MASTER

Development of a Decision Support System

Heijnen, M.J.M.

Award date:
1986

DEVELOPMENT OF A DECISION SUPPORT SYSTEM

M.J.M. Heijnen,
24 December 1985.

Dept. of Industrial Engineering and Management Science, Technical University of Eindhoven.

Master's thesis

Student: Math J.M. Heijnen, 155767.

Title: Development of a Decision Support System

Company:

University: Dept. of Industrial Engineering and Management Science,
Technical University of Eindhoven.
Mentors: Dr.Ir. P. Griep,
Prof.Dr. J. Wessels,
Drs. F. Klarenaar.

Period: March until December 1985.

Eindhoven, 4 febr. 1986

L.S.

Conform de wens van het bedrijf, waar ik stage gelopen heb, zijn gedeelten van dit rapport vertrouwelijk gebleven. Deze gedeelten zijn overgeplakt.
Het is daarom goed mogelijk dat de loop van de tekst daardoor wordt verstoord en het betoog moeilijk te volgen, hiervoor mijn excuses, het was echter onvermijdelijk.

Hoogachtend,

M.J.M. Heijnen

P R E F A C E

To complete my study at the Department of Industrial Engineering of the Technical University in Eindhoven, I had to absolve a final test of capability before getting the degree. This last test is a 9 months apprenticeship at a company, where one can prove his skills. In my search for an interesting project, I came in contact with who offered an interesting project.
In march 1985 I started as a student-trainee at the Computer Service Department

I would like to express my thanks to my mentors at Doug and Herman for their daily support and my mentors at the University: Dr.Ir. Paul Griep, Prof.Dr. J. Wessels and Drs. F. Klarenaar for their scientific advices.
Also I would like to thank Martin McKweon for his advices on the correct usage of English in this report.

The master's thesis, comprising two parts, is the conclusion of this 9 months study
Part one, the report you're reading now, explains the problem to be tackled, how it was done, the kind of methodologies and methods used and why they were used. Further it explains what was accomplished using these methods. As a conclusion each method was evaluated as to whether it was the most suitable one.
The second part of the thesis consists of a number of reports, these reports are Project Documentation and 'theoretical' reports.

T A B L E   O F   C O N T E N T S

I N T R O D U C T I O N

This thesis is a description of the master's project of Math Heijnen conducted It describes how the project was carried out.
The project, and this thesis, was divided into several parts:

- Theoretical orientation phase, (Chap. 2).
- First analysis of the information needs, (Chap. 3, 4, 5).
- Construction of the first prototype, (Chap. 6).
- Renewed analysis of the information needs, (Chap. 7, 8, 9).
- Construction of the second prototype, (Chap. 10, 11, 12).

Each of the phases, mentioned above, is analysed and described in this report.
The emphasis of this thesis lies on the description of the way the project was tackled, not on the end-product, the Decision Support System.
Each part starts with an overview of the methods which were use and explains why that particular method was chosen.

Each part, mentioned above, consists of a number of chapters. Chapters start with a brief overview of the purpose and content of the chapter and ends with a conclusion. Footnotes, which contain references to part two of the master thesis, are marked with a '*'. Appendices were used to give a detailed explanation of a certain problem or topic, without disturbing the flow of the text.

The contents of the chapters are discussed in the management summary. An overview of the two parts, of the master's thesis, and how they link together, can be found on the last page of the management summary.

MANAGEMENT SUMMARY

This report is the master's thesis of Math Heijnen, student at the Dept. of Industrial Engineering and Management Science, Technical University of Eindhoven.

The master's project consists of the investigation, analysis and implementation of a Decision Support System (DSS) aimed at the managers The emphasis, of this Decision Support System, lies on the production characteristics. This project, called the Plant Profiles Project,

The development of a system, which supports these information requirements of the decision makers, requires a different development approach in comparison to transaction processing systems.
The development of the DSS is not straightforward, it is an iterative process aimed at the changing information needs of the managers.

Prototyping is used as a methodology in developing the DSS. The information needs are translated in a relatively short period (days, weeks) into prototypes of the system. These prototypes can be used as a feedback to the user, to get him talking about his requirements. In a series of iterations the final system will be constructed.
The prototyping approach needs, however, some control. Control is exercised by using the MAP method, which is a System Development Methodology (SDM) kind of project control method.

The Critical Success Factors method of John F. Rockart has been used to assess the information needs of the managers. This method tries to identify the so-called Critical Success Factors which are important indicators which determine success for a company.

After the information needs had been gathered, the first prototype was constructed on a PC using the tool PC/FOCUS. This prototype was used as a feedback to the users, whether the system suited their requirements. The prototype was also shown to the European managers Switzerland who were also interested in the project.

Decided was to go ahead with the project and upgrade it to European level.

In order to construct the second prototype a renewed analysis of the information needs was conducted. The Information Modelling method of Hoskyns was used for this information analysis.

In analysing the information needs of the managers several entity types came up. Entity types are objects the organisation wants to have information about. Entity types, distinguished in this project, are: Companies; Industrial Areas; Production Units; Products; Raw Materials; Process Units and Processes. Relations between these entity types were analysed and normalised. Entity types with their relations can be considered as the infological model.

One of the subsystems of a Decision Support System is the Model Bank. Out of the information needs of the users one important model came up: the Raw Material Model. This model makes it possible to calculate the raw material consumption of an integrated site back to a cracker.

The Infological Model was normalised into the Data Model using the Normalisation method of Codd. The Data Model is checked against the information needs of the managers in a formal test procedure, the Access Path Analysis.

The second prototype was developed with mainframe-FOCUS, using a communication link to the IBM-mainframe in , Switserland.
The first part of this prototype, which has been implemented, is the physical storage structure. Because of performance and flexibility arguments, a combination of separate and hierarchical files was constructed.

All the components of a Decision Support System were analysed, i.e. a Data Bank, a Model Bank and a User Interface. The main objective of the system lies however on the inquiry aspects; the Model Bank, at least at the moment, is less important. The system was constructed with FOCUS, using its database facilities and its programming language.
The User Interface consists of several parts. One part consists of a series of update programs, which enable the user to load data into the database. Another part are predefined reports the user is able to access. The Model Bank is not yet implemented. The emphasis, at this moment, is to get more data into the system, and move it out of the test environment.

This will prove to be a major effort because people all over the company will be involved. It is important to start small. The objective of the first implementation phase is the collection and insertion of the data and data concerning the LDPE business.
After this data has been loaded into the system, the system and its results must be communicated and 'marketed' to the potential users.

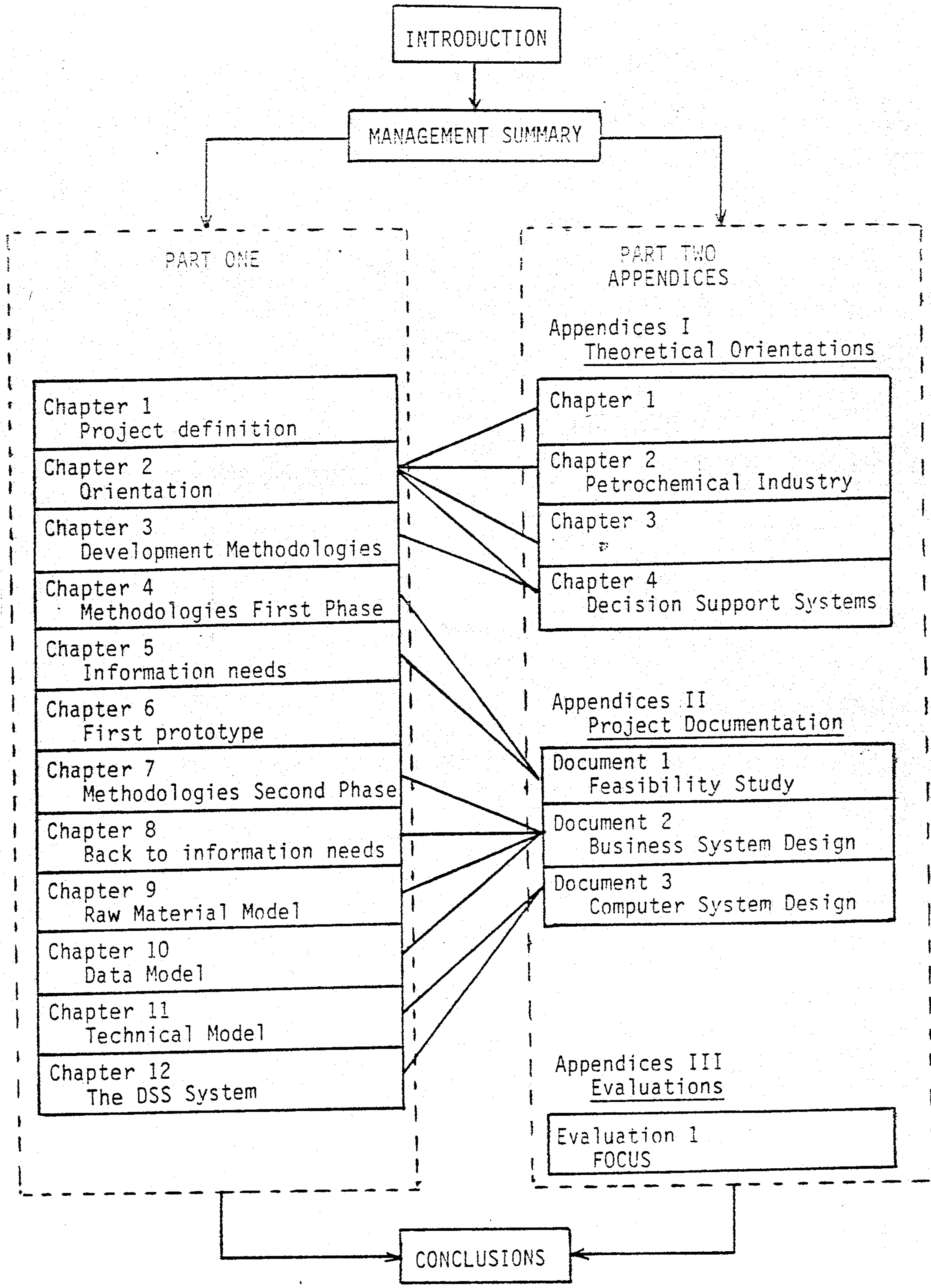


fig.1. Overview report structure

1. PROJECT DEFINITION

After the first meeting with the supervisors at the following project definition was agreed upon.
The initial project definition, was to:

1. Investigate the information needs in the "material flow management" functions, which can be supported by means of a Decision Support System.
2. Construct a Data Model based on the investigation and analysis of the information needs in point 1.
   Investigate how the data defined in the Data Model can be derived from data, coming from transaction systems.
3. Evaluate 4th generation tools (Adabas-Natural, FOCUS, TIF etc) by which the DSS can be constructed. Aspects of the evaluation can be:
   - Productivity improvement in regard to more conventional application development tools.
   - Effects on necessary hardware.
   - Effects on the users.
   - Guidelines for usage.

After further discussion with the mentor at initially Ferdinand followed by Doug (both of the Computer Service Department), the scope of the project was changed.

Furthermore it was decided that it should not be a theoretical exercise; the Data Model should also be implemented into a working system, using one of the Fourth Generation tools mentioned in point 3.

CONCLUSION

The project consists of the investigation, analysis and implementation of a Decision Support System , aimed at the managers, with an emphasis on the Production Characteristics.

2. O R I E N T A T I O N

INTRODUCTION

Looking at the project definition several questions arose, such as

- What is a Petrochemical Industry and what are its characteristics and problems ?
- What are Decision Support Systems ? Where are they aimed at ? How should they be constructed ?
- Fourth Generation languages. What are these ? What can they do, and can't they do ?

All these questions should be answered before one is able to analyse the problem situation.
In the following sections these questions shall be answered.

2.1.

2.2. PETROCHEMICAL INDUSTRY *

Petrochemicals are derived from petroleum (crude oil) fractions (and by-products), but also from ethane, propane and butane.

Major characteristics of petrochemicals are: large volume, low number of production steps, low degree of differentiation, high capital intensity and a price competition. Petrochemicals are bulkchemicals. Examples of petrochemicals are Ethylene, Propylene, PVC, Polystyrene, Polyethylene etc.

The petrochemical industry starts with cracking the feedstocks, Naphtha or LPG, into Ethylene, Propylene, Benzene and Butadiene in a so-called 'cracker'. This is the first and most upstream plant in the chain of petrochemical production facilities. At a cracker long molecules 'are broken into' short molecules at temperatures of approximately 850 °C. Next the mixture of molecules is distilled to pure product: ethylene, propylene etc.

After the cracker several follow-up (downstream) plants transform the basic components into end products.

Not many of the products manufactured by are consumer-products. Most of products serve as raw materials to other industries, consequently products are not very well known to the public. For example the plastics leave as granules, other industries melt them again and produce consumer products.

* (see also chapter 2 of appendix A in the second part)

2.3. PETROCHEMICAL MARKET SITUATION *

The profitability of bulk-products is directly related to the differences in the total cost of the competitors.
Differences in total cost come from many factors, such as, used production technology, size and age of the production plant and cost of feedstocks and energy
Production and marketing of petrochemicals takes place all over the world. The major markets and production centres are Western Europe, USA and Japan. Towards the end of the eighties however, Canada and the Middle East are expected to take over the lead as petrochemical exporters.

During the seventies the fixed/variable cost ratio for producing petrochemicals has changed completely. At the beginning of the seventies fixed cost accounted for 80% of the production cost. After two energy crises fixed cost accounts for 10% to 20% of the production cost. So because of the sharp rises of feedstock and energy cost, production of petrochemicals is now a variable cost operation.

The current problems of the petrochemical industry can be explained by the combination of several factors:

1. Supply side
   - Price rises of oil with their impact on the price of raw materials and the price of energy.
   - Aging of the main production technologies (which reduce the scope for further economy of scale and energy savings).
   - The appearance of new competitors, causing a huge overcapacity.
2. Demand side
   - Maturing of many markets for the products of the industry.
   - Slow growth in demand (now 1.5% per year).

Products and industries that get into the maturity stage of their lifecycle are often faced with the following problems:

- Slow growth leads to more competition for market share.
- Buyer-power increases as customers become more experienced in dealing in a declining business. Prices become increasingly dominant in sup-

plier selection decisions.

- Overcapacity frequently prevails for some time while industry adjusts itself to lower growth margins and profits decline, resulting in a shake out of companies.
- Cost becomes an overriding factor in competition. Capital investments are needed to decrease operating cost at a moment that profits are falling.
- International competition increases (falling exports, increasing imports) if technology matures, patents expire and the maximum penetration level is reached, all competitors seek wider markets.
- Increasing government, labour union and/or EC involvement, when the industry is of national/European importance. This poses a poor divesting climate, companies can't go out of business because of exit barriers. Companies can't go out of business due to integration, interrelations, between the Production Units a Company operates. A company can not close one operation without influencing other Production Units.

How can a company react to these problems ?
Michael Porter (PORTER80) describes a framework of possible successful strategies a company can undertake:

1. Overall cost leadership.
   This is producing at the lowest cost, this requires a high market share. When market share is high, the company can profit from declining costs per unit produced when the production is increased, because of learning curve and economies of scale effects. Or the company needs other advantages, such as favourable access to raw materials.
2. Differentiation.
   Differentiating the products of the firm, in order to create something that is perceived industry wide as being unique.
3. Focus.
   Focusing on a particular buyer group, segment of the productline, or geographic market.

2.4.

2.5. D E C I S I O N   S U P P O R T   S Y S T E M S

A Decision Support System (DSS) is a computer-based information system used to support decision making activities in situations where it is not possible or not desirable to have an automated system perform the entire decision process (GINZ82).

There are several ways a DSS can be characterised, that is, in functional parts of a DSS, in DSS usage patterns and in DSS development patterns.
A DSS consists of three parts, a Data Base, a Model Bank and a User Interface (POLY84). The user interface contains software which enables the user to apply the Models to the Data available in the Data Base.
A second characteristic of DSS are the patterns of usage: these are amongst others: evolutionary, interactive and unplanned usage.
The third characteristic of DSS consists of the DSS development patterns: the development process should be participative and oriented towards change.

A DSS can be used to support the manager in his strategic decision making process.
What kind of information does the manager need and what characteristics can be derived from strategic data ?
McCosh, Rahman & Earl give in (McCOSH82) some characteristics of strategic data. Strategic data are:

1. External rather than internal
2. Future oriented rather than historical
3. Aggregate yet robust
4. Quantitative and qualitative
5. Partial and transient
6. Formal and informal

The expected benefits from using a Decision Support System can be (KLEIN85):

1. Increase in number of alternatives examined.
2. Better understanding of the business
3. Fast response to unexpected situations
4. Ability to carry out ad-hoc analysis.
5. New insight and learning.
6. Improved communications.
7. Control
8. Cost savings.
9. Better decisions.
10. More effective teamwork.
11. Time savings.
12. Making better use of the data resource.

Decision Support Systems provide however a limited kind of analysis. They're used to plot a path to an objective by achieving the best solution within the stated constraints. So a DSS used without creativity and without criticism poses a great danger.

* (see also chapter 4 of appendix A in part two)

2.6. F O U R T H G E N E R A T I O N T O O L S * **

James Martin gives characteristics a tool should have in order to be a 4th generation tool (MARTIN81):

- It is user-friendly.
- Somebody who isn't a programmer should be able to get results out of it.
- The tool should use non-procedural codes.
- The tool makes assumptions of what the user wants, and the tool sets the defaults accordingly.
- The tool should have a possibility to be operated on-line.
- Non-DP-professionals should be able to learn a relevant subset of the language in two days.

An important characteristic of a 4th generation language is that it uses non-procedural code, this as a contrast to procedural code. Procedural code specifies HOW something is achieved. Non-procedural code specifies WHAT should happen, how it is done remains below the surface. The objective, the WHAT, which has to be achieved, is most central.

FOCUS can be considered as such a 4th generation language. FOCUS is an integrated tool; it employs its own hierarchical database, query and reporting language. Further it incorporates statistical, graphical and financial facilities.

* (See also Evaluation 1 of appendix C of part two)

** (See also report of author about Fourth Generation languages)

2.7. I N F O R M A T I O N S Y S T E M T Y P E S

Managers need information not computer systems. The latter is a means to the former end. Managers need flexible access to relevant data and the ability to analyse that data.
Computer-based information systems are intended to provide a means to make this feasible, economical and easy for managers.

In analysing the systems a company uses, a classification of systems can be made:

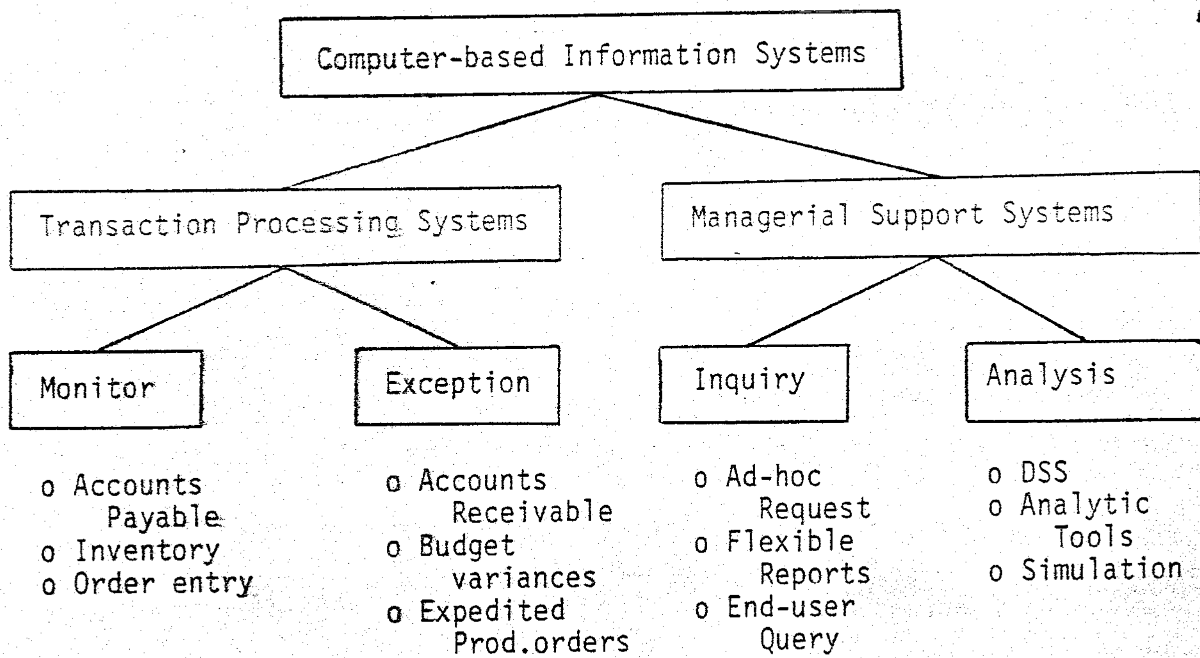


fig.2. Classification of computer-based information systems, source (ALLO82).

Monitor — These systems monitor daily detailed activities, producing standard reports on a fixed schedule (daily, weekly, or monthly).

Exception — These systems produce detailed activity reports on command, the definition of exception reports is fixed.

Inquiry — These systems provide a database with flexible inquiry capability, enabling managers to design and change their own monitoring and exception reports.

Analysis — These systems provide powerful data analysis capabilities (moddelling, simulations, optimisation or statistical routines) and the appropriate database to support managerial decision making.

Transaction processing systems generate management reports by successive stages of summarisation of detailed activities. There are however, problems with regard to this.
The implicit assumption in this traditional MIS approach to management information is that summarised daily activity, which is appropriate for first line managers, when further summarised, is also appropriate for higher levels of management.
Mintzberg gives several reasons why MIS were unsuccessful in supporting semi-structured decision making (MINZ75):

- The formal system is too limited.
- Formal information systems tend to aggregate data, as a result much of the information produced is too general for the manager.
- Much formal information is too late and unreliable.

Inquiry and analysis systems are more managerial oriented in their intention, design and use than transaction processing systems. It is the difference between starting with the data and sending summary reports to the manager's information needs, and, working down to the data and the systems necessary to support those needs.

C O N C L U S I O N S

To develop a DSS requires a completely different development approach in comparison to a transaction processing system.
The DSS should be aimed at the information needs of the managers.
The development process of a DSS is not a straightforward one, it is evolutionary in nature, a very tight communication between the users and system analyst is required.
The system is not one which uses transaction data, but external data (i.e. external to the firm). Future data are also required.
In the terminology of section 2.7., needs an inquiry and an analysis kind of system.
The eventual system should be tailored to suit the information needs of the managers.

3. D E V E L O P M E N T   M E T H O D O L O G I E S

As stated in section 2.5., the development process of the DSS is an evolutionary one, this doesn't imply that the development process is an un-structured one.

Prototyping seems to be an appropriate development methodology which can be used in developing a DSS
A prototyping approach fits in the evolutionary development philosophy. The information needs, which are sufficiently structured, are translated into prototypes of the eventual system in a relatively short time (days, weeks). These prototypes can be used as a feedback to the user, he can give comments on these. The prototypes can be developed into the final system in a series of these cycles. Whether a final system can be reached remains a question, because the evolution process of the managers, and thus of the system, continues.
The advantage of this approach is that the user and the analyst can profit from the experiences gained from the prototype in order to structure the information needs of the users in a step-by-step fashion. The basic philosophy behind this evolutionary approach is that the development of an information system is a learning process, both for the user and the analyst.
A requirement for this prototype approach is the availability of hardware and software which makes it possible to construct a working prototype within a reasonable period of time and using reasonable resources.

In developing information systems one can make a distinction between process and data orientated approaches. In the process approach the analyst takes the activities and the data processing as a start, followed by an analysis of the data which should be available for these processes. The data approach starts with the analysis and structuring of the necessary data, and then analyses what should happen with the data. The ~~development of this~~ DSS is not process oriented, but data oriented, it is aimed at the data which is being used in the 'companies processes'.

These concepts, however, are not sufficient enough to carry out a development process. In order to have a manageable development project, some kind of supervision should be exercised over the development process, a project control method is used.
A development process (also a cycle of a prototype approach) can be split into phases: Preliminary investigation, Analysis, Construction and Implementation.
In , the Computer Service Department (CSD) uses its own kind of methodology, the so-called MAP method. This is a method derived from the SDM (Systems Development Methodology) from Pandata. Map is also, like SDM, a project control method.
Using such a method too strict causes some problems, such as rigidness in the development process, there is no flexibility, mistakes or misconceptions can't be corrected in an iterative loop. Another disadvantage of these methods (MAP and SDM) is that they don't give sufficient substantial assistance in the development process, in other words, they do not give a technique to use. ,MAP indicates WHAT

should happen, not HOW it should be done. It is a project management method.

It is therefore necessary to choose a technique in every phase of the development process.
The MAP methodology recognises a 'Project Development Cycle', this is a sequence of phases in a project between the first problem indication and the end of all project development activities i.e. the operational system.

Control is achieved by setting clearly defined objectives for successive phases. The output per phase is used:

* To evaluate the results obtained per phase against its objectives.
* As a basis for the next phase (the objectives and planning).
* As an outline basis for the whole project (the objectives and the planning).

The Project Development Cycle of MAP divides the project into 5 major phases:

1. Feasibility Study
2. Depth Study
3. Business System Design
4. Construction
5. Implementation

In the following chapters objectives of each of these phases shall be explained. In each phase a technique shall be chosen, which shall be used in that particular phase.

C O N C L U S I O N

A prototype approach shall be followed in constructing a DSS for the Plant Profile Project. This is an evolutionary approach, which is the most suitable method in building a DSS. The development of a DSS is evolutionary because the system requirements of the users will fluctuate in time.
In order to have project control the internal method MAP will be used, which is derived from the SDM method.
It should be remembered that MAP is a traditional method, used for developing transaction processing systems, which does not support the evolutionary approach of the prototype methodology.
Another point is that MAP is basically a method which can be used to exercise control over large projects. This DSS project is relatively small, so it is possible that there's some kind of overkill while using the MAP method in this project.
It can thus be expected that the MAP shall not be followed to the letter, because of the arguments given above.
Whatever kind of system is being developed, a 'project development cycle' can always be recognised !

In developing a DSS, a data orientated approach seems to be better than a process approach, because the DSS is aimed at the information requirements of the managers, which are not derived from business processes. A process approach is aimed at the business processes within a company.

4. METHODOLOGIES FIRST PHASE

INTRODUCTION

Phase one in the MAP method is called the Feasibility Study.
The objectives of the Feasibility Study are to:
- Obtain a clear understanding of the problem area, its scope and limitations at 'User-Management' level.
- Study the area in sufficient depth to identify and evaluate basic solution options and select the best one.
- Determine the expected results of the solution selected plus cost-benefit analysis.

The first activity in the Feasibility Study phase is to establish a clear understanding of the project definition, scope and constraints. After this is done the information needs of the managers can be assessed and analysed.

4.1. PROJECT DEFINITION

The persons involved in this project other than the author as system analyst are:
- The 'major' user, this is the representative of the user department; he initiated the project.
- The users, these are all people who might have an interest in the results of the project.
- The supervisors, one from the users department and one from the Computer Service Department.

These roles need to be fully understood by the actors. After these roles were established the project definition was put up.

The Plant Profiles project has the following objectives, scope and constraints:

OBJECTIVE

SCOPE
The information, which the system provides, is intended for the management and shall be used to support the decision making of those managers; so the presented information must be of the right aggregate level.

The information will also be of great help for the Europe) product marketing teams and business marketing teams in which many managers participate.

It's not expected that managers themselves operate the database from start on. The database will also be operated by Economic Evaluation, Product Flow Planning and product marketing managers.
The owner of the system will be the Economic Evaluation function.

CONSTRAINTS

- business:
  * The system must fit in with the ISIS concept. This is the Integrated Site Information System, which is building.
  * The system must employ, as much as possible, existing internal and external databases.
  * The code structure, etc. must be compatible with the currently used code structures.
- environment:
  * Because information will be used throughout the departments and plants, support is needed from all concerned.
  * The project shall be tackled with the MAP methodology.
  * The databases must be secured, so that only authorised people can alter and access the data.
  * There must be a high degree of user-friendliness by means of menus and question- and answer-screens.
  * Possibility to communicate with Europe) computersystems, in particular DSS-systems.
  * Possibility to communicate with PC and PC-software (downloading).
- technical:
  * The system shall be developed with FOCUS or PC/FOCUS. This with regard to the need to evaluate FOCUS and PC/FOCUS as a future development tool for Decision Support Systems.
  * The application must be convertible from PC-version to mainframe-version and vice versa.

4.2. ASSESSING INFORMATION NEEDS

The target area was still too large to be manageable. In cooperation with the Major User the target group was limited to the Plastic Department.

It can therefore be expected that the plastic managers give their support to this project because they have some experiences in managerial support systems

The following functions were contacted and interviewed:

What these functions mean and how they are related to one another can be found in Chapter 1 of appendix A in part two.

4.3. METHOD: CRITICAL SUCCESS FACTORS

MAP doesn't give a technique which can be used in assessing the information needs of the managers in question.

The following approaches can be considered in doing an information analysis to get the information needs of managers (ROCK79):

1. By-product approach
   In this method, management information is an offspring of existing transaction processing systems. These are the monitor and exception systems mentioned in chapter 2.
   The DSS for competitor intelligence is looking outside the company, so it cannot be expected that there are transaction processing systems which can give relevant information.

2. Null approach
   This method states that managers' needs are dynamic, and that they can't be gathered in a system.
   This is only partly true. The Plant Profiles Systems should be developed in such a way that new information needs can be added at will. This was already stated when saying that the development process of a DSS is evolutionary and therefore never ending.

4. Total study approach
   Here a widespread sample of managers are queried about their total information needs, and the results are compared with the existing information systems. One of the methods based on this approach is the Business System Planning method (BSP).
   This total understanding process is expensive in terms of manpower and all-inclusive in terms of scope.
   Because the plant profiles project is relatively small, the BSP method can be considered to be extensive.

5. Critical Success Factors
   This method focuses on the individual managers and on each manager's current information needs, both hard and soft.

The CSF method seems to be the best choice for assessing the information needs of the managers.
The CSF method tries to identify the factors which must be performed exceedingly well for a company to be successful. The CSF's are areas of activity that should receive constant and careful attention from management. CSF's differ from company to company and from manager to manager.

Rockart isolated five prime sources of Critical Success Factors:
1. Structure of the particular industry.
2. Competitive strategy, industry position, and geographic location.
3. Environmental factors.
4. Temporal factors.
5. Managerial factors.

Critical Success Factors have several characteristics:
- Many CSF's require information external to the organisation, information concerning market structure, customer perceptions or future trends.
- CSF's may require coordinating pieces of information from multiple data sets that are widely dispersed throughout the company.
- CSF's may require subjective assessment on the part of persons who give the information, rather than being neatly quantifiable.

The Critical Success Factors method appears to be the proper one to use in assessing the information needs of the managers, because of the characteristics listed above.

How should a CSF method be conducted ?
All the managers identified in section 4.2. shall be interviewed. The following topics need to be discussed. First the interviewed managers are asked to describe their jobs, and the areas which they believe are most critical to them. The CSF provide the managers with a vehicle for thinking about their information needs. Finally the CSF method focuses on the definition of those information data elements which are necessary to support the information needs of the managers.

In order to have a discussion with the managers, the interviewer should have some basic knowledge about the problem area.
On the other hand the managers should know what to expect in the interview. They should know the objective, scope and constraints of the project.
In order to communicate these goals a letter was sent to each manager, asking whether he would be prepared to read and participate in an interview about the project.
An interview questionnaire was set up. The interview questions can be found in appendix 2 at the end of this report.

C O N C L U S I O N

The method used for assessing the information needs of the managers was the Critical Success Factors method of John F. Rockart.
This method tries to identify the so-called Critical Success Factors, which are important indicators to determine company success.
They may be considered as a vehicle to get the managers talking about their information needs.
The method was carried out by interviewing the managers individually, after they were given the opportunity to review the objective, scope and constraints of the project.

5. INFORMATION NEEDS

INTRODUCTION

After the managers received the letter describing the project and what was expected from them, they were contacted to make an appointment for an interview.
In this chapter the most important results of these interviews shall be presented. Next an evaluation on the usage of the Critical Success Factors method for this project shall be presented.
The interviews had two important objectives. The first one was to assess the information needs of the users. Second, to inform them of the development of a system were they could possibly profit from.

5.1. RESULTS INTERVIEWS

The MAP methodology indicates that key leverage points should be identified. Key leverage points are those operational parameters which have a major impact on performance in the user area.
So key leverage points can be seen a some sort of Critical Success Factors.

The following key leverage points were recognised in regard to the plant profiles project:
- The need for a flexible, accurate information system

- The possibility to 'model' the data in order to do 'what-if' analysis, sensitivity calculations, and to produce reports in the layout the user wants.
- The need to 'shield' the data from unauthorised use.

The following information needs consist with the managers:

The managers had also other kind of demands, which can't be classified as information needs.
It is a listing of system requirements. These requirements are the following:
-Necessity to present the information in a layout the user wants (flexibility).
-Quick response.
-It will not be likely that the managers themselves will operate the terminal within short notice.
-It is very important to be able to display only those items which have significantly changed.
-The need to verify data by means of a reference (where does the data originate from ?).
-The possibility to indicate how reliable the resource data or the results are (ex. range the reliability from very doubtful to very reliable).
-To be able to include "text-blocks" (one or two pages of information).

These are "must" objectives that need to be accomplished in order to be effective for the users.

The managers were asked to list the data elements they wanted information about. A list of these data elements is presented in appendix 3 of this report.

5.2. COMMENTS ON USAGE OF METHOD

The following problems arose while trying to conduct the interviews:
- The managers were extremely busy, it took a very long time to arrange the interviews.
- The interview appointments were often changed, because the managers hadn't had the time, or that they didn't show up, because they forgot to cancel the appointment.
- Because of these delays the interview period took too long, approximately 6 weeks.
- The managers, especially the superintendants where mainly busy with day-to-day running of their plant, their involvement in strategic decision making was not large.
- The prepared interviews almost couldn't be conducted. The interviews went very unstructured. Talking about Critical Success Factors was not a success.
- Not many new information requirements came up from the interviews, what wasn't already known.
- A few interviews sanded into discussion about data security and privacy aspects.

Taking all these points into account several conclusions can be made. First, the Critical Success Factors method is probably a good one to use in assessing information needs, but not on the target group, because the major objective of the interviewed managers was to solve day-to-day problems. So the target group may not be the correct one to use the CSF method on.
Second, the interviews needed more preparation. Better is to send the list of interview questions in advance to managers, in this way they can think about it. Maybe it is even better to construct a questionnaire and send it to the managers. This gives a faster response.

C O N C L U S I O N

The managers considered the system as being useful, but were a bit defensive. The system was not considered to be critical to their functions.

The most important 'model' the managers were interested in was a model giving an insight into an integrated site . that is, how the plants are connected to each other.

The results of this initial phase, the feasibility study, are put in the feasibility report which can be found as document I in appendix B in part two.

After the report was presented to the supervisors, the next phase of the project was approved.

6. F I R S T P R O T O T Y P E

I N T R O D U C T I O N

After the information needs of the ' , managers were assessed, the first prototype was constructed.

The construction of the first prototype had several objectives:
- To prove that it is possible to build such a system.
- To be able to give the users feedback about the system, in order to generate thought about their requirements.
- To investigate whether the tool FOCUS is sufficiently equipped to be used.
- To gain experience in programming in FOCUS and building databases and programs with it.

6.1. P R O T O T Y P E

The system the users want, can be classified as a combination of an inquiry and an analysis system (see section 2.7 of this report).

The objective of the first prototype was to build a system which makes it possible to do inquiries. Modelling aspects, needed in an analysis kind of system (DSS) were not considered.

The information needs were broken down into entities, and the entities into attributes.
The next step was to construct the relations between the entities. Since FOCUS employs a hierarchical database, it was decided to use the hierarchical facilities FOCUS supplies to build the prototype.

The entity types which were identified were:

Company
Location
Plant
Process type
Product
Feedstock
Yearly Plant Information

Between these entities hierarchical relations were constructed.
Companies have many locations; in a location several plants have their operation, in a plant several process types can be used, the plant can produce several products, using a number of feedstocks, capacities of the plants were monitored yearly.

The entity types of the first prototype and their relations can be found in fig.3. on the following page.

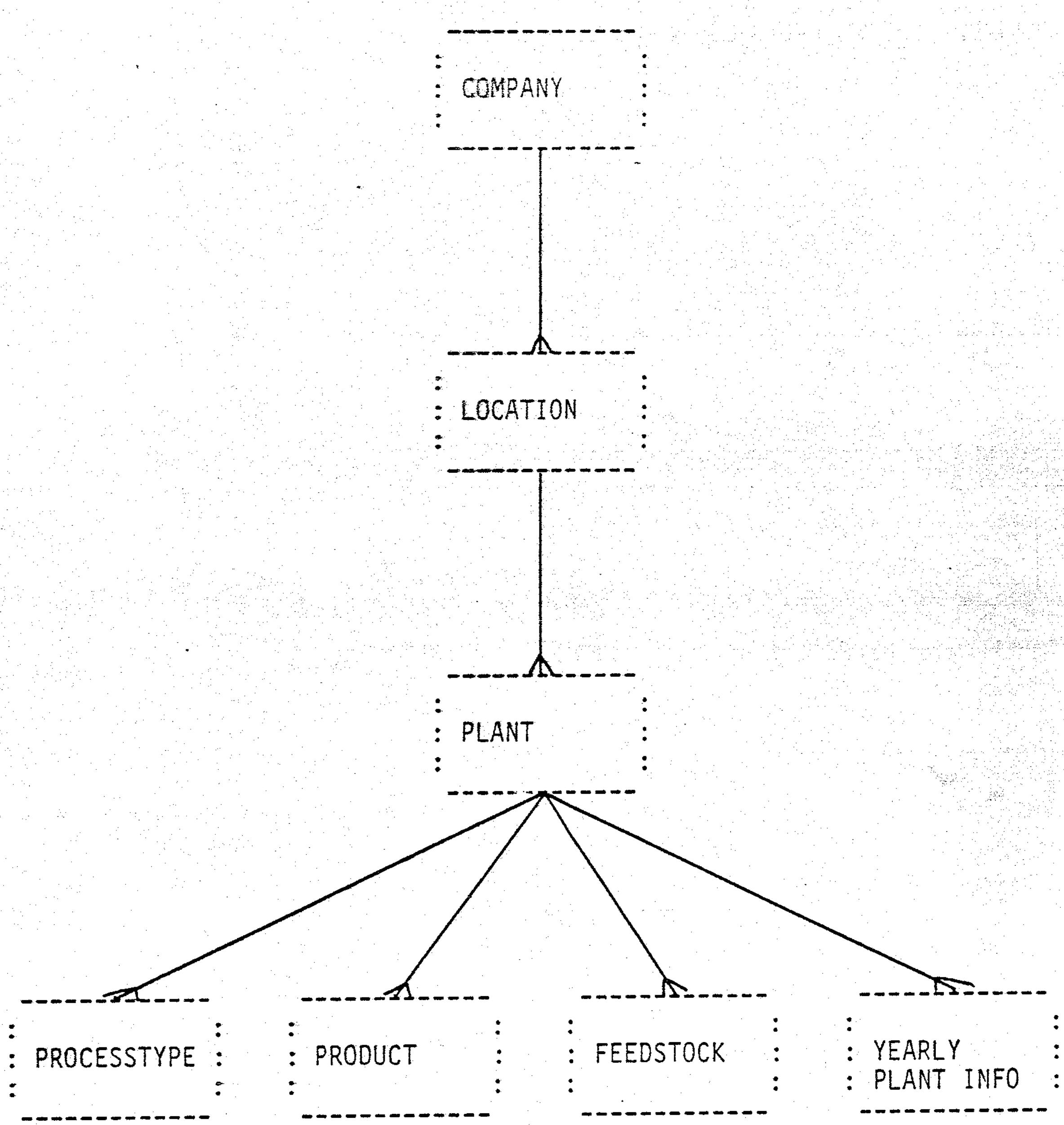


fig.3. Hierarchical infological model of the first prototype

This hierarchical structure was implemented into a FOCUS database. In order to be able to load data into the database, input screens were developed.
The input programs were constructed in a user-friendly way, the user can go through a series of menus to get to the place where he/she may type the data in.

Several output programs were developed to show that it's possible to obtain reports from the system.

The system was implemented in PC/FOCUS, the Personal Computer version of mainframe-FOCUS.
The tool FOCUS was used because it's the standard tool, within , to construct managerial database systems.
Even though FOCUS is the standard, nobody in had any knowledge about FOCUS. Nobody could answer programming questions. This, in combination with the user-unfriendly documentation of FOCUS, made the development of the database and the programs a bit difficult.

6.2. RESULTS OF THE FIRST PROTOTYPE

It was proven, by means of the prototype, that it is possible to build a database system which allows enquiries about data elements stored in the database.

After discussion with the major user, the prototype was refined several times, and the information needs clarified.

The major user and the analyst perceived at this moment that there was also interest for this system in the European headquarters Switzerland. Several 'strategy makers' are located there who can benefit from this system. Presentations of the system to managers and computer experts in led to a decision to expand the operation to an European basis.

The system was received very well and it was decided to go ahead.

CONCLUSIONS

After the entities were identified, a hierarchical database was constructed. This database, together with some input and output programs formed the first prototype.

It was decided to expand the project to European level, after the system was presented to the managers and computer experts.

It was decided to conduct a renewed information analysis, using the gained experiences of the first prototype, by carrying out an extensive analysis of the information needs assessed in the first phase of the project.

7. M E T H O D O L O G I E S   S E C O N D   P H A S E

I N T R O D U C T I O N

The objective of this phase is to construct a sound infological model of the object system.
After the model is constructed, this model shall be implemented into a database using the tool FOCUS.

7.1. B U S I N E S S   S Y S T E M   D E S I G N

After the Feasibility Study the next phase in MAP would be the Depth Study.
The objectives of the Depth Study are:

- To study the current system at operational level.
  Comment: there is no operational level in the plant profiles project.
- To identify the functional requirements for the new information system.
  Comment: These requirements are vague and the essential parts are listed in section 5 of the feasibility report in appendix B of part two of this thesis).
- To establish recommendations for immediate improvements.

,

decided was to skip the Depth Study phase.

The next phase in the MAP cycle is the Business System Design.
The objectives of the Business System Design are:

- To provide the user with a complete and detailed functional description of the proposed system (logical part).
- To identify those functional elements which will be automated and those elements that will be performed manually (physical part).
- To provide a firm basis for the construction phase in which the system is built.

7.2. M E T H O D O L O G Y

The techniques suggested in the BSD-phase are process oriented, not data oriented.
What is needed here is a methodology how the development of a data-orientated system should look like.

In (POLY84) a good picture is presented, which relates the phases in a data oriented approach, for developing information systems, to each other (see fig.4.).
This figure gives the position of information analysis and data analysis in relation to one another. An important conclusion coming from this picture is, that after the data model has been put up, the programs and the physical storage structures can be developed parallel. Independence between the physical storage structures and the programs is accomplished.

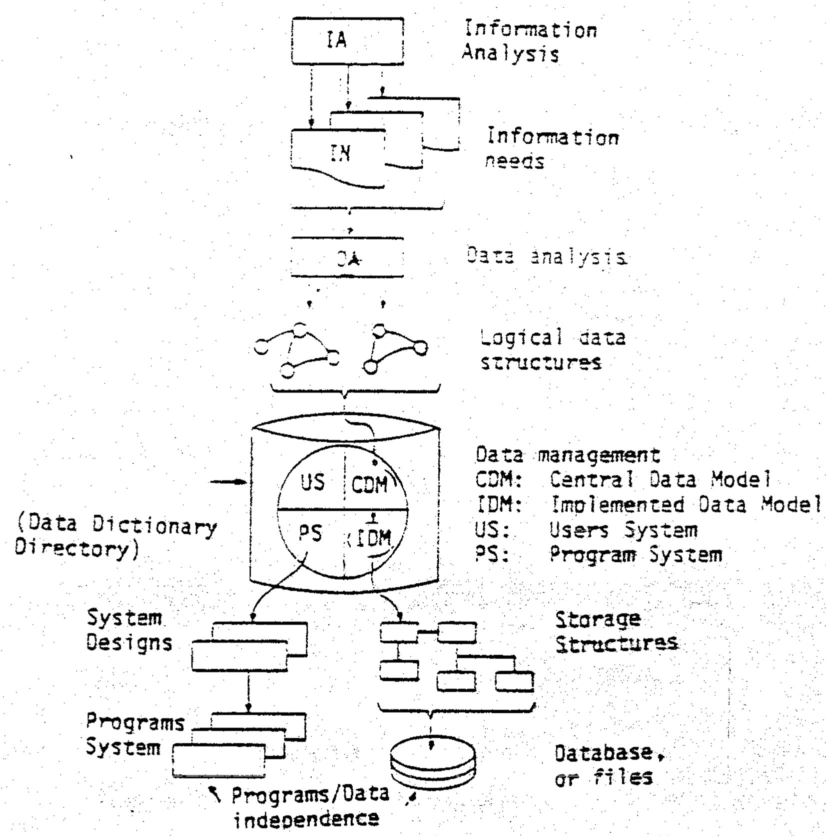


fig.4. Place data analysis and modelling in the development process of an information system (POLY84).

Another classification of the development process is in aspect systems, represented in the model cycle in figure 5 on the following page.

The first step in the 'model cycle' is to make an information model of the system which is to be controlled.
This model, the Systelogical Model states which management functions need WHAT information, and WHY and WHEN, to make decisions. The most important thing to recognise in this model is the connection between information and organisation. The end-users of the system will have their interest focused at the pragmatic aspects of information:

- -Does the information get to them in time and in a form that they can use it (Cybernetic sub-aspect).
- -Does the information get to them on the right aggregate level and is it well readable (Human Factors sub-aspect).
- -Does the information system pay off (Cost-Benifit sub-aspect).

The second model is the Infological Model. Given the WHAT and WHY of the information, this model gets into the information and the business processes generating the information. Which information items serve as a input for the desired information, and does everybody define the information elements the same way (semantical aspects).

The third model in the cycle is the Data Model. This model gives answers to questions as: how does the information processing look like. In this phase the syntactical aspects are most important, that is what kind of coding, what kind of data base organisation is to be employed.

In the fourth model, the Technical Model one chooses the tools to store, to access and to model the data.

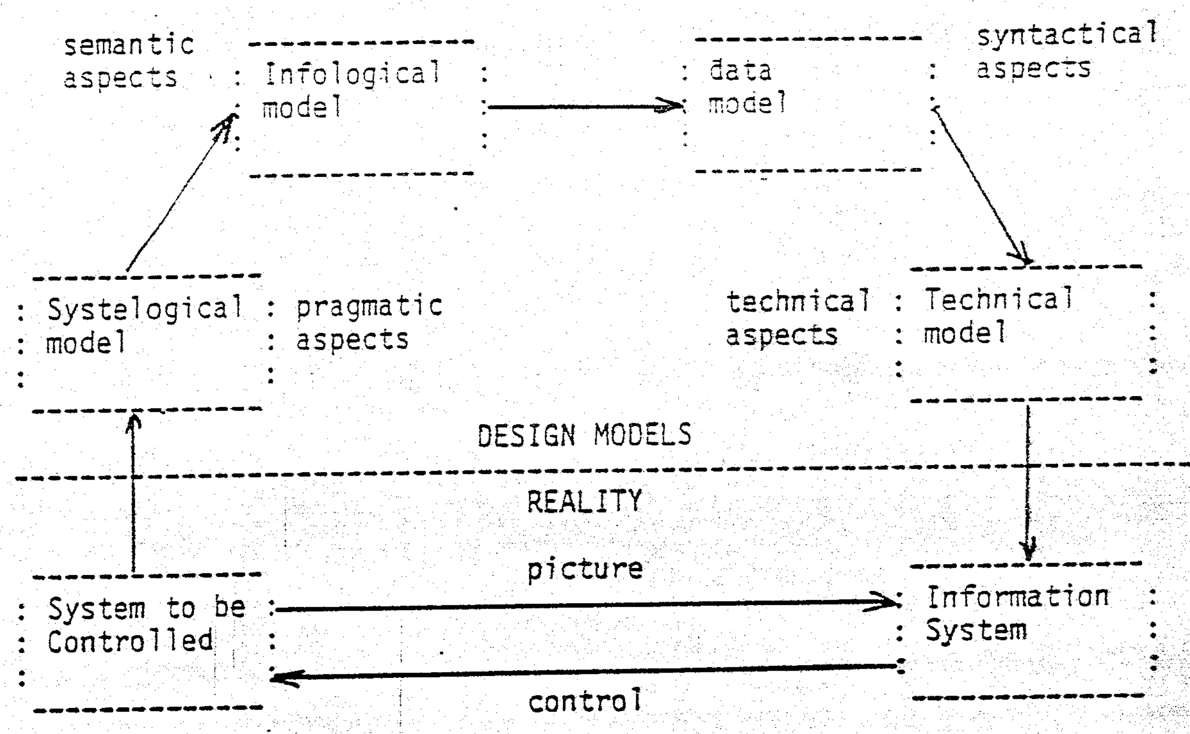


fig.5. The modelcycle (aspect systems) source (BEMEL83)

The model cycle, does suggest a straightforward process, where one model succeeds the other. But it is a cyclic process, the models are refined in iterative steps, much like a prototype approach where the information systems are refined in each step.

This aspect characterisation is also implicit present in fig 4. In this figure the emphasis is on the action to be performed in the development process.
In both figures the independence of the logical model (structure) and the physical model (structure) is stressed.

The model cycle and the information analysis figure give a philosophy how the development of an information system could be.

7.3. M E T H O D : I N F O R M A T I O N M O D E L L I N G

The method chosen in this phase is the Information Modelling Method of Hoskyns.
This is the method which is presently used within   to do the analysis.
This method starts with decomposing the information needs of the users into entities, it starts with the Infological Model.
So in the Hoskyns method the model-cycle can be recognised implicitly.

The information modelling of Hoskyns consist of:

- Data Analysis
- Normalisation
- Optimisation
- Access Path Analysis
- Conversion to technical model
- File construction

C O N C L U S I O N

The methodology used in the Business System Design phase is the distinction in the development of aspect models; the so-called 'model-cycle' (BEMEL83).
This methodology identifies, the systelogical, the infological, the datalogical and the technical model. This is not a straightforward following up of the models but a iterative process, the models are refined in each iteration.

The method chosen to do the data analysis is the Information Modelling technique of Hoskyns.
This method is chosen because this is the technique currently in use at . Secondly it seemed that the method could fit in the aspect method characterisation, although this is not explicitly stated in the method. Thirdly, the method gives a number of techniques which are to be followed with well-defined check points.

8. B A C K T O T H E I N F O R M A T I O N N E E D S

I N T R O D U C T I O N

In a previous phase the information needs of the managers were assessed by means of interviewing them.
After these information needs were assessed the first prototype was made. In the first prototype entity types were identified and analysed using common sense. This first system did not cover all of the information needs, also the relations between the entity types were not correct.
The first prototype was used as a vehicle to get the users thinking about their requirements. A major requirement which came up was the requirement to have good definitions for the entity types.

So in order to conduct a sound data analysis, the information requirements need to be reviewed.

8.1. C A T E G O R I S I N G I N F O R M A T I O N N E E D S

The information needs of the managers can be summarised and grouped into two categories:

1. Queries
   These are ad-hoc queries or predefined questions

   These queries can be answered without any calculations on the data.
   Example: *

   *

   The answers on these kind of questions fulfil the information needs on 'static' facts

2. Models
   The information needs, which can be supported by models, consists of questions about the relations of a certain (user defined) group of factors with each other. (So the user can add or delete data elements to his model, given a certain database which restricts the number of data elements he can access).
   Examples: * How well is a site integrated ?
   (Upstream/Downstream integration; flow diagrams)
   * What-if questions.

   *

These two groups pose completely different demands on the information system.
However the infological model should be independent of the physical information model.

The Infological Model gives a description of how the users see their environment, it is a logical description.

The queries can be seen as an inquiry system, and the models as the analysis system (see fig.2. in section 2.7.).

8.2. ANALYSING INFORMATION NEEDS

The Data Analysis phase of the Hoskyns Method can be summarised as a series of key questions to be asked to the users, with supporting documentation provided by the analyst.
The key questions are:

1. What are the entity types the organisation is interested in ?
2. What should we call them ?
3. What are the known relationships between the defined entity types ?
4. What attribute types are associated with each entity type ?
5. What should we call them ?
6. What is the range of values for each attribute ?

The information needs which came up from the interviews and the experiences coming from the first prototype were taking as a start. An extensive analysis of the information needs can be found in document 2 in appendix B of part two.

The following entity types can be distinguished from the information needs:

Entity types: COMPANY
INDUSTRIAL AREA
PRODUCTION UNIT
PRODUCT
RAW MATERIAL
PROCESS UNIT
PROCESS

These entity types are partly the same as in the first prototype. Relations, identified between the entity types, are however completely different (see following text).

The second key question, in the Hoskyns method, is how to name the entity types. The names listed above are good, but the entity types need to be defined exactly.

Definition entity types

COMPANY : Independently operating organisation which has its own unique legal status. With an organisation is meant the total company, ex.
Joint ventures shall be considered also as an occurrence of the entity type COMPANY, there should however be an indication who is the owner and the controller of the joint venture.

INDUSTRIAL AREA: Geographical spot with production facilities.
An Industrial Area is independent of a Company. (The entity type Location of the first prototype was Company related.)

PRODUCTION UNIT: A group of processing units forming a series of actions in manufacturing a particular product.
A Production Unit can be compared with a group of machines, which are functional the same, that is which perform, or produce the same function or product.

PRODUCT : Something produced by a Production Unit or a Process Unit.

RAW MATERIAL : Input to a Production and/or Process Unit

PROCESS UNIT : A group of equipment forming a continuous operation or treatment in manufacturing a Product.
A Process Unit is a part of a Production Unit.
It can be considered as a single machine in a cluster of machines (a Production Unit).

PROCESS : Technology type used in manufacturing a product.

These entity types can be distilled out of the information needs at this moment. This doesn't mean that it isn't expected that the information needs change. Changes are inherent to this kind of systems. But that doesn't mean that the analysis should go on and on. The system should be flexible enough to be able to incorporate new information needs.
Possible future enhancements are:

- Supply side; where do the raw materials come from, transportation cost. Distance between supplier to the company.
- Demand side; where are the main customers located in regard to the production facilities which supply the products, transportation cost. Distance between customer and the company.
- The relation between the sites, for instance pipeline grids, or inter-company transactions.

Between the entity types, described above, the following relations can be identified:

- A company can have operations in a number of Industrial Areas, and in an Industrial Area more than one Company can have its operations. This is a many-to-many relation.

- A company can operate many Production Units, but a Production Unit always belongs to one Company. This is an one-to-many relation.

- In an Industrial Area several Production Units can have their operation. But a Production Unit can only belong to one Industrial Area. This is also an one-to-many relation.

- A Production Unit can consist of several Process Units. A Process Unit belongs always to one Production Unit.
Another name for a Production Unit is a Plant, another name for a Process Unit is a Train.
The Process Units in a Production Unit are not forming a network where the products flow from one Process Unit to another, but rather a parallel processing of Raw Materials into the same sorts of Products. This relation between the Production Unit and a Process Unit is of course an one-to-many relation.

- A Production Unit is a transformation facility which transforms Raw Materials into Products.
A Raw Material can be used in many Production Units; and a Production Unit can use several Raw Materials (many-to-many relation).
A Product can be produced by a number of Production Units and a Production Unit can produce several Products (many-to-many relation).

- A Process Unit uses a certain kind of Process, one Process can however be used in many Process Units, this is a many-to-one relation.

- A product can have a certain kind of product level structure. This is however not a Bill of Materials kind of structure, but a characterisation of Products into levels.
This is a so called Master-Slave relation: One higher level product can have several lower level products (a master has several slaves), and a lower level product has only one higher level product (a slave only has one master).
This is an optional many-to-many relation to the entity type itself.

- The entity type Company has also an optional many-to-many relation with itself. A Company can have subsidiaries, and a Company can have several holding Companies.

- An important thing to mention is the relation between a Product and a Raw Material.
The relation exists because in an integrated site, a Production Unit can use an output of another Production Unit (thus a Product) as a Raw Material to its own operations.
One can consider the entity type Product and the entity type Raw Material as specialisations of an entity type called Chemical Material. However, within there is no (not yet) entity type Chemical Material. This is a historical grown thing, because of the parallel development of the departments such as Sales and Purchase which each pose different demands on the entity type Chemical Material.
This is an one-to-one relation between Product and Raw Material

The entity type Production Unit can be considered the heart of the Object System of this project. It is the most important entity type. From a Production Unit the most important characteristics to monitor are:
- Its capacities in producing all sorts of Products
- The flexibility in using different Raw Materials, however only to be monitored for a cracker.
- The conversion factors which indicate the ratio in transforming Raw Materials to Products.

These characteristics 'belong to' the entity type Production Unit, but will 'move' to combinations of two or more entity types, because of their dependance on both of them.

An extensive explanation of the relations between the entity types will be presented in section 4.3. of document 2 of appendix B in part two.

The entity types and their normalised relations can be put into one figure, see fig.6. on the next page. Normalisation of the relations means that many-to-many relations are divided into two one-to-many relations.

The attributes belonging to the entity types can be found in appendix 4 of this report.

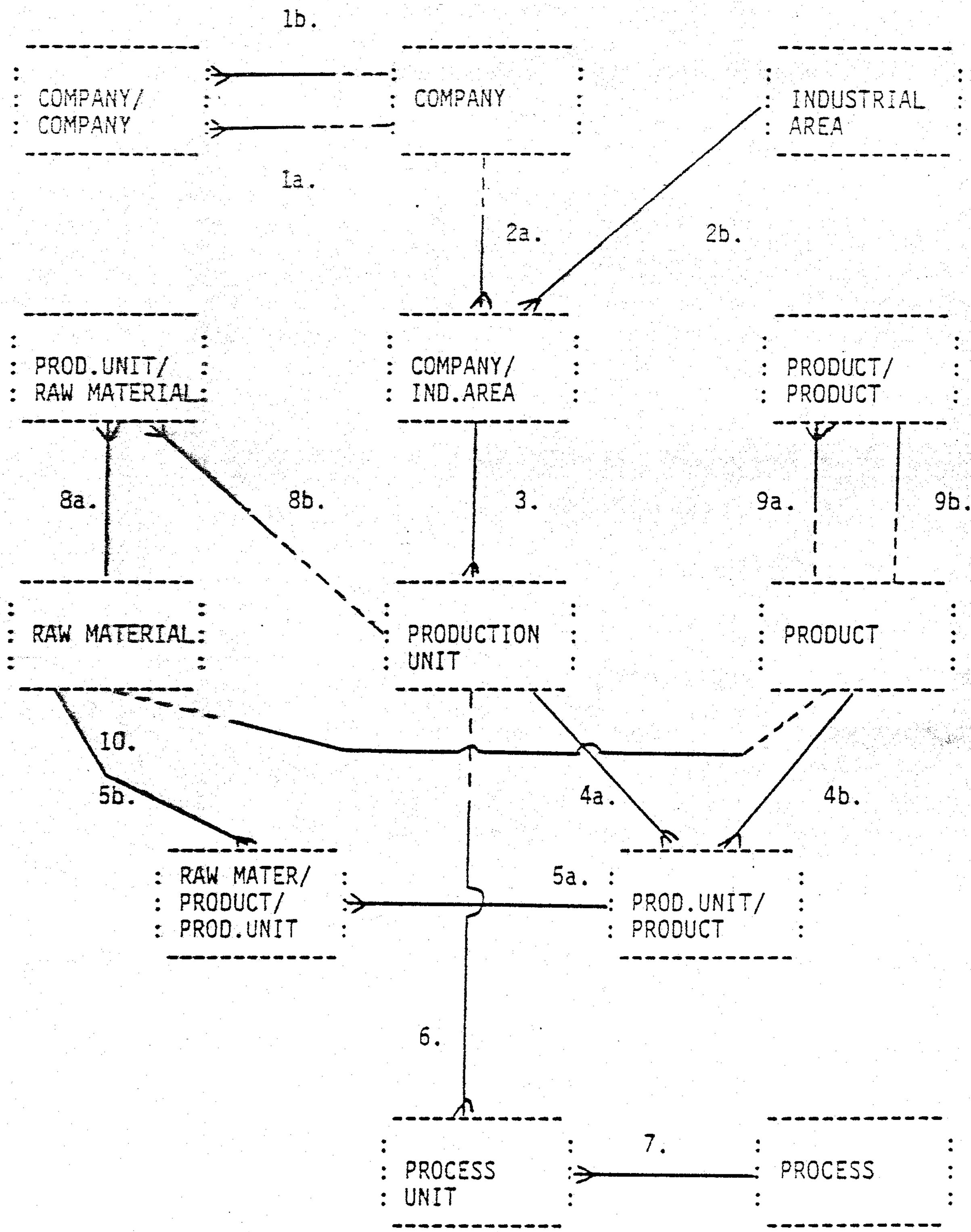


fig.6. Infological Model. Normalised relations between the entity types.

C O N C L U S I O N

In this chapter the infological model was constructed. This was done by analysing the information needs of the users (the systelogical model). Using the Information Modelling technique, first the entity types were identified, then the relations between the entity types and the attribute types belonging to the entity types.
The infological model can be seen as a logical description of the relevant business environment for this system, this has nothing to do with implementation aspects.
The approach followed here is a data orientated one, the objective was to identifying those data elements which the users want to have information about.
The information modelling method of Hoskyns was a good one to use because it leads the analyst and the user in a step-by-step approach to the identification of the entity types coming from the informations needs. The approach followed in this method fits well into the aspect model concept.

9. RAW MATERIAL MODEL

INTRODUCTION

A Decision Support System consists of three parts, one of which is a Model Bank (POLY84).
Out of the information needs of the managers one model came up: the so-called Raw Material Model.

The Raw Material Model is a model which should make it possible to calculate the raw material consumption back to the cracker.

This model makes the integration of a site visible. That is how are the plants interrelated to one another.

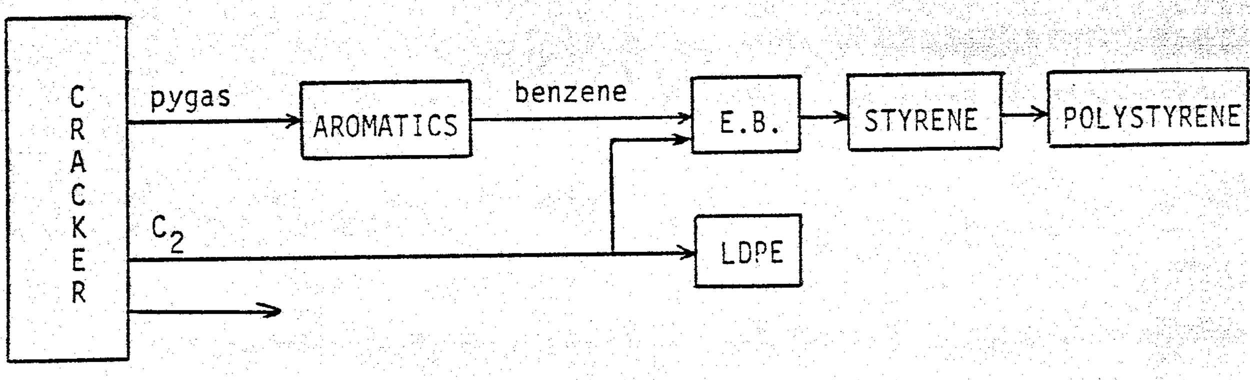


fig.7. Flowchart (part) of an integrated site.

Upstream and downstream is the relative place of a Production Unit to other Production Units. For instance the Production Units left of the Ethyl-Benzene (E.B.) plant are called upstream plants, the plants to the right are the downstream plants.

In order to be able to calculate the raw material consumption of a Production unit it is necessary to know:
- The output of a production unit.
- The conversion factor, which indicates how much raw material is needed in order to make the required output.

The complete model can be found on the following page.
The model has the following parameters:

Raw Material = R_i (Input)
Product = P_j (Output)
Conversion Factor = X^j_{ij} .

Def X_{ij}: The amount of Raw Material i (expressed in some kind of unit) which is needed to produce a unit of Product j.

This conversion factor is the attribute type called Unit Ratio Raw Material Consumption.
The Unit Ratio depends on the Production Unit, the Product which is being produced on the Production Unit, and the Raw Material used. The last determinant of the Unit Ratio is the Year. The Unit Ratio is a technical factor which is not stable over a longer period of time, because of process improvements and debottleneckings (economies of scale), which can cause a better conversion rate.

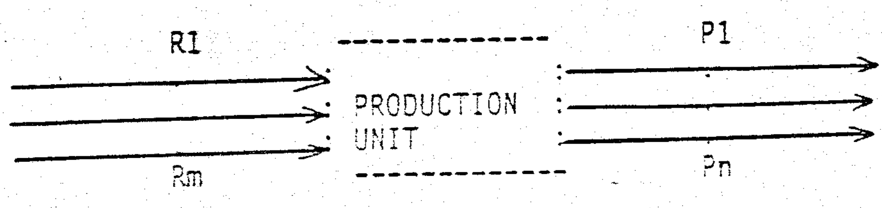
The model presented on the following page accounts for one production unit. But the results of the calculation can be used as an input of following calculations. This because the input of the production unit is the output of the upstream production unit. In other words: the product (output) of the upstream plant is the raw material (input) of the downstream plant.
An integrated site consists of a series of interrelated Production Units (see fig.7.).

Assumptions:
- When calculating the consumption, the quantity of Raw Materials used in a Production Unit is determined by means of the conversion matrix, which consists of the Unit Ratios of that particular Production Unit.
- A Production Unit uses several Raw Materials to produce a number of Products.
- A Production Unit producing more than one Product can be interpreted as follows: One main product and a number of by-products, or the Production Unit can produce two kind of Products, but not on the same time.

Extensive explanations of model and assumption can be found in the Business System Design report (document 2, appendices B, part two).

9.1. RAW MATERIAL MODEL

Model:




```
                         I Products
                         I P1............Pn
                ---------I---------------------
Raw Materials      R1    I X11............X1n     = [X]
                   .     I  .    .    .    .
                   .     I  .      .       .
                   .     I  .    .    .    .
                   Rm    I Xm1............Xmn
```

Analysis:
So there is an input-vector [R] and an output-vector [P] .
The conversion factors are put in a matrix [X] .

The formula is now a matrix calculation:

$$[R] = [X] * [P]$$

One must know: - The output vector [P]
- The conversion matrix [X] .

Model:

$$\begin{bmatrix} R1 \\ \cdot \\ \cdot \\ \cdot \\ Rm \end{bmatrix} = \begin{bmatrix} X_{11} & \cdots & X_{1n} \\ \vdots & & \vdots \\ X_{m1} & \cdots & X_{mn} \end{bmatrix} * \begin{bmatrix} P1 \\ \cdot \\ \cdot \\ \cdot \\ Pn \end{bmatrix}$$

Example:

$$X = \begin{bmatrix} .2 & .4 & .5 & .6 \\ .4 & .4 & .7 & .9 \\ .5 & .5 & .8 & .9 \end{bmatrix} \qquad P = \begin{bmatrix} 10.000 \\ 13.000 \\ 14.000 \\ 20.000 \end{bmatrix}$$

Then: [R]= [X] * [P]
equals:
R1 = 10.000 * .2 + 13.000 * .4 + 14.000 * .5 * 20.000 * .6
= 26.200
R2 = 37.000
R3 = 35.700

C O N C L U S I O N

The raw material model gives the users the possibility to calculate the raw material consumption back to the first production unit (a cracker) of an integrated production location
The integration of a site lies at the basis of this model.
The parameters used in the raw material model are attribute types of the infological model. The attributes types which are needed and the relations which are needed are present in the infological model. Therefore the raw material model can be calculated, from a logical viewpoint. A programming tool, that is, a Database Management System and a query/programming language should make the implementation of the model possible.
Whether the tool FOCUS is suitable to implement this model is still to be found out.

10. D A T A M O D E L

I N T R O D U C T I O N

In the logical analysis no implementation aspects were taken into consideration, the infological model is a logical description of the 'problem area'.
In this chapter the first steps towards implementation shall be taken, starting with defining the entity and attribute types and their relations in a Data Dictionary.

10.1. D A T A D I C T I O N A R Y

Information can be considered as a corporate asset, because information is one of the critical resources to a company.
Therefore a company should plan information, just as it plans capital. A company, such as , should have its information organised into a system where all the entity types and attribute types and the relations between them are defined. Such a system is often called a Data Dictionary. The Data Dictionary is controlled by the Data Administrator. The objective of such a Data Dictionary is to establish a logical description of all relevant entity and attribute types within a company, their relations, the usage of those items and the responsibility.

In developing an information system there should be a close cooperation between the analyst, the users and the Data Administrator, because the data administrator can help define the entity and attribute types. After the entity and attribute types and their relations were identified in the infological model, these were defined in close cooperation with the Data Administrator of .
In document 2 of appendix B of part two these definitions can be found.

The another objective of the Data Dictionary is to have a company wide agreement of how the information elements should be coded, that is, what type they are (alfa-numeric, integer, decimal, Boolean) and what length the data field should have.

10.2. N O R M A L I S A T I O N

After the logical part of the system is well identified and defined, the steps towards the physical information system can be made.

The first step is the normalisation of the attribute types within an entity types.
This phase is part of the Hoskyns Information Modelling method, and is based on the normalisation methodology of Dr. E.F. Codd.

Source data for normalisation can take a number of forms:

- Manual records or documents
- Computer file layout
- Computer layouts
- etc.

In this case the sources for Normalisation will be the infological model which was put up in the previous pages.

Each entity type will be considered as one source.
These entity types could be exactly the type of structure the user might wish to see in the form of a report, but it does present certain problems if one wants to record the data as a file in the same format. These problems can be expressed as storage anomalies:

- If it is impossible to store a fact until another fact is known, then we have an INSERTION anomaly.
- If a fact is deleted, and this causes the deletion of other still relevant facts, than one has a DELETION anomaly.
- If a fact changes, and multiple modifications are necessary, then there is a MODIFICATION anomaly.

Normalisation is a stepwise method which defines storage structures which remove such anomalies.

In the infological model, normalisation took place on the level of the entity types (see section 8.2. of this report), in this phase the attribute types within an entity type need to be normalised.

The Normalisation methodology consists of four phases:

1. List the source in Un-Normalised Form (UNF).
2. Remove repeating groups -- First Normal Form (FNF).
3. Remove partial dependencies -- Second Normal Form (SNF).
4. Remove mutual dependencies -- Third Normal Form (TNF).

Step 1: The purpose of step 1 is to recognise the inherent structure (that is the relations between the attribute types within a entity type) present in the source (NB. entity type), and to list it in a form which can be normalised. This listing is called the UNF(-relation).

Step 2: The second step (removing repeating groups) splits the UNF-relation into two relations, creating a separate relation from the repeating data. However, if one divides the UNF into two parts one loses the relationship which exists between the two parts. Therefore, to preserve the relation, the UNF-key must be carried down to the repeating data relation and a compound key needs to be created.

Step 3: Partial dependencies imply data which is dependent on only a part of a key, so in this step only the compound keys will be studied. The following question should be asked: "Does each attribute depend on the whole key, or only part of the key". Those partial dependent items will be removed with their identifying key, to a separate relation.

Step 4: To convert to TNF, one needs to examine the SNF-relations to determine which, if any, mutual dependencies exist between pairs of attributes which are not the key(s) of the SNF-relations.

In document 2 of appendix B of part two results of the normalisation process are presented.

In the normalisation phase, different sources (entity types) are normalised without looking at the coherences which exist between the entity types. So the following logical step would be to translate all the Third Normal Form entity types into a coherent set. This is called the process of Optimisation.

Optimisation can therefore formally be defined as "the process of combining Third Normal Forms relations derived from different sources, when those relations have identical keys".
It is obvious that at this stage we must generalise and define "identical" as:

i) Spelled the same.
ii) Means the same.

In analysing the Third Normal Forms of the normalisation process, one finds only two "identical" entities:
Country of Company (country code, country name), and
Country of Industrial Area (country of ind.area code, name).

These two can be combined into one entity type Country.
This 'new' entity type has an one-to-many relation to the entity type Company and an one-to-many relation to the entity type Industrial Area.

Hoskyns defines a formal test strategy in its Data Analysis method. The data model is to be tested whether it suits the information needs of the users. This test strategy is called the Access Path Analysis. This technique takes the information needs (queries) of the user as a start. These queries were analysed and broken into parts. The data model was checked as to whether the query could be answered, that is, whether the entry points in the database were correct, whether the relations, the entity types and the attribute types were put in the right spot.
These tests and their results can be found in the Business System Design report (document 2, appendices B, part two)

The raw material model in chapter 9 can also be considered as a test of the data model.
These test strategies are on the logical level, no physical database structure is considered, neither is a query language of any sort.

10.3. CONVERSION TO TECHNICAL MODEL

After the data model was put up in the previous section, the technical model and the data storage structures could be developed.
In this section the principles, which have been followed in constructing the physical storage structures, are explained.

The technical model shall be implemented with the fourth generation tool FOCUS. FOCUS is a software package which employs its own hierarchical database.
For insertion of data into the database FOCUS has a programming language, which must be used in the update programs.
To obtain the data from the database, FOCUS has a query language, which is very straightforward and userfriendly. Users don't need to know how the database is physically implemented.

When implementing a database with FOCUS, one needs to construct a so-called master file, were the data elements are defined.

A FOCUS file consists of segments which have a specified relation with each other, this can be an one-to-many (multiple segments) or an one-to-one (unique segment) relation.
Elements in a segment are called (data)fields, these represent the attribute type identified in the Business System Design report.
(see also section 11 of this report).

The basic principle followed in constructing the technical model was to construct, as far as possible, a separate file for each entity type. The argument for this is that when the the database contains a lot of data, more and more time is needed to open data files, because all the records of the file are loaded into the core memory.
Further, separate files provide greater flexibility when new information requirements are included into the system.

FOCUS has a hierarchical database, and since the data model is relational, it can be expected that FOCUS will not be able to construct and maintain all the relations.
FOCUS has however some relational facilities, like JOIN, but joining is only a reporting facility. There is no possibility to open two files in one update procedure.

So some trade-offs can be expected because of these limitations. Since FOCUS employs a hierarchical database, it has some facilities for these kind of structures. Wherever possible these hierarchical facilities were used.

Arguments used to connect entities into a hierarchical file can be found in document 2, chapter IV in appendix B in part two.

CONCLUSION

The infological model was normalised into a datalogical model, using the normalisation method of Codd.

The next step is the construction of the technical model. The following principle was followed: to construct a separate file for each entity type, except where hierarchical relations are present in the data model. This kind of structure was chosen with the implementation tool in mind. The tool is FOCUS, which employs its own hierarchical database.
Since the data model is relational, some trade-offs can be expected in using a hierarchical database such as FOCUS.

11. TECHNICAL MODEL

The basic principle in file construction was to build a separate file for each entity type, except for the obvious hierarchical relations.

The following files were constructed:

File 1: Company
File 2: Company/Company
File 3: Industrial Area
File 4: Company/Industrial Area
Production Unit
Process Unit
File 6: Product
File 7: Production Unit/Product
Raw Material/Product/Production Unit
File 8: Raw Material
File 9: Production Unit/Raw Material
File 12: Process
File 13: Product/Product
File 14: Country

In file 4 and file 7 several entities were put in a hierarchical relation.
Explanations of this file construction can be found in document 2 in appendix B in part two.

There is a problem in regard to the coding of the occurrences of the entity types.
It may not be possible to have lists of existing codes from for the entities which have been constructed for this system.

So the codes which aren't available will have to be constructed.

When implementing a database with FOCUS, one needs to construct a so-called master file where all the data elements are defined, in combination with their type and length, also relations between the data fields can be established. The master file is the FOCUS dictionary. FOCUS needs this master file whenever data files are to be accessed. Without the master file programs (queries or update programs) won't work.

The basic element of a FOCUS file is a (data) field. A group of data fields which have an one-to-one relation to each other can be grouped into a segment. A FOCUS file consists of segments which have a specified relation with each other, this can be an one-to-many or an one-to-one (unique segment) relation.

When 'opening' a file, for update or a query, all the records of the top-segment are loaded into the core-memory. The total length of a record is a summation of the lengths of all the data fields.

These records are loaded into 'pages'. The efficiency of FOCUS improves when all the occurrences of the top-segment can be loaded into one page. Access and search times are efficient this way.
While constructing a file the objective should be to construct a relative small top-segment, with only short data elements, for instance only keys, codes and short attributes. The rest of the attribute types (fields) should be in a separate segment. This segment is a unique segment, this means that it has a one-to-one relation with the top segment. By means of this unique relation it isn't necessary to have a key-field in the unique segment. The unique segment can be considered as an appendix of its top segment.

C O N C L U S I O N

The database was constructed with the 4th generation tool FOCUS, which has its own hierarchical database.
A series of files compose the database. Every entity type is put in a separate file. Some entities were put together in a hierarchical file. The rest of the files have no fixed relation which each other.
Files can be linked together by JOINing them by means of an attribute type occurring in both files.

12. T H E D S S S Y S T E M

I N T R O D U C T I O N

A decision support system consist of three sub-systems (see 2.5):
- A Data Bank
- A Model Bank ·
- An User Interface

This chapter deals with the constructing of these three sub-systems.

12.1. T H E D A T A B A N K

The basis of this database was the infological model of the problem situation. This infological model was normalised into the Data Model. After the Data Model was constructed, technical aspects were considered in a Technical model. Out of the Technical Model the physical storage structure, the database could be constructed. The database was constructed with FOCUS.

The construction of the database was discussed in the previous chapters.

In the Computer System Design report the exact file construction can be found (see document 3, appendix B, part two).

12.2. T H E M O D E L B A N K

The model bank is that part of a Decision Support System which contains different models which support the decision making process of the user(s).

The Plant Profiles Decision Support Systems has one concrete model; the Raw Material model (see chapter 9).
This model is not yet implemented, because the emphasis at this moment lies on the query aspects of the system. Logical it should be possible to calculate the model out of the database. From a practical point of view it might be quite complicated to implement this model with FOCUS, this because the programming language of FOCUS is not powerful enough. As stated before, the basic strength of FOCUS lies in the reporting and query facilities.
The first step, in calculating the model, is to calculate the raw material consumption of the most downstream plant of the site. The results of this calculation can be used as input parameters of the following query, in combination with the Unit Ratios of the upstream plant(s).
The problem with FOCUS is that procedures can't be nested, so results of the first query have to be stored in a temporary file which serves as a input to the next calculation.

The users may want to have also other kinds of models. For example:

All these models are based on an extraction of data out of the database, after which a calculation can take place. This kind of data manipulation can take place on a Personal Computer using the tool PC/FOCUS.

12.3. USER INTERFACE

The most important part of a Decision Support System is the User Interface, because this is the only part of the DSS the user sees. The acceptance of the system depends on the userfriendliness of the User Interface.

Userfriendly means: the user only has to specify WHAT he wants, HOW it's done is not important to him.
The user has no interest how the programs are built, how the physical data is stored, how the files are opened or how the data is searched. The users need to understand how the infological model looks like. If he understand this model he should be able to work with the system.

The User Interface of the plant profiles database consist of several parts.
The first part is the User Interface which is needed to get the data into the files.
The second part are predefined queries the user can access.

The user should be provided with a means to get the data into the database.
These should be userfriendly update programs, preferably in the form of menu's.

12.3.1. UPDATE INTERFACE

FOCUS provides facilities to load the data into the database, it has a programming facility for this called MODIFY.
FOCUS is designed as a query and reporting tool, the programming language is not sufficiently equipped for programming, for instance nesting of procedures is not possible, also there's a limit to the number of screens which can be programmed in one procedure. The MODIFY facility is therefore a limited facility, it's is not as flexible and powerful as a third generation language such as COBOL, ALGOL or PASCAL.

A positive point of the update facility is, that the programming language is non-procedural. This means that the emphasis of this language is on the WHAT of the programming, not on HOW its done.

The update interface appears to the user as a series of menu's which are tied together. The user has the capability to choose his own way through the update program.

The update system has three layers:

1. Sign-on layer
   In this level the user sees the logo of the system which he is going to work with.
   The identity of the user is checked and stored. The identity is already checked from the user's signing on to the computer, so the user is not obliged to type in another user identification or password.
   The user's identity is stored and will be logged on every update transaction the user makes. This identity check consists of the user's name and his location.

2. Main menu layer
   When the sign-on is completed the user gets the main menu, he can choose 14 different functions, each to update one particular part of the database.

3. Update functions layer
   In this layer the user sees the sections of the data he can alter.
   (This layer will be explained in the following text).
   When the user is finished he exits the program and returns to the main menu.

These update facilities can also be used for inquiries. This has however several disadvantages:

- The user sees a screen, not a report.
- The user can't change the contents of the screen.
- The function, the user has chosen, only allows him to change one file, he/she can't relate files together. So the user will only see the contents of a (part) of a file.
- If the user indicates that the transaction is completed his name will be logged to that particular data set.

These update system layers can schematic be presented by fig.8.

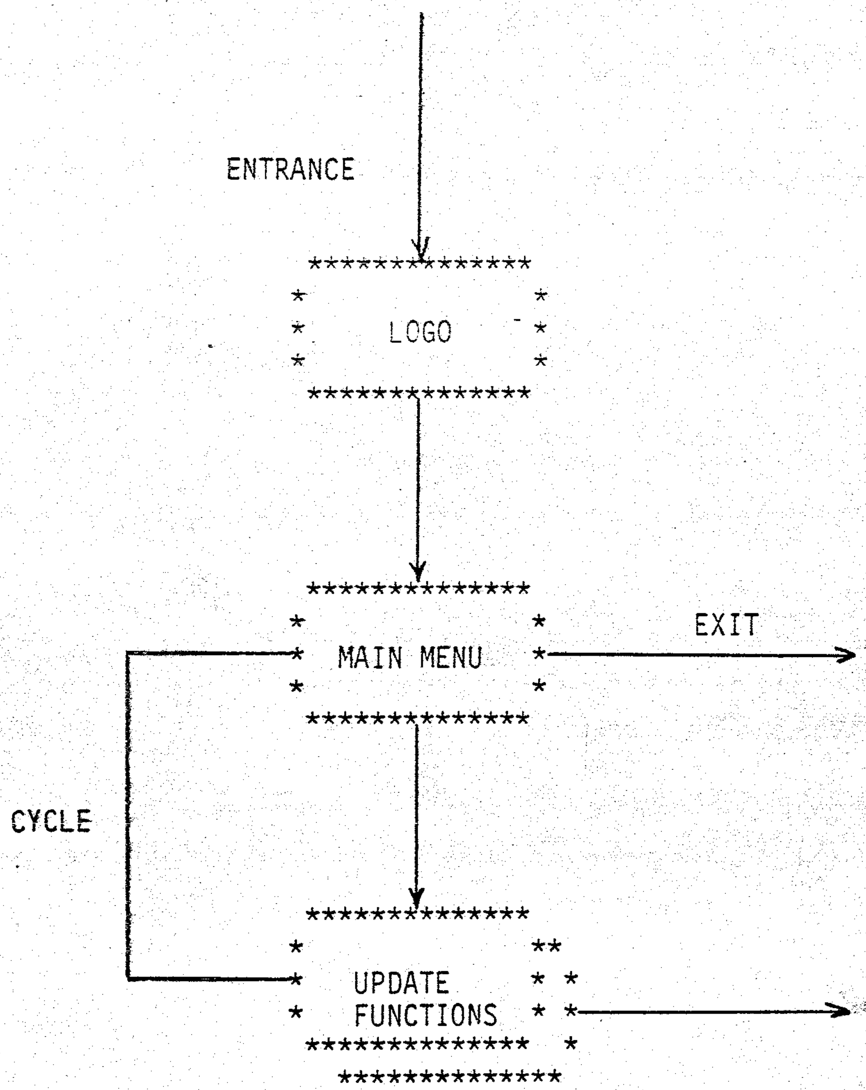


fig.8. Plant Profile System layers

Update functions are also constructed in layers.
The user chooses in the main menu a function (option). Depending on the function, he/she sees different screens.
The screens look all the same, same function keys and screen layouts are used throughout the function modules.

When the user has chosen a function he is asked to provide values to the key fields, in order to let the system know what part of the database (record) he wants to change.
Second the user should indicate whether he wants to:

- create a new record (occurrence),
- to update an old one,
- to delete a record.

Summarising, first the user has to select a part of the database he wants to add or change, next he chooses the action he wants to be performed on the data.
The two questions are asked in one screen. (Layout of the screens is presented in appendix 5).

The next screen, the user sees, depends on the action he/she wants to be performed on the record.
For instance, the user sees an 'empty' screen when he/she wants to create a new record. The user gets the actual data which is stored in the record when he/she wants to update the data.
When the user selects the wrong values the system will tell him/her so and provide him/her with a list of possible values the item can have. This is important when the user should type in a code, and he/she is unfamiliar with them.

When the user has typed in the new data and he selects the save option, the new or updated record is stored and the old one is overwritten. The user will now return to the first screen of the program module, where he/she can choose another record to alter or to add, or he/she can decide to go back to the main menu.

When an occurrence has some text lines, these can be found in a separate page, function keys enable the user to page forward and backwards.

The screen sequence in a function module look like figure 9.
How the programs are structured can be found in the Computer Systems Design report (See document 3 in appendix B of part 2).

All the screens the user sees can be found in appendix 5.

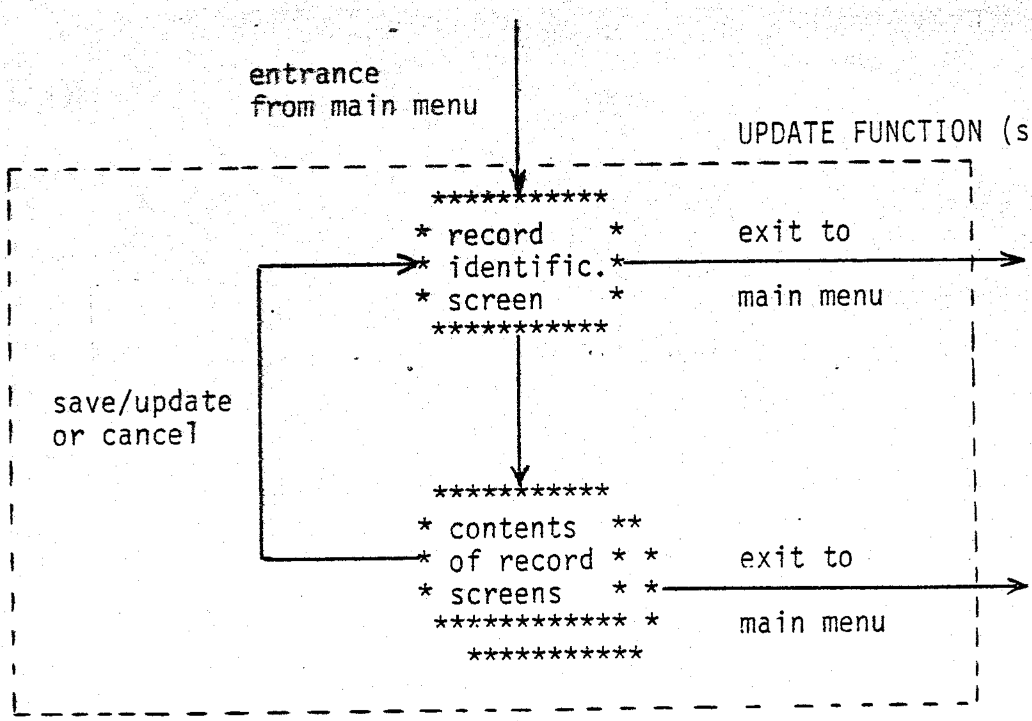


fig.9. Function module structure, User Interface.

4.3.2. QUERY INTERFACE

The update interface is constructed to give the user a tool which enables him to get data in the database.

Information needs (questions, queries) of users are, however, not stable in time, it makes therefore no sense to construct an extensive user interface to do queries.
Some standardised queries, such as questions about capacities of the production units of the competitors can be presented in predefined reports which can be presented either on a screen or on a paper. The advantage of having it on a screen is that it represents the latest data available, the advantage of a paper is that it is on hand, but it can be out-of-date, which can pose a great danger, because the data can be wrong.

For the non-standard questions, an intermediate person is needed who rebuilds the questions of the users into queries the computer understands. It's not expected that the consumers of the information which comes out of the system will also have their hands on the system.

So the second part of the user interface there are some predefined queries. These queries shall mainly be constructed for querying of production capacities. These standard reports shall be developed with another tool, namely TIF (the Information Facility), this is an IBM-product. This tool makes it possible to construct a hierarchy of standardised, predefined screens which the user can access. The user is able to move around the hierarchy, for instance choosing a certain company or a summarisation of the production capacities of a certain country or product.

FOCUS has a facility which enables the unexperienced user to program his own queries. This facility is window-driven and ask the user what file he wants to query, next which data fields he wants information about, etc. This facility can access only predefined files, when files should be joined together this has to be done in advance.

12.4. SECURITY ASPECTS

One of the user requirements is that the database, should be secured.

The system manager, who has the responsibility over the system, must be able to shield certain parts for some users or allow users to have read access only (maybe restricted to certain parts of the database).

The security package of TOP SECRET has not been installed on the mainframe the system will run on. A separate security subsystem will be developed.

FOCUS gives several facilities to ensure the security of the data base. These security facilities are coupled with the dictionary of FOCUS, that is with the master files.
After speaking with the FOCUS experts in it was decided to have the security for this application arranged with the facilities in FOCUS.
FOCUS provides several facilities:
- ability to shield the data from certain users;
- facility to encode the programs and the master descriptions of the files. This means that the programs are not in a readable form for

the users, a password is required to read the programs and the master descriptions.

With the major user the following agreement was made.
Only the system manager should have the ability to create occurrences of certain (relatively stable) items, such as:

- Countries
- Companies
- Industrial Areas
- Company/Company relations
- Company/Industrial Area combinations
- Products
- Raw Materials
- Processes

So only the dynamic items and the dynamic relations (such as the Production Unit/Product combinations) can be changed by the 'common' user.

Another user requirement is to know who did the last update transaction and when he/she did it.
After analysing the alternatives it was decided that the logging had to take place at the entity level. Consequently the updates aren't visible per data item. This has a consequence that when a user is in an 'updating mode' he should check all the items and verify them; his name will be connected with all the items involved in that screen.
The name of the user and the day the transaction took place will be taken out of the logging procedure.

12.5. M U L T I U S E R S Y S T E M

This system is a multi user system. This means that people all over the company will work with the system.
This poses some integrity problems. For instance, when two users want to update the same record at the same time. The user who did the last transaction on the record must see first the results of the transaction of the user who did the first update. Otherwise the second user doesn't know what the first user did.

Facilities need to be constructed in order to preserve the integrity.
FOCUS has a multi user facility; called Simultaneous User mode.
When a user logs on to the mainframe he/she is given access to the central database, where the database and programs are stored.

C O N C L U S I O N

All the ingredients of a Decision Support System are present: A Data Bank, a Model Bank and an User Interface.
The main emphasis of the system is however on the inquiry aspects. The model bank is, at least at the moment, less important.
The system was constructed with FOCUS, using its database facilities and its programming language. The user interface consists of several parts. Part one are update programs which enable the user to load data into the database.

Part two are some predefined reports the user can access.
The model bank is not yet implemented. Propositions of how some models can be implemented are given in section 12.3.

C O N C L U S I O N S

In the development of the Plant Profiles Project the most important factor was the evolutionary character of the development process. The iterative nature of the prototyping approach fitted perfectly in this. Two prototypes were constructed, the second prototype can be considered as the final system. This system will however be refined during the implementation phase.

The first prototype was constructed after the information needs of the target group managers were assessed by using the Critical Success Factors method. The method focuses on the identification of those elements which are critical to success to those managers. The target group consist mainly of production managers, whose top priorities were to run their production unit. They were only partly involved in strategic decision making.
Using the CSF's method in this project can therefore not be considered as a success.
After information needs were assessed a prototype was constructed. The first prototype served as a proof that it was possible to build a database system. The prototype gave also the opportunity to use it as a vehicle to check whether the information needs were fully understood, also it served as a tool to get support for the project in the European headquarter of . Decided was to expand the scope of the project to European level.

The second prototype iteration started with a renewed analysis of the information needs of the managers using the Information Modelling Method of Hoskyns. Also experience gained from the first prototype were used. After the analysis phase all the sub-systems of a Decision Support System could be constructed, starting with the database.
User interfaces were constructed to give the users the possibility to load data in the database and to extract data from it. The model bank is not yet implemented, because the priority on this moment lies on the query aspects of the system.

The system is now in the phase that data can be loaded into the database, using the User Interface.
Gathering and insertion of the data in the system will prove to be the major effort, because it consists of a tremendous amount of data, which is located all over the company.
The insertion of the data in the database will serve also as a test as to whether the database and the User Interface are well constructed.

Due to the extensiveness of data only a subset of the total data is initially loaded into the system, starting with the site of and the LDPE business in Western Europe.

After the data has been loaded into the database the system will have to prove itself. The system must be marketed to the users.

L I T E R A T U R E

(ALLO82) Robert M. Alloway,
User managers' system needs,
CISR WP n° 86, Sloan WP n° 1301 82, 1982.

(BEMEL83) Theo M.A. Bemelmans,
Bestuurlijke Informatie Systemen en automatisering,
Stenfert Kroese, Leiden, 1983.

(GINZ82) Michael Ginzberg, et.al.
Decision Support Systems,
North Holland, Amsterdam, 1982.

(KLEIN85) H.K. Klein, et.al.
Fundamental issues of DSS: a consequential perspective,
North Holland, Amsterdam, 1982.

(McCOSH82) A.M. McCosh, et.al.
Developing Managerial Information Systems,
McMillan Press Ltd London, 1981, page 353-379.

(MINZ75) H. Mintzberg,
Impediments to the use of Management Information,
Brochure: National Ass. of Accountants, 1975.
(in syllabus BIVII, THE).

(POLY84) Polyautomatiserings zakboekje,
PBNA, Arnhem, 1984.

(PORTER80) Michael Porter,
Competitive strategy,
Free Press, New York, 1980.

(ROCK79) John F. Rockart,
Chief executives define their own data needs,
Harvard Business Review, March/April 1979.

(ROCK81) John F. Rockart & Christine V. Bullen,
A primer on Critical Success Factors,
CISR n° 69, Sloan WP n° 1220-81, 1981.

APPENDIX 2. INTERVIEW QUESTIONS

Preliminary

In this first interview session, shall be tried to investigate the kind of decisions the managers make. And what kind of information is important to do that.
After this first session a prototype shall be build, and presented to the user to get their comment on it.
This prototype shall be further developed to suit the new requirement of the managers.

QUESTIONS

1. Can you indicate what your function is and on what area your responsibilities lie ?

2. What are your major goals you want to obtain ?

Critical Success Factors (CSF's) are those factors or areas, where results, when positive can influence the performance of the total organisation.
These are the key areas where "thing must go right", to have an optimal performance.

3. What are the CSF's in your area ?

4. Which information you need to control these CSF's ?

5. Give me important information you don't get on this moment, or the quality of the information is not good ? And can you indicate why you need this information ?

6. What kind of decisions (strategic, tactical) you make, and who can computer-based system assist you with it ?

7. What is the time span of your decisions ?

8. Do you participate in productmarketing teams or business marketing teams ? What is your role in it ?

9. Do you find you have sufficient information for these meetings ? If not, why ?

10. What is the value of having better information ? (opportunities, money, manpower etc.)

11. How can the necessary information be gathered ?

12. Which internal or external databases can be consulted ?

13. Which data elements you find important to have information about in order to get your information needs covered ?

14. In which form should the information be presented to you ? (matrices, graphics, pie-charts etc.)

==============

Appendix 5 : SCREEN LAYOUTS

LOGO:

```
   PPPPPP   LL          A      NN        NN  TTTTTTTT
   PP   PP  LL         AAA     NNN       NN     TT
   PP   PP  LL        AA AA    NNNN      NN     TT
   PPPPPP   LL       AA   AA   NN   NN   NN     TT
   PPPPP    LL       AAAAAAA   NN    NN  NN     TT
   PP       LL       AAAAAAA   NN     NNNN      TT
   PP       LL       AA   AA   NN      NNN      TT
   PP       LLLLLL   AA   AA   NN       NN      TT

PPPPPP   RRRRRR    OOOOO    FFFFFF  II  LL      EEEEEE   SSSS
PP   PP  RR   RR  OO   OO   FF      II  LL      EE      SS   S
PP   PP  RR   RR  OO   OO   FF      II  LL      EE      SS
PPPPPP   RRRRRR   OO   OO   FFFFFF  II  LL      EEEEEE   SS
PPPPP    RRRRR    OO   OO   FF      II  LL      EE        SS
PP       RR  RR   OO   OO   FF      II  LL      EE         SS
PP       RR   RR  OO   OO   FF      II  LL      EE      S  SS
PP       RR   RR   OOOOO    FF      II  LLLLLL  EEEEEE   SSSS
```

MAIN MENU:

```
**************************************************************
THIS PROGRAM ENABLES YOU TO UPDATE THE PLANT PROFILE DATABASE
**************************************************************

>
M.HEIJNEN
>
 CHOOSE AN OBJECT

1. COMPANY                         8. RAW MATERIAL
2. COMPANY/COMPANY                 9. PROD.UNIT/RAW MATERIAL
3. INDUSTRIAL AREA                10. RAW MATERIAL/PRODUCT/PROD.UNIT
4. COMPANY/IND.AREA               11. PROCESS UNIT
5. PRODUCTION UNIT                12. PROCESS
6. PRODUCT                        13. PRODUCT/PRODUCT
7. PROD.UNIT/PRODUCT              14. COUNTRY
                  99. END MAINTENANCE SESSION

PLEASE SUPPLY VALUES REQUESTED

OBJECT=
```

PROGRAM SCREEN 1:

```
PLANT PROFILES MAINTENANCE: --COMPANY--  ACTION=
-----------------------------------------------------

COMPANY CODE   :  1

ACTION         :  2    (1=CREATE; 2=MODIFY; 3=DELETE)



CONTINUE: ENTER; NEXT LINE: RETURN; CANCEL: PF2; QUIT: PF3;
```

```
PLANT PROFILES MAINTENANCE: --COMPANY--  ACTION= 2
-----------------------------------------------------

COMPANY CODE       :      1

COMPANY NAME ABBR. :
COMPANY FULL NAME  :
COUNTRY CODE       :    2
COUNTRY NAME       :  United States of America
JOINT VENTURE      :  n    (Y=YES; N=NO)


LAST USER =  M.HEIJNEN


NEXT LINE: TAB; SAVE: ENTER; CANCEL: PF2; QUIT: PF3; TEXTPAGE: PF8;
```