The MERMAID Project

A J C Cowderoy, J O Jenkins and A Poulymenakou
School of Informatics, City University
London, EC1V 0HB
United Kingdom

1. INTRODUCTION

The tendency for software development projects to be completed over schedule and over budget has been documented extensively(1, 2) Additionally many projects are completed within budgetary and schedule target only as a result of the customer agreeing to accept reduced functionality.
In his classic book, The Mythical Man Month, Fred Brooks(3) exposes the fallacy that effort and schedule are freely interchangeable. All current cost models are produced on the assumption that there is very limited scope for schedule compression unless there is a corresponding reduction in delivered functionality.
The Metrication and Resource Modelling Aid (MERMAID) project, partially financed by the Commision of the European Communities (CEC) as Project 2046 began in October 1988 and its goals were as follows:

- Improvement of understanding of the relationships between software development productivity and product and process metrics

- To facilitate the widespread technology transfer from the Consortium to the European Software industry

- To facilitate the widespread uptake of cost estimation techniques by the provision of prototype cost estimation tools.

MERMAID has developed a family of methods for cost estimation, many of which have had tools implemented in prototypes. These prototypes are best considered as toolkits or workbenches. Figure 1 gives an architectural overview of these prototypes.

The applicability of the tools developed by the MERMAID consortium is considered to encompass both embedded systems and management information systems.

The first prototype was demonstrated in November 1990. It was developed on a SUN 3/60 Workstation using an objective oriented extension of the C language, Objective C. Two versions exist, one running under the Portable Common Tool Environment (PCTE) and the other is a UNIX implementation. The second prototype was demonstrated in November 1991, versions of which were developed on an IBM PS/2 running either WINDOWS 3 or OS/2 and Presentation Manager.A first commercial version was demonstrated at the 14th International Conference on Software Engineering in May, 1992. A third and final prototype was demonstrated at the 1992 ESPRIT Technical Conference.

2. MERMAID ESTIMATION METHODS

At the time of the start of the MERMAID project, October 1988, the commonly used approaches to cost estimation were as follows:

- Expert Judgement, ie informal guestimate of the resources required.
- Analogy, similar to the above but influenced by the identification of a completed project, similar to the one being planned.
- Use of a cost estimation tool based on one of a number of existing parametric models of the relationships between project cost parameters and cost drivers. Models on which tools were based included SLIM(4), COCOMO(5), and COPMO(6).

Tools based on these models can be calibrated for a particular environment. However, research has shown that despite calibration, the accuracy of estimates produced by cost estimation tools is poor.(7)

There are several contributing factors to this inaccuracy. These include the difficulty, if not impossibility, of estimating the size of the product to be developed as early in the lifecycle as the Requirement Analysis Phase. Additionally calibration depends on the existence of moderate quantities of past project data collected in a consistent manner.

The MERMAID approach is based on the use of locally-based and user-defined metrics. This has the advantage of ensuring consistency and increasing accuracy of forecasts. As indicated above,today's tools based on parametric models normally require the project manager or estimator to input an estimate of the size of the software product to be developed. This is either expressed in Lines of Code (LOC) or in the form of a function-based metric such as Function Point Count.(8) It is the view of the MERMAID project that wherever possible estimates must be based on measures not other estimates. This is coupled with the development lifecycle model assumed by MERMAID, ie a project is always regarded as a series of milestones separated by phases.Care must be taken not to confuse this use of the term phase with its use in a particular lifecycle model.

3.PROJECT METHODOLOGY

The MERMAID approach was systematic in determining what research to undertake and how to put the results of this research into practical use.
The basic steps of this process were:

* Understand the current state of the art and the user requirements in the area under investigation

* Analyse and undertake the formal research and develpoment into the subject area,subject to a peer review

* Provide a proof of concept vehicle(POCV);this was either in the form of a paper or an executable model.The quality level associated with such was lower than that required for the prototype toolsets.Some results of this process are discussed in the following section.The overall results of MERMAID emphasise the benefits of carrying out research and development in such collaborative projects.There was a mixture vital to producing tools and methods of practical use to European Software Engineering and often impossible to achieve in one organisation.

4. RESEARCH ACTIVITIES.

The project carried out an extensive programme of empirical work associated with the analysis of data made available to the project.One result of particular interest to the practitioner community was that function point counts are not independent.This work is reported fully in a number of Project Deliverables.(9)Additionally the research included the building of a number of models and proof of concept vehicles(POCV).If we consider resource modelling,the project examined 2 fundamentally different approaches, systems dynamics coupled with soft system methodology(10) and parametric modelling.(11)In the latter case a new model,KUNAMAA, was proposed and a POCV is currently being validated though this work is hampered by a lack of appropriate data.

5. MERMAID AND ESPRIT

The Commission of the European Communities (CEC) launched ESPRIT in 1983 largely as a consequence of the realisation in Europe that its Information Technology industry was becoming increasingly uncompetitive particularly viz-a-viz Japan. ESPRIT projects cover both hardware and software technologies and address both factory and office automation. Both software engineering and artificial intelligence are primary focuses. MERMAID is one of a number of projects, some current and others which have been completed, which address aspects of software development, management and metrics. Two immediate predecessors of MERMAID in which some of ideas were developed were IMPW(12) and REQUEST(13). The former developed an integrated project management toolkit which included support for cost estimation and latter was concerned with modelling aspects of software reliability.A new project (P6283),GOAL,intends to develop tools for managing multi-partner distributed systems projects.

ESPRIT has begun its third phase and today's projects have a clearer user requirement orientation than those in the first phase which were largely technology push projects. The change in emphasis followed early evaluations of the programme which showed a disappointing take-up of the research output by industry even in cases where industry itself was the prime mover behind the project.

6. THE IMMEDIATE FUTURE

Considerable effort has been devoted to developing a risk assessment capability which conforms to the MERMAID philosophy. Such a capability will require access to a knowledge base of previous projects and measures of risk to project budget and schedule will be estimated using similar statistical and analogical techniques to those used for effort estimation. This is presently available in a seperate tool,RiskTool. A further pair of estimation methods for use when there is a shortage of data describing past projects are provided;these are the Analogy-Based Estimation Method,suitable when there is between 4 and 10 projects and the Experience-Based Estimation Method for fewer than 4 projects.The ready availability of past and present project data is ensured by the use of another MERMAID developed tool,DCSS which supports the user in defining what they wish to collect with an extensive knowledge base of attributes and metrics.A commercial version of DCSS is now available.Whilst the exploitation plans of the partners in the MERMAID project are by no means clear it is likely that commercial versions of most if not all of the prototypes will become available during 1993.

LIST OF REFERENCES

1. De Marco, T (1982), Controlling Software Projects, Yourdon Press, New York
2. US Congress (1989), Bugs in the Program: Problems in Federal Government Computer Software Development and Regulation. Staff Study for the House of Representatives Committee on Science, Space and Technology.

3. Putnam L H (1987), A general empirical solution to the macro software sizing and estimation problem IEEE Trans. Software Engineering SE-4 (4)

4. Brooks F (1975), The Mythical Man Month, Addison-Wesley, London

5. Boehm, B W (1981) Software Engineering Economics, Prentice Hall, Englefield Cliffs, N J

6. Conte S, Dunsmore H and Shen V Y, (1986), Software Engineering Metrics and Models, Benjamin-Cummins, Menlo Park CA.

7. Kemerer C F (1987), An Emprical Validation of Software Cost Estimation Models, Comm, ACM 30(5).

8. Albrecht A J,Measuring Application Development Productivity Proceedings Joint SHARE/GUIDE Symposium,October 1979,pp83-92.

9. MERMAID Deliverables D3.3B,D3.3D and 3.3E,Analyses of MERMAID Data.

10. Bell G and Jenkins J (1991), Presentation to the 7th International COCOMO Users Group Meeting, SEI, Pittsburgh.

11. Harry, C M and Jenkins J (1992) Chapter in MERMAID Deliverable,D 5.8A.

12. Bosco M, Jenkins J and Verbruggen R (1987) Integrated Management Process Workbench(IMPW) Advance Papers,1st International Workshop on CASE,Cambridge,Mass.

13. Linkman S J (1990) Quantitative monitoring of software development by time based and intercheckpoint monitoring. Software Engineering Journal Vol 5(1).

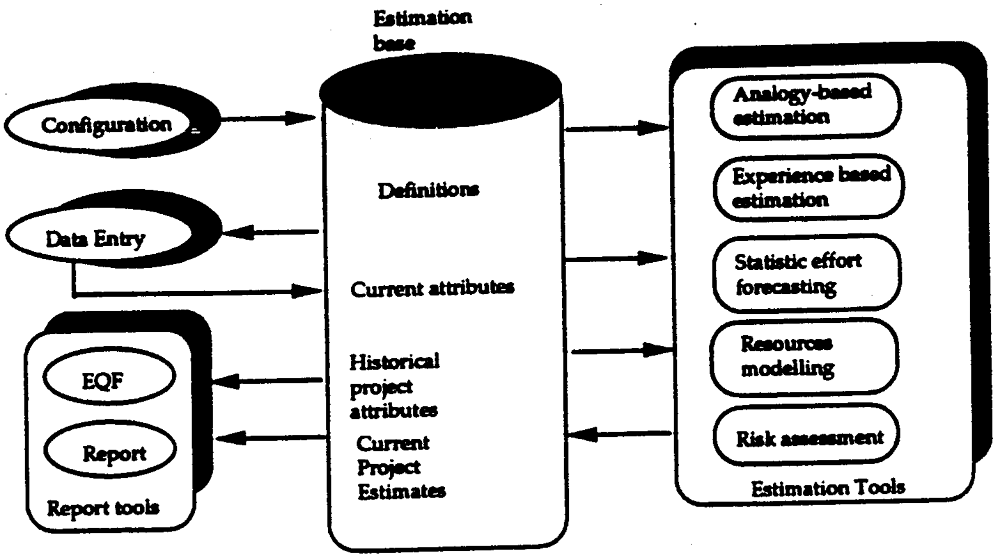
PROJECT PARTICIPANTS

Volmac Software Groep(Holland)
Data Management sPa (Italy)
National Computing Centre (UK)
City University(UK)
University College Cork(Ireland)
Valtion Teknillinen Tutkimuskeskus(Finland)

Details of all Esprit projects referred to in this paper can be obtained from Dr John Jenkins at the address above.

FIGURE 1

Mermaid Architecture



MERMAID

(the project)

Dr John O Jenkins

Department of Business Computing
City University,
London, England

Introduction

Issues,
a management view point

'There are three kinds of lies:
lies, damned lies and statistics.'
Benjamin Disraeli, 1804-1881

Estimates are often unbelievable

Missed opportunity because risks too difficult to manage

Unexpected risks consume the profits

Introduction

Structure of presentation

MERMAID, scope of approach

MERMAID, the project

Methods of estimation
and risk assessment

Conclusions

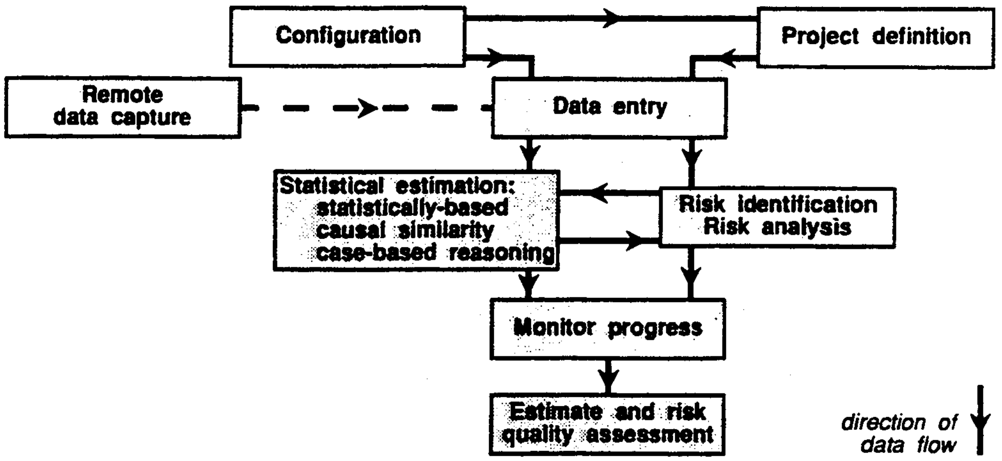
Scope of approach

MERMAID,
Scope of approach

Emphasis of:
integrated data storage
integrated concepts
integration within environment

Scope of approach

Extent of support: tools


Configuration

Project definition

Remote data capture

Data entry

Statistical estimation:
statistically-based
causal similarity
case-based reasoning

Risk identification
Risk analysis

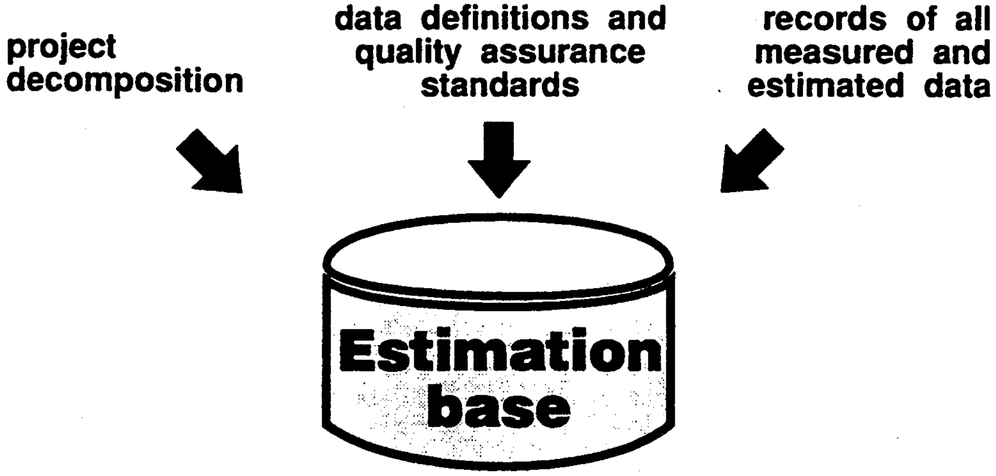
Monitor progress

Estimate and risk quality assessment

direction of data flow

Scope of approach

Extent of support: depth of detail


project decomposition

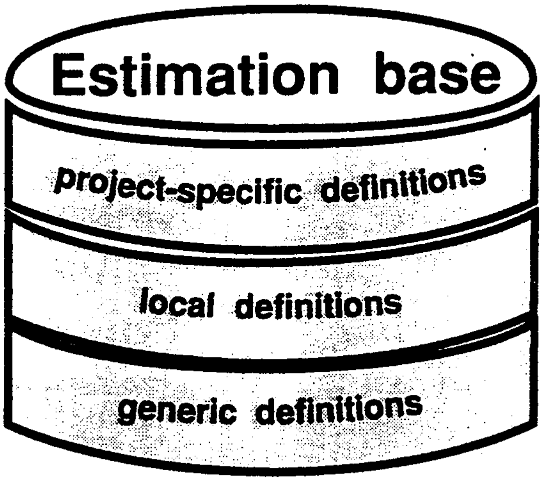
data definitions and quality assurance standards

records of all measured and estimated data

Estimation base

Scope of approach

Extent of support: integration with local environment



Estimation base

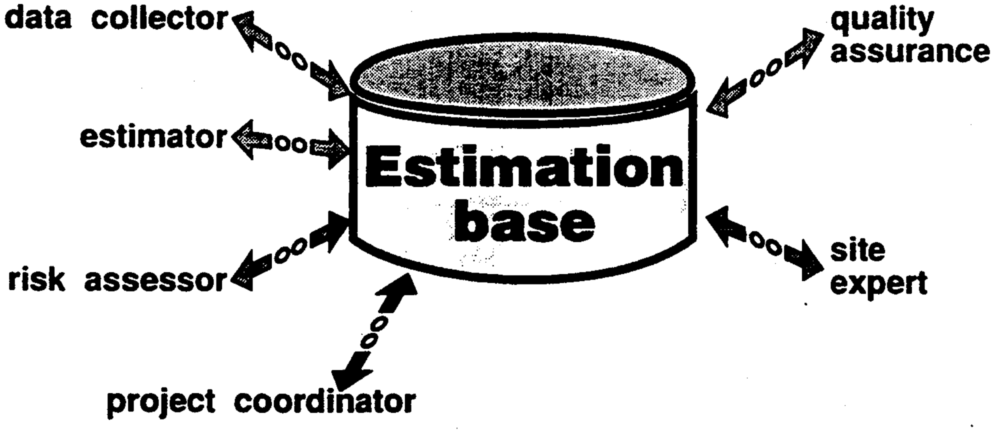
project-specific definitions

local definitions

generic definitions

Scope of approach

Extent of support: people



Estimation base

data collector

estimator

risk assessor

project coordinator

quality assurance

site expert

MERMAID, the project

A project partially funded by
the Commission for the European Community
as project 2046 within
the European Strategic Programme for Research in Information Technology.

Facts

Timescale

October 1988 to October 1992

Effort

48 person years

Partners

Volmac Nederland bv	(Utrecht, NL)
City University	(London, UK)
Data Management spa	(Pisa, I)
National Computer Centre	(Manchester, UK)
University College Cork	(Cork, IRL)
Valtion Teknillinen Tutkimuskeskus	(Helsinki, SF)

MERMAID, the project

Outputs

Research:
improved methods

Information dissemination

Toolset for Windows 3 on P/C

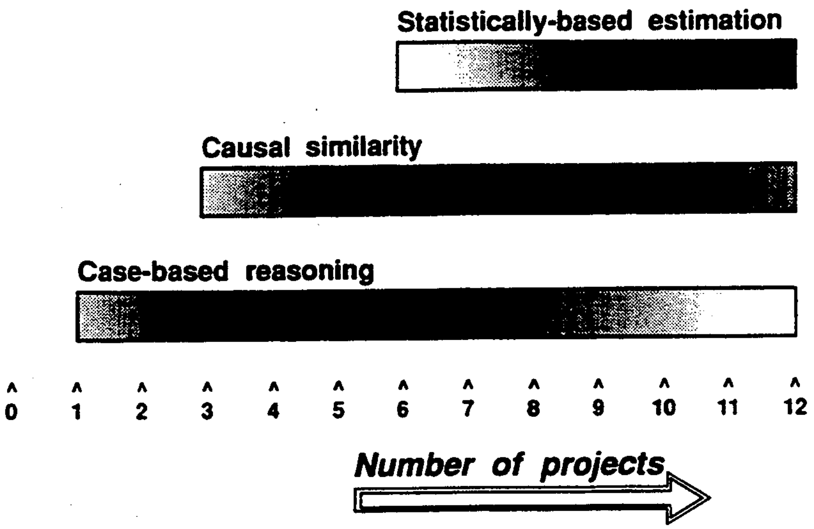
Version 1 (May 1992): initial set of features

Version 2 (Nov 1992): demonstration at ESPRIT Technical Week

Methods of ...

Methods of estimation and risk assessment.

Alternative methods of estimation



Statistically-based estimation

Derives a complete estimation equation from records of historic projects/tasks

Only effective cost-drivers are used:

Report of closeness of fit of equation to historic projects.

Skilled needed for:

selecting historic projects that are similar

identifying exceptions

Methods of ...

Causal similarity estimation method

Identifying the historic project or task
which is most similar
to the current project or task.

Skill needed for:

selecting the project attributes to be used.

Methods of ...

Case-based reasoning

Based on estimating the effect of the important differences between the current project or task and a case history.

Skill-based approach, supported by:

methods of quantifying this difference

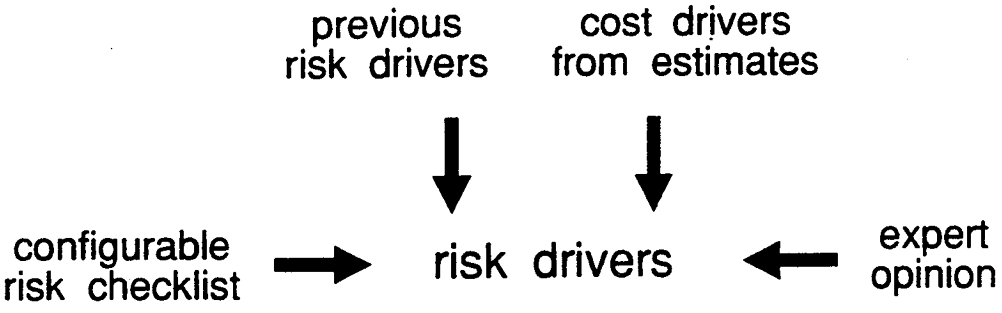
logical framework for recognising assumptions in the estimate

ability to identify and assess external risks

logical sequence for making an estimate

Methods of ...

Risk identification



previous risk drivers

cost drivers from estimates

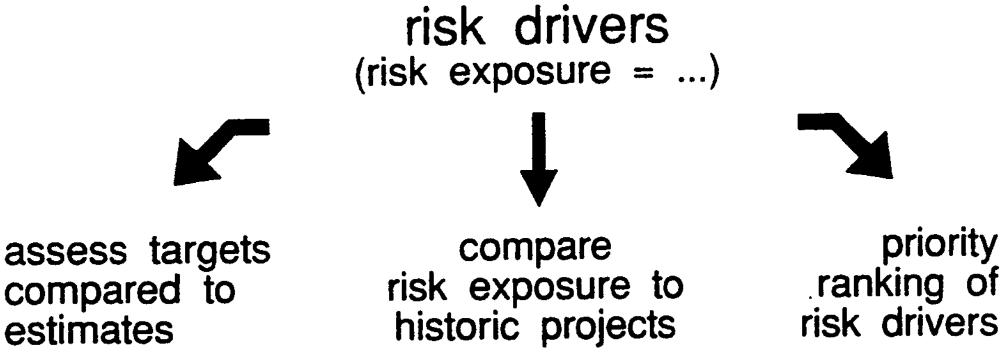
configurable risk checklist

risk drivers

expert opinion

Methods of ...

Risk analysis



risk drivers
(risk exposure = ...)

assess targets compared to estimates

compare risk exposure to historic projects

priority ranking of risk drivers

Methods of ...

Progress monitoring

Following milestones

... monitoring of targets

... new estimates made using newly available data

... revision of risk assessments

At regular intervals

... monitoring of risks which occur
against expected level of risks

Conclusions

Conclusions

MERMAID approach

Based on sound methodology

Extensive

Fully integrated

To help you with the way you work

and support changes and improvements in your work.