, politieke, journalistieke en rechtsvergelijkende beschouwingen*, J.B. van den Brink, Lochem, 1989. (In Dutch)
- EIJCK, D.T.T. VAN, *Designing Organizational Coordination*, Doctoral dissertation, Delft University of Technology, Delft, The Netherlands, 1996. (In preparation)
- EIJCK, D.T.T. VAN, E.M.W.M. PHILIPS, AND H.G. SOL, *Designing Organizational Coordination: An Object Oriented Approach*, Internal report, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, Delft, The Netherlands, 1995.
- EIJCK, D.T.T. VAN, R.J. STRENG AND H.G. SOL, Dynamic Modelling to Support Organizational Decision Making. The Stevedore Case, in: *Proceedings of the 1993 Systems Dynamics Conference*, Cancun, Mexico, 1993.
- EIJCK, D.T.T. VAN, AND G.J. DE VREEDE, A Body of Concepts for Simulating Coordination, in: *Proceedings of the International Simulation and Gaming Association Conference (ISAGA '94)*, 20-29 July, Ann Arbor, Michigan, USA, 1994.
- EIJCK, D.T.T. VAN, AND G.J. DE VREEDE, The Dynamics of Organizational Coordination, in: J.F. Nunamaker Jr., R.H. Sprague Jr. (eds.), *Proceedings of the 28th Hawaiian International Conference on Systems Sciences*, pp. 851-860, IEEE Computer Society Press, 1995.
- EIJNATTEN, F.M. VAN, *The Paradigm that Changed the Work Place. Annals of STSD*, Van Gorcum Publishers, Assen/Maastricht, 1993.
- EISENHARDT, K.M., Building Theories from Case Study Research, *Academy of Management Review*, Vol. 14, No. 2, pp. 532-550, 1989.
- ELLIS, C.A., S.J. GIBBS, AND G.L. REIN, Groupware: Some issues and experiences, *Communications of the ACM*, Vol. 34, No. 1, pp. 38-58, 1991.
- ELMARAGHY, H.A., Simulation and graphical animation of advanced manufacturing systems, *Journal of Manufacturing Systems*, Vol. 1, No. 1, pp. 53-63, 1982.
- ELSEVIER, De bankbediende als politieagent, psycholoog en undercover, Vol. 50, No. 5, pp. 60-62, 1994. (In Dutch)
- EMERY, F.E., AND M. EMERY, *Participative Design*, Canberra: Centre for Continuing Education, Australian National University, 1974.
- ENGELSTAD, P.H., AND B. GUSTAVSEN, Swedish network development for implementing national work reform strategy, *Human Relations*, Vol. 46, No. 2, pp. 219-248, 1993.
- ESHOF, P. VAN DEN, AND D. VEGTER, Georganiseerde Twijfel?, *Tijdschrift voor de Politie*, Vol. 56, No. 1/2, pp. 9-13, 1994. (In Dutch)
- FIJNAUT, C.J.C.F., Georganiseerde misdaad, een vergelijking tussen de Verenigde Staten en West-Europa, in: C.J.C.F. Fijnaut (ed.), *Georganiseerde misdaad en strafrechtelijk politiebeleid; rechtshistorische, politieke, journalistieke en rechtsvergelijkende beschouwingen*, J.B. van den Brink, Lochem, 1989a. (In Dutch)
- FIJNAUT, C.J.C.F., De reorganisatie van de Nederlandse politie en de organisatie van de misdaad in de loop van de 19e en 20e eeuw, in: C.J.C.F. Fijnaut (ed.), *Georganiseerde misdaad en strafrechtelijk*

- politiebeleid; rechtshistorische, politieke, journalistieke en rechtsvergelijkende beschouwingen*, J.B. van den Brink, Lochem, 1989b. (In Dutch)
- FIJNAUT, C.J.C.F., Ontwikkelingen op het gebied van de reguliere recherche in de omliggende landen, in: C.J.C.F. Fijnaut, A. Heijder (eds), *Recht van Spreken, Twintig jaar Recherche Advies Commissie*, Van den Brink, Lochem, 1989c. (In Dutch)
- FLORES, F., M. GRAVES, B. HARTFIELD, AND T. WINOGRAD, Computer Systems and the Design of Organizational Interaction, *ACM Transactions on Office Information Systems*, Vol. 6, No. 2, pp. 153-172, 1988.
- FLOYD, C., A Systematic Look at Prototyping, in: Budde, R., K. Kuhlenkamp, L. Mathiassen en H. Zullighoven (eds.), *Approaches to Prototyping*, pp. 1-18, Springer-Verlag, Berlin, 1984.
- FOSTER, L.W., AND D.M. FLYNN, Management Information Technology: Its Effects on Organizational Form and Function, *MIS Quarterly*, Vol. 8, pp. 229-235, 1984.
- FRANCESCATO, D., A Multidimensional Perspective of Organizational Change, *Systems Practice*, Vol. 5, No. 2, pp. 129-146, 1992.
- GALBRAITH, J.R., Organization Design: An Information Processing View, *Interfaces*, Vol. 4, No. 3, pp. 28-36, 1974.
- GALBRAITH, J.R., *Organization design*, Addison-Wesley, 1977.
- GALLIERS, R.D., Choosing Appropriate Information Systems Research Approaches: A Revised Taxonomy, in: Nissen, H.-E., H.K. Klein, and R. Hirschheim (eds), *Information Systems Research: Contemporary Approaches and Emergent Traditions*, Elsevier Science Publishers (North-Holland), pp. 327-345, 1991.
- GALLIERS, R.D., AND F.F. LAND, Choosing Appropriate Information Systems Research Methodologies, *Communications of the ACM*, Vol. 30, No. 11, pp. 900-902, 1987.
- GALLUPE, R.B., The Executive Decision Centre, in: Bostrom, R.P., R.T. Watson and S.T. Kinney (eds.), *Computer Augmented Teamwork, A Guided Tour*, Van Nostrand Reinhold, New York, 1992.
- GEORGE, J.F., A.R. DENNIS, AND J.F. NUNAMAKER JR., An Experimental Investigation of Facilitation in an EMS Decision Room, *Group Decision and Negotiation*, Vol. 1, No. 1, pp. 57-70, 1992.
- GEORGE, J.F., AND J.L. KING, Examining the Computing and Centralization Debate, *Communications of the ACM*, Vol. 34, No. 7, pp. 63-72, 1991.
- GEURTS, J.L., I. WENZLER, AND H.J.J. VAN KUPPEVELT, Communicating complexity through visualisation: the use of schematics in gaming/simulations, in: *Proceedings of the 1993 Systems Dynamics Conference*, Cancun, Mexico, 1993.
- GHOSHAL, S., AND C.A. BARTLETT, The Multinational Cooperation as an Interorganizational Network, *Academy of Management Review*, Vol. 15, No. 4, pp. 603-625, 1990.
- GHOSHAL, S., AND C.A. BARTLETT, Changing the Role of Top Management: Beyond Structure to Processes, *Harvard Business Review*, Vol. 72, No. 1, pp. 86-96, 1995.
- GHOSHAL, S., H. KORINE, AND G. SZULANSKI, Interunit Communication in Multinational Corporations, *Management Science*, Vol. 40, No. 1, pp. 96-110, 1994.
- GIPPS, P.G., The role of computer graphics in validating simulation models, *Mathematics and computers in simulation*, Vol. 28, pp. 285-289, 1986.
- GLASER, B., AND A. STRAUSS, *The Discovery of Grounded Theory: Strategies for Qualitative Research*, Aldine, Chicago, 1967.
- GOPAL, A., S. HAYNE, P. LICKER, C. POLLARD, AND W. CHIN, GSS Above the Snow Line, in: Bostrom, R.P., R.T. Watson and S.T. Kinney (eds.), *Computer Augmented Teamwork, A Guided Tour*, Van Nostrand Reinhold, New York, 1992.
- GRANT, J.W. AND S.A. WEINER, Factors to consider in choosing a graphically animated simulation system, *Industrial Engineering*, Vol. 18, No. 8, pp. 36-68, 1986.
- GRAY, P., Group Decision Support Systems, *Decision Support Systems*, Vol. 3, pp. 233-242, 1987.
- GRAY, P., The Decision Laboratory, in: Bostrom, R.P., R.T. Watson and S.T. Kinney (eds.), *Computer Augmented Teamwork, A Guided Tour*, Van Nostrand Reinhold, New York, 1992.

- GRAY, P., J.S. ARONOFSKY, N. BERRY, O. HELMER, G.R. KANE, AND T.E. PERKINS, The SMU Decision Room Project, in: *DSS-81 Transactions of the First International Conference on Decision Support Systems*, pp. 122-129, Execucom Systems Corporation, 1981.
- GRAY, P., AND I. BOROVITS, Gaming and Group DSS, in: *DSS-86 Transactions: Sixth International Conference on Decision Support Systems*, Washington D.C., April, 1988.
- GROHOWSKI, R., C. MCGOFF, D. VOGEL, B. MARTZ, AND J.F. NUNAMAKER JR., Implementing Electronic Meeting Systems at IBM: Lessons Learned and Success Factors, *MIS Quarterly*, Vol. 14, No. 4, pp. 369-384, 1990.
- GULICK, L., Notes on the Theory of Organization, in: Gulick, L., and L. Urwick (eds.), *Papers on the Science of Administration*, Institute of Public Administration, New York, 1937.
- GURBAXANI, V., AND E. SHI, Computers and Coordination in Manufacturing, *Journal of Organizational Computing*, Vol. 2, No. 1, pp. 27-46, 1992.
- GURBAXANI, V., AND S. WHANG, The Impact of Information Systems on Organizations and Markets, *Communications of the ACM*, Vol. 34, No. 1, pp. 59-73, 1991.
- HALL, R.H., *Organizations: Structure and Process*, Prentice-Hall, 1972.
- HAMMER, M., Reengineering Work: Don't Automate, Obliterate, *Harvard Business Review*, Vol. 68, no. 4, pp. 104-112, 1990.
- HAMMER, M., AND J. CHAMPY, *Re-engineering the Corporation: A Manifesto for Business Revolution*, Harper Business, New York, 1993.
- HARST, G.M.A. VAN DER, (ED.), Workflow Evaluatieproject eindrapport, *Report 94/04*, Software Engineering Research Centre, Utrecht, The Netherlands, 1994. (In Dutch)
- HARTLEY, J.F., Case Studies in Organizational Research, in: Cassell, C., and G. Symon (eds.), *Qualitative Methods in Organizational Research - A Practical Guide*, Sage Publications, London, 1994.
- HENGST, S.O. DEN, *Reorganisatie Bureau Recherche Informatie (BRI) en Ontwerp VoortSchrijdend Inzicht (VSI)*, MSc Thesis, Faculty of Technical Mathematics and Informatics, Delft University of Technology, 1994. (In Dutch)
- HENGST, S.O. DEN, *VoortSchrijdend Inzicht: Stand van zaken*, Internal report, Faculty of Technical Mathematics and Informatics, Delft University of Technology, 1995. (In Dutch)
- HENGST, S.O. DEN, AND G.J. DE VREEDE, *Is er inzicht in het Voortschrijdend Inzicht?*, Results of the GroupSystems session, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, Delft, 1993. (In Dutch)
- HENGST, S.O. DEN, G.J. DE VREEDE, AND A.H. LEEUWRIK, *Criteria voor het VSI*, Results of the GroupSystems session, Amsterdam Municipal Police Force, Amsterdam, 1993a. (In Dutch)
- HENGST, S.O. DEN, G.J. DE VREEDE, A.H. LEEUWRIK, J.W. VAN DER AALST, M. BOUWMAN, AND B. NOORDUIN, *Introductie Prototype VSI*, Results of the GroupSystems session, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, Delft, 1993b. (In Dutch)
- HENGST, S.O. DEN, G.J. DE VREEDE, AND H.G. SOL, *Onderweg naar een projectmatige aanpak van de georganiseerde criminaliteit - Projectvoorstel*, Internal report, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, Delft 1994. (In Dutch)
- HENGST, S.O. DEN, G.J. DE VREEDE, AND H.G. SOL, A group approach for societal problem solving: Toward an agenda for fighting crime in the inner city of Amsterdam, in: *Proceedings of the GroupSystems' Sixth Annual Users Conference*, 13-16 March, Tucson AZ, USA, 1995.
- HERIK, K.W. VAN DEN, Supporting Group Design Tasks in Policy-making - A Case Study in Supporting Policy Makers with Group Decision Technology, in: Lundberg, B.G. (ed.), *Proceedings of the Second Scandinavian Research Seminar on Information and Decision Networks*, Växjö, Sweden, 11-13 May, 1995.
- HERZOG, J.P., People: The Critical Factor in Managing Change, *Journal of Systems Management*, pp. 6-11, March 1991.

- HILTZ, S.R., AND M. TUROFF, Virtual Meetings: Computer Conferencing and Distributed Group Support, in: Bostrom, R.P., R.T. Watson and S.T. Kinney (eds.), *Computer Augmented Teamwork, A Guided Tour*, Van Nostrand Reinhold, New York, 1992.
- HOFSTEDE, G., *Culture's consequences: International differences in work-related values*, Sage, Beverly Hills, 1980.
- HOGENBRINK, J.W., AND A. TEERDS, De Recherche Advies Commissie, in: C.J.C.F. Fijnaut, A. Heijder (eds), *Recht van Spreken, Twintig jaar Recherche Advies Commissie*, Van den Brink, Lochem, 1989. (In Dutch)
- HOGENWIND, F.J., *Patiëntenstromen als planningsinstrument*, Nationaal Ziekenhuisinstituut, Utrecht, 1988. (In Dutch)
- HOLT, A.W., Diplans: A new language for the study and implementation of coordination, in: *ACM transactions on office information systems*, vol.6, no. 2, pg 109-125, 1988.
- HOLT, C.C., Conceptual Environment For Organization Support Systems: Information Technology In Organizations, in: Sol, H.G. and J. Vecsenyi (eds), *Environments for Supporting Decision Processes*, pp. 215-238, Elsevier Science Publishers (North Holland), Amsterdam, The Netherlands, 1991.
- HOOGVEEN, M.J., *The Viability of Multimedia Retrieval Systems for Marketing and Sales*, Doctoral Dissertation, Delft University of Technology, Delft, The Netherlands, 1994.
- HOOGVEEN, M.J., AND K. VAN DER MEER, Integration of Information Retrieval and Database Management in Support of Multimedia Police Work, *Journal of Information Science*, Vol. 20, No. 2, pp. 79-87, 1994.
- HOOGVEEN, M.J., K. V.D. MEER, AND H.G. SOL, Integratie van information retrieval en database management in OCTOPUS, *Informatie*, Vol. 34, No. 4, 1992. (In Dutch)
- HUBER, G.P., The Nature and Design of Post-Industrial Organizations, *Management Science*, Vol. 30, No. 8, pp. 928-951, 1984.
- HUBER, G.P., AND R.L. DAFT, The Information Environments of Organizations, in: Jablin, F.M., L.L. Putnam, K.H. Roberts, and L.W. Porter (eds), *Handbook of Organizational Communication - An Interdisciplinary Perspective*, Sage Publications, London, 1987.
- HUBER, G.P., AND R. MCDANIEL, The Decision Making Paradigm of Organizational Design, *Management Science*, Vol. 32, No. 5, pp. 572-589, 1986.
- IVARI, J., A Paradigmatic Analysis of Contemporary Schools of IS Development, *European Journal of Information Systems*, Vol. 1, No. 4, pp. 249-272, 1991.
- JABLIN, F.M., Formal Organization Structure, in: Jablin, F.M., L.L. Putnam, K.H. Roberts, and L.W. Porter (eds), *Handbook of Organizational Communication - An Interdisciplinary Perspective*, Sage Publications, London, 1987.
- JACOBS, J.B., The Failures of American Law Enforcement in Combating Organized Crime, in: Fijnaut, C.J.C.F., J.B. Jacobs (eds), *Organized Crime and its Containment, A Transatlantic Initiative*, Kluwer Law and Taxation Publishers, Deventer/Boston, 1991.
- JARVENPAA, S.L., The Importance of Laboratory Experimentation in IS Research, *Communications of the ACM*, Vol. 31, No. 12, pp. 1502-1504, 1988.
- JARVENPAA, S.L., G.W. DICKSON, AND G. DESANCTIS, Methodological issues in experimental IS research: Experiences and recommendations, *MIS Quarterly*, Vol. 9, No. 2, pp. 141-156, 1985.
- JARVENPAA, S.L., V.S. RAO, AND G.P. HUBER, Computer Support for Meetings of Medium-Sized Groups Working on Unstructured Problems: A Field Experiment, *MIS Quarterly*, Vol. 12, pp. 645-666, 1988.
- JESSUP, L. M., AND VALACICH, J. (EDS.), *Group support systems: New perspectives*, New York: Macmillan, 1993.
- JOHANSEN, R., *Groupware: Computer support for business teams*, New York: The Free Press, 1988.
- JOHANSEN, R., An Introduction Computer-Augmented Teamwork, in: Bostrom, R.P., R.T. Watson and S.T. Kinney (eds.), *Computer Augmented Teamwork, A Guided Tour*, Van Nostrand Reinhold, New York, 1992.
- JOHANSEN, R., J. VALLEE, AND K. SPANGLER, *Electronic Meetings: Technical Alternatives and Social Choices*, Addison-Wesley Publishing Company, Reading, Massachusetts, 1979.

- JONG, R. DE, *An Adaptive Trip Planning Support Environment*, Doctoral dissertation, Delft University of Technology, Delft, The Netherlands, 1994.
- JÖNSSON, S., Action Research, in: Nissen, H.-E., H.K. Klein, and R. Hirschheim (eds), *Information Systems Research: Contemporary Approaches and Emergent Traditions*, Elsevier Science Publishers (North-Holland), pp. 371-396, 1991.
- KAPLAN, B., AND D. DUCHON, Combining qualitative and quantitative methods in information systems research: A case study, *MIS Quarterly*, Vol. 12, No. 4, pp. 571-586, 1988.
- KEEN, T., E-Mail: From Simplicity to Ubiquity, in: Lloyd, P. (ed.), *Groupware in the 21st Century - Computer Supported Cooperative Working Toward the Millennium*, pp. 92-96, Adamantine Press, London, Great Britain, 1994.
- KICKERT, W.J.M., *Organisation of Decision-Making: A Systems-Theoretical Approach*, Doctoral Dissertation, Eindhoven University of Technology, Eindhoven, The Netherlands, 1979.
- KIESER, A., AND H. KUBICEK, *Organisation*, Walter de Gruyter, Berlin/New York, 1977. (In German)
- KIESLER, S., Thinking Ahead: The Hidden Message in Computer Networks, *Harvard Business Review*, Vol. 64, No. 1, pp. 46-60, 1986.
- KIM, W., AND F.H. LOCHOVSKY (EDS.), *Object-Oriented Concepts, Databases, and Applications*. ACM Press, Addison-Wesley, 1989.
- KIRST, M.W., G. MEISTER, AND S.R. ROWLEY, Policy Issue Networks: Their Influence on State Policymaking, *Policy Studies Journal*, Vol 13, No. 3, pp. 247-264, 1984.
- KLEIJNEN, J.P.C., *Verification and Validation of Simulation Models*, Research Memorandum, Tilburg University, March 1992.
- KORF, D.J., Drugsgebonden criminaliteit, in: C. Bouw, H. van de Bunt, H. Franke (eds), *Kernbegrippen in de criminologie*, Gouda Quint BV, Arnhem, 1992. (In Dutch)
- KORTEKAAS, J.J.C., Fenomeenaanpak, *Tijdschrift voor de Politie*, Vol. 56, No. 7/8, pp. 24-28, 1994. (In Dutch)
- KORTHALS-ALTES, F., Het beleid inzake georganiseerde criminaliteit, in: C.J.C.F. Fijnaut (ed.), *Georganiseerde misdaad en strafrechtelijk politiebeleid; rechtshistorische, politieke, journalistieke en rechtsvergelijkende beschouwingen*, J.B. van den Brink, Lochem, 1989. (In Dutch)
- KRAEMER, K.L., *The Information Systems Research Challenge: Survey Research Methods*, Harvard Business School, Boston, Massachusetts, 1989.
- KRAEMER, K.L., AND J. KING, Computer-Based Systems for Cooperative Work and Group Decision Making, *ACM Computing Surveys*, Vol. 20, No. 2, pp. 115-146, 1988.
- KRAMER, J., AND K. NG, Animation of Requirements Specifications, *Software - Practice and Experience*, Vol. 18, No. 8, pp. 749-774, 1988.
- KRCMAR, H., LEWE, H., AND G. SCHWABE, Empirical CATeam Research in Meetings, in: Nunamaker, J.F. Jr., and R.H. Sprague (eds), *Proceedings of the 27th Hawaiian International Conference on System Sciences*, Volume IV, pp. 31-40, IEEE Computer Society Press, Los Alamitos CA, 1994.
- KUIJLAARS, G.F.S.J., *Coördinatie bij projecten Georganiseerde Criminaliteit*, MSc Thesis, Faculty of Technical Mathematics and Informatics, Delft University of Technology, 1995. (In Dutch)
- KUIJLAARS, G.F.S.J., AND G.J. DE VREEDE, *Functionele Specificatie VoortSchrijdend Overzicht (VSO)*, Results of the GroupSystems sessions, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, Delft 1995a. (In Dutch)
- KUIJLAARS, G.F.S.J., AND G.J. DE VREEDE, *Evaluatie Prototype VoortSchrijdend Overzicht (VSO)*, Results of the GroupSystems sessions, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, Delft 1995b. (In Dutch)
- KUIJLAARS, G.F.S.J., G.J. DE VREEDE, AND H.G. SOL, *Onderweg naar een projectmatige aanpak van de georganiseerde criminaliteit - Functioneel Ontwerp VoortSchrijdend Overzicht*, Internal report, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, Delft 1995. (In Dutch)
- KUMAR, K., S. QURESHI, AND G.J. DE VREEDE, Supporting Soft Systems Thinking with Group Support Systems: A Proposal, in: G.J. de Vreede, and H.G. Sol (eds), *Proceedings of the INFORMS '95 Sessions*

- on GDSS as Support for Designing Organizations and Information Systems, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, 1995.
- KÜSTER, D., Das Lagebild der organisierten Kriminalität in der Bundesrepublik Deutschland, illustriert anhand typischer Ermittlungsverfahren, in: *Organisierte Kriminalität in einem Europa durchlässiger Grenzen*, Bundeskriminalamt, Wiesbaden, 1991. (In German)
- KYNG, M., Designing for Cooperation: Cooperating in Design, *Communications of the ACM*, Vol. 34, No. 12, pp. 65-73, 1991.
- LAND, F., Managing the Change Process, in: *Proceedings of the 3rd IT Conference on the Role of IT in Business Success Today and in the 1990s*, Athens, Greece, 1990.
- LAND, F., AND R. HIRSCHHEIM, Participative Systems Design, *Journal of Applied Systems Analysis*, pp. 91-107, 1983.
- LAWRENCE, P.R., AND J.W. LORSCH, *Organization and Environment*, Harvard Business School, Boston, 1967.
- LEAVITT, H.J., Applying Organizational Change in Industry: Structural, Technological, and Humanistic Approaches, in: March, J.G. (ed.), *Handbook of Organizations*, Rand McNally, Chicago, Illinois, 1965.
- LEAVITT, H.J., AND T.L. WHISLER, Management in the 1980's, *Harvard Business Review*, Vol. 36, pp. 41-48, 1958.
- LEDYARD, J.O., The Design of Coordination Mechanisms and Organizational Computing, *Journal of Organizational Computing*, Vol. 3, No. 1, pp. 121-134, 1993.
- LEE, A.S., A scientific methodology for MIS case studies, *MIS Quarterly*, Vol. 13, No. 1, pp. 33-50, 1989.
- LEEUW, A.C.J. DE, *Organisaties: Management, Analyse, Ontwerp en Verandering*, Van Gorcum, Assen/Maastricht, 1990. (In Dutch)
- LEEUWRIK, A.H., AND G.A.M. OTTE, *Invoeringsplan - Definitieve Versie (3.0)*, Criminal Investigations Department, Amsterdam Municipal Police Force, 1993.
- LENSING, J.A.W., Reacties in wetgeving en rechtspraak op zware, georganiseerde criminaliteit: een vergelijking tussen Nederland en enkele andere landen, in: C.J.C.F. Fijnaut (ed.), *Georganiseerde misdaaden strafrechtelijk politiebeleid; rechtshistorische, politieke, journalistieke en rechtsvergelijkende beschouwingen*, J.B. van den Brink, Lochem, 1989. (In Dutch)
- LEONARD-BARTON, D., H.K. BOWEN, K.B. CLARK, C.A. HOLLOWAY, AND S.C. WHEELWRIGHT, How to Integrate Work and Deepen Expertise, *Harvard Business Review*, Vol. 72, No. 5, pp. 121-130, 1994.
- LEWIN, A.Y., AND C.U. STEPHENS, Designing post-industrial organizations: theory and practice, in: Huber, G.P., and W.H. Glick (eds), *Organization Change and Redesign: Combining Ideas and Insights for Improving Managerial Performance*, pp. 393-409, Oxford University Press, New York, 1993.
- LEWIS, P.S., AND T.A. LEWIS, Using GroupSystems V to Facilitate Strategic Planning, in: *From Shared Visions to Shared Realities - Proceedings of the Fifth Annual GroupSystems Users' Conference*, Tucson, Arizona, March 14-16, 1994.
- LEWIS, E., AND C. NEWTON, The Use of the Grouputer in Preparing Information Management Plans, in: G.J. de Vreede, and H.G. Sol (eds), *Proceedings of the INFORMS'95 Sessions on GDSS as Support for Designing Organizations and Information Systems*, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, 1995.
- LEYENBURG MANAGEMENT, *Masterplan Poliklinieken Ziekenhuis Leyenburg - Voorontwerp*, Internal report, Leyenburg Hospital, The Hague, 1991.
- LIKERT, R., *New Patterns of Management*, McGraw-Hill, New York, 1961.
- LIMAYEM, M., J.E. LEE-PATRIDGE, G.W. DICKSON, AND G. DESANTIS, Enhancing GDSS Effectiveness: Automated versus Human Facilitation, in: Nunamaker, J.F. Jr., and R.H. Sprague (eds), *Proceedings of the 26th Hawaiian International Conference on System Sciences*, Volume IV, pp. 95-101, IEEE Computer Society Press, Los Alamitos CA, 1993.
- LISSENBERG, E., (De)criminaliseren, in: C. Bouw, H. van de Bunt, Herman Franke, *Kernbegrippen in de criminologie*, Gouda Quint BV, Arnhem, The Netherlands, 1992. (In Dutch)
- LISTER, A.M., *Fundamentals of Operating Systems - 3rd edition*, MacMillan Education, London, 1984.

- LOVEMAN, G.W., *An Assessment of the Productivity Impact of Information Technologies*, Working paper 90s:88-054, MIT Sloan School of Management, Cambridge, Massachusetts, 1988.
- LUCAS, H.C. JR., AND M. OLSON, The Impact of Information Technology on Organizational Flexibility, *Journal of Organizational Computing*, Vol 4, No. 2, pp. 155-176, 1994.
- LUNDQUIST, E.T., AND M.M. HUSTON, Information-rich environments for continuous organic development in organisations: research in progress, *Journal of Applied Systems Analysis*, Vol. 17, pp. 79-87, 1990.
- MALONE, T.W., Modelling Coordination in Organizations and Markets, *Management Science*, Vol. 33, No. 10, pp. 1317-1332, October 1987.
- MALONE, T.W., AND K. CROWSTON, What is coordination theory and how can it help design cooperative work systems?, in: *Proceedings of the Conference on Computer-Supported Cooperative Work*, pp. 357-370, Association for Computing Machinery, New York, 1990.
- MALONE, T.W., AND K. CROWSTON, The Interdisciplinary Study of Coordination, in: *ACM Computing Surveys*, vol. 26, no. 1, march 1994.
- MALONE, T.W., AND J.F. ROCKART, Computers, Networks and the Corporation, *Scientific American*, Vol. 265, No. 3, pp. 92-99, 1991.
- MALONE, T.W., AND S.A. SMITH, Modelling the Performance of Organizational Structures, *Operations Research*, Vol. 36, No. 3, pp. 421-436, 1988.
- MALONE, T.W., J. YATES, AND R.I. BENJAMIN, Electronic Markets and Electronic Hierarchies, *Communications of the ACM*, Vol. 30, pp. 484-497, 1987.
- MARCH, J.G., AND H.A. SIMON, *Organizations*, John Wiley & Sons, Inc., New York, 1958.
- MARCH, J.G., AND H.A. SIMON, *Organizations - 2nd Edition*, Blackwell Publishers, Cambridge MA, USA, 1993.
- MARKUS, M.L., Case Selection in a Disconfirmatory Case Study, in: Cash, J.I., and P.R. Lawrence (eds), *The Information Systems Research Challenge: Qualitative Research Methods*, Harvard Business School, Boston, Massachusetts, 1989.
- MARTZ, W.B., D.R. VOGEL, AND J.F. NUNAMAKER JR., Electronic Meeting Systems - Results from the Field, *Decision Support Systems*, Vol. 8, No. 2, pp. 141-158, 1992.
- MCGOFF, C.J., AND L. AMBROSE, Empirical information from the field: A practitioners' view of using GDSS in business, in: J.F. Nunamaker Jr. (eds), *Proceedings of the 24th Hawaiian International Conference on System Sciences*, Volume III, pp. 805-811, IEEE Computer Society Press, Los Alamitos CA, 1991.
- MCGRATH, J.E., *Groups: Interaction and Performance*, Prentice Hall, Inc., Englewood Cliffs, NJ, 1984.
- MCGRATH, J.E., AND A.B. HOLLINGSHEAD, Putting the "Group" Back in Group Support Systems: Some Theoretical Issues About Dynamic Processes in Groups with Technological Enhancements, in: Jessup, L.M., J.S. Valacich (eds), *Group Support Systems - New Perspectives*, Macmillan Publishing Company, New York, 1993.
- MCGREGOR, D., *The Human Side of Enterprise*, McGraw-Hill, New York, 1960.
- MCKENNEY, J.L., AND F.W. MCFARLAN, The Information Archipelago - Maps and Bridges, *Harvard Business Review*, Vol. 60, No. 5, pp. 109-119, 1982.
- MEEL, J.W. VAN, *Dynamics in Business Engineering*, Doctoral dissertation, Delft University of Technology, Delft, The Netherlands, 1994.
- MEEL, J.W. VAN, G.J. DE VREEDE, AND H.G. SOL, *Projectmatige aanpak van georganiseerde criminaliteit - Bedrijfsprocessen Huidige Situatie*, Internal report, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, Delft 1993. (In Dutch)
- MEEL, J.W. VAN, G.J. DE VREEDE, AND H.G. SOL, *Projectmatige aanpak van georganiseerde criminaliteit - Eindrapport*, Internal report, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, Delft 1994. (In Dutch)
- MEER, K. VAN DER, *Documentaire Informatiesystemen (2nd edition)*, NBLC Uitgeverij, Den Haag, The Netherlands, 1994. (In Dutch)

- MEINSMAN, R.R., C.J.A.M. TERMEER, AND G.J. DE VREEDE, *A Game Experiment to Explore the Impact of Simulation Models on Communication Processes*, Internal report, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, Delft, 1995.
- MEYER, B., *Object Oriented Software Construction*. Prentice-Hall, 1988.
- MIDDELBURG, B., *De dominee: opkomst en ondergang van mafiabaas Klaas Bruinsma*, L.J. Veen, Amsterdam, 1992.
- MIDDELBURG, B., AND K. VAN ES, *Operatie Delta - Hoe de drugsmafia het IRT opblies*, L.J. Veen, Amsterdam, 1994. (In Dutch)
- MINISTRY OF JUSTICE, *Samenleving en criminaliteit. Een beleidsplan voor de komende jaren*, The Hague, The Netherlands, 1985. (In Dutch)
- MINTZBERG, H., *The Nature of Managerial Work*, Harper & Row, New York, 1973.
- MINTZBERG, H., *The Structuring of Organizations*, Prentice-Hall, Englewood cliffs, NJ, 1979.
- MINTZBERG, H., *Structures in Fives*, Prentice-Hall, Englewood cliffs, NJ, 1983.
- MITROFF, I.I., F. BETZ, L.R. PONDLY AND F. SAGASTY, On Managing Science in the Systems Age: Two Schemas for the Study of Science as a Whole Systems Phenomenon, *TIMS Interfaces*, Vol. 4, No. 3, pp. 46-58, 1974.
- MONGE, P.R., AND E.M. EISENBERG, Emergent Communication Networks, in: Jablin, F.M., L.L. Putnam, K.H. Roberts, and L.W. Porter (eds.), *Handbook of Organizational Communication - An Interdisciplinary Perspective*, Sage Publications, London, 1987.
- MOONEY, J.D., The Principles of Organization, in: Gulick, L., and L. Urwick (eds.), *Papers on the Science of Administration*, Institute of Public Administration, New York, 1937.
- MOORE, C.M., *Group Techniques for Idea Building - Second Edition*, Sage Publications, London, 1994.
- MORGAN, G., *Beyond Method: Strategies for Social Research*, Sage, New York, 1983.
- MORRISON, J., AND D.R. VOGEL, Group Support From a Task-Based Perspective, in: A.A. Verrijn-Stuart, H.G. Sol, and P. Hammersley (eds.), *Support Functionality in the Office Environment*, Elsevier Science Publishers B.V., North Holland, Amsterdam, 1991.
- MORTON, M.S.S., Reshaping the Corporation for the Future - Managing the Organizational Challenge Successfully, *Revue des systèmes de décision*, Vol. 1, No. 4, pp. 341-352, 1992.
- MOTSHAGEN, P.A., *Treasury Management en In-house Banking: Een informatiekundige analyse*, Dissertation, Delft University of Technology, 1991. (in Dutch)
- MULLER, P.C., AND G.J. DE VREEDE, Supporting the Conceptualization of a New Telecommunication Service with Group Decision Support, in: *Proceedings of the GroupSystems' Sixth Annual Users Conference*, 13-16 March, Tucson AZ, USA, 1995a.
- MULLER, P.C., AND G.J. DE VREEDE, Improving Interaction in New Product Development: A Case Study, in: G.J. de Vreede, and H.G. Sol (eds), *Proceedings of the INFORMS'95 Sessions on GDSS as Support for Designing Organizations and Information Systems*, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, 1995b.
- MUMFORD, E., *Designing Human Systems for New Technology - the ETHICS method*, Manchester Business School, 1983.
- NATIONAL CRIMINAL INTELLIGENCE SERVICE, *Georganiseerde criminaliteit en interregionale groepsriminaliteit*, The Hague, The Netherlands, 1988. (In Dutch)
- NATIONAL CRIMINAL INTELLIGENCE SERVICE, *Jaarverslag 1991*, The Hague, The Netherlands, 1992. (In Dutch)
- NEWSWEEK, Global Mafia: A New and Dangerous Criminal Threat, Vol. 122, No. 24, pp. 12-18, 1993.
- NUNAMAKER, J.F. JR., L.M. APPEGATE, AND B.R. KONSYNSKI, Computer-aided Deliberation: Model Management and Group Decision Support, *Journal of Operations Research*, Vol. 36, No. 6, pp. 826-848, 1988.
- NUNAMAKER, J.F. JR., AND R.O. BRIGGS, Groupware User Experience: Ten Years of Lessons with GroupSystems, in *Proceedings of Groupware 1994*, Boston, 1994.

- NUNAMAKER, J.F. JR., R.O. BRIGGS, AND N.C. ROMANO JR., Meeting Environments of the Future: Meeting to Plan Work or Meeting to Do Work?, in: Coleman, D., and P.R. Huckle (eds.), *Proceedings of the GroupWare '94 Europe Conference*, 6-8 June, pp. 521-546, London, 1994.
- NUNAMAKER, J.F. JR., M. CHEN, AND T.D.M. PURDIN, Systems Development in Information Systems Research, *Journal of Management Information Systems*, Vol. 7, No. 3, pp. 89-106, 1990.
- NUNAMAKER, J.F. JR., A.R. DENNIS, J.S. VALACICH, D.R. VOGEL, AND J.F. GEORGE, Group Support Systems Research: Experiences from the Lab and Field, in: Jessup, L.M., J.S. Valacich (eds), *Group Support Systems - New Perspectives*, Macmillan Publishing Company, New York, 1993.
- NUNAMAKER, J.F. JR., A.R. DENNIS, J.S. VALACICH, D.R. VOGEL, AND J.F. GEORGE, Electronic Meeting Systems to Support Group Work, *Communications of the ACM*, Vol. 34, No. 7, pp. 40-61, 1991.
- NUNAMAKER, J.F. JR., D.R. VOGEL, A. HEMINGER, B. MARTZ, R. GROHOWSKI, AND C. MCGOFF, Experiences at IBM with Group Support Systems: A Field Study, *Decision Support Systems*, 5:2, pp. 183-196, 1989.
- OCID (OPERATIONAL CRIMINAL INVESTIGATIONS DEPARTMENT), *Onderzoeksplan*, Amsterdam Municipal Police Force, 1992. (In Dutch)
- OPPER, S., AND H. FERSKO-WEISS, *Technology for Teams: Enhancing Productivity in Networked Organizations*, Van Nostrand Reinhold, New York, 1992.
- ÖREN, T.J., AND B.P. ZEIGLER, Concepts for Advanced Simulation Methodologies, *Simulation*, Vol. 32, No. 3, pp. 69-82, 1979.
- ORLIKOWSKI, W.J., The Duality of Technology: Rethinking the Concept of Technology in Organizations, *Organization Science*, Vol. 3, No. 3, pp. 398-427, 1992.
- ORLIKOWSKI, W.J., AND J.J. BAROUDI, Studying Information Technology in Organizations: Research Approaches and Assumptions, *Information Systems Research*, Vol. 2, No. 1, pp. 1-28, 1991.
- ORLIKOWSKI, W.J., AND D. ROBEY, Information Technology and the Structuring of Organizations, *Information Systems Research*, Vol. 2, No. 2, pp. 143-169, 1991.
- OSBORN, A.F., *Applied Imagination*, Scribes, New York, NY, 1957.
- OXFORD DICTIONARY, *The Concise Oxford Dictionary*, Oxford University Press, Oxford 1982.
- PALYS, T.S., Addressing the "Third Criterion" in Experimentalist Research: Towards a Balance of Manipulative and Analytic Control, in: Benbasat, I. (ed.), *The Information Systems Research Challenge: Experiment Research Methods*, pp. 133-146, Harvard Business School, Boston, Massachusetts, 1989.
- PANKO, R.R., 38 Offices: Analyzing Needs in Individual Offices, *ACM Transactions on Office Information Systems*, Vol. 2, No. 3, pp. 226-234, 1984.
- PANKO, R.R., Managerial Communication Patterns, *Journal of Organizational Computing*, Vol. 2, No. 1, pp. 95-122, 1992.
- PARKER FOLLETT, M., The Process of Control, in: Gulick, L., and L. Urwick (eds.), *Papers on the Science of Administration*, Institute of Public Administration, New York, 1937.
- PATERNOTTE, F.J.M., AND H.W. LIGTERINGEN, *Project Urenregistratie*, Amsterdam Municipal Police Force, Amsterdam, 1994. (In Dutch)
- PEGDEN, C.D., R.E. SHANNON, AND R.P. SADOWSKI, *Introduction to Simulation using SIMAN*, McGraw-Hill, 1990.
- PEGDEN, C.D., R.E. SHANNON, AND R.P. SADOWSKI, *Introduction to Simulation using SIMAN - 2nd Edition*, McGraw-Hill, 1995.
- PERVAN, G.P., A Case for More Case Study Research in Group Support Systems, in: Glasson, B.C., I.T. Hawryszkiewicz, B.A. Underwood, and R.A. Weber (eds.), *Business Process Re-engineering: Information Systems Opportunities and Challenges*, Elsevier Science Publishers (North-Holland), pp. 469-479, 1994.
- PETRI, C.A., Introduction to general net theory, in: W. Brauer, (ed.), *Net theory and applications, proceedings of the advanced course on general net theory, processes and systems*, vol.84 of lecture notes in computer science, Springer-Verlag, New York, 1980.

- PETTIGREW, A.M., Issues if Time and Site Selection in Longitudinal Research on Change, in: Cash, J.I., and P.R. Lawrence (eds), *The Information Systems Research Challenge: Qualitative Research Methods*, pp. 13-19, Harvard Business School, Boston, Massachusetts, 1989.
- PINSONNEAULT, A., AND K.L. KRAEMER, The impact of technological support on groups: An assessment of the empirical research, *Decision Support Systems*, Vol. 5, No. 2, pp. 197-216, 1989.
- POPPEL, H.L., Who Needs The Office of The Future?, *Harvard Business Review*, Vol. 60, No. 6, pp. 146-155, 1982.
- QUINN, J.B., H. MINTZBERG AND R.M. JAMES, *The Strategy Process*, Prentice-Hall International, Englewood Cliffs, 1988.
- QURESHI, S.S., *Organisations and Networks: Theoretical Considerations and a Case Study of Networking Across Organisations*, Doctoral dissertation, London School of Economics, London, 1995.
- QURESHI, S.S., AND T. CORNFORD, Networking and Development: The COMNET-IT project, in: R. Baskerville, S. Smithson, O. Ngwenyama, and J.I. DeGross (eds), *IFIP Transactions on Transforming Organisations with Information Technology*, pp. 419-436, North Holland, Amsterdam, 1994.
- RAMAN, K.S., AND K.K. WEI, The GDSS Research Project, in: Bostrom, R.P., R.T. Watson and S.T. Kinney (eds.), *Computer Augmented Teamwork, A Guided Tour*, Van Nostrand Reinhold, New York, 1992.
- RAMONDT, J.J. AND A. RHÉBERGEN, *Verslag vooronderzoek Leyenburg*, Rotterdams Instituut voor Sociologisch en Bestuurskundig Onderzoek (RISBO), Rotterdam, 1991a. (In Dutch)
- RAMONDT, J.J. AND A. RHÉBERGEN, *Ziekenhuis Leyenburg, Researchmemorandum 1*, Rotterdams Instituut voor Sociologisch en Bestuurskundig Onderzoek (RISBO), Rotterdam, 1991b. (In Dutch)
- RANDOLPH, W.A., Organizational Technology and the Media and Purpose Dimensions of Organization Communications, *Journal of Business Research*, Vol. 6, pp. 237-259, 1978.
- REISIG, W., *Petri nets: an introduction*, Prentice-Hall, Englewood Cliffs, 1985.
- REUTER, P., *Disorganized crime. Illegal markets and the Mafia*, Cambridge, Massachusetts, MIT Press, 1983.
- REUYL, J.C., *On the Determination of Advertising Effectiveness: An Empirical Study of the German Cigarette Market*, Stenfert Kroese, Leiden, The Netherlands, 1982.
- RIESSEN, J.C. VAN, *Visie op recherchefunctie*, Amsterdam Municipal Police Force, 1991. (In Dutch)
- ROACH, S.S., Services Under Siege - The Restructuring Imperative, *Harvard Business Review*, Vol. 69, No. 5, pp. 82-91, 1991.
- ROBEY, D., Computer Information Systems and Organization Structure, *Communications of the ACM*, Vol. 24, No. 10, pp. 679-687, 1981.
- ROBEY, D., Information Systems and Organizational Change: A Comparative Case Study, *Systems, Objectives, Solutions*, Vol. 3, pp. 143-154, 1983.
- ROCKART, J.F., Chief executives define their own data needs, *Harvard Business Review*, Vol. 57, No. 2, pp. 81-93, 1979.
- ROCKART, J.F., AND C.V. BULLEN, *The Rise of Managerial Computing*, Dow Jones Irwin, Homewood, 1986.
- ROER, R. V.D., De miljoenengok van Ger van D.V., in: C.J.C.F. Fijnaut (ed.), *Georganiseerde misdaad en strafrechtelijk politiebeleid: rechtshistorische, politieke, journalistieke en rechtsvergelijkende beschouwingen*, J.B. van den Brink, Lochem, 1989. (In Dutch)
- ROGERS, D.L., AND D.A. WHETTEN (EDS), *Interorganizational Coordination - Theory, Research, and Implementation*, Iowa State University Press, Ames, 1982.
- ROHRBAUGH, J., Demonstration Experiments in Field Settings: Assessing the Process, Not the Outcome, of Group Decision Support, in: Benbasat, I. (ed.), *The Information Systems Research Challenge: Experiment Research Methods*, pp. 113-130, Harvard Business School, Boston, Massachusetts, 1989.
- ROSS, S., The Economic Theory of Agency, *American Economist Review*, Vol. 63, pp. 134-139, 1973.
- RUMBAUGH, J., M. BLAHA, W. PREMERLANI, F. EDDY, AND W. LORENSEN, *Object-Oriented Modeling and Design*, Prentice-Hall, 1991.
- SCHRIJVER, P.R., *Supporting fleet Management by Mobile Communications*, Dissertation, Delft University of Technology, 1993.

- SCID (SUPPORTIVE CRIMINAL INVESTIGATIONS DEPARTMENT), *Onderzoeksplan Reorganisatie*, Amsterdam Municipal Police Force, 1992. (In Dutch)
- SENGE, P.M., AND J.D. STERMAN, Systems thinking and organizational learning: Acting locally and thinking globally in the organization of the future, *European Journal of Operational Research*, Vol. 59, pp. 137-150, 1992.
- SHANNON, R.E., *Systems Simulation, the Art and Science*. Prentice-Hall, 1975.
- SIMON, H.A., *Administrative Behavior*, Macmillan, New York, NY, 1947.
- SIMON, H.A., The Structure of Ill Structured Problems, *Artificial Intelligence*, Vol. 4, pp. 181-201, 1973.
- SIMON, H.A., Applying Information Technology to Organization Design, *Public Administration Review*, Vol. 33, No. 3, pp. 268-278, 1973.
- SIMON, H.A., *The Sciences of the Artificial - 2nd edition*, MIT Press, Cambridge, MA, 1981.
- SMITH, R.L. AND L. PLATT, Benefits of animation in the simulation of a machining and assembly line, *Simulation*, Vol. 48, No. 1, pp. 28-30, 1987.
- SMITHSON, S., Combining Different Approaches: Discussant's Comments, in: Nissen, H.-E., H.K. Klein, and R. Hirschheim (eds.), *Information Systems Research: Contemporary Approaches and Emergent Traditions*, Elsevier Science Publishers (North-Holland), pp. 327-345, 1991.
- SOL, H.G., *Simulation In Information Systems Development*, Doctoral dissertation, University of Groningen, Groningen, The Netherlands 1982.
- SOL, H.G., Prototyping: A Methodological Assessment, in: Budde, R., K. Kuhlkamp, L. Mathiassen en H. Zullighoven (eds.), *Approaches to Prototyping*, pp. 368-382, Springer-Verlag, Berlin, 1984.
- SOL, H.G., Information Systems Development: A Problem Solving Approach, in: *Proceedings of the International Symposium on System Development Methodologies*, Atlanta 1990.
- SOL, H.G., *Shifting Boundaries in Systems Engineering and Policy Analysis*, Inaugural Address, Delft University of Technology, Delft, 1992a.
- SOL, H.G., Dynamics in Information Systems, in: Sol, H.G., and R.L. Crosslin (eds.), *Dynamic Modelling of Information Systems, II*, North-Holland, Amsterdam 1992b.
- SOL, H.G., AND P.W.G. BOTS, Information Systems to Support Decision Processes: From Decision Support to Networks of Inquiry Systems, in: Holsapple, C.W., and A.B. Winston (eds.), *Recent Developments in Decision Support*, Springer-Verlag, Berlin 1992.
- SOL, H.G., AND R.L. CROSSLIN (EDS.), *Dynamic Modelling of Information Systems, II*, North-Holland, Amsterdam, The Netherlands, 1992.
- SOL, H.G., AND K.M. VAN HEE (EDS.), *Dynamic modelling of information systems*, North-Holland, Amsterdam, The Netherlands, 1991.
- SPECHT, P.H., AND D. ROBEY, Organization design and Information Systems Design: An Integrative Model Based on Information Processing Requirements, *International CIS Journal*, No. 1, pp. 6-13, 1988.
- SPEK, R. VAN DER, A. VERBRAECK, R.J. STRENG, Modelling Interorganizational Processes, in: *Proceedings of the European Simulation Multiconference (ESM'94)*, Barcelona, Spain, 1994.
- SPROULL, L., AND S. KIESLER, Reducing Social Context Cues: The Case of Electronic Mail, *Management Science*, Vol. 32, pp. 1492-1512, 1986.
- STRAATMAN, M., *Chain Management - Interorganizational production in the public sector*, Working paper, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, 1995.
- STRENG, R.J., *Dynamic Modelling to Assess the Value of Electronic Data Interchange - A Study in the Rotterdam Port Community*, Doctoral dissertation, Delft University of Technology, Delft, The Netherlands, 1993.
- SUSMAN, G., AND R. EVERED, An Assessment of the Scientific Merits of Action Research, *Administrative Science Quarterly*, Vol. 23, No. 4, pp. 582-603, 1978.
- SWAIN, J., Simulation Software Survey, *OR/MS Today*, pp. 81-102, October 1991.
- TACTICAL DEPARTMENT, *Jaarverslag 1993*, Amsterdam Municipal Police Force, Amsterdam, 1993. (In Dutch)

- TANENBAUM, A.S., *Structured Computer Organization - 2nd edition*, Prentice Hall International, Englewood Cliffs, NJ, USA, 1984.
- TASK FORCE CRIME ANALYSIS, *Georganiseerde misdaad in Nederland. Verslag van de landelijke inventarisatie georganiseerde-en groeps criminaliteit 1991*, National Criminal Investigation Committee, The Hague, 1991. (In Dutch)
- TASK FORCE 'KIT IN DE LIJT', *De kit in de lift. De Centrale Recherche, een nieuw perspectief*, Amsterdam Municipal Police Force, 1986. (In Dutch)
- TASK FORCE OBJECTIVES, *Doelstellingen Dienst Centrale Executieve en Ondersteunende Recherche Jaren 90*, Amsterdam Municipal Police Force, 1992. (In Dutch)
- TASK FORCE 'ZELFBESTUUR & RESULTAATVERANTWOORDELIJKHEID', *Initiatieffase*, Amsterdam Municipal Police Force, Amsterdam, 1994a. (In Dutch)
- TASK FORCE 'ZELFBESTUUR & RESULTAATVERANTWOORDELIJKHEID', *Definitiefase*, Amsterdam Municipal Police Force, Amsterdam, 1994b. (In Dutch)
- THOMPSON, G., J. FRANCES, R. LEVAČIĆ AND J. MITCHELL, *Markets, Hierarchies & Networks, The coordination of social life*, Sage Publications, London, 1991.
- THOMPSON, J.D., *Organizations in Action*, McGraw-Hill, New York, 1967.
- TODD, P., AND I. BENBASAT, Process Tracing Methods in Decision Support Systems Research: Exploring the Black Box, *MIS Quarterly*, Vol. 11, No. 4, pp. 493-512, 1987.
- TOFFLER, A., *Future Shock*, Random House, 1970.
- TOFFLER, A., *The Adaptive Corporation*, Growen Publishing Company Ltd., Aldershot, Great Britain, 1985.
- TRAUTH, E.M., D.W. FARWELL, AND D. LEE, The IS Expectation Gap: Industry Expectations Versus Academic Preparation, *MIS Quarterly*, Vol. 17., No. 3, pp. 293-303, 1993.
- TSALGATIDOU, A., D. GOUSCOS, AND C. HALATSIS, Dynamic Process Modelling through Multi-Level RBNs, in: A. Verbraeck, H.G. Sol, and P.W.G. Bots (eds.), *Proceedings of the Fourth International Working Conference on Dynamic Modelling and Information Systems*, pp. 327-341, Delft University Press, 1994.
- TYRAN, C.K., A.R. DENNIS, D.R. VOGEL AND J.F. NUNAMAKER JR., The Application of Electronic Meeting Technology to Support Strategic Management, *MIS Quarterly*, Vol. 16, No. 3, pp. 313-334, 1992.
- TYRE, M.J., AND W.J. ORLIKOWSKI, Windows of Opportunity: Temporal Patterns of Technological Adaptation in Organizations, *Organization Science*, Vol. 5, No. 1, pp. 98-118, 1994.
- URWICK, L., Organization as a Technical Problem, in: Gulick, L., and L. Urwick (eds.), *Papers on the Science of Administration*, Institute of Public Administration, New York, 1937.
- VALACICH, J.S., A.R. DENNIS, AND T. CONNOLLY, Idea Generation in Computer-Based Groups: A New Ending to an Old Story, *Organizational Behavior and Human Decision Processes*, Vol. 57, pp. 448-467, 1994.
- VAN DE VEN, A.H., AND A. DELBECQ, The effectiveness of nominal, Delphi, and interacting group decision making processes, *Academy of Management Journal*, Vol. 17, pp. 605-621, 1974.
- VAN DE VEN, A.H., A. DELBECQ, AND R. KOENIG, Determinants of Coordination Mode within Organizations, *American Sociological Review*, Vol. 41, pp. 322-338, 1976.
- VAN HORN, R.L., Empirical Studies of Management Information Systems, *Data Base*, Vol. 5, pp. 172-182, 1973.
- VANGUNDY, A.B., *Techniques of Structured Problem Solving - Second Edition*, Van Nostrand Reinhold, New York, 1988.
- VEN, M.B.M. VAN DER, *A Group Decision Support Systems for International Transfer Pricing in the Pharmaceutical Industry*, Doctoral Dissertation, Delft University of Technology, The Netherlands, 1989.
- VENNIX, J.A.M., W. SCHEPER, AND R. WILLEMS, Group model-building: what does the client think of it?, in: *Proceedings of the 1993 Systems Dynamics Conference*, Cancun, Mexico, 1993.
- VENTANA CORPORATION, *GroupSystems V - Manual Basic Tools*, Tucson, Arizona, 1993a.
- VENTANA CORPORATION, *GroupSystems V - Manual Advanced Tools*, Tucson, Arizona, 1993b.
- VENTANA CORPORATION, *TeamGraphics Manual Version 1.0*, Tucson, Arizona, 1993c.

- VERBRAECK, A., *Developing an adaptive scheduling support environment*, Dissertation, Delft University of Technology, Delft 1992.
- VERBRAECK, A., AND G.J. DE VREEDE, Animation as a Communication Instrument in Simulation Studies, in: Pave, A. (ed.), *Proceedings of the European Simulation Multiconference (ESM'93)*, pp. 670-674, Society for Computer Simulation, San Diego, 1993.
- VERBRAECK, A., AND G.J. DE VREEDE, *Animating Business Processes: Insight Eases Change*, paper presented at TIMS XXXII, Anchorage, Alaska, USA, June 12-15, 1994.
- VERBRAECK, A., G.J. DE VREEDE, P.W.G. BOTS, AND H.G. SOL, *A General Approach to Solving the Problem of Organizational Change*, Internal report, Faculty of Technical Mathematics and Informatics, Delft University of Technology, Delft, 1995.
- VICIAN, C., G. DESANTIS, M.S. POOLE, AND B.M. JACKSON, Using Group Technologies to Support the Design of "Lights Out" computing Systems: A Case Study, in: Kendall, K.E. et al. (eds.), *The Impact of Computer Supported Technologies on Information Systems Development*, Elsevier Science Publishers BV, North-Holland, 1992.
- VITALARI, N.P. AND G.W. DICKSON, Problem Solving for Effective Systems Analysis: An Experimental Exploration, *Communications of the ACM*, Vol. 26, No. 11, pp. 948-956, 1983.
- VOGEL, D.R., AND J.F. GEORGE, Group Support for Analysis and Design, *Informatica, The Journal of Management Information Systems Development*, Vol. 1, No. 1, pp. 96-114, 1992.
- VOGEL, D.R., AND J.F. NUNAMAKER JR., Group Decision Support System Impact: Multi-Methodological Explorations, *Information and Management*, Vol. 18, pp. 15-28, 1990.
- VOGEL, D.R., J.F. NUNAMAKER JR., L. APLEGATE, AND B. KONSYNSKI, Group Decision Support Systems: Determinants of Success, in: *DSS-87 Transactions of the Seventh International Conference on Decision Support Systems*, pp. 118-128, June, 1987.
- VOGEL, D.R., R. ORWIG, D. DEAN, J. LEE, AND C. ARTHUR, Reengineering with Enterprise Analyzer, in: J.F. Nunamaker Jr., and R.H. Sprague (eds), *Proceedings of the Twenty-Sixth Annual Hawaiian International Conference on Systems Science (HICSS)*, pp. 127-136, IEEE Computer Society Press, Los Alamitos, California, 1993.
- VOGEL, D.R., AND J.C. WETHERBE, MIS Research: A Profile of Leading Journals and Universities, *Data Base*, vol. 16, No. 3, 1984.
- VOLPATO, M.C., Decision Conferencing: An Organizational Teamworking Tool, in: *DSS-89 Transactions of the Ninth International Conference on Decision Support Systems*, 1989.
- VONK, R., *Prototyping van informatiesystemen*, Academic Service, The Hague, The Netherlands, 1987. (In Dutch)
- VREEDE, G.J. DE, *Alternatieve Coördinatiemechanismen in Overvol*, Internal report, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, Delft 1992a. (In Dutch)
- VREEDE, G.J. DE, *Simulatie Financiële Stroom Leyenburg*, Internal report, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, Delft, 1992b. (In Dutch)
- VREEDE, G.J. DE, *Dynamisch Modelleren en Coördinatie in Ziekenhuis Leyenburg*, Internal report, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, Delft, 1993a. (In Dutch)
- VREEDE, G.J. DE, Improving Coordination within Organizations through Information Technology - Handling Patients and Combating Organized Crime, *Contribution to the Doctoral Consortium at the Fourteenth International Conference on Information Systems (ICIS'93)*, 3-5 December, Florida, 1993b.
- VREEDE, G.J. DE, *De Praktijk van de Onderhoudsautomatisering*, Lansa Publishing, Rijswijk, 1993c. (In Dutch)
- VREEDE, G.J. DE, Ondersteuning voor verspreid werkende groepen, *Research memorandum*, School of Systems Engineering, Policy Analysis, and Management, Delft University of Technology, Delft, The Netherlands, 1994. (In Dutch)

- VREEDE, G.J. DE, Participative Modelling for Understanding: Facilitating Organizational Change with GSS, in: J.F. Nunamaker Jr., R.H. Sprague Jr. (eds.), *Proceedings of the 29th Hawaiian International Conference on Systems Sciences*, IEEE Computer Society Press, 1996. (in press)
- VREEDE, G.J. DE, P.W.G. BOTS, AND A. VERBRAECK, Simulation as a Tool to Improve Coordination Within A Service Organization, in: Verbraeck, A., and E. Kerckhoffs (eds), *Proceedings of the European Symposium on Simulation (ESS'93)*, 25-28 October, Delft, 1993a.
- VREEDE, G.J. DE, AND D.T.T. VAN EIJCK, Modelling Organizational Coordination, in: A. Verbraeck, H.G. Sol, and P.W.G. Bots (eds.), *Proceedings of the Fourth International Working Conference on Dynamic Modelling and Information Systems*, pp. 303-325, Delft University Press, 1994.
- VREEDE, G.J. DE, AND D.T.T. VAN EIJCK, Modelling Organizational Coordination, *Simulation & Gaming*, 1996. (in press).
- VREEDE, G.J. DE, D.T.T. VAN EIJCK, AND H.G. SOL, Dynamisch Modelleren voor het Ontwerpen van Organisaties, in: Hertog, J.F. and J.J. Ramondt (eds.), *Competente Vernieuwers*, Merit, Maastricht, 1994. (In Dutch)
- VREEDE, G.J. DE, D.T.T. VAN EIJCK, AND H.G. SOL, Dynamic Modelling in Practice - The Application of Dynamic Modelling in Two Service Organizations, in: *Proceedings of the Third European Conference on Information Systems (ECIS'95)*, June 1-3, June, Athens, Greece, 1995a.
- VREEDE, G.J. DE, D.T.T. VAN EIJCK, AND H.G. SOL, Dynamic Modelling for Re-engineering Organizations, *Journal of Information Systems and Operations Research (INFOR)*, 1996. (in press)
- VREEDE, G.J. DE, S.O. DEN HENGST, AND H.G. SOL, Facilitating User Involvement in Information System Design and Development with GDSS - The Organized Crime Case, in: Olman, L. (ed.), *Proceedings of the ACM SIGCPR Conference*, Nashville Tennessee, 6-8 April, 1995b.
- VREEDE, G.J. DE, S.O. DEN HENGST, AND H.G. SOL, Participative Modelling for Understanding - The Case at the Criminal Investigations Department, in: *Proceedings of the GroupSystems' Sixth Annual Users Conference*, 13-16 March, Tucson AZ, USA, 1995c.
- VREEDE, G.J. DE, S.O. DEN HENGST, G.F.S.J. KUIJLAARS, AND H.G. SOL, *Onderweg naar een projectmatige aanpak van de georganiseerde criminaliteit - Conceptualisatie en specificatie huidige situatie*, Internal report, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, Delft 1995d. (In Dutch)
- VREEDE, G.J. DE, A.H. LEEUWRIK, AND S.O. DEN HENGST, *Management 1 - Projecten en trends*, Results of the GroupSystems session, Amsterdam Municipal Police Force, Amsterdam, 1993b. (In Dutch)
- VREEDE, G.J. DE, A.H. LEEUWRIK, AND S.O. DEN HENGST, *Bestuurlijke vernieuwing en Georganiseerde Criminaliteit*, Results of the GroupSystems session, Amsterdam Municipal Police Force, Amsterdam, 1993c. (In Dutch)
- VREEDE, G.J. DE, A.H. LEEUWRIK, AND S.O. DEN HENGST, *Landelijke coördinatie tegen georganiseerde criminaliteit*, Results of the GroupSystems session, Amsterdam Municipal Police Force, Amsterdam, 1993d. (In Dutch)
- VREEDE, G.J. DE, AND H.G. SOL, GroupSystems V as Robocop IV? Combating Organized Crime with GroupSystems, in: *Proceedings of the GroupSystems' Fifth Annual Users Conference*, 14-16 March, Tucson AZ, USA, 1994a.
- VREEDE, G.J. DE, AND H.G. SOL, Combating Organized Crime with Groupware - Facilitating User Involvement in Information System Development, in: Coleman, D., and P.R. Huckle (eds.), *Proceedings of the GroupWare '94 Europe Conference*, 6-8 June, London, 1994b.
- VREEDE, G.J. DE, AND H.G. SOL, *Ontwerpen van Informatiesystemen volgens de GRIPPER-benadering - Aanpak en ervaringen bij de Gemeentepolitie Amsterdam*, Internal report, School of Systems Engineering, Policy Analysis and Management, Delft University of Technology, Delft, 1994c. (In Dutch)
- VREEDE, G.J. DE, H.G. SOL, AND G.W. DICKSON, A Group Problem Solving approach to Business Process Redesign: Combating Organized Crime in Amsterdam, in: J.F. Nunamaker Jr., R.H. Sprague Jr. (eds.), *Proceedings of the 28th Hawaiian International Conference on Systems Sciences*, pp. 493-502, IEEE Computer Society Press, 1995e.

- VREEDE, G.J. DE, AND A. VERBRAECK, Animating Organizational Processes: Insight Eases Change, *Journal of Simulation Practice and Theory*, 1996. (In press)
- WADDINGTON, D., Participant Observation, in: Cassell, C., and G. Symon (eds.), *Qualitative Methods in Organizational Research - A Practical Guide*, Sage Publications, London, 1994.
- WAGNER G.R, B.E. WYNNE, AND B.E. MENNECHE, Group Support Systems Facilities and Software, in: Jessup, L.M., J.S. Valacich (eds), *Group Support Systems - New Perspectives*, Macmillan Publishing Company, New York, 1993.
- WALSHAM, G., AND T. WAEMA, Information Systems Strategy and Implementation: A Case Study of a Building Society, *ACM Transactions on Information Systems*, Vol. 12, No. 2, pp. 150-173, 1994.
- WANNINGER, L.A., AND G.W. DICKSON, GRIP™ - Group Requirement Identification Process: The Integration of GDSS Technology into the Systems Development Life Cycle, in: Nunamaker, J.F. Jr., and R.H. Sprague (eds), *Proceedings of the 25th Hawaiian International Conference on System Sciences*, IEEE Computer Society Press, Los Alamitos CA, 1992a.
- WANNINGER, L.A., AND G.W. DICKSON, Phased systems design, development, and implementation: process and technology, in: *The impact of computer supported technologies on information systems development*, pp. 133-149, Elsevier Science Publishers B.V., 1992b.
- WATSON, R.T., M.B. ALEXANDER, C.E. POLLARD, AND R.P. BOSTROM, Perceptions of Facilitators of a Keypad-Based Group Support System, *Journal of Organizational Computing*, Vol. 4, No. 2, pp. 103-125, 1994.
- WATSON, R.T., R.P. BOSTROM, AND A.R. DENNIS, Fragmentation to Integration, in: Lloyd, P. (ed.), *Groupware in the 21st Century - Computer Supported Cooperative Working Toward the Millennium*, pp. 29-40, Adamantine Press, London, Great Britain, 1994.
- WEILL, P., AND M.H. OLSON, An assessment of the contingency theory of management information systems, *Journal of Management Information Systems*, Vol. 6, No. 1, pp. 59-85, 1989.
- WELTEN, B.J.A.M., *De 4e wereld conjunctie*, Address to the Dutch Secretary of the Treasury, Amsterdam Municipal Police Force, 1992. (In Dutch)
- WELTEN, B.J.A.M., AND B.TH. DE KONINGH, *Visie op de Recherche functie 1994 - Met het oog op morgen*, Amsterdam Municipal Police Force, Amsterdam, 1994. (In Dutch)
- WETHERBE, J.C., Executive Information Requirements: Getting It Right, *MIS Quarterly*, Vol. 15, No. 2, pp. 51-65, 1991.
- WIERDA, F.W., *Developing Interorganizational Information Systems*, Doctoral dissertation, Delft University of Technology, Delft, The Netherlands, 1991.
- WILLIAMSON, O.E., *Markets and Hierarchies: Analysis and Antitrust Implications*, The Free Press, New York, 1975.
- WINOGRAD, T., AND F. FLORES, *Understanding Computers and Cognition: A New Foundation for Design*, Norwood, N.J., 1986.
- WOMACK, J.P., AND D.T. JONES, From Lean Production to the Lean Enterprise, *Harvard Business Review*, Vol. 71, No. 2, pp. 93-103, 1994.
- WOUW, R. VAN DER, *Projectplan Octopus Politie Amsterdam*, Amsterdam Municipal Police Force, 1992. (In Dutch)
- WYNEKOOP, J.L., AND S.A. CONGER, A Review of Computer Aided Software Engineering Research Methods, in: Nissen, H.-E., H.K. Klein, and R. Hirschheim (eds), *Information Systems Research: Contemporary Approaches and Emergent Traditions*, Elsevier Science Publishers (North-Holland), pp. 327-345, 1991.
- YIN, R.K., *Case Study Research: Design and Methods* (revised edition), Sage Publications, Newbury Park, 1989a.
- YIN, R.K., Research Design Issues in Using the Case Study Method to Study Management Information Systems, in: Cash, J.I., and P.R. Lawrence (eds), *The Information Systems Research Challenge: Qualitative Research Methods*, Harvard Business School, Boston, Massachusetts, 1989b.

- ZIGURS, I., Methodological and Measurement Issues in Group Support Systems Research, in: Jessup, L.M., J.S. Valacich (eds), *Group Support Systems - New Perspectives*, Macmillan Publishing Company, New York, 1993.
- ZIGURS, I., G. DESANCTIS, AND J. BILLINGSLEY, Attitudinal development in computer-supported meetings: An exploratory study. *Journal of Management Information Systems*, Vol. 7, No. 4, pp. 51-70, 1990.
- ZMUD, R.W., M.H. OLSON, AND R. HAUSER, Field Experimentation in MIS Research, in: Benbasat, I. (ed.), *The Information Systems Research Challenge: Experiment Research Methods*, pp. 97-111, Harvard Business School, Boston, Massachusetts, 1989.
- ZUBER-SKERRITT, O., *Action Research for Change and Development*, Gower Publishing, Aldershot, 1991.

Author index

*'Why is it the world never remembered the name of Johann
Gambolputty de von Ausfernschplenden-schlitter-crasscrenbon-fried-
digger-dingle-dangle-dongle-dungle-burstein-von-knacker-trasher-apple-
banger-horowitz-ticolensic-grander-knotty-spelltinkle-grandlich-
grumblemeyer-spelterwasser-kurstlich-himbleeisen-bahnwagen-
gutenabend-bitte-ein-nürnberg-er-bratwustle-gerspurten-mitz-
weimache-luber-hunds-fut-gumberaber-shönendanker-kalbsfleisch-
mittler-aucher von Hautkopf of Ulm?'*
- Monty Python

A

Aalst, J.W. van	179
Aalst, W.M.P. van der	40
Ackermann	127
Ackoff	2,26,36-37,42,211
Akkermans	216,257
Alavi	21-22,127
Alexander	3
Ambrose	130
Angell	2
Anson	127,131-133
Anton	56-57
Argyris	10,27,216
Arrow	6
Ashby	2
Atkinson	127
Attewell	15
August	122

B

Baarle	148-149
Baily	2
Bakos	2-3,16
Baroudi	21-22,25-26
Bartlett	2-3
Barton	149
Beer	124,216
Bell	58
Benbasat	25-28
Benjamin	2

Bird Schoonhoven	11
Boehm	143
Booch	42
Borovits	127
Bosman	5,23
Bostrom	14,124,127-128,130-133
Botkin	2
Bots ..	3,13,24,37-38,40-42,48-49,127,136,277
Bouwman	285
Briggs	127,132,141
Broer	221
Brussaard	136
Brynjolfsson	2-3,16
Bullen	140
Burns	2
Burrell	21-22,25-26

C

CABLA	90
Carlson	22
Carmel	133
Carsol	21
Cash	13,25
Chakrabarti	2
Champy	2
Charan	9
Checkland ...	2,27-28,31-32,36,42,45,57,136, 215,267,272
Chen	2,17
Chidambaram	131
Child	3,17

Ching 6,8-10
 Churchman 22-23,31
 CID 150-151
 Clark 26-28
 Clemons 3
 Coleman 14,123
 Conger 25-26,28
 Cornford 127
 Council of Europe 220
 Crosslin 40
 Crowston 5-7,11-12,14-16,18,87,268,277
 Culnan 13,15,18,129

D

Daft 1,15,130
 Davenport 2,13
 Davis 2
 Davison 127
 Dekker 91
 Delbecq 129,135,141
 Dennis 13,26,125,127-132,280-282,284
 Denzin 26
 DeSanctis 2,16,22,26,125,127,131-133,135,280
 Dick 27
 Dickson 22,26,122,127,130,133,135,138-140,
 144,280
 Diehl 129
 Dijk-Veenenga Kingma 102,104
 Douma 6
 Drucker 1-3,9,123
 Duchon 22
 Duck 213
 Dur 2-3,15,24,35,38,40-42,47,60,63,79,82,
 84,87,277-278
 Duyne 148-149

E

Eason 212
 Eden 127
 Egmond 149
 Eijck 3,19,31,42,52,285
 Eijnatten 18-19
 Eisenberg 12
 Eisenhardt 27,29
 Ellis 14,123,212
 ElMaraghy 58,60
 Elsevier 220
 Emery 2,42,122
 Engelstad 122

Es 220
 Eshof 220
 Evered 22

F

Fersko-Weiss 14,123
 Fijnaut 148-149
 Flores 13,15,41
 Floyd 143
 Flynn 16
 Foster 16
 Francescato 1

G

Galbraith 4,9
 Galliers 21-23, 25-28
 Gallupe 125,127-128,130,132,280-281,284
 George 16,19,130
 Geurts 216
 Ghoshal 2-3
 Gipps 60
 Glaser 31
 Gopal 127
 Grant 60,61
 Gray 123,125,127
 Grohowski 130
 Gulick 4
 Gurbaxani 16,19
 Gustavsen 122

H

Hall 2
 Hammer 2
 Harst 14
 Hartley 27,31
 Hee 40
 Heminger 131
 Hengst 141,154,163,173,178,182-183,
 222-223,234
 Herik 127
 Herzog 212
 Hiltz 127
 Hirschheim 136
 Hofstede 131
 Hogenbrink 149
 Hogenwind 90,93
 Hollingshead 15,130,132,281
 Holt 6,11,13,17-18,41

Hoogeveen 14,153
 Huber 1,14,17-18,123
 Huckle 14
 Huston 1-2,12,45

I

Iivari 25

J

Jablin 8,12
 Jacobs 149
 Jarvenpaa 23,26,130
 Jessup 14,124
 Johansen 14,123-125
 Jones 3
 Jong 16,23,24
 Jönsson 28

K

Kaplan 22
 Keen 14
 Kickert 5
 Kieser 10
 Kiesler 14,16
 Kim 42
 King 16,130
 Kirst 3
 Kleijnen 38
 Kling 19
 Koningh 220,222,229
 Konsynski 13
 Korf 149
 Kortekaas 220
 Korthals-Altes 149
 Kraemer 25,127,130-131
 Kramer 56,58
 Krcmar 127
 Kubicek 10
 Kuijlaars 275,285,223,232,241,243,245-246,
 249-250,253
 Kumar 132,135,144
 Küster 148
 Kyng 13

L

Lancione 271
 Land 2,22,25-26,136

Lawrence 2,4,25
 Leavitt 17
 Ledyard 6
 Lee 26
 Leeuw 2,7-9,26,36
 Leeuwrik 173
 Lengel 15
 Lensing 148
 Leonard-Barton 8
 Lewin 13
 Lewis 127-128
 Leyenburg Management 90
 Ligteringen 221
 Likert 10
 Limayen 131
 Lissenberg 149
 Lister 6
 Lochovsky 42
 Lorsch 2,4
 Loveman 2
 Lucas 2,16
 Lundquist 1-2,12,45

M

Malone 1,3,5-9,11-16,19,72,87,268,277
 March 2,4,9-10,12
 March 22,45
 Markus 13,15,18,29,31,129
 Marshall 127
 Martz 280
 Massman 271
 McDaniel 1,14,17,18,123
 McFarlan 2
 McGoff 130
 McGrath 15,130,132-133,281
 McGregor 10
 McKenney 2
 Meel 23-24,30-31,2-3,13,17-19,35,41,60-61,
 80,82,87-88,107,110,141,147,149,152,
 154,163,178,198-199,205,212,219,221-222,
 234,238,255,261,278
 Meer 14
 Meinsma 31,56
 Meyer 42
 Middelburg 148,220
 Ministry of Justice 148
 Mintzberg 3-4,7,9,11,15,22
 Mitroff 22,37
 Monge 12
 Mooney 4

Moore 141,196
 Morgan 21-22,25-26
 Morrison 132
 Morton 1-2,16-17,271
 Motshagen 35
 Muller 127
 Mumford 19,122,215

N

National Criminal Intelligence Service . . . 148
 Newsweek 148,220
 Newton 127-128
 Ng 56,58
 Nunamaker 2,17,26,125-128,131,132,141

O

O'Keefe 58
 OCID 150
 Olson 2,16,26
 Oppen 14,123
 Ören 61
 Orlikowski 16-18,21-22,25-26,138
 Osborn 129
 Otte 173

P

Palys 26
 Panko 14-15,185
 Parker Follett 5
 Paternotte 221
 Pegden 50-51,58-59,62,82,105,106,110,198,242
 Pervan 21,23,127,280
 Petri 40
 Pettigrew 29
 Pinsonneault 127,131
 Platt 60
 Poole 2,16,131,135
 Poppel 15

Q

Quinn 9
 Qureshi 9,12,45,127,283

R

Raman 131
 Ramondt 90,118

Randolph 10
 Reisig 40
 Reuter 148
 Reuyl 16
 Rhébergen 90,118
 Riessen 148-150
 Roach 1-2,15,271
 Robey 2,14,16-18
 Rockart 1,3,12-13,15,140
 Roer 148
 Rogers 3
 Rohrbaugh 26
 Ross 6
 Rule 15
 Rumbaugh 42

S

Scholes 136
 Schön 27
 Schreuder 6
 Schrijver 35
 SCID 150
 Senge 216
 Shannon 38
 Shi 16,19
 Short 2,13
 Simon 2,4-5,9-10,12,14,16,22,36-37,45
 Smith 6,8-9,60
 Smithsonian 2,26
 Sol 1-3,5,13,16-17,19,23-24,26,31,
 35-38,40,50,61,63,80,88,128,132,136,143,
 187,192,197-199,215,257
 Specht 17
 Spek 56
 Sproull 14
 Stalker 2
 Stephens 13
 Sterman 216
 Straatman 3
 Strauss 31
 Streng 3,35,51,60,193,278
 Stroebe 129
 Susman 22
 Swain 55,58

T

Tactical Department 220
 Tanenbaum 6
 Tas 136

Task Force Crime Analysis 148
 Task Force 'Kit in de lift' 149
 Task Force Objectives 151
 Task Force Z&R 221
 Teerds 149
 Thompson, G. 6,8-9,72
 Thompson, J.D. 5,10,268,277
 Todd 26
 Toffler 1,9
 Trauth 2
 Tsalgatidou 58
 Turoff 127
 Tyran 127
 Tyre 17
 Tyre 138

U

Urwick 4

V

Valacich 14,124,129
 Van de Ven 10,129,135,141
 Van Horn 25
 VanGundy 37
 Vegter 220
 Ven 132
 Vennix 216
 Ventana Corporation 133
 Verbraeck 24,31,35,37,198-199
 Vician 127,131,280
 Vitalari 122
 Vogel 13,19,25-26,125,127,132
 Volpato 127
 Vonk 143
 Vreede 13-15,31,52,84,91,127-128,
 130-132,163,172,187,198-199,214,
 223,232,235,237,243,245,253,259

W

Waddington 28
 Waema 29
 Wagner 124
 Walsham 29
 Wanninger 122,127,135,138-140,144,280
 Watson 15,124,264
 Wei 131
 Weill 26
 Weiner 60-61

Welten 148,150,220,222,229
 Wetherbe 25-26,140-141
 Whang 16
 Whetten 3
 Wierda 3,24,35,51,57,60,199,278
 Williamson 6
 Winograd 41
 Womack 3
 Wouw 153
 Wynekoop 25-26,28

Y

Yin 27-29,31

Z

Zeigler 61
 Zigurs 26,127,131,280
 Zmud 27
 Zuber-Skerritt 27

Subject index

*'Well there's egg and bacon; egg, sausage and bacon; egg and spam;
egg, bacon and spam; egg, bacon, sausage and spam; spam, bacon,
sausage and spam; spam, egg, spam, spam, bacon and spam; spam,
spam, spam, egg and spam; spam, spam, spam, spam, spam, spam,
baked beans, spam, spam, spam, and spam; or lobster thermidor aux
crevettes with a mornay sauce garnished with truffle pâté, brandy and
a fried egg on top and spam.'*
- Monty Python

A

action research 26-27,272
actor 5,36,43,52,68,93,155,225,278
actor action 41,44,96,158,228
actor model 48,74,100,139,162,193,233
Amsterdam Municipal Police Force . . 148,219
animation 55
animation model . . 50,58,80,105,108,119,122,
. 198,278,285
animation view 244
anonymity 128,280
application area 32,267
appointment facility 113,116
Arena 242,261-262,285
aspect model 39,45,139,192
Atlas of Criminal Organizations and Societal
Trends 177
attribute 41
attribute assignments 45

B

basic classes 42
bounded rationality 37

C

case study design guidelines 28
case study research 26-27,272
change agents 213
chauffeured GSS 124

Cinema 59,242
closed system 107
communication 12,55
communication; explicit . . 12,98,160,232,269,
. 271,277
communication; implicit 12
conceptual model 37,39,139,256
conceptualization 190
conceptualization filter 39,139
concurrent processors 59
constructed setting 63
cookbook 215
coordination 3-4
coordination as control 5
coordination by feedback 10
coordination by plan 10
coordination costs 15,270
coordination mechanisms 7,231,268,277
coordination medium 84
coordination processes 7,10,99,162,232
coordination; adaptive . . 10,73,100,162,232,241,
. 248,277
coordination; anticipatory 10,73,99,162,232,241,
. 248,277
coordination; indicators 11
coordination; interorganizational 3
coordination; organizational 3-4,7,13,40
coordination; review . . 10,74,100,118,162,233,
. 241,248,269,277
coordination; work place 3
Criminal Investigations Department . . 30,148,
. 219,268

critical processes	142
Critical Success Factor	140
cycle time	107

D

debugging	110
decision	44,55,71,96,158,228
decision makers	56
decision making	4
decision rules	45
Department of Unemployment Benefits	30,64, 268
deployment agency structure	222,255
design approach	19,273
diagramming techniques	56
diplans	41
direct supervision	9,161,231,268,277
Documentary Information Systems	14
dynamic modelling	32,35,87,118,121,147,189

E

E-mail facilities	14
Electronic Document Handling (EDH) system	85
empirical data	50
empirical data; availability	119
empirical model	39,49,137,139,195,259
empirical model; descriptive	38
empirical model; prescriptive	38
endogenous data	38
exogenous data	38
experiment frame	51
experiment model	50,61,198
experimental design	50,61
experimental research	127
experimentation	25
external validity	26

F

face-validity	60
facilitation	130,282
field studies	127

G

game shell	285
gaming techniques	285
generalizability	26
globally networked society	1

GRIP phase	144
GRIPPPER approach	122,139,144
group	128
group experience	282
group memory	129,281
group size	131,282,284
Group Decision Support System	123
Group Support Systems	14,123,280,286
Group Support Facility	125
group; context	128
group; culture	131
group; task	128
GroupSystems for Windows	125
GroupSystems V	125,133,163,166,173,177,182, 187,194,201,204,209,211,214,216,235,237,243, 259,264,284
groupware	14,123
GSS-experience	131
GSS functionality mapping	135

H

hierarchy	8,97,268
hierarchy; functional	9,72,97,158,173,229
hierarchy; product	9,97,159,175,229-230
historical data	26
Human Relations School	10

I

ill-structured problem	36,136
inductive-hypothetic strategy	23,272
information availability	16,271
information overload	129
information quality	16,271
information system	136
information system research	22
information technology	2,12,16
information technology; enabler	13,270
information technology; requirement	13,270
Information Department	151,255
initial cause hypotheses	202
initial replications	80,107
initial theoretical framework	31
initialization conditions	38
inquiring system	22
inquiry system	19,136
intellectual framework	32
interaction model	45,72,97,139,159,193,230
interdependency	5,268,277
interdependency; pooled	5

interdependency; reciprocal 5,16,277
interdependency; sequential 5
interpretivism 22
item 42,52,66,93,155,223,278
item processor 43,68,93

J

joint modelling 284

K

keypad GSS 124

L

lab experiments 127
Law of Requisite Variety 2
Leyenburg Hospital 89
link 43,53,71,96,157,227,278
Lotus Notes 286

M

macro perspective 3
manufacturing industry 1,271
market 6,8,283
market; centralized 8
market; decentralized 8
Market I 150
Market II 150
measuring instruments 26
media richness 130,281
meeting design 282
meeting process 132
meeting task taxonomy 135,144
meeting; different time different place . . . 126
meeting; different time same place 126
meeting; same time different place 126
meeting; same time same place 126
mental model 57,257
meso perspective 3
message 42-43
methodology 32
micro perspective 3
model 56
model frame 50
model reductions 197
model sequence 39
modelling concepts 88,184,192,277
modelling strategies 193,201

modelling strategies; incremental 256
mutual adjustment 9,232-233,268,277

N

network 9,222,283
networked society 283
networking relations 159
Neurology Outpatients' Department . . . 30,91,268
node 43
Nominal Group Technique (NGT) 129
number of replications 80,107

O

object 36
object; action part 42
object; attribute part 42
object-attribute-action worldview . . . 38-39,41,
. 139,192
object-oriented view 42
organization 2
organization research 22
organizational change 35,56
organizational processes 36
organizational requirements 137,163,195,
. 235,259
organizational structure 7,16
organized crime 148,219
output data 107

P

pacman effect 261
parallel communication 129
parallelism 281
participant observation 28
participation 17,19
participative decision making 123
participative design 32,122,136,147,189
participative modelling 216
patient-friendliness 90,118
person 43
Petri net formalism 40
phenomena approach 220
planned opportunism 29
pluralism 26,287
policy document 212
portrayal animation models 58
positivism 21
post evaluation 214

post-industrial society	1
postconditions	44,54
postprocessors	58
preconditions	44,54
problem list	141,165,237,259
problem owners	56
problem solvers	56
problem solving	136,276
problem solving framework	215
problem statements	199
procedural rationality	37
process model	47,73,98,139,193
producer/consumer relationships	5
product	43
productivity of groups	15,271
PROject Management Information System	249
prototype	17,138,140,143,207
prototype; horizontal prototype	143
prototype; vertical prototype	143
prototyping filter	140
prototyping strategy	209
purposeful systems	2

Q

qualitative data sources	30
quantitative data sources	30

R

reduction	76,101,163,234
reduction activities	50
reduction transformations	39
reorganizing	2
replication	61
repository	44,52,95,157,227,278
representative power	272
research instruments	21,25
research objective	19
research philosophy	21
research question	19,267
research strategy	21,22
research teams	31
researcher bias	26
review processes	86
run control conditions	38
run length	80,107

S

SAMM	133

scientific rigor	26
service industry	1,15,271
shared resources	5
SIMAN	50,242
SIMAN blocks	51
SIMAN elements	51
SIMAN entities	51
SIMAN simulation code	102
simulation code	50
simulation experiment	38,61
simulation language	50,56
simulation model	196,207,242,244
simulation run	38
simultaneity constraint	6,277
Singerian inquiring system	23
social system	136
social-technical systems design theory	18
solution characteristics	204
solution directions	204
solution requirements	204
specification filter	39,139
speech act	41
stakeholder participation	119,122
standardization	9,72,98,100,160,162,231, 268,277
standardization of norms	9
standardization of work outputs	9,73,160
standardization of work processes	9,72,74,98, 160,162
standardization of worker skills	9,73,98,160, 162,231
start up time	79,107
substantive rationality	36
super-industrial society	1
system	2,36,41
systems perspective	7,36

T

Tactical Department	151,222,254
task	55,71,96,158,228
task structures	41
task/subtask relationships	6
TeamGraphics	125,184,194,223,259,263,284
Technical Department	152
template	285
test scenario	208
theory	24
theory building	23,273
theory development	23
theory expanding	23
theory testing	23

third wave society 1
time delay 45
top-clinical care 90
tracing 82,110
transformation procedure . 88,119,198,207,262,
. 275,285
treatment 38,61

U

units 151,222
units; base units 222
validation 199
validation; implicit validation 167,239
validation; replicative validation 38,82,111,199
validation; structural validation . 38,82,111,199
verification 82,110,198
verification; implicit verification . 167,239,257
VisionQuest 133

W

way of controlling 19,35
way of modelling 19,39
way of thinking 19,36,189
way of working 19,37
well-structured problem 36
Weltanschauung 36
workflow systems 14
workstation GSS 124
worldview 19

Samenvatting

Het 'mooie' van de ons omringende werkelijkheid zit niet in de dingen zelf, maar in de wijze, waarop ze door ons bekeken worden. Pas Jacob Maris heeft de Hollandse sloten ontdekt.

- G. Bomans

Probleemgebied

De moderne organisatie ziet zichzelf veelvuldig geplaagd voor een noodzaak tot verandering. Ontwikkelingen binnen en buiten de organisatie stellen andere eisen waaraan zij moet voldoen en scherpen de randvoorwaarden aan waaronder zij moet opereren. Als antwoord op dergelijke ontwikkelingen passen organisaties (gedwongen) hun structuren, processen en technologieën aan. Centraal binnen deze veranderingen staat *coördinatie*, de afstemming van de samenstellende activiteiten binnen organisatorische processen. Voor een organisatie is de succesvolle coördinatie van haar processen een voorwaarde om in haar omgeving te kunnen functioneren. Dit betekent dat de druk om bestaande processen en werkwijzen binnen een organisatie aan te passen, de noodzaak impliceert om een andere invulling te geven aan de coördinatie van deze processen en werkwijzen.

Coördinatie kan vanuit drie perspectieven worden bekeken: interorganisationele coördinatie (macro-perspectief), organisationele coördinatie (meso-perspectief) en werkplek coördinatie (micro-perspectief). In dit proefschrift is met name gekeken naar *organisationele coördinatie*, hetgeen is gedefinieerd als *het wederzijds beïnvloeden van de onderlinge afhankelijke activiteiten van twee of meer actoren om een bepaald doel te bereiken*. Om afhankelijkheden tussen activiteiten te coördineren passen organisaties verschillende coördinatiemechanismen toe. Deze mechanismen kunnen worden onderverdeeld in drie categorieën op basis van hun invalshoeken:

1. *Organisatiestructuur*. De structuur van een organisatie als geheel heeft tot doel het gewenste gedrag van de organisatie te dicteren en te faciliteren. De meest voorkomende basisstructuren die hiervoor worden aangewend, zijn: de markt

(gecentraliseerd en gedecentraliseerd), de hiërarchie (functioneel of produktgericht) en het netwerk.

2. *Afhankelijkheidsstructuur*. De afhankelijkheden tussen activiteiten kunnen worden gecoördineerd door standaardisatie (van werkprocessen, van resultaten, van competentie en van normen), door wederzijdse beïnvloeding en door directe aansturing.
3. *Coördinatieprocessen*. Vanuit een procesmatige invalshoek zijn er drie verschillende coördinatieprocessen zichtbaar: planningsprocessen, adaptieve processen en evaluatieprocessen.

Om hun afhankelijke activiteiten te kunnen coördineren moeten actoren communiceren. Zij moeten impliciet (bijvoorbeeld bij standaardisatie) of expliciet (bijvoorbeeld bij directe aansturing) informatie uitwisselen om elkaar van de status van hun activiteiten op de hoogte te brengen of te houden. Vanwege dit informatie-intensieve karakter van coördinatie-activiteiten, komt het steeds vaker voor dat organisaties informatiesystemen inzetten om organisatorische coördinatie te ondersteunen. Beschrijvend onderzoek doet vermoeden, dat door de inzet van informatiesystemen de kosten van organisatorische coördinatie zullen afnemen en de hoeveelheid coördinatie zal toenemen. Ook verwacht men een toename van de produktiviteit van actoren die informatiesystemen gebruiken voor de coördinatie van hun activiteiten. Tenslotte wordt vaak verwacht dat deze systemen een positieve invloed heeft op de beschikbaarheid en kwaliteit van informatie.

We zien dus dat organisaties aan de ene kant gedwongen worden te veranderen en daardoor de coördinatie van hun processen en samenstellende activiteiten moeten heroverwegen. Aan de andere kant verschaffen verschillende soorten informatiesystemen de mogelijkheid om (nieuwe vormen van) organisatorische coördinatie te ondersteunen. Het probleem is nu om met deze simultane ontwikkelingen en mogelijkheden adequaat om te gaan. Echter, dit is eenvoudiger gezegd dan gedaan. Literatuur over organisatie-ontwerp geeft weinig richtlijnen en voorbeelden voor het *ontwerpen* en *verbeteren* van organisatorische coördinatie. De ervaring leert verder dat het herinrichten van processen en hun coördinatie, alsmede het ontwerpen en inzetten van informatiesystemen niet los van elkaar kan worden gezien. Literatuur over organisatie-ontwerp besteedt echter geen aandacht aan informatiesystemen, terwijl de meeste literatuur over informatiesysteemontwerp de organisationele context uit het oog verliest.

Op basis van het bovenstaande is voor het promotie-onderzoek de volgende *vraagstelling* geformuleerd:

Hoe kunnen we organisaties ondersteunen bij het verbeteren van hun coördinatie en ondersteunende informatiesystemen?

Om hierop antwoord te geven is het volgende *onderzoeksdoel* nagestreefd:

Formuleer een ontwerpaanpak om het verbeteren van organisatorische coördinatie en ondersteunende informatiesystemen te faciliteren.

Onderzoeksaanpak en -verloop

Om aan het onderzoeksdoel te voldoen is het voorliggende promotie-onderzoek uitgevoerd. In dit onderzoek is op inductief-hypothetische wijze de beoogde ontwerpaanpak geformuleerd en uitgetest. Hiertoe zijn vier praktijkstudies uitgevoerd, waarbij we ons bedienen van de onderzoeksinstrumenten case- en actie-onderzoek.

Vertrekpunt voor onze ontwerpaanpak vormde 'dynamisch modelleren', een probleem-gerichte aanpak voor het analyseren van bestaande organisatorische situaties en het ontwerpen van organisatorische veranderingsalternatieven. Binnen deze aanpak wordt gebruik gemaakt van conceptuele en empirische modellen voor het beschrijven van huidige en toekomstige situaties. Simulatie- en animatietechnieken worden gebruikt om situaties kwantitatief te analyseren en de dynamiek erin te visualiseren.

In onze eerste case-studie hebben wij op exploratieve wijze onderzocht in hoeverre de aanpak 'dynamisch modelleren' geschikt was voor het verbeteren van organisatorische coördinatie. Op basis van de ondervindingen tijdens de case-studie zijn modelleer-concepten ontwikkeld voor het maken van conceptuele modellen. Eveneens is een procedure voorgesteld voor de transformatie van een conceptueel model in een empirisch model. Deze beide elementen van onze beoogde ontwerpaanpak zijn zowel in de eerste als in de tweede case-studie toegepast, geëvalueerd en verfijnd. In de tweede case-studie is tevens onderzocht in hoeverre animatiemodellen kunnen worden ingezet om de communicatie te ondersteunen tussen de analist en de belanghebbenden vanuit de organisatie en om deze belanghebbenden bij het veranderingsproces te betrekken.

De tweede casus leerde dat om optimaal van de kennis van belanghebbenden te kunnen profiteren en om hen actief bij het veranderingsproces te betrekken, animatiemodellen *alleen* niet voldoende zijn. Daarom is in de derde case-studie onderzocht of Group Support Systemen (GSS) kunnen worden gebruikt om actieve betrokkenheid te stimuleren en te faciliteren. Tevens is in deze case-studie gebruik gemaakt van additionele concepten om empirische modellen te ontwikkelen die niet uitsluitend gestoeld zijn op statistische, kwantitatieve gegevens. Voor het ontwikkelen van een informatiesysteem om het gekozen organisatorische veranderingsalternatief te ondersteunen is tenslotte gebruik gemaakt van GSS in combinatie met prototyping-technieken.

De derde case-studie leerde ons allereerst dat GSS zeer succesvol bleken voor het ondersteunen van verscheidene ontwerpactiviteiten. Verder werd, mede door deze ervaringen, besloten om ook op basis van de concepten de conceptuele modellering op participatieve wijze uit te gaan voeren. Tenslotte werd geconstateerd dat het gebruik van uitsluitend deze modelleerconcepten niet voldoende is om veranderingsalternatieven *inzichtelijk* over te brengen aan organisatorische belanghebbenden.

Op basis van de ervaringen uit de eerste drie case-studies werden de zojuist beschreven concepten, technieken en methoden geïntegreerd tot de voor ons onderzoek beoogde ontwerpaanpak. Een vierde en laatste case-studie is uitgevoerd om deze ontwerpaanpak toe te passen en te evalueren.

Ontwerpaanpak

Onze ontwerpaanpak kan beschreven worden in termen van een denkwijze, een werkwijze, een modelleerwijze en een beheerswijze. In vernieuwende zin heeft het promotie-onderzoek zich voornamelijk op de werkwijze en modelleerwijze van de ontwerpaanpak gericht. De werkwijze van de aanpak bestaat uit zes hoofdactiviteiten. Deze worden hieronder apart behandeld, waarbij steeds wordt aangegeven welke modelleringstechnieken worden gebruikt.

Bij de *conceptualisatie van een probleemsituatie* wordt de structuur en de afbakening van het probleem gemodelleerd. Uitgaande van een object-attriboot-actie Weltanschauung worden de relevante objectklassen geïdentificeerd en gedefinieerd in termen van de concepten 'actor', 'repository', 'link', 'message', 'product' of 'person'. Om het zo ontstane conceptuele model vanuit verschillende perspectieven nader te kunnen onderzoeken, worden drie aspectmodellen geconstrueerd: het *interactiemodel*, dat de uitwisseling (communicatie) van messages, products en persons tussen actoren en/of repositories weergeeft, het *procesmodel*, dat de organisatorische processen beschrijft alsmede de wijze waarop deze messages, products en persons verwerken, en het *actormodel*, dat het werkproces van individuele actoren naar voren brengt.

Bij de *bouw van een empirisch model* wordt het conceptuele model verder gespecificeerd zodanig dat het resulterende empirische model kan worden gebruikt voor het analyseren en diagnostiseren van de probleemsituatie. Hiertoe wordt allereerst een aantal 'organisatorische eisen' geïdentificeerd, voorwaarden waaraan moet zijn voldaan wil de organisatie in staat zijn haar functie in haar omgeving te vervullen. Vervolgens kan het conceptuele model met behulp van een transformatieprocedure worden omgezet in een simulatiemodel in de taal SIMAN of Arena. De dynamiek in het simulatiemodel wordt gevisualiseerd in een animatiemodel. Nadere empirische gegevens worden aan het model toegevoegd waarna het wordt geverifieerd en gevalideerd. De bouw van het

empirisch model wordt voltooid door, mede op basis van het simulatiemodel, een aantal probleembeschrijvingen (feitelijk 'aannames') op te stellen en deze te relateren aan de processen uit het conceptuele model.

Het doel van het *diagnostiseren van de probleem situatie* is het vaststellen van de oorzaken die aan de geconstateerde problemen ten grondslag liggen. Hiertoe wordt eerst een aantal 'oorzaak-hypothesen' vastgesteld. Deze worden vervolgens met het simulatiemodel onderzocht en/of in een aantal groepssessies nader geanalyseerd. Op basis van de onderzoeksresultaten met betrekking tot de oorzaak-hypothesen worden tenslotte mogelijke oplossingsrichtingen gedefinieerd.

Naar aanleiding van de oplossingsrichtingen kan een aantal mogelijke oplossingen worden geselecteerd en verder worden *uitgewerkt in een aantal veranderingsalternatieven*. Een veranderingsalternatief wordt eerst geconceptualiseerd met de bovenbeschreven modelleerconcepten. Vervolgens wordt deze conceptualisatie omgezet in een simulatiemodel inclusief animatie. Op basis van simulatie-experimenten en -demonstraties wordt het model geëvalueerd samen met een aantal belanghebbenden en/of experts. Indien dit wenselijk wordt geacht, wordt op basis van de specificatie van het veranderingsalternatief een prototype van een ondersteunend informatiesysteem gemaakt, dat samen met de belanghebbenden wordt getest en geëvalueerd.

Na evaluatie van de resultaten uit de voorgaande activiteit wordt een *veranderingsalternatief gekozen* in nauw overleg met de belanghebbenden. Bij dit keuzeproces kan naast het feit of het veranderingsalternatief het oorspronkelijke probleem wegneemt, ook een aantal andere selectiecriteria worden toegepast. Deze hebben bijvoorbeeld betrekking op de politieke, sociale, economische of psychologische aspecten van een alternatief.

Tenslotte wordt het gekozen *veranderingsalternatief geïmplementeerd* in de probleemsituatie. Ter voorbereiding hiervan wordt door de organisatie zelf een beleidsdocument geschreven, waarin (a) een heldere samenvatting van de oorspronkelijke problemen, (b) een duidelijke beschrijving van de toekomstige processen en activiteiten en (c) de te volgen implementatiestrategie aan bod komen. Na het voorlichten van de betrokkenen met behulp van de modellen van het veranderingsalternatief, worden de organisatorische processen en hun coördinatie aangepast zoals het gekozen alternatief voorschrijft. Indien van toepassing worden ook de hierbij ontwikkelde informatiesystemen gerealiseerd. Tenslotte kan een post-evaluatie worden uitgevoerd om na te gaan of de oorspronkelijke probleemsituatie inderdaad is verholpen.

Bij de bovenstaande activiteiten kan op verscheidene momenten GSS-ondersteuning worden ingezet om de actieve betrokkenheid van belanghebbenden uit de organisatie te stimuleren en te faciliteren. In het promotie-onderzoek is gebruik gemaakt van

GroupSystems V, een elektronisch vergadersysteem, en TeamGraphics, een modelleer-tool voor groepen. De toepassing van deze GSS voor de belangrijkste probleemoplossende activiteiten is samengevat in tabel 1. Hierbij is voor GroupSystems V steeds de betreffende module aangegeven.

Activiteit	GSS-ondersteuning
<i>Conceptualiseer probleem:</i>	
• Verzamel conceptuele gegevens	TC
• Definieer probleemgrenzen	TC, IO
• Identificeer en definieer relevante objectklassen	IO
• Bouw aspectmodellen	TeamGraphics
<i>Bouw empirisch model:</i>	
• Formuleer organisatorische vereisten	TC, IO, Vote
• Verzamel empirische gegevens	TC, IO, QE, SV
• Voer structurele validatie uit	IO, QE, SV
• Formuleer probleembeschrijvingen	TC, IO, Vote
• Relateer problemen aan processen	GO, GM
<i>Diagnostiseer probleemsituatie:</i>	
• Formuleer oorzaakhypothesen	TC, IO
• Onderzoek oorzaakhypothesen	TC
• Formuleer oplossingsrichtingen	TC, IO
<i>Werk veranderingsalternatieven uit:</i>	
• Identificeer mogelijke veranderingsalternatieven	EBS, TC, IO
• Selecteer uit te werken veranderingsalternatief	Vote, AE
• Conceptualiseer veranderingsalternatief	TC, IO, TeamGraphics
• Evalueer veranderingsalternatief	TC, IO, Vote, AE
• Bouw prototype IT-ondersteuning:	
- Stel vereiste functionaliteit vast	TC, IO
- Verzamel testdata	TC, IO
- Evalueer prototype	TC, IO, Vote
<i>Kies veranderingsalternatief:</i>	
• Vergelijk evaluatieresultaten	GM
• Identificeer extra criteria	IO, Vote
• Stel te implementeren alternatief vast	Vote, AE
<i>Implementeer veranderingsalternatief:</i>	
• Schrijf beleidsdocument	GW
• Licht betrokkenen voor	TC
• Voor post-evaluatie uit	TC, IO, QE, SV, AE, GM, Vote

TC = Topic Commenter, IO = Idea Organization, QE = Questionnaire, SV = Survey, GO = Group Outliner, GM = Group Matrix, EBS = Electronic Brainstorming, AE = Alternative Evaluation, GW = Group Writer

tabel 1 GSS-ondersteuning voor participatieve probleemoplossingsactiviteiten.

Case-studies

De voor het opstellen en evalueren van onze ontwerpaanpak benodigde ervaringen kwamen voort uit vier verschillende case-studies. Hieronder wordt elke case-studie kort

toegelicht met betrekking tot de probleemsituatie en de aangedragen organisatorische en informatietechnische oplossingen.

Case 1: Afdeling Pensioen- en Wachtgelden (P&W)

De eerste case-studie betrof een geconstrueerde casus, gesitueerd bij het Ministerie van Binnenlandse Zaken, afdeling Pensioen- en Wachtgelden (P&W). Deze afdeling is verantwoordelijk voor het uitvoeren van maandelijkse compensatiebetalingen aan ambtenaren die onvrijwillig hun baan zijn verloren. Het probleem waarvoor de afdeling zich zag geplaatst kende verschillende aspecten: Het serviceniveau was te laag; men had het sterke vermoeden dat te veel opgaven van inkomsten en mutaties te laat werden verwerkt, wat resulteerde in onjuiste betalingen. Het aantal cliënten van de afdeling zou in de toekomst toenemen terwijl de afdeling zelf zou moeten afslanken.

Om de situatie bij P&W te analyseren, zijn de processen en coördinatiemechanismen geconceptualiseerd en vervolgens vertaald in een geanimeerd simulatiemodel. De afdeling bleek een functionele hiërarchie te kennen, waarbinnen met name standaardisatiemechanismen werden gebruikt om afhankelijkheden tussen activiteiten te coördineren. Het belangrijkste coördinatiemiddel betrof de documentenstromen door de afdeling. De coördinatieprocessen waren adaptief.

Uit experimenten met het simulatiemodel bleek dat het huidige serviceniveau 86 procent was. Om hier verbetering in aan te brengen zijn twee alternatieven geëvalueerd, die er specifiek op waren gericht om de coördinatie van de activiteiten binnen de afdeling te *versnellen*. Eerst is gekeken naar het verhogen van het aantal boderrondes. Vervolgens is het effect van een documentair informatiesysteem onderzocht dat elektronische documentuitwisseling ondersteunt. De resultaten van de experimenten met deze alternatieven leerden dat het eerste alternatief slechts geringe verbeteringen van het serviceniveau tot gevolg zou hebben (87 procent). Dit in tegenstelling tot het tweede alternatief, dat het serviceniveau tot ruim 98 procent zou doen stijgen.

Case 2: De Polikliniek Neurologie

De tweede case-studie vond plaats bij Ziekenhuis Leyenburg in Den Haag, in het bijzonder de Polikliniek Neurologie. Leyenburg zag zich geconfronteerd met een toenemende vraag om zorg, terwijl de middelen om deze zorg te verschaffen (personeel, apparatuur, tijd) afnamen. Bovendien namen de kosten van zorg voortdurend toe. Het adequaat inspelen op deze ontwikkelingen werd bemoeilijkt door spanningen tussen de medische staf en de directie van het ziekenhuis over de te volgen reorganisatiekoers. Verder veroorzaakte de toenemende vraag naar poliklinische zorg een stijging van de bezoektijd van patiënten en het drukker worden van de polikliniek. Tenslotte bleek het

niet beschikbaar zijn van managementinformatie over de huidige situatie een hindernis te zijn om tot besluitvorming over mogelijke organisatie-aanpassingen te komen.

Binnen de case-studie is gekeken naar de situatie bij de Polikliniek Neurologie van het ziekenhuis. De patiëntenstromen in de uitgangssituatie werden hierbij in kaart gebracht met behulp van een conceptueel model dat werd vertaald in een simulatiemodel. De dynamiek in dit model werd gevisualiseerd in een animatiemodel. De polikliniek behandelde patiëntenstromen volgens een gemengde functionele en produktgerichte hiërarchie. Afhankelijkheden tussen activiteiten werden voornamelijk door standaardisatie en onderlinge afstemming gecoördineerd. Er werd in hoofdzaak gebruik gemaakt van planningsprocessen.

Uit de diagnose van de probleemsituatie op de polikliniek bleek dat patiënten in feite werden gebruikt om de activiteiten van de betrokken actoren te coördineren: een patiënt was hier gemiddeld meer dan de helft van zijn of haar bezoek mee bezig. Samen met een aantal betrokkenen uit de Polikliniek zijn enkele verbeteringsvoorstellen geformuleerd voor deze situatie. Experimentatie leerde dat van de volgende combinatie van de verbeteringsvoorstellen de meest positieve effecten verwacht konden worden: een decentralisatie van de bloedafname, een verandering in het artsenrooster, een delegatie van afspraakactiviteiten naar artsen en de invoering van een ziekenhuisbreed afsprakensysteem. Deze veranderingen tezamen zouden de gemiddelde bezoektijd van een patiënt met circa 25 procent terugbrengen.

Case 3: De Dienst Centrale Recherche (DCR)

De derde case-studie vond plaats bij de Dienst Centrale Recherche (DCR) van de Regiopolitie Amsterdam/Amstelland. De DCR is verantwoordelijk voor de bestrijding van de georganiseerde criminaliteit in haar regio. Om het hoofd te bieden aan de toenemende complexiteit en dreiging van de georganiseerde criminaliteit, verving de DCR haar zaakgerichte strategie door een marktgerichte strategie. Analooq hiearaan was de organisatie overgegaan van een functionele structuur naar een meer produktgerichte projectstructuur. Een belangrijke activiteit in de nieuwe organisatie vormde het *voorbereiden* van projecten. Speciaal hiervoor werd het Bureau Recherche Informatie (BRI) opgericht. Echter, hiervoor diende een aantal problemen en vragen te worden aangepakt. Zo was de taakinhoud van de actoren in de toekomstige situatie nog onbekend. De wijze waarop het BRI het projectvoorbereidingsproces precies moest inrichten liet vragen open. Hetzelfde gold voor de wijze waarop selecties uit potentiële projecten moesten worden gemaakt. Tenslotte was onduidelijk hoe de verwachte informatiestromen binnen het nieuwe BRI konden worden ondersteund met informatiesystemen.

Tijdens de case-studie is de bestaande wijze van projectvoorbereiding geconceptualiseerd en vervolgens gespecificeerd met behulp van een aantal organisatorische eisen en probleembeschrijvingen. In de aanvangssituatie werkte het BRI met name volgens een gegevensgerichte produktiërarchie. De belangrijkste coördinatiemechanismen waren directe aansturing en, met name, standaardisatie van competentie. Coördinatieprocessen waren in hoofdzaak adaptief. Als belangrijkste problemen rond de uitvoering en coördinatie van voorbereidingsactiviteiten werd geconstateerd dat actoren soms te autonoom waren, dat de voorbereidingsprocessen moeilijk aanstuurbaar leken, dat het lastig was om voorbereidingsinspanningen te concentreren en dat projectinformatie te weinig 'rijkheid' kende. Deze problemen maakten dat het BRI niet aan haar belangrijkste organisatorische eis kon voldoen: een gedegen inzicht in het werkterrein van de DCR.

In nauwe samenwerking met de betrokkenen is een aantal veranderingsalternatieven ontworpen. Dit betrof ten eerste een selectieproces voor het 'rationaliseren' van projectkeuzes. Ten tweede is voorgesteld om het BRI als *projectgerichte* produktiërarchie in te richten. Hierdoor zou de coördinatie van de projectvoorbereiding directer en inzichtelijker kunnen plaatsvinden. Om deze voorbereidingsprocessen en de coördinatie ervan te ondersteunen is tenslotte een informatiesysteem ontworpen, ontwikkeld en geïmplementeerd.

Case 4: De Dienst Centrale Recherche (DCR)

De laatste case-studie vond wederom bij de DCR plaats, maar nu binnen de afdelingen die betrokken waren bij het *uitvoeren* van projecten tegen de georganiseerde criminaliteit. De probleemsituatie waarin deze afdelingen zich bevonden, had betrekking op de onduidelijkheid over de optimale structuur van projectteams, de afstemming tussen uitvoerende projectteams en ondersteunende bureaus, de feitelijke aansturing van de uitvoering van projecten en het gemis aan managementinformatie die het verloop van de projectuitvoering inzichtelijk zou maken.

Tijdens de case-studie is samen met de betrokkenen de huidige situatie rond de uitvoering van projecten geconceptualiseerd en gespecificeerd met behulp van een aantal organisatorische eisen en probleemformuleringen. Binnen de bij de projectuitvoering betrokken afdelingen bleek een gemengde functionele en produktiërarchie te bestaan. Binnen deze structuur was directe aansturing het meest voorkomende coördinatiemechanisme, gevolgd door wederzijdse afstemming. De coördinatieprocessen waren uitsluitend adaptief. De belangrijkste problemen in de aanvangssituatie betroffen het ontbreken van goede planningsprocessen, de gebrekkige afstemming tussen voorbereidings- en uitvoeringsprocessen, de onoverzichtelijke invloed van calamiteiten op de projectvoortgang en, zoals verwacht, het ontbreken van managementinformatie over de uitvoering van projecten en met name over de inzet van de ondersteunende

afdelingen hierbij. Deze problemen hadden tot gevolg dat de twee belangrijkste organisatorische eisen niet werden gehaald: inzicht in de uitvoering van projecten en inzicht in de vereiste ondersteuning.

Samen met de betrokkenen is een aantal veranderingsalternatieven geïdentificeerd en uitgewerkt. Deze hadden met name betrekking op het inrichten van plannings- en evaluatieprocessen. Ten eerste is een agenderingsproces ontworpen voor het ondersteunen van het jaarplanningsproces. Ten tweede zijn coördinatiestructuren en -processen ontwikkeld voor het continu plannen, uitvoeren en evalueren van de projectuitvoering. Tenslotte is een prototype van een informatiesysteem ter ondersteuning van de nieuwe plannings- en evaluatieprocessen ontworpen en ontwikkeld.

Conclusies en vervolgonderzoek

Ten aanzien van de uitgevoerde case-studies merken wij op dat deze een goed beeld geven van het probleemgebied van het onderzoek. Alle vormen van afhankelijkheden tussen activiteiten alsmede alle coördinatiemechanismen zijn aan bod geweest, uitgezonderd markt- en netwerkstructuren. Hiërarchische structuren in combinatie met standaardisatiemechanismen en adaptieve processen lijken favoriet. De resultaten uit de case-studies bevestigen de verwachting dat de toepassing van informatiesystemen coördinatiekosten kan verlagen en de beschikbaarheid van informatie, die benodigd is voor coördinatie-activiteiten, kan verhogen. Ook zijn er aanwijzingen dat de toepassing van informatiesystemen de produktiviteit van de gebruikers verhoogt. De ervaringen geven geen eensluidend antwoord op de vraag of informatiesystemen de hoeveelheid coördinatie verhogen of de kwaliteit van de beschikbare informatie doen verbeteren.

Ten aanzien van de gehanteerde onderzoeksapproach merken wij op dat deze ons in staat heeft gesteld een rijk beeld van de complexe werkelijkheid te krijgen. Doordat een continue leerproces ontstond, konden de relevante elementen van de beoogde ontwerp-approach worden geïdentificeerd en uitgewerkt.

Ten aanzien van de door ons ontwikkelde ontwerp-approach merken wij op dat deze aan de voor het onderzoek geformuleerde doelstelling voldoet. De ervaringen uit de case-studies tonen aan dat zowel de individuele elementen van de ontwerp-approach als de approach als geheel succesvol kunnen worden toegepast bij het verbeteren van organisatorische coördinatie en het ontwikkelen van ondersteunende informatiesystemen. In het bijzonder stellen wij dat onze ontwerp-approach:

1. De analist in staat stelt organisatorische coördinatiemechanismen kwalitatief en kwantitatief te analyseren, te diagnostiseren en te ontwerpen.

2. De (groeps)processen structureert voor het verzamelen van gegevens die voor de bouw van ken- en maakmodellen noodzakelijk zijn.
3. De actieve betrokkenheid van belanghebbenden uit de organisatie bij de modelleren en evaluatie-activiteiten stimuleert en faciliteert.
4. Ervoor zorg draagt dat de specificaties van veranderingsalternatieven een basis vormen voor het succesvol ontwikkelen van prototypes van ondersteunende informatiesystemen voor organisatorische coördinatie.

Mogelijkheden voor vervolgonderzoek om de kennis over het ontwerpen van organisatorische coördinatie en informatiesystemen te verdiepen en de ontwerpaanpak uit te bouwen, liggen op de volgende terreinen:

1. *Netwerk- en marktstructuren.* Organisaties die zijn ingericht volgens markt- en netwerkstructuren zullen in de toekomst een prominente plaats innemen. Het is nog onduidelijk hoe binnen dergelijke organisaties samenwerkings- en coördinatieverbanden ontworpen, gefaciliteerd en ondersteund kunnen worden.
2. *Participatief modelleren.* De eerste ervaringen met het participatief bouwen van conceptuele (grafische) modellen met belanghebbenden zijn positief. Meer onderzoek is echter vereist om vast te stellen (a) welke modelleringstechnieken hiervoor het meest geschikt zijn, (b) wat de optimale groepsgrootte tijdens modelleersessies is, en (c) welke modelleerstrategieën het meest vruchtbaar zijn.
3. *Transformatie-templates.* De transformatie van conceptuele modellen in empirische (simulatie) modellen lijkt verder te kunnen worden ondersteund door het gebruik van 'templates', bibliotheken met modelleerbouwstenen. Onderzocht dient te worden in hoeverre de ontwikkelde modelleerconcepten in een dergelijke template gevat kunnen worden.
4. *Spelsimulatie.* Het gebruik van animatie- en (discrete-event) simulatiemodellen van veranderingsalternatieven geeft belanghebbenden wel inzicht in toekomstige situaties, maar laat ze deze niet daadwerkelijk ervaren. Spelsimulatietechnieken bieden hiertoe wellicht betere mogelijkheden, met name wanneer deze binnen een zogenaamde 'gaming shell' worden geïntegreerd met GSS-ondersteuning.

Deze mogelijkheden vormen een agenda voor toekomstig vervolgonderzoek. Deze onderzoeksagenda vergt een *pluralistische aanpak*: een combinatie van inductieve, exploratieve strategieën om relevante vraagstukken te identificeren en deductieve strategieën om deze nader te onderzoeken.

Curriculum Vitae

*'Wees u-zelf!' zei ik tot iemand;
Maar hij kon niet: hij was niemand.
- P.A. de Génestet*

Gert-Jan de Vreede was born in Delft on December 12th, 1967. After graduating from the 'Rijswijkse Openbare Scholengemeenschap' in Rijswijk in 1986, he studied 'Computer Science' at Delft University of Technology and specialized in 'Information Systems'. During his Master's project, he evaluated a number of information systems development projects in maintenance environments. After graduating in September 1991 (cum laude), he became a research assistant at the Department of Information Systems where he began the research described in this dissertation. In January 1992, he moved to the newly founded School of Systems Engineering, Policy Analysis and Management where he completed the dissertation research. The results of his research have been published in a number of international refereed journals and have been presented at various international conferences in Europe, Asia, and the United States. In December 1993, he was selected to present his work at the Doctoral Consortium of the International Conference on Information Systems (ICIS) in Tallahassee, Florida. During his time as a research assistant, he organized two international conferences, he was a member of the School's Council, and he was involved in the construction of the School's GSS facilities. He gave a number of international guest lectures on GSS and dynamic modelling and assisted two master students with the completion of their theses. Currently, he is an assistant professor at the School of Systems Engineering, Policy Analysis and Management where he is responsible for the GSS research programme and facilities.