

## **How to succeed with Modern Tools such as the CASE tool "Deft" on the Macintosh.**

**Christina Lagerkvist**

**AB Programator, Box 20072, 161 02 Bromma, SWEDEN**

**Åke Nyberg**

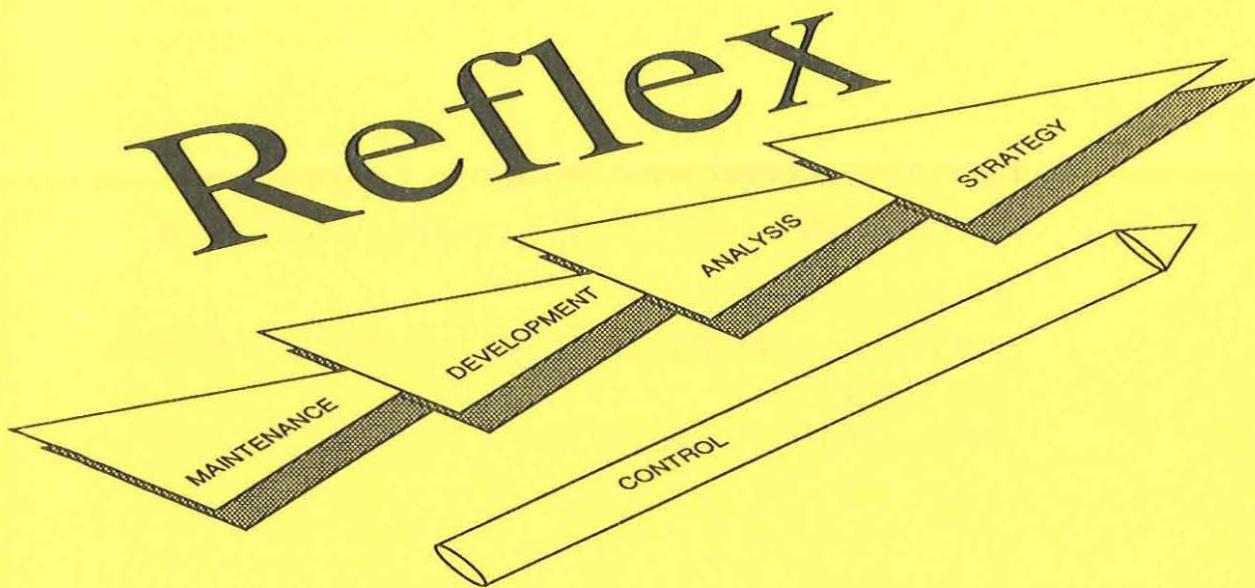
**Infotool Data AB, Box 101, 182 12 Danderyd, SWEDEN**

REFLEX is a working method for developing information systems starting with information strategy continuing with business analysis and concluded by systems development and maintenance. The method is well proven and used by Programator and its customers, including many of the largest corporations in Scandinavia and northern Europe, for more than twenty years. The method has been continuously enhanced to embrace the use of modern tools for documentation and prototyping.

DEFT is a CASE tool for the Macintosh developed by DEFT INC., Canada, marketed in northern Europe by Infotool, Stockholm. DEFT supports the REFLEX method. DEFT is easy to use and provides the analyst with documentation and the results from the analysis can be transferred into the prototyping stage automatically.

The talk includes a survey of the REFLEX method as used for analysis and development supported by DEFT. Experiences from a development project at The Swedish Tourist Board are described. Results from this project include a better user participation, a more creative way of systems development, less efforts in manpower and reduced development time and of course a much better documentation.





## **THE OBJECTIVE IS INCREASED PROFITABILITY.**

In today's business world, well functioning information systems are more important than ever. Information is necessary to rationalize routine tasks, to improve logistics, to minimize inventory costs, to bring responsibility and authority closer to the clients. Information is necessary for the development of the business, to follow up on marketing strategies, for the development of new services and products, etc.

Today, information is strategically one of the most important competitive resources. The problem lies in providing information at the right time and at the right price.

Therefore, we always must weigh benefits against costs whether we are working with specifying priorities, development, usage or maintenance of computer systems or computer production. Focusing on profitability is a main feature in the cooperation with management, end users and computer professionals.

## **The road to profitability is a structured working method supported by effective tools — REFLEX with DEFT**

Programator has more than twenty years of experience as computer consultants. A working method has been developed and refined aiming towards profitable information processing, while ensuring high quality in each link of the development chain.

Quality means selecting the right function, the right technology and the right usage at the right time and at the right price. The method which have been tested and developed to fulfill these requirements is called REFLEX. REFLEX is short for REsults orientation and FLEXibility.

REFLEX can be used to describe Programator's basic working philosophy. It consists of four blocks: Information Strategy, Business Analysis, Development and Maintenance. REFLEX also has a model for Project and Quality Control, used in all four blocks.

Each block has a general objective, decision points, working process, techniques and help facilities. The client will use only those parts which will fit his actual requirements. He may utilize his own working techniques if they are available and function well. A project or an assignment may utilize the whole REFLEX concept, or only the working techniques within one separate part.

Deft is a CASE-tool that supports Reflex. The tool is easy to use and gives documentation in most of the steps in the method-chain. Deft gives higher quality in the system development process. The enduser can be active in the work and ascertain that they get the system they want to have.



# Reflex with Modern Tools

Today, system development is more often than before carried out in environments where modern tools are available. Fourth Generation Tools (4GT's) are included in the realm of modern tools, sometimes supplemented with CASE tools as computer support for analysis and development.

New tools in themselves do not guarantee success. One can not rely only on the tools and disregard all methodological working procedures. Therefore REFLEX with modern tools presents an ideal solution — a user adapted method complete with all the new possibilities. The end user's role is enhanced and the computer itself is utilized more during the system development process.

Systems analysis and development can be carried out more gradually and iteratively in REFLEX than traditionally possible. In order to test technical or user solutions, prototypes are constructed to support decisions regarding system design. Simplification is attained, since certain steps can be combined. In addition, the tool itself performs many operations that used to be done by hand.

REFLEX with modern tools has as its motto the following for system development:

- Quick and effective
- Provides easy computer system maintenance
- Correct functional and technical quality

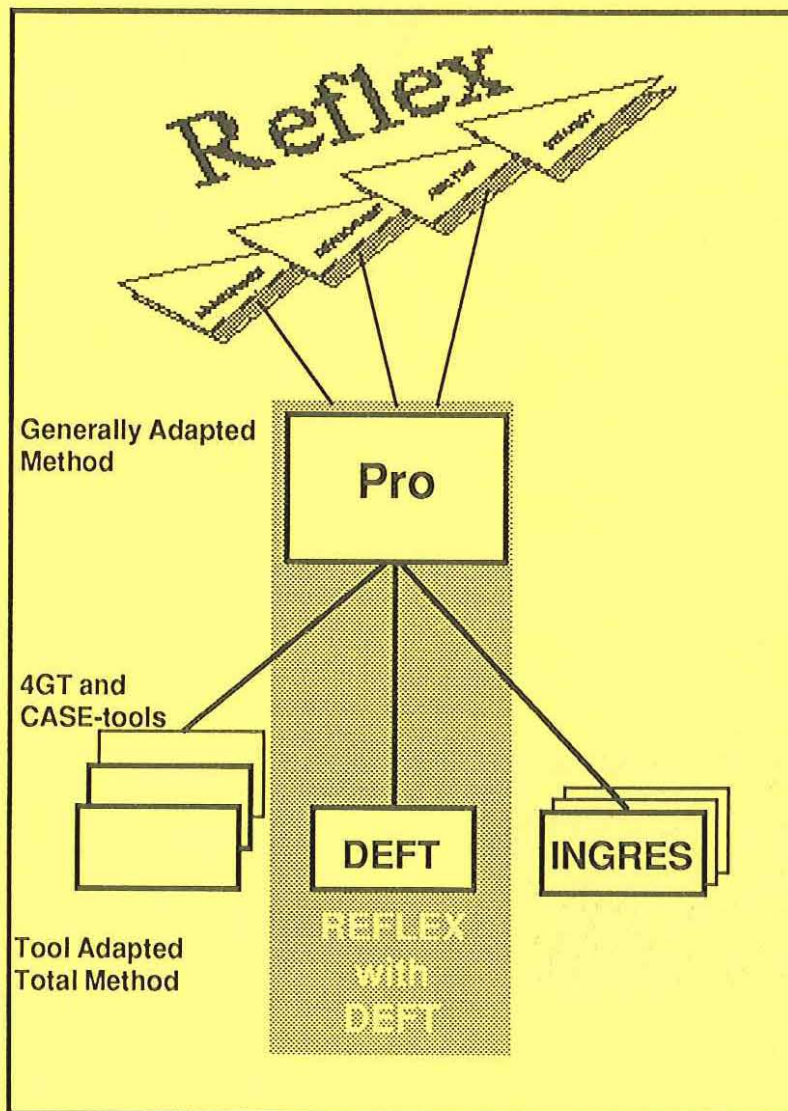
In order to reach the objectives formulated in the motto, a well thought out methodology is required as well as correct usage of tools. The method can be explained as follows:

## 1. Prototyping

This segment consists of those adaptations that have been made in REFLEX Analysis, Development and Maintenance and applied to system development through the use of modern tools and active end user involvement.

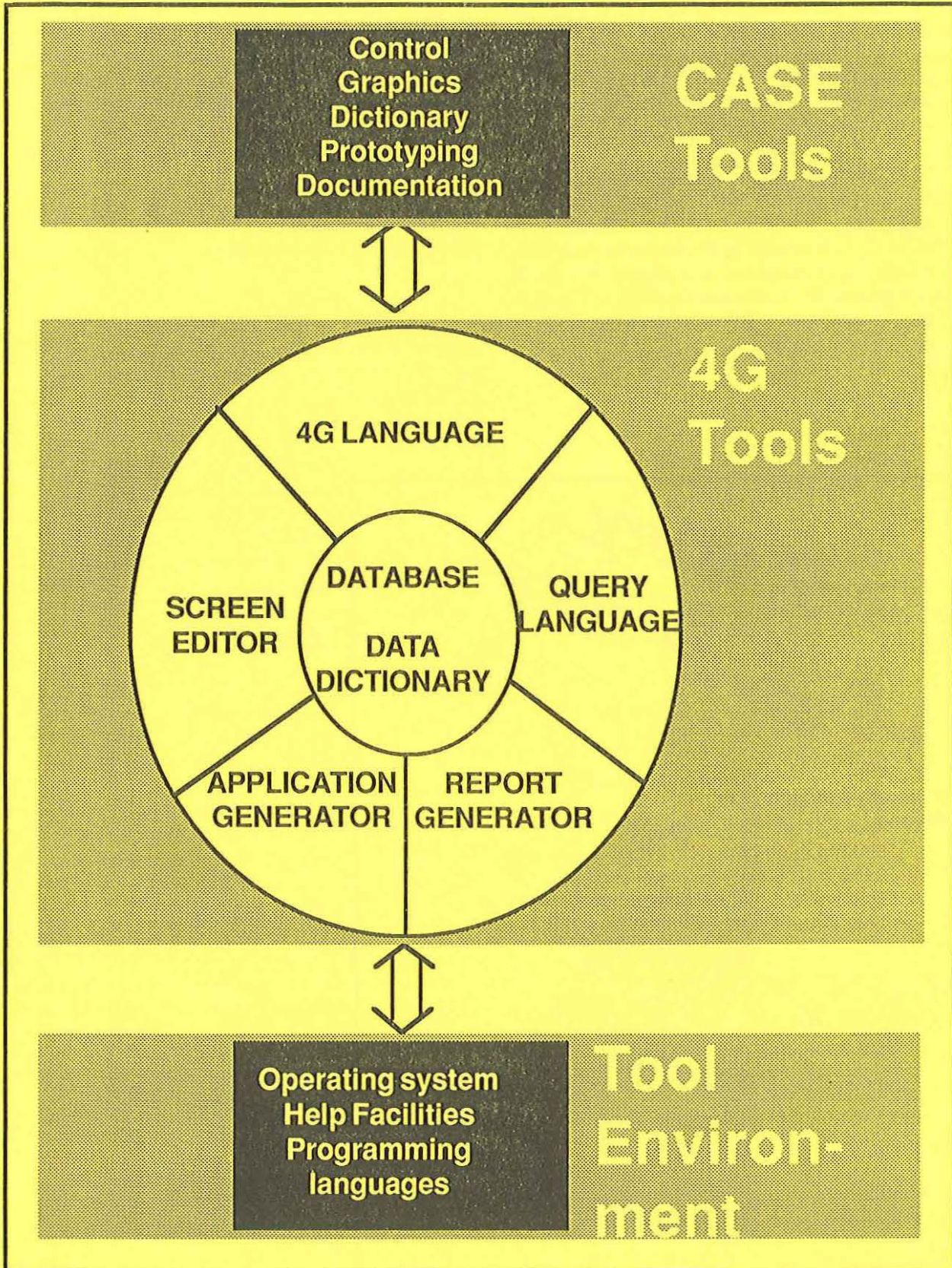
## 2. 4G tools, CASE

This segment describes how the different tools can best be used in practice. Specific method and tool adaptations already exist for some 4G tools and CASE-tools, such as INGRES and DEFT. They contain examples of documentation and tailored method instructions for the tools. For each step in the method an explanation is given how to best use the tool and what to do if the tool does not support the method. Further tool adaptations are being developed.





# Modern Tools





## CASE

CASE, acronym for Computer Aided Software Engineering, stands for computer support of analysis and development. This term encompasses all from simple drawing tools to more advanced CASE tools.

These tools support the initial phases in the development process. The CASE tools that exist today have become complements to those 4G tools that lack analysis support, or else they are independent products which can be combined with different tools.

*CASE tools should have the following capabilities:*

- *Graphical analysis support*
- *Data dictionary support*
- *Analysis/verification rules*
- *Interactive prototyping tools*
- *Automatic documentation*

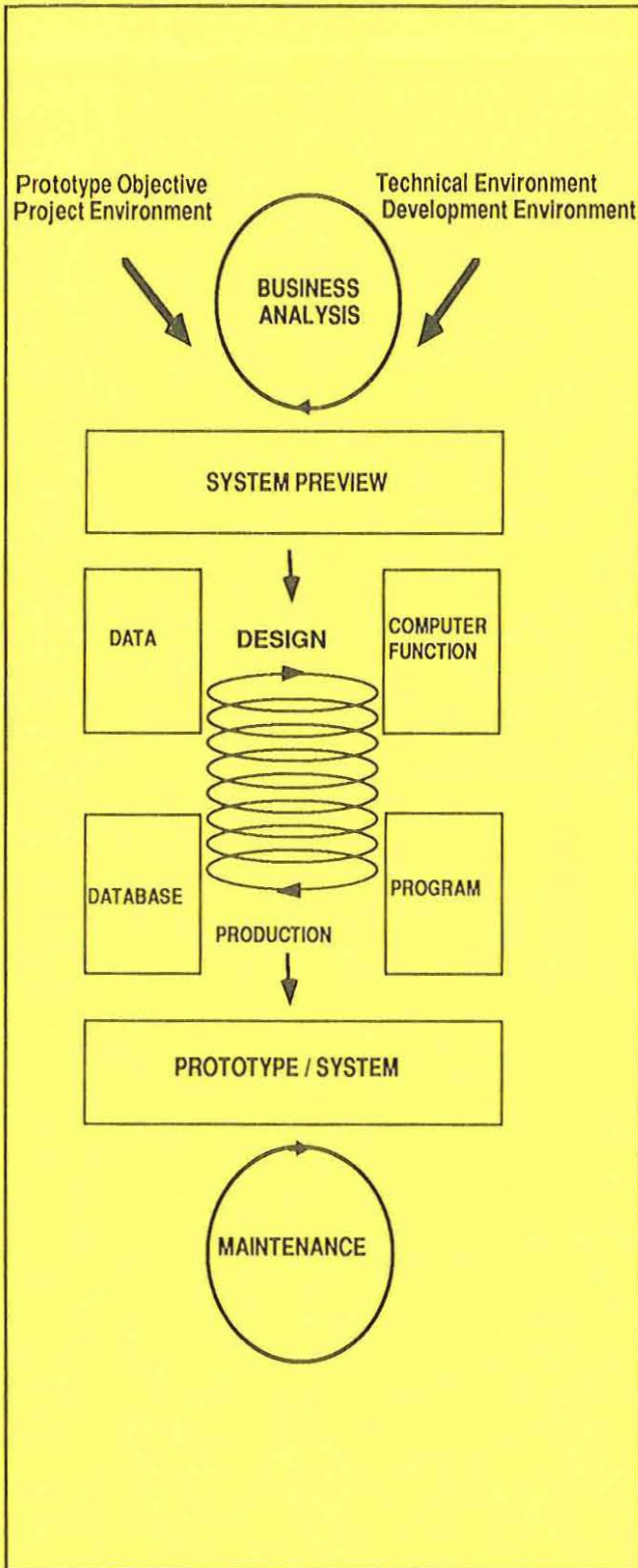
## DEFT

DEFT is an independent CASE tool that supports all phases in system development. DEFT runs on the Macintosh and has bridges to host computer environments such as INGRES, ADABAS, ORACLE, Rdb, Tandem, AS400 etc. Additional gateways to INFORMIX, Sybase, and DB2 are being developed. A gateway to MIMER is also under development. DEFT is easy to use and is compatible with most Macintosh word processing and graphics programs. Today Deft produces screens and data definition-syntax for the 4GTool.

*DEFT CASE tool includes:*

- *Graphical analysis support*
- *Data flow editors, programming structure editors, entity relationship diagramming and screen format editors*
- *Data dictionary for SQL definitions*
- *Analysis/verification module*
- *Automatic documentation*
- *Gateway to host computer*

## This is How it Works



Even with modern tools it is important to begin system design with a business analysis. In simple situations, this is done quickly, in complex situations, a total business analysis must be carried out. By working with simple prototypes of the future system, some results from the final stage of the REFLEX business analysis are attained faster, other steps shortened. The business analysis results in a preview of the information systems required. The preview describes the business, its routines and need of computer support. It divides business activity into small neighbouring subsystems that allow for short projects with active end user participation and prototyping.

The next phase in REFLEX consists of two parallel activities — database construction and program creation.

When working with development, it is important to have a positive working environment and the correct technical conditions.

Development is carried out incrementally and iteratively, symbolized by the spiral in the middle. We call this for prototyping/proto-cycling.

Databases and programs are gradually developed and documented within the 4G tool until the prototype and/or a production system is completed. The design results in a description over the subsystem's data structure as well as data processing functions — the foundation upon which databases and programs will be built.

After successive development and modification, the prototype approaches completion and the finished system can be installed. During its lifetime the system is supervised and subject to continuous maintenance. This provides for further development of the system in step with changes occurring in the business itself.



## Why REFLEX with DEFT?

The following effects and results are attained:

- The correct solution within each business area is found. This is done through business analysis, where each business problem is backed by the correct computer solution.
- The end users have the knowledge, understanding and skill to fully utilize the computer system. This is reached through active user participation during all development phases.
- Quick results and effects are reached in the business. This is attained by a shortened introduction period, since the end users can form and test organizational solutions parallel with forming of the prototype.
- Continuous quality checks through testing and decision-making warrant a high quality computer system. Unknown and critical parts in respect to technology or

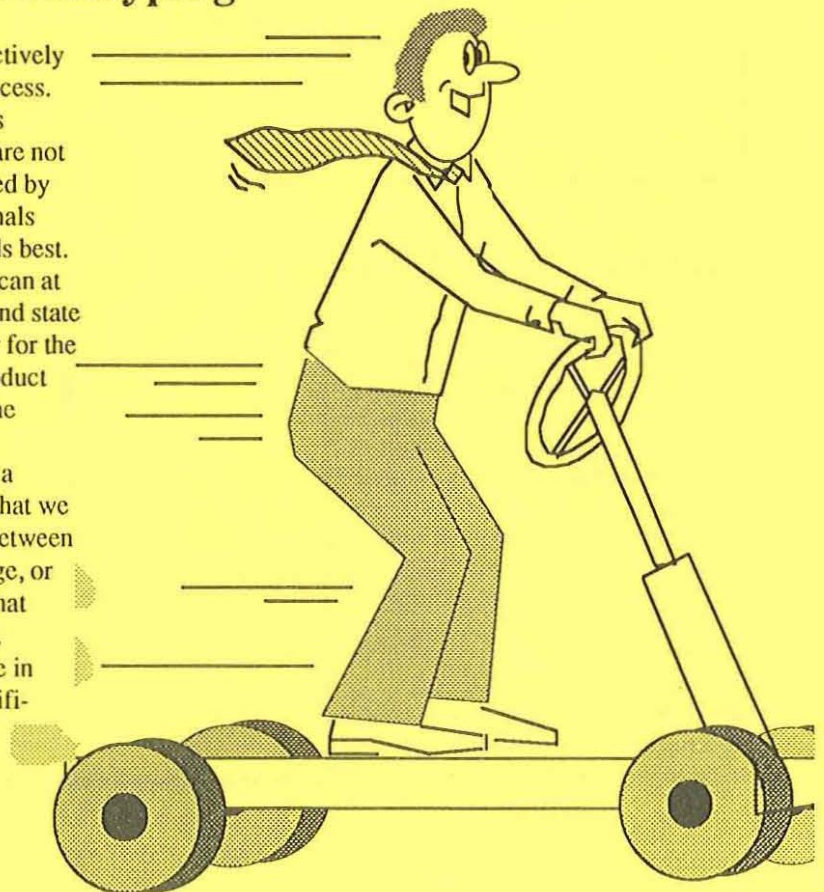
functionality are investigated and tested early during the analysis.

- Project control is facilitated, since the working procedures provide a basis for planning and follow-up.
- Since computer professionals and end users cooperate during the whole development process, their combined competence is utilized more efficiently than in traditional system development.
- Design and production is standardized and therefore can be carried out quickly. This is attained by the effective utilization of tools and an early definition of guidelines, general concepts and standards.
- Maintenance is facilitated by building in a flexible and well documented system structure. This structure is based upon how responsibility, authority and work is organized in the business. Active user participation also results in higher competence in system maintenance

## Prototyping

With Prototyping we mean that the end user actively participates during the whole development process. Results are tested in order to determine what is optimal for the business. Since specifications are not fixed from the beginning, they can be optimized by the end user together with computer professionals. In this case it is the driver who knows his needs best. By testing the first simple model of the car he can at an early stage see the flaws and deficiencies, and state his demands upon the design. It is much easier for the driver to be knowledgeable about a tangible product than in complicated descriptions on paper of the planned final product.

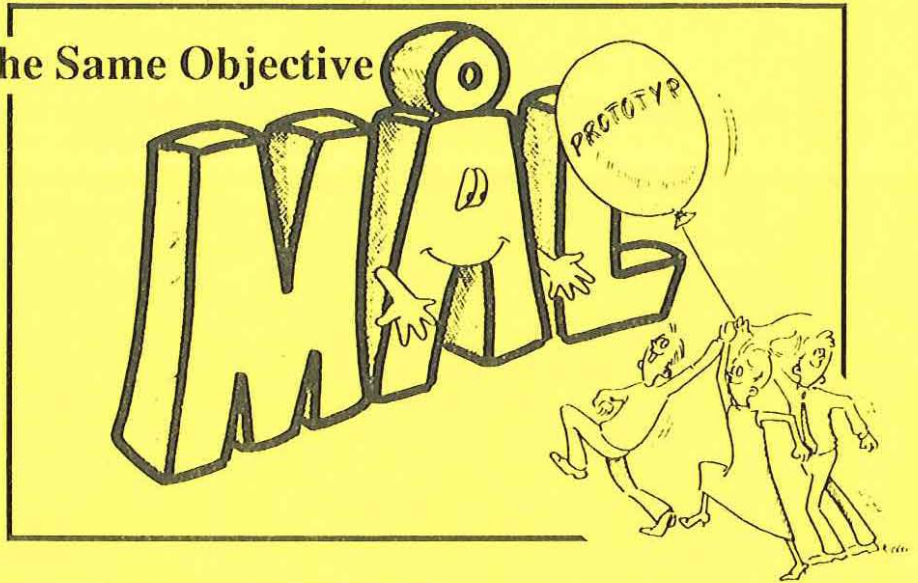
This first simple prototype is then formed into a trimmed and tested finished product. This is what we call prototyping. Sometimes the differences between the prototype and the final objective are so large, or else the operating environment is another, so that production is started from the beginning again. This is called prototyping. Both processes have in common clear and thoroughly tested user specifications.





## Steering Towards the Same Objective

It is very important that objectives and conditions are defined for prototyping, before the actual development takes place. It is important that all are striving towards the same goal, and that only the proper parts of the system become an integral part of the prototype. In REFLEX this is called defining the prototype objective.



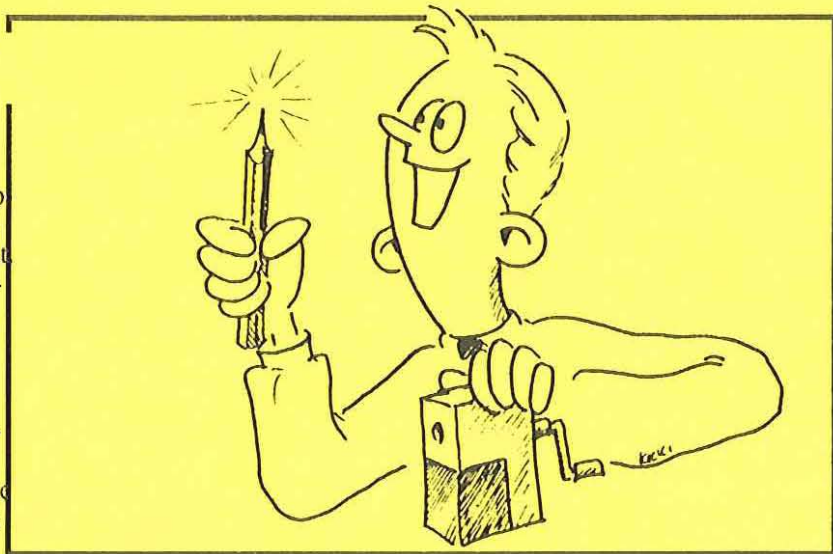
## A Positive Project Environment



In order to work efficiently, the correct work procedures must be applied and the end users must be given the time and the possibility to engage themselves in the project. Furthermore, all practical details must be settled so that end users and computer professionals can cooperate well and that development flows smoothly.

## Sharpen Your Tools

To sharpen one's tools in order to facilitate work is a quite obvious but often forgotten measure. In REFLEX with Modern tools we place much emphasis on this point in order to increase efficiency and to simplify development. In the development and production environment, the tools are modified and supplemented while at the same time standards, guidelines and general concepts are defined. Resulting experiences are systematically utilized. Requirements are specified for the upcoming environment. All this is done with the aim in mind to optimize work effort in the procedural steps to come.

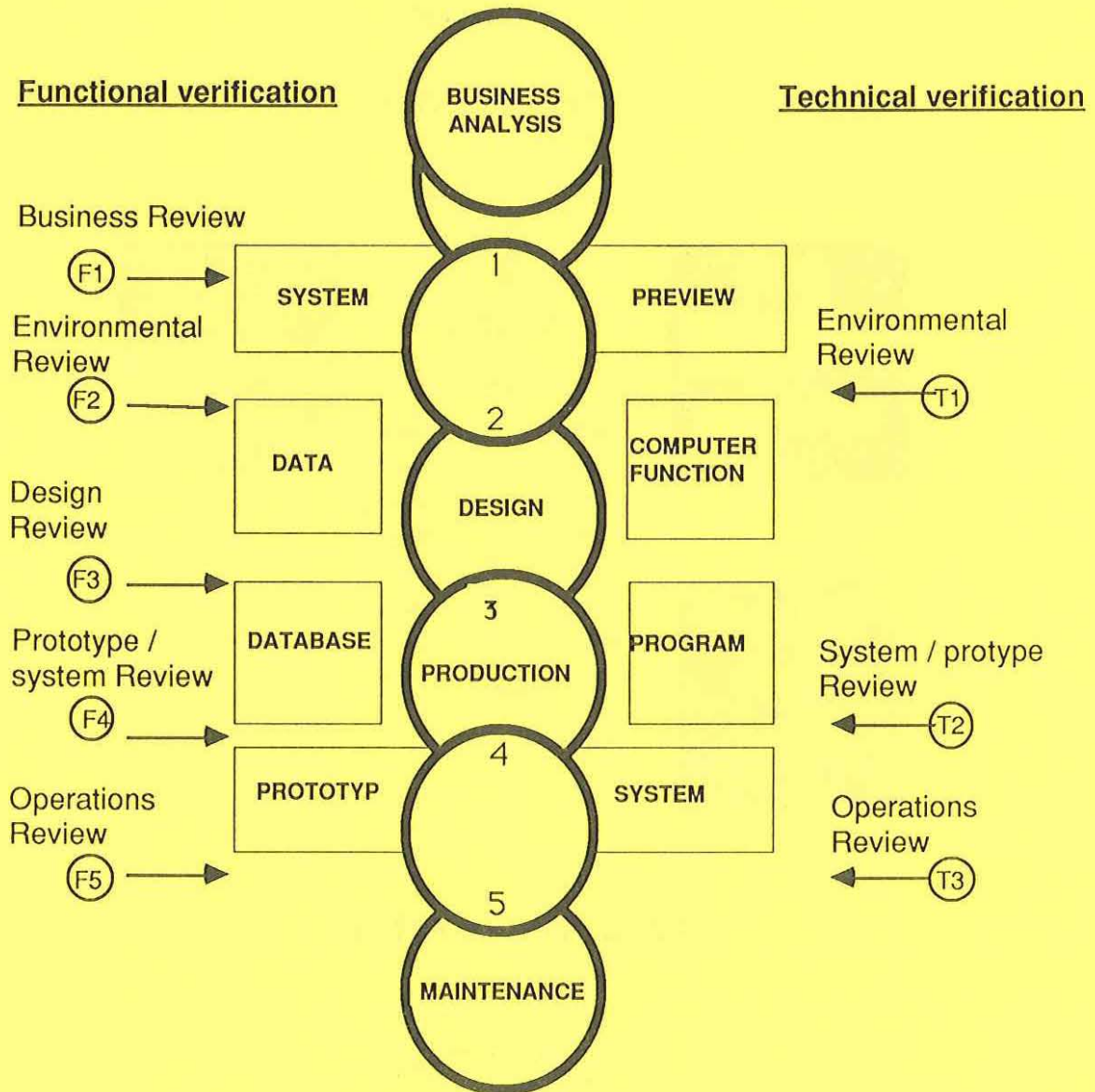




# Quality Control

To build quality into a system after it is completed is almost impossible. Therefore quality must be designed into the system during development. The gradual manner of working which is one of the foundations of REFLEX with Deft,

provides a completely different possibility for checks and verification than what traditional system development allows. Therefore a quality control and decision-making module has been added to this procedure.

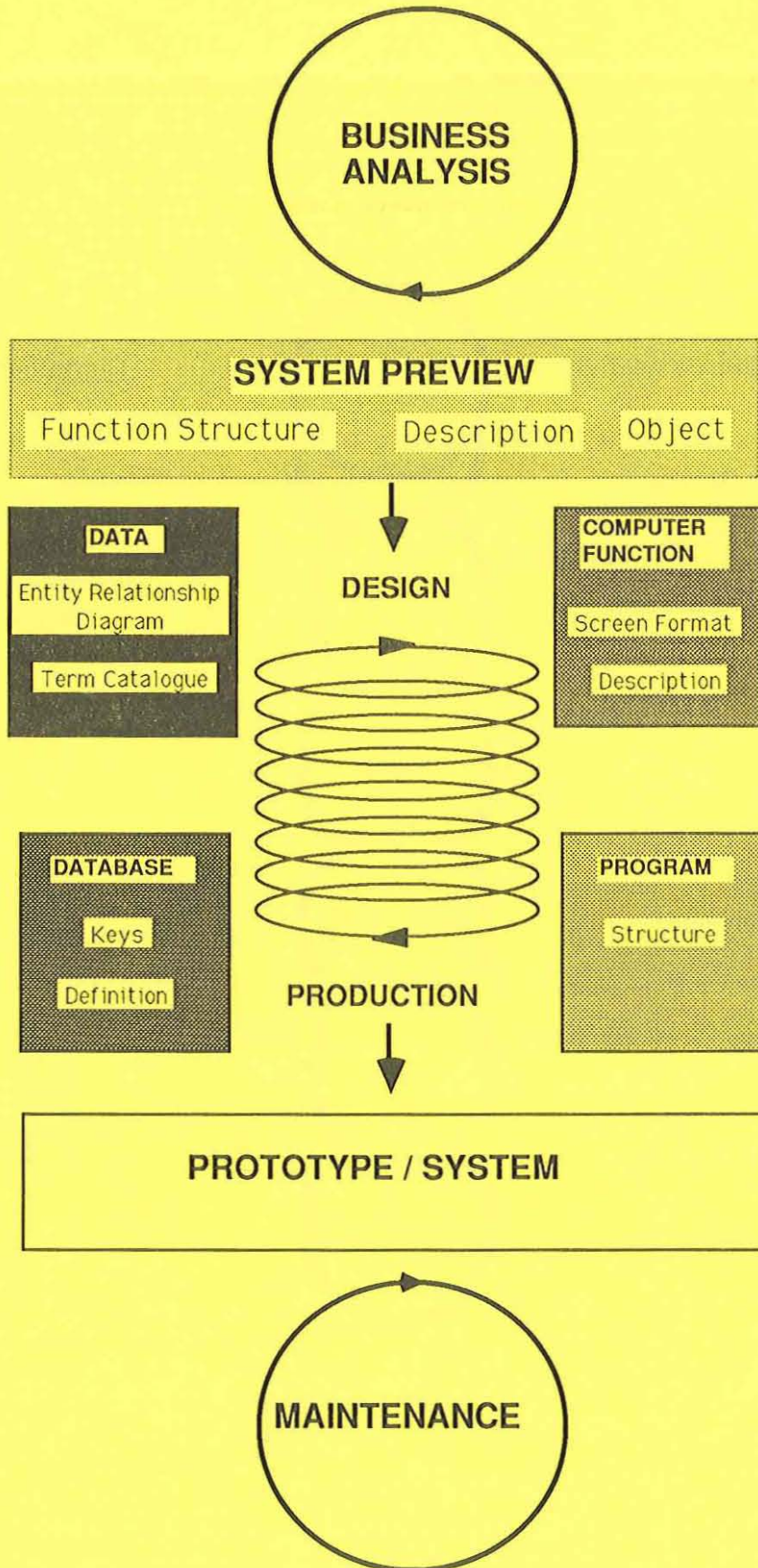


The most important check and review points are shown in the diagram above, to which decision-making points are also attached. The different circles designate the successive testing procedure within each phase. Where the circles meet, the respective decision-making points

are found. If a decision is made to approve a result, one continues into the next circle where new decisions are made. If a result is not approved of, then another round of development and decision making is begun.



# REFLEX with DEFT





## Experiences from a Project carried out at the Swedish Tourist Board

### **Description of the Company and their situation**

The Swedish Tourist Board (STB) is a foundation whose activity is based upon the input and output of information. In other words, STB's business is knowledge oriented, with a great need of information exchange and flexibility. However, it has been difficult to formalize and specify information needs.

Some of the main goals of the STB are increases in tourism income for the Swedish State and to facilitate "tourism for everybody".

In the beginning of 1987 there was no extensive computer support at the STB other than some Macintosh and other PC applications for desktop publishing, with no communication links between them. They also had problems with excessive dependency on a few people and unclear responsibilities. They decided to instigate changes in their situation. A project with consultants from AB Programator and end users from the STB was commenced during Spring 1987.

A new technical environment was implemented during 1987-88, using a Micro VAX II and integrated Macintosh and PC's in a Apple Talk Network for local information processing.

An initial business analysis resulted in uncovering the necessity for different information systems. Powerful and uncomplicated tools were needed in order to simply and quickly change between systems and retrieve information in various ways. Development of these systems is still in progress.

As a first step, an information system named the Product Bank was constructed. The Product Bank keeps track of events, worthy sights, touristic activity in Sweden, etc.

The goal of the Product Bank is to make product information universally available and easily accessible within STB, as well as to rationalize the effort involved in the production of printed materials.

The system must be flexible in respect to new products and to the possibility to search for information.

### **Methodology**

REFLEX adjusted to DEFT and INGRES has been used as the method of system development.

INGRES has been used as a 4G tool and DEFT on the Macintosh as an aid to analysis. Use has been made of the following: DFD (structure of functions), ERD (data models), FORMS (handling of screens) as wells as GATEWAY (transfer to VAX for data base and picture generation).

During Stage 1, a functional system (a prototype) was constructed that treated information concerning only those events which would go into the production of a daily journal. The reason for this was to:



- Test an operational method where the Macintosh and VAX computers were utilized to the maximum.
- Judge the effects of this type of information processing in the daily business routine.
- Find and test routines in order to keep continuously changing data up to date.
- Ascertain the intended method of operations in respect to both maintenance and information retrieval before the system is completely finished.

All documentation is on the Macintosh. Screen layouts and data models are DEFT documents, which are also the basis of the actual pictures and data bases. Word and MacDraw have been used for other documentation. Word documents can be transferred to the VAX and be used in the system as mechanical user's manuals.

There are in fact some steps in the method chain which cannot be done with DEFT. It is very important not to forget these method steps even though the tool does not support them.

### **Results, Effects and Experiences**

- Users that have taken part in the project have thought it to be fun and exciting. Through active participation, they have helped in forming the system and have seen results. In fact, they have increasingly taken over more of the development and construction themselves.
- The system development environment with VAX (INGRES) and Macintosh (DEFT/Word) has functioned well. Descriptions are found only in one place in DEFT.
- The technical solution where the Macintosh and VAX communicate with one another making possible the transfer of data between the INGRES data base and the Macintosh, results in a new and flexible way for the end user to process information. However this process still need some technical improvements to work better.
- If REFLEX methodology and prototyping together with the effective use of tools had not been used, then STB's needs probably could not have been specified nor the technical difficulties solved.
- The end users were educated in the use of the method and DEFT. Because of the clear and simple way of using the tool, they can be active in the systems development process. This is a very stimulating and a creative way of systems development, both for EDP experts and the end users.