Selection of software cost estimation packages

tu/e
Research Report

Eindhoven
University of Technology
Netherlands

Department of Industrial Engineering and Management Science

Selection of Software Cost Estimation Packages

door

F.J. Heemstra
M.J.I.M. van Genuchten
R.J. Kusters

Report EUT/BDK/36
ISBN 90-6757-037-0
Eindhoven, 1989

SELECTION OF SOFTWARE COST ESTIMATION PACKAGES

door

F.J. Heemstra
M.J.I.M. van Genuchten
R.J. Kusters

Report EUT/BDK/36
ISBN 90-6757-037-0
Eindhoven, 1989

Eindhoven University of Technology
Department of Industrial Engineering and
Management Science
Eindhoven, Netherlands

CIP-GEGEVENS KONINKLIJKE BIBLIOTHEEK, DEN HAAG

Heemstra, F.J.

Selection of software cost estimation packages / door
F.J. Heemstra, M.J.I.M. van Genuchten, R.J. Kusters. -
Eindhoven : Eindhoven University of Technology,
Department of Industrial Engineering and Management
Science. - (EUT report / Eindhoven University of
Technology, Department of Industrial Engineering &
Management Science ; BDK/37)
Met lit. opg.
ISBN 90-6757-037-0
SISO 520.9 UDC 657.31:681.3.06
Trefw.: software ; begroten.

SELECTION OF SOFTWARE COST ESTIMATION PACKAGES

Ir. F.J. Heemstra

Ir. M.J.I.M. van Genuchten

Dr. R.J. Kusters

CONTENTS

INTRODUCTION

The use of a model is one of the ways for estimating an software development project. Dozens of software cost estimation models have been developed in the last ten years and today there are many on sale. Well-known examples of estimation models are: function point analysis, COCOMO, PRICE and ESTIMACS. The evaluation of a number of automated versions of estimating models is the subject of a study which the "Management of Software Development Projects" (BAP) research group of Eindhoven University of Technology has carried out for the ISA-TMS department of Philips. This report describes the performance and results of the study. It has been compiled on the basis of a number of interim reports which have been discussed with ISA-TMS. The present report represents the conclusion of the study. The automated versions of the models will be referred to as estimation packages below.

The design of the study is discussed in Section 1. Section 2 deals with the selection of the packages which were evaluated. The study consists of two parts: a theoretical study of the packages and an experiment. In the theoretical study the packages were evaluated on the basis of an assessment method which was devised for this investigation. The theoretical study is described in section 3 of the report. In the experiment, 14 project leaders made a number of estimates using two packages which were selected in the theoretical study. The experiment is described in section 4. Section 5 contains the conclusions and recommendations.

1 DESIGN OF THE STUDY

The aim of the study was derived from the objective formulated by the customer. The statement of the problem is included as Appendix 1 (in Dutch). The aim was formulated in the first report to TMS (TMS1 1988) as follows:

> To develop a method for the evaluation of estimating packages and to test a number of selected packages both theoretically and experimentally.

First of all, a number of packages were selected from the wide range available. This selection is described in section 2. The actual study is divided into two parts: the theoretical study and the experiment. A great many requirements can be tested theoretically, among other things, by studying the documentation and experimenting with the package. Other requirements can only be tested by working with them in practice. In the theoretical study, an evaluation method was first developed on the basis of which the selected packages can be assessed. This method and the evaluation of the selected packages are described in section 3.

The packages rated as adequate were then tested in an experiment. Fourteen experienced project leaders were asked to make an estimate using the selected packages. This related to the estimation of a project which had actually been carried out. The experiment is described in section 4.

As the last part of the study it was planned to have several packages used experimentally by a number of interested departments. It proved, however, that the theoretical study and the experiment formed an adequate basis for drawing conclusions and making recommendations to the customer. For this reason the user test intended as the last part of the study was omitted.

2 PRE-SELECTION OF ESTIMATION PACKAGES

Dozens of estimation packages are currently available on the market. An overview of these is given in Appendix 2. A number of packages were selected from this wide range for testing in the theoretical study. The packages were selected on the basis of the following five criteria:

a The package must be up-to-date and must be supported by a professional supplier.

b The package must be based on projects in which information systems have been developed.

c The package must not use lines of code as input variables. An important requirement of TMS is that the packages must be applicable at an early stage of information system development. In our opinion, however, it is not possible to estimate the number of lines of code accurately at that stage.

d An automated version of the package should be available in view of the possible distribution over a number of ISA departments.

e The packages should be available for carrying out the study within a reasonable period of time.

The preliminary selection resulted in four packages which would be theoretically evaluated: BEFORE YOU LEAP (BYL), ESTIMACS, SPQR20 and BIS/ESTIMATOR.

3 THEORETICAL TEST

3.1. Introduction

This section indicates the requirements to be met by a cost estimation package if it is to provide useful support in estimating software. In addition, a description is given of the method devised for investigating the extent to which the four packages selected (BYL, ESTIMACS, SPQR and BIS Estimator) meet these requirements.

A distinction can be made between the following three categories of main requirements, namely those relating to:
- the context within which the package is used
- the model itself and
- user-friendliness of the package.

Each of these main requirements is subdivided into a number of requirements. An overview of these is given in table 1.

Table 1: Requirements for evaluating estimation packages

REQUIREMENTS		
context	model	user friendliness
- linked to phasing - linked to approach - applicable at an early stage - calibration - use of data which become available - completeness of output - adjustment to objectives	- accuracy - explainability - objectivity - sensitivity analysis - open model - scope - cost drivers - expansion factor	- ease of use - execution time - learning time

These requirements are the result of literature research (Boehm 1981, Noth and Kretzschmar 1984, Heemstra 1986), theoretical considerations and practical experience acquired in working with estimation packages by the "Management of Software Development Projects" (BAP) research group of Eindhoven University of Technology. These requirements are further explained in section 2.2 and the method used for investigating to what extent the selected packages meet each requirement is indicated. Needless to say, not all requirements will weigh equally heavily in each situation. For example, an organization which primarily wishes to use a model at an early stage of software development will attribute great importance to the requirement "applicable at an early stage". The demands to be met as regards other requirements, such as the accessibility of the model, are frequently less high. Therefore we made a distinction between mandatory requirements and other requirements. If a package does not meet a mandatory requirement it is given a negative assessment in any event.

To enable packages which meet the mandatory requirements to be compared we allotted a weighing factor to each requirement. For example, the requirement "linked to approach" weighs three times as heavily as the requirement "adaptation to objectives". These factors for the various requirements are given in section 2.2. In addition, we have assumed that the context and package requirements weigh twice as heavily as those relating to user-friendliness. The extent to which a package meets a requirement is expressed in a mark. A five-point scale is used for this purpose. The marks have the following meanings:

0: does not meet the requirement

1: does not meet the requirement adequately

2: meets the requirement satisfactorily

3: meets the requirement well

4: meets the requirement excellently.

The allocation of weighing factors and the evaluation on the five-point scale depend on the way in which, and the basic aims for which, the relevant organization intends to use estimation models. In view of this we allotted these weighing factors and made the evaluations in close consultation with TMS. The final evaluation of a package is established by adding the evaluations for the individual requirements. The weighing factors and evaluations were chosen in such a way that the final evaluation of a package corresponded to a mark between 0 and 4 (including the limits).

Every member of the research team evaluated each package individually when performing the theoretical test. Next, the individual evaluations were compared with each other. Differences in evaluations were analysed more closely. A unanimous final evaluation was given on a basis of mutual consultation. After a period of fourteen days each package was evaluated again, but this time by the combined research team. At this stage we checked whether the team could still support the previous evaluation. When necessary, adjustments were made in this.

3.2 Requirements

This section lists the requirements which, in our view, must be met by estimation packages. A brief explanation of a requirement is given were necessary.

3.2.1 Context requirements

An estimation package should support the project leader in managing and/or estimating the costs of developing software. It is important for an estimation package to be able to be linked to a project control method (e.g. SDM, PRODOSTA, PARAET, PROMPT). Among other things, this linking means that at the start of each phase and/or activity the use of an estimation package is prescribed by the control method. Another consequence of linking is that terms, definitions, etc. are coordinated with each other.

Linked to phasing

Linked to approach

Applicable at an early stage

Calibration

This means calibrating the package with respect to its environments. If the package does not support this, and also does not make it possible to store data from old projects systematically for the purpose of calibration, it must be regarded as inadequate.

Use of data which become available

During the course of a project, more and more knowledge is acquired about it. A package should use this information.

Completeness of output

Adjustment to objectives

In the first place, most packages will give an estimate of costs, effort and lead time. A package should be capable of making an estimate with which the consequences of restrictions on lead time, quality and effort are made visible.

All requirements relating to the main requirement "context" were tested in the theoretical part of the study. The weighing factor for each requirement is given in table 2, together with an indication of whether the requirement is mandatory.

Table 2: Main requirement "context". The rating is given for each requirement. The sum of the weighing factors is 1.

Main requirement: context (weighing factor: 0,4)		
requirements	factor	Mandatory
linked to phasing	0.15	no
linked to approach	0.05	no
applicable at an early stage	0.3	yes
calibration	0.2	yes
use of data which become available	0.15	no
completeness of output	0.10	no
adjustment to objectives	0.05	no

3.2.2 Model requirements

This second main requirement relates to the required quality of the model. From the user's viewpoint this primarily means that the output (the estimate) must be accurate and checkable. Eight requirements can be distinguished within the main requirement "Model". These will be explained separately below.

Accuracy

A model must produce predictions which approximate sufficiently closely to reality to be a useful tool for estimating and controlling an automation project.

Explainability

It must be possible to investigate how the estimation results have been arrived at.

Objectivity

The way in which a value is assigned to the input variables of the model must be unambiguous.

Sensitivity analysis

The model must support the execution of sensitivity analyses. It must be easy to investigate what effect changing one or more input parameters has on the final result.

Open model

A model is open when the underlying concepts and ideas are known.

Scope

The model should indicate the type of projects on which it is based and for what type of projects it can be used.

Cost drivers

If a model is to be at all capable of producing reliable results, those factors which have a great effect on the necessary costs, effort and lead time will in any event have to be included in the estimate. An overview of these is given in Appendix 3 and the extent to which the four selected packages meet this requirement is also indicated.

Expansion factor

When the estimate has been made for part of a project, it is possible to calculate the estimate for the rest of the project by using a fixed multiplication factor. An approach like this must be avoided, because small estimating errors in the relevant part of the project or a small error in the multiplication factor can lead to major errors for the estimate as a whole.

All the "model requirements", except for the requirement "accuracy", will be evaluated in the theoretical part of the study. The extent to which the models meet the requirement "accuracy" will be investigated in the practical part. The various demands for the main requirement "model", including the related ratings and weighing factors, are given in table 3.

Table 3 Main requirement "Model".

Main requirement: Model (weighing factor: 0.4)		
requirement	factor	mandatory
accuracy	0,25	yes
explainability	0,10	no
objectivity	0,10	no
sensitivity analysis	0,15	no
open model	0,10	no
scope	0,10	no
cost drivers	0,10	no
expansion factor	0,10	no

3.2.3 User-friendliness

In view of the data- and calculation-intensive character of estimation models it must be possible to implement the model on a computer. All this means that user-friendliness constitutes an evaluation aspect for the automated aid. There are requirements as to:

Ease of use

Execution time

Learning time

These three requirements will be tested in both the theoretical part and the practical part (experiment) of the study. Finally, table 4 gives an overview of the three requirements mentioned above.

Table 4. Main requirement: "User-friendliness".

Main requirement: User friendliness (weighing factor 0.2)		
requirement	factor	mandatory
ease of use	0,6	no
execution time	0,3	no
learning time	0,1	no

3.3 Evaluation

The results of the theoretical test are described in this section. The extent to which the packages Before You Leap, ESTIMACS, SPQR and BIS-Estimator meet the requirements formulated are given in succession. In this report we have confined ourselves to the most striking results. For a detailed description of the evaluation results see the report (TMS2, 1988). The theoretical evaluation of the four packages is given in section 3.4.

3.3.1 Before You Leap (BYL)

Before You Leap is a package based on a link-up between function point analysis and Boehm's COCOMO (1981). The package is entirely menu-driven and gives many possibilities for producing graphic results and pre-defined reports. Although BYL, like all the other packages still to be tested, is basically a stand-alone application, it has interface possibilities to Lotus-123, Symphony, Project Managers Workbench, MS Project, DbaseIII and Time Line.

BYL gives predictions about lead time, effort and costs from "preliminary planning & requirements analysis" to "software maintenance". The following activities are distinguished here:

- requirement analysis
- product design
- programming
- test planning
- verification & validation
- project office
- configuration management/quality assurance
- manuals.

Among other things, BYL puts strong emphasis on the cost component "personnel", divided into costs for management and "types" of developers (analysts, designers, etc.). The costs incurred by the user organization for the cost factor "personnel" are not included. In addition, overhead costs in the form of travelling and subsistence costs are not explicitly mentioned.

BYL achieves a satisfactory score for all the mandatory criteria. Calibration is possible. The effect of the COCOMO and FPA cost drivers can be adapted to the user's own situation. BYL is partly based on FPA. This means that it can be used at a reasonably early stage of system development. Both the documentation and the interactive help facility are of good quality and the interaction with the package is greatly facilitated by the clear and accessible set-up. BYL is exceptionally user-friendly and the learning time and processing time are short. Among other things, this is the result of the fact that BYL is based on FPA and COCOMO, two models which are extremely well described in the literature. BYL is therefore completely transparent and the estimation results can thus be followed and checked. BYL takes account of the most important cost-determining factors (see Appendix 3) and sensitivity analyses can be carried out with it.

A less strong point of BYL is that it is based on a fixed life cycle, as formulated by Boehm in his COCOMO model. Estimates are made for the entire path and then divided over the various phases, and over various activities within these phases, in accordance with an apportionment formula. Adjustments to this are not possible. In addition, BYL offers no facilities for approaches such as prototyping, evolutionary development, and so on. Nor is it possible to adapt the objectives. Furthermore, BYL makes absolutely no use of information which becomes known while the project is in progress. BYL claims to be usable for "financial, military, standard, scientific, or any other type of software". This claim does not seem very realistic because the FPA part of BYL is aimed at administrative software and the COCOMO part mainly at non-administrative software. It is therefore difficult to indicate for which field of application the package is explicitly suitable.

On an overall view, BYL is evaluated as satisfactory from the theoretical viewpoint.

3.3.2 ESTIMACS

ESTIMACS consists of a number of modules which enable project management to generate predictions relating to effort, lead time, required personnel costs (including management and "types" of developers, but excluding costs in use), the necessary hardware configuration and risks involved in a project. In addition, Estimacs has a separate model for estimating smaller projects. As an extra, the package also provides the possibility of project management in the case of several projects.
The package makes an estimate covering everything from the "requirements definition" phase to the "installation" phase.
A distinction is made according to the activities management, analysis and programming.
The package supports every division into phases which the user wishes to

make. By means of different phasings and varying productivity for each phase, approximations such as the life cycle, package selection, prototyping, etc. can be supported. All the questions which ESTIMACS asks the user can be readily answered at the start of the project. ESTIMACS can be used at an early stage. It offers a host of output possibilities. The output is divided into phases, within which the necessary effort is divided into the effort to be made by management, analysts and programmers. No distinction is made according to the modules of the product. A positive aspect of ESTIMACS is the possibility of adjusting the objectives. The package offers two facilities for this purpose, namely a restriction in lead time and a restriction in the number of employees available. Sensitivity analyses can be performed with the package. This is neither fast nor easy, however. A strong point of ESTIMACS is that it gives an overview of the variables which have the greatest effect on the final result. The package is well documented with a clear and readily accessible manual. The use of language is generally fairly clear and unambiguous. However, there are no interactive help facilities and the (menu-driven) user's interface is not always equally convenient.

ESTIMACS has a number of disadvantages. For example, it is not readily possible to calibrate the package. For example, the user cannot adjust the influence of cost drivers to his own situation. On the other hand, ESTIMACS supports the possibility (albeit to a limited extent) of adjusting the productivity for each phase, and the distribution of effort over the phases, to the productivity in one's own development environment by adding the project data obtained to the package. A second disadvantage of ESTIMACS is that it does not use new information which becomes available during the project. In addition, the underlying concept on which ESTIMACS is based is unknown. The results of the package are therefore not fully explainable. Far from all the input variables can be measured. The way in which these variables are defined also leaves something to be desired. It frequently happens that various interpretations are possible. The package does not support an unambiguous use of definitions. ESTIMACS is not clear as regards the field of application. It claims to be suitable for "all mainframe applications". We again believe that this is an unrealistic claim. Far from all the most important cost drivers are represented (Appendix 3). It is also not possible to make changes in the collection of cost drivers which are of current importance for a project.

ESTIMACS scores satisfactorily on all mandatory criteria; the final evaluation based on the theoretical test is positive.

3.3.3 SPQR

Like BYL, SPQR/20 is partly based on FPA. FPA is used as an aid for eventually arriving at an estimate of the size of a product in terms of lines of code. The packet is further based on the ideas of Jones. Apart from covering effort and lead time, the model pays a fair amount of attention to the numbers of errors to be expected and the documentation to be produced.

SPQR gives predictions covering everything from the "planning" phase to the "integration/test" phase. Apart from the normal development activities, "management" and "documentation" are also distinguished within this range. The Jones package pays ample attention to employees as a cost factor. A differentiation is made for each phase and/or activity in terms of personnel costs for management and for development (analysis, design, etc.). No estimates are made of the user's personnel costs.

Positive aspects are that SPQR generates a fairly detailed output. Separate indications are given of how much effort is required for management and documentation. A fair amount of information is generated about the numbers of errors to be expected and the quantities of documentation to be produced. In addition, SPQR explicitly supports the possibility that varying objectives may exist. Eight different objectives can be formulated. Although the package is not accessible, the approach is based on a combination of function point analysis and Jones's productivity model. The ideas underlying the package are described in detail in Jones (1986). In addition, the input questions are well documented. Although the results of the package are not complete, they are sufficiently explainable. Admittedly, many of the input variables are not measurable, but the excellent definitions can make a great contribution to consistent and unambiguous use. The most important cost drivers are represented in the model (see Appendix 3). SPQR scores sufficiently well as regards user-friendliness. The use of language in the package is reasonably clear. The documentation, too, is clear and readily accessible. However, no interactive help facilities are available. The processing time and learning time are short.

As against these positive aspects, there are a number of negative points. The most important of these is that SPQR falls short as regards possibilities for calibration. In addition, it uses a fixed phase division. It is impossible for the user to make any changes in this. SPQR supports approaches other than the traditional ones to only a limited extent. In determining the size of the software, SPQR uses the "real" FPA approach. This means that, just as in the case of BYL, there is no such thing as an interface between the FPA questions which are sometimes difficult to answer (at an early stage) and the user/project. In order to find the effect of cost-determining factors (other than the size) the user must reply to a long series of questions, all of which can be answered effectively at an early stage of system development. Another disadvantage is that SPQR does not use new information which becomes available during the project. Furthermore, SPQR admittedly supports the performance of sensitivity analyses, but as in the case of ESTIMACS this is neither quick nor easy. Another disadvantage is that SPQR claims to be suitable for every type of program. In view of the nature of the approach the limitation of the scope to administrative applications appears to be a more realistic assumption.

The most important point on which this package scores unsatisfactorily is the mandatory requirement of calibration. In our opinion, SPQR has insufficient calibration facilities. We therefore cannot advise the continued investigation of this package.

3.3.4 BIS-Estimator

BIS-Estimator is an estimation package based on completely different principles from those already described. According to the documentation it is a "knowledge-based model". This claim is also made by all the other packages. However, we believe that this is the only package which can truthfully make this claim. The method of approach is therefore completely different from that of the other packages. Here, an estimate is made of the necessary costs, effort and lead time for the total project in a very (and in our opinion much too) simple way. This is what is termed the "soft" estimate. After that a so-called "hard" estimate is made for each phase. On the basis of the estimated results for each phase, a new estimate is obtained for the total project by using a kind of extrapolation. The intention is to make the hard estimate for a phase only during or even after the preceding phase. Finally, the package also provides the possibility for making an estimate on the basis of a direct comparison with a number of completed projects to be selected by the user. The user must then indicate how the present project is to be compared with the historical projects.

The package gives estimates for lead time from the "feasibility" phase to the "implementation" phase. Within each phase, a large number of activities are again distinguished, with management and documentation being given a prominent place in each case. In addition, in estimating the personnel costs, the tool distinguishes between the costs incurred for analysis, design, coding, testing, management and so on.

When testing BIS theoretically, the following items were evaluated as positive. The package has various calibration possibilities. In the first place, the productivity for each phase can be adjusted to the actual circumstances in the user's own development environment. Account is taken of this adjustment both in the "hard" and the "soft" estimation method. In addition, in the "soft" method, both the heuristics and the distribution of the effort over the various phases can be adjusted. Finally, an estimate can be made which is directly based on a comparison with a number of completed projects.
BIS is the only package which takes explicit account of the data that becomes available during the project. The estimate is made for each phase of the life cycle. In each phase new questions are asked which take explicit account of the kind of knowledge that should be collected up to this phase.

BIS offers a range of output facilities. Most of the questions asked by the package are measurable. The explanation accompanying some questions is sometimes very inadequate.
BIS supports sensitivity analysis and also permits the execution of what-if analysis. This is neither fast nor easy. The package is clear as regards its field of application and is explicitly aimed at data processing projects.

Less strong points of BIS-Estimator are that the package uses its own, permanent phase division and is based on the fixed life-cycle approach. The use of prototyping is supported, however. In addition, BIS is not equipped for obtaining an estimate for the project as a whole at the start of the project. The method proposed for this (the so-called "soft estimating" based on the number of outputs) is completely inadequate for this purpose.
No changes can be introduced into the objectives of an estimate.
BIS is an open package. The underlying rules, on which a "hard" estimate is based, are known. It is, however, questionable to what extent the

availability of these rules give the user an insight into the model and enable him to explain the estimation results.
Far from all the items in our list of the most important cost drivers are represented in the model (see Appendix 3). The documentation for the package is very brief. No further documentation is given for the questions. Nor is there any further explanation of the underlying model. The package was not very fast and tended to become increasingly slower as more data were collected during the progress of a project.

In view of the poor score which this package obtained for the aspect "applicable at an early stage" it does not appear sensible to us to continue investigating it. This is particularly the case since "early applicability" was one of the most important requirements given by the customer.

3.4 Results of the theoretical evaluation

The final evaluation is expressed in a mark between 0 and 3.75 (including the limits). The requirement "accuracy" has a weight of 0.25. The evaluation of the packages for the requirements listed is given in Appendix 4, in which a number of evaluation points are also mentioned for each requirement. The extent to which a package satisfies this requirement was not investigated until the practical part of the study. Satisfactory is now equivalent to an evaluation of more than 1.75. The final evaluation is listed in table 5.

Table 5 Final evaluation

Package	evaluation
BYL	2.11
ESTIMACS	1.86
SPQR	1.64
BIS/ESTIMATOR	1.51

The BYL and ESTIMACS packages achieve a satisfactory score and meet all the mandatory requirements. SPQR and BIS-ESTIMATOR both have a final score which is unsatisfactory. In addition, SPQR scores unsatisfactorily as regards the mandatory requirement of calibration. BIS-ESTIMATOR does the same for early applicability. This is an extra reason for including only BYL and ESTIMACS in the experiment.

4 THE EXPERIMENT

4.1 Aim of the experiment

In the theoretical study, the packages were tested on the basis of a number of requirements. It is, however, not possible to test the packages with regard to every aspect. Among other things, this applies to the requirement of accuracy. It is also not possible to test the extent of the acceptance of the packages by the possible future users theoretically. These aspects were therefore tested in the experiment.

The objectives of the experiment were:

1 To determine whether the use of packages influences the project leaders

2 To determine the accuracy of the estimate using packages in a semi-realistic situation

3 To have the packages evaluated by the project leaders with regard to a number of the requirements which were distinguished in the theoretical study

4 To determine whether an information plan gives sufficient information on which to base a practicable estimate

5 To determine whether these and similar packages will be accepted in practice

6 To determine whether the number of lines of code can be used at an early stage of development as a good indication for the size of the product to be developed

7 To make a contribution to the ultimate valuation of the BYL and ESTIMACS packages.

4.2 Experimental design

During the experiment, experienced project leaders were asked to make a number of estimates for a project. This related to a project which had actually been carried out. In this project a bonus system was developed for the Philips sales organization in Italy.

The first estimate of the effort and lead time was made on the basis of the project leader's knowledge and experience. From now on, we shall refer to this estimate as the manual estimate. Next, two estimates were made using the packages selected. We shall call these estimates the package estimates. In conclusion, a final estimate was made on the basis of the project leaders' knowledge and experience together with the package estimates. Each estimate was evaluated directly using a questionnaire and the experiment ended with a discussion session.

The experiment was carried out with project leaders from a number of ISA departments. A total of 14 project leaders took part. They came from Consumer Electronics, Medical Systems, Philips Nederland, Philips Applications Services and Concern Service.

4.3 Accounting for the experimental design

It will be indicated below that the objectives of the experiment can be achieved with the experimental design selected.

Re 1) The influence of the packages

The intention is to investigate whether the project leaders allow themselves to be influenced by the use of the models when making an estimate. In this respect, a test set-up is normally chosen in which one group uses only the first package, one group uses only the second package and another acts as a control group. The size of the various groups depends on the size of the variance to be expected. Since this expected variance is great, it follows that the size of the group will also have to be relatively large if reliable results are to be expected. In this respect a total of sixty participating project leaders can be envisaged. Involving the necessary numbers of project leaders leads to costs which are out of all proportion to the importance of the study. We therefore opted for the test set-up in Figure 1.

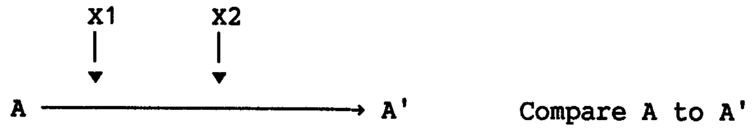

Figure 1   The test set-up

Here, the BYL package was interpreted for a section of the subjects for X1 and the ESTIMACS package for another section. At a starting situation A (to be measured with the manual estimate) two influences X1 and X2 were exerted, after which an end situation A' occurred (to be measured with the final estimate).
The result of this trial set-up was that it was no longer really possible to distinguish between the effects of X1 and X2 on the final result A'. The test was therefore "Can an effect be attributed to the use of the

packages?". Possible differences between the packages can thus no longer be derived from the quantitative data. To do this it is, however, possible to use the answers which the subjects gave to questions specially included in the questionnaires for this purpose.
Summing up, on the basis of the above we believe that it may be said that the set-up selected was not the most ideal one imaginable. It may, however, be assumed that within the limits indicated the results are valid.

Re 2) The accuracy

In assessing the accuracy of an estimate made with a package there are basically two points which must be looked at, namely the average and the variance in the estimation error.

The mean

A requirement which must be met by an estimation method is that of unbiasedness. In the long term, therefore, the mean of the estimation errors obtained by using the method must be small. In this case, however, it should be noted that the packages were not calibrated either with respect to the environment in which the project was actually carried out, or with respect to the environment in which the experiment was performed. There is therefore little point in a direct comparison such as this. We assume that it is possible to adjust the packages to the environment in such a way that an accurate estimate becomes possible. (See, for example, Miyazaki and Mori 1985).

Variance

Even if the above requirement is met, however, this still does not mean that the estimator can be used. To permit this, the variance in the estimation error must be sufficiently small. What "sufficiently small" means in this context will have to be determined by each user himself by indicating which deviations are still acceptable. As regards the operationalisation of the concept "small" it was chosen to compare the variance of the prediction error with the variance in the manual estimates. The point is that these are the estimates as they are normally drawn up. If the package estimates are better than the manual estimates, that is an indication of the practicability of the packages.

Re 3) Evaluation with respect to a number of sub-aspects of the packages.
A number of the evaluation requirements mentioned during the theoretical part of the study were also presented to the project leaders participating in it. The requirements chosen were those for which it could reasonably be assumed that the project leaders were capable of making a judgement based on their acquaintanceship with the packages during the experiment.

Re 4) Suitability of an information plan.
One of the objectives of the experiment was to determine whether the information available after drawing up an information plan constitutes an adequate basis for an estimate. To this end, the project leaders were asked a number of questions. On the basis of the answers to these questions an insight can be obtained into the project leaders' opinions about the quality of the information. These insights can then be combined with the quantitative results of the experiment. Here, the variance of the manual estimates obtained plays a particularly important role.

Re 5) Acceptance.
Testing acceptance will basically mean that a number of future users will be asked in one way or another what they think of the package. The problem here is how to give these users a sufficiently realistic picture of the packages. Another problem is obtaining answers from these users which actually reflect their opinions. We believe that we were able to cope with both problems reasonably well. Firstly, by carrying out this experiment in a semi-realistic situation. An estimate had to be made on the basis of a real case. Secondly, by ensuring that the anonymity of the users was preserved when taking part in the experiment.

Re 6) Size.
The objective was to determine whether it is possible to give a reliable estimate of the size of the system to be developed at an early stage of system development if this size has to be determined on the basis of the expected number of lines of code. The background to this question is the idea that a fairly large number of packages take the number of lines of code as the main input. These packages were removed in an earlier selection because we believe the question about the number of lines of code cannot be answered at an early stage of development. This assumption was tested in the experiment. When making the manual estimate, the participating project leaders were also asked to estimate the size of the system in lines of code.

They were also asked their opinion about this method of estimating the size of a system.
The method of function point analysis was chosen as the basis for comparing the quality of the estimate of the size in lines of code. The BYL package and the ESTIMACS package each indicate the size of the system to be developed in terms of function points.
Both in the case of lines of code and function points a conversion is needed which yields the ultimate size in man-months. Since it is not the quality of this conversion, but the quality of the underlying dimensions which we wish to assess, a problem occurs, namely that lines of code and function points are not directly comparable. It was decided not to compare the original observations directly with each other, but to make a comparison on the basis of the variables x_i/μ_x, in which x_i is the i^{th} observation and μ_x the average of all observations. After this transformation the data are represented in the same unit of measurement, and both are now dimensionless, so that comparison is possible.

Re 7) Package evaluation.
The following were added to the theoretical evaluation already performed:
- the project leaders' views on the quality of the package
- data on the accuracy of the package
- an addition to the theoretical evaluation with regard to five points
- the project leaders' views about the acceptance of the package.

In addition to this, a number of questions emerged during the experiment and during the discussion which could supplement the evaluation of the packages. We therefore believe that the project leaders are sufficiently capable of evaluating the packages.

4.4 Results of the experiment

The results of the experiment are described below. First, the results will be presented. Next, the statistical material obtained will be examined in greater detail. Finally, all the objectives of the experiment will be considered in succession. An extended report on the experiment is (TMS3 1988).

4.4.1 The results of the estimates

The results of the experiment are presented in this section. As has been seen from the description of the experiment, the fourteen project leaders were asked to make an estimate four times for the "bonus system" project.

The results, i.e. the estimated lead time, effort and size, of the four estimates (manual, BYL, ESTIMACS and final estimate) are shown in Table 6.

Table 6 Overview of the estimation results

Project leaders	Estimation results in man-months							SIZE (source lines)
	LT(1)	LT(BYL)	LT(2)	EFF(1)	EFF(BYL)	EFF(EST)	EFF(2)	
A	11	8,9	11	18	28,3	33,3	18	12500
B	8	8	10	16	26	30,2	25	-
C	12	9,2	12	72	30,4	34	40	10000
D	10	8,6	10	10	25,8	41,4	10	8000
E	12	8,3	15	30	30,5	71,5	36	20000
F	12	9,8	12	48	49,2	47	48	-
G	21	10	16	45	53	49,7	47	-
H	9	12,5	20,6	15	36,3	53,1	20,6	-
I	10	8	10	42	23	62	35	-
J	3,5	7	6	8,5	15	33,4	10	8000
K	14	12	12	12	30	42,2	15	20000
L	10,5	4,2	10,5	17,5	3,9	70,9	17,5	-
M	12	4,1	12	30	3,6	62	30	1000
N	12	9,4	12	33	32,9	48,4	33	-
Real lead time: 6 months				Real effort: 8 man-months				6500

The questions relating to the packages were also answered by the Italian developers of the system. Filled into the packages, this yielded the following results:

Effort with BYL: 18 man-months

Lead time with BYL: 7.5 man-months

Effort with ESTIMACS: 54.4 man-months.

The difference between the package estimates and the reality is remarkable. In view of their familiarity with the development environment and their complete knowledge of the project, better package estimates would have been expected here. Furthermore, it is noteworthy that the package estimates of the developers of the system come close to the average package estimates, obtained during the experiment.

4.4.2 Evaluation of the case used

Before the results of the experiment can be developed further, it will first of all be necessary to see whether the case used is of sufficient quality. The participating project leaders were asked a number of questions about this. Asked whether the description of the case fitted in with a project description such as the project leader is accustomed to using when making an estimate, four project leaders replied positively and ten negatively. Three of the fourteen project leaders said that the description given offered less information than they were used to in their everyday practice, ten took the opposite view and one did not reply to this question. Six project leaders stated that the project description offered sufficient pointers for drawing up an estimate, seven said that this was not the case and one project leader had no opinion about this.

Table 7 Overview of the answers to the questions about the case.

Questions		Answers		
		yes	no	missing
is the project leader accustomed to description		4	10	
does the description offer sufficient pointers		6	7	1
description offers more information than usual		10	3	1
this information available at the end of PRODOSTA		phase 1	phase 2	phase 3
		5 x	7 x	2 x
				number
Extra information required on	existing organization			4
	other existing systems			5
	output of the system			4
Most important problem	interface to other systems			11
	acceptance users			5
	different data base environments			3

Asked about the subjects on which they would like to have more information available, extra information about existing systems was mentioned five times, more information about the organization four times and more extensive information about the required output of the software to be developed four times. During the concluding discussion the subject of the quality of the case presented was also dealt with. The general opinion was that the case gave more information than usual. An overview of the answers is given in table 7. Based on these answers we conclude that the description of the case was of sufficient quality to be useful in the experiment.

4.4.3 More detailed examination of the statistical material

Before the information obtained could be used in testing the objectives, the following aspects were looked at first:

a what variables are introduced for processing the available statistical material, or

b may it be assumed that the observations come from a normally distributed population, and

c is there any apparent influence of the sequence in which the project leaders use the packages?

Re a) Description of the variables.

A description of the variables used for processing the results is given in Appendix 5. A number of data on these variables are presented in table 8.

Table 8 Summary of the data on the variables used.

variable	number of observation (N)	mean (M)	variance (S)
LTMAN	14	11.2	3.7
LTBYL	14	8.5	2.4
LTEIND	14	12.1	3.4
EFFHAND	14	28.4	18.3
EFFBYL	14	27.7	14.0
EFFEST	14	48.5	13.9
EFFEIND	14	27.7	12.8
SIZE	7	11.4	6.7
FPABYL	13	138.1	30.8
FPAEST	14	165.1	40.7
FIRST	14	-	-

Re b) Normally distributed population.

The answer to the question of whether the observations were normally distributed is important in selecting the tests for evaluating the results. The Kolmogoroff-Smirnov test for goodness of fit was used to check this. (For this and also the other tests used, see Hollander and Wolfe 1973). The test was performed for all the variables, in which respect we assume:

H_0 the underlying population is normally distributed, and

H_1 the underlying population is not normally distributed.

The zero hypothesis, with a significance level of 5%, was rejected for the variables LTMAN, LTEND AND FPABYL. In addition, the hypothesis with a significance level of 10% was rejected for the variable EFFHAND. Because of this, non-parametric tests will be used in evaluating the results from now on.

Re c) Influence of the sequence

In examining the results, account will also have to be taken of possible influences resulting from the sequence in which the project leaders use the packages. To be able to examine whether such an influence was involved, the observations were divided into two groups based on the variable FIRST. Observations in which the project leader first used the BYL package were placed in one group and the remaining observations were placed in the other group. Within these two groups a test was carried out on the variables LTBYL, LTEND, EFFBYL, EFFEST and EFFEND, using the Wilcoxon rank sum test, in which we assume:

H_0: there is no difference between the two groups, and

H_1: there is a difference between the two groups.

The zero hypothesis was rejected with a significance level of 5% for the variable EFFEST and with a significance level of 10% for the variable LTEND. For these variables it may be assumed that the sequence influenced the results.

4.4.4 The objectives considered individually

As already mentioned in the design of the experiment, there were seven objectives. The results of the experiment will be discussed on the basis of these objectives, using both the quantitative results (the statistical material obtained) and the qualitative results (the answers to the open questions and the discussion results).

Re 1) The influence of the packages

One question was whether the project leaders allowed themselves to be influenced by using the packages when making an estimate. In answering this question we shall first consider whether a significant difference is observable between the manual estimate and the final estimate. Both the estimate of the effort and of the lead time will be examined.

First of all, the Wilcoxon rank sum test for paired observations was used for this. The hypotheses are:

H_0: the difference between the estimates is symmetrically distributed round 0, and

H_1: the difference is not symmetrically distributed.

In both cases the zero hypothesis is not rejected ($\alpha = 0.05$), and no difference can be shown between the manual estimate and the final estimate either for effort or for lead time.

The test used above was a test of location and therefore only considers whether a difference in mean can be shown. If the aim is also to see whether a difference in variance can be shown, a test of the difference in variance for paired observations is needed. No such test exists. To enable an opinion to be expressed in spite of this, the Ansari-Bradley-Freund test for the difference in variance of non-paired observations was used. This is a non-parametric test. Given that not all the assumptions were fulfilled with regard to this test in this situation the results must be treated with a certain caution. The hypothesis is:

H_0: there is no difference in variance between the two estimates,

H_1: there is a difference.

The zero hypothesis is not rejected either for effort or for lead time ($\alpha = 0.05$).

On the basis of this information it cannot therefore be shown that the packages influence the behavior of the project leaders. There are, however, other indications which lead one to suspect that such an influence nevertheless exists. Firstly, it has already been stated above that the sequence in which the packages are used can be shown to have a significant effect on the final estimate of the lead time. In addition, we can look at the answers to the following questions:

- Was your manual estimate the same as your final estate?
- Did the use of the packages influence your final estimate?

To the first question, six answered "yes" and eight answered "no". In the case of the second question, nine answered "yes" and five answered "no". The majority of the project leaders were convinced that the use of the packages did indeed have an influence. Summing up, we therefore believe that the project leaders are slightly influenced by the use of the packages. The direction of this effect cannot be indicated on the basis of the statistical material. A drop in variance can admittedly be observed, but the effect is not so great that much importance can be attached to it.

Re 2) The accuracy

In evaluating the accuracy of a package in the chosen experimental design it is only possible to look at the variance of the observations. In order to be able to judge whether this variance is large or small, the variance of the manual estimate was taken as a reference point. Using the Ansari-Bradley-Freund test, this leads to the following hypothesis:

H_0 no difference in variance is observable

H_1 a difference is observable.

This test was carried out for the two packages, both for lead time and for effort. The zero hypothesis could not be rejected in any of the cases (α = 0.05). Statistically speaking, no difference in variance can therefore be shown between the estimates made by the project leaders and the packages. It is also striking to note the great difference between the actual effort involved in the project and the package estimates made by those who carried out the actual project.

What conclusions can be drawn from this?

The first conclusion is that it has not been shown that the package estimates are poorer than the manual estimates. Looking at the figures, it can be seen that the variances in the package estimates are admittedly not statistically significant, but they are nevertheless lower than those of the corresponding manual estimates (see table 8). A second conclusion can be drawn on the basis of the remarkable difference between the average estimation results for the BYL and ESTIMACS packages. Here there is a difference of almost a factor of two, while the variances do not differ much from each other. This again underlines the need for calibration.

Re 3) Evaluation of a number of sub-aspects of the packages.

The project leaders were asked to what extent the BYL and ESTIMACS packages met five requirements. Table 9 shows the evaluations with regard to the five requirements, averaged over all fourteen project leaders. The evaluation which emerged from the theoretical study is also given.

Table 9: Mean and standard deviation in the evaluation of the BYL and ESTIMACS packages by the project leaders with regard to five criteria compared with the theoretical evaluations.

Criteria		evaluated in : experiment M	S	theory
EASE OF USE	BYL EST	2.8 2.4	0.9 1.0	3.5 1.7
LINKED TO APPROACH	BYL EST	1.4 1.4	1.0 0.6	2 4
APPLICABLE AT AN EARLY STAGE	BYL EST	2.0 2.6	0.5 0.7	3 3
COMPLETENESS OF OUTPUT	BYL EST	1.7 2.2	1.1 1.1	2 2
OBJECTIVITY	BYL EST	1.9 1.9	1.0 0.7	3 2

Note that, in general, the project leaders give slightly poorer marks than those which emerged during the theoretical part of the experiment. This might be explained by the fact that the theoretical evaluation included packages with a poorer rating against which these packages stood out more positively. In any event, this indicates the relative value of such an evaluation.

Re 4) Suitability of an information plan.

It is necessary to examine whether the information included in an information plan is of sufficient quality to form the basis for an estimate.

The project leaders were asked a number of questions for this purpose. The answers to these questions have already been dealt with in 4.2. Six project leaders took the view that this information does not give sufficient pointers for making an appropriate estimate.

A look at the results confirms this picture. On the basis of the estimates obtained, a $1-\alpha$ reliability interval can be established, the lower limit of which is equal to $M-\mu_\alpha/2*S$ and the upper limit is equal to $M+\mu_\alpha/2*S$. In this respect, M and S are the random sample mean and the random sample standard deviation, respectively. In addition, $\mu_\alpha/s*S$ is a quartile of the standard normal distribution. The actual estimate will now lie within this interval with a probability of 1-α. If we now take an 80% confidence interval for the various estimates (80% is not such a strict requirement) this can act as a guide in evaluating the quality of the information included in the case (see table 10). If we compare the range of the field with the magnitude of the variable to be estimated, it is clear that we need not have much faith in the quality of the prediction.

Table 10: 80% confidence intervals for a number of attempted estimates. $\mu_{10} = 1.28$.

variables	M	S	lower limit	upper limit	interval
LTBYL	8.5	2.4	5.4	11.6	6.2
LTEIND	12.1	3.4	7.7	16.5	8.8
EFFBYL	27.7	14.0	9.7	45.7	36.0
EFFEST	48.5	13.9	30.9	66.7	35.8
EFFEND	27.3	12.8	10.8	43.8	33.8

The following points emerged in the above:
- the project leaders cannot make an estimate of "sufficient quality" on the basis of this information,
- the packages also yielded poor results on the basis of this information.

From this it may be concluded that on the basis of the information from an information plan no good results need be expected even from experienced project leaders. In addition, it may be stated that on the basis of this information no assistance may be expected from the packages investigated. This is in line with the comments made about this by the participating project leaders both during the experiment and during the discussion.

Re 5) Acceptance

The question asked here is whether such packages would also be accepted in practice by those who will have to use them. For this purpose, the following questions were asked in the evaluation of both the BYL package and the ESTIMACS package:
- Do you think that the package can be used for estimating projects in practice?
- Would you use this package in practice?

The following questions were asked in addition to this:
- Do you regard such packages as a possible aid in estimating software

projects?

- If one or more of these packages were available to you would you use it or them for estimating software projects?

The answers to these questions are summarised in table 11.

Table 11: Overview of the answers to questions about acceptance.

question	answer		
	yes	no	missing
can BYL be used	9	5	0
would you use BYL	6	8	0
can Estimacs be used	8	5	1
would you use Estimacs	7	5	2
are such packages a possible aid	11	2	1
would you use one of these packages	11	2	1

Finally, the project leaders were also asked to give an overall verdict on both packages. The results are shown in table 12.

Table 12: The verdict of the project leaders on both packages, combined with the mean and the standard deviation over the total evaluation for each package. For an explanation of the marks 1, 2 and 3 see section 3.1.

Package	Project leaders														total	
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	M	S
BYL	1	1	1	3	3	3	2	3	3	3	1	1	1	1	1.9	0.96
ESTIMACS	1	1	3	2	2	2	2	2	1	2	3	1	2	1	1.8	0.67

The BYL package is given an average evaluation of 1.9, in other words almost but not quite satisfactory. For ESTIMACS this score is a fraction lower, namely 1.8. It is interesting to note that the standard deviation of the evaluation of the ESTIMACS package is significantly smaller than that of the BYL package (Ansari-Bradley-Freund test, $\alpha = 0.05$). There is obviously greater unanimity among the project leaders about the quality of ESTIMACS than about that of BYL. The view that the present method of drawing up an estimate is inadequate is virtually unanimous. Even though the quality of the present packages is not very great, it is still advisable to use them as a tool. In the project leaders' opinion, the greatest advantage attainable with such packages at present is the possibility of using them as a kind of check list; "The models draw your attention to a number of aspects which you would otherwise have overlooked". Another advantage was the possibility of ascertaining the sensitivity of the cost-determining factors.

re 6) Volume

The question asked was whether the number of lines of code can be used at an early stage of system development as a measure for the volume of the system to be developed. Function point analysis - another method for determining the volume of a product - was used as a reference in the statistical analysis. Both the BYL package and the ESTIMACS package give an estimate for the volume of the product, expressed in function points. The Ansari-Bradley-Freund test was used for the comparison:

H_0: the relative variance of the volume, estimated in function points, is equal to that of the volume estimated in lines of code

H_1: the relative variance of the volume, estimated in function points, is smaller than that of the volume estimated in lines of code.

Both FPABYL and FPAEST were used for the test. In both cases, the zero hypothesis was rejected ($\alpha = 0.05$). On the basis of the statistical material it can therefore clearly be shown that lines of code as an estimator for the volume of a product at an early stage of development function less well than an available alternative, namely function points. This conclusion was further confirmed by the fact that only seven of the project leaders regarded themselves as capable of giving such an estimate of the volume in lines of code and that also during the discussion it emerged that the project leaders had absolutely no confidence in this measure.

Re 7) Package evaluation

What was said above with regard to accuracy (re 2), the evaluation of a number of sub-aspects of the package (re 4) and its acceptance (re 6), can be used in the final evaluation of the packages.

In addition to this, the project leaders were also asked questions in order to arrive at a more finely honed verdict on the packages. Strong points of BYL are:

- the user-friendliness (mentioned nine times)
- the possibility of using this package as a check list (four times).

The weak points of BYL according to the project leaders are:

- the lack of clarity about how the package works (mentioned five times)
- the impossibility of introducing one's own phasing (three times)

- difficulties in adjusting the coefficients (three times)
- over-emphasis on the coding phase (three times).

The strong points mentioned for ESTIMACS were:
- user-friendliness (mentioned ten times)
- the package asks clear questions (five times).

According to the project leaders the weak points of ESTIMACS are:
- too few questions are asked and they are not the right ones (five times)
- the content/operation of the package is unknown (four times).

The project leaders were also asked what items they missed in both packages. As regards BYL the answers were:
- the package must take more account of environmental influences (four times)
- the package should offer the possibility of using one's own phasing (three times)
- the package must offer possibilities for manipulating with the relation between lead time and effort (three times).

In the case of the ESTIMACS package, all the items which the project leaders missed related to the poor possibilities of linking up the package with the user's own development environment.
A defect of the two packages evaluated which was mentioned very emphatically during the discussion was that there was no possibility of a link-up with Philips development standards, definitions, concepts, etc.
All in all, the participants were not wildly enthusiastic about these packages, but they were nevertheless felt to be useful. The following comment must be added to this. A mandatory requirement set by the customer was that the packages must provide support in making an estimate at an early stage of system development. From the results, it may be concluded that neither of the two packages meets this requirement.

5 CONCLUSIONS AND RECOMMENDATIONS

5.1 Conclusions

The BYL, SPQR, ESTIMACS and BIS packages were evaluated in the theoretical study. The BYL and ESTIMACS packages scored satisfactorily and meet all the mandatory requirements. These packages were further examined in the practical test. Both SPQR and BIS-ESTIMATOR have an end score which is unsatisfactory. For this reason alone these packages were not included in the rest of the study. In addition, SPQR scores unsatisfactorily as regards the mandatory requirement of calibration and BIS-ESTIMATOR as regards early applicability.

The BYL and ESTIMACS packages were evaluated in the experiment. The conclusions of the experiment were based on quantitative results and the opinions of the project leaders concerned. The most important conclusions are:

- On the basis of the differences found between the estimates and reality, we conclude that it has not been shown that the selected packages can be used for estimating projects at an early stage of system development. These results are more or less a confirmation of the results of Kemerer (1987).
- If the requirement "applicable at an early stage" is disregarded, then although both packages are not evaluated as good, they are regarded as useful. There are no great differences between the two packages.
- Based on the differences between the manual estimates and reality we doubt whether on the basis of the information from an information plan experienced project leaders can be expected to achieve results which approximate to reality with a reasonable degree of certainty.
- The package estimates are no poorer than the manual estimates made by the project leaders.
- On the basis of the striking difference between the average estimation results of the BYL and ESTIMACS packages it is apparent that simply using a package without adapting it to the environment in which it is used will not lead to accurate results. Calibration is essential.
- The project leaders expressed the need for a tool to help them when estimating projects. If a package is used as a tool it will, in their opinion, mainly be valuable as a check list and as a means of communication.
- The fact that the terminology does not link up with the terminology used

in Philips is regarded as a weak point of the packages.

- On the basis of the statistical material it has been shown that lines of code work less well than function points as an estimator for the volume of a information system at an early stage of development. In addition, the project leaders have absolutely no confidence in the unit "lines of code".

5.2. Recommendations

On the basis of the theoretical study and the experiment we arrive at the following recommendations.

- The original requirement "applicable at an early stage" should be adjusted. It is unlikely that there are any packages which give useful estimates at such an early phase of the project.
- On the basis of the need for tools for estimating automation projects a package should be purchased and distributed to departments interested in this. In view of the results of the theoretical study we advise that a choice be made between BYL and ESTIMACS. The choice between these two packages cannot be based on the results of this study. Both packages score approximately equally in both the theoretical study and in the experiment. The opinions expressed on the packages by the participants in the experiment are also divided. The choice between BYL and ESTIMACS will partly have to be based on aspects such as price and the reliability of the supplier. Since these aspects were not included in the study we leave the choice between the two packages to TMS.
- Various participants confirmed that an estimation package must be part of an development approach. We share this view and therefore advise that any estimation package to be purchased should be integrated in ISES.

Literature

(BIS 1987) BIS Estimator user manual, version 4.4, BIS applied systems Ltd., 1987.

(Boehm 1981) Boehm, B.W., Software engineering economics, Prentice Hall, Englewood Cliffs NJ, USA, 1981

(BYL 1986) Before you leap, users guide, Gordon group, 1986.

(CA 1986) Computer associates, CA-Estimacs user guide, release 5.0, July 1986.

(Heemstra 1987) Heemstra, F.J., Wat bepaalt de kosten van software, Informatie, June 1987, special

(Jones 1986) Jones, T.C., Programming productivity, McGraw-Hill, 1986.

(Kemerer 1987) An empirical validation of software cost models, Communications of the ACM 30, 5, 1987.

(Noth 1984) Noth, Th., Kretzschmar, M., Aufwandschatzung von DV Projekten, Springer-Verlag, Berlin, 1984

(SPQR 1987) SPQR user manual, 1987.

(TMS1 1988) Genuchten, M.J.I.M., Heemstra, F.J., Kusters, R.J., The evaluation of software cost estimation models; a preliminary report, 1988.

(TMS2 1988) Genuchten, M.J.I.M., Heemstra, F.J., Kusters, R.J., Theoretische toetsing van begrotingspakketten (in Dutch), 1988.

(TMS3 1988) Genuchten, M.J.I.M., Heemstra, F.J., Kusters, R.J., Verslag van het experiment (in Dutch), 1988.

Appendix 1 Statement of the goal

Appendix 2 Overview of the available models

Since the mid seventies a lot of software cost estimation packages have been developed. The most important ones are listed in table 13. Next to the name a reference for more information is given.

Table 13: List of existing software cost estimation models

MODEL	REFERENCE
BIS Estimator	BIS Estimator, user manual, version 4.4, BIS applied systems ltd. 1987
BYL	Before you leap, users guide, Gordon group, 1986
COCOMO (and derivates)	Boehm, B.W., Software engineering economics, Prentice Hall, 1981
De Marco's model	De Marco, T., Controlling software projects, Yourdon Press, 1982
CA-ESTIMACS	Computer associates, CA-ESTIMACS user guide, release 5.0, July 1986
Function point analysis	Van Straten, R., Functie punt analyse, theorie, praktijk en resultaten, Informatie, Special, 1987
Halstead metrics	Halstead, M.H., Elements of software science, North Holland, 1977
PRICE	Cuelenaere, A.M.E., e.a., Calibration of a software cost model, why and how, Information and software technology, December 1987
SLIM	Putnam, L.H., Fitzsimmons, A., Estimating software costs, September and October 1979
SPQR	Jones, C., Programming productivity, Mc.Graw-Hill,1986
Walston & Felix	Walston, C.E., Felix, C.P., A method for programming measurement and estimation, IBM systems journal 16, 1977
Wolverton	Wolverton, R.W., the cost of developing large-scale software, IEEE transactions on computers, June 1977

Appendix 3 Representation of cost drivers

Research, carried out at the Eindhoven University (Heemstra 1987) indicates that it is possible to distinguish the twenty most important cost-drivers. In table 14 the cost drivers are given. In the same table is shown which of the cost drivers are represented in the selected models.

Table 14 Representation of cost drivers

	BYL	Estimacs	SPQR	BIS/EST
size	*	*	*	*
reliability	*	*	*	
size database	*			
complexity	*	*	*	*
amount of documentation			*	*
reuse of software	*	*	*	
constraints	*	*	*	
tools	*		*	*
use of modern programming techniques	*			
quality team	*			
experience team	*	*	*	
turnover personnel				
quality management				
duration	*		*	
projectcontrol method				
user participation		*		
number of users	*		*	*
changing requirements		*	*	
education/training				

EVALUATION ON REQUIREMENT:	OF PACKAGE:	BYL	ESTIMACS	SPQR	BIS
1	**CONTEXT**				
1.1	Linked to phasing	2	4	1	2
-	does the package use a phase distribution				
-	are more phase distributions possible				
-	is it possible to adjust the phase distribution				
-	is it possible to use an organization specific phase distribution				
1.2	Linked to approach	2	4	2	2
-	does the package support more approaches				
-	does the package support prototyping				
1.3	Is the package early applicable	3	3	3	1
1.4	Calibration	2	2	1	1
-	is it possible to calibrate the package				
1.5	Use of data which become available	0	0	0	4
-	are during the course of a project				
	- different questions asked				
	- more detailed questions asked				
	- consistent definitions being used				
1.6	Completeness of output	2	2	2	1
-	distinction between the phases of a project				
-	distinction to the modules of a product				
-	distinction to classes of personnel needed				
-	increase of the amount of detail during the project				
1.7	Adjustments to objectives such as:	0	3	4	0
-	shortening of lead time				
-	decrease of available personnel				
-	changes in quality demands				

EVALUATION ON REQUIREMENT: **OF PACKAGE:**

	Requirement	BYL	ESTIMACS	SPQR	BIS
2	**MODEL**				
2.1	Accuracy	-	-	-	-
2.2	Explainability	4	2	2	2
2.3	Objectivity	3	2	2	2
-	is the way in which a value is assigned to the input variables unambiguous				
2.4	Sensitivity analysis	3	2	3	1
-	does the model support sensitivity analysis				
-	does the model indicate which parameters exert the main influence on the end result				
-	is a what/if analyse possible				
-	is the sensitivity analysis simple				
-	is the sensitivity analysis fast				
-	is it possible to compare the results of the sensitivity analysis directly with previous results				
2.5	Open model	4	2	2	2
-	is the underlying model published				
-	are the principles on which the underlying model is based published				
2.6	Scope	1	2	2	4
-	is the scope given				
-	if the scope is not given, is it then possible to deduce the type of projects at which the package is specifically aimed				
2.7	Cost drivers	2	1	1	2
-	the most important cost drivers are represented				
-	it is possible to make a selection of relevant cost drivers from a pre-defined set				
-	it is possible to add relevant cost drivers				
-	the number of cost drivers is not to high				
-	the number of cost drivers is not to low				
-	the input parameters do not overlap				
2.8	Expansionfactor	3	2	2	1

EVALUATION ON REQUIREMENT: OF PACKAGE:

3 USER FRIENDLINESS

Requirement	BYL	ESTIMACS	SPQR	BIS
3.1 Ease of use				
- straightforward questions	3	2	3	2
- no inordinate use of jargon				
- unambiguous use of language				
- documentation	4	3	3	2
- presence				
- quality				
- support use of package				
- support model				
- support questions				
- man-machine interface	4	2	2	2
- ease of use				
- edit facilities				
- switching between parts of the package				
- on-line help facilities	3	0	0	2
- presence				
- context sensitivity				
- clarity				
3.2 Execution time				
- input	3	2	3	1
- starting the system				
- processing	2	2	1	1
- run time				
- time needed for generating reports				
- time needed for interpreting the output	2	2	-	2
3.3 Learning time	4	2	3	2
- time needed to study the package (norm, approximately 0.5 day)				

Appendix 5 The variables introduced for the processing of the data

Name: LTMAN
Description: the lead time in months that was estimated manually

Name: LTBYL
Description: the lead time in months that was estimated using BYL

Name: LTEND
Description: the final estimate of the lead time in months

Name: EFFMAN
Description: The effort in man-months that was estimated manually

Name: EFFBYL
Description: the effort in man-months that was estimated using BYL

Name: EFFEST
Description: the effort in man-months that was estimated using Estimacs

Name: EFFEND
Description: the final estimate of the effort in man-months
Name: SIZE
Description: the manual estimate of the size of the software program in thousands of lines of code

Name: FPABYL
Description: the estimate of the size of the software program in net function points, as calculated using BYL

Name: FPAEST
Description: the estimate of the size of the software program in net function points, as calculated using Estimacs

Name: FIRST
Description: this variable indicates in which order the project leader used the packages.
FIRST=0 means Estimacs was used first
FIRST=1 means BYL was used first