# *Using and Evaluating CASE Tools: From Software Engineering to Phenomenology*

Stephen F. King

A Thesis submitted for the degree of

Doctor of Philosophy

at the University of Warwick

January 1995

Warwick Business School

University of Warwick

Coventry, U.K.

# *Summary*

CASE (Computer-Aided Systems Engineering) is a recent addition to the long line of "silver bullets" that promise to transform information systems development, delivering new levels of quality and productivity. CASE is particularly intriguing because information systems (IS) practitioners spend their working lives applying information technology (IT) to other people's work, and now they are applying it to themselves. CASE research to date has been dominated by accounts of tool development, normative writings (for example practitioner success stories) and surveys recording IT specialists' *perceptions*. There have been very few in-depth studies of tool use, and very few attempts to quantify benefits, therefore the *essence* of the CASE process remains largely unexplored, and the views of stakeholders other than the IT specialists have yet to be heard.

The research presented here addresses these concerns by adopting a hybrid research approach combining *action research*, *grounded theory* and *phenomenology* and using both qualitative and quantitative data in order to tell the story of a system developer's experience in using CASE tools in three information systems projects for a major UK car manufacturer over a four year period. The author was the lead developer on all three projects. Action research is a *learning process*, the researcher is an explorer. At the start of this project it was assumed that the tools would be the focus of the work. As the research progressed it became evident that the tools were but part of a richer *organisational context* in which culture, politics, history, external initiatives and cognitive limitations played important roles. The author continued to record experiences and impressions of tool use in the project diary together with quality and productivity metrics. But the diary also became home to a story of organisational developments that had not originally been foreseen.

The principal contribution made by the work is to identity the narrow positivistic nature of CASE knowledge, and to show via the research stories the overwhelming importance of organisational context to systems development success and how the exploration of context is poorly supported by the tools. Sixteen further contributions are listed in the Conclusions to the thesis, including a major extension to Wynekoop and Conger's CASE research taxonomy, an identification of the potentially misleading nature of quantitative IS assessment and further evidence of the limitations of the "scientific" approach to systems development.

The thesis is completed by two proposals for further work. The first seeks to advance *IS theory* by developing further a number of emerging *process models* of IS development. The second seeks to advance *IS practice* by asking the question "How can CASE tools be used to stimulate awareness and debate about the effects of organisational context?", and outlines a programme of research in this area.

# *Table of Contents*

## Part 1 - Introduction

## Part 2 - CarMaker Research

## Part 3 - Contribution

## Appendices

*To Helen*

*and Bethan*

# *Acknowledgements*

The story told in this thesis represents four years of intensive learning by the author. An action research approach was adopted, which meant that the author had the opportunity to assume the dual role of researcher and practitioner in order to evaluate the new technology of Computer-Aided Systems Engineering (CASE). Other research approaches allow the researcher to observe from a distance, and not to become involved in the messy reality of organisations. Action research forces the researcher to confront the reality of technology, people and organisations in the pursuit of knowledge. Over the four years the author worked alongside many people, building computer-based information systems for a major UK car manufacturer, and evaluating the CASE tools in the process. What started out as a technology study soon became a socio-political study as well, as the limitations of the technology in the face of the complex organisational environment became evident. The research topic expanded rapidly and became far more demanding as the need to understand culture, politics and cognitive limitations overtook the need to understand the tools.

The final story is far richer for this change of emphasis, but in order to tell the whole story I have had to change the names of the people, the organisations and the tools with which I have worked. The unfortunate consequence of this is that I cannot acknowledge by name the many individuals who helped me in this work. They will remain anonymous, but my thanks goes to them for the unique opportunity that I have had to experience the excitement and emotion of *action* and *research* rolled into one.

One person that I can name is my supervisor, Professor Bob Galliers of the Warwick Business School. Bob's enthusiasm for information systems research is infectious, he somehow finds time in a breath-taking international schedule to encourage tentative ideas to bloom. He's also a fun guy, thanks for your help Bob.

# *Declaration*

The work described herein is entirely the original work of the author. Two publications have arisen from this work to date, abstracts of the publications are presented in Appendix 1.

# *List of Figures*

# *List of Tables*

# *List of Abbreviations*

| | |
|---|---|
| 3GL | Third Generation Language |
| 4GL | Fourth Generation Language |
| AMI | Application of Metrics in Industry |
| BCS | British Computer Society |
| BoM | Bill of Material |
| CAD/CAM | Computer Aided Design / Computer Aided Manufacturing |
| CASE | Computer Aided Systems Engineering |
| CBIS | Computer-Based Information System |
| CSF | Critical Success Factor |
| DD | Data Dictionary |
| DFD | Data Flow Diagram |
| DPM | Data Processing Manager |
| DOS | Disk Operating System |
| ERM | Entity-Relationship Model |
| EUC | End-User Computing |
| FP | Function Point |
| ICASE | Integrated CASE |
| IFIP | International Federation of Information Processing |
| ISO | International Standards Organisation |
| IPSE | Integrated Project / Programming Support Environment |
| ISD | Information Systems Development |
| JAD | Joint Application Development |

| | |
|---|---|
| LOC | Line of Code |
| MIS | Management Information Systems |
| MRP | Manufacturing Requirements Planning |
| PCA | Product Change Authorization |
| PSG | Product Support Group |
| PSR | Problem Status Report |
| SDV | Special Development Vehicle |
| SISP | Strategic Information Systems Planning |
| SLOC | Source Line of Code |
| SQL | Structured Query Language |
| SSADM | Structured Systems Analysis and Design Methodology |
| UDI | Unilateral Declaration of Independence |
| UIS | User Information Satisfaction |
| VPG | Vehicle Part Group |

# *Prologue*

"Assessment is not a simple task because the relationship between technical solutions and organizational consequences is not well understood. Successes are frequently sheer luck, and failures are often due to our lack of ability to foresee the organizational consequences of technical solutions and/or because the chosen technical solutions do not have the anticipated results." (Bjorn-Andersen and Davis, 1988, p. v)

"The acceptance of 'academic' [IS development] methods in practice is low and, in general, the rate of transfer of research results and 'know-how' from scientific research to industry is embarrassingly slow." (Bubenko cited in Avison and Fitzgerald, 1988, p. 275)

# *Part 1*

# *Introduction*

# *Chapter 1*

# *Introduction*

## 1.1 CASE *Zeitgeist*

Software is big business. The demand for software-based products and systems is increasing inexorably as the microprocessor colonises everything: domestic appliances, entertainment systems, telephones, motor cars, aeroplanes, offices, factories and schools. As the demand for flexible and high-functionality systems increases so does the demand for the software that controls these systems.

> *"Demands for greater functionality and integration of applications call for larger and larger programs"* (Swanson *et al.*, 1991, p. 566).

Swanson *et al.* go on to report that the average size of software systems has risen from 23,000 lines of code in 1980 to 1,246,000 lines of code in 1990. As the demand for software increases, so does the pressure on software and system developers. The information technology community is notorious for reaching for Silver Bullets (Brooks, 1987), miracle cures that will dramatically transform software development providing orders of magnitude improvements in cost, quality and productivity. From the mid-1980s the latest in a succession of silver bullets captured the profession's imagination: CASE (Price Waterhouse, 1989, Holloway and Bidgood, 1991).

> *"CASE is the latest in a long line of techniques to solve the timescale, quality and staff shortage problems of computer systems development."* (Price Waterhouse, 1989, p. 17)

> *"Unlike other panaceas, CASE will not go away, since it provides the framework that embraces them all."* (Price Waterhouse, 1989, p. 21)

Expectations of CASE ran high in 1989, CASE dominated the pages of the computing press and was the subject of numerous marketing campaigns as CASE tool vendors scrambled to exploit the new panacea. The CASE *zeitgeist* was one of excitement at the possibility of doing to systems development what systems developers had been doing to everyone else for years: automating the work. Since there were staff shortages, there was little fear of job losses, instead there was excitement at the opportunity to use advanced IT to develop IT and, in doing so, increase the skill levels of the profession so that everyone could become analysts.

## The British Computer Society CASE Conferences

The mood of the time was summarised by the type of paper presented at two British Computer Society CASE conferences in the early 1990s (Spurr and Layzell, 1990 and 1992), where the majority of papers presented either new tool developments, such as object-oriented repositories or knowledge-based advisors, or practitioner success stories. The occasional doubting paper usually concluded that better technology or better methods would resolve the problems encountered.

The research programme described in this thesis investigates the reality of using state-of-the-art CASE tools, to see if the claimed improvements in system development performance and the unquestioning faith in technology are justified.

## 1.2 What is CASE?

There is no agreed definition of CASE, although most subscribe to a small group of concepts. Firstly, the acronym CASE has two common interpretations. CASE is Computer-Aided Software Engineering if your focus is software engineering, or Computer-Aided Systems Engineering if your focus is information systems (IS) development. This distinction is helpful, in that it highlights the centrality of the purpose for which the tools are used. Software engineering has the objective of creating software for embedding in a deterministic system, such as a washing machine, or a motor car engine, or a telephone exchange. The environment in which the software is to function can be defined, and the range of inputs and outputs prescribed. In contrast, information systems development attempts to embed software in a human activity system (Checkland, 1981), which is a combination of software, hardware, and human beings - who are non-deterministic! The challenge of software engineering is to produce a highly-reliable product that performs as expected in potentially hostile environments, and fails safe if the unexpected occurs. The challenge of information systems development is to develop a reliable software product that performs as expected, but which must satisfy the potentially differing expectations of many customers. Furthermore, these expectations change over time and often cannot be predicted. Usability, flexibility and maintainability therefore join reliability as critical success factors. As this study explores information system development, CASE will be interpreted as "Systems Engineering".

Although there is no single definition of the components of CASE, the Price Waterhouse (1989) definition is representative of the majority:

> *"For this year's answer to life, the universe and everything, as far as DPMs are concerned, is CASE ... CASE, we are told, will do for computing what CAD/CAM did for manufacturing. It will automate much of the task of production, (the programmer's job), and add tremendous power to the designer's elbow (that of the systems analyst)."* (p. 17)

The report goes on to define three CASE components:

1. Upper CASE: tools for automating systems analysis.

   Upper CASE tools are often called Analyst Workbenches. They typically comprise of a graphical user interface that supports the creation and maintenance of logical models of the information system being developed (typically data flow diagrams and entity-relationship models). In addition to the diagrams further data is stored including narrative descriptions of processes and data, data element definition, screen designs and report layouts. Both the diagrams and the additional data are stored in a *repository*, or central database. Many upper CASE tools provide interfaces to other software products, such as lower CASE tools and word processors.

2. Lower CASE: tools for automating programming.

   The term Lower CASE tool typically encompasses fourth-generation languages (4GLs), relational database systems with query languages and application/code generators. The tools accept input in a higher level "language" than third generation programming languages and either execute the instructions internally, or generate third generation code for subsequent execution. Input may be in the form of high-level program statements or by the user responding to prompts or filling in pre-defined fields presented by the 4GL.

3. IPSE (Integrated Programming/Project Support Environment): tools for automating management of systems development.

This study focuses on upper CASE tools, lower CASE tools and integrated tools (ICASE), which comprise of upper and lower CASE tools which can share data. IPSEs will not be discussed further, apart from a brief review of the insightful study by Land *et al.* (1992).

## 1.3 Why is CASE Important?

Thompson (1990, p.18) identifies four categories of problems evident in software and systems development that result in an often poor productivity and quality record.

"1. A failure to understand requirements.

From various viewpoints ...

Business: the system does not address the business problem.

Service: system performance, reliability or availability is poor.

User: the system is difficult to learn and use.

Software: contains bugs.

Package: key parts of the documentation are poor or missing.

2. An under-estimation of effort required for development.

(or productivity over-estimation) resulting in ...

Cost overrun: the system costs more than forecast.

Time overrun: the system is delivered late.

3. An inadequate quality control mechanism.

resulting in ...

Variability: in the standard of deliverables.

4. An inadequate consideration of the impact of change.

causing difficulties in ...

Support: difficult to know what parts of the system actually do.

Enhancement: too much effort to add new features.

Correction: of requirements analysis and design errors.

Adaption: hard to "port" to new hardware and software platforms.

Documentation: analysis and design documentation not maintainable."

CASE tools aim to address many of these problems, for example by facilitating the use of more intuitive and comprehensive analysis techniques (e.g. data flow diagrams (DFDs) and entity-relationship models (ERMs)); checking the consistency and completeness of the data stored in the repository; co-ordinating the work of multiple developers; automating part, if not all, of the programming activity; enabling the impact of changes to be assessed prior to the change and automatically propagating changes throughout the system.

## Productivity and Quality

If CASE has many interpretations, productivity and quality are equally elusive. Improvements in system development productivity and quality are the most frequently cited objectives for CASE introduction. By addressing the kinds of problem identified above, the hope is that system development costs will fall, systems will be delivered on time and within budget, and that the delivered systems will satisfy customer requirements.

A widely used definition of productivity is the amount of system developed per unit of development effort (Boehm, 1981, Fenton, 1991 and Symons, 1991) i.e.

*productivity = system size / development effort.*

Quality is more problematical, although an International Standards Organisation (ISO) definition exists for the term (ISO, 1986):

> *"Quality: The totality of features and characteristics of a product or service that bear on its ability to satisfy stated or implied needs".*

Gilb (1988) defines system quality in terms of quality attributes, which are quantifiable measures defined by the user. A high quality system is one that meets its quality attribute targets. Gilb identifies four generic classes of quality attribute:

- workability

  (process capacity, storage capacity, responsiveness and other related measures)

- availability

  (reliability, maintainability and integrity)

- adaptability

  (improvability, extendability and portability)

- usability

  (entry requirement, learning requirement, handling ability and likability).

Fenton (1991) supports Gilb's comprehensive approach, but observes that software quality data is often difficult to collect. He proposes a pragmatic subset of readily-collectable metrics embracing reliability, usability and maintainability, foremost of which is reliability. Fenton proposes the following single, simple measure of system quality:

*quality = number of defects / system size.*

Complimenting these "hard" quality metrics, attempts have been made to measure the softer concept of user satisfaction, which accords with the ISO definition of quality. A number of questionnaires have been developed to measure user satisfaction (Bailey and Pearson 1983, Ives *et al.*, 1983, Baroudi and Orlikowski, 1988, QA Forum 1989 and Barki and Hartwick 1993). Holistic assessment was taken a stage further at the IFIP 8.2 conference on IS assessment in 1986 (Bjorn-Andersen and Davis, 1988) where the participants recognised the essentially subjective nature of IS assessment and the importance of stakeholder perspective.

## 1.4 Summary of Terminology

To summarise, for the purposes of this study CASE is defined as Computer-Aided Systems Engineering, the tools of interest being upper, lower and integrated CASE tools and the objective of CASE is to improve information systems development quality and productivity. Productivity will be measured in terms of system size delivered per work-month; quality will be measured both quantitatively and qualitatively: the former by dividing defects by system size, the latter by conducting post-project stakeholder interviews[1]. The origins of the CASE research programme described in this thesis are described next.

[1] A user satisfaction survey replaced the stakeholder interviews for the third CASE project presented in this thesis.

## 1.5 Genesis of the Research[2]

The author was interviewed for a research post at the Research Centre, University of Hilltown in October 1989. The Research Centre was funded by the CarMaker car manufacturing group, and was a high-profile organisation at the forefront of industry-academic collaboration. The major area of research was in advanced engineering technology. The Research Centre also provided a home for the CarMaker IT Strategy team, who were tasked with developing and implementing a company-wide IT strategy in conjunction with the business units. To complement the work of the IT Strategy team, two new IT research fellows were recruited to bring the IT research fellow compliment up to four. The author was one of the new recruits.

There was no formal job description, only a vague notion of supporting the IT Strategy team as required. The author was excited by the prospect of working on significant IT projects for a major UK company and having the opportunity to learn how high-level IT issues were addressed in practice. The research fellowship also offered the opportunity to study for a PhD. This would enable the author to continue his academic career (he was previously a lecturer in computer science at another UK university and had bachelors and masters degrees in computer science), whilst gaining valuable industrial experience.

The author joined the Research Centre in February 1990. A few weeks earlier a consultant from a CASE tool vendor, ToolVendor, had approached the Director of the Research Centre with a proposal for a programme of collaborative research

[2] To enable the full story to be told, the names of the organisations and people involved in the research and the tools used have been changed.

between ToolVendor, the Research Centre, and CarMaker. The aim of the programme was to evaluate the impact of the vendor's integrated CASE toolset, consisting of the upper CASE tool Analyser, and the lower CASE tool FourthGen, on system development quality and productivity. Clearly ToolVendor was motivated by the opportunity to gain a marketing edge over its competitors by showing that the claimed benefits of its tools were supported by objective university research, and by the expectation that the research collaboration with CarMaker would eventually lead to sales of the tools to the car manufacturer. In return, the Research Centre was willing to provide staff resources to support the project in the expectation that the work would benefit CarMaker and add another well-known industrial company to its list of collaborators, and CarMaker was happy to accept the offer of free consultancy from ToolVendor and the prospect of some low-cost systems development for the business units. If the tools delivered business benefits so much the better.

During February the CarMaker IT Strategy Director spoke to the business units and identified two potential systems development projects: the New Model project and the Warranty project. In the meantime the author was sent on CASE tool training courses. On 19 March 1990 the author and the ToolVendor consultant had their first meeting with a manager from the Interior Trim department at CarMaker's Design Centre. The first of the three CASE projects had begun[3].

[3] In addition to the two projects identified initially, a third CASE project, the vehicle recycling project, was initiated some time later.

# 1.6 Research Details

## Research Question

The aim of the work was to evaluate CASE benefits, particularly impact on quality and productivity. Although not formally stated, the research question became:

> *"Do CASE tools enable quality systems to be developed quickly?"*

## Research Approach

The research was motivated by the Consultant's concern to counter the backlash forming against CASE "hype" by providing objective evidence of benefits. The Consultant's initial proposal to the Research Centre represented a timely window of opportunity, enabling academics and practitioners to work together to address these issues. In this respect the work benefitted greatly from being motivated by a genuine research problem that was of concern to both practitioner and academic communities.

There was no theory to guide data collection, instead a *hybrid* research approach evolved as the research progressed. We were guided by a shared belief that data on effort, stakeholder participation, activity descriptions and problems/thoughts would form a rich resource for subsequent analysis. In this sense, the research approach was *grounded theory* (Eisenhardt, 1989 and Strauss and Corbin, 1990), the intention being to experience tool usage and systems development first and through that experience, to allow important issues to emerge. Secondly, the research approach was *action research* (Checkland, 1981 and Avison and Wood-Harper, 1990), in that the author and the Consultant were acting as both researchers and practitioners, using the tools themselves in order to study tool usage in a real organisational setting. Thirdly, the work represented a *phenomenological* study (Boland, 1987 and Boland and Day, 1989) in that the researchers kept a diary into which were written thoughts, problems and issues as the research progressed. The diary provided the major source of material for subsequent analysis and was supplemented by other project documentation, including minutes of meetings, progress reports, defect reports, change requests, data flow diagrams, data models and the output from the computer systems developed as a result of the work. The analysis of the documents has been undertaken by the author as a *hermeneutic* study (Boland and Day, 1989) in chapter 5; major events and their consequences were identified and classified then revisited later and refined until the essence of the author's experience of being an analyst and using the tools was uncovered. The essence of experience has been presented as a set of lessons in chapter 5.

## CASE Tools Used

### Upper CASE

The ToolVendor upper CASE tool Analyser was used for analysis and design. The tool supported the structured techniques of data flow diagramming, entity-relationship modelling and enabled prototype systems consistıng of menus, screens and report samples to be developed for demonstration purposes. The tool could generate a file containing data store specifications, screen layouts and report layouts which could be read into the ToolVendor lower CASE tool, FourthGen. The tools therefore constituted an integrated toolset.

Lower CASE

FourthGen was a fourth generation *environment*, as opposed to a 4GL. The user created an application by "filling in the boxes", but had to add 3GL-like logic strings for any special-purpose processing.

Analyser and FourthGen were used on the first project, only Analyser was used on the second (a system was not developed), whilst FourthGen was replaced by the relational database, QuickStore for the third project. Analyser was used for analysis on project three, whilst the NumberCrunch spreadsheet was used to generate graphical output from QuickStore data. None of the tools used on the third project were integrated, although NumberCrunch could read QuickStore export files.

## 1.7 Thesis Structure and Evolution

The thesis is divided into three parts. The first part is the introduction (Chapter 1). The second part describes the three CarMaker research projects and the quality and productivity data collected (Chapters 2,3 and 4 respectively) and concludes with a number of lessons learned from the projects (Chapter 5). The third part identifies the contribution to knowledge made by this work by reviewing the CASE, IS development, IS assessment and IS research methodology literatures (Chapter 6) and then grounding the results of the action research in the literature (Chapter 7). The final chapter (Chapter 8) reflects on the grounded practice and concludes with a summary of the contributions made by the work and presents two proposals for further work. Figure 1.1 shows the thesis structure as a process map, with the "products" of each chapter (process) shown as outputs and, where appropriate, source material from preceding chapters shown as inputs.

The structure of the thesis is somewhat unusual in that the literature survey (Chapter 6) follows the empirical research rather than providing the foundation for the empirical research. This is because the research programme was instigated by *practitioners* (the ToolVendor consultant and the CarMaker IT Strategy Director) interested in seeing CASE hype replaced by objective research. The research represented a window of opportunity for the author to become involved in a *relevant* piece of work at a time when CASE was a high-profile topic. As Keen (1991) states, relevance should come before rigour, rigour without relevance is irrelevant. The rigour has been added subsequently, by grounding the CarMaker results in the literature (Chapters 6 and 7), and has served to enhance the relevance of the study.

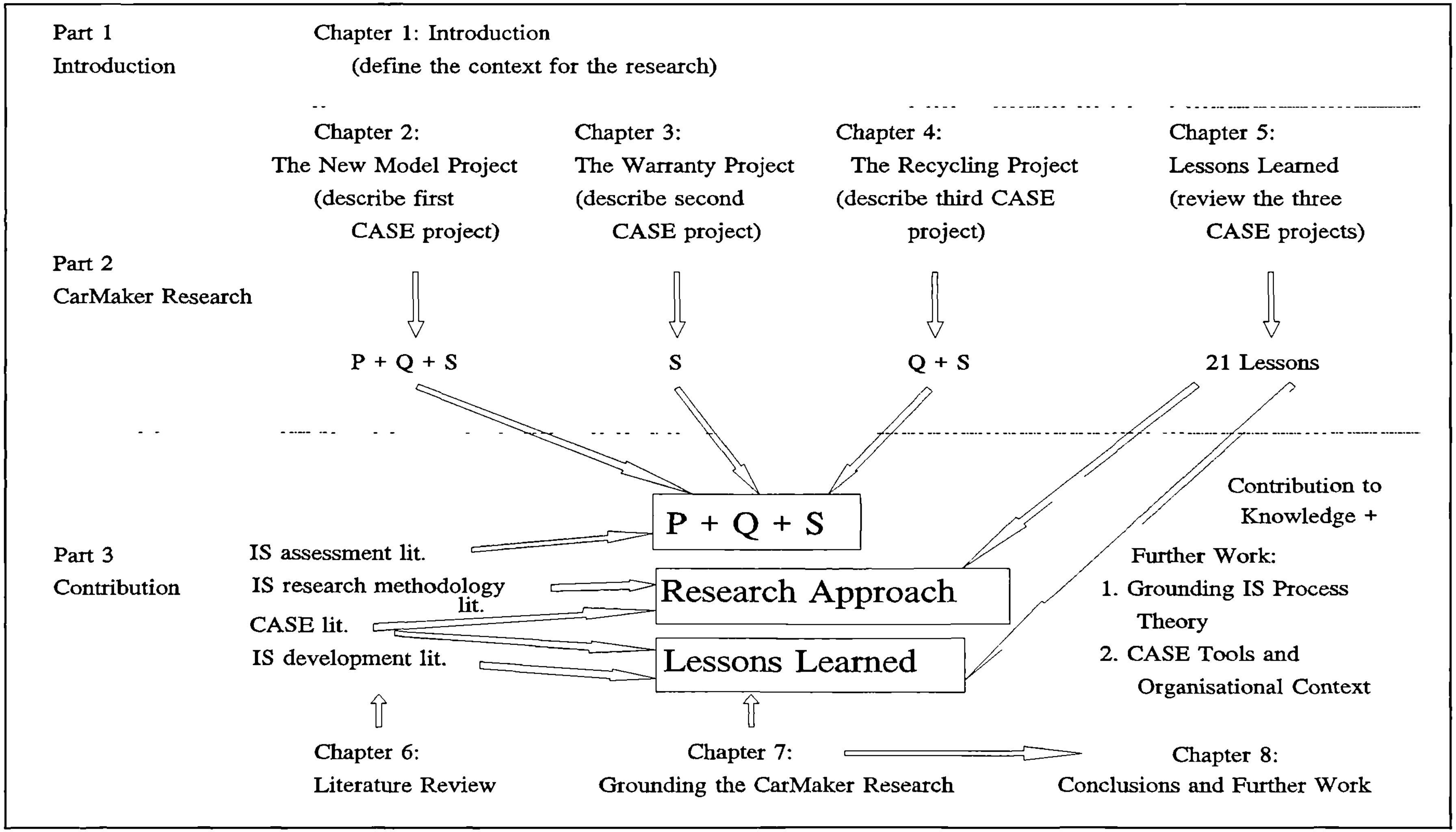
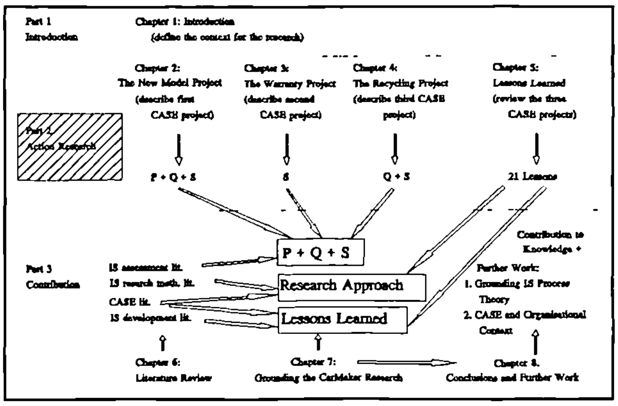


**Figure 1.1** Process Map of Thesis

**Key:** P = Productivity Metrics
Q = Quality Metrics
S = Stakeholder Views

# *Part 2*

# *CarMaker Research*



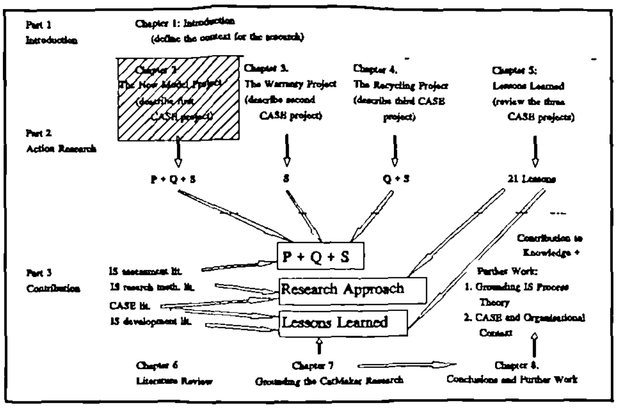
## Introduction to Part 2

In Part 2 the three CASE research studies are presented. The first study, the New Model project, ran from February 1990 to May 1991. The second study, the Warranty project ran from May 1990 to January 1991, therefore running in parallel with the first project. The two projects represented a collaboration between the Research Centre, ToolVendor and CarMaker, and used the two ToolVendor CASE tools: Analyser (upper CASE) and FourthGen (lower CASE). After the end of the two projects, the author and the ToolVendor consultant attempted to expand their work to form a Club of UK CASE using organisations. Funding for the research was sought from the UK government but the bid was not successful. This ended ToolVendor's involvement in the research programme. In the meantime, CarMaker had secured funding from the government for an investigation into vehicle recycling in partnership with a large UK recycling firm, Recyclate Ltd. The recycling project aimed to develop a cost-effective recycling process and to identify improvements required in the UK vehicle recycling industry as well as to advise on appropriate government legislation to promote recycling. The project was to create a prototype recycling factory at the Recyclate Ltd. site. The author was assigned the task of developing an information system for the project which would advise on the most cost-effective method for dismantling cars given the current market price of recycled materials. The project started in November 1991, with the author's involvement commencing in earnest in May 1992. The author chose to use the recycling project to continue his investigation into CASE tool effectiveness, using the ToolVendor upper CASE tool, but replacing the ToolVendor lower CASE tool with a relational database product, QuickStore, in use at Recyclate Ltd.

The following three chapters describe the three CASE projects in the form of a story. The stakeholders (or actors) are introduced at the start of each story, and the story unfolds by way of a series of events as experienced by the author. The aim is to explain what happened and why it happened in the author's own words. The stories combine detail of use of the technology with an appreciation of the personalities and organisational context of action. This is, therefore, a *phenomenological* account (Boland and Day, 1989), the subjective nature of the writing contributing to the realism of the work; the author is not attempting to interpret someone else's experience. To counterbalance the stories, productivity and quality metrics are presented where available. Finally, each project description is concluded with a summary of the views of the other stakeholders in order to triangulate the findings.

# *Chapter 2*

# *Study 1 - The New Model Project: The "Technical" Project*



## 2.1 Introduction

The aim of this project was to improve the New Model Centre problem management process at the South Factory and West Factory manufacturing plants. The project entailed the development of a Design Problem Tracking System for the pre-production build phases of CarMaker new model development. The system was intended to replace an existing spreadsheet-based system which was seen as very difficult to use, slow and storing minimal data. The new system was to improve the tracking of problems from identification to resolution and to provide various problem status analyses. Problems were entered into the system from Problem Status Report (PSR) forms raised by engineers. The system was developed on a PC at the University and subsequently implemented on a VAX at CarMaker. The project started in February

1990 and finished in May 1991. The information system was developed by the author and the consultant from ToolVendor using Analyser and FourthGen.

## 2.2 *The Stakeholders*

*The Author*: Research Centre Research Fellow

Initially highly committed and enthusiastic about the project. Subsequently frustrated by FourthGen inflexibility and complexity and confused by CarMaker IT Strategy team apathy. Proud of technical skills; overwhelmed by New Model system maintenance workload.

*The Consultant*: ToolVendor Consultant

A proactive salesman and strong personality. Driven by a professional vision of tool possibilities, technically capable but eventually lost interest in project as prospect of sales to CarMaker lessened. Diverted onto another project by ToolVendor.

*The Research Manager*: Research Centre Research Manager

An individual who liked to dabble in new technology. He was involved in setting up the ToolVendor collaboration but played little further part in the projects. Later left the Research Centre.

*The Sponsor*: CarMaker New Model Centre Manager and Project Sponsor

Experienced manufacturing manager, low level of IT understanding. Over-busy elsewhere with the imminent launch of the new Executive Car. Trusting in IT possibilities and Research Centre reputation; not closely involved.

*Tom*: West Factory New Model Manager

Gentle, pleasant, technically and politically cautious. Kept his distance after initial meetings. Co-operative when asked.

*David*: West Factory Spreadsheet New Model System Developer

Ex-YTS trainee, technically competent and self-taught in IT. Trusted by West Factory colleagues, aware of political developments and frustrated by SystemsHouse costs and poor service. Co-operative when asked but kept his distance.

*Andrew*: South Factory New Model Engineer

Trusting in IT and Research Centre reputation. Polite, honest and friendly. Bright but new to IT projects. Eventually let-down after championing the Research Centre's cause.

*Stuart*: South Factory New Model Manager

Forceful, energetic young manager. Busy elsewhere, disappointed and frustrated by FourthGen New Model system.

## 2.3 *The Story*

**Note**: CASE issues are highlighted using a small font in the Story text.

*The Honeymoon Period*

January 1990. The Research Manager raises project proposal to investigate CASE tools with ToolVendor.

1/2/90. The author joins the Research Centre.

5/2/90. The Consultant presents his "Collaboration Proposal" to Research Centre and CarMaker staff.

19 to 22/2/90. The author attends Analyser training course.

2/3/90. The Consultant writes to the Research Manager with initial timescales for the first project.

12 to 14/3/90. The author attends FourthGen training course.

FourthGen is extremely difficult to use and has a "hidden" architecture that the instructors find hard to explain.

19/3/90. First meeting with CarMaker "customers".

The author and the Consultant meet Bill Green, Chief Engineer, Interior Trim Family Cars, based at CarMaker's Design Centre. Bill sees a need for an improved IS to support problem tracking. There is a need to speed up problem analysis, share data, reduce anomalies and provide statistical and exception reports. The manual New Model forms would be retained to provide sketches of problems, with a computer-based problem index for reporting. Bill suggests we talk to the Sponsor at West Factory as a potential CarMaker project owner, and also that we view the computerised Warranty problem register as well. Bill introduces many engineering terms new to the author. The author relies on the Consultant, who has an extensive manufacturing background, for interpretation.

20/3/90. The author attempts to follow ToolVendor MiniMethod methodology.

Due to lack of experience, the author attempts to work by the rule book by following the phases and activities defined by the ToolVendor IS development methodology. The Consultant prefers to take short cuts based on experience, arguing that the project is small-scale and the methodology pedantic. The author concedes. The author and the Consultant draw an initial DFD and ERM, based on Bill Green's comments, by hand then enter them into Analyser. The diagram editor is tedious to use and some difficulties are experienced in getting the software to print out the diagrams. Documentation is plentiful but superficial.

26/3/90. Second interview with Bill Green at CarMaker's Design Centre.

Review diagrams with Bill Green who agrees that they are a correct picture of current processes and data. Bill's objectives are rather vague and are not quantified. Agree that the new computer system will address a subset of the processes - namely those that are easy to computerise! No attempt is made to change existing processes substantially.

28/3/90. First meeting with West Factory staff and project sponsor.

The Sponsor, David, Tom and some others are present. David demonstrates his spreadsheet-based New Model register. The author and the Consultant present work to date and table the DFD and ERM agreed by Bill Green. The West Factory staff agree that the diagrams accurately represent their system, even though the ERM proves to be inaccurate subsequently. The Sponsor states his objectives, which again are somewhat vague and not quantified. He notes that other areas may wish to add to the requirements at a later date but thinks this is not important at present. The author and the Consultant collect examples of system outputs and New Model forms from David and Tom for further analysis.

Three further meetings take place at West Factory over the next week in order to better define the data model and data definitions. The focus is very much one of modelling existing data in order to computerise it rather than attempting to improve working practices. The assumption being that a properly structured database will automatically deliver the expected system benefits.

Diagrams are printed from Analyser prior to each meeting; tabled at the meeting; manually corrected; and updated in Analyser on return to the Research Centre.

30/3/90. The author and the Consultant validate data model with CarMaker Corporate Data Modelling Staff.

6/4 to 22/4/90. The author goes on holiday.

23/4/90. The Consultant issues the Project Determination Report.

Having recorded all of the process and data definitions in the Analyser repository, the Consultant is forced to retype the data into an Apple Macintosh in order to produce the Project Determination Report because of the poor word processing and reporting facilities in Analyser.

### *Using the Tools in Anger*

26/4/90. Evaluate the Analyser/FourthGen interface to identify what data can be transferred between the tools.

The author and the Consultant attempt to build some simple screen-based "test" functions in FourthGen in order to gain some experience with the tool. The FourthGen user interface is text-based and rather old-fashioned (e.g. there is no mouse support). On many data entry screens it is impossible to go back to correct mistakes, instead the whole screen has to be filled and then exited and re-entered in order to return to the point of error. This slows development considerably. The documentation is voluminous and written for technicians with few examples of how the myriad of features could be used in practice. In particular, the "FourthGen Timing Cycle" flowcharts are impenetrable, forcing the author and the Consultant to proceed by trial and error. The flowcharts show at what stage in data processing FourthGen reads or writes to file. Unlike a 3GL, the user cannot simply issue a "read" command at will, but must predict what FourthGen is doing and place the appropriate code in the right "slot" in the cycle.

Screen handling is similarly governed by undisclosed rules which are counter-intuitive. In particular, the paging screens that enable many records to be viewed in a scrolling window require careful programming in order to read from file.

Meanwhile, the Analyser/FourthGen interface suffers from equally poor documentation. The Consultant admits that he is using the project to enable ToolVendor to learn for the first time how the tools fit together. "Design Units" are the items transferred between the tools, but these are not defined in the documentation. After further experimentation, the design unit is found to contain one incoming data flow from an external entity to a process and any number of data stores associated with that process. Any screen "painted" onto the data flow is transferred to FourthGen together with each data store and store contents, which become files and fields respectively. The interface cannot handle composite data flows or data stores, therefore a rule is adopted that DFDs must be exploded down to the lowest level before design units are created.

A further limitation of the Analyser diagram editor is discovered in that data store contents cannot be copied into data flow contents and vice versa. The user must therefore sometimes retype long lists of attributes, with the associated danger of mistyping attribute names thereby inadvertently creating new attributes!

9/5/90. Rough sketching of screens and reports at West Factory.

Screen and report "images" are attached to the appropriate data flows using the Analyser Image Painter. On selecting a data flow, all of the attributes are automatically posted to the image painter's scratchpad. The attributes can then be added to the screen or report as desired. This is a useful facility but is undermined by the eccentric use of function keys to operate the image painter. It is quite easy to lose work by accidentally hitting the wrong key. Furthermore, new fields added via the image painter are not automatically added to the data flow, therefore the data flow and the screen image can easily become unsynchronised.

*Enter the South Factory Staff*

15 5/90. Demonstration of Analyser-based New Model system prototype at the Research Centre.

By this time the Sponsor had relocated to South Factory to set up the New Model Centre for the new Executive Car launch. He brought with him a member of the South Factory staff, Andrew, who had not been involved in the New Model project previously. The prototype demonstration was very successful; the users made a number of valuable suggestions for enhancements and were pleased with the work to date.

22/5/90. The Consultant issues the Requirements Specification Report.

23/5/90. Implementing the Analyser prototype in FourthGen.

The author and the Consultant started with a very simple process: *Maintain Launch Team.* This process was shown on the DFD for the proposed system and added/changed/deleted and displayed data from the *Launch Team* data store. The aim was to show a screen-full of records at a time, therefore a FourthGen "paging screen" was required. The screen image and data store contents were exported easily from Analyser; but considerable work had to be done in FourthGen to get the paging screen to show more than one record at a time. Progress was hampered by the lack of a screen painter in FourthGen, which meant that x,y co-ordinates for every new or changed piece of text or data field on the screen had to be calculated manually and typed in. Furthermore the "one way" cycle through the FourthGen function screens described earlier made the correction of mistakes a time consuming process.

Over the next few days two more simple maintenance functions were created: *VPG Maintenance* and *Fault Code Maintenance*. These functions were created quickly, having learnt from the first function. However, progress was hampered by the tedious FourthGen line-based logic editor which made moving quickly around the code impossible. The editor had no "cut and paste" facility for copying sections of common logic between functions, therefore common code had to be re-entered whenever required. Each piece of logic was numbered and could be called from any point in the FourthGen processing cycle. However, there was no facility for listing where in the cycle a particular piece of logic was called, nor for listing all of the logic numbers used in a function. Since logic could be called from many points in the half-dozen function creation screens, logic maintenance rapidly became a problem; it was easy to call the wrong piece of logic, and equally easy to have pieces of logic that were never called.

## 25/5/90. Starting work on the most complicated function: *Maintain PSR.*

The central process in the system was *Maintain PSR.* Each PSR contained a wealth of problem information and had to be validated against the master tables to ensure data integrity (this was a "feature" of the FourthGen New Model system that had not be available in the spreadsheet system). The screen contained a number of scrolling windows where multiple data items could be entered. Unfortunately the Analyser prototype screen for this function had been designed before FourthGen's processing cycle had been fully understood. The result was that the scrolling windows were scattered across the screen in the order shown on the original PSR form. FourthGen required that scrolling windows appear at the end of a screen, after all of the "single" fields had been placed. This required a major trial-and-error redesign of the screen in FourthGen. Since changes in FourthGen could not be reflected back into Analyser (there was no shared repository), the changes had to be rekeyed into Analyser. The interface was therefore a "One Way" interface; from Analyser to FourthGen. Furthermore, the transfer of design units from Analyser to FourthGen overwrote any corresponding functions in FourthGen. Since considerable further work had to be undertaken in FourthGen to get even the simple functions to work, it was easier to key changes directly into FourthGen than to import the updated design units from Analyser. The interface therefore became both "One Way" and "One Time". Once a design unit had been exported into a FourthGen function all further development was undertaken in FourthGen. The result was that the Analyser repository gradually became out of date.

## 31/5/90. Sinking deeper into the technical mire of FourthGen.

Working with FourthGen was becoming increasingly frustrating. The idea of providing a pre-coded screen, report and file processing "cycle" in order to reduce the need for bespoke code was sensible. Unfortunately the implementation was eccentric and poorly documented. For example, once a screen-full of data had been entered the screen was automatically blanked and the cursor returned to the first field to await entry of the next record. It is often useful to see the last record prior to entering the next, for example, to visually check that the data is correct. Despite tortuous coding efforts and calls to the ToolVendor help-desk we were unable to devise a mechanism for delaying screen blanking.

A further usability problem arose because the contents of a paging screen (scrolling window) were not retrieved until the cursor entered the paging screen. The *Maintain PSR* function had seven paging screens, which contained about half of the fields on the PSR screen. Viewing all of the data relating to a PSR therefore required several keystrokes instead of one. Unlike more modern database tools, a separate function had to be written to allow more than one PSR to be viewed on the screen (*table* view in database parlance). And since FourthGen had no query language, a separate function had to be written for each query requested. Nor were wildcards accepted; for example, to select all PSRs that related to build phase "QP" (Quality Proving) would normally be handled by an SQL statement such as:

*select * from PSR where build-phase = "*QP*".*

In FourthGen this required string processing commands to isolate the substring. The logic then had to be added to the appropriate point in the processing cycle (not at all obvious). Finally, the string "QP" had to be hard-coded into the logic which meant that a user wanting to see problems relating to a different build phase would have to call the programmer to recode the logic!

The lack of integration between Analyser and FourthGen has been described previously. The lack of integration within FourthGen now became a problem. Functions were constructed by filling in details on several screens. The first screen asked for function name, description etc.; the second for the text (labels) and their screen location (x,y co-ordinates); the third screen for field names, types and locations (x,y co-ordinates) and the fourth screen for file names, read write modes, and fields to be including in scrolling windows etc. Ideally screens two, three and four would be driven by a single screen painter that would allow the user to define text, fields and paging areas at the same time. Splitting the work across three screens slowed the process down and introduced tremendous opportunity for error. Some errors were simple, such as getting the text and fields out of place so that the screen looked messy. Others were more insidious, such as adding a new field in the middle of the screen, which entailed "shuffling" all subsequent fields down one in the field list. Since the scrolling windows were defined by stating the start field number and the end field number, the addition of a new field would destroy the boundaries on all subsequent scrolling windows. FourthGen issues no warnings when this happens and the symptoms are hard to debug: the cursor moves erratically about the screen and unexpected records are displayed in the windows.

At this point, the Consultant and the author started working in parallel to speed up development. A second copy of Analyser and FourthGen was loaded onto another PC at the Research Centre and the repository data from the first PC exported to the second. Analyser is a single-user PC tool that uses an import/export file transfer facility to exchange data between workstations. Every workstation has a user ID and every object has an owner ID that must match the user ID if the object is to be updated. Otherwise objects can only be viewed on "non-owner" workstations. Synchronising the two repositories was a further tedious and time-consuming activity that could have been eliminated by provision of a shared, networked repository. In contrast, FourthGen could run across a network and share files. Unfortunately our PCs were not networked therefore the FourthGen work also had to be synchronised from time to time.

## 7/6/90. Writing reports in FourthGen.

Report development was little easier than screen development. As stated previously, there is no query langauge in FourthGen, therefore all queries and reports have to be created as separate functions. This is a task for an IT professional, not for a novice user. Therefore the potential for *ad hoc* management reporting is lost - a problem that was to escalate once the system became operational. In addition, separate functions had to be written to maintain each file created (add, change, delete and view records). Referenial integrity was not supported, enabling the user to delete a PSR thus making orphans of all of the related records such as parts affected and launch teams involved in diagnosis. Furthermore, report functions could not accept parameters e.g. to select a subset of records to print; instead a separate "parameter" screen function had to be written and parameters passed as global variables to the report function. Finally, all of this tedious detailed coding was carried out against a backcloth of logic file "corruption errors" that intermittently corrupted sections of FourthGen logic! FourthGen bugs were to appear again later and led to considerable user unrest.

The central role of the entity-relationship model in the FourthGen processing cycle was gradually becoming clear. Most report functions, and several screen functions, processed data from several related entities. Typically these were one-to-many relationships, with data from the "one" entity appearing in the top area of the screen and data from the "many" entities appearing in scrolling windows in the bottom area of the screen. Reports worked in a similar manner, with the "one" data appearing at the start of a row and "many" data appearing further down the row, after the one data, and taking up one report line per "many" record. Generally, a FourthGen logical screen processed one entity. Therefore a screen that accessed several entities had to be written using several logical screens. The entity-relationship model was not mentioned in the FourthGen documentation, nor in the training course, and the Consultant agreed that ToolVendor staff did not appreciate the importance of the ERM in understanding the FourthGen processing cycle.

The rest of June was spent coding the remaining functions in *FourthGen. Simple functions could* be copied, which speeded up development, but even copied functions required variable name changes and other minor edits that were time-consuming using the primitive logic editor.

All of the functions were coded by the 29th June. Superficial testing was undertaken using a small quantity of test data based on the existing New Model system.

3/7/90. Demonstration of working FourthGen New Model system at Research Centre.

Six weeks had passed since the last meeting with the "customers". During this time the author and the Consultant had considerably furthered ToolVendor's understanding of the Analyser/FourthGen interface and the theory behind the FourthGen processing cycle. FourthGen had proven to be a complicated and old-fashioned tool and was far from bug-free. Despite this, the author and the Consultant were pleased with their achievements to date and looked forward to the demonstration. The author believed that the FourthGen problems were conquered and that the system could be handed over to the users with little further alteration, apart from occasional maintenance! The Consultant was working on a different project for ToolVendor, so the author demonstrated the system alone. The Sponsor was not available, but there was a good turn out of the remaining staff, including David, Tom, Andrew (from South Factory) and Andrew's boss, Stuart, also from South Factory. After a brief introduction, the author let David operate the system.

It was obvious to the author from watching a FourthGen "novice" operate the system that FourthGen suffered from some fundamental usability problems. A different function key is used to select each of the screen modes: add, change, delete or lookup. Each logical screen has its own mode which can be toggled independently of the other logical screens; and once in a logical screen the user cannot go back to a previous logical screen to correct an error. Instead the user must complete the screen and re-enter the function again from the start. David seemed to tolerate these eccentricities, but Tom was clearly a little confused. Certainly the spreadsheet-based New Model system that David had developed and Tom was using happily was a lot simpler to use! Nonetheless, the working system provided a valuable basis for discussion and stimulated a number of thoughts and ideas for improvement, including:

1. The need for two classes of operator: those who could change data, and those who could only view data.

2. The need to be able to enter "ALL" against SDV number on data entry, and "ALL" as default value for VPG number, Cell and Launch Team in report starter screens.

3. The need to allow the user to direct reports to screen, printer or file as required.

4. The need to speed up calculation of PSR totals against each launch team. This report took a long time to run. The author's solution was to introduce a new data entity to store the totals, thus avoiding the need to calculate the totals on each run of the report. This decision was to have significant repercussions later in the project.

After the meeting the enhancements were added to the FourthGen system and the Analyser repository updated to reflect the changes. Even sitting in a quiet office, with the changes written down on a piece of paper, it was easy to omit a change or make a mistake. It is clearly not realistic to rely on the developer to keep the tools synchronised.

17/7/90. Demonstration of Analyser and working FourthGen New Model system to Research Centre managers and CarMaker IT Strategy team at Research Centre.

The Consultant was keen to sell the project to the Research Centre and CarMaker management in order to maintain support for our work and to maximise the opportunity of sales of the software to CarMaker. To this end he arranged a demonstration to the senior managers based at the Research Centre, including the CarMaker IT Strategy Director. The meeting was cordial and friendly, with few difficult questions asked. The author was pleased to show off his work. But the meeting proved no more than a public relations exercise; it did not lead to greater openness from the IT Strategy team regarding their strategic plans, or their views on the project and the ToolVendor collaboration.

20/7/90. "Sales" demonstration to another client of ToolVendor: FreightWay Ltd.

Having demonstrated the system to people unfamiliar with the application area, a useful extension to the DFD technique became evident: there is no way of visually identifying processes that are subsequently exploded; or complex data stores or complex data flows. Therefore the "depth" of the DFD is hidden, which can mislead the reader. A simple diagramming extension such as shading of exploded processes or data stores or the use of thicker lines to denote complex data flows may clarify matters.

3/8/90. Staff from AgriCo Ltd. view the system.

The Consultant persuaded two former colleagues from his days at AgriCo Ltd. to analyse the New Model system and estimate how long it would take AgriCo IT staff to code an equivalent system. Subsequently they returned an estimate of six times the effort expended by the author and the Consultant. They based their estimate on an IBM mainframe teleprocessing environment and included full testing and error-recovery procedures. Clearly the two implementations were very dissimilar, and the AgriCo time-recording practices were unknown, therefore the comparison was poorly grounded. Despite this, both the author and the Consultant were pleased with the results and included the six-to-one productivity improvement in their project report which was subsequently used by the ToolVendor CASE salesforce.

*Waiting for Funding*

August/September 1990. The author and the Consultant prepare the "Detailed Project Report".

In the first draft of the report the author expressed several serious criticisms of FourthGen. To his credit, the Consultant let most of these go through to the final report with only slight rewording to make the criticisms sound more constructive. However, the author's final sentence was omitted from the final report:

*"FourthGen is NOT a tool for the inexperienced user."*

These words were to haunt the author after system implementation. The report signalled the end of the Consultant's day-to-day involvement in the New Model project. From here he concentrated solely on sales and public relations issues leaving the author to continue with coding, repository maintenance and user support.

*The Soft Sell*

2/10/90. Demonstration of updated New Model system and agreeing of sale at Research Centre.

Three months after the last demonstration to CarMaker "customers" the time had come to give a final demonstration in order to secure funding for implementation from the Sponsor. During this time the author had implemented the updates requested at the July meeting and the Sponsor had been busy preparing for the launch of the new Executive Car. The work to date had been funded by the Research Centre (the author's time) and ToolVendor (the Consultant's time and free CASE software and training). Now it was CarMaker's turn to fund the work by purchasing a FourthGen licence for the Manufacturing Division VAX at West Factory. The FourthGen New Model system could then be made available at West Factory, CarMaker's Design Centre and South Factory via the CarMaker network. The author, the

Consultant, the Sponsor, David and Andrew were joined by a ToolVendor CASE salesman, the CarMaker Manufacturing Systems Operations Manager and yet another new South Factory man - the Chief Engineer, Executive Car. David and Andrew were clearly keen to obtain the new system. The Sponsor regarded David highly and was happy to follow his advice. The decision to purchase was made, despite a last minute criticism from the Chief Engineer who stated that he was disappointed that the system was so old-fashioned and had expected to see pictures of problem sketches and graphs showing problem resolution rates, not just plain old text and numbers. He was right, of course, but the Consultant persuaded him that such technology was still expensive and slow. Further concerns were raised with the author by the Systems Manager some weeks later regarding the wisdom of CarMaker buying yet another 4GL on top of several incumbent products.

The demonstration led to nine further alterations in particular to cater for the Chief Engineer's wish to track problem resolution and to know if any problems were not being resolved quickly. The author listed the nine items on a memo sent to the Sponsor and the Chief Engineer shortly after the meeting. Four of the items were to be addressed by the author prior to installation, two to be addressed by ToolVendor at installation and three to be addressed by CarMaker post-implementation. The assumption being that David would take over system maintenance after implementation with only occasional help from the author and the Consultant. The problem of "requirements creep" due to the demands of new users and increasing familiarity with the system was to arise again and again during the coming months.

## 11/10/90. Essential improvements to FourthGen.

During the final round of enhancements, the author compiled the following list of essential improvements to FourthGen:

1. Provision of a screen painter to replace the x,y co-ordinate specification method.

2. Relative field numbers to be used to associate screen/report fields with logical screens/reports and with files. Absolute numbering means that whenever a screen or report field sequence is changed the developer must painstakingly review all the sections of FourthGen that might be impacted.

3. Provision of a full-screen logic editor with search/replace and cut/paste facilities to enable copying of common code between functions.

Time and again the lack of these facilities resulted in errors and delays to progress.

## 12/11/90 to 5/12/90. FourthGen and the New Model system installed on the CarMaker network.

One of FourthGen's major selling points is its portability across different hardware platforms. This proved to be true in this instance as the New Model system developed on a PC at the Research Centre was ported to the CarMaker VAX with only minor alterations. Significantly, these alterations were not described in the ToolVendor documentation and it took two ToolVendor support staff, the CarMaker Systems Manager and the author several days to complete the installation. The ToolVendor staff were clearly "learning on the job", which may be typical of an under-resourced organisation that relies on a few key staff to support the product. In the UK one senior member of the ToolVendor support staff had worked with FourthGen for several years, the rest had only recently joined the company and were learning as fast as they could.

The FourthGen security facilities are extensive but unfathomable to the novice. Security can be applied at operator, function or file level, but it is very difficult to obtain a report showing the current level of security across the whole system. The result is that security set-up is very much a trial-and error process and often omissions are only found when a user tries to add records to a file in a paging window somewhere deep in the system and finds himself thrown out of FourthGen and back to the system prompt! This problem was to manifest itself often over the coming months and proved particularly frustrating because once thrown out the user would often be timed out of the CarMaker network connection and have to wait ten or twenty minutes before the network responded to a new login request.

## *The Hard Sell*

### 28/11/90. Joint Research Centre/ToolVendor CASE Seminar at the Research Centre.

### 29/11/90 to 6/12/90. The author writes the user manual.

The user manual was written using a word-processing package. It should have been stored along with the rest of the project documentation in the Analyser repository. Unfortunately Analyser did not provide word-processing facilities so the manual (and all other project reports) were held elsewhere.

In comparison to the spreadsheet-based New Model system in use at West Factory, the FourthGen system required the user to master many function keys and screen concepts for effective operation. On the PC at the Research Centre changes in screen mode (add, change, delete and lookup) were accomplished by hitting function keys F5, F6, F7 and F8 respectively. It was not until implementation that the author realised that the VT terminals used by CarMaker forced the user to use an even more obscure set of keys located on the numeric keypad. For example, to change to add mode to add new data the user would have to hit keypad 4. For lookup mode, keypad ",". The author found this sequence difficult to master; pity the operator working under pressure at West Factory or South Factory.

*D-Day! The System goes Live at South Factory*

7/12/90. The author and ToolVendor support engineer hand-over system to South Factory users.

Despite the many FourthGen usability problems the author was optimistic that the system would prove beneficial to the CarMaker users. West Factory staff were between model projects and therefore not using the New Model system actively at the time. Instead, South Factory was chosen for the "launch" because they were just starting to record problems on the new Executive Car and needed improved IS support. To introduce an under-supported, under-tested IS in the middle of a critical project was, with hindsight, a risky thing to do. Nonetheless everyone involved was optimistic and Andrew greeted the author warmly and said he was keen to start using the new system. The author demonstrated the system to Andrew and another new South Factory colleague who was to help Andrew operate the system. Several points were raised at the demonstration:

1. Some of the fields in the PSR entry screen should be carried over from one PSR to the next to avoid rekeying information common to a batch of PSR forms.

2. Stop the system automatically logging the user out if they hit Escape at the main menu. This sometimes resulted in a twenty minute delay until the user could log back on but was not easy to fix.

3. Stop the system logging the user out of the current screen if a low-security user (read only access) accidentally attempts to switch to add, change or delete modes. A mistake easily made given the obscure keypad keys which FourthGen used to control mode changes on the VT terminals.

4. Changes to the PSR signal colour, denoting the status of problem resolution, in screen function DU2 caused a system error message to appear:

*"018 - ERROR IN WRITING TO FILE - DB9 (99484)"*

This message was alarming for the user and not particularly helpful to the developer. The error was never resolved despite several calls to the ToolVendor help-desk, and denied the users the facility to change the status of several PSRs on a single screen. Instead changes had to be made via another function, DU1, that only displayed one PSR at a time. This slowed down PSR maintenance and frustrated the users. More insidiously, the logic in the part of the screen near where the error occurred maintained the running totals in the PSR totals file (DB17). Later the South Factory users were to complain that the numbers shown on the PSR totals reports were incorrect. Since these reports were vital for monitoring New Model progress and were used in key meetings every week, the numeric errors seriously undermined user confidence in the system.

5. The users requested various enhancements including a new report (PSRs listed by Part Number) and user-friendly report direction screens (to direct reports to printer, screen or file). The author agreed to implement these changes but was concerned at the lack of willingness being shown by David to support the system.

*West Factory Abrogate: the Author Left Holding the Baby*

19/12/90. The author visits West Factory to train the users.

Despite the South Factory bug and changes requested, the author still felt optimistic that the system would be a success. The next stage was to get West Factory to use the system and to hand over support to David as had been agreed with the Sponsor sometime earlier. The visit to West Factory was soon to turn into one of frustration as it became evident that Tom was confused by the FourthGen user interface and both Tom and David requested a number of further enhancements.

Some of the enhancements required major changes to the data model because the PSR number, which appeared as a key field in most of the entities in the model was found not to be unique across models. Since PSRs for several different models would be stored in the database at any one time, and the same PSR number could be used in more than one model, a model identifier would have to prefix the PSR number wherever it was used. This represented a major re-organisation of the database which was currently filling up with PSRs being entered at South Factory! It was a sign of the inadequacy of the techniques used for modelling (and of the analysts' capabilities) that such a fundamental data issue had remained hidden.

At the end of the session it was clear that the West Factory staff could not use the system until the database changes had been made. It would have been unfair to expect David to make such a major change, therefore the author agreed to implement the changes provided that David learned how to maintain the system. The author leant David his FourthGen training course notes and set up a high-privilege user code on the system for David to create some test functions. The author assumed that David would be sent on a FourthGen training course shortly by the Sponsor, although this never materialised.

**** *CHRISTMAS 1990* ****

January/February 1991. The author maintains the New Model system.

Two principal activities were undertaken during this period:

1. Ongoing small-scale enhancements for South Factory and West Factory e.g. fixing reports that didn't work; attempting to hard-code a wildcard record selection facility for some of the reports (a contradiction in terms, but FourthGen provided little help in this area). Two of the more irritating problems were proving hard to solve: the security violation error that threw the user out of the system and the system error on writing to file DB9. The longer these problems persisted the more frustrated the users became.

2. Reorganising the database structure to accommodate the discovery that the PSR number was not a unique identifier. This was a time-consuming activity that impacted all of the FourthGen functions apart from the menu functions.

Whilst these changes were being made to the FourthGen database, the Analyser data model remained unchanged. There was barely enough time to fix the FourthGen problems, yet alone log on to a separate system in order to maintain a model that only the author was using. Instead the author kept track of the changes by marking the changes in pen on a print-out of the ERM.

### *The South Factory FAX Arrives*

28/2/91. Andrew sends a FAX to the author describing several serious problems.

The system was being heavily used at South Factory, which was pleasing to the author, but meant that any problems arising would have a greater impact. Andrew's FAX was the first time the author had realised that the PSR totals drawn from file DB17 were incorrect. DB17 was introduced early in the project to speed up calculation of running totals by maintaining an up-to-date set of totals for each launch team in one file. DB17 was updated from two screen functions, one of which was the function that triggered the DB9 system error described earlier. Despite trying many different fixes, the author could not resolve the problem, and the erroneous totals were to increasingly undermine user (and developer) confidence in the system.

To make matters worse, Andrew continued to champion the system at South Factory, which made the author feel very guilty about not being able to solve an apparently simple problem. To quote from the FAX:

*"Now we are running at full speed with the new system, by & large much improved from the old system, some anomalies have surfaced in the publishing*

*of statuses. This now presents one of the major bottlenecks in our functional improvement. ...*

*I have already spoken to David on these problems, but he feels that you are better qualified to give us rapid resolution.*

*I apologise for the fact that every time I talk to you these days I appear to be criticising the system, but in truth what I am actually trying to do is make the good system better.*

*I hope that you will be able to resolve these problems quickly & I hope to give you an unequivocal thumbs-up next time you ask me how everything is going."*

The FAX was copied to David at West Factory who apparently had been trying to support the users at South Factory, although he had not been keeping the author informed of progress. A few days later the author visited South Factory for the first time since system hand-over just before Christmas. All subsequent support had been carried out remotely, by telephone and from the VAX at CarMaker's Design Centre. The author received a friendly but concerned welcome from Andrew and the other two system operators. Later in the day Stuart arrived to discuss requirements for a new, and very complicated, report which was currently being prepared manually. The author added the report to his list of enhancements and set to work trying to fix the PSR totals problem. The new report was never completed.

13/3/91. David tells the author of South Factory plans to abandon FourthGen.

David and the author met at CarMaker's Design Centre around this time to discuss extending the list of users who could access the system. During the discussion David stated that CarMaker had re-organised and he was no longer available to support the South Factory system since South Factory were now part of a separate business unit to West Factory. System support therefore passed back to the author. David was disappointed that the FourthGen system was not succeeding because it gave the "pro-SystemsHouse" IT staff at South Factory an excuse to commission SystemsHouse to write a replacement system using their ApplicationMaster 4GL. Poor and expensive support from SystemsHouse had prompted the Sponsor to ask David to write the original spreadsheet-based New Model system and now we were about to go backwards again. This was the first that the author had heard of moves to abandon the FourthGen system.

18/3/91. The author visits South Factory to continue system maintenance.

Stuart requested more reports, which required alterations to the data model to accommodate the concept of "sub-launch teams" and had a knock-on effect to several other reports and screens. The maintenance list was growing longer with each visit!

Stuart discussed the future of FourthGen system with the author. The Executive Car project was nearing launch; 90 days after launch any unresolved problems would be handed over by the New Model Centre to the Product Support Group (Warranty) at South Factory. Warranty had their own problem management system, developed and supported by SystemsHouse with end-user development undertaken by the South Factory Warranty manager. The system had been enhanced recently and was regarded as a successful system with good support at a reasonable price (this was different to David's view of SystemsHouse). The author and Stuart discussed a facility for exporting data in ASCII format from FourthGen to the Warranty system, but at this point Stuart was still undecided about whether to retain both systems or to standardise on the Warranty system subject to extensions to provide for New Model support. The author was disappointed that his creation was likely to have a short life, but also aware that he could not give the system adequate support. Stuart showed the author the sales proposal received from SystemsHouse for the development of a New Model system in ApplicationMaster. The system consisted of a single table containing a fixed number of fields for launch teams, SDV numbers, fault codes and part numbers; whereas the FourthGen system allowed any number of entries in these fields via the scrolling window mechanism. It appeared that the ApplicationMaster system was no more sophisticated than the spreadsheet-system that the FourthGen system replaced, except that it was supported by SystemsHouse and not by end users! Nor was it clear whether the many reports that had been developed for the FourthGen system would be made available in ApplicationMaster. As David so rightly said "it's back to the future".

During the day a further FourthGen bug was shown to the author by Andrew. Spurious launch team details were retrieved when trying to change data on a PSR. The file that stored the launch team details was DB9 - the file that was subject to the system error described previously. On closer examination the data in the file looked fine, therefore the error lay in the FourthGen file-handling routines which could not be accessed by application programmers. The ToolVendor help-desk were unable to help, but noted the problem "for future releases". Nor could the author fix the problem, therefore another important area of the system failed and the author had to advise the users to avoid triggering the problem. The author could not recreate the problem on the PC subsequently, therefore it appeared to be a problem specific to the VAX release of FourthGen.

### 4/4/91. The author visits South Factory again.

The South Factory staff requested more flexible record selection in reports, in particular the facility to specify "ALL" against any parameter. This was very difficult to program in FourthGen and resulted in a considerable amount of new logic. The inflexibility of FourthGen for basic data querying tasks was quite breathtaking at times.

In addition to the various serious bugs and work-arounds, the South Factory staff were now experiencing problems getting FourthGen to properly recognise end of page when printing reports. End of page was exceeded by several lines before a page-break was thrown. The result was that the reports started off correctly then gradually "moved down" the page with every new page. Again system (and developer) credibility was undermined by an apparently trivial problem. Again the ToolVendor help-desk could offer little help and the problem persisted.

5/4/91 to 15/5/91. The author continues system maintenance.

During the next six weeks the author continued to maintain the system from CarMaker's Design Centre with occasional visits to South Factory. Stuart requested several more reports; the bugs and page-throwing problems persisted and the PSR totals still didn't add up correctly. Two issues recurred:

1. FourthGen bugs undermined the benefits of the system.

2. There was an urgent need for a flexible query language that could be used by novice users to generate their own reports. Most of the maintenance work revolved around the need for flexible reporting.

*SystemsHouse Win the Day*

15/5 91. The author discusses transfer of data with SystemsHouse.

The author returned to the Research Centre from South Factory to find a telephone message to "Ring SystemsHouse". Record formats were discussed and the author agreed to liaise with the CarMaker Systems Manager about transferring New Model data from FourthGen to the ApplicationMaster system.

This was the last act of the New Model project. After this the phone calls from South Factory stopped. No more was heard from SystemsHouse either.

The author had lost interest in supporting such a problematical system alone and was grateful for the release. Two months later the author revisited South Factory to conduct a post-project review with the South Factory staff. By this time all unresolved PSRs had been rekeyed into the ApplicationMaster system by a temp working one weekend, therefore there was no need to transfer data. The FourthGen system had been largely phased out and problem resolution was being continued using the ApplicationMaster system. Details of the interviews are presented in Section 2.4.2. The author felt a mixture of relief that he no longer had to support the bug-ridden idiosyncratic system, guilt that he had not been able to fix the bugs, and sadness that all of the pre-Christmas optimism had been rung out of the project by the realities of system support in a poorly-resourced environment.

## *2.4 Metrics and Stakeholder Views*

The research question driving the project was

> *"Do CASE tools enable quality systems to be developed quickly?"*

In this section metrics and stakeholder views from the New Model project are presented. The data is not discussed in detail here, instead the results from all three projects are compared with the literature in Chapter 7.

### 2.4.1 Productivity Assessment

The Mark II function point counting technique (Symons, 1991) was applied to the New Model system data flow diagrams as at 22/10/90[4]. Therefore the changes arising from the demonstration to the Sponsor and the Chief Engineer, Executive Car, at the Research Centre on 2/10/90 were included, but no changes after that date. Changes arising from post-implementation maintenance are therefore not assessed. The system size as at 22/10/90 using the Mark II analysis technique was 320 function points (FPs).

[4] Function point analysis calculates the size of an information system based on the number of inputs, outputs and files accessed.

The effort figures used here are for IS staff only (i.e. the author and the Consultant) for all pre-implementation activities[5]. The total effort figure came to 582 work-hours. Therefore the productivity figure for the New Model project came to:

320 / 582 = 0.55 function points / work-hour.

Assuming 7 hours per day and 20 days per month, this figure becomes:

0.55 * 7 * 20 = 77 FPs / work-month.

### 2.4.2 Quality Assessment

A defect and stakeholder (customer satisfaction) analysis for the New Model project follows.

**Summary of Defects and Changes**

Appendix 2.2 lists all of the defects[6] and changes that occurred during the project. The defect and change count is summarised by priority below.

[5] This is in accord with Symons recommendations (1991, p. 82-85). Casual user involvement, such as "... being interviewed about, or reading and clearing a specification ..." (Symons, 1991, p. 85), has not been included in the effort calculations.

[6] A defect is a variation from specification, or a fault in the information system.

|  | Priority |  |  |  |
|---|---|---|---|---|
|  | High | Medium | Low | Total |
| Defects | 8 | 8 | 3 | 19 |
| Changes | 0 | 12 (6) | 37 (18) | 49 (24) |
| Total | 8 | 20 | 40 | 68 |

**Table 2.1** Summary of Defects and Changes: New Model Project

Note: The figures in parentheses for change requests denote changes to the reports output by the system.

Several serious defects were reported together with many low priority change requests. If the users could have made these changes themselves, the author could have devoted more time to the serious defects. The eight high priority defects were:

1. No referenial integrity checking facility.
2. System error 18 in file DB9.
3. *Four* defects arising from incorrect PSR totals calculation.
4. Spurious launch team details retrieved in function DU1 (*Maintain PSR*).
5. Screens failed to scroll when directing report output to screen on the VT340 terminals at South Factory.

None of the eight defects were fixed by the author, all required specialist FourthGen expertise or were due to bugs in the FourthGen system software on the VAX. Of the

49 change requests, half (24) were for changes to reports, or requests for new reports, or for more flexible record selection facilities. Nearly all of these changes could have been undertaken by the users given an adequate query language. This may well have greatly reduced user frustration with the system.

### Stakeholder Interviews

The New Model project was never formally terminated, from the author's perspective it just quietly faded away. The author ceased maintenance of the system after 15 May 1991 and the South Factory staff stopped ringing the Research Centre. After a couple of months the author felt sufficiently confident to conduct a project post-mortem using the questions that he and the Consultant had used for the post-mortem of the CarMaker Warranty project, which had started after the New Model project but finished before. The post-mortem consisted of a series of one-to-one structured interviews with some of the major stakeholders conducted at South Factory by the author in July 1991. The West Factory staff were not interviewed since they had not been involved in the project for some months and had not implemented the FourthGen New Model system.

## The Questions

The following questions were used to structure the interviews:

1. What was expected of the project?

2. What was achieved?

3. What were the perceived problems with the project?

4. Were the modelling techniques useful?

5. Were the right people involved? If not, why not?

6. Was the scope established correctly?

7. Were the right issues/objectives established?

8. Would locally-based system support have helped?

Summary of Stakeholder Interviews

| Stakeholder | Positive Comments | Negative Comments |
|---|---|---|
| The Sponsor | IS provided useful support to the Executive Car project | IS "suffered from its own success" - unleashed latent demand for reports |
| | | IS lacked local support, led to user frustration at inflexibility and eventual replacement by ApplicationMaster system |
| | | More frequent on-site support expected from Research Centre/ToolVendor team |
| Stuart | IS provided greater variety of reports and was more usable than the spreadsheet New Model system | Unexpectedly high demand for local reports |
| | Visits to Research Centre useful for meeting West Factory staff | Lack of on-site support led to frustration and eventual replacement |
| | | Inflexible reporting; end-user querying facility expected |

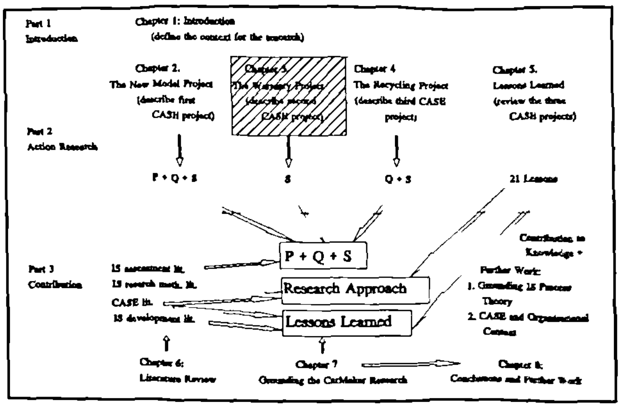
| Stakeholder | Positive Comments | Negative Comments |
|---|---|---|
| | | West Factory staff should have identified need for Warranty interface |
| Andrew | Extensive range of standard reports | "Victim of own success"; increased user demand for customised reports |
| | Much faster response time than spreadsheet system | Perceived easier to customise reports in spreadsheet system (logic strings) |
| | Extensive data validation facilities | PSR entry screen layout different from PSR form - slowed down data entry |
| | ApplicationMaster system used two screens for PSR entry/maintenance; FourthGen system used one | Lacked local support |
| | SystemsHouse support expensive and problematical; bugs present | Warranty staff not keen to transfer several years of back-data from ApplicationMaster to FourthGen |

| **Stakeholder** | **Positive Comments** | **Negative Comments** |
|---|---|---|
| | Research Centre visits greatly improved system understanding | No consultation with PSR Control's main customers - the Joint Engineering Teams (major report users) |
| | | Inadequate presentation of system capabilities to South Factory users |
| Stan (the South Factory IS Operator) | Extensive data validation | More data collected, led to more demand for reports ("the more you put in, the more they want out") |
| | "Great" range of standard reports | Need for *ad hoc* end-user reporting facility (cf. logic strings in spreadsheet system) |
| | Replacement ApplicationMaster system benefitted greatly from FourthGen lessons and facilities | System inflexible and screen layout sometimes unhelpful |
| | | No printer spooling |
| | | Printer paging and screen scrolling defects irritating |

**Table 2.2** Summary of Stakeholder Interviews: New Model Project

# *Chapter 3*

# *Study 2 - The Warranty Project: The "Marginalised" Project*



## 3.1 Introduction

The aim of this project was to improve the Warranty problem management process at the CarMaker East Factory manufacturing plant. The project entailed the development of a Problem Management System for problems received by the Warranty Group after the vehicles have entered volume production. The work of the Warranty Group involves the collation, assessment and resolution of vehicle quality problems. The environment at the start of the project was one in which problems were identified by a number of sources and described in a wide variety of forms - some computer-based, some manual. All recording and evaluation of problems was undertaken manually. The aim of the project was to provide a computer-based facility which would reduce the elapsed time between identification and resolution of a problem.

The project started in May 1990 and finished in January 1991, and therefore ran in parallel with the New Model project. The analysis was undertaken by the same two analysts as before, the author and the Consultant, supplemented on a part-time basis by two more analysts from the Research Centre. Again, Analyser was used to support the analysis process, but, as no software was developed, FourthGen was not used.

## *3.2 The Stakeholders*

*The Author*: Research Centre Research Fellow

Keen at the start of the project, but gradually disillusioned by the political manoeuvrings of the CarMaker IT Strategy team behind the scenes. Felt more comfortable with the openness of the Consultant and ToolVendor staff than with the "closed shop" of the CarMaker IT organisation.

*The Consultant*: ToolVendor Consultant

Highly-motivated and focused on personal and ToolVendor objectives. Became frustrated with delays in progress due to Warranty senior management lack of support and CarMaker IT Strategy team politics. Eventually lost interest and returned to mainstream ToolVendor CASE activities.

*Paul*: University Teaching Fellow

Acted as the Consultant's fellow analyst at start of project, later replaced by the author. Inexperienced in IT matters.

*The Research Manager*: Research Centre Research Manager

A dabbler who rapidly lost interest in the project. Later left the Research Centre.

*The Sponsor*: CarMaker Warranty Manager and Project Sponsor

Enthusiastic and visionary project champion who became increasingly frustrated by the narrow-mindedness and lack of support of his boss, the CarMaker Warranty Director. Involved and committed to the project although often busy elsewhere. Resigned soon after the end of the project to take up a Directorship with another engineering company.

*Jack*: CarMaker Warranty Systems Manager

Joined Warranty just before start of project. Keen to make his mark, very enthusiastic and friendly. Saw project as an opportunity to develop new methodological skills and strengthen his position as a key player at Warranty through the development of his own, alternative problem management system in parallel with the Research Centre project. Involved in parallel company-wide problem management strategy formulation in conjunction with the CarMaker IT Strategy team based at the Research Centre. The author and the Consultant were not told about either of these parallel developments, although they suspected something was afoot. Became concerned about the "hard sell" approach of the Consultant.

*Malcolm*: CarMaker Warranty Engineer

Quiet, gentle, laid-back character. Extensive and detailed knowledge of Warranty activities. Acted as main Research Centre contact during the detailed analysis phase. Helped Jack develop the alternative system in parallel with the Research Centre project.

*The Warranty Director*: CarMaker Warranty Director

Uninterested, anti-IT due to previous "bad experience". Not prepared to commit resources or political support.

*Pete*: CarMaker Warranty Manager

Peer of the Sponsor. Supportive of project, but not closely involved. Not IT-literate.

*Tony*: CarMaker Warranty Manager

Peer of the Sponsor. Not committed to project, involved in personal battle with the Sponsor. Sent subordinates to meetings but only attended one meeting himself.

*Simon*: CarMaker IT Strategist

Nominally IT Strategy link man for the Research Centre project. Working on parallel company-wide problem management strategy with Jack. A highly-political operator, concerned about ToolVendor "hard sell".

*The IT Director*: CarMaker IT Strategy Director

Authorised the project then lost interest.

## *3.3 The Story*

**Note:** CASE issues are highlighted by small font in the body of the Story text.

*Slow Start: Too Many Chiefs*

8/3/90. The Research Manager sends project outline memo to Warranty management (the Warranty Director and the Sponsor).

14 5/90. Initial project discussion at CarMaker Warranty, East Factory.

24/5/90. "Selling" of project idea to Warranty Director and senior managers.

> The Sponsor, Jack and Malcolm lobby for the project at East Factory. The Consultant, Paul and the author join them in a meeting to sell the project to the Warranty senior management (the Warranty Director, Pete and Tony). The Warranty Director is wary and non-committal, the Sponsor and Jack go ahead anyway.
>
> After the meeting the author, the Consultant, Paul. Jack, the Sponsor and Malcolm return to the Sponsor's office to draw up a context DFD on the whiteboard. The Sponsor shows Warranty at the centre of a vast "web" of information sources. Warranty's problem is how to manage the information effectively. The Sponsor tables an example of a "Problem Control Folder",

which holds all of the largely-manual documents that support problem identification and resolution.

*It's A Big One!*

29/5/90. Refining DFD and ERM at the Research Centre.

There is no print spooler in Analyser, therefore diagram printing locks up the machine until the printing has finished. Furthermore, there is no "bulk print" facility for specifying a set of diagrams to print. This means that the analyst must remain near the machine in order to initiate the next diagram print after the previous one has finished. Given the slow dot-matrix printer at the Research Centre, the task of printing off a complete set of diagrams ties up the analyst and the machine for over an hour.

5 6 90. Feedback and refine diagrams at East Factory.

The meeting ended with the identification of six objectives for the project:

1. Improve the flow of information into Warranty.
    - structure
    - regulate (currently overwhelmed)
    - make possible to handle.

2. Improve prioritisation of problems.

3. Monitor progress of problem resolution.

4. Monitor effect of problem resolution on the product.

   - continue to monitor until problem eliminated.

5. Predict potential problems before "the warnings flash".

   - e.g. by monitoring statistical trends.

6. Identify problems that need to be eliminated from future product designs.

   - the <u>top priority</u> objective.

Further meetings at East Factory over the next two weeks led to the production of a detailed set of DFDs for the current system. Seven level 1 processes were identified; two of which were chosen as the focus for the project's efforts:

- Process 1 *Raise Problem Folder*

- Process 7 *Predict and Monitor Success of Solution.*

19/6/90. Cost-benefit analysis undertaken.

In contrast to the New Model project, an extensive quantified assessment of IS benefits was undertaken by the Consultant and Jack for the Warranty project. The ToolVendor systems development methodology included a technique for assessing the benefit of an information system. The technique required the stakeholders to list project benefits under three headings:

1. Improve Revenue.

2. Avoid Cost.

3. Improve Service.

The major tangible saving identified was in avoidance of warranty costs due to a reduction in time taken to raise a problem folder process 1 from the current average figure of four weeks to a target figure of one week following computerisation. This would result in a saving of approximately £1 million per year in warranty claims avoided.

Paul had used Analyser to document the existing processes down to level 2 in some places. He had also added an ERM. The diagrams were erratically drawn and labelled and resembled a flowchart therefore the author and the Consultant dismissed Paul's efforts and started a new version of the project (v2) comprising of a single level 1 DFD to start with.

The following two weeks were taken up with some further meetings to clarify the cost-benefit analysis, and some further small refinements to the new level 1 DFD. The bulk of the analysis effort was devoted to filling the Analyser repository with descriptions of processes, external entities, data stores and any other item that could be documented. The danger of "analysis paralysis" was becoming evident. Certainly the fact that the repository could hold and structure so large a volume of data tempted the analysts to use the facilities and to over-document the system.

The author was still only involved occasionally at this stage, the Consultant and Paul were the principal analysts.

3/7 90. The Consultant produces the Project Determination Report.

The Project Determination Report contained a summary of the current system (a single level 1 DFD, v2 repository) showing the scope of the proposed project. The scope embraced two out of the seven processes, relabelled Process 1 *Define Problem and Analyse Warranty Impacts* and Process 7 *Predict and Monitor Success of Solution*. This was the only diagram in the v2 repository at this stage. Paul's ERM had been discarded by the author and the Consultant. The report included descriptions of the processes, external entities and data stores and cost-benefit analysis.

The report was circulated to all senior Warranty management, the CarMaker IT Strategy Director and the Research Centre and ToolVendor senior management. No feedback was received from the CarMaker IT Strategy team.

6/7/90. Project Determination Report review meeting with Warranty Director at East Factory.

The Consultant, Paul, the Research Manager, the Sponsor, Jack and Malcolm presented their case for continued support to the Warranty Director at East Factory. The author was not involved. The Warranty Director raised some reservations about the cost savings but allowed the project to continue as long as it didn't take up too much Warranty resource.

*Analysis Paralysis*

20 7 90. Commence the Draft Requirements Phase.

The project now entered a two month period of extremely detailed analysis. Malcolm was subjected to hours of document analysis and process and data definition by various combinations of Paul, the Research Manager, the author and the Consultant as the Analyser repository was gradually filled with vast quantities of data. Two issues created confusion:

1. Whether we should be documenting the current system in detail since it was already undergoing change via other initiatives.

2. Whether to document data flows that were unlikely to be computerised in the new system.

Issue 1 was impossible to resolve since the project team could not suspend the other initiatives. Instead, the author asked Malcolm to mentally discard the other initiatives and try to describe the system "as it stood". The second issue was resolved by the author deciding to document all data flows in order to obtain the complete picture of the current situation. This was to lead to a overwhelming level of detail subsequently.

Processes 1 and 7 were exploded down one level. Malcolm brought copies of many documents used by Warranty to the Research Centre and sat with the available analyst while the document details were entered into Analyser. The intention was to use the tool interactively as a prompt during these sessions. Whilst the tool did help to structure and focus the discussions, it also tempted the analysts to fill in every field on each screen. The result was that the analysis sessions became lengthy and rather tedious for all involved with seemingly no end in sight. Paul, the Research Manager, the author and the Consultant all took turns to lead the analysis during this period, which confused Malcolm somewhat. Each analyst had a different style. In particular, Malcolm told the author later that Paul had encouraged him to describe the system as it will be in the near future, after the current initiatives have been completed. The author, on the ther hand, insisted on modelling the "current" situation, ignoring these initiatives, with the aim of producing a second set of diagrams later to model the "future system .

13 8 90. AllTogether system described.

At an analysis session with the author, Malcolm described the information system that he and Jack were developing at East Factory to analyse warranty cost data. The spreadsheet component of an integrated software package for the PC called AllTogether was being used for development.

16/8/90. Review DFDs with hostile CarMaker Corporate Data Analyst at East Factory.

## 20/8/90. Version 3 of the Analyser repository created.

Paul had gradually faded out of the project, leaving the author to drive the detailed analysis. The current version of the repository had become a confused mixture of "current system" and "future system" due to the author and Paul's different emphasis. The author decided to start a new version of the repository (v3) to contain only details of the current system. The v2 repository was copied over to form the basis for v3. A major development at this stage was the addition of an ERM to the repository showing the relationship between problems, parts, manufacturing processes and cost data.

As the number of different data items stored in the repository increased, naming of new items became a problem. In particular, trying to impose naming conventions for similar items was made very difficult because there was no wildcard search facility to list all items containing the same substring. For example, there was no way of quickly seeing how all data elements containing the word "DATE" in their names were named; was DATE a prefix or a suffix? A similar problem arose when adding an existing item to a composite item; for example adding a data element to a data flow. There was no mechanism for viewing all current data elements and choosing from the list. Instead the operator either had to guess at the name used at the risk of getting the name wrong and inadvertently creating a new data element, or exit the creation of the data flow, enter the data element screen and page through the list of elements until the required element was found. The element name would then have to be noted, the data element screen exited and the data flow screen re-entered. Needless to say Analyser was not a Windows application, although limited windowing facilities were provided within the product.

## 4/9/90. Malcolm agrees to proof read and update all repository contents.

14/9/90. Progress meeting at East Factory.

The author, the Consultant, the Sponsor, Jack and Malcolm meet to discuss progress.

The sheer complexity and interrelatedness of the documentation used by Warranty was becoming clearer. To keep the DFDs simple and readable, much use was made of complex data flows which reduced the number of flows cluttering the diagrams. However, Analyser did not support "nesting" of complex data flows, therefore the component flows had to be "expanded" and all the constituent elements typed into Analyser separately. To add to the complexity of the exercise, there was no "cut and paste" facility within Analyser to copy long lists of data flow contents between composite data flows that shared data. Maintaining the increasingly voluminous repository without adequate tool support was becoming an error-prone and time-consuming activity.

18/9/90. Jack sends the author FAX of Warranty ApplicationMaster Problem Register.

27 9 90. Examining CarMaker Corporate Data Model for problem management.

The author and the Consultant discussed the relationship between the Warranty project and the CarMaker corporate data model for problem management with CarMaker IT Strategist, Simon at the Research Centre. Simon was guarded and indifferent and of little help.

A few days later the author received a telephone call from an anxious Jack who wanted confirmation that Warranty wasn't committed to implementing the system in FourthGen. He explained that he was making good progress with his AllTogether-based system and that he did not want to lose the work. By this time the author had developed considerable doubts about the efficacy of FourthGen as an end-user computing language and was happy to reassure Jack that implementation could be in AllTogether if desired. The author was also keen to encourage end-user computing and didn't want to be burdened with maintaining both the New Model system and a Warranty system in FourthGen. This clearly was against the Consultant's interests, but the author felt that a FourthGen implementation would have failed without considerable professional support.

At this time the author and Malcolm were putting considerable effort into validating and completing the v3 repository before moving on to define the improved system Another feature missing from Analyser that would have proved useful at this time was a facility to list all repository items that had yet to be defined (e.g. had a name but no description). We had been filling the repository for several weeks and knew there were many loose-ends but Analyser often lacked the tools to help us complete the task.

5/10/90. First "Lunchtime" Meeting at East Factory.

The project now entered a new phase. The detailed analysis of the summer, with Malcolm visiting the Research Centre, was now replaced with fewer, large group meetings usually on Friday lunchtimes at East Factory. The purpose of the meetings was to communicate work to date to a larger group of Warranty engineers and managers than had previously been reached. The author and the Consultant realised that the project had become becalmed and needed to be re-energised with new ideas and fresh support. Furthermore, we felt that we had almost finished the analysis of the current system and wanted to canvass ideas for improvements so that a new system could be constructed.

The fact that the group meetings were held in the lunch hour was an indication that Jack and the Sponsor had only gained limited support for the project from the other Warranty managers. Clearly the managers thought it more important that their staff attend to the "real work" during work time, and spend their own time attending the project meetings.

The first meeting was attended by the author, the Consultant, the Sponsor, Jack and Malcolm together with one of the Sponsor's peers, Pete, and two Power Train engineers from Tony's staff (Tony and the Warranty Director did not attend). The meeting was very successful. The author and the Consultant presented copies of the DFDs using an overhead projector and solicited comments and suggestions for improvements to the current system. The engineers seemed to understand the DFDs with little need for explanation.

8/10/90. Create v4 repository: Proposed System.

After five months of analysis we were at last turning our attention away from the existing system towards an improved system. A new version of the repository was started up to store the proposed system data. The v3 contents were copied across to provide a foundation. At this stage the proposed system was described by three DFDs, a level 1 DFD showing the seven high-level processes and a level 2 DFD for each of the processes of interest: Process 1 and Process 7. Process 1 (*Define Problem and Analyse Warranty Impacts*) had received most attention during the summer, now it was the turn of Process 7 (*Predict and Monitor Success of Solution*) to be investigated.

### *Good Ideas*

10 10 90. Second "Lunchtime" meeting at East Factory.

The second meeting continued the "selling" of the project to a larger audience. This time two further Warranty engineers were addressed by the author, Jack, the Sponsor and Malcolm. Again the meeting was very successful. The DFDs were understood and plenty of suggestions put forward for improving the system. The author summarised the issues arising under two headings: Issues Arising from Data Flow Diagrams; and General Issues. Examples of general issues included:

- Need to ensure that the engineer feeds problem resolution back into the system. Therefore engineer must understand why the system is needed.

- Ease of use vital. Engineer must want to use the system.

- System can ensure that everyone applies same problem solving method.

- Currently engineers update several clumsy parallel systems in order to close a folder.

- 90% of folders are not up to date.

- Typically one engineer will be looking at 40 to 50 folders.

- Currently Warranty problem owner may hold on to a folder and not pass it on to the appropriate engineer for fear of overloading the engineer.

- Partial fixes to open folders are currently discussed in meetings but the problem folder is not updated.

and finally ...

- Need to present the project to other interested parties e.g. engineering, warranty and service. This was clearly becoming a very big project!

19/10/90. Third "Lunchtime" meeting at East Factory.

A further meeting was held the following week. This time Pete and three new engineers attended as well as the established project team. Again, the meeting was positive and brought forward more suggestions for improvement.

*Loss of Momentum*

During the next few days the author put together an "Open Issues" document that summarised all of the issues outstanding and, where appropriate, allocated the issues to processes in the DFD of the Proposed System. The DFDs were updated to reflect the foregoing discussions. Despite the success of the lunchtime meetings the project was again losing momentum. Even the author was putting in fewer hours than before as the project moved into a pattern of meeting followed by brief update of repository, followed sometime later by another meeting.

12/11/90. JAD session at ToolVendor's offices.

In a final attempt to breath life into the project, the Consultant and the author organised a Joint Application Development (JAD) session at the ToolVendor offices. The aim of the session was to get everyone together "off-site" for a day of intense discussion in order to finalise a definitive system description (DFDs, an ERM and associated descriptions). The description could then be used as a basis for system implementation, which the Consultant still believed could be carried out using FourthGen. The Consultant, the author and Paul were present from the Research Centre/ToolVendor team together with the Sponsor, Jack and Malcolm from the "core" Warranty team, Pete and Tony, the Sponsor's peers in the Warranty management hierarchy, and nine other Warranty engineers. The Warranty Director did not attend.

The meeting was to be driven by reviewing each process shown on the DFDs and identifying whether the process was on line or batch; computerised of manual; daily or weekly; operational or managerial. Any further information was to be obtained and assumptions and open issues listed. The intention was to cover all of the processes in the course of the day and not to spend too much time on any one process. In the event the meeting rapidly became immersed in detail as different points of view were expressed and discussion drifted away from the process to wider issues. The use of DFDs and process descriptions seemed to encourage people to focus on detail rather than the overall system objectives. Some progress was made, but by the end of the day only about two-thirds of the processes had been covered and many issues remained unresolved. Certainly we failed to arrive at an agreed list of actions which could have taken the project forward.

3/12/90. The author updates repository in light of JAD session.

18/12/90. Final project meeting at Research Centre.

Jack met the author and the Consultant at the Research Centre to review the updated DFDs and discuss project progress. The meeting had not been intended as a finale to the project, but all three participants knew that the project was virtually dead. Jack introduced some further changes in the light of a new CarMaker strategic initiative in problem management that made it clear that the East Factory project had been overtaken by events elsewhere.

16 1/91. The Consultant makes final updates to repository.

## *3.4 Metrics and Stakeholder Views*

The research question driving the project was

> *"Do CASE tools enable quality systems to be developed quickly?"*

In this section stakeholder views from the Warranty project are presented. The data is not discussed in detail here, instead the results from all three projects are discussed in Chapter 7.

### 3.4.1 Productivity Assessment

A working system was not developed by the Research Centre/ToolVendor project team for this project, therefore coding productivity cannot be assessed. It is often argued that upper CASE tools, such as Analyser, slow down analysis because they encourage or enforce thorough analysis and discourage short-cuts. The use of Analyser in this project certainly encouraged more thorough analysis. The tool was used interactively to structure analysis sessions with Malcolm at the Research Centre, but much of the data recorded was never referred to again and the thorough analysis soon became "analysis paralysis" as too much detail was collected.

### 3.4.2 Quality Assessment

Since a working system was not produced there is no defect or change request assessment. Therefore the measure of quality used here is a customer satisfaction analysis based on interviews with the stakeholders.

**Stakeholder Interviews**

Like the New Model project, the Warranty project was never formally terminated; from the author's perspective it just quietly faded away. Three months after the last meeting with Jack, the author and the Consultant returned to East Factory to commence a post-project review with the key stakeholders. Most of the interviews were conducted jointly by the author and the Consultant, although some were handled by only one analyst. One stakeholder was interviewed at a time. In addition to the Warranty staff, the member of the Research Centre-based CarMaker IT Strategy team, Simon, was interviewed. Simon had been a shadowy figure throughout the project and seemed to have been carrying out a parallel company-wide study. The author and the Consultant wanted to find out more about his involvement.

## The Questions

The following questions were used to structure the interviews:

1. What was expected of the project?

2. What was achieved?

3. What were the perceived problems with the project?

4. Were the modelling techniques useful?

5. Were the right people involved? If not, why not?

6. Was the scope established correctly?

7. Were the right issues/objectives established?

8. Would a formal steering body and meetings schedule have helped?

Summary of Stakeholder Interviews

| Stakeholder | Positive Comments | Negative Comments |
|---|---|---|
| The Sponsor | Project provided a clear picture of how Warranty worked; understood by all | Insufficient user commitment |
| | Project enabled East Factory staff to lead the subsequent corporate re-organisation discussions | Users had not understood continuous effort required |
| | Jack and Malcolm were now skilled and active users of structured techniques | Project seen as low-priority, inadequately sold to the Warranty Director and the management team |
| | | The Sponsor and Tony disagreed on problem management strategy |
| | | Reactive culture of manufacturing a barrier to analytical thinking |
| | | The Warranty Director disagreed with the cost/benefit analysis |

| Stakeholder | Positive Comments | Negative Comments |
|---|---|---|
| | | The Sponsor and Jack "owned" the project to the exclusion of other Warranty managers |
| | | Divergent expectations of mode of IS support required (centralised versus end-user computing) |
| | | Scope of project constrained to quantifiable savings rather than bigger issues such as improvement of service to Warranty customers |
| Jack | Techniques "tremendously useful" in clarifying thinking, especially regarding the corporate re-organisation | Greater user commitment needed |
| | Helped identify duplication of data and enabled Jack to target own IS development to maximum effect | Over-emphasis on quantifiable savings |

| Stakeholder | Positive Comments | Negative Comments |
|---|---|---|
| | | Research Centre/ToolVendor team moved too quickly to modelling the new system |
| | | Hoped end users could use CASE tool (Analyser) |
| | | Project overtaken by corporate re-organisation |
| Malcolm | Original aims and objectives correctly identified and still relevant | CASE tool encouraged "box filling", led to too detailed analysis |
| | "Accurate" analysis meant that the DFDs were of value in corporate re-organisation | JAD session too detailed |
| | User involvement encouraged by structured techniques | Project low priority in comparison with day-to-day operations |
| | DFDs self-explanatory | Lacked clearly visible deliverables to gain senior management support |
| | | Prototype would have helped |

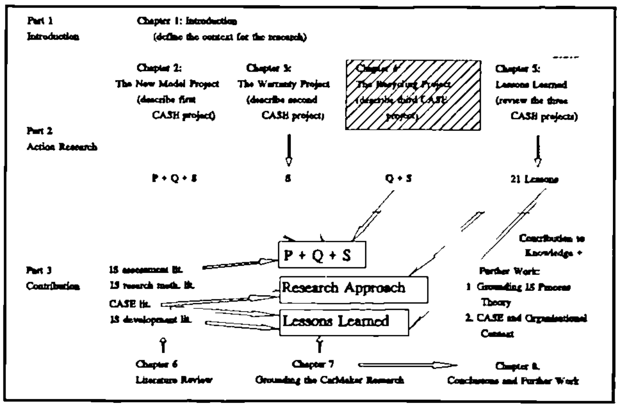
| Stakeholder | Positive Comments | Negative Comments |
|---|---|---|
| | | Little pressure on Malcolm, Jack and the Sponsor to deliver a system. |
| | | Confusingly different analysis styles of Paul, the author and the Consultant |
| | | Earlier training in structured techniques would have enabled greater user contribution |
| | | Greater project control regarding schedule, deliverables and "signing off" |
| Simon | East Factory work a "useful detailed case study" for corporate re-organisation | Broader scope necessary to address life-cycle problem analysis (Jack should have contributed here) |
| | Aims and scope appropriate for short-term East Factory needs | Lacked senior management input |
| | | Not embedded in CarMaker IT Strategy |

| Stakeholder | Positive Comments | Negative Comments |
|---|---|---|
| The Warranty Director | Engineers no longer devoting excessive time to information processing | Lack of management involvement seen as "fact of life"; more important issues than systems development |
| | Warranty systems personnel now much better at presenting information | Regular, tangible deliverables vital |
| | East Factory had retained Warranty function, other divisions had not | Top-down IS development "too grand", evolution better |
| | | Felt like "systems development" not "information development |

**Table 3.1** Summary of Stakeholder Interviews: Warranty Project

# Chapter 4

# Study 3 - The Recycling Project: The "Political" Project



## 4.1 Introduction

The aim of this joint project between CarMaker and Recyclate Ltd. was to develop a cost-effective vehicle disassembly and recycling process at Recyclate's plant. The project started in November 1991 and was still underway when the author left the Research Centre in December 1993. The planned completion date was April 1994. Unlike the New Model and Warranty projects, information system development was a *sub-project* within the main project, which was to develop a viable recycling process. The author led the IS development sub-project from May 1992 to December 1993. The Recycling Information System was to provide information on how to disassemble vehicles in a cost-effective manner by identifying parts which were worth recovering and showing how best to remove them. The IS was to act as both a "disassembly manual" and a decision support system for comparing vehicle recycling profitability. System development was to be undertaken in two stages; the first stage was the development of a trial information system. This was essentially a data collection and algorithm development exercise where the details of the disassembly operations and vehicle material content were entered into the system by engineers from the two companies. The second stage, the development of a production system for *Local Vehicle Recyclers* which would tell recyclers what parts to remove according to the current market value of the materials, had not been reached when the author left the Research Centre. Therefore the work described here represents the first stage of the planned IS development.

The Consultant had left ToolVendor at Easter 1992, therefore neither the Consultant nor any other member of the ToolVendor staff were involved in this project, leaving the author to lead IS development. Again, Analyser was used to support the analysis process, with system development taking place using the QuickStore relational database software and the NumberCrunch spreadsheet for graphical output. After the loss of momentum suffered in the Warranty project, the author decided to adopt an evolutionary approach to systems development for the Recycling project (see for example Gilb, 1988). This approach was considered appropriate due to the newness of the business process and the exploratory nature of the project.

## *4.2 The Stakeholders*

The IS Developers

*The Author*: Research Centre Research Fellow

Enthusiastic about the prospect of prototyping and rapid delivery at the start, eventually became overwhelmed by the resulting support and maintenance workload. IS made a scapegoat by project management for poor project progress.

*Mary*: Research Centre Research Fellow

Initially MSc. project student of the author, became full-time project member in November 1992. Technically inexperienced but calm and persistent; good at placating excitable users.

The Data Customers

*Geoff*: Research Centre Research Fellow and Recycling Economic Model Developer

Played dual role of "customer" of the IS and occasional IS developer; this led to strained relations with the author and Mary who were never sure which role Geoff was playing. Manic and eccentric at times. Bright and close confidant of other major "customer": CarMaker materials engineer Chris.

*Chris*: CarMaker Materials Engineer

Extremely enthusiastic and acknowledged materials expert. Inconsistent in demands for detailed data collection - sometimes extremely definite, other times nonchalant. Drove project into extremely detailed data collection with late appearance of requirements for accurate materials data. IT enthusiast, long-suffering awaiting materials reports from the information system.

The Data Suppliers

*Robert*: Recyclate Ltd. Special Projects Manager

Diplomatic, amusing, committed and shrewd. Adept at diverting IS resources to support his work (data collection and validation) at the cost of the data analysis work. QuickStore dabbler who occasionally caused the developers problems. Long-suffering with bugs in QuickStore. Under pressure from Recyclate Ltd. management to force pace of project against his better judgement.

*Jim*: Recyclate Ltd. Engineer and Information System Operator

Friendly, easy-going and humorous. A pleasure to work with. Long-suffering due to QuickStore usability problems, but eventually an advocate of the IS, despite being an Apple Macintosh owner and devotee. Somewhat undisciplined when entering data which led to reduced data quality and antagonised Geoff and Chris.

*Ronnie*: Recyclate Ltd. Disassembly Engineer

Eccentric and extremely knowledgeable about many subjects. "IT junky" who enjoyed building his own computers at work. Opinionated Windows fan who took an instant dislike to QuickStore because it was DOS-based.

## The Project Management

*Terry*: CarMaker Project Manager

An egotist who couldn't understand complexity of the IS task and made the IS the scapegoat for disappointing project progress. Political manoeuvrer who seemed to be working to a hidden agenda and chose to use his own data sources rather than use the data generated by the project.

*The Environmental Strategist*: CarMaker Environmental Strategy Director

Urbane, mild-mannered apart from when confronted by lack of progress. Weak grasp of IT, but expected much of the IS. Good grasp of own requirements which he explained clearly - unfortunately they were extremely difficult to realise using QuickStore. Often missed meetings, generally "busy elsewhere".

*The Accountant*: Recyclate Ltd. Project Manager

Shrewd, political accountant with weak grasp of IT. Seemed to be dominant partner in the joint project management team of the Accountant and Terry. Like Terry appeared to have a hidden agenda.

## *4.3 The Story*

**Note:** CASE issues are highlighted by small font in the Story text.

*The Early Cars*

November 1991. Vehicle stripping starts at Recyclate Ltd.

> The Research Centre was not involved in the project at this point. The genesis of the project is not known to the author, although the intention was to seek part-funding from the UK government Ministry in order to defray the costs. The project represented a major business opportunity for Recyclate Ltd. to develop the first vehicle recycling franchise in the UK. CarMaker's motivations were more defensive: to influence UK and European Community legislation on vehicle recycling and to ensure future designs met recycling standards.
>
> The accuracy and extent of the data collected during these early strips was extremely poor. Data was recorded on paper forms developed by Recyclate Ltd. staff. There was little input from the future customers of the data, Chris and Geoff.

10/1/92. The Research Centre becomes involved in the project.

13/1/92. The author and Geoff develop initial DFD and ERM.

Over the next three months, Geoff, Robert and Chris developed simple database in QuickStore for recording data. The author provided occasional modelling advice. However, Robert's staff were recording data using paper forms which had a different format to the database! A paper-based data "backlog" built up and the database remained unused.

QuickStore was chosen because it was the database in use at Recyclate Ltd. at the time. CarMaker were also users. After the experience of both ToolVendor projects, where lack of FourthGen skills and support contributed to the demise of the New Model system and where Jack was fearful of FourthGen being imposed upon Warranty in place of his AllTogether system, the author strongly advocated the use of the incumbent database for the Recyclate Ltd. project. Unlike the ToolVendor projects, the database developers were not formally trained in the use of QuickStore. Everyone learnt by using the system.

The stripping stopped at the end of February and did not resume until late August. The reason for this "lull" is not known to the author, but may have been due to disagreements in who was to pay for the scrap cars, a need to improve the stripping facility or a need to understand better the data to be gathered.

24/3/92. Official start date of the project.

Ministry funding is approved at 40% of costs.

April 1992. The "Graduate Engineer" report is produced.

A CarMaker graduate engineer who had been working on the project for the Environmental Strategist produced an analysis of weight of material removed from the vehicles by time. The analysis was based on a small set of inaccurate data and had been manually calculated. The simple curves were well presented but did not take account of "dependencies" i.e. the sequence in which the parts had to be removed given the physical structure of the car; therefore the analysis was unrealistic. The report was warmly welcomed by the Environmental Strategist, the Accountant and Terry and was to prove a millstone around the neck of the author and Mary later in the project when the true complexity of the data prohibited such simplistic analysis.

1/5/92. The author officially joins the project as the half-time "database sub-project owner".

13/5/92. Monthly project progress meeting at Recyclate Ltd.

The author's first progress meeting. The meetings were the main tool by which the management monitored and controlled the project. As the project proceeded the inadequacy of this "hands off" approach became increasingly apparent as the managers became alarmingly out of touch with the issues confronting the project team.

20/5/92. Visit to Research Centre by AutoStrip originator.

AutoStrip was to be a computerised disassembly manual showing graphical images of the car and highlighting the parts that should be recycled, together with their material type (colour coded). The project was based in Germany and was supported by the European car manufacturers association. CarMaker (via Chris) had committed to supply data from the Recycling project to test the AutoStrip prototype.

21 5/92. Demonstration of QuickStore IS at Recyclate Ltd.

The author and Mary demonstrated a first-cut prototype IS to Ronnie at Recyclate Ltd. Ronnie requested an on-line part name "glossary" to ensure that part names were recorded correctly. Apparently Chris had produced a list of 76 parts names for recycling purposes. By the end of the project, the on-line glossary had expanded to store nearly 2000 names.

The system was based on a hand-drawn data model (ERM) at this stage. The ERM is an essential tool for database design and therefore important when a prototyping approach is adopted. DFDs were not used at this stage because they did not appear to add value. After the Warranty "paralysis" the author was keen to introduce the IS to the users early and let development evolve through use. The DFDs were not essential for prototype development whereas the ERM was. The value of Analyser was therefore significantly reduced since one of the major techniques, data flow diagramming, was not being used and the lower CASE tool, QuickStore, was not integrated with Analyser. Not surprisingly, Analyser was seldom used in the project.

## 4/6/92. Demonstrate revised IS at Recyclate Ltd.

The suggested changes were implemented by the author. The ERM showed a strange amalgam of "part-centred" model which was favoured by Geoff, and "process-centred" model, which was favoured by Ronnie. The intention was for the Recyclate Ltd. staff to enter the data backlog into the database immediately. Vehicle and part details were to be recorded together with the "process" time required to remove the part. The bin into which the part was thrown was also recorded as this represented the "material stream". Multi-material parts were thrown into several bins, therefore the bin entity became a scrolling window on the Process screen. Finally any parts that had to be removed in order to access the current part were logged as "dependencies". At this stage the IS structure seemed simple and the author was optimistic that the IS would be used successfully in the near future. There was very little pressure on the IS developers or users; stripping had ceased, so the data backlog was not increasing. The summer was approaching and the project seemed to be drifting along comfortably and amicably. Indeed, the author and Mary were enjoying their trips out into the countryside to visit the eccentric world of the scrap metal dealer.

## 11/6/92. Monthly project progress meeting at Recyclate Ltd.

The author circulated print-outs of the database screens and reports for comments. He subsequently received a list of required new reports and changes from Robert and Chris.

Ronnie complained about the eccentric use of function keys in order to save records and to search for records, and the inability to see data from more than one entity when viewing records in table view. He wanted to see a screen that was the mirror image of his paper data collection forms. The forms contained data that was spread across several data entities, especially the dependency lists. QuickStore could only show data from a single entity in table view. This was to become a major cause of dispute between the IS developers and the Recyclate Ltd. users over the coming months and was a symptom of the "spreadsheet" loyalty of the Recyclate Ltd. staff. With a spreadsheet the user can enter multiple data items into a single cell by separating the datum with commas and reducing the font size. Similarly, data is saved automatically in a spreadsheet, there is no need to hit F2 to save or F8 to modify as in QuickStore.

## 22/6/92. The author commences work on the "Component Cost Report".

The Accountant and Terry had been calling for a report showing the "most profitable parts" to remove for some weeks. The author saw this as a tremendous intellectual challenge and an opportunity to show off his IT prowess. The report proved far more difficult than appeared at first due to the difficultly of summing the total cost of removing a part taking into account the dependency "tree" which represented the time taken to remove the part. Furthermore, in the course of removing one part, others might be freed. How should the time be apportioned between the freed parts? Tree processing is best accomplished using a recursive programming language. QuickStore did not provide recursion and the author was to expend considerable time and energy over the remainder of the project grappling with this problem. Needless to say, the "customers", the Accountant and Terry, considered the problem trivial.

8/7/92. Usability problems with QuickStore.

The author and Robert discussed QuickStore usability problems. In particular, Recyclate Ltd. wanted to work in "table view", but could not see all of the strip data on one screen (see 11/6/92 entry). By default, table view showed the records in the chronological order of entry into the database. If records for Car A were added, followed by records for Car B, followed by late additions for Car A, the Car A records would not be held in a contiguous group but would reside either side of the Car B records. Records could be sorted in table view but the process was tricky and easy to forget.

### *The Pressure Mounts*

9/7/92. Monthly progress meeting at Recyclate Ltd.

The pressure to deliver information was beginning to mount. The Accountant stated that he wanted to see reports from the database "like the Graduate Engineer's report". Geoff and Robert criticised the report as being "simplistic" and "unrealistic" because dependencies were ignored. The author described the "Component Cost Report" which aimed to take account of dependencies. The Environmental Strategist and Robert supported this work as a better alternative to the report. Geoff stated that the bin data which showed material type and weight was not sufficiently fine grain for the analysis that he and Chris wished to undertake. The author felt trapped in the middle of the warring factions but was also a victim of his own intellectual arrogance in wanting to show that he could "deliver the goods". Furthermore, his lack of experience of QuickStore programming meant that he found it difficult to judge whether the report could be programmed, or whether it was simply impossible given the restrictions of the tool.

30/7/92. Terry sends "Issues" memo to the Accountant.

The somewhat *ad hoc* approach to vehicle disassembly taken by Recyclate Ltd. so far was leading to several problems. Terry identified two major problems in his memo. Firstly, vehicle models were being mixed in the strip which reduced the opportunity to learn by focusing on one model at a time. Secondly, too much unnecessary data was being collected because "target" parts were not being identified prior to strip. Terry proposed the following "strip schedule" that would ensure that the top 12 scrap cars were targeted and that a common, small group of important recyclable parts were removed from each car[7]:

[7] The strip schedule identified the cars that were most likely to be scrapped in the next year or two i.e. the biggest sellers approximately ten years ago (assuming an average vehicle life of ten years). The schedule was subsequently modified to include more recent CarMaker vehicles (Cars 9, 10 and 11) hence the erratic sequence numbering after Car8.

|  | Vehicle | Number in Car Parc | % of Car Parc | No. to be stripped | Start date | Weeks allowed |
|---|---|---|---|---|---|---|
| 1 | Car1 | 116,580 | 7.7 | 6 | 3/8/92 | 3 |
| 2 | Car2 | 86,760 | 5.7 | 6 | 24/8/92 | 2 |
| 3 | Car3 | 86,020 | 5.6 | 5 | 7/9/92 | 2 |
| 4 | Car4 | 59,730 | 3.9 | 5 | 21/9/92 | 1 |
| 5 | Car5 | 49,420 | 3.2 | 4 | 28/9/92 | 1 |
| 6 | Car6 | 42,110 | 2.8 | 4 | 5/10/92 | 1 |
| 7 | Car7 | 41,850 | 2.7 | 4 | 12/10/92 | 2 |
| 8 | Car8 | 41,630 | 2.7 | 4 | 26/10/92 | 1 |
| 9 | Car14 | 32,050 | 2.1 | 4 | 2/11/92 | 1 |
| 10 | Car12 | 31,210 | 2.0 | 4 | 9/11/92 | 1 |
| 11 | Car16 | 29,600 | 1.9 | 4 | 16/11/92 | 1 |
| 12 | Car17 | 28,040 | 1.8 | 4 | 23/11/92 | 1 |

**Table 4.1** Initial Vehicle Stripping Schedule

The list of target parts suggested by Terry was as follows:

- bumpers
- air cleaner
- grille
- wheel trims
- external rubbing strips
- facia
- roof post finishers
- seat foams
- carpets
- wiring loom.

This attempt to re-focus the project was laudable but suffered from three weaknesses that were to come to light subsequently:

1. The time scales were far too tight, and Terry was to prove extremely loath to adjust the initial estimates.

2. No new CarMaker vehicles were included in the programme which upset the Environmental Strategist and Chris who wanted to use the project as an opportunity to learn about the recyclability of their current model range in order to feed in design improvements to future models.

3. Robert, Geoff, Chris and Ronnie disagreed with the ten target parts chosen and argued for a longer list. Furthermore, they wanted to strip at least one car in each set fully so that a complete picture of all the parts and all the processes and times could be recorded, based on the argument that we could not predict which materials would become valuable in the future therefore we should record all parts of any size in the information system for future reference.

10/8/92. Analyser used to create ERM.

This was the first use of Analyser on the project. Following discussions with Chris and Terry, the author thought it time to produce a presentation-quality ERM to describe the database structure to the project team. The ERM showed the main Vehicle/Process/Bin structure.

*Storm Clouds Forming*

12/8/92. Monthly progress meeting at Recyclate Ltd.

The author wrote in his report to the meeting:

> *"Awaiting active use [of database] at Research Centre and Recyclate Ltd.. Who are the 'customers' for the reports? 'Supplier' driven so far."*

also ...

> *"Some concern over accuracy of data collected to date."*

IS development to date had largely been driven by Robert at Recyclate Ltd. This was because Robert had shown great interest in the IS and was a good communicator. In contrast, the main "customer", Geoff, had hardly spoken to the author about the IS for months and was busy developing his own IS using the NumberCrunch spreadsheet, based on an economic model of the entire recycling life cycle. In meetings, Geoff had become increasingly disaffected, complaining that the data collected by Recyclate Ltd. (and stored in the IS) was "unusable". Geoff was unable to provide the economic analysis required by Terry and the Accountant due to the inaccuracy of the data and his own view that his economic model required further development. The author saw this as an opportunity to increase support for the IS by developing the "Component Cost Report" which seemed to answer the Accountant, Terry, Robert and the Environmental Strategist's needs using current data. Naturally this led to a certain competitive tension between the author and Geoff. Chris and Geoff were close confidants. Like Geoff, Chris felt that the data was inadequate for his purposes because materials were defined at "bin level" rather than the more fine-grain "material level". The Environmental Strategist threw in a new request, for a *graph* of weight of material removed against time. QuickStore did not support graphing and this innocent request was to generated considerable work in the future. The Accountant abdicated responsibility for his information needs by asking the author to "suggest uses for the data collected". Finally Robert spoke up in praise of the author's efforts and stated that he "has most of [his] needs defined now".

Given the above tensions and the belligerent attitude of Terry and, to a lesser extent, the Accountant, the project was heading for trouble. Meanwhile, the author still felt that he could distance himself from blame by explaining the ERM to the users; training the users in IS operation and *ad hoc* report generation and concentrating his efforts on producing the "Component C st Report" (now renamed the "Dependency Report"). He naively assumed that the bulk of the IS development was done and that others could squabble over data accuracy.

20 8/92. The "Optimiser" appears.

Within a space of four days both Terry and the Environmental Strategist requested a similar, sophisticated decision support facility (the "Optimiser') that would enable them to undertake "what if" analyses of the strip data. The user was to be able to change the market price for given materials and then see how the "value time" and "weight/time" curves for each car changed in response. The tool was to automatically re-order the strip sequence to ensure the most valuable parts were removed first, and was to take account of dependency constraints (of course).

## **** 27/8/92 to 18/9/92 Car1 stripping at Recyclate Ltd. ****

## *(4 cars)*

Recyclate Ltd. collected strip data on paper forms and subsequently entered the data into a NumberCrunch spreadsheet for simple totals calculations and validation. The QuickStore database was still not being used. Furthermore, Terry's July strip schedule had already slipped by 24 days and only four cars were stripped instead of the planned six.

24/9/92. Terry investigates "database problems" at Recyclate Ltd.

Terry had become increasingly frustrated by the acrimonious stand-off between the Recyclate Ltd. data providers (Robert and Ronnie) and the data users (Geoff and Chris). He therefore wanted to "sort it out" and called Robert, Ronnie and the author to a meeting at Recyclate Ltd. Terry could not grasp the complexity of the "dependency tree" argument and told the author to use a linear model to calculate costs based on one dependency per process, then agreed with Robert and the author that the dependency "chain" was, in fact, a "tree" with potentially many dependencies per process and therefore not readily amenable to computation using QuickStore.

The difficulties in producing the "Optimiser" were recognised and Robert proposed a compromise solution wherein Recyclate Ltd. marked those processes that they felt (subjectively) were "optimal" i.e. would produce an optimal strip sequence given current market prices for materials. Clearly if market prices changed this sequence could change, therefore hard-coding the sequence into the data could present problems in the future.

Robert tabled the paper forms being used for data collection at Recyclate Ltd. and Terry and the author agreed to restructure the IS to mirror the forms. This led to a new data model based on a one-to-many relationship between "level 1 processes" (assemblies) and "level 2 processes" (components).

Splitting parts data across two entities worried the author as part-based queries and reports would not be easy to write. Similarly, level 3 and beyond processes were not catered for, therefore disassembly of complex assemblies would be difficult to record. Furthermore, Robert and Ronnie wanted the IS to look and feel like a spreadsheet. In particular they wanted to be able to insert process records in the middle of a list and to cut and paste records as needed. This was not possible with a database product.

## 30/9 92. Monthly progress meeting at Recyclate Ltd.

Again, the database was the source of acrimonious discussion. The author's report to the meeting was similar to the previous months' report and defined current status as:

*"Awaiting active use at Research Centre and Recyclate Ltd."*

The author's solution to the stand-off remained unchanged; to train users to generate their own reports based on instruction in the data model. The intention was to off-load the routine support activities onto the users so that the author could concentrate on the more complicated tasks such as writing the Dependency Report. Since the author was only working half-time with no support he saw this approach as sensible given his limited resources. Indeed, after the New Model system experience with FourthGen the author had been keen to use a product that would allow the users to write their own reports. QuickStore provided a simple query language for this purpose.

Unfortunately the users seemed happy to let the author continue to service their needs, and whilst not overtly objecting to the suggestion, no user volunteered for training. The author missed his opportunity to make a stand on the issue and instead chose to concentrate on the Dependency Report and leave the users to their own devices.

### *The Warring Barons*

The project was split into six sub-projects, each owned by an individual. Terry and the Accountant acted as the joint project management team. As the project evolved the tension grew between some of the sub-project owners, namely the author, Geoff, Robert and Chris. The reasons for this tension were many: partly "personality clashes", partly misunderstandings and poor communications, partly differing objectives and loyalties. Each "Baron" appeared more intent on defending his sub-project than working with the other "Barons" as a team. The most visible chasm was between Robert (and Ronnie) as the data suppliers and Chris and Geoff as the data customers. The author tended to side with Robert partly due to his good relationship with Robert and his poor relationship with Geoff. Terry and the Accountant acted as "Aunt Sallys" uniting everyone against them by their lack of grasp of the detailed issues confronting the project and their occasional unhelpful interventions. From time to time the author talked to each of the protagonists in turn in order to try to consolidate IS requirements but always failed to find a solution that would satisfy the warring factions within the extremely limited IS resource available. Against this unresolved political backcloth, the author was attempting to develop the Dependency Report that stretched QuickStore capabilities to the limit, as well as providing ongoing support and maintenance of the IS at Recyclate Ltd. as the Recyclate staff evaluated the system.

1/10/92. IS meeting at Recyclate Ltd.

The author met the Accountant, Robert and Ronnie at Recyclate Ltd. to discuss the IS. Some useful progress was made, particularly the author's realisation that the Level 1 Process/Level 2 Process ERM was inflexible and should be replaced by a model that had two main entities: Process and Part. This could help resolve the different viewpoints of Robert/Ronnie and Geoff, since the Recyclate Ltd. data suppliers viewed stripping as a process-oriented activity in which they "do things" to the car and parts fall off. For analysis purposes, Geoff and Chris took a part-oriented view and wanted to know the cost per kg of removing a given part or type of material. The data model reflected the author's continued confusion however in that process-related attributes such as "Level", "Removal Method", "Removal Comments" and "Removal Time" were allocated to Part instead of Process.

## **** 5/10/92 to 16/10/92: Car2 Stripping at Recyclate Ltd. ****

## *(6 Cars)*

Once again the Recyclate Ltd. workshop was alive to the sound of stripping as the next vehicle on the "strip list" was taken apart and the data recorded on the Recyclate-designed data collection forms. The database was still not being used for data collection. Geoff and Chris's concerns regarding quality and accuracy of data remained unresolved whilst Terry and the Accountant were happy to see the "facility" being used and seemed to equate progress with cars being stripped regardless of whether the data collected was of any use.

At the end of stripping, the author obtained a copy of one set of Car2 data from Recyclate Ltd. to use as test data for the IS and the Cumulative Value Report (the new name for the Dependency Report).

The strip schedule slip was now 41 days.

### *The Cumulative Value Report Challenge*

The bulk of the author's effort for October was devoted to the Cumulative Value Report. The report relied on "cascading" strip times down assembly/sub-assembly hierarchy in order to determine whether to disassemble an assembly or to sell it intact. Cascading of strip times caused QuickStore to run very slowly, to hang up, or to crash according to the complexity of the calculation! Once net values were calculated for each part, the report evaluated the most profitable level at which to strip the assembly. Dependencies were ignored at this stage since the calculations were sufficiently difficult to implement without further complications; therefore the report followed the same process sequence as the original strip but showed where assemblies should be disassembled further and the cumulative value as the strip progressed. The author felt very proud of the report and felt he had moved ahead of Geoff in the pursuit of a flexible financial analysis tool.

### *The Author Congratulated*

3/11/92. Monthly progress meeting at Recyclate Ltd.

The Ministry project officer visited the project for the first time today. Since the Ministry were funding 40% of project costs the team put on a united front to give the impression that the project was making good progress and that major issues were being resolved harmoniously.

The author felt very optimistic that solutions to the major IS issues were in sight, namely:

1. Data entry: Robert was altering his paper data collection forms to reflect the data model and database input screens. Robert acknowledged database progress had been held up whilst he finalised his manual data collection procedure and congratulated the author on his patience and quality of support! The author requested that Robert enter data directly into the database for the next vehicle strip in order to test the system and to reduce errors in data recording, particularly the inconsistent spelling and format of part names.

2. Cumulative value report: the author tabled a copy of the report for the Car2 data that he had entered into the database. Robert, the Accountant and Terry stated that they were very pleased with the report and again congratulated the author on his efforts! Geoff remained silent during these discussions. His next move was indicated by a sentence in his report to the meeting:

> *"Consider the use of recycling rate as a framework for measuring the viability of car disposal."*

The innocuous phrase "recycling rate" was to have serious implications for the IS later as Geoff was to demand that the IS deliver a report that showed the recycling rate for different materials. The rate required accurate time calculations, including dependencies, and accurate materials definition at a level of detail far greater than the "bins" which Recyclate Ltd. used to record material type.

3. Extra IS resources: Terry had agreed to recruit the author's former MSc. project student Mary for six months as a full-time database staff member.

4. Planning of database workload: the author planned to delegate day-to-day support for Recyclate Ltd. to Mary while he concentrated on the Cumulative Value Report and on developing a <u>process model</u> (DFD) of the recycling process in order to define the scope of the database work. Surprisingly, one of Terry's colleagues tabled his "information flow map" at the meeting. This addressed the same issue using somewhat *ad hoc* diagramming conventions. The author was concerned that overlapping initiatives were taking place and that his "domain" was being taken over by others.

### *Requirements Scoping and Project Planning*

The Analyser DFD facility was used for the first time, some six months after the start of the project, to enable the IS scoping exercise to be clearly documented. Unlike the process model, the data model was being used extensively, but was paper-based, not tool-based. This was because the model was relatively simple, subject to constant change and used by only two people (the author and Mary). Furthermore, there was no interface between Analyser and QuickStore so no "integration" benefits would accrue from maintaining the model using Analyser.

Tool integration problems of a different kind were beginning to appear due to the Environmental Strategist's request to have strip data presented graphically. QuickStore did not support a graphics facility so the data had to be exported to the NumberCrunch spreadsheet and manipulated in NumberCrunch to produce the curves. The manipulation entailed a great deal of effort and had to be repeated for every "what if" calculation. Furthermore, no-one at the Research Centre had attempted to interface a database to NumberCrunch before, nor to plot graphs in NumberCrunch therefore the work progressed slowly and absorbed much of the scarce IS resource.

### *** *10/11/92 to 11/11/92: Car3 Stripping at Recyclate Ltd.* ***

### *(2 Cars)*

Robert continued to record strip data on paper forms. The forms were subsequently entered int the database in the Recyclate Ltd. office by a new Recyclate recruit, Jim. Jim was an Apple Macintosh owner and devotee and was initially amazed at the complexity and unfriendliness of the QuickStore user interface.

The strip programme was 63 days behind schedule at the start of Car3 stripping.

### *** *19/11/92 to 8/12/92: Car4 Stripping at Recyclate Ltd.* ***

### *(3 Cars)*

### *CarMaker Watch Closely*

Chris took a close interest in the Car4 stripping because it was the first CarMaker vehicle to be stripped and he required data from the strip to populate the "Strip Manuals" that he was developing as part of his contribution to the AutoStrip initiative (see 20/5/92 entry). The presence of Chris at Recyclate Ltd. soon led to tensions as Chris became alarmed at the inaccuracy of Recyclate's part naming and material identification efforts. Chris had a volatile nature and occasionally exploded! Terry was also interested in how CarMaker vehicles compared with other makes in terms of recyclability. For the first time in the project the data was subject to close inspection and was found to be far from acceptable. While tension was mounting at Recyclate Ltd., Mary was helping Geoff to develop the "Marginal Cost Report" using QuickStore at the Research Centre. The report was based on Geoff's recycling rate metric and needed to take account of dependencies. Mary and Geoff struggled with the report while the author continued to refine the (rival) Cumulative Cost Report and develop his master plan for IS development based on Boehm's (1988) Spiral Model.

The strip schedule slip was now down to 59 days as only two Car3s had been stripped instead of the scheduled five.

25/11/92. Terry rejects the author's "Spiral" Database Project Plan.

*The Explosion*

3/12/92. Mary and Geoff visit Recyclate Ltd. to collect Car4 data.

> Geoff required high quality data to test his Marginal Cost report and to provide input to his economic model. Instead he found Robert and Jim struggling to enter data into the database due to various usability problems, and, more importantly, he found that the Recyclate Ltd. data collection forms enabled parts to appear in the database without weights, part names and removal times. This prevented his report from working and was the "final straw" - Geoff exploded and rejected all the data collected by Recyclate Ltd. as "rubbish". Furthermore, Recyclate Ltd. had invented their own part naming conventions that did not meet Geoff's requirements.

The reason for the "parts without names/weights" problem was that Recyclate Ltd. would include in the data sheets details of processes that led up to the removal of a major part but disgorged no parts themselves. For example, the process that removed the engine was preceded by ten "preparatory" processes. Apart from chronological order of entry, there was no way of showing in the database that the time for engine removal should include the previous ten processes. Therefore engine "cost" was grossly understated. Similarly, at level 2, where assemblies were disassembled on the bench, it was common for a single process to disgorge several parts. For example pulling the grille from the bonnet would disgorge two parts: the bonnet and the grille. Recyclate Ltd. assigned the time for the process to the first part and left the second part time as zero, again grossly understating the "cost" of the second part.

Clearly the data model was too simplistic and failed to recognise the complexities of stripping and the intricate relationship between process, part and sub-part. Both Robert and Geoff were frustrated by the database, which was seen as a major part of the problem.

## 4/12/92. Geoff redesigns the data model at the Research Centre.

Geoff was in a manic mood on returning to the Research Centre. He was familiar with the ERM technique and rapidly set about redesigning the data model with Mary. The author looked on in amazement as Geoff intelligently expressed his requirements and showed how Recyclate's requirements could also be met. The author felt embarrassed that he hadn't realised the deficiencies in the model himself and also threatened by Geoff taking over his role as data modeller.

The heart of the problem lay in the relationship between process and part. The author had defined this as a one-to-many relationship, which meant that a sequence of processes that precede the removal of a part could not be modelled. In addition, the removal time, method and comments were stored against the part entity. In Geoff's redesigned model these attributes were placed against the process entity; a many-to-many relationship was established between part and process to cater for a process chain that eventually "disgorged" one part; a recursive one-to-many relationship established on part to represent the parent-child (assembly-subassembly) relationship and the lack of materials data granularity was resolved by the addition of a many-to-many relationship between part and material and a one-to-many relationship between bin and material. At a stroke many of the flaws in the model were removed! Despite the loss of face, the author could see that Geoff's model was a clear improvement and agreed to reprogram the software to correspond to the new model. Virtually every screen and report had to be changed.

Geoff became magnanimous in victory and admitted to the author that he hadn't realised the precise nature of his data requirements until very recently. Therefore he was unable to explain his requirements to the author and was apologetic for causing such an upset!

During the following week the author, Mary and Geoff refined the data model, re-programmed the database and transferred the data from the old structure to the new structure. Bearing in mind Robert's usability complaints, the author was loath to implement the many:many relationship between part and process because this would entail a convoluted screen design for data entry.

Instead, he persuaded Mary and Geoff that a more usable implementation would result from a one:many relationship between part and process. This was to prove a serious mistake. Meanwhile Recyclate Ltd. finished stripping Car4s.

### *Black Friday*

### 11/12/92. Database meeting at Recyclate Ltd.

The author and Mary visited Recyclate Ltd. on the Friday morning to demonstrate the new database, which they were confident would be welcomed by Robert, Jim and Chris (also present). After several minutes Robert stopped the demonstration complaining that the new system did not support the way Recyclate Ltd. worked. In particular there was no facility for recording many parts against one process; for example when an assembly was pulled apart revealing several components. The part-centric interface was rejected and Geoff summoned by telephone to come to Recyclate Ltd. and explain what was going on.

With Geoff present the afternoon turned into an acrimonious blood-letting exercise. Once the anger had subsided, Robert took control and explained the Recyclate Ltd. data recording format using a flipchart. This was the first time all interested parties had talked together at length about the detailed issues that were causing concern. As Robert spoke the structure of the data became clear to the author for the first time. Robert described his two-part 'spreadsheet". The first part recorded "Framework" data i.e. parts removed directly from the car. Generally the processes at this level disgorged zero or one part, and could form a chain of processes leading up to the removal of one major part, such as the engine. The second part of the "spreadsheet" recorded the "Benchwork" data i.e. the disassembly of assemblies on the bench in another part of the workshop. Here it was important to record the assembly being disassembled and likely that a single process would yield several parts. In Robert's spreadsheet process and part data were confusingly mixed but the underlying structure was clear to the author. The framework/benchwork separation suggested the key to the problem: the relationship between process and part was not many:many as the author feared, but, in fact, <u>two</u> one:many relationships: "disgorges" and "disassembled by". This would facilitate a usable implementation in QuickStore and provide a realistic data model for reporting purposes.

The second stage of the debate was less successful as all present wrestled with how to apportion removal time to individual parts given that the part may have been removed at the end of a long tree of dependencies with other parts also being freed en route. The Accountant was keen to have the database "cherry pick" the most profitable parts for him, but it was almost impossible to assign a removal time (and therefore cost) to a part that came off as part of a complex dependency tree. By the end of the afternoon everyone was exhausted, but relieved to have come together as a team for the first time, after eight months of increasingly acrimonious working. From feeling an isolated scapegoat, the author was heartened by the thought of becoming part of a team and the comradeship that was evident by the end of the meeting. Two factors had brought the team together: firstly Geoff's explosion and the pressure to start producing believable results from the stripping and secondly the common mistrust of the project management team of the Accountant and Terry and their unrealistic demands (the strip schedule slip was now over 80 days).

To ensure that everyone had a detailed understanding of what happened when a car was stripped it was agreed that all would attend the next vehicle strip at Recyclate Ltd. and experience the "process" for themselves. This was in contrast to the 'traditional" CASE approach adopted by the author to date - to build a system based on off-site discussions with users and the analysis of documents used. This approach had contributed to the confusion surrounding database requirements and had lessened the author's awareness of the frustration felt by the Recyclate Ltd. users.

14/12/92. Chris provides list of materials reports required.

For the first time, Chris provided a detailed specification of the materials reports he needed. The reports were of two forms:

1. For a given part, list all the materials from which the part is made across the range of cars stripped.

2. For a given material, list all of the parts which are made from the material across the range of cars stripped.

Both types of report were to present difficulties. The first type required a consistent use of part names across all cars stripped. In fact, the part name glossary had been changing and evolving with each successive car and there was no guarantee that the same name had been used for the same part on two different models. The second type of report was driven by Chris's material identification code, yet all of the data collected so far had used Recyclate Ltd.'s bin IDs. A bin could contain many similar, but different, materials. Therefore knowing that a part had been consigned to a bin would not provide sufficient information to identify the material content. Most of the parts from the early strips had been disposed of, so the only solution was to repeat the strips in order to obtain accurate materials data!

Meanwhile, Geoff was taking control of the team and produced a detailed disassembly plan using a simple "objectives/analysis schedule reports required" format for a four page document. The plan was commendably clear and well-received by the project team. It was a far better way of describing information requirements at this stage than a DFD, combining objectives, activities, decisions and reporting requirements in one concise document.

In response to the need for additional resource the author became a full-time member of the Recyclate Ltd. project from 14 12/92 to 31 1 93.

16/12/92. Monthly progress meeting at Recyclate Ltd.

The contrast with the previous progress meeting couldn't have been more stark. In November, the author had been congratulated for his efforts in producing an excellent IS. In December Terry opened the meeting by stating that the normal order of subproject reports (Robert, the author, Geoff, Chris etc.) would be resequenced by moving the database discussion to the end to allow plenty of time for an inquiry. The author immediately felt uneasy, singled-out for "special attention". Prior to the meeting the author had felt that Terry and the Accountant would demand explanations and so had assembled a "case for the defence" consisting of the database progress reports from the past five meetings which showed that the author had consistently voiced concern regarding the lack of interest shown by many project members in the database development and the lack of feedback received despite requests for input. In his "plan for next month", the author stated:

*"Database [the author and Mary], C.E.D. (Cost Effective Disassembly) [Robert, Jim and Ronnie] and Economic Modelling [Geoff] to work as closely-integrated team in future. Database is an integral part of C.E.D. and economic modelling and cannot proceed without effective communication between these areas."*

The author had entered the meeting confident of the support of the other "team" members following the recent teambuilding experience. Instead he found himself being criticised by Terry and the Accountant whilst the others sat in silence. Exasperated, the author took to his feet and drew a model of the project on the flipchart showing that the database could not be used to "glue" together in software feuding individuals. This was a thinly-veiled criticism of the project management which Terry and the Accountant chose to ignore, preferring to focus on "why the database isn't working". After a very unpleasant few minutes, the author sat down and listened as Terry and the Accountant reasserted their position.

Whilst the author felt strongly that non-technical issues were the root cause of database problems, he also knew that the poor usability of QuickStore and the difficulties in implementing cost reporting meant that technical issues were present and could be used as a convenient smokescreen for Terry, the Accountant and others to hide behind when it suited them. The argument that if we just solve the technical problems, the other problems will disappear was a difficult one to disprove. Unfortunately, the failure to reject this argument led to the diversion of IS effort over the next few weeks as Mary spent time evaluating alternative database products to assess whether they would be more usable than QuickStore.

20/12/92. The author lists Issues, Assumptions and Reports Required.

Towards the end of the Warranty project, the author and the Consultant had become interested in the Joint Applications Development (JAD) method. The aim of JAD is to obtain quickly an agreed set of systems requirements based around intensive workshops involving all users. To guide the JAD, the facilitator prepares a list of Issues and Assumptions which can be altered as the workshop progresses. At the end of the workshop an agreed list of requirements emerges together with actions on individuals to implement the work. Clearly the conventional analysis approach adopted so far had failed to overcome the Warring Barons culture of the project. Now that the "team" had emerged, the time seemed right for a JAD.

## *** *21/12/92: Car4 #2 (Trial 26) Stripping at Recyclate Ltd.* ***

The chaotic, *ad hoc* stripping and data recording previously undertaken by Recyclate Ltd. was now replaced by a carefully-planned activity undertaken by the whole "team". Each car was given a trial number which acted as a unique identifier for the vehicle. The first car in a batch was stripped fully; this was the "full technical strip" or "#1 strip". The second car was stripped more quickly for target parts only, and with process improvements implemented following a review of the #1 strip - this was the "reduced technical" or "#2" strip. In theory several #2 strips could take place to evaluate the benefits of different stripping methods. Once an optimal stripping method had been agreed an "optimal" or "#3" strip was carried out to provide final timings and parts data. The next Car4 strip was a "#2" strip.

21/12/92. The "Team" view a Car4 strip at Recyclate Ltd.

The day began at 8 am with everyone grasping their mugs of hot coffee in the freezing cold Recyclate workshop. The project was being run on a low-budget and the Recyclate Ltd. facility suffered from poor heating and lighting. After the comfortable colour-coded Research Centre offices this was quite a shock. Many of the team were wearing thermal underwear and the atmosphere was akin to a school outing with jokes and wisecracks flying around. Geoff brought a video camera to record the strip in details, Mary took a "team photo", and the author helped man the flipchart upon which was pinned an enlargement of the Recyclate Ltd. data capture form.

Ronnie stripped the cars while another Recyclate Ltd. employee. Don, took the parts to be weighed. Jim, the author and Mary took turns to record the process times using a stopwatch. Process and part details were written up on the flipchart. Watching the strip, it became clear that simple things like what constitutes a process and whether a part is an assembly or a component (therefore whether it will be subject to further disassembly), were not at all obvious. There was much discussion and data was often changed as the strip progressed. The workshop was noisy and Ronnie wanted to make quick progress and objected vociferously if the "data entry" team asked him to slow down because they hadn't finished recording the previous process. Part naming was highly subjective, typically Chris would disagree with Ronnie's names and a lengthy discussion would ensue. The mix of characters was also revealing. Ronnie wanted to rush through the strip but was prepared to argue over points of detail. Chris was very excitable and would often confuse the issue by jumping to conclusions. Geoff took a more cerebral stance and tried to think through the implications of the stripping process on the viability of the entire recycling life-cycle. Jim was cheerful and accommodating and Robert seemed able to settle an argument with one or two witty interjections and was happy to listen to all points of view. The day was enjoyable and the author felt again that he could rely on the support of the team in making the IS a success. However, it was clear that the stripping process was a noisy and volatile activity and direct data entry would require a highly usable system.

Following the strip, Geoff once again took command of the team, and organised two "debriefing" meeting at Recyclate Ltd. and the Research Centre. The meetings followed a carefully defined agenda and resulted in very useful discussions about the issues arising from the Car4 strip, including data collection issues.

**** *CHRISTMAS 1992* ****

5/1/93. Geoff provides the first detailed example of Car4 dependencies.

8/1/93. Parallel development of cost analysis tool.

Everything was happening very quickly and it was difficult for the author and Mary to keep abreast of developments. Communication with Geoff had improved considerably, but he was clearly determined to drive through his ideas and the author and Mary still had to fight to make sure that they had a say in developments.

While the author and Mary developed the new database at the Research Centre, Geoff, Chris and the Recyclate staff were experimenting with a spreadsheet for data analysis at Recyclate Ltd. Recyclate Ltd. had been transferring the paper form data into NumberCrunch for basic validation for some months. Now they were attempting to use NumberCrunch for all IS activities. The author and Mary felt threatened by this action as it appeared that the QuickStore IS could soon become obsolete and the end users would have developed their own, superior IS without their help!

### *The JAD Meeting*

11/1/93. IS JAD meeting at Research Centre.

Every member of the "team" attended the JAD; the project management did not attend. Each attendee was issued with a twenty page document which detailed all the issues and assumptions and showed database screen layouts and a suggested redesign for the NumberCrunch and paper forms. The author led the JAD and worked through the items in the document, noting comments, suggestions and actions as the meeting progressed.

Many of the issues were resolved and the JAD proved invaluable in giving the author and Mary the opportunity to focus team attention on the database for a whole day. Some issues could not be resolved at the meeting including:

1. Whether to use QuickStore or NumberCrunch for data capture. NumberCrunch was far easier to use in the noisy stripping environment and was liked by Ronnie and Jim. However, QuickStore provided an on-line part name glossary and extensive choice fields and data validation. Furthermore, data would have to be transferred to QuickStore for reporting purposes, therefore NumberCrunch represented an additional stage in the process. The author had mixed feelings about this issue. He was keen to see QuickStore used and to prove the worth of the IS, but he knew that NumberCrunch was more usable and would lead to fewer complaints from Recyclate Ltd. staff during data capture.

2. Manual resequencing of processes into an "optimal" sequence could be done either in NumberCrunch or in QuickStore. NumberCrunch provided a "cut and paste" facility which would make resequencing easier.

3. The algorithm to be used for cost calculations was still a source of dispute between the author and Geoff. As an interim solution the author agreed to provide a "Disassembly Decision" report that would determine if an assembly should be disassembled or sold intact. This side-stepped the dependency issue since dependencies could only occur at the framework level, not at the benchwork (disassembly) level.

4. Of the 28 reports listed, 5 were deleted, 19 were prioritised for completion sometime before the end of the Car6 stripping (now scheduled for mid-January), and 4 were set aside for further

investigation. Both Geoff's Marginal Cost report and the "Optimiser" were set aside. Chris's materials reports were scheduled for completion. All of these reports were to prove problematical in the coming weeks.

Following the reports, the author described the new database interface and the "drill down" facility that would enable the user to view a process, then look at the parts removed, then "drill down" to the part details, then view any processes that further disassembled that part (assemblies only) and so on. The author emphasised that users should view the database as an implementation of a Bill of Materials in that the parts and subparts were related as they are in a BoM. The team were impressed with the sophistication of the database and readily agreed to a redesign of the spreadsheet and paper forms to adopt a compatible data structure to the database. In principle all three "systems" should now be compatible. The choice of most suitable system for data collection could then be decided by a trial.

Geoff requested one change to the data model: that the term "process" be replaced by the term "operation" and that process be used instead to describe a "timed block of operations . Dependencies could then be defined against processes instead of at the operation level. This would reduce the number of dependencies since all operations within a process were dependent upon one another implicitly. It was also hoped that this would simplify dependency calculation.

The author and Mary enjoyed the JAD and felt that they had been given a clear remit to develop the database into a valuable system. They were also daunted by the amount of work arising and by the complexity of some of the issues still unresolved. Much still had to be learnt and it would have been premature to freeze the requirements at this stage; however the flexible, evolutionary approach to systems development was clearly difficult to accomplish successfully without adequate resources and experience.

14/1/93. Car4 Trial #26 data imported into QuickStore from NumberCrunch.

The supposedly simple task of importing data from NumberCrunch into QuickStore highlighted several problems. Recyclate Ltd. had not been consistent in part name spelling and format; the activity and bin values in the spreadsheet did not always correspond to those in the database; Recyclate Ltd. had used blank rows to improve readability - these became "blank" records in the database! In addition to this, much team effort was being expended on rationalising the part name glossary which had become full of synonyms, duplicates and redundant names.

## *** *18/1/93: Car6 #1 (Trial 27) Strip Commences* ***

## *** *at Recyclate Ltd.* ***

104 days later than shown in the July strip schedule.

18 1/93. Car6 strip at Recyclate Ltd.

Another big day for the database. As with the previous Car4 strip[8], the whole team was present for the first Car6 strip. Geoff was still driving the teamworking according to his plan. The team had decided to compare the usability of NumberCrunch, QuickStore and the paper forms as data capture

[8] Car5 stripping was postponed.

systems in a "head to head" contest at Recyclate Ltd. Two PCs were wheeled into the workshop area and placed near Car6. Jim was to use the NumberCrunch PC and the author and Mary the QuickStore PC. After about an hour of parallel input it became clear that the database was keeping up with NumberCrunch and had the benefit of an on-line glossary which supplied a list of validated part names and prevented mis-spellings. To ironic cheers, the team decided to abandon NumberCrunch and the flipchart and to use the database as the sole data capture system. The author and Mary were delighted!

During the course of the day several improvements were made to the database "on the fly". Unlike FourthGen, QuickStore was sufficiently flexible to enable changes to be made to the system quickly in a noisy environment. The author noted the usability problems as they arose and produced a list at the end of the day, including:

1. The need for separate formats for framework and benchwork data entry screens. QuickStore did not separate screens (forms) from tables; in other words, there was no *view* facility therefore the screen was the table. Therefore it was not possible to provide two different, customized views of the same table. This meant that some of the screens were unnecessarily cluttered because they contained all of the fields in the table, not the subset relevant to the task in hand.

2. The need to hit a key to save a new record (F2) or change an existing record (F8) caused confusion. In the spreadsheet all data was saved automatically. The users preferred to keep the previous record on the screen and overtype any changed fields since much of the data remained unchanged between records. This meant that if F8 was hit accidentally instead of F2 the previous record would be overwritten and, effectively, "lost". Unlike FourthGen, QuickStore provided no mechanism for carrying over field values between records so the user either had the choice of

blanking out the whole screen and retyping the unchanged fields, or risking accidentally losing the previous record. A lot of records were lost due to this error in the noisy stripping environment.

3. Many part names were not in the glossary, therefore many lengthy glossary searches proved in vain. Instead the part was given the name "other" and a suggested part name entered in the part comments field for off-line addition to the glossary. Searching for glossary names required the user to enter a complicated sequence of keystrokes which drew criticism from Robert and Jim.

4. Ronnie tended to strip the car more quickly than the database operator could enter data. After several heated discussions Ronnie was finally tamed and it was agreed that the database operator should also control the stopwatch. Ronnie couldn't start an operation until the timer was ready so this effectively stopped him racing ahead.

Reflecting on the day's events the author wrote in the project diary:

*"This trial, Car6 #1, is effectively the first trial. All previous trials can be viewed as a 'learning process' for the team. It takes a certain level of 'pain' for the learning to take place - can this pain be avoided by e.g. better methods/better project management? Formal project management (as practised by Terry and the Accountant) has been in operation throughout the project and has not ensured success:*

> *- Monthly meetings too divorced from detail and the reality of the project to ensure that Terry, the Accountant and the Environmental Strategist understand the issues.*

*- Project managers not interested in viewing for themselves what is going on (the Environmental Strategist has yet to see a strip, Terry saw first one today!)*

*- Project managers busy planning strategies (e.g. new 'bigger' Phase 2 project) which make assumptions about our work without consulting us first!"*

20/1/93. Geoff produces the Resequencing Procedure.

The procedure was a carefully thought-out ten-point plan for determining which parts were to be removed from the #2 car and the sequence of removal based on an analysis of the #1 car data. The procedure made extensive use of the database and two database reports: the Disassembly Decision report and the Sorted Net Value of Parts report.

Geoff and the author had lunch together the next day. The author reflected on the conversation in the project diary:

*"Have we moved from the 'MIS knows best' days to 'end user knows best'?*

*- End users are increasingly IT-literate.*

*- Tools are available to build powerful, user friendly applications*

*e.g. spreadsheets,*
*Windows,*
*PC 486s,*
*LANs.*

> *Usability is often the major symptom of the MIS malaise.*
>
> > *- Analysts cannot impose systems that are less usable than the spreadsheets! (Evidenced in all 3 CarMaker studies).*
> >
> > *- Users will not use these systems, will revert to their own, therefore analysts need lower CASE tools that provide highly usable HCI."*

21/1/93. The author and Geoff define a materials analysis procedure in Chris's absence.

> Chris was becoming increasingly concerned about the lack of granularity in materials data capture. He had agreed to use "bin" as the materials identifier, but bin also served as the Recyclate-defined recyclate stream and therefore was semantically overloaded. Instead of a separate materials field, the author and Geoff developed a convoluted "materials analysis" procedure that would enable the *existing* database structure to be used to record materials data, thus avoiding a redesign of the software.

*Teamworking Begins to Collapse*

27/1/93. The "team" validate the Car6 #1 data at the Research Centre.

All of the Car6 #1 data had now been entered into the database. Far more data had been collected for this car than for any previous car as *ad hoc* data collection had been replaced by a carefully considered stripping procedure and Chris had insisted on detailed materials analysis. The part name glossary had grown dramatically and had absorbed much time in debate and maintenance. The purpose of today's meeting was to validate the data and to produce a picking list of "valuable" parts to guide the Car6 #2 strip using Geoff's resequencing procedure. Sadly, the day soon degenerated into a confused wrangle as the author later reflected in the project diary:

*"All reports from the database ran correctly but were met with confusion by the users who were unable to differentiate between an operation and a part and therefore could not understand the reports! Chris, as usual, asked for complicated changes to the Level 2 worklist to show his materials analysis data alongside the assembly from which it was stripped."*

Chris's request to see the material breakdown of an assembly was entirely reasonable. Unfortunately entity-relationship modelling can lead to databases made up of many small tables and the materials data was in a different table to the assembly data, and the report processed the operation table as the main table so the materials data could not be accessed because QuickStore could not traverse the one:many:many relationship. Instead Chris had to be content with the materials data appearing on a separate report.

Having advocated the evolutionary approach after the "analysis paralysis" of the Warranty project, the author ruefully reflected on the hidden dangers of this approach:

*"Prototyping gives the users something tangible to focus their requirements. But can have reverse effect i.e. unclear requirements lead to focus on the database leads to criticism of database for not meeting requirements! Therefore there is a danger in the RAD/4GL approach in that the analyst is encouraged to prototype before the requirements are clear. This doesn't just lead to rework, it also leads to scapegoating of the database!"*

A subsequent diary entry continued the theme:

*"Many users seem to prefer to work in a reactive mode, with little planning of effects/benefits of their actions. Spreadsheet technology supports this way of working because it has no pre-defined structure and can be abused e.g. by entering multiple values into a cell - dependencies for example, and by shrinking the font size to add more if required. The fact that this data cannot be analysed is not foreseen! Analysts, conversely, are taught to think in a structured manner which leads to conflict with the users."*

Meanwhile a bug in QuickStore appeared when Robert attempted to restore the Car6 database. An "internal error" was reported in the glossary form which corrupted the glossary. The author also ran into "stack overflow" errors when calculating the dependency tree time for parts in the new Net Part Value report.

1/2/93. The author reverts to half-time working and the Accountant and Terry attempt to salvage old strip data.

The Accountant and Terry were becoming increasingly concerned at the lack of progress. They regarded vehicle stripping as progress, not data analysis meetings. Terry was keen to achieve his "stripping schedule" and assumed that the team would learn rapidly and that after a slow start would strip the rest of the cars at a tremendous rate. In order to increase the amount of data available to management, the Accountant and Terry asked the author to review the data from the earlier cars (Cars 1, 2, 3 and 4) to see whether it could be entered into the database.

The author entered the Car3 level 1 data into the database and found that changes in activity names, bin names, part names and materials identification meant that the Car3 data (trial 18) was seriously flawed. Data entry took some time due to the updating required.

8/2/93. Robert rejects Geoff's "scientific" resequencing.

Robert had never been sure about Geoff's "scientific" resequencing procedure (see entry dated 20/1/93). The procedure was not yet fully working due to the difficulties in implementing a dependency cost calculation algorithm in QuickStore. Geoff had accepted the database limitations and had evolved the procedure to use a combination of database reports and "rules of thumb". Robert preferred to select parts based on his subjective opinion about their resale value and the current market demand for materials. Geoff and Chris had criticised this approach because it could not cope with changes in market value and the selection rules were ill-defined. Up to now Robert had been prepared to allow Geoff to dictate the resequencing procedure, but as time went by and the Recyclate Ltd. stripping facility remained idle Robert was put under increasing pressure by the Accountant to start stripping again. The result was that Robert announced that he would use his own subjective procedure for the Car6 #2 sequence.

The next day, Terry and the Accountant convened a database progress meeting at the Research Centre to discuss the dispute. The author's diary entry for the meeting reads as follows:

*"The day began with the Accountant collecting my [Car3] report. He was very pleased with the report and went through all the other reports asking for the addition of subtotals to facilitate checking. He went off to his next meeting in very good spirits quipping*

> *"You mustn't believe all you are told, must you?"*

*That afternoon we met [the author, Mary, Geoff, Terry and the Accountant] in Terry's office. The Accountant began by stating that he was now much happier that the database was delivering useful reports. It had been a long time and a lot of effort with no output to management (there's an important lesson here in making sure <u>all</u> our customers are served not just Robert!) The meeting progressed amicably with little flak for the database beyond the usual 'when will it be finished?' All the more reason to develop a process model to show the extent of our work.*

*The meeting ended with me persuading Terry and the Accountant to approve an experiment with NumberCrunch for data collection on Car7 [next model to strip]. I argued that QuickStore could only be used in a calm environment, and that hoping for this was too much therefore we needed something more robust. Geoff fought hard against this move as he sees our control of the database as a way of ensuring our control of data collection. I afterwards explained that this was at great cost to the stress levels of the database developers."*

10/2/93. Monthly progress meeting at Recyclate Ltd.

A sadly acrimonious meeting that seemed to undo all of the good teambuilding work that had taken place during the preceding two months. The meeting commenced with Robert citing "database problems" as the cause of the delay in stripping. This was doubly infuriating since the monthly meetings were the only time when the project managers, Terry and the Accountant, the Environmental Strategist and the other senior CarMaker engineers were informed of database developments. All they seemed to hear was that the database was the main source of project delay. Robert had run into database problems because he had used an untested version of the database in his rush to copy his resequenced Car6 #1 data to a Car6 #2 database.

The author had hoped to reduce the pressure on the database team by getting Recyclate Ltd. to collect data using NumberCrunch. This would solve many of their usability problems at the cost of reduced data validation on input. Terry and the Accountant had agreed to this idea the day before, but Robert adamantly refused to relinquish QuickStore at the progress meeting. He viewed this as "a step backward" and insisted that the way ahead was to improve the database.

Terry tabled his revised strip schedule which aimed to begin the "accelerated" stripping phase on 26 4 93. From this date only #1 strips of each model would be undertaken. The schedule assumed that "process improvement" (which included the database development for Robert) would be completed by

23/3/93. The schedule was to become a bone of contention between the project team and Terry, who refused to accept that the schedule was unrealistic and to change the predictions despite growing evidence of the inappropriateness of the plan. The damaging implications of abandoning the #2 and #3 strips was not realised at this stage. The #1 data set was collected for the purpose of thorough analysis of part and material content, not for speed of strip. Many of the operations could be accomplished more quickly if speed was the objective. However, the original, slower operations were recorded in the database. Reports that attempted to produce an optimised strip sequence from this data would therefore be based on the "slow" data and therefore could never produce a truly optimal time. This flaw did not become apparent until the autumn, by which time a number of #1-only models had been stripped.

The author left the meeting in a furious mood. He felt he was being blamed for poor progress, betrayed by trusted colleagues, and thwarted in his attempts to control expectations and the database workload. He resolved to ignore the Recyclate Ltd. data collection activity, Mary could support that if she wished, and instead to concentrate on the other customers.

11/2/93. QuickStore limitations impact "scientific" resequencing.

Following the author's defence of the database the day before, a QuickStore limitation was encountered that put an end to attempts to develop the Net Process Value report for automatic process resequencing. The report calculated the cost and value of each process and stored the result in a temporary variable: Net Process Value. The intention was to list all the processes in descending order of NPV. When the report was run the records came out in a different sequence. Calls to the QuickStore help-desk revealed that QuickStore could not sort on temporary variables, therefore the report had to be abandoned and with it our dream of fully automated resequencing. Geoff had been a staunch supporter of the database during the resequencing period, now even his patience was wearing thin.

## *** *15/2/93: Car6 #2 (Trial 2900)* ***

## *** *Stripping Commences at Recyclate Ltd.* ***

Using Robert's *ad hoc* strip sequence.

16/2/93. The author develops a set of DFDs for the IS using Analyser.

Data modelling had been used successfully to support database development to date, but the ERM could not show the variety of reports now provided nor the breadth of activities and users served by the database. The author therefore developed a set of levelled DFDs to show to the project management that the database had many successful attributes in addition to the failings emphasised by Robert. The top level DFD showed two processes:

1. Vehicle Analysis System
2. Volume Disassembly System

Process 1 had been the focus of the project so far. Process 1 was then exploded to show six sub-processes:

1.1 Disassemble Vehicles (C.E.D.)

1.2 Maintain Part Name Glossary

1.3 Analyse Materials

1.4 Maintain Financial Data

1.5 Assess Economic/Environmental Viability

1.6 Summarise Disassembly Data

Process 1.1 was exploded further to show six sub-processes and sixteen reports. Each of the reports was denoted by a numbered data flow, and a list of reports with comments was included as an appendix to the set of DFDs. Judging by the data flows, process 1.1 had consumed the bulk of the effort; this was Robert's area. The DFD therefore supported the author's argument that Robert had received special attention, and that the database provided extensive facilities. The fact that ten of Robert's sixteen reports were data validation reports was an indication of the quality problems present in data capture!

A usability problem was encountered when using Analyser for DFD drawing. The author wanted to eliminate one of the processes and merge the other two processes at the same level, thereby "raising" their sub-processes up one level. This would make the lower-level processes more visible and the DFDs more readable. Ideally, a cut-and-paste facility was required where the sub-processes could be pasted into the high level DFD. Analyser did not allow such cavalier actions and the author had to resort to deleting all of the subprocesses and redrawing them at the higher level, wasting half a day.

17/2/93. The author reviews "Data Backlog".

The author obtained copies of the early strip data from Robert and summarised the data in a table that showed that dependencies, activity names, part names and bin identifiers had to be updated for all trials up to the first Car6 (trial 27). Most of the data was on paper, some in NumberCrunch and none (apart from Car3 - trial 18) in QuickStore. Finally, none of the pre-Car6 data contained adequate material detail for Chris's needs. In short, the pre-Car6 data was largely useless!

### *** *18/2/93: Car6 #3 (Trial 3000)* ***

### *** *Stripping Commences at Recyclate Ltd.* ***

Recyclate Ltd. were moving ahead quickly with the strip schedule using their own subjective resequencing procedure that did not depend on the delayed dependency-based database reports.

18/2/93. The author creates the Problem Reporting Form (PRF) system.

As a further attempt to bring order to the chaos of changing user requirements and also to provide an "audit trail" of progress made, the author instigated a problem and change request logging system for the database sub-project. Every problem was to be logged by either the author or Mary. The author and Mary would review outstanding problems regularly.

23/2/93. QuickStore bug leads to crashing and loss of data.

Following the disappointment of not being able to sort on temporary variables and the resulting abandonment of the dependency-based reports, the team's confidence in QuickSt re was reduced still further by a series of crashes when backing up the database. On restoring the crashed backup, some of the forms were corrupted and data lost. A side effect of the bug was the creation of a 1.5 MB "scratch" file each time the system crashed. The files were only disc vered when the Recyclate Ltd. users complained of running out of disk space and found a set of strangely named files on disk.

****** 25/2/93: Car7 #1 (Trial 3100) ******

****** Stripping Commences at Recyclate Ltd. ******

## 4/3/93. Robert splits Car6 and Car7 data into separate databases.

As more trials were added to the database the response time was becoming noticeably slower and, more importantly, the lack of a "view" facility meant that it was easy to accidentally edit a record for a previous car instead of the current car. Mistakes were being made and the Recyclate Ltd. users were becoming increasingly frustrated.

Robert's latest idea was to create a new database for each model range instead of storing all models in a single database. This was superficially attractive and seemed to overcome some of his usability problems. However, as time went on, changes were made to the data in the "master" forms: part name glossary, bin details, materials details etc. and to the structure of forms and reports. In particular many new reports were added. Each database had its own, increasingly incompatible forms, reports and set of "master" data. The separate databases were soon suffering from the same incompatibility problem as the data held on the paper forms. The author and Mary tried to explain the potential maintenance nightmare to Robert, but Robert thought that the performance and usability benefits were worth the maintenance cost. By 5 March there were three separate databases:

- REC005-CAR6.
- REC006-CAR3.
- REC006-CAR7.

The author and Mary introduced the "RECnnn" numbering system to try to keep track of structural changes to forms and reports. In theory, Car3 and Car7 databases had the same structure, although the master data may have been different.

The multiple database nightmare spurred the author and Mary to devote time to seeking a technical solution. They investigated the cost of installing a LAN at Recyclate Ltd. with a remote node at the Research Centre to ensure that everyone was working on the latest copy of a database at any time. They also investigated QuickStore for Windows and another leading Windows database to see if they supported the "view" concept which would enable users to only see data from one car at a time if required. These sensible initiatives were never completed because of the constant pressure to support the installed database. It was a "catch 22" situation - the maintenance overload spurred the author and Mary to seek a solution, but they didn't have time to properly investigate the solution because of the pressures of maintenance!

Other activities that were draining resources at the time included:

1. Ongoing development of the "Optimiser" using both QuickStore and NumberCrunch. (Unintegrated CASE!)

2. Robert complained about excessive keystrokes and poor performance when using the on-line part name glossary and wanted to call in a QuickStore consultant to investigate all "database problems". The author and Mary reacted defensively to this proposal.

3. The author continued to investigate the data backlog to ascertain the changes required to salvage the early data.

### *** *9/3/93: Car7 #2 (Trial 3200)* ***

### *** *Stripping Commences at Recyclate Ltd.* ***

9/3/93 (am). "Data Status" meeting held at Recyclate Ltd.

Stripping was speeding up, but the database team was beginning to wilt under the pressure of responding to user demands. Terry, the Accountant and the Environmental Strategist were still seeing very few reports from the database and called a meeting with Geoff, the author and Mary to discuss the state of data collection and analysis. Chris and Robert were also present.

The meeting was calm and rational. Geoff began with a presentation of the disassembly process using the IDEF0 process modelling technique. The author followed by presenting his DFDs and argued that the database was now an extensive IS but that the majority of the IS resource had been devoted to Robert's work because he had "shouted loudest" and had immediate needs. Samples of stripping curves were tabled that showed the value and weight of polymers removed over time. The Environmental Strategist was pleased with these, but wanted further amendments. Chris looked at the data behind the curves and identified a number of inaccuracies which again cast doubt on the quality of the materials data being collected. The author asked if any new reports were required and was asked for only one new report, by Terry, showing the weight of polymer removed over time. The lack of communication between Geoff and the author was clearly illustrated by the choice of different diagramming techniques to explain the same process. They also used different names for processes and data flows, so the audience may have believed that they were talking about two separate systems instead of two views of the same system.

9/3/93 (pm). Monthly progress meeting at Recyclate Ltd.

Following the amicable morning meeting, the Environmental Strategist exploded in the afternoon meeting when discussions turned to the data backlog. He was under pressure from the Ministry and "outside bodies" to present evidence that the project was "doing something". He had asked for his "curves" in August and had seen nothing until today! Now the author was telling him that the strip data from August (Carl) to February (Car4) was unusable! He wasn't bothered if the data was realistic or not, he simply wanted the curves to "look right"!

The meeting was stunned by the realisation that the project had taking almost a year to learn how to collect data and that the process was still highly error-prone. The author exploded in response. The Environmental Strategist's voice had been "lost" amongst the noise of the project. This was not a "database problem", but again the focus was on the technology. Geoff calmed the Environmental Strategist down. Terry and the Accountant supported the author and explained to the Environmental Strategist that resequencing data to show an "optimal" land fill avoided curve was not a trivial matter because of the impact of dependencies which meant that some valuable parts could not be removed before seemingly less valuable parts because the valuable parts were dependent on the less valuable ones. Robert rounded on the Environmental Strategist and stated that he was not willing to let the Environmental Strategist "misrepresent" the physical reality of stripping to the public. He would rather delay publication until the "truth" could be portrayed. The Accountant attempted to mediate, and explained that the pressures to justify our work were "immediate". The author was amazed that these pressures were not communicated to the project team earlier! The management team had kept their distance and allowed the project to muddle on.

Attention now swung away from Robert and stripping towards the Environmental Strategist and "outward facing" reports. Car4 and Car6 data were imported into NumberCrunch and stripping curves produced showing strip profitability and land-fill avoided.

Interfacing QuickStore to NumberCrunch was tricky. No one on the project had done it before so the learning curve was steep. The problem of unintegrated tools that had become apparent in the New Model project surfaced again here as extensive further work had to be undertaken in NumberCrunch before the imported data could be plotted. Before long multiple, duplicated sets of data were scattered across PCs and between tools. Maintaining these data sets proved a further drain on the limited IS resources, and Robert still demanded support! Fr m the Environmental Strategist's point of view, all he had requested were "a few simple curves": the resulting workl ad was enormous.

*** *18/3/93: Car7 #3 (Trial 3300)* ***

*** ***Stripping Commences at Recyclate Ltd.*** ***

*** *24/3/93: Car8 #1 (Trial 3400)* ***

*** ***Stripping Commences at Recyclate Ltd.*** ***

Stripping progress was being delayed by the unavailability of Chris. At this time Chris and Recyclate Ltd. were evaluating an "electronic nose" that could automatically identify material content from a rubbing from a part. This would release Chris and enable stripping to progress more quickly. Unfortunately the nose had to be trained and produced many false diagnoses initially. Robert was still citing "database problems" as the major cause of delay, despite the fact that stripping was held up for days at a time if Chris was unavailable.

31/3/93. The first "Management Reporting" meeting at the Research Centre.

Following the Environmental Strategist's outburst at the monthly meeting it was agreed to hold a series of management reporting meetings to redirect project effort to providing publicity ammunition for management. The first meeting was held at the Research Centre and was attended by Terry, the Environmental Strategist, Chris, Geoff, the author and Mary. The Accountant could not attend therefore no Recyclate Ltd. staff were present.

The meeting was the first time that the author and Mary had had the opportunity to discuss information requirements with the Environmental Strategist at length. The Environmental Strategist was still exasperated that his "simple curves" were causing so many problems. The author tried to explain the difficulties of apportioning costs to individual parts due to the presence of dependencies that meant that parts and processes were related in a "tangled web", not a simple sequence. Resequencing a part was like pulling up a strand of spaghetti - the rest of the bowl (parts) tended to follow! The Environmental Strategist was not entirely convinced, but agreed to let the author produce what he could and to progress from there. A second meeting was arranged for a month later.

*The Stripping Curves*

6/4/93. Monthly progress meeting at Recyclate Ltd.

Another acrimonious meeting, despite the author and Mary's hard work and cordial relationships with the users between meetings. Robert began the meeting by listing five reasons for the slippage in stripping the cars:

1. Database backup problems led to crashes and loss of data.

2. Extra materials analysis was being undertaken at the request of Chris.

3. Car8 was the most complicated car stripped to date, with more parts and assemblies than previous cars.

4. The process of updating and synchronising the part name glossaries was very time-consuming.

5. Database response time and backup time was very slow.

Chris also cited the glossary maintenance as a major drain on his time.

The author again reacted angrily, partly because he had attended a very friendly data backlog meeting with Robert and Chris the day before and had no inkling of Robert's impending attack on the database work. He made the point forcefully that the database was a victim of the lack of teamworking and agreed procedures, especially for glossary maintenance. Ronnie would make changes, Chris would disagree, discussions would ensue and the changes would be altered. Sometimes Chris was not available. One of Chris's colleagues was supposed to own the glossary and be the final judge on naming conventions, but he was never available. The database was simply a tool in this process. It was the process that needed fixing, not the tool. By now the author was getting tired of having to defend his work at every monthly meeting. The author and Mary were constantly launching initiatives to help control development and to publicise their efforts to management, but still the database was the focus of criticism and ill-will.

Terry ruled that there would be no more changes to the database, apart from new reports. Recyclate Ltd. staff were not to alter the forms or reports, and the author and Mary would implement a change control procedure and log book. This was a step forward in that the author and Mary had management support. But Terry was not respected by Recyclate Ltd. and his dictat was likely to be ignored by Robert given the outstanding list of changes already requested.

The author presented Car4 and Car7 stripping curves showing optimum strip sequences for value/time and weight/time. The Car4 curves ignored dependencies because the information was not available. The Car7 curves included dependencies, although resequencing had to be done manually by Mary from an initial non-dependency report. The Environmental Strategist was pleased with the tangible progress and agreed that the curves had the "right shape". The author stated that a complete set of dismantling curves would be available for the next Management Reporting meeting, consisting of the Car3, Car4, Car6 and Car7 data (ignoring dependencies) and Car6 and Car7 data (including dependencies).

8/4/93. Robert FAXs the author and Mary regarding database performance problems.

Response time when saving changes to data on the Car8 database had become intolerable. Robert had timed a modification to an operation that disgorged four parts at 26 seconds between hitting the "save" key and the "record updated" message appearing on the screen on the 386 PC at Recyclate Ltd.. Many single-record changes were taking more than 10 seconds to complete which was slowing down data collection and validation.

The author spent several days investigating the problem and tried many unsuccessful solutions. The QuickStore help-desk suggested various possible causes, including too many "unique" key fields on the Part form. This was the cause. Before a new or modified record was saved the software checked that the set of "unique" fields contained a unique data value, thus ensuring that duplicate keys were trapped. The checking algorithm was obviously very slow, so an operation with four parts led to four checks, hence the lengthy delay. The author replaced the two unique fields in the Part form with a single, concatenated unique field and the performance returned to normal.

26/4/93. Second Management Reporting meeting at the Research Centre.

Light relief from the battles with Robert and QuickStore was provided by the stripping curve development, which, due to the power of graphics to convey information, was proving an enlightening and informative exercise. The curves "excluding dependencies" had the steep gradient that the Environmental Strategist was looking for. The curves "including dependencies" started well with the removal of the large, valuable battery, then plunged down in value as the fluids were removed (most fluids had a negative value), only to recover the "correct shape" later. Curves for Car4, Car6, Car7 and Car8 were presented together with supporting spreadsheets showing the part names, bins, weights and values. Chris poured over the data and identified various anomalies, but, in general everyone was impressed and pleased with the work. The highlight of the meeting was the Environmental Strategist's definition of a set of prioritised "research questions" that, for the first time, defined precisely the objectives of the project. The top six questions were as follows:

1. Which are the worst and best cars [re. recyclability] and why, and where is CarMaker?

2. How do old designs compare with modern ones?

3. What happens if material values go up as markets develop?

4. Can we decide simply what should be stripped? / what is a recyclable car?

5. How quickly do we need to be *able to strip to break even?*

6. What happens if we halve the number of material types [in new cars]?

The simplicity and clarity of these objectives, in contrast with Terry and the Accountant's highly political leadership, astonished the author and Geoff. Terry saw achieving his strip schedule as the project "goal", but the research questions were far more credible. The author and Geoff wondered why the questions couldn't have been stated months ago, before the project disappeared into a jungle of detail and confusion.

### *** *28/4/93: Car8 #3 (Trial 3500)* ***

### *** ***Stripping Commences at Recyclate Ltd.*** ***

5/5/93. QuickStore v2 installed at Recyclate Ltd.

The new version of QuickStore provided mouse support. The author and Mary also added colour-coded lookup fields, mandatory fields and display-only fields to help the users determine the options available on data entry. These changes were welcomed by Recyclate Ltd. and Jim soon became adept at "drilling down" the process part hierarchy using the green lookup fields.

12/5/93. Monthly progress meeting at Recyclate Ltd.

A less acrimonious meeting this month. The author and Mary tabled the extended and revised set of stripping curves. Unfortunately, Robert again cited the database as the major source of stripping delay in his report to the meeting:

*"As the amount of data has increased, so the system has become slower, particularly when saving and updating records and changing between data forms. Printed reports now take in excess of an hour each and copy routines, several hours.*

*A five page 'Procedure' has been introduced which guides us through the various stages and helps to avoid some of the pitfalls of an unfriendly database."*

He went on to state that

*"Significant savings can be made by:*

*Installing QuickStore v2 which is mouse driven.*

*Change the database and remove the process function.*

*Change the database and assign dependencies to operations, not processes.*

*Make amendments to all data procedures (reports).*

*Change the dismantling procedure to reflect the alterations.*

*Further savings could be made by one or more of the following:*

*Reducing the number of parts removed.*

*Reducing the materials analysis.*

*Not applying a reduced sequence [for #2 and #3 cars].*

*Less copy trials. "*

The author again found himself defending the database and arguing that the technological myopia of Robert missed the key issues. The list of "significant" savings was insignificant in comparison to the savings possible from the "further savings" list. However, Robert's complaints were valid, and it was agreed to proceed with the list of changes identified. In response to Robert's other points, it was agreed to target c. 50 parts in order to speed up stripping. Again, much of the meeting was spent discussing how the strip schedule targets could be achieved, rather than whether we had the answers to the Environmental Strategist's research questions!

17/5/93. QuickStore failings listed.

The author reflected on QuickStore failings in the project diary:

*"1. No graphing facility leads to duplication of data sets leads to change control problems (especially since formulae have to added in NumberCrunch).*

*2. Intermittent crashes/problems:*

*(i) Glossary sort routine fails to output records to disk, glossary is therefore deleted.*

*(ii) Various crashes when run under Windows.*

*3. Slow performance with two field relationships encourages "techie" optimisations which lead to maintenance problems.*

*4. 'Unique' fields cripple performance [but are essential for data integrity!]*

*5. Cannot view process table view records in a sorted order, leads to elaborate routines (which often fail) to sort glossary, operations etc.*

*6. No 'impact analysis' facility, therefore difficult to trace side-effects of changes to forms.*

*7. Windows version does not interface properly to v2 forms.*

*8. Cannot customise function key line in v2 (users have to remember complex sequences of keystrokes)."*

Furthermore, Recyclate Ltd. over-emphasis on the database may have been due to it being their main tool, which they used every day.

## *** *18/5/93: Car9 #1 (Trial 3600)* ***

## *** *Stripping Commences at Recyclate Ltd.* ***

Car9 represented the start of the "accelerated phase", to save time the #2 and #3 strips were to be discontinued. This meant that the opportunity for process optimisation was lost; the repercussions in terms of data analysis *were not* appreciated at this point. Furthermore, this was a clear example of the dominant driving force being Terry's strip schedule and not the Environmental Strategist's research questions. In fact, the loss of the #2 and #3 data seriously impaired the project team's ability to answer any of the research questions.

*The Book of Curves*

26/5 93. Third Management Reporting meeting at the Research Centre

The author worked at the Research Centre until 11 pm the night before the meeting to print off copies of the "Book of Curves . Instead of a scattered set of graphs, the author and Mary had produced a bound report containing an introduction, a set of curves for Car3, Car4, Car6, Car7 and Car8, and a set of comparison curves showing all five cars. Furthermore, the report included a "what if" curve for Car8 showing strip profitability given forecast maximum (virgin) market values for polymers. The author and Mary were extremely proud of their publication. The book neatly summarised several months work and seemed to help answer a number of the Environmental Strategist's research questions. The book was well received by the Environmental Strategist and Terry. Nonetheless a number of improvements were requested including:

- A virgin polymer report for Car8 showing dependencies.

- A more sophisticated financial model that included transport, granulation and other costs.

- A comparison of the five cars showing only the first 40 minutes stripping (no dependencies), since the bulk of the valuable polymers were removed within 40 minutes.

- A Net Part Value report which could generate a picking list of parts including dependencies.

The "Holy Grail" of automatic resequencing had been abandoned in February when the QuickStore sorting problems seemed to rule out further progress. Now, in May, our experience with QuickStore was much greater and our confidence was renewed due to the progress made with the curves. The "picking list" now seemed achievable, although the complexity of the dependency tree was still an issue. The author tried, unsuccessfully, to convince the Environmental Strategist that the picking list was not straightforward. But this point was lost on the Environmental Strategist, who admitted he'd never used a spreadsheet in his life, yet alone a recursive programming language!

27/5/93. The author observes data entry at Recyclate Ltd.

The author and Mary had been visiting Recyclate Ltd. regularly for several months, maintaining the database on-site. Despite the occasional wrangle at monthly meetings, they had a very friendly relationship with the Recyclate Ltd. staff. During this time they had generally worked in the office and had not viewed data capture in the workshop. Today the author sat alongside Jim as a vehicle was stripped and operated the database himself for a while. Jim had been complaining of usability problems and the author could see clearly that improvements were due:

1. There was no automatic carry over of trial number, operation number, activity or assembly number when operations were being entered. The user had to write the previous number down on a piece of paper and re-enter where necessary.

2. When entering new part numbers, the first part of the part number was automatically generated, but changing the part number letter suffix in the subform was error-prone and tedious. [The part number consisted of the operation number plus a letter suffix e.g. operation 123 could disgorge parts 123A, 123B, 123C etc.]

Furthermore, the noise level in the workshop was high due to power tools, hammering, and the impatient comments of Ronnie and Chris as they waited for the database operator to record the operation.

The author spent the next few days working on improving the usability of the database. Robert asked for a second suffix letter for part numbers to allow more than 26 parts to be recorded against a single operation. This was part of the general move towards reducing the amount of data collected, but had the unforseen side effect of reducing the timing granularity for the "picking list" report. It was very difficult to be precise about allocating time to a part when the part was one of thirty removed in a single, five minute operation! The "cherry" parts were therefore bundled in with the "chaff" and it became impossible for the analysis reports to separate the two. Other unhelpful short-cuts included leaving the operation description field blank and only timing operations that disgorged targeted parts ("cherries"). Both developments were to make a mockery of the data analysis reports for the later cars.

Further work at Recyclate Ltd. saw the author attempt to set up a number of "intelligent defaults" to save the operator having to key in data. The defaults were targeted at the "materials analysis" operation, which was a convoluted concept dreamt up by the author and Geoff in January to save altering the database structure at a time of great pressure and overwork. Instead of recording a material directly against a part, the operator had to create a further operation to "analyse the material", with time set to zero, and log the materials as "parts" disgorged by this operation. As the quantity of materials analysis increased the procedure became irksome for the operator. The defaults were an attempt to reduce the burden on the user but soon became very complicated and were more trouble to maintain than they were worth. Eventually the author realised the only solution was to redesign the data model and database so that materials could be entered directly against the part, eliminating the redundant "materials analysis" operation.

****** 9/6/93: Car10 #1 (Trial 3700) ******

****** Stripping Commences at Recyclate Ltd. ******

10/6/93. Geoff is tasked by Terry to write the Final Ministry Report.

After a year of wayward leadership, Terry delegated arguably the most important task in the entire project to Geoff: writing the final report. Geoff and the author discussed the report and Geoff outlined a report structure based around two research questions:

1. How can an existing dismantler upgrade? [to recycle cars]

2. What are the opportunities to increase the recycling rate?

Geoff was determined to write a rigorous report and thought that Terry would not argue with his conclusions. Geoff had developed his own NumberCrunch IS using data collected from other sources. He had used the Recyclate Ltd. strip data to validate this data, but the strip data was not central to his analysis. After over a year of acrimony and stress, the author felt rather deflated that most of the detailed database information was not going to be used. Was it worth the pain after all?

24 6/93. Monthly progress meeting at Recyclate Ltd.

The author was away that day. In his absence, the meeting agreed to "freeze" the database (for the second time).

28/6/93. Materials Analysis meeting at the Research Centre.

Chris was becoming increasingly agitated at the complexity of the database - he simply found it very difficult to extract materials data. A meeting was called at the Research Centre to resolve these issues. The outcome of the meeting was an agreement to revise the database to enable many materials to be entered directly against a part instead of via the convoluted "materials analysis" operation.

### *** *30/6/93: Car11 #1 (Trial 3800)* ***

### *** *Stripping Commences at Recyclate Ltd.* ***

*The Archive Arrives: Erratic Bin Allocation Revealed*

The major change to the database structure due to the new method of recording materials directly against parts meant that all of the preceding vehicle databases had to be restructured to correspond to the Car11 database. This was a major task, given that there were now seven separate databases. The author and Mary devised the "archive" concept to overcome the problem of maintaining consistency of structure. Once a model was stripped and validated by Recyclate Ltd. they handed the disk over to the Research Centre where it was merged with the preceding databases into a single database. Once merged, changes to the structure or data could only be made at the Research Centre. After four months of separate databases, the maintenance nightmare was almost over. The original motivation for separate databases was poor response time and usability at the point of data capture (Recyclate Ltd.). The new solution still allowed Recyclate Ltd. to collect one model per database, but kept the rest of the models at the Research Centre, where response time was not so important since we were not using the database for data capture.

The archive database contained the latest set of reports, including a new materials analysis report written by Mary for Chris. As each vehicle was merged into the archive, the report was run to analyse the quality of materials data captured. The report showed that Recyclate Ltd. had been placing many parts in the "Contaminated Materials" bin when a more accurate classification was expected by Chris and Geoff. The contaminated materials bin had a negative market value, thus understating the value of the parts removed, and the bin totals for the car. This was further evidence to support Chris and Geoff's allegations that the quality of the data collected was poor. This, in turn, undermined the worth of the database. A time-consuming data correction exercise was initiated to review all contaminated materials data and correct the records as appropriate. This took up more of Chris's limited time, and further encouraged Geoff and Terry to pursue alternative sources of data for the Ministry report.

### 13/7/93. Chris FAXs the author with revised materials reporting requirements.

Chris gave his original list of four materials reports to the author and Mary in December 1992. It was now July 1993, and the four reports had still not been provided due to work overload and poor data quality that had meant that the base data was not available in sufficient detail to provide the information Chris required. The database structure for materials was now considerably improved and the quality of data was being improved with the contaminated materials review. Chris had updated his original report request as follows:

1. List of vehicle composition by material standard code, showing parts and weights.

2. List of part composition showing material content down the parent-child hierarchy.

3. List of vehicle composition by material class, showing parts and weights.

4. List of tools and methods used to remove a given part.

The author took responsibility for these reports because he felt that they were high priority and required significant programming skills in traversing the parent-child hierarchy. Mary was less skilled in this area.

*Reduced Data Collection*

27/7/93. Monthly progress meeting at Recyclate Ltd.

Chris stated that he had completed his strip manuals for two more of CarMaker's vehicles: Car9 and Car10. This was part of CarMaker's commitment to the AutoStrip project (see entry dated 20/5/92), and was one of Chris's major activities. He was using the data collected by Recyclate Ltd.

to validate his manuals and hence was concerned about data accuracy. The author was vaguely aware that the strip manuals were at the root of Chris's concern regarding data collection, but rarely saw a manual.

Robert stated that four and a half weeks had elapsed since the start of the Car11 strip, whereas only two weeks had been scheduled. This was due to "... sheer volume of parts being analysed, *interruptions (e.g. filming)*, materials analysis and [electronic nose] problems and the discontinuity thus caused." The author was relieved as this was the first time that Robert had not cited the database as the major cause of delay!

Terry provided a revised strip schedule that stated that around 50 parts were to be identified prior to stripping and full operations data (strip time, method, tools, dependencies etc.) to be collected for these parts only. No operations data was to be collected for the remaining parts, only part name, weight and material content. Furthermore, "Cars not completed in their program period will be disposed, and the next vehicle continued with." These measures would speed up stripping and focus data collection, however, they would also reduce the amount of information available from the later cars, such as the time taken to strip certain non-target parts made from important material and the time taken for operations that were in the dependency chain of target parts. Chris was to ask the author for this information later! Chris had won his battle with Terry and the Accountant regarding the strip sequence. The new schedule showed that new CarMaker models were to be stripped ahead of the more popular ten year old cars to ensure that CarMaker understood their own cars before the project ended.

16/8/93. Mary sends "40 minute" stripping curves to the Environmental Strategist, Terry and the Accountant.

Stripping curve production was still absorbing a lot of the author's and Mary's time. As vehicles were added to the Archive database, Mary would generate output files for NumberCrunch and spend some considerable time manipulating the data in NumberCrunch and printing out the curves. Little progress had been made on automating the process. The curves were then sent to the project management with a request for comments. No replies were received apart from a brief "thankyou" and "can we have some more?" at the monthly meetings. Given the amount of effort expended on producing the curves, the author and Mary felt somewhat deflated at the lack of feedback.

### *** *31/8/93: Car12 #1 (Trial 3900)* ***

### *** *Stripping Commences at Recyclate Ltd.* ***

In a surprising about turn, Terry's July 1993 strip schedule, which had seen CarMaker vehicles moved to the head of the list, was dramatically altered to strip the more popular cars next, followed by a re-stripping of Car3, Car4 and Car7 due to the poor quality of the original strip data. The new CarMaker vehicles were pushed to the end of the schedule.

The author had taken on a time-consuming new activity: to write a "Copy Part Tree ' routine for Recyclate Ltd. that would enable them to enter the part number of a part that had already been removed, and automatically copy the part record and any child part records and as ociated materials and operations records to a new set of records, incrementing the key fields, and descending the part tree as it went. The routine would save *Recyclate Ltd. rekeying similar data* for similar parts, for example left and right front doors, and would therefore peed up data capture. The routine was very difficult to write using the QuickStore query language which did not support nested queries.

### *** *15/9/93: Car13 #1 (Trial 4000)* ***

### *** *Stripping Commences at Recyclate Ltd.* ***

*Picking List: The Final Glory*

22/9 93. Monthly progress meeting at Recyclate Ltd.

Geoff tabled a copy of the latest European Community directive on vehicle recycling dated Brussels, 12 August 1993. The directive stated the following targets:

*"- Maximum of 15% landfill by 2002 per car, based on the new car weight,*

*- Maximum of 5% landfill per car by 2015, based on new car weight,*

*- Depolluted shredder residue so that it will not be classified as hazardous waste.*

*The targets include the possibility of energy recovery (combustion)."*

and went on to address design for recycling:

*"Car manufacturers will ensure that, by the year 2002, their new models may be reprocessed to generate a weight of final waste not exceeding 5% of the vehicle's initial weight."*

The directive did not make clear whether the target applied to new vehicles being produced in 2002 and 2015 or to end-of-life (c. ten year old vehicles) at those dates. Nonetheless, the directive spurred Geoff and Robert to investigate energy recovery (burning) as a more cost-effective alternative to segregation of materials and recycling. This would lessen the need for precise information on the material content of the car, since all the major non-metal parts would be removed, placed in a single bin and subsequently burned. At a single stroke of the Brussels pen, car recycling could become a far less information-intensive activity, with a concomitant reduction in the need for an information system!

Meanwhile, as the database became increasingly robust and the reporting facilities became increasingly sophisticated, Terry, the Accountant, the Environmental Strategist and Robert seemed to be losing interest in our work. Robert continued to question the relevance of the non-dependency stripping curves that the author and Mary were continuing to issue for each new car, claiming they presented an unrealistic and misleading picture to the outside world. Recyclate Ltd. demanded that the author and Mary spend two days a week working as database operators so that Recyclate Ltd. staff could be deployed on other, more lucrative Recyclate projects. This threatened to jeopardise the author's plan as the long awaited "Picking List", that would automatically suggest the optimum strip sequence for a car given dependencies and current material market prices, was nearing completion. The author believed that this would be the final glory for the database team (he was leaving the Research Centre at Christmas) and would represent a supreme triumph in algorithmic sophistication and provide a unique decision support system for vehicle recyclers. Geoff supported this view and wrote in the minutes of the meeting that

*"Robert, Geoff, the author and Mary will meet to discuss how to generate Pick Lists for cars. The ability to do this is a Primary Deliverable from the project."*

However, what constituted the "primary deliverables" from the project was a subject of some debate. Geoff believed that Terry already had sufficient data from his own sources to present to the Ministry. He controlled dissemination of information to the Ministry and presented highly-aggregated information and did not describe his sources. This was one reason why he was no longer interested in the strip data or the database reports and also why he was so exasperated with Chris and Recyclate Ltd. insistence on recording stripping data at a detailed level. Geoff and Chris continued to show interest in the database work; Geoff because he believed the Picking List was a deliverable; Chris because he wanted the data and the Picking List report to fine-tune his strip manuals and provide AutoStrip with factual data on "best stripping practice".

*** *24/9/93: Car4 #1 (Trial 4100)* ***

*** ***Re-stripping Commences at Recyclate Ltd.*** ***

*** *7/10/93: Car3 #1 (Trial 4200)* ***

*** ***Re-stripping Commences at Recyclate Ltd.*** ***

Car4 and Car3 were re-stripped due to the poor quality of the data collected from the original strips in 1992.

14 10 93. Picking List report grows.

The author was now working frantically on the Picking List, hoping to complete the report (or "tool" as he was now calling it) before he left the Research Centre. For most of the project he had considered tool implementation an impossibility due to the lack of recursive programming facilities in QuickStore. After eighteen months of intensive QuickStore programming experience, he had discovered a solution: to split the report into several separate reports (stages), each stage writing to file so that the next stage could read the intermediate results and reprocess the database in a different sequence, possibly using different tables. The "iterative" approach worked well, but took several minutes to run per vehicle.

Mary was providing most of the on-site support to Recyclate Ltd. as the author was still working half-time and wanted to devote his time to the Picking List. Mary reported that the Recyclate Ltd. staff were having far fewer problems entering data since a one-hour, informal "training" session by the author in July where he described the data model and the relationship between the model and the data collection screens.

27/10/93. QuickStore programming limitations.

Chris was pleased with his four materials reports (see 13/7/93 entry) and wanted the tooling report run for every part on Car9. The author had originally written the report to explode the part tree for one part. Unfortunately it was not possible to put the report inside a "for every part" loop, as would be done with a programming language, because QuickStore would not allow procedures to be called by iteration statements. Chris's colleague had to run the report over twenty times during the following weeks to get the information which could have been produced in a single run. This was an extremely time-consuming activity with each run printing many pages of data and taking over an hour to execute on the Archive PC, which now held data on several cars.

2/11/93. Monthly progress meeting at Recyclate Ltd.

The Ministry project officer attended his second meeting. The author recorded his view of events in the project diary:

*"[The Project Officer] stated ... that our project was 'exemplary' and showed far more results and action than virtually any other project he had seen! Terry and the Accountant proudly claimed that the project had largely kept within time, budget and deliverables despite some astonishing events regarding the detail of analysis required, and made no attempt to praise/thank the team for their efforts. They implied that the project had succeeded due to <u>their</u> efforts and despite <u>our</u> efforts! Tragic; this Ministry praise will simply justify Terry and the Accountant's management style!"*

Later on the Environmental Strategist stated that he was "very pleased" with his curves. The author and Mary asked the Environmental Strategist, Terry and the Accountant to meet them to discuss reporting issues outstanding.

17/11/93. Picking List tool shows a profit.

The author applied the Picking List tool to different cars and showed how it could be used to automatically generate a profitable strip sequence. The author was delighted with his work, and Mary very impressed. No one else seemed very interested.

*Final Thoughts*

The last entry in the project diary was dated 25 November 1993. The author left the Research Centre at the end of December 1993 and focused his efforts almost entirely on the Picking List for the last few weeks, leaving Mary to support Recyclate Ltd. and provide *ad hoc* reports to the other team members The diary entry shows that a "Picking List/Curves" meeting was held at Recyclate Ltd. in early November and that those present (Terry, the Accountant and Geoff) "couldn't see the point of the tool"; the auth r was away that day. Mary persuaded Terry to arrange another meeting to look at what had been done, especially since a major customer, the Environmental Strategist, was not present. Terry agreed to re-arrange the meeting but never came back to Mary or the author with a date.

Thus ended the author's involvement with the Recycling project. The evolutionary development approach that seemed so attractive after the analysis paralysis of the Warranty project had proved to be a "black hole" which drained resources and sapped the morale of both the author and Mary. The IS requirements had simply been too great for the resources available, and the project confusion and politics led to scapegoating of the database because it was one of the few tangible elements of the project. The IS represented an attempt to clarify ill-thought out ideas. Once these ideas were embodied in software they were visible and open to criticism. Unfortunately, it was the software and not the ideas that bore the brunt of the criticism!

# 4.4 Metrics and Stakeholder Views

The research question driving the project was

> *"Do CASE tools enable quality systems to be developed quickly?"*

In this section metrics and stakeholder views from the Recycling project are presented. The data is not discussed in detail here, instead, the results from all three projects are discussed in Chapter 7.

## 4.4.1 Productivity Assessment

Like the New Model project an information system was implemented, used and maintained during the Recycling project. Again, a PC-based, fourth-generation development environment was used. Like FourthGen, QuickStore provided a pre-defined environment for IS development but benefited from a less rigid, less idiosyncratic and more flexible set of facilities, including a query language. The New Model project followed the waterfall model of systems development (Boehm, 1981), with requirements being analysed and defined by data flow diagrams and an entity-relationship model before the software was developed. In contrast, an evolutionary development approach was adopted for the Recycling project in order to avoid the "analysis paralysis" of the Warranty project. An ERM evolved in parallel with the software, but DFDs were used only occasionally in order to clarify the scope of the work and to sell the database achievements to managers and users. Function point analysis is difficult to undertake given the combination of changing system size and lack of DFDs, therefore no productivity figures are available for the Recycling project.

### 4.4.2 Quality Assessment

Again, in common with the New Model project, a working system was implemented and used. This allowed quality to be assessed using quantitative techniques as well as qualitative techniques. The author and the Consultant knew very little about quality metrics at the start of the New Model project, and quality assessment was based on a post-project analysis of defects and change requests gleaned from the project diary. In the year between the end of the New Model project and the start of the Recycling project, the author became more familiar with the software metrics literature, and during the Recycling project implemented a formal Problem Report Form (PRF) system for recording defects and change requests and conducted a User Information Satisfaction (UIS) survey to obtain both quantitative and qualitative data on perceived IS quality. The two sets of metrics are presented in the next section.

### Summary of Defects and Changes

The Problem Report Form (PRF) system was instigated in March 1993 to enable the author and Mary to monitor problem resolution and co-ordinate their efforts. The system was used up to the end of July 1993 with defect and change data recorded for this period. A full list of defects and changes that occurred during this is presented in Appendix 4.2, with a summary presented in Table 4.2.

| | Priority | | | |
|---|---|---|---|---|
| | High | Medium | Low | Total |
| Defects | 6 | 2 | 0 | 8 |
| Changes | 1 | 17 | 3 (1) | 21 (1) |
| Total | 7 | 19 | 3 | 29 |

**Table 4.2** Summary of Defects and Changes: Recycling Project

Note: The figures in parentheses for change requests denote requests pertaining to reporting changes that, in the author's opinion, could have been carried out by the users.

Like the New Model project, several serious defects were reported and many medium priority change requests. The six high priority defects were:

1. No glossary sort facility (PRF# 16).
2. No process number propagation (PRF# 19).
3. Poor response time (PRF# 23 and 24).
4. Failure to include low level operations in worklists (PRF# 28).
5. Duplicate records generated on data entry (PRF# 30).

Unlike the New Model project, none of the six defects were due to bugs in the system software. The defects were due either to omissions by the author and Mary that were later fixed (2 and 4) or to attempts to add facilities that QuickStore should have provided in the package (1, 3 and 5). Of the 21 change requests, only one was for a change to reports that could have been carried out by the users. Most of the report requests were for complicated reports that required specialist programming and could not be carried out by the users. In this respect the project differed markedly from the New Model project.

### User Information Satisfaction Survey

The author and the Consultant prepared a list of questions to structure the New Model and Warranty project review interviews. Following these projects, the author reviewed the literature on user satisfaction assessment and discovered the work on UIS surveys which would enable user responses to be quantified in order to compare user perceptions. A quantified instrument was developed by the author, based on the UIS work of Bailey and Pearson (1983) and the QA Forum (1989), and given to the stakeholders in the Recycling project for completion and return to the author; the author did not conduct interviews with the stakeholders. Six out of the eight stakeholders completed the questionnaire. The Accountant and the Environmental Strategist did not, despite reminders from the author. The responses are analysed below.

**Summary of Stakeholder Views**

| Stake-holder | Positive Comments | Negative Comments |
|---|---|---|
| Geoff | Good support from the author and Mary | Usability "eccentric" |
| | IS reliable; information relevant, precise and complete | |
| | Confident in developing own reports | |
| | Overall very satisfied with IS | |
| Chris | | Not involved sufficiently |
| | | Unable to use IS; user interface "unfriendly" |
| | | IS development needed more resources |

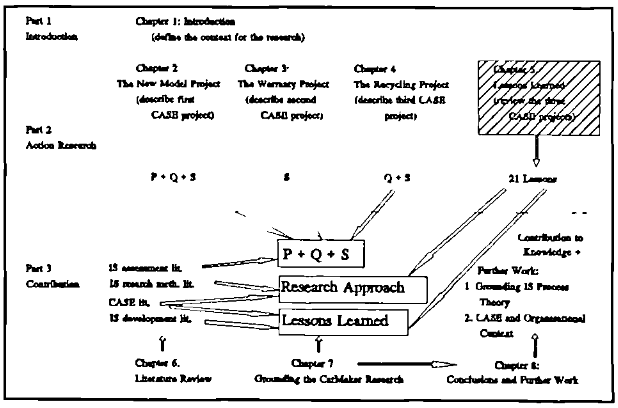
| **Stake-holder** | **Positive Comments** | **Negative Comments** |
|---|---|---|
| | | No benefits from IS to date |
| Ronnie | | Not a Windows product! |
| | | Observed that usability was poor |
| | | Report layout poor |
| | | Documentation non-existent |
| | | Had no control over IS development |
| Jim | Satisfied with participation | Limited understanding of IS capabilities |
| | | IS very inflexible, failed to meet expectations |
| | | Permanent on-site support needed |
| | | Upgrade to Windows version as soon as possible |
| | | Database consultant needed |
| | | Error recovery poor |
| | | Can't generate own reports |
| | | Documentation non-existent |

| **Stake-holder** | **Positive Comments** | **Negative Comments** |
|---|---|---|
| Robert | Very satisfied with participation | More on-site support needed |
|  | Good understanding of IS capabilities | Data modification very difficult |
|  | IS very flexible | IS difficult to learn |
|  | Information very accurate, precise and readily available | Better training required |
|  | Very good response time |  |
|  | Data entry straightforward |  |
| Terry | Good relationship with IS developers | More time required "up front" for information analysis |
|  |  | Perceived IS as inflexible |
|  |  | Surprised at IS development resource requirements |
|  |  | Perceived IS as unreliable, difficult to use and with poor response time |
|  |  | "Seems to have been the single largest problem on the project" |

**Table 4.3** Summary of Stakeholder Views: Recycling Project

# Chapter 5

# Lessons Learned



## 5.1 Introduction

The three CarMaker projects were undertaken in the spirit of action research, but without reference to theory. In other words, they were not well-grounded in theory. The stories described in Chapters 2, 3 and 4 were extracted by the author from a substantial collection of documents generated by the projects and also from the project diaries. The mass of data had to be edited severely to avoid overwhelming the reader with detail, and yet at the same time the important events and the richness of the stories could not be lost. The choice of material to include and material to leave aside was based on the author's subjective judgement of "relevance" to the conduct and outcome of the projects. This is in keeping with the approach adopted by Boland and Day (1989) and Newman and Robey (1992) who rely upon subjective judgement when determining relevance. No attempt was made to apply a theory to the data in order to interpret events in a particular light. Therefore this chapter describes the author's atheoretical interpretation of the data.

## 5.2 Lessons Learned

The following lessons were identified by revisiting the three stories and extracting the events that appeared most important with hindsight. Similar events were then grouped together and summarised by a "lesson". Twenty lessons were identified by this process. A second, higher-level classification was then applied in order to group the lessons into meaningful categories. Again, the choice of category was a subjective one undertaken by the author. It is hoped that the category headings will facilitate comparison of the lessons with the literature reviewed in the next chapter.

Even a simple, high-level classification scheme presents problems. Many lessons could justifiably appear in more than one category. For example, lesson 11 detailing the effects of prototyping, has been presented under the heading "Cognitive Limitations", but could arguably reside under "IS Methodology". Similarly, lesson 16, describing the difficulty of persuading users to give time to a systems project when they already have a busy schedule of day-to-day work could be classified as a "Cognitive Limitations" issue instead of the "Culture" classification used here. Clearly, no single classification scheme is likely to satisfy all interpretations of such rich material.

### 5.2.1 Category 1: Politics

The word politics is used here to denote the way in which individuals act in order to obtain or preserve power in an organisation. Sometimes the actions are overt, such as Terry's interventions in the Recycling project; sometimes they are covert, such as the Simon's company-wide analysis of problem management in parallel with the Research Centre/ToolVendor study in the Warranty project. Politics is seen here as a negative use of power from the author's perspective. Clearly the individual who is acting "politically" may interpret his actions in a more positive light if challenged. Since the individuals concerned were rarely challenged, the interpretation given here is the author's, with all the bias and limitations that that implies.

*Lesson 1*

*Certain stakeholders can dominate projects and wield power to divert projects to serve their own interests. This can lead to unrealistic overselling of project benefits.*

For example, the Consultant pursued his own and ToolVendor's interests in both the New Model and Warranty projects in that he used his greater experience of IS development to limit the scope of the New Model project to avoid time-consuming investigation of alternative systems (such as the South Factory Warranty system), and to ensure that a significant amount of effort was devoted to exploring the technical issues relating to the Analyser/FourthGen interface. Once the interface had been understood, and a report written that showed ToolVendor as a pioneering CASE developer in search of objective evidence of tool benefits, the Consultant withdrew his effort from the project. This left the author to support and maintain the system and to discover the various FourthGen bugs that had not come to light prior to the publication of the report.

Similarly, Jack used the Warranty project to strengthen his position at East Factory. He learned analytical techniques and used the detailed analysis facilitated by the author and the Consultant to identify opportunities for his own AllTogether system development. He was also the official liaison between Warranty and the corporate problem management initiative. Neither project was discussed with the author or the Consultant despite the overlap of interests.

The Recycling project was a minefield of political manoeuvring. At the heart of the problem lay the different objectives behind Recyclate's and CarMaker's involvement. CarMaker wanted to strip their own, current vehicles to ensure that future designs were more recyclable. Recyclate Ltd. wanted to strip older, popular models from all manufacturers in order to assess the profitability of a new business opportunity: a vehicle recycling network. The CarMaker manager, Terry, was an unpopular character who eventually lost out to the greater persuasive power of the Recyclate Ltd. manager, the Accountant. High selling vehicles from many manufacturers were therefore stripped ahead of the new CarMaker models. A sub-plot was Terry's insistence that his strip schedule be used as the sole measure of progress. The Accountant agreed to this, and, despite frequently acrimonious monthly meetings, the rest of the project team were forced to strip cars more quickly than our understanding of the process or the data should have allowed. The end result was the collection of vast quantities of flawed data that was eventually disregarded by Terry and the Accountant.

*Others "hide in the shadows" or are defensive when confronted with new initiatives with which they have not been closely involved.*

The CarMaker IT Strategy team, in particular the IT Strategy Director and Simon, remained a shadowy presence during the first two projects. The Director had personally arranged the initial introductions, but showed no further interest in either project. This was symptomatic of a greater malaise: the lack of involvement of the Research Centre IT research fellows in the CarMaker IT Strategy process. Without CarMaker IT support, both projects were weakly anchored in corporate change initiatives. The New Model system should have been positioned within the South Factory systems portfolio and had full-time CarMaker staff maintaining it instead of a part-timer: the author. Similarly, Simon should have explained the corporate problem management initiative to the author instead of blaming Jack for bad communication.

The author's occasional conversations with CarMaker IT staff, such as the data analyst at East Factory during the Warranty project, showed that the staff were not aware of the Research Centre projects, and were defensive over the "threat" to their territory from non-CarMaker personnel.

Lesson 2

*Communication problems and political differences exist within the IS user community as well as between developers and users. In other words, users are not homogeneous.*

For example, the CarMaker re-organisation from functions into strategic business units during the New Model project meant that David was no longer available to provide IS support to his erstwhile colleagues at South Factory. In the same project, Stuart was surprised that the West Factory New Model staff had not informed the author and the Consultant of the eventual need to interface to the Warranty system. The project also showed that users are subject to the influence of "outside" stakeholders. In this case, SystemsHouse exerted considerable influence at South Factory, whereas David and Tom had been concerned to avoid using the company at West Factory.

The author's dismay with the attitude of the CarMaker IT Strategy team was echoed by the Sponsor and Andrew (New Model project) who were surprised that the team had not kept in touch with South Factory developments as there was much to be learnt.

The Warranty project was launched against a backdrop of conflict and mistrust within Warranty. The Sponsor championed the project without the support of his peer, Tony, or their boss, the Warranty Director. The lack of top-level support meant that the other managers rarely made their staff available during factory hours, hence the Friday lunchtime meetings.

The conflict between users in the Recycling project was described in lesson 1. This conflict propagated throughout the project and resulted in regular battles at the operational level, particularly between the Recyclate Ltd. stripping staff and Geoff and Chris as "data customers". The author was also deeply immersed in the conflict, generally favouring the Recyclate Ltd. camp due to its greater openness.

*Likewise, the developers do not always agree.*

The Consultant tended to lead the first two projects, whilst the author learnt about systems development. Nonetheless, the author was often concerned about the Consultant's narrow focus and pro-ToolVendor stance. For example, the Consultant dismissed the author's attempt to follow the MiniMethod structured method at the start of the New Model project, and "softened" a number of the author's comments in the final project report for the New Model project.

### 5.2.2 Category 2: History

Everyone is influenced to some degree by their previous experiences. Past experience can be used to short-cut the decision making process. If a new situation appears similar to a situation encountered in the past, the experience of the past event can be brought to bear on the current situation. This may well lighten the cognitive load, but can result in prejudice. The legacy of past IS failures can confront developers of new systems, in that users who have experienced IS failure in the past may well be wary of future IS investment.

Lesson 3

*Past IS experiences, particularly bad ones, can influence user behaviour and expectations of new IS projects.*

For example, the West Factory New Model staff's bad experiences with SystemsHouse prompted them to develop their own spreadsheet-based New Model system. They were equally prepared to invest in the FourthGen-based system despite the ill-conceived support arrangements and idiosyncratic technology, preferring to trust in fate rather than consider the benefits of a SystemsHouse-supported product.

The Warranty Director confessed in his post-project interview that he had been involved in an unsuccessful IS development in the past and was sceptical of the savings claimed by the Sponsor, Jack and the Consultant for the Research Centre/ToolVendor project. This was one reason for his limited support for the project.

Ronnie was a keen Windows user and a vociferous critic of the DOS-based QuickStore Recycling IS (Recycling project). This was not a "20th Century" system

(Ronnie interview). Ronnie's constant criticism helped to maintain the technology-focus of the Recycling project debate and reduce the energy available to address the more deep-rooted organisational problems facing the team.

Users in all three CarMaker projects exhibited powerful "spreadsheet loyalty" due to their past and ongoing use of spreadsheet-based information systems. The FourthGen and QuickStore-based information systems developed by the author were less usable than the spreadsheet systems and the users were well aware of the loss of usability, as indicated by Andrew and Stan's South Factory interviews (New Model project), Jack's concern to avoid FourthGen (Warranty project) and the Recyclate staff's clamour for spreadsheet-features such as "table view" data entry.

### 5.2.3 Category 3: Technology

Technology is used in a narrow sense here to mean the software tools and information system software used and delivered during the projects. Wider issues of the technology development process are addressed by many of the other categories, particularly the Methodology category.

<u>*Lesson 4*</u>

*Upper CASE tools (e.g. Analyser) are used by the IT specialist away from the user workplace, generally as a background documentation aid.*

For example, the logical models (DFDs and ERMs) of both the New Model and Warranty systems were developed through meetings with users in offices away from the factory floor where the systems were to be implemented. The models were initially developed using pen and paper or whiteboards and subsequently entered into Analyser by the analysts on returning to the Research Centre. The models were then printed out for use in subsequent discussions with the users.

Exceptions to this rule occurred when the Analyser prototyper was used to develop and demonstrate a prototype system to the New Model staff at the Research Centre; and when Analyser was used as a prompting and data capture device in interviews with Malcolm at the Research Centre during the Warranty project.

Analyser was used for a different purpose in the Recycling project: to document an evolving system in order to "sell" the progress made so far. Again, the tool was used by the analysts at the Research Centre, away from the users, with the DFDs printed out and tabled at user meetings.

*Poor integration with downstream tools (e.g. 4GLs and spreadsheets) leads eventually to logical model and repository obsolescence since changes to the IS have to be* *reflected in the downstream "code" but not in the upstream "model". This problem is exacerbated if the development team is under-resourced since alterations to models (i.e. documentation) provide no immediate benefits.*

The two projects where more than one tool was used (New Model and Recycling projects) both showed clearly the great difficulty in maintaining two or more separate sets of "descriptions" of a system. Two tools were used in the New Model project: Analyser and FourthGen. The Analyser/FourthGen interface was "one-way, one-time" in that changes made to the Analyser model could be exported to FourthGen but wiped out any further work undertaken in FourthGen. Conversely, changes made to the FourthGen instantiation could not be "imported" back into Analyser. Once the FourthGen system was implemented at South Factory the author had little time to maintain the Analyser repository in line with the changes he was making to FourthGen. The repository soon became out of date as new functions were added in FourthGen and existing functions altered.

A similar problem was encountered in the Recycling project. After the analysis paralysis of the Warranty project, ERMs were preferred to DFDs as a more useful modelling technique for database development. A paper-based ERM was maintained throughout the project and provided valuable documentation. Since QuickStore and Analyser were not integrated, there was no incentive to record the ERM in Analyser. Analyser was therefore relegated to a minor documentation role later in the project when the author attempted to "sell" the progress made by the database team to the project management. In addition to the Analyser/QuickStore integration issue, the project used a third tool, the NumberCrunch spreadsheet. NumberCrunch was used for two purposes; firstly by the Recyclate Ltd. stripping team to collect strip data early in the project and secondly by the Research Centre team to generate stripping curves later in the project. Data was therefore both imported from NumberCrunch into QuickStore and exported from QuickStore into NumberCrunch. Again, the tools were poorly-integrated, which meant that duplicate copies of the same data set resided in both tools. The problem was compounded by the fact that the project PCs were not networked, therefore data was duplicated both between tools and between PCs.

As with the New Model project, alterations tended to be made to the QuickStore IS since this was the core system; there was insufficient development resource to ensure that the Analyser models and NumberCrunch datasets were updated in line with QuickStore. Furthermore, the advent of one database per model meant that changes to later QuickStore databases were not reflected back to earlier QuickStore databases. Therefore there was inconsistency between tools, between PCs and <u>within</u> QuickStore!

*Using CASE tools does not automatically lead to high levels of user participation.*

One of the Consultant's original aims was to demonstrate that the use of CASE tools reduces the amount of time spent on coding and therefore increases the amount of time spent on analysis and other user-centred activities. The productivity figures for the New Model and Recycling projects show that coding productivity was high; but the corollary, that user contact would increase is not borne out by the effort tables for the three projects (Appendices 2.1, 3.1 and 4.1). Instead, high productivity tools used in an evolutionary manner tend to enable the developer to take on more coding work. The focus of effort remains technical, i.e. programming, and user involvement remains sporadic.

Similarly, as described in the first part of this lesson, the upper CASE tool tended to be used by the analyst as a documentation aid away from the user site, thus not contributing to increased user involvement.

*Lesson 5*

*Tools suffer from usability problems; both from the developer viewpoint and from the end user viewpoint. Documentation is often of poor quality.*

Analyser was the least difficult tool to use from a developer's perspective. This may have been due to the fact that the tool was largely used as a diagramming package with a repository underneath. In comparison, the 4GLs were used to implement some complex algorithms. Despite this, Analyser had its limitations. There was no word processing facility, so project reports had to be created and stored using separate word processor packages (particularly evident in the New Model project). There was no "cut and paste" facility so common data elements could not be copied between data flows and data stores. This became particulary laborious when documenting large numbers of similar documents with Malcolm for the Warranty project. The Analyser image painter was an eccentric device that relied on poorly-documented, obscure sequences of function keys to control image painting (New Model project). Finally, there was no printer spooler or bulk diagram printing facility in Analyser, therefore the user could never venture far away from the PC when printing off documentation for meetings.

FourthGen must have been written by technicians for technicians. The documentation was poorly written, full of obscure technical detail and offered few examples. The software was extremely complicated, and was driven by an underlying "processing cycle" that meant that programming consisted of filling in the options and slotting in small pieces of code at the appropriate point in the cycle. Since the cycle was poorly documented and obfuscated by the old-fashioned user interface, the programmer had to resort to programming by trial-and-error. The logic editor was a primitive line-editor with no cut-and-paste facility to enable common chunks of code to be copied between logic routines. Screens and reports were created using a line editor; individual pieces of text or fields had to be located on the screen by specifying their x,y co-ordinates. Fields were referred to by their absolute field number which changed when new fields were inserted, old fields deleted, or current fields moved. The field number was "hard-coded" into the timing cycle which meant that changes to field order could result in the wrong data being used in the wrong place at the wrong time. The timing cycle also applied to reports; there was no user-friendly query language. Furthermore, report parameters had to be hard-coded into the record selection logic; naturally wildcard matching was most difficult!

After working with FourthGen, QuickStore came as a relief. The product supported a query language, screen painter, and had no mysterious timing cycle lurking beneath the surface. The QuickStore usability limitations were less serious; each occurrence of a choice list had to be edited if the list changed; the central repository was simply a copy list of fields which was often overlooked when changes were made to the system and therefore soon became obsolete (cf. Lesson 4). There was no mechanism for sharing code between triggers; no auto-repeat flag for field values that should be carried over between successive records; and, finally, no graphing facility, which meant that NumberCrunch had to be used for graphing, with the resulting data maintenance problems described in Lesson 4.

Given the above comments, it came as a surprise to the author to find that QuickStore came in for more criticism from the IS users than FourthGen. The South Factory users' major criticism of FourthGen was in its inflexible reporting facilities, bugs and lack of support. Usability was only directly criticised by Stan (see interview who pointed out that the PSR entry screen layout differed from the paper PSR form which slowed down data entry. In contrast, QuickStore was severely criticised by the Recyclate Ltd. stripping team. This was partly due to the strong Windows and spreadsheet loyalty of the team (see Lesson 3), partly due to the rapidly changing nature of the recycling activity, and largely due to the very noisy conditions in which the data was collected with the operator sitting at the PC a few feet away from cars that were being torn apart. Users were particularly critical of the need to hit the correct function key to save each record (F2 to save, F8 to modify); the lack of auto-repeat fields which meant that data common to consecutive records had to be re-entered for each record and the complex sequence of keystrokes that had to be entered to search the on-line part name glossary. Furthermore, the data structure underpinning the Recycling IS, the Bill of Materials (BoM), was far more complex than the problem-based data structure of the New Model system. In a sense, the recycling IS was too sophisticated for the users in that the BoM could be traversed both up and down by moving the cursor to the appropriate sub-form and then hitting F10. In this way, the complete BoM hierarchy for any assembly could be traversed. IIowever, it was quite easy to forget whereabouts in the hierarchy you were. Ironically, the traversal facility eventually became a much-liked feature once the mouse-driven version of QuickStore arrived.

### Lesson 6

*The Lower CASE tools (FourthGen, QuickStore and NumberCrunch) provided inflexible reporting facilities that increased developer workload, reduced the opportunity for end-user computing and restricted user ownership of the systems.*

Data input problems were discussed in Lesson 5. This lesson focuses on data output problems. Again, this lesson applies only to the New Model and Recycling projects since an IS was not implemented as part of the Warranty project. Again, FourthGen was far less flexible than QuickStore, but QuickStore was used in a more difficult environment and on a more complex data structure.

Most of the changes requested by South Factory for the New Model system were for new reports or modifications to existing reports (see Appendix 2.2). Most of the requests were conceptually simple; maybe a change in report format or the addition of some new fields. These changes could have been made by the end users given an adequate query language and sufficient training. Unfortunately the FourthGen reporting facility was extremely complicated (see Lesson 5); the author found it difficult to use yet alone the end users. There was no query langauge, the timing cycle was unfathomable, and selection parameters had to be hard-coded into the logic routines. This meant that the author had to undertake all report development. Since the demand for reports increased rapidly once the system went live this soon became a cause for concern.

In contrast, QuickStore provided a useable query language and flexible parameter-driven reporting. Unfortunately, the Recycling IS data structure (the BoM) was far more complicated than the vehicle problem structure of the New Model system. Consequently, many of the reports requested by the users, such as Chris's materials reports and the dependency-based reports, required the development of tree traversal algorithms. It took the author over a year to develop these algorithms so, again, end-user reporting was out of reach. The end result was the same as the New Model project: end user frustration at delays in delivering reports and work overload for the developers.

Finally, NumberCrunch might have been expected to be the easiest tool of the three for end user reporting. Again, this expectation was confounded. The tricky interface between QuickStore and NumberCrunch and the need for further data manipulation in NumberCrunch meant that the production of the stripping curves was not a task for the unskilled end user. This was particularly disappointing since the users were generally pleased with the curves and may well have willingly used the tool if the database and spreadsheet had been fully integrated.

*Lesson 7*

*The Lower CASE tools, FourthGen and QuickStore, suffered from bugs that prevented the software from working as expected. The bugs were often difficult to work around and the suppliers' help desks were often unable to fix the problem. This resulted in a loss of user confidence in both the software and the developers, who were unable to provide the service expected of them. The inexperienced developers were often too proud, or too unsure whether the cause of the problem was due to their programming deficiencies or to a bug, to call for help and instead struggled on with increasingly complicated work-arounds.*

FourthGen was particularly bug-ridden. The first bug appeared on the first day of implementation at South Factory when a "system" error meant that PSR status could not be changed on function DU2. This function had been implemented specifically to enable the users to change the status on a batch of PSRs quickly by seeing up to twenty PSRs on the screen at once. The bug was never fixed, forcing the users to use the "single" PSR screen to change PSR status thus reducing productivity. Andrew's FAX (28/2/91) reported the problem of incorrect PSR totals being generated by the status reports. Again, this bug was never fixed, and caused the author to attempt several time-consuming and ultimately unsuccessful work-arounds. In the meantime the users continued to run meetings using suspect PSR data. The third bug emanated from the same file as the first bug. This time spurious launch team details were retrieved in the PSR maintenance screen. Again, the bug was never fixed.

The QuickStore bugs were less serious. Some Process records were lost during data entry early in the project, but this problem did not re-appear; the software would crash if the part/sub-part BoM hierarchy was too deep for the memory available on the PC; Robert experienced numerous data corruption problems in restoring databases mid-way through the project; the use of several "unique" key fields in the Part form crippled response time when saving records; the Glossary sort routine sometimes crashed, erasing the glossary; the DOS-product often crashed when run under Windows and the Windows product did not interface properly to the DOS version. Unlike the FourthGen problems, none of these problems were insurmountable. However, they did divert some of the limited development resource into bug-fixing and work-around activities and certainly undermined user confidence in the system.

### 5.2.4 Category 4: Cognitive Limitations

Cognitive limitations bedevil systems development. It is unreasonable to expect users to have a comprehensive and detailed grasp of their requirements at the start of a project, before they have seen or used the system. In any sizable organisation most users will only have a grasp of their immediate working environment. Information generated or used by other parts of the organisation may be known of vaguely, but its relevance to the users' work will not be clear. It is now appreciated that information requirements change over time, hence the development of the prototyping, evolutionary and spiral development models (Boehm, 1988, Gilb, 1988 and King and Galliers, 1994).

Like users, developers also suffer from cognitive limitations. Often developers are educated to focus on technology rather than user information needs. This encourages premature freezing of requirements so that the "real work" (the programming) can proceed at the earliest opportunity. The rate of technological change means that developers often have to master new technology which is unproven and, possibly, bug-ridden. Each new software product promises to be the solution to all preceding problems. This, combined with the inherent "unknowability" of software, means that a developer may take several years to fully understand the capabilities and limitations of the tools at his disposal. In the meantime, progress is made by trial and error, gradually experimenting with the software and often travelling up blind alleys. In summary, cognitive limitations can play a major role in limiting project progress despite the best intentions and hard work of all involved.

*Lesson 8*

*User information requirements are often not stated clearly, nor explained and are subject to change. Analysts can overlook requirements or assume that they have been agreed. The tendency to undertake analysis away from the workplace exacerbates the problem. Current working practices are often not challenged by projects that focus on IT.*

For example, the need to support multiple new model projects was not identified during the initial system analysis with the West Factory staff (New Model project). Numerous new and enhanced reports were requested by the South Factory users once the system became operational. These had not been identified by the earlier analysis; indeed the Sponsor and Stuart both remarked that the system "suffered from its own success" by stimulating latent demand for information (post-project interviews). This project focused on IT, particularly the Analyser/FourthGen relationship, and certainly did not challenge working practices at West Factory or South Factory.

Several examples of unclear requirements analysis arose during the Recycling project. The "blood letting" that took place at Recyclate Ltd. (11/12/92) after Geoff's earlier explosion was the first time that Robert had explained to the group his data collection procedure and the division of work (and data) between "framework" and "benchwork". This had significant implications for the design of the database. Furthermore, most of the group had not viewed or taken part in the stripping of a vehicle up to that point. The suggestion to view the next vehicle strip arose after the meeting. Another significant communication failure was exposed during the management reporting meetings in the spring of 1993. At the second meeting (26/4/93) the Environmental Strategist defined a list of research questions. After a year of the project, and significant disagreement stemming from opaque objectives, this was the first time that any objectives had been presented succinctly.

Like the New Model project, user information requirements evolved rapidly during the Recycling project. Unlike the New Model project, the underlying data model was also subject to significant evolution and revision due to the newness of the vehicle recycling process.

*Lesson 9*

*Outside events cannot always be predicted and can have a significant impact on the project.*

For example, the CarMaker re-organisation into strategic business units that denied David's resource to the New Model project; the far greater of volume of PSRs than expected and the decision to bring the Executive Car launch forward that added pressure to the system users at South Factory (Stan interview).

The Warranty project suffered from a lack of pressure until the corporate re-organisation of problem management prompted the Warranty Director to introduce the Problem Information Control Centre at East Factory. This happened shortly after the end of the Research Centre/ToolVendor involvement with East Factory (Malcolm interview).

New European Community legislation threatened to considerably reduce the need for detailed disassembly information by allowing "energy recovery" as an alternative to dismantling and reuse (Recycling project, 22/9/93).

### Lesson 10

*Alternative (substitute) solutions are often overlooked, ignored or hidden from the project team.*

This was a very common feature of the CarMaker projects. For example, the Sponsor did not regard the other problem tracking systems already in place at CarMaker as relevant at the start of the New Model project (the Sponsor interview). Ironically, it was one of these systems, the South Factory Warranty system, that finally replaced the FourthGen system! Similarly, Stuart was surprised that the West Factory staff had not identified the need to interface to Warranty earlier in the project (Stuart interview). The CarMaker Manufacturing Systems Manager expressed concern at the addition of a new 4GL, FourthGen, when CarMaker already had several supported 4GLs in place (2/10/90).

The Warranty project also faltered due to the presence of alternative, hidden, initiatives. Jack developed his own local system solution in AllTogether, whilst he and Simon were also working together on a company-wide re-organisation of problem management. Neither initiative was communicated to the author or the Consultant.

Likewise, many alternative information systems to the QuickStore-based system emerged during the course of the Recycling project. Chris devoted much of his energy to the AutoStrip project (graphical strip manuals); Geoff developed his own economic model using NumberCrunch; the "engineers" briefly flirted with their own recycling IS using NumberCrunch before admitting that the problem was more complicated than they realised (8/1/93); Terry's assistant developed his own "information flow map" without discussing his ideas with the author or Mary (3/11/92) and, finally, Terry used his own information sources when presenting the project's work to the Ministry (22/9/93).

### Lesson 11

*Demonstrating a prototype system increases user understanding and stimulates discussion and suggestions for improvement (albeit at a detailed technical level).*

Successful demonstrations of New Model system prototypes took place at the Research Centre on 15/5/90 (Analyser-based prototype), 3/7/90, 17/7/90 and 2/10 90 (FourthGen-based prototypes). Indeed, both Stuart and Andrew commented on the value of the Research Centre visits in their post-project interviews.

The evolutionary development approach adopted for the Recycling project meant that "prototyping" was ongoing. Certainly this generated a stream of user comments and change requests, but crucially the approach adopted failed to educate the users adequately as evidenced by Jim's UIS survey comments that he felt he had limited understanding of IS capabilities, and the general lack of appreciation from all the users of the sophisticated cost reporting facilities delivered later in the project.

Lesson 12

*Complex software products are inherently "unknowable" because the capabilities of the product cannot be seen nor touched; they are only revealed during execution. A developer needs time to work with a product in order to fully appreciate the product's capabilities and limitations. In the meantime, the developer can easily assume too much of a product and can attempt to implement software solutions that are beyond the product's capabilities.*

For example, the author's prototype PSR maintenance screen designed using the Analyser prototyper could not be implemented in FourthGen because it mixed single fields with scrolling windows. The prototype screen mirrored the layout of the paper PSR form in use at West Factory and South Factory; the modified FourthGen screen did not, which caused some delay in data input (see interview with Stan). During the support phase of the New Model system at South Factory, the author attempted to implement some sophisticated reports for Stuart that stretched FourthGen, and the author's knowledge of FourthGen, to the limits (18/3/91). The reports consumed much effort and were not completed successfully.

Like FourthGen, QuickStore also flattered to deceive. The data model for the Recycling project was based on the Bill of Material. Recursive programming techniques are most appropriate for processing this kind of tree data structure. Unfortunately, QuickStore did not support recursive programming and the author spent much time attempting to emulate recursion using the QuickStore query language. With hindsight, it may have been more sensible to have used a recursive programming language, or to have called in an experienced QuickStore programmer to code the routines. Instead, due to a combination of ignorance, pride and lack of resources, the author persevered and finally achieved the desired result only to find that the report's customers had turned to other sources for their data needs (17/11/93).

### 5.2.5 Category 5: Professionalism/ISD Methodology

Much has been written about IS development methodology (see Avison and Fitzgerald, 1988 for a comprehensive overview). Strictly speaking, methodology can be defined as "the study of method". In practice it is often used to describe the mixture of rules, advice, tools and techniques by which systematic problem-solving should take place (Sauer, 1993). A method may be viewed as an instantiation of a methodology; a specific, defined set of rules, advice, tools and techniques that should be followed as stated in order to achieve a defined result. Methods can be viewed as the distillation of many years of practitioner experience; as "canned" problem-solving that reduces the cognitive load on the developer (Sauer, 1993). In this respect, methods are valuable. Critics of methods point out that following the "book" can also lead to rigid, inappropriate problem-solving (Checkland, 1981, Avison and Wood-Harper, 1990 and Walsham, 1993). In this section, the IS development methodology used during the CarMaker projects is described and reviewed.

Lesson 13

*Problem-solving rarely follows a defined development method. Instead, ad hoc problem-solving takes place using familiar tools and techniques. Well-known techniques, such as data flow diagramming and entity-relationship modelling have value, but are not valuable in all situations. Furthermore, different actors develop and use their own ad hoc techniques.*

For example, the author attempted to follow a defined method (MiniMethod) at the start of the New Model project (20/3/90) but was overruled by the more experienced the Consultant who preferred an *ad hoc* problem solving approach. The project made extensive use of DFDs, ERMs and prototyping with all three techniques being of value.

In contrast, the Warranty project used DFDs as the primary modelling technique with minimal entity-relationship modelling and no prototyping. Two further techniques were introduced in this project: cost/benefit analysis and the JAD approach for rapid requirements capture. Unlike the New Model project, the extensive use of DFDs contributed to the analysis paralysis that overcame the project. The cost/benefit analysis identified £1 million of potential cost savings but was not believed by the Warranty Director. Finally, the JAD was not successful, partly due to the use of DFDs to model requirements which resulted in discussions becoming over-focused on detail.

An evolutionary development approach was adopted for the Recycling project, again not based on a defined method. The primary modelling technique was ER modelling which provided a useful foundation for evolutionary database development. However, ER modelling is a technically-oriented technique and has no facility to represent the scope, reporting facilities or personnel involved in the system. It also hides complexity to some extent: for example, the tremendous problems encountered when trying to traverse the parent-child BoM hierarchy were not highlighted by the single relationship line labelled "parent-of" in the ERM! The inadequacy of the technique for "selling" project achievements led the author to turn to DFDs later in the project for management presentations (9/3/93).

In parallel with the techniques used by the author and Mary, other actors adopted their own modelling techniques during the Recycling project. For example, Geoff preferred IDEF0 to DFDs for process modelling (9/3/93), Terry's assistant developed his own modelling language for his "information flow map" and Robert described the stripping process by way of a flowchart.

Finally, a comment on the "correctness" of ERMs. ER models are generally viewed as being less ambiguous and less open to interpretation than DFDs. This may well be so, but it does not make them foolproof. The ERM for the New Model system was agreed by the users before implementation. However, the first demonstration at West Factory (19/12/90) revealed a serious flaw in that the PSR number was not sufficient to uniquely identify a vehicle problem. The Recycling project was dogged with disagreement over what was the "correct" data model. Geoff wanted a "part" focus, Recyclate Ltd. wanted a "process" focus; finally, after several months of argument the author identified the need for two relationships ("disgorges" and "disassembled by") between part and operation (process) to fully describe the relationship. The relationship between part and material was also problematical and not fully resolved until late in the project.

*Lesson 14*

*Pressure on developers increases greatly once the system becomes operational. This is because analysis doesn't directly change people's work, implementation does. Evolutionary development can catapult developers into the maintenance stage without sufficient analysis. This can lead to "scapegoating" of the developers since the IS is tangible and open to criticism, whereas ill-conceived requirements remain intangible and invisible. Supporting and maintaining an evolving IS can be highly stressful and may require considerable resources as the direction of new requirements is often difficult to foresee.*

This lesson is derived from the painful experiences of supporting and maintaining evolving systems with woefully inadequate resources during the New Model and Recycling projects. In contrast, the Warranty project, which failed to deliver a working system, remained calm and pleasant throughout.

The New Model project was unstressful until implementation, after which point the author found himself struggling to cope with FourthGen bugs, demands for new reports and lack of support from CarMaker and ToolVendor. It was not surprising therefore that the FourthGen system was eventually discarded in favour of the better-supported SystemsHouse-based Warranty system (the Sponsor and Stuart interviews). Even then support was not as good as expected (Andrew interview).

The Recycling project started in a relaxed manner, with the author commenting on the enjoyable summer trips out to Recyclate Ltd. in the countryside (4/6/92). A first-cut QuickStore IS was put in place but was not used by Recyclate Ltd. for some months whilst Robert refined his data collection procedure. When the database was finally used, both the data collectors at Recyclate Ltd. and the ultimate data customers (Geoff and Chris) found the system did not support the way they wanted to work (3/12/92). The next six months saw the author and Mary buffeted by scapegoating, changing requirements, software limitations, lack of resources, lack of experience and algorithmic complexity until some calm was restored by the summer of 1993. By this time the database had settled down and had become a usable data collection tool. Unfortunately, the project management and Geoff had begun to turn to other sources for their data and the database team eventually lost support.

*Lesson 15*

*Developer training, user training and system testing were largely ignored in all three projects. User expectations often exceeded the developers' capability to deliver.*

The developers attended the ToolVendor Analyser and FourthGen training courses at the start of the New Model project. These courses gave the author and the Consultant basic training in tool skills, but did not enhance the author's systems analysis and design skills. User training during the New Model project was limited to occasional demonstrations of prototype systems at the Research Centre and a brief demonstration of the system at the South Factory and West Factory hand-overs. A user manual written by the author was given to both sets of users but was not referred to by the author again. System testing for the New Model system entailed the author entering various "boundary" data and a sample of PSRs collected from West Factory into the PC-based FourthGen system at the Research Centre. Each report was run against the test data. A subset of this data was entered into the VAX version of the system during installation on CarMaker's network in December 1990. The testing amounted to about two or three days work in total and failed to uncover any of the bugs that were to undermine the system subsequently. As for user expectations: the Sponsor expected an integrated problem tracking system to replace the disparate systems that had evolved over the years; Stuart expected a flexible reporting system and Andrew expected a package that required little maintenance or support.

Jack and Malcolm were sent on ToolVendor systems analysis techniques training courses mid-way through the Warranty project. Training for the rest of the participants consisted of a brief description of the data flow diagramming symbols prior to meetings in which the technique was presented to an audience for the first time. Malcolm in particular commented that he felt he could have contributed more to the project if he had attended the ToolVendor course earlier (Malcolm interview); however, the Sponsor and Jack felt the East Factory staff learnt much from exposure to the DFD technique. There was disagreement over what to expect from the project (the Sponsor and Jack interviews); however, it was generally agreed that an integrated problem management system that could pool data from multiple sources was desired.

The Recycling project was characterised by a total absence of training for both users and developers. The author and Mary learnt how to use QuickStore through experience in use, not from training courses. Similarly, NumberCrunch and the QuickStore/NumberCrunch interface had to be learnt through use. Certainly the lack of in-depth understanding of the tools meant that the author and Mary attempted to develop solutions that were highly demanding of the software. A better understanding might have resulted in less ambitious designs and greater confidence in identifying when user requirements could not be implemented. There was no formal user training during the Recycling project, although both the author and Mary spent time informally describing the system capabilities to the users throughout the project. User expectations of the system remained greater than the developers' capability to deliver for most of the project. In particular, the Recyclate Ltd. staff continued to want Windows/spreadsheet levels of usability from a DOS/database product. The data customers expected sophisticated cost-reporting and "cherry-picking" reports that simply could not be implemented with confidence using a relational database query langauge. Furthermore, they expected graphical output from a non-graphical package.

### 5.2.6 Category 6: Culture/Commitment

The stakeholders in systems projects tend to be divided into two camps: developers and users. In practice there is a spectrum of involvement from specialist developer through end-user developer, direct user, indirect user and peripheral involvement. Furthermore, these people may belong to a variety of different organisations, both within one company and across multi-company teams. The location of an individual within this structure can have a significant bearing on their involvement in a project. For instance, full-time specialist developers are likely to devote the majority of their energy and effort to the project whereas a part-time indirect user may feel far less committed. The impact of organisational structure, culture and commitment on the CarMaker projects is discussed in this section.

*Lesson 16*

*It is often difficult to persuade users to give sufficient time to a systems project. Users are often "busy" with their operational work and do not appreciate the commitment demanded of a systems project. Furthermore, the manufacturing culture may well encourage "reactive" working styles, whereas the systems culture tends to expect a more studied, analytical approach.*

For example, David's training requirements and availability were barely considered in the New Model project. Similarly, South Factory staff were not made available for system support due to the pressure to launch the Executive Car ahead of schedule.

Many of the group meetings during the Warranty project were squeezed into the lunch breaks at East Factory. Furthermore, the Sponsor, Jack and Malcolm each identified the greater importance of day-to-day operational work as a barrier to project progress during the post-project interviews.

Finally, the Recycling project suffered from the general management view that the database work was trivial. This partly stemmed from the superficial data analysis undertaken by the Graduate Engineer at the start of the project (April 1992) which ignored complicating factors such as dependencies between parts. As the project progressed the author called on numerous occasions for greater input from the data "customers" (Geoff, Chris, Terry, the Accountant and the Environmental Strategist) - see entries dated 12/8/92 and 30/9/92. After Geoff's "explosion" (3/12/92) user involvement became easier to obtain and a whole day database JAD session took place on 11/1/93. Despite this, the management team of Terry, the Accountant and the Environmental Strategist remained detached from the database efforts, relying on the monthly progress meetings to keep themselves informed (see participation analysis in Appendix 4.1).

### 5.2.7 Category 7: IS Assessment

The research question that originally motivated this project asked "Do CASE tools enable quality systems to be developed quickly?" IS assessment is therefore a key aspect of this work. Many researchers and practitioners are striving to make IS assessment quantitative and objective in the hope of differentiating once and for all between "good" systems and "bad" systems (Gilb, 1988, Fenton, 1991, and AMI, 1992). They belong to the school that believe that IS requirements can be defined clearly and unambiguously and that achievement of those requirements can be measured and verified. Such objectivity is certainly attractive. However, as Lyytinen and Hirshheim (1987) point out, IS requirements are rarely clear nor measurable. Those that are form a subset of the true requirements and different stakeholders may well hold different and possibly contradictory views of what is wanted from the IS. Stakeholders evaluate IS deliverables (Sauer, 1993). The evaluation may be objective or subjective. Either way, a "successful" system is one that retaıns stakeholder support; not necessarily one that meets stated requirements. In this section lessons from the assessment of the three CarMaker projects are presented.

*Lesson 17*

*Both productivity and quality metrics can be misleading.*

The productivity and quality metrics for the New Model project show a high level of productivity and a low level of defects. Despite this, the system was abandoned after only six months of operation and the stakeholder interviews showed that all of the users were frustrated by the inflexibility of the system and disappointed about the lack of on-site support. The metrics told of "success", the stakeholders were less enthusiastic.

Similarly, the defect count for the Recycling project was low and two of the stakeholders were very satisfied with the system (Geoff and Robert). However, the other four stakeholders were very dissatisfied. Again metrics and stakeholder views contradict, showing that metrics may be a narrow and potentially misleading measure of success.

*Lesson 18*

*The analyst's interpretation of project outcomes may differ substantially from other actors.*

This lesson re-enforces Lesson 17. The author felt dejected at the end of the New Model project. The early optimism had been crushed by the realities of coping single-handedly with system maintenance and support at South Factory. The author certainly felt that the FourthGen New Model system had "failed". Whilst the post-project interviews with the South Factory staff identified a number of failings, the interviewees also identified benefits arising from the project that the author had not appreciated. For example, the Sponsor described the system as "suffering from its own success" in that the greater range of reports available had stimulated users to use the system more and to request further reports. Stuart and Andrew re-iterated this view. Furthermore, Stuart and Andrew both found the visits to the Research Centre to view the prototypes useful opportunities to meet with their West Factory counterparts. Andrew liked the improved response time of the system and the data validation facilities; Stan also liked the data validation facilities and the "great" range of reports. Finally, Stan pointed out that the FourthGen system had served as an excellent prototype for the ApplicationMaster system that replaced it!

The Warranty project also ended in disappointment for the author, particulary due to the behind-the-scenes manoeuvring of Jack and Simon as they worked on alternative initiatives to the Research Centre/ToolVendor work. Like the New Model project, the post-project interviews revealed a number of benefits as well as criticisms. The Sponsor felt that the project delivered significant benefits in terms of providing a clear picture of Warranty operations that was invaluable in communicating East Factory's requirements in the corporate re-organisation. He felt Jack and Malcolm had benefitted greatly from close involvement with the author and the Consultant, and that their analysis skills were now considerably enhanced. Jack stated that the techniques were "tremendously useful" in clarifying Warranty requirements and providing a firm foundation for both his local system initiative and the corporate re-organisation. Malcolm felt that the DFDs had been a major aid to communications and that user involvement had been very good. Finally, the Warranty Director had not been closely involved in the project, but had noticed a marked improvement in Jack and Malcolm's ability to present information after the project.

As for the highly problematical Recycling development, even here some positive comments were forthcoming. Geoff felt that the IS was reliable and that the information produced was relevant, precise and complete. Overall, he was very satisfied with the IS. Robert was very satisfied with his participation and understanding of the database. He felt that the system was very flexible and that the information delivered was very accurate, precise, complete and readily available with very good response time. It is interesting to note that both Geoff and Robert were closely involved in system development and both had a good understanding of QuickStore. The less involved and less aware users were generally highly critical of the system.

### 5.2.8 Category 8: Noise

"Noise" is an emergent property of IS development. It arises from a combination of factors already described including physical environment, political battles, cognitive limitations, system complexity, project structure, reactive culture, outside events, lack of resources, lack of experience and more. The noisier the project the greater the likelihood of mistakes being made. The noise levels of the three CarMaker projects are described in this section.

Lesson 19

*Good ideas and good initiatives can be drowned by the noise of the project.*

The Consultant exploited the noise of the New Model project to push through the purchase of FourthGen despite the concerns of the CarMaker Manufacturing Systems Manager regarding additional 4GLs and the Executive Car Chief Engineer regarding the old-fashioned presentation of information. Similarly, the author's concern regarding lack of method was overruled by the Consultant in the haste to start work on the Analyser/FourthGen interface. The author's memo defining the responsibilities of the Research Centre and of the end-user support programmer (David) drew no response from the Sponsor or Chief Engineer (2/10/90).

Turning to the Warranty project, the Warranty Director's lack of enthusiasm and support was ignored by the Consultant, the Sponsor and Jack in their keenness to pursue a systems solution to the problem management problem. This lack of "official approval" was to constrain the resources available to the project subsequently.

However, both of these projects pale in comparison to the noise generated by the Recycling project. Not only was this project physically noisy due to the core activity of stripping cars using power tools and hammers, but it was also cognitively noisy in that the data complexity was very high which required tremendous concentration and experience on behalf of the database user in order to input data accurately. The noise was compounded by the continual political in-fighting between sub-project "barons" and between the "workers" and the "management". Against this background a number of sensible initiatives were either lost or diluted including Terry's initial list of "target" parts to reduce the quantity of data collected (30/7/92); the Environmental Strategist's initial request for strip curves (12/8/92) - finally addressed in the spring of 1993; the author's spiral project plan for the database sub-project that aimed to visit each sub-project "baron" in turn thus avoiding exclusive concentration on any one area (25/11/92); Chris's request for materials reports (14/12/92) - finally addressed in the summer of 1993; and the database PRF system for recording problems and change requests and describing the database workload to the other project members (18/2/93).

### 5.2.9 Category 9: Emotion

Most studies of systems development assume that the actors act in a calculating manner in order to achieve their objectives. Many studies assume that these objectives are made public and agreed by all participants. Some studies acknowledge that different actors may have different, possibly conflicting objectives (see for example Lyytinen and Hirshheim, 1987). However, few studies delve deeper into the emotions of the system development process. Action research is an appropriate approach for revealing the emotional side of IS development since emotions are best recorded immediately and can dissipate rapidly once the moment has passed. It is certainly difficult to see how an outsider (such as a researcher using the case study or survey approach) can experience or understand the emotional side of a project which only comes from being closely involved and committed to the work. Emotion has been touched upon in a number of the preceding lessons, here a summary of the emotional aspects of the CarMaker projects is presented.

Lesson 20

*Developers are often highly-focused and committed professionals who nurture their IS like a child. Criticisms of the IS are taken to heart and can erode developer confidence and contribution if sustained over a long period of time.*

For example, the author still felt proud of the New Model system when it went live at South Factory despite his serious concerns about the efficacy of FourthGen as an implementation language (7/12/90). The arrival of Andrew's FAX describing various reporting problems (28/2/91) surprised the author. Andrew's trusting comments made him feel very guilty for imposing a flawed system on the busy South Factory users. Despite the bugs, the author was surprised and felt betrayed when David told him of South Factory's plans to abandon the FourthGen system (13/3/91). The ongoing bugs and reporting inflexibility troubled the author during March, April and May and the final decision to abandon the system came as a relief. The emotional commitment that the author had developed for the system meant that he felt dejected at the end of the project.

In comparison, the Warranty project was relatively unemotional. The author was disappointed that a working system had not been implemented, but conversely had been saved the stress of system support in an under-resourced environment. The major irritation was the political manoeuvring of Simon, the CarMaker IT Strategist, who was leading the corporate problem management re-organisation. It was no surprise that the one formal meeting between the author, the Consultant and Simon should end unsatisfactorily (27/9/90).

The problems the author faced during the New Model and Warranty project paled in comparison with those arising from the Recycling project. This project was the "noisiest" of the three (Lesson 19). It also left the author and Mary exposed to sustained criticism for several months during which time they battled to come to grips with the limitations of QuickStore, the true complexity of the data, and the maintenance load generated by wilful users who insisted on changing forms, reports and lookup tables without notifying the developers. Highlights of this unpleasant period include Geoff's "explosion" (3/12/92) when he declared that the database and the data collected by Recyclate Ltd. were "useless" and proceeded to redesign the data model whilst the author looked on in horror. "Black Friday" (11/12/92) when the author and Mary presented the modified database to the Recyclate Ltd. staff and had it rejected as totally inappropriate. The "scapegoating" monthly progress meeting of 16/12/92 when Terry had the Database sub-project report moved from its usual location to the end of the agenda to allow time for an inquiry. The feeling of being made redundant by IT-literate end users busy developing their own system (8/1/93); the disappointment when teamworking collapsed due to Recyclate's "UDI" over scientific resequencing and the subsequent citing of the database as the major cause of project delays by Robert at the monthly meeting (10/2/93); the Environmental Strategist's explosion at the next monthly meeting because he still hadn't seen any useful information from the database; Robert's five database failings presented at the start of the April progress meeting, to which the author again responded angrily due largely to a feeling of betrayal after the amicable discussions with Robert the day before. By the July progress meeting much of the emotional strain of the project had subsided, largely due to improvements in the database. The author's focus was now on achieving the "Holy Grail" of delivering an automatic picking list that could define the optimal sequence by which to disassemble a car given the current price of materials. The report was finally completed shortly before the author left the Research Centre (17/11/93), the final emotional point of note was that by this time few others really cared.

### 5.2.10 Category 10: Research Approach

The previous categories addressed the *content* of the CarMaker studies; and described the *process of IS development* as experienced by the author in a real organisational context. The following, final lesson reflects on the *process of carrying out the research*.

## Lesson 21

*The research did not follow a standard approach; instead, a hybrid approach emerged as the research progressed. The hybrid approach exploited a window of opportunity to develop a rich description of working with CASE tools, however the approach had a number of limitations. Firstly, combining the two roles of researcher and practitioner places great demands on the researcher. The practical work often takes precedence over the research because it is more immediate and affects the interests of many stakeholders, whereas the research is of primary interest to the researcher alone. Secondly, the use of a defined research framework may well help the researcher to explain his work to the other stakeholders and to appreciate better the activities that should be undertaken to produce substantive results.*

The research was motivated by the Consultant's concern to counter the backlash forming against CASE "hype" by providing objective evidence of benefits. The Consultant's initial proposal to the Research Centre represented a timely window of opportunity, enabling academics and practitioners to work together to address these issues. In this respect the work benefitted greatly from being motivated by a genuine research problem that was of concern to both practitioner and academic communities.

There was no theory to guide data collection. Instead, we were guided by a shared belief that data on effort, stakeholder participation, activity descriptions and problems/thoughts would form a rich resource for subsequent analysis. In this sense, the research method was *grounded theory*, the intention being to experience tool usage and systems development first and, through that experience, to allow important issues to emerge. Secondly, the research method was *action research*, in that the author and the Consultant were acting as both researchers and practitioners, using the tools themselves in order to study tool usage in a real organisational setting. Thirdly, the work represented a *phenomenological* study in that the researchers kept a diary into which were written thoughts, problems and issues as the research progressed. The diary provided the major source of material for subsequent analysis and was supplemented by other project documentation, including minutes of meetings, progress reports, defect reports, change requests, data flow diagrams, data models and the output from the computer systems developed as a result of the work. The analysis of the documents has been undertaken by the author as a *hermeneutic* study in this chapter; major events and their consequences were identified and classified (referred to as "bracketed" by Boland and Day, 1989) then revisited later and refined until the essence of the author's experience of being an analyst and using the tools was uncovered. The essence of experience has been presented as a set of lessons in this chapter.

The preceding analysis shows that elements of four standard research approaches were used during the study. However, neither the author nor the consultant were aware of these approaches at the start; we seized an opportunity to address a pressing practical issue and chose to answer our research question by means of extensive qualitative and quantitative data collection; this was our overall research "method". To claim that this work was pure *grounded theory* would be untrue. The work stops at the "accurate description" stage (Strauss and Corbin, 1990, p. 22) and does not attempt to identify relationships between the key constructs[9]. Theory building has been left for further work (see section 8.3.1) and this study has provided a rich source of data for that work. Similarly, the work embodies elements of *action research*, in that the researchers were also practitioners and were undertaking work that was of value to all stakeholder groups, but the work was not guided by a theory or conceptual framework. Thirdly, a *phenomenological* dimension has been introduced by the author's emphasis on the personal, emotional experiences of systems development in an under-resourced environment. But the work is not pure phenomenology as practised by Boland and Day (1989): for instance there was no third party interpreting the author's words or soliciting further insights. Table 5.1 summarises the similarities and differences between the CarMaker research and the research approaches discussed in this section.

The major limitation of not following a standard research approach was that there was no frame of reference to guide our research. It was therefore difficult to decide what data to collect and when to stop collecting. In practice, we collected as much data as we could and stopped when the author left the Research Centre. This meant that a lot of time was spent recording thoughts in diaries, noting effort and stakeholder involvement, recording IS defects and changes, counting function points etc. Whilst this represented a large, ongoing overhead, the major problem came at the end of the field work when the author was confronted by voluminous data to analyse and

[9] Strauss and Corbin (1990) identify three levels of interpretation of qualitative research. Level 1, *No Analysis*, presents the data without interpretation and leaves interpretation to the reader. Level 2, *Accurate Description*, weaves descriptions of phenomena, quotes from stakeholders, and the author's interpretations into a "believable" story. Level 3, *Theory Building*, defines constructs and relationships within the data in order to develop a theory that can be used to explain reality and provide a framework for practitioner action.

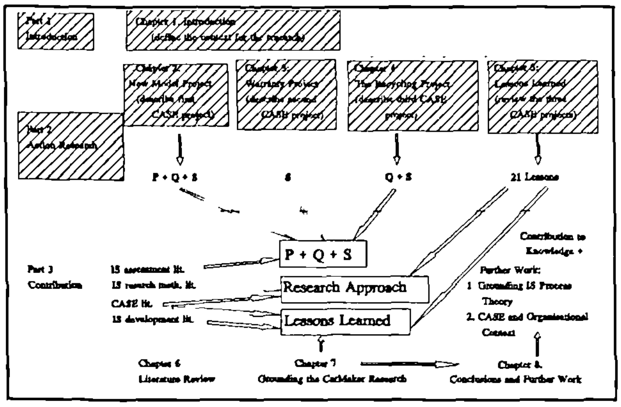
interpret! Perhaps attention to theory would have reduced the volume somewhat (although qualitative research seems to be synonymous with large quantities of data, Eisenhardt, 1989). A second problem was the tendency for the research to be overshadowed by the *practice* of IS development. This relates to lessons 4, 14 and 16, in particular, which show that IS development in an under-resourced environment tends to lead to stressful reactive working. In these circumstances it is tempting to devote all of the project effort to keeping the IS working and to forget about the research objectives. Whilst the author managed to persevere with data collection throughout the three projects, a clearly-defined research approach would have provided a valuable source of re-assurance and may well have enabled the author to raise the profile of the research objectives in the eyes of the other stakeholders, thereby gaining more time to devote to considerations of research methodology and theory development.

| Research Approach | Similarities | Differences |
|---|---|---|
| Grounded theory | Theory not used to guide study, instead data collected and subsequently interpreted to enable theory to emerge. | Research has not progressed to theory building stage yet, this has been left for further work (section 8.3.1); instead work has reached "accurate description" stage (Strauss and Corbin, 1990, p. 22) with the intention of using existing theory e.g. innovation theory, Sauer (1993), to help draw out theory from the CarMaker data. |
| Action research | The researchers were also practitioners "making things happen" in order to answer a research question. We generated our own empirical data and exploited a window of opportunity in which the interests of an academic, a tool vendor and a tool customer coincided. | The work was not guided by theory; our guide was the original research question: "Do CASE tools enable quality systems to be developed quickly?" |
| Phenomenology/ Hermeneutics | The essence of being an analyst using CASE tools in an organisational setting has been described in the author's own words (without the constraint of theory). During the writing-up of the work, the author has repeatedly revisited the original data in order to identify categories and lessons learned ("bracketing", Boland and Day, 1989); thereby developing a hermeneutic interpretation of the original "text". | The process was carried out by the author alone, without third party involvement (cf. Boland and Day, 1989). Therefore there was no opportunity to benefit from another researcher "drawing out" my experience and clarifying my thoughts. |

**Table 5.1** Comparison of CarMaker Research Approach

with Standard Approaches

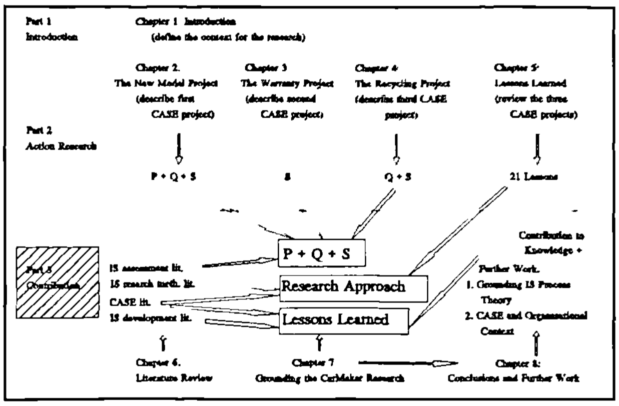
# *Summary of Part 2*



Part 2 is now complete. The three stories have been told and the productivity and quality evidence presented. The twenty-one lessons compliment the raw metrics and stakeholder views by allowing the author to present his interpretation of the stories. A considerable body of evidence has been presented in Part 2. In Part 3 this evidence is grounded in the IS literature in order to show where contributions to knowledge have been made.

# Part 3

# Contribution

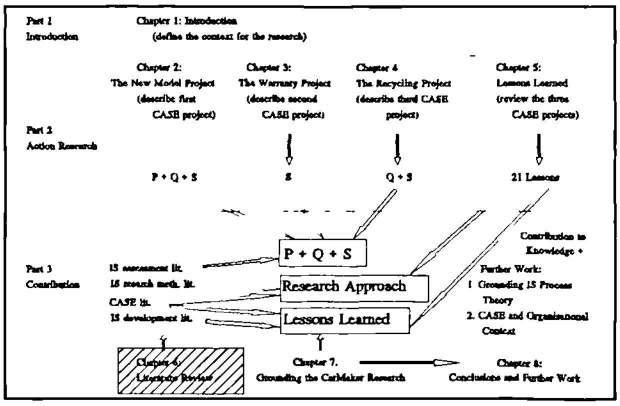


## Introduction to Part 3

In Part 3 the contribution to knowledge is evaluated by first reviewing the CASE, IS development, IS assessment and IS research methodology literatures (Chapter 6) and then by taking each of the three principal elements of the CarMaker studies: the research approach, the productivity, quality and stakeholder views (P + Q + S), and the lessons learned and grounding them by comparison with the literature (Chapter 7). The process of grounding enables results that are in common with the literature to be identified, thus adding to the *cumulative body* of IS knowledge; and enables new results to be identified, thus *advancing knowledge* into new areas. The contribution is summarised in Chapter 8, and two proposals for further work are presented which aim to advance both IS development theory and CASE technology.

# Chapter 6

# Literature Review



## 6.1 Introduction

The research presented so far is not well-grounded in theory. Neither the author, nor the chief instigator of the research programme, the Consultant, were familiar with the relevant academic literature at the start of the programme. Instead, the programme represented a "window of opportunity" whereby an academic, a tool vendor and a tool customer could work together to understand CASE potentialities better. In this chapter two issues are addressed: firstly, contribution to knowledge. Whilst it is possible to describe the results and lessons learned from a piece of research, as presented in Chapter 5, it is far more difficult to define precisely what is known in a given field. The twenty-one lessons presented in Chapter 5 clearly represent a substantial contribution to the author's knowledge of CASE, IS development and IS assessment; but what do other people know? Are these lessons "common knowledge" already? This is an open question. It would be impossible to define precisely everything that is known by everyone in the field of CASE and related areas. The problem is simplified by imposing a boundary on the domain of discourse to include only published literature, and within the published literature, to define what subjects are "inside the boundary" and what subjects are "outside the boundary". Figure 6.1 defines the boundary of the domain of discourse for the research. Clearly the CASE literature lies at the heart of the domain, and is shown as the "primary domain". But the research also investigated issues relating to IS development in general and also IS assessment and software metrics. These form the "secondary domain". The research was conducted in the spirit of action research without appreciating that this was a "named" approach. The author and the Consultant simply believed that the time was right to adopt such an approach to find out more about CASE. The third domain is therefore "IS research methodology".

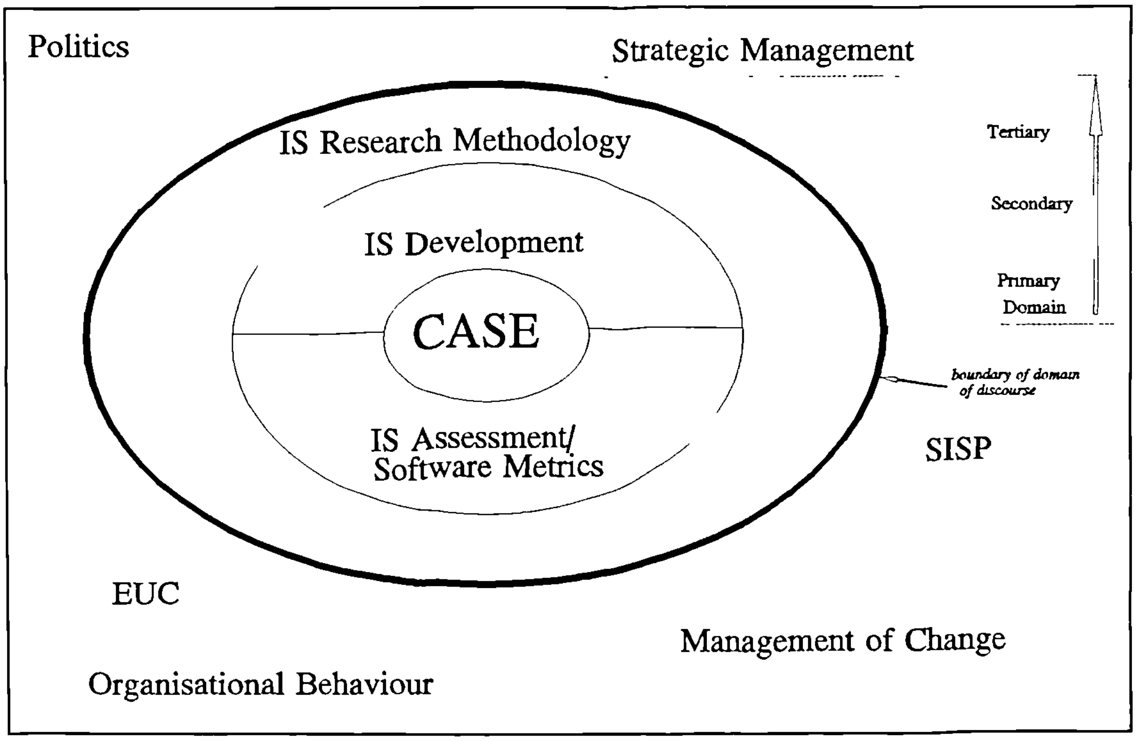


Figure 6.1 Map of Knowledge Domains

Two IS domains that are arguably pertinent to the CarMaker studies were ruled as being outside of the boundary: Strategic IS Planning (SISP) and End User Computing (EUC).

Admittedly both the New Model and Warranty projects suffered from not being anchored to CarMaker's IT Strategy; indeed, the Warranty project faltered partly due to being overtaken by a corporate re-organisation of problem management that was driven by the IT Strategy team. However, the author was not party to the IT Strategy planning process and was only aware of the initiatives when they impacted the CASE research studies. It would therefore be inappropriate to claim that SISP was a focal area of study.

Similarly, EUC was an element of all three studies. The FourthGen New Model system replaced a spreadsheet system developed by David at West Factory; Jack developed his own problem management system in AllTogether during the Warranty project; and various EUC developments took place during the Recycling project, most notably Recyclate's development of a NumberCrunch IS to capture data early in the project and Geoff's development of a NumberCrunch-based economic modelling system that possibly contributed more information to the final Ministry report than the QuickStore system developed by the specialists. Whilst these systems all contributed to the context of the projects, the author's focus was on the systems developed using the CASE tools. Like the SISP manoeuvrings, the author's experience of the parallel EUC initiatives is limited and largely anecdotal. Therefore EUC cannot claim to be a focal area of this study.

Given that "knowledge" is not homogeneous, one can argue that a "contribution" can be made to one or more of the distinct domains. Clearly a contribution to several domains is more substantial than a contribution of similar depth to only one domain. "Substantial" therefore has both a depth and a breadth element. In Chapter 7 it will be argued that this work makes a substantial contribution to the CASE domain. The work also contributes to knowledge in the areas of IS assessment and IS development, but not to the same extent. This is because there already exist a number of empirical studies of IS development that explore the social and political implications of computerisation (for example, Kling, 1987, Avison and Wood-Harper, 1990, Newman and Robey, 1992, Orlikowski, 1993, Sauer, 1993 and Walsham, 1993). The CASE domain has yet to achieve this level of holistic maturity and is still strongly focused on tool development, and, to a lesser extent tool implementation. The questions of what tools should be developed based on the requirements of all of the stakeholders in IS development, and what happens when inappropriate tools are used in practice are rarely addressed.

The Management of Change, Politics, Strategic Management and Organisational Behaviour literatures lie at the boundaries of this study (figure 6.1); the author's exposure to these subjects being primarily through the writings of the holistic IS researchers such as Walsham (1993). The contribution of this work to these domains is therefore not assessed.

### Literature Review Methodology

A subset of the literature in each of the focal domains will now be reviewed. The choice of literature to review within each domain is based on the author's perception of what constitutes relevant and/or rigorous work, and largely on the cognitive limitations of the author in the face of a rising tide of potentially significant literature. The following criteria were used to select work for review:

1. *Relevant domain as defined in figure 6.1.*

2. *Reputable author/journal.*

3. *Research questions/hypotheses.*

    Work focusing on IS quality and/or productivity is clearly highly relevant to this study.

4. *Research approach.*

    Empirical work is especially relevant to this study. The literature in the following section is classified under four headings: surveys, case studies/action research, laboratory experiments and normative writings. Normative writings is a term proposed by Wynekoop and Conger (1991) to cover a wide variety of research, especially literature reviews, retrospective practitioner accounts of experience and opinion/future-gazing work by academics and practitioners (see Galliers, 1991 and Wynekoop and Conger, 1991 for taxonomies of research approaches).

5. *Application area.*

Information system development or software engineering. The former being more relevant than the latter.

6. *Theory/model/framework proposed or tested.*

7. *Results/conclusions/lessons learned.*

## 6.2 The CASE Literature

### 6.2.1 CASE Survey Research

The 19 CASE surveys reviewed are listed below:

> Aaen *et al.* (1992); Banker and Kauffman (1991); Burkhard (1990); Dimbleby and Lo (1992); Doke (1989); Glasson *et al.* (1992); Hayley and Lyman (1990); Howard and Rai (1993); Jones (1992); Jones and Arnett (1993); Kievit and Martin (1989); Low and Jeffery (1991); Norman and Nunamaker (1989); Price Waterhouse (1989); Siltanen (1990); Smolander *et al.* (1990b); Statland (1989); Stobart *et al.* (1991); Wijers and van Dort (1990).

**Discussion**

Many organisations have less than three years experience of CASE and the tools may not be widely diffused. Survey respondents were nearly all IT specialists (17 out of 19 surveys); the remaining two surveys were based on project records. The USA was by far the most surveyed country (8 surveys), followed by the UK, Australia and Finland (all 2). The sample sizes were generally small: six surveys returned between

1 and 20 CASE using organisations/projects; five returned between 51 and 100 with the largest sample size being 313 organisations (Howard and Rai, 1993). Eight surveys looked at both upper and lower CASE tools; five looked at upper CASE only; one at lower CASE only, whilst five surveys failed to define what was meant by "CASE".

Only two surveys attempted to quantify CASE impact (Banker and Kauffman, 1991 and Low and Jeffery, 1991). These surveys only addressed productivity impact; quality impact was not measured. Banker and Kauffman reported exceptionally high productivity results for First Boston Bank's use of CASE in contrast Low and Jeffery concluded that lower CASE benefits were not statistically significant when compared to "conventional" approaches (i.e. PL/1). The overall assessment of productivity impact was (n=19):

- Positive perceptions (7).
- Productivity not addressed by survey (6).
- Inconclusive/depends on IT maturity (4).
- Positive quantified results (1).
- Negative perceptions (1).

The overall assessment of quality impact was:

- Quality not addressed by survey (11).
- Positive <u>perceptions</u> (6).
- Depends on IT maturity (1).
- Negative <u>perceptions</u> (1).

These figures show that CASE impact is often not addressed, mainly because a number of surveys were investigating the uptake of CASE rather than CASE impact.

### 6.2.2 CASE Case Study/Action Research

The 9 case study/action research publications reviewed are listed below:

> Bendure (1991); Gavin and Little (1994); Katz (1990); Land *et al.* (1992); McChesney and Glass (1993); Orlikowski (1990); Orlikowski (1993); Software Management (1992); Swanson *et al.* (1991).

**Discussion**

Five of the studies are of US practice, four of UK practice. Four of the studies are based on a single "case"; two look at two cases; one study at three; one at four and one at six (Software Management, 1992). One study is action research (Gavin and Little, 1994); one is normative writing by a participant (Katz, 1990); two others may be normative writings by participants or may be case studies, the approach is unclear from the papers (Bendure, 1991, and Swanson *et al.*, 1991). The remaining five studies are written by non-participants. As with the CASE survey literature, the predominant views expressed are those of the IT specialists; only three studies report user views. Upper CASE tools are the subject of five studies; integrated CASE tools are reviewed in three studies; lower CASE tools in two and IPSEs in one.

Only one study attempted to quantify CASE impact (Swanson *et al.*, 1991). The study addressed both productivity and quality impact and reported exceptionally high productivity and quality results for the two projects developed by the Application Software Factory studied. The overall assessment of productivity impact was (n=9):

- Positive perceptions (4).
- Productivity not addressed by study (3).
- Positive quantified results (1).
- Negative perceptions (1).

The overall assessment of quality impact was:

- Positive perceptions (5) (includes Orlikowski's PCC "developers", 1993).
- Quality not addressed by study (3).
- Positive quantified results (1).
- Negative perceptions (1) (includes Orlikowski's PCC "users", 1993).

As with the CASE survey literature, the above figures show that CASE impact is often not addressed; the reasons here being that one study looked at implementation only (Land *et al.*, 1992), another looked at the analysis phase only (Gavin and Little, 1994), whilst a third did not report impact results (McChesney and Glass, 1993). The four studies where participants reported on their own work all made great claims of CASE success. In contrast, Land *et al.* (1992) and Orlikowski (1990) and (1993) report that CASE implementation is a complex socio-political process with both winners and losers and McChesney and Glass (1993) criticise SSADM's technical focus which discourages user involvement. The Software Management report (1992), whilst generally very positive, indicated that CASE can encourage "analysis paralysis" and "gold plating" and that generated code can have limited functionality and be difficult to modify.

### 6.2.3 CASE Laboratory Experiments

The 3 laboratory experiments reviewed are listed below:

Gryczan and Kautz (1990); Prokit (1991); Yellen (1990).

## Discussion

One study originated from the USA, one from the UK and one from Germany. One study looked at upper CASE, one at lower CASE and one at ICASE. Gryczan and Kautz (1990) and Yellen (1990) used university students as subjects; Prokit (1991) used experienced staff from the tool vendor being reviewed. Only Yellen quantified his results by measuring quality impact using *ad hoc* metrics for DFD correctness, completeness and communicability. The overall assessment of productivity impact was (n=3):

- Not addressed (2).
- Positive <u>perception</u> (1).

The overall assessment of quality impact was (n=3):

- Negative <u>perception</u> (1).
- Positive quantified result (1).
- Not addressed (1).

Gryczan and Kautz reported that tool inflexibility led to developer frustration and to work-arounds. The McDonnell Douglas Prokit and PRO-IV tools (Prokit, 1991) were reported to be "poorly integrated" with much detailed work required to be done in the lower CASE tool (PRO-IV) in order to get a generated system to work.

## 6.2.4 CASE Normative Writings

The 17 normative writings reviewed are listed below:

Aaen (1993); Aaen (1994); Aaen and Sorensen (1991); Avison *et al.* (1992); Chikofsky *et al.* (1992); Coad and Yourdon (1991); Coupe (1994); Huff (1992); Huff *et al.* (1992); Kemerer (1989); Kemerer (1992); Lyytinen *et al.* (1989); Lyytinen (1990); McGinnes (1994); Smolander *et al.* (1990a); Thompson (1993); Wynekoop and Conger (1991).

### Discussion

A number of writers advocate that CASE tools should not be introduced until the software development organisation has reached a high level of maturity (measured via the Software Engineering Institute's maturity models) (Huff, 1992, Huff *et al.*, 1992 and Thompson, 1993). "Organisational issues" (organisation size, model size and tool diffusion) are seen as having a major impact on CASE success, technical issues are less important (apart from tool response time) (Aaen and Sorensen, 1991). Upper CASE tools are criticised for inflexibility, being error-prone and having poor functionality (Aaen and Sorensen, 1991); having rigid graphics and being over-sold (Coad and Yourdon, 1991); and being inflexible and uninspiring (McGinnes, 1994).

Analysts struggle to cope with organisational politics (Avison *et al.*, 1992). Furthermore, rational IS development methods can eliminate *context* which can result in the wrong problem being solved; chaos theory usefully describes organisational complexity and explains why reductionism can fail to cope with complexity (Avison *et al*, 1992). There are no widely accepted methods of CASE tool assessment (Chikofsky *et al.*, 1992) and users must acknowledge CASE learning curve (Kemerer, 1992); overall, culture change is the key (Huff, 1992).

A number of writers criticise the quality of CASE research produced to date. There is a need for more objective research with quantified results and to hear the views of the users of the systems being developed, not just the IT specialists (Coupe, 1994). We need more rigour; there is no proof of CASE benefits yet; the metrics collected are poor; and organisational issues are ignored although very important (Kemerer, 1989). Lyytinen *et al.* (1989) identify an over-focus on technical issues and a need for more empirical work whilst Wynekoop and Conger (1991) identify applied research (e.g. tool development) and normative writings as the dominant research approach evident in the CASE literature with no action research undertaken to date. Therefore we still have a poor understanding of how analysts analyse and how useful tools are. Case studies are generally of poor quality and triangulation of results is needed.

# 6.3. IS Development Literature

## 6.3.1 IS Development Survey Research

The four survey papers reviewed are listed below:

> Banker *et al.* (1991); Boehm (1981); Cusumano and Kemerer (1990); Necco *et al.* (1987).

### Discussion

All four publications address IS development, although both Boehm and Cusumano and Kemerer focus primarily on software engineering. The survey respondents were either IT specialists (2 publications) or were based on project records (2). Three of the surveys were conducted in the USA, the fourth compared US and Japanese software development practice (Cusumano and Kemerer, 1990). The sample sizes were generally small, all between 20 and 100 respondents/projects. The overall measurement of productivity was (n=4):

- Quantitative (3).
- Qualitative (1).

The overall measurement of quality was (n=4):

- Not addressed (2).
- Quantitative (1).
- Qualitative (1).

Banker *et al.* (1991) found that the use of structured methods had a negative impact on maintenance productivity and that IS staff capability and development hardware response time had positive impacts. Boehm (1981) found that system size and personnel capability were the major productivity drivers. Cusumano and Kemerer (1990) found that system size was the major quality driver, whilst software reuse was the major productivity driver. Finally, Necco *et al.* (1987) found that user involvement and senior management support were the two major IS development success factors; that DFDs and data dictionaries were widely used and well received; that prototyping was very successful; but that current IS development performance left some room for improvement.

## 6.3.2 IS Development Case Study/Action Research

The 13 case study/action research publications reviewed are listed below:

> Avison and Wood-Harper (1990); Boland and Day (1989); Galliers (1985); Kling (1987); Kling and Iacono (1984) and (1989); Markus (1983); Mouakket *et al.* (1994); Newman and Noble (1990); Newman and Robey (1992); Orlikowski (1992); Sauer (1993); Walsham (1993).

**Discussion**

Eight of the studies are based entirely in the USA, four entirely in the UK and one combines UK and "third world" practice (Walsham, 1993). The studies are predominantly single "case" (9 studies); two studies present two cases; one study presents three cases and one study presents six cases (Avison and Wood-Harper, 1990). Three studies adopt an action research approach (Galliers, 1985, Avison and Wood-Harper, 1990 and Mouakket *et al.*, 1994). Both IT specialist and user views are expressed in the studies but only Avison and Wood-Harper (1990) discuss IS metrics; instead, most of the studies focus on the socio-political process of IS development and the subjective nature of IS assessment.

Avison and Wood-Harper identify seven lessons from their action research:

1. A methodology takes time to learn.
2. The waterfall model is inappropriate.
3. The methodology is not a "guarantor of truth".
4. The political dimension is important.
5. Responsible participation is contingent.
6. In certain situations the methodology gives insufficient guidance.
7. The methodology is interpreted by users and analysts.

and draw three conclusions:

1. The Multiview methodology is in a continuing state of development.
2. Defining an IS is contingent on methodology, analyst and situation.
3. Defining an IS can be considered as a social process.

Boland and Day (1989) adopt a "phenomenological" research approach which "suspends theories" about IS development and lets the analyst describe what her experience meant to her in her own words; the analyst was excited by the prospect of IS development initially but was soon worn down by the political nature of her work. The study focused on "significant moments" as perceived by the analyst. Three issues were of central concern:

1. Bi-polar battles between managers.
2. Interaction with users.
3. Moral dilemmas.

Galliers (1985) identifies senior management commitment and participation as key success factors whilst Kling (1987) argues that organisational boundaries are social, not based on formal organisational structure and proposes a *web model* of organisations in contrast to the prevalent *discrete-entity* models. Web models acknowledge:

- social relations
- actual infrastructure
- history of commitments,

whereas discrete-entity models are:

- a-contextual
- a-historical
- and assume adequate resources.

Kling warns that failure to adopt a web perspective can mean that causes of problems are not correctly diagnosed and that remedial action is inadequately identified; in other words discrete-entity models offer an "... impoverished conceptual vocabulary ..." (p. 350).

Kling and Iacono (1984) describe the PRINTCO case study and propose four major metaphors for IS development:

1. Technological evolution.
2. Economic rationality.
3. Organisational drift.
4. Organisational politics.

They identify the fourth model as most relevant, particularly the role of coalitions and of ideology and language. Kling and Iacono (1989) revisit PRINTCO and propose that information systems can become "institutions" which are difficult to change; they identify usability and stable social structures as key to IS longevity. In a similar vein Markus (1983) reviews the development of the Financial Information System using an historical research approach that covers 15 years of the system's life; the study focuses on resistance and the political struggle between the head office and the divisions. Markus proposes three theories of resistance:

1. People-determined resistance.
2. System-determined resistance.
3. Interaction theory (between people and system).

Interaction theory is shown to have greater explanatory power.

Mouakket *et al.* (1994) highlight the *ad hoc* nature of IS development in a case study of a university student records system whilst Newman and Noble (1990) use a case study of a US university student information system (SIS) to evaluate four <u>process</u> models of user involvement in IS development:

1. The learning model (specialist-led design/one-way learning or user-centred design/mutual learning).
2. The conflict model (leading to mutually acceptable solution).
3. The political model (solution satisfies the most powerful parties).

4. The garbage can model (all participants lack clear goals, time and knowledge; choices made by oversight).

The study focuses on "key episodes" of the SIS project and shows that the models can be applied to different stages of the same project; the SIS development progressed from garbage can to one-way learning to conflict/political model to mutual learning. Newman and Noble recognise that the learning model can break down where powerful political forces are at work and where conflicts of interest are inherent in the project context. Newman and Robey (1992) revisit the SIS case study and add another study of the Hartfield Insurance Corporation in proposing another process model of IS development based on user-analyst relationships. Process models are contrasted with the more common factor models, which are seen as useful for identifying what happened but inadequate for answering why? or how? Again, the model focuses on encounters and episodes over time; with choice of data based on what is "... judged to be more critical to the history of the project". The model begins by defining the "antecedent conditions" at the start of the project (e.g. analyst-led development) and proceeds to plot the course of critical encounters over time as the project moves across the spectrum from acceptance by all parties to equivocation to rejection by one or more parties. Newman and Robey claim that the model can be used for prediction as well as description in that encounters can be "characterised" in order to predict subsequent events.

Orlikowski (1992) also addresses socio-political issues in describing the Beta case study of conflict between tool designers and tool users within a large US software consultancy. Two concepts are introduced and used to interpret the Beta case study:

1. Structuration Theory which characterises social action as an interplay between meaning, power and norms which is both influenced by social context and, in turn, changes social context.

2. The Duality of Technology which shows that technology is created by human action and, in turn, is used by humans to accomplish some action.

Sauer (1993) develops a further socio-political process model of IS development based on the Mandata case study. The model describes the interaction between context, process and product (the IS) and highlights the key role of the project organisation in developing the IS as a "solution" to problems in the organisational context perceived by the supporters and the key role of the supporters in perpetuating a system's existence. Sauer argues that all problem solving is inherently flawed in that no method is capable of addressing all aspects of the organisational context; therefore the role of the project organisation developing the IS is to manage support in such a way as to minimise the impact of flaws in the IS. Sauer states the IS staff are often unable to cope with the political dimension of systems development.

Walsham (1993), like Orlikowski, uses structuration theory to develop a framework for analysing the IS process over time; the framework consists of four elements:

1. Content.
2. Social context.
3. Social process.
4. Context/process linkage.

Three case studies are analysed using the framework which shows how chaotic IS development can be and the importance of recognising developer morale as a key element in sustaining the high levels of commitment needed to see a project through. The use of a rich socio-political theory is seen as highlighting the poverty of the narrow, technically-focused structured systems development methods, which often fail to "... embody shared interests and values" (p. 236). Walsham's conclusions include the following: that senior management must be intimately involved in IS development to signal commitment, to learn and to negotiate; that the project team should have a broad composition; that all stakeholder groups should participate and that training requires adequate resources.

### 6.3.3 IS Development Normative Writings

Since the focus of the CarMaker research was on CASE and empirical work, only one publication was reviewed in this category: Avison and Fitzgerald (1988), which gives a comprehensive overview of IS development methodologies, techniques and tools.

**Discussion**

Avison and Fitzgerald describe eight IS development methodologies and seven types of tool. They propose a framework for methodology comparison with special emphasis on the philosophy of the methodology. Two contrasting paradigms are identified: the scientific paradigm and the systems paradigm. The scientific paradigm is dominant and is based on reductionism, repeatability and refutation; in contrast, the systems paradigm is holistic and is seen as being particularly applicable to living systems such as information systems; the central argument being that living systems cannot be broken down (e.g. into sub-processes) without losing the emergent properties that are the essence of such systems.

The ontology of the scientific paradigm is "realism"; the world consists of "... objectively given, immutable objects and structures ... independent of the observer's appreciation of them." (p. 289). In contrast, the systems paradigm supports ontological "nominalism", where "... different perceptions of reality are not wrong, but reflections of different viewpoints, cultures, or societies ... What is needed is a methodology that can handle a variety of different perceptions of a subjective reality." (p. 289).

## 6.4. IS Assessment Literature

The motivation for this research study was to evaluate the impact of CASE tools on systems development productivity and quality. In order to do this, the terms productivity and quality must be defined and measurements taken and compared with the literature.

### 6.4.1 Productivity

**Definition**

The most widely-used definition of productivity is

*system size / development effort.*

This metric is used by Boehm (1981), Fenton (1991), Symons (1991) and many more. Unfortunately neither system size nor development effort have a consistent definition, which makes comparison of results reported in the literature difficult. The variety of metrics used is evident from the tables below. In Chapter 7 the disparate metrics are "normalised" to a common unit of measure:

*function points per work-month*

in order to evaluate the productivity results of the CASE action research studies.

## Measurements

The results reported in the literature are presented in this section.

### 1. CASE Survey Research

Banker and Kauffman (1991) present the results from 20 IS development projects undertaken by the First Boston Bank using the HPS ICASE tool:

| | Person days | Function points | FP/person month |
|---|---|---|---|
| Mean | 589.2 | 1368.7 | 57.0 |
| Std. Dev. | 489.3 | 1630.0 | 73.4 |
| Maximum | 2193.0 | 5876.0 | 286.8 |
| Minimum | 85.0 | 97.9 | 4.2 |

**Table 6.1** Productivity: Banker and Kauffman (1991)

Note: [1] 18 person days per month.

Banker and Kauffman also cite typical productivity means:

1. MIS applications: c. 8 FPs/person-month.
2. Large financial institutions: 8 to 10 FPs/person-month.

Low and Jeffery (1991) compare lower CASE and "conventional" (PL/1) productivity in three large Australian organisations:

| IS type | No. of projects examined | Ave. effort (range) in staff-months | Ave. IS size (range) in function points | Ave. productivity (range) in FP/staff-day[1] |
|---|---|---|---|---|
| Organisation 1 | | | | |
| "Conventional" | 51 | 11.9<br>(0.9-51.3) | 164<br>(25-779) | 0.86<br>(0.70 S.D.)<br>(0.13-4.01) |
| lower CASE | 8 | 12.0<br>(3.5-21.5) | 286<br>(102-524) | 1.15<br>(S.D. 0.70)<br>(0.46-2.58) |
| Organisation 2 | | | | |
| "Conventional" | 6 | 36.6<br>(8.7-60.9) | 858<br>(97-2449) | 0.86<br>(S.D. 0.51)<br>(0.31-1.77) |
| lower CASE | 2 | 70.5<br>(60.9-80.2) | 1253<br>(643-1863) | 0.73<br>(S.D. 0.39)<br>(0.46-1.01) |
| Organisation 3 | | | | |
| "Conventional" | 5 | 47.9<br>(8.0-80.3) | 326<br>(122-463) | 0.39<br>(S.D. 0.20)<br>(0.21-0.66) |
| lower CASE | 3 | 56.0<br>(43.0-65.0) | 439<br>(392-487) | 0.35<br>(S.D. 0.04)<br>(0.32-0.40) |

**Table 6.2** Productivity: Low and Jeffery (1991)

Note: [1] 23 staff-days per staff-month.

2. CASE Case Study/Action Research

Swanson *et al.* (1991) describe the results from the first two projects delivered by the Application Software Factory:

| | **Software Factory** | **Transaction Processing Systems "norm"** |
|---|---|---|
| **Number of projects** | 2 | not stated |
| **Lines of COBOL code/day** | 862 - 2873 | 200 |
| **IS size (lines of code)** | 210k - 250k | not stated |

**Table 6.3** Productivity: Swanson *et al.* (1991)

3. IS Development Survey Research

Banker *et al.* (1991) analyse 65 COBOL maintenance projects at a US bank. Analysis productivity is measured in function points per man-month and programming productivity in source lines of code per man-month. The average maintenance project size is 118 FPs but no productivity results are presented.

Boehm (1981) analyses a database of 63 projects, mostly software engineering. All of the projects are "normalised" to facilitate comparison of productivity. The normalised unit of measure is *Delivered Source Instructions (DSI) per man-month*, where a DSI is equivalent to a line of code and a man-month is 152 man-hours (i.e. 8 hrs per day, 19 days per month):

| Project class | Number of projects | Productivity (DSI/MM) |
|---|---|---|
| Entire database | 63 | 20 - 1250 |
| 1. Mode | | |
| Organic | 23 | 82 - 1250 |
| Semi-detached | 12 | 41 - 583 |
| Embedded | 28 | 20 - 667 |
| 2. Type | | |
| Business | 7 | 55 - 862 |
| 3. Language | | |
| COBOL | 5 | 55 - 862 |
| PL/1 | 4 | 93 - 1250 |

**Table 6.4** Productivity: Boehm (1981)

Cusumano and Kemerer (1990) compare US and Japanese software development performance over 40 projects, mainly software engineering:

| | US | Japan |
|---|---|---|
| Mean project size | | |
| - work years | 102 | 47 |
| - equivalent FORTRAN source lines of code (ESLOC) | 288k | 389k |
| Productivity (ESLOC/work-year) | 7290 | 12447 |

**Table 6.5** Productivity: Cusumano and Kemerer (1990)

## 4. IS Assessment Literature

Dreger (1989) compares COBOL and LINC (Unisys 4GL) productivity. The number of projects is not stated:

| | Productivity (hours function point) | |
|---|---|---|
| | COBOL | LINC |
| Average | 20 | 1 |
| Best | 3.5 | 0.53 |
| Worst | 85.5 | 1.38 |

**Table 6.6** Productivity: Dreger (1989)

Dreger presents a table of conversion factors to facilitate comparison between 3GL and function point-based measures (p.136) including:

- COBOL: 105 SLOCs per function point.
- FORTRAN: 105 SLOCs per function point.

A classification of IS size is also presented:

"Small" IS: < 300 FPs.

"Medium" IS: 300 - 800 FPs.

"Large" IS: 801 - 1000 FPs.

"Very large" IS: > 1000 FPs.

Symons (1991) presents data from Nolan, Norton & Co's database and client projects which shows "typical 3GL" productivity as 0.10 function points per work-hour and "typical 4GL" productivity as 0.16 function points per work-hour. A further sample of "small to medium sized projects" (less than 750 FPs), both 3GL and 4GL developments (p. 137) reveals the following figures:

"Best" organisation mean productivity: 0.28 FP/work-hour.

Interpolated median organisation: 0.11 FP/work-hour.

"Worst" organisation: 0.02 FP/work-hour.

### 6.4.2 Quality

#### Definition

Defining quality is more problematical than productivity, although an ISO definition exists for the term (ISO, 1986):

> *"Quality: The totality of features and characteristics of a product or service that bear on its ability to satisfy stated or implied needs".*

Gilb (1988) defines system quality in terms of quality attributes, which are quantifiable measures defined by the user. A high quality system is one that meets it quality attribute targets. Gilb identifies four generic classes of quality attribute:

- workability

  (process capacity, storage capacity, responsiveness and other related measures)

- availability

  (reliability, maintainability and integrity)

- adaptability

  (improvability, extendability and portability)

- usability

  (entry requirement, learning requirement, handling ability and likability).

The recent European AMI initiative adopts a similar quality attribute model for software measurement (AMI, 1992). Fenton (1991) also supports Gilb s comprehensive approach, but observes that software quality data is often difficult to collect. He proposes a pragmatic subset of readily-collectable metrics embracing reliability, usability and maintainability; reliability is seen as the most important metric. Fenton proposes the following single, simple measure of system quality:

*quality = number of defects / size.*

Complimenting these "hard" quality metrics, attempts have been made to measure the softer concept of user satisfaction, which accords with the ISO definition of quality. A number of questionnaires have been developed to measure user satisfaction (Bailey and Pearson 1983, Ives *et al.*, 1983, Baroudi and Orlikowski, 1988, QA Forum 1989 and Barki and Hartwick 1993). Holistic assessment was taken a stage further at the IFIP 8.2 conference on IS assessment in 1986 (Bjorn-Andersen and Davis, 1988) where the participants recognised the essentially subjective nature of IS assessment and the importance of stakeholder perspective.

Despite these efforts, the limited uptake to date of softer metrics is shown by the fact that all of the studies reported below use "hard" (defect-based) quality metrics. In comparison, both "hard" and "soft" (user satisfaction-based) measurements were taken to assess the quality of the CASE research work undertaken by the author.

## Measurements

The results of IS quality assessment reported in the literature are presented below.

### 1. CASE Case Study/Action Research

Swanson *et al.* (1991) measured quality in terms of errors per thousand lines of code. They compare the performance of the Application Software Factory with "transaction processing systems norms":

- Application Software Factory: 0.01 - 0.25 errors/1000 LOCs.
- TPS "norm": 4.4 errors/1000 LOCs.

### 2. IS Development Survey Research

Cusumano and Kemerer (1990) define quality as the number of failures per 1000 SLOCs in the first 12 months of operation and compare US performance with Japanese performance:

- US: 4.44 failures/1000 SLOCs.
- Japan: 1.96 failures/1000 SLOCs.

3. IS Assessment Literature

Capers Jones (1989) defines a "high quality" IS as having fewer than 10 user-reported defects per 100 function points per year.

Finally, Fenton (1991, p. 228) lists six studies that report defect densities:

| Study | Defect rate (defects/KSLOC) | Software size (SLOC) |
|---|---|---|
| Akiyama, 1972 | 20 | 25k |
| Ruby, 1975 | 3 | 149k |
| Endres, 1975 | 5 | 53k |
| Thayer, 1978 | 28 | 74.6k |
| Belford, 1979 | 57 | 9.7k |
| Sunazuka, 1983 | 14 | 19.2k |

**Table 6.7** Quality: Fenton (1991)

Note: KSLOC = One thousand source lines of code.

# 6.5 IS Research Methodology Literature

The fourth domain of knowledge drawn upon by this research is IS research methodology knowledge (see figure 6.1). The state of knowledge in this domain is reviewed in this section.

## 6.5.1 Dominant Paradigm

Swanson (1987) describes information systems as "a young field of study" with, as yet, no "... general consensus as to its appropriate foundations ..." (p. 27). Nonetheless, a dominant research paradigm has emerged: positivism. Wood-Harper (1992) contrasts positivism with action research (the approach adopted in the CarMaker studies):

| | **Positivism** | **Action research** |
|---|---|---|
| **Scope** | Context-free | Context-based, dynamic |
| **Methods** | Cause-effect relationships | Insights which may not be quantifiable |
| **Role of researcher** | Detached observer | Actively involved |
| **Goals** | Set by researcher and selected participants | Negotiated with whole client group |
| **Outcomes** | Laws, generalizations | Context-dependent insights |

**Table 6.8** Comparison of Positivistic Science and Action Research

(after Wood-Harper, 1992, p. 235)

The appropriateness of positivism for improving understanding of the rich socio-political context of IS practice is questioned by Wood-Harper. Furthermore, the dominance of positivistic research designs in the MIS literature is shown graphically by Orlikowski and Baroudi's (1991) classification of articles appearing in four leading US MIS publications for the period January 1983 to May 1988[10]:

| Research Design | Frequency | Percent |
|---|---|---|
| Survey | 76 | 49.1 |
| Laboratory Experiment | 42 | 27.1 |
| Case Study | 21 | 13.5 |
| Mixed Method | 5 | 3.2 |
| Field Experiment | 4 | 2.6 |
| Instrument Development | 4 | 2.6 |
| Protocol Analysis | 2 | 1.3 |
| Action Research | 1 | 0.6 |
| **Total** | 155 | 100 |

**Table 6.9** Articles Classified by Research Design

(after Orlikowski and Baroudi, 1991, p. 4)

[10] The publications reviewed were *Communications of the ACM*, *Proceedings of the International Conference on Information Systems*, *Management Science* and *MIS Quarterly*.

The domination of positivism (for example, surveys and laboratory experiments) has been identified as a cause for concern by many IS researchers (Benbasat *et al.*, 1987, Kaplan and Duchon, 1988, Galliers, 1991, Orlikowski and Baroudi, 1991, Newman and Robey, 1992, Wood-Harper, 1992 and Walsham, 1993) and similar concerns have been raised in the CASE research literature concerning the domination of normative writings and applied research (Kemerer, 1989, Wynekoop and Conger, 1991 and Coupe, 1994).

### 6.5.2 Pluralism

Many of the above writers call for an acknowledgement of the inherent advantages and disadvantages of each research approach, and a willingness to select the approach to suit the goals of the researcher and the nature of the research topic. Galliers (1991, p. 337) has produced a classification of research approaches which defines their key features, strengths and weaknesses:

| Approach | Key Features | Strengths | Weaknesses |
|---|---|---|---|
| Laboratory experiments | Identification of precise relationships between chosen variables via a designed laboratory situation, using quantitative analytical techniques, with a view to making generalisable statements applicable to real-life situations. | The solution and control of a small number of variables which may then be studied intensively | The limited extent to which identified relationships exist in the real world due to over-simplification of the experimental situation and the isolation of such situations from most of the variables that are found in the real world. |
| Field experiments | Extension of laboratory experiments into the real-life situations of organisations and/or society. | Greater realism; less artificial/sanitised than the laboratory situation. | Finding organisations prepared to be experimented on. Achieving sufficient control to enable replication, with only the study variables being altered. |
| Surveys | Obtaining snap shots of practices, situations or views at a particular point in time (via questionnaires or interviews) from which inferences are made (using quantitative analytical techniques) regarding the relationships that exist in the past, present and future. | Greater number of variables may be studied than in the case of experimental approaches. Description of real world situations. More easy/appropriate generalisations. | Likely that little insight obtained re. the causes processes behind the phenomena being studied Possible bias in respondents (cf. self-selecting nature of questionnaire respondents); the researcher, and the moment in time at which the research is undertaken. |

| Approach | Key Features | Strengths | Weaknesses |
|---|---|---|---|
| Case studies | An attempt at describing the relationships which exist in reality, usually within a single organisation or organisational grouping. | Capturing 'reality' in greater detail and analysing more variables than is possible using the above approaches. | Restriction to a single event/organisation. Difficulty in generalising, given problems of acquiring similar data from a statistically meaningful number of cases. Lack of control of variables. Different interpretations of events by individual researchers/stakeholders. |
| Forecasting, Futures Research | Use of such techniques as regression analysis and time series analysis, or the delphi method and change analysis, to extrapolate/deduce likely/future possible events or impacts. | Provision of insights into likely future occurrences in situations where existing relationships may not hold true in the future. Attempts to deal with the rapid changes taking place in IT and their impacts on individuals, organisations and society in general. | Complexity and changing relationships of variables under study. Lack of real knowledge of future events. Scenarios are not 'true' picture of the future but enable decisions re. reactions in different 'futures'. Dependent on precision/relevance of past data and expertise of scenario builders. Possibility of self-fulfilling prophesies. |

| Approach | Key Features | Strengths | Weaknesses |
|---|---|---|---|
| Simulation, game role playing | An attempt at copying the behaviour of a system which would otherwise be difficult/impossible to solve analytically, by the generation/introduction of random variables. | Provision of an opportunity to study situations that might otherwise be impossible to analyse. | Similar to experimental research in regard to the difficulties associated with devising a simulation that accurately reflects the real world situations. |
| Subjective, argumentative (cf. phenomenology, hermeneutics) | Creative research based more on opinion/speculation than observation, thereby placing greater emphasis on the role/perspective of the researcher. Can be applied to an existing body of knowledge (reviews) as well as actual/past events/situations. | Useful in building theory that can be subsequently tested. Creation of new ideas and insights. Recognition that the researcher will interpret what is being studied in a particular way. Contributes to cumulative knowledge. | Unstructured, subjective nature of research process. Despite making the prejudice of the researcher known, there is still the likelihood of biased interpretations, a problem which is compounded by the time at which the research is undertaken. |

| Approach | Key Features | Strengths | Weaknesses |
|---|---|---|---|
| Action research | Applied research where there is an attempt to obtain results of practical value to groups with whom the researcher is allied, while at the same time adding to theoretical knowledge. | Practical as well as theoretical outcomes most often aimed at emancipatory outcomes. Biases of researcher made known. | Similar to case study research, but additionally places considerable responsibility on the researchers when objectives are at odds with other groupings. The ethics of the particular research are a key issue. |

**Table 6.10** A Summary of the Key Features, Strengths and Weaknesses of Alternative Information Systems Research Approaches (after Galliers, 1991, p. 337)

Wynekoop and Conger (1991) present a similar research taxonomy for the CASE literature which identifies eight research methods. Their classification is compared with Galliers' classification in the following table.

| Research approach/method | Galliers (1991) | Wynekoop and Conger (1991) |
|---|---|---|
| Laboratory experiment | yes | yes |
| Field experiment | yes | yes |
| Surveys | yes | yes |
| Case studies | yes | yes |
| Forecasting, futures research | yes | no |
| Subjective, argumentative | yes | "normative writings" |
| Action research | yes | yes |
| Applied research | no | yes |
| Basic research | no | yes |

**Table 6.11** Comparison of IS Research Taxonomies

The table shows that there is considerable agreement between the two classification schemes. Wynekoop and Conger add two further methods: applied research and basic research[11].

[11] Both basic and applied research may be seen as generic terms which describe the purpose of the research rather than the method; in which case they are not directly comparable with Galliers' approaches.

**Applied Research**

The goal is to solve a known problem by developing a product. The product may be a CASE tool for example, or a survey instrument, or a systems development methodology or a tool evaluation methodology.

**Basic Research**

Again, the problem is known, but there is no obvious route to a solution. Basic research aims to develop new theories, for example the development of a theory of software engineering to understand better the application development process (Wynekoop and Conger, 1991, p. 308).

In Chapter 7, Wynekoop and Conger's CASE taxonomy is used to ground the research approach adopted for the CarMaker studies.

### 6.5.3 Choice of Research Approach for CarMaker CASE Studies

Many of the above writers argue that the researcher should be aware of the range of research approaches available and should select the approach most appropriate for the research proposed. This is sound advice, assuming that the researcher has a good understanding of the variety of approaches and the freedom to select the approach. Neither was true at the start of the CarMaker CASE studies. The author was aware of some of the popular approaches such as surveys, laboratory experiments and case studies but was not familiar with the classification literature. Indeed, neither Galliers (1991) nor Wynekoop and Conger's (1991) papers had been published at that time[12]. More importantly, even if the variety of possible approaches had been fully appreciated, it is unlikely that the author's funders, CarMaker, would have supported research that did not deliver clear benefits to the company. Furthermore, the work was initiated by an approach from ToolVendor to evaluate their CASE tools in practice. Given the need to deliver practical benefits to both CarMaker and ToolVendor, the newness of CASE technology and the dearth of evaluative CASE research, an approach that drew upon both *action research* and *grounded theory* was clearly an appropriate approach to adopt. A *phenomenological* element was added as the work progressed by way of the diary writing undertaken by the author and the Consultant; the resulting hybrid approach is described more fully in lesson 21 (section 5.2.10).

Benbasat *et al.* (1987) support this reasoning in their review of the case study approach by arguing that due to the constant change and innovation in IT, IS researchers are often "... trailing behind practitioners in proposing changes or in evaluating methods for developing new systems" (p. 370), therefore the case study is seen as a way of capturing practitioner knowledge in this fast-moving area. They propose three motivations for adopting a case study approach:

[12] Although earlier thinking on IS research approaches had been published by, for example, Galliers (1985) and Galliers and Land (1987).

Motivation 1:

study IS in a natural setting,

learn about "state of the art",

generate theories from practice.

Motivation 2:

answer "how" and "why" questions i.e. understand the nature and complexity of the processes taking place.

Motivation 3:

study an area in which few previous studies have been carried out.

The hybrid research approach adopted for this study offers all of the above opportunities and goes further in that the researcher is not trying to keep up with practice, instead he takes the lead by becoming a practitioner. This was the opportunity presented to the author in the spring of 1990. Keen (1991) makes this point forcefully:

> *"Relevance must drive rigor. Until relevance is established, rigor is irrelevant. When relevance is clear, rigor enhances it."* (p. 47)

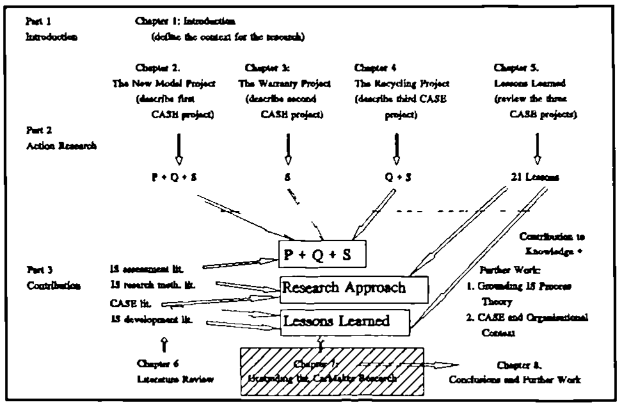
The work was relevant to CarMaker, and relevant to ToolVendor, rigour has been added by the subsequent analysis (chapters 2, 3 and 4), interpretation (chapter 5) and grounding of the data in the extant CASE and IS development literature (chapter 7).

# Conclusion to Chapter 6

The aim of Chapter 6 was to present current thinking in the four focal domains of discourse: CASE, IS development, IS assessment and IS research methodology. The work has been presented largely without commentary. The purpose of Chapter 7 is to use this "raw material" to ground the CarMaker action research.

# *Chapter 7*

# *Grounding the CarMaker Research*



## 7.1 Introduction

The aim of this chapter is to bring together all of the work described so far; to compare the results of the CarMaker CASE research with the literature described in Chapter 6 and to identify where a contribution to knowledge has been made. The diagram above shows that each chapter has generated some results. Chapter 2 (the New Model study) generated productivity and quality metrics and stakeholder views (P + Q + S), Chapter 3 (the Warranty study) contributed another set of stakeholder views (S), Chapter 4 (the Recycling study) added quality metrics and stakeholder views, whilst Chapter 5 reviewed the action research and summarised the experience in the form of lessons learned. The relevance of the research can only truly be appreciated by placing the results in the context of the IS literature. Therefore a review of the IS literature was undertaken in Chapter 6. In this chapter the significance of the action research programme is evaluated by grounding the CarMaker work in the literature. The work makes a contribution in each of three areas:

1. Research approach.
2. Productivity and quality results.
3. Lesson learned.

Each of these contributions is addressed in the following sections.

## 7.2 Grounding the Research Approach

In Section 6.5 the IS research methodology literature was reviewed and the dominance of positivism identified. The domination of a single approach is regarded as potentially harmful by many, and a call for pluralism and better understanding of strengths and weaknesses has been made (see for example Galliers, 1991) Wynekoop and Conger's (1991) taxonomy of the CASE literature represents an excellent foundation from which to identify the methodological contribution that the CarMaker work has made to this debate. Wynekoop and Conger identified two dominant research strategies; normative writings with the purpose of describing and understanding CASE, and applied research with the purpose of tool building. Out of 40 CASE publications reviewed, 12 fell into the former category and 15 into the latter. It is worth noting that there were few case studies or evaluative works, and no action research[13].

Table 7.1 updates Wynekoop and Conger's matrix to include all of the CASE literature reviewed by the author in Section 6.2 together with the papers from the British Computer Society CASE conferences described in Section 1.1 (Spurr and Layzell, 1990 and 1992). Wynekoop and Conger's original count of publications is shown, together with the reference numbers for the BCS conferences and each of the publications reviewed in Section 6.2 in parentheses. The second section of the

[13] In order to compare the CarMaker research with Wynekoop and Conger's classification scheme, the CarMaker studies will be classified as *action research* for the purpose of comparison, whilst recognising that the research approach adopted was a hybrid of action research, grounded theory and phenomenology (see lesson 21). Wynekoop and Conger do not mention grounded theory or phenomenology in their paper.

Bibliography to this thesis shows the CASE publications in reference number order. The CASE literature reviewed in this thesis has been further separated into the "Initial Selection" (reference numbers 1 to 46) and the "BCS CASE Conference papers" (reference numbers 51 onwards). The matrix shows that the BCS CASE conference proceedings follow a very similar pattern to Wynekoop and Conger's original results i.e. a dominance of normative writing and applied tool building research. Further inspection of the updated matrix reveals that the quantity of *survey research* and e*valuation research* has increased considerably since Wynekoop and Conger's study (from 7.5% of the original 40 publications to 25.3% of the 75 additional publications for the survey method and from 22.5% to 42.7% for evaluative research).

The CarMaker CASE research is counted as a single "publication" in the matrix and can be seen to be one of only two CASE action research studies identified by the author to date, the other being the work of Gavin and Little (1994). The addition of a number of in-depth longitudinal case studies (Land *et al.*, 1992, McChesney and Glass, 1993 and Orlikowski, 1993) has increased the richness of the CASE literature, but the matrix shows clearly that positivism (for example, surveys and applied research) and normative writings still hold sway; there is still an absence of context-rich research. The implications of this observation are discussed in Section 7.3.3 when the lessons arising from the CarMaker studies are reviewed.

|  | Research Method |  |  |  |  |  |  |  |
|---|---|---|---|---|---|---|---|---|
| **Purpose** | **Case** | **Field** | **Lab.** | **Action** | **Survey** | **Applied** | **Basic** | **Normative** | **Purpose totals** |
| **Understand /describe** | (***27,31,33***) |  |  |  | (***12,18,22,38***) | 1 |  | 12<br>(***1,4,9,10,19,20,23,24,25,29,30,43,45*** 51,58.61,65,66,71,74,75,78) | 13<br>(20 and 9) |
| **Engineer** |  |  |  |  |  | 15<br>(***5,32,37*** 55,56,59,60,62,70,72,73,79,80) |  |  | 15<br>(3 and 10) |
| **Re-engineer** |  |  |  |  |  | 3<br>(77) |  |  | 3<br>(0 and 1) |
| **Evaluate** | 4<br>(***39,42***) |  | 1<br>(***16,35,46,***57) | **CarMaker Studies plus**<br>(***14***) | 3<br>(***2,3,6,8,13,15,17,21,26,28,34,36,40,41,44***) | (64,76) |  | 1<br>(***7,11*** 52,53,54,67,69) | 9<br>(24 and 8) |
| **Method totals** | 4<br>(5 and 0) | 0<br>(0 and 0) | 1<br>(3 and 1) | 0<br>(2 and 0) | 3<br>(19 and 0) | 19<br>(3 and 13) | 0<br>(0 and 0) | 13<br>(15 and 14) | 40<br>(47 and 28) |

**Table 7.1** Updated Classification of CASE Literature (update to Wynekoop and Conger, 1991, p. 312)

Notes

1. The figures in the cells not in parenthesis are Wynekoop and Conger's original totals.
2. The figures in parentheses in the cells show the publication reference number in the second section of the Bibliography for, firstly, the author's initial CASE selection (bold and italics), and secondly the BCS CASE conferences (normal text).
3. The three totals figures shown in the Purpose Totals column and Method Totals row represent Wynekoop and Conger's original figures, the author's initial CASE selection and the BCS CASE conferences respectively

# 7.3 Grounding the Research Results

In the first part of this section the productivity and quality metrics and stakeholder views collected from the CarMaker studies are compared with the literature. In the second part, the lessons learned from the CarMaker studies are compared with the issues identified in the literature in order to show where existing knowledge has been strengthened and where new knowledge has been created.

## 7.3.1 Grounding the Productivity and Quality Results

### 1. Productivity

### The New Model Project

The following table compares the productivity result from the New Model project with the literature. Note that all the studies have been "normalised" to function points and work-months using the following ratios:

1 FP = 105 COBOL or FORTRAN SLOCs (Dreger, 1989).

1 work-month = 140 work-hours (7 hours per day, 20 days per month).

| Study | IS size<br>(function points) | Productivity (FP/work-month) |
|---|---|---|
| *New Model project (FourthGen)* | *320* | *77* |
| 3GLs | | |
| COBOL (Dreger, 1989) | not stated | 7 mean to 40 max [illegible] |
| MIS "mean"<br>(Capers Jones, 1989) | not stated | 8 mean |
| Large financial institutions (Banker and Kauffman, 1991) | not stated | 8 to 10 mean |
| PL/1 "conventional" (Low and Jeffery, 1991) | | |
| Organisation 1: | 164 mean to 779 max | 17 to 80 |
| Organisation 2: | 858 to 2449 | 17 to 35 |
| Organisation 3: | 326 to 463 | 8 t 13 |
| "Transaction processing systems norm" (Swanson *et al.*, 1991) | not stated | 38 mean |
| COBOL maintenance work (Banker *et al.*, 1991) | 118 mean | not stated |
| Business/COBOL applications (Boehm, 1981) | not stated | 0.5 minimum to 8 maximum (mean not stated) |
| Cusumano and Kemerer (1990) | | |
| - US | 2743 mean | 6 mean |
| - Japan | 3705 mean | 10 mean |
| "Typical 3GL"<br>(Symons, 1991) | not stated | 14 mean |
| | | |
| 4GLs | | |
| LINC (Dreger, 1989) | not stated | 140 mean to 264 max |
| HPS ICASE tool (Banker and Kauffman, 1991) | 1369 mean to 5876 max | 57 mean to 287 max |
| Back-end CASE (Low and Jeffery, 1991) | | |
| - Organisation 1 | 286 mean to 524 max | 23 mean to 52 max |
| - Organisation 2 | 1253 mean to 1863 max | 15 mean to 20 max |
| - Organisation 3 | 439 mean to 487 max | 7 mean to 8 max |
| Application Software Factory (Swanson *et al.*, 1991) | 2190 and 2381 (two projects) | 356 and 547 |
| "Typical 4GL"<br>(Symons, 1991) | not stated | 22 mean |

**Table 7.2** Productivity Comparison

The above table shows that the New Model system was a small system, but by no means tiny. In fact, Dreger's (1989) classification of IS size (Section 6.4.1) places the system at the bottom end of the "medium" category. The productivity figure of 77 FPs/work-month is high, even for a 4GL. Only the Application Software Factory (Swanson *et al.*, 1991), LINC (Dreger, 1989) and the higher figures for the HPS ICASE tool (Banker and Kauffman, 1991) exceed the New Model productivity rate.

### The Warranty and Recycling Projects

The Warranty project did not proceed beyond the analysis stage, therefore productivity cannot be assessed for this project. The Recycling project delivered a working system, but productivity is difficult to calculate due to the evolutionary nature of development. Therefore productivity data is not available from the project.

## 2. Quality

### The New Model Project

Defects

A summary of post-implementation defects[14] for the New Model project is presented in the following table:

[14] Defects dated 5/12/90 onwards in Appendix 2.2.

|  | Priority |  |  |  |
|---|---|---|---|---|
|  | High | Medium | Low | Total |
| Number of defects | 7 | 6 | 2 | 15 |

**Table 7.3** New Model Project Defect Summary

Defect density is one of the most widely used quality metrics (Fenton, 1991 and Symons, 1991). To calculate defect density, the number of defects is divided by system size. The figure for the New Model system has been "normalised" by converting the system size of 320 function points to 33,600 SLOCs using Dreger's (1989) conversion ratio of 105 SLOCs per FP for 3GLs. This gives a defect density of:

15 / 33600 = 0.45 defects/KSLOC for the six months of operation.

A comparison with the literature can now be made.

| Study | Defect density (defects/KSLOC) | Software size (SLOC) |
|---|---|---|
| *New Model system* | *0.45* | *33.6k* |
| Akiyama, 1972<br>(in Fenton, 1991) | 20 | 25k |
| Ruby, 1975<br>(in Fenton, 1991) | 3 | 149k |
| Endres, 1975<br>(in Fenton, 1991) | 5 | 53k |
| Thayer, 1978<br>(in Fenton, 1991) | 28 | 74.6k |
| Belford, 1979<br>(in Fenton, 1991) | 57 | 9.7k |
| Sunazuka, 1983<br>(in Fenton, 1991) | 14 | 19.2k |
| Capers Jones, 1989 | a "high quality system" has less than 10 defects / 100 FPs per year i.e. a defect density of c. 1 defect per KSLOC | not applicable |
| Cusumano and Kemerer, 1990 | | |
| - US | 4.44 mean | 288k mean |
| - Japan | 1.96 mean | 389k mean |
| Swanson *et al.*, 1991 | | |
| - Application Software Factory | 0.01 and 0.25<br>(two projects) | 210k and 250k |
| - "Transaction processing systems norm" | 4.4 | not stated |

**Table 7.4** Defect Comparison

The defect density comparison shows that the New Model project had a remarkably low level of defects. Only the Application Software Factory (Swanson *et al.*, 1991) had better figures.

Stakeholder Views

The stakeholder views for the New Model project cannot be directly compared with the quality literature since none of the publications reviewed present a stakeholder analysis. Instead, a summary of the stakeholder views presented in Chapter 2 follows.

Overall, the New Model stakeholders were pleased with the system features. Surprisingly, the failure of the PSR totals to add up was not mentioned. The biggest problems were inflexible reporting and lack of on-site support. Interestingly, Andrew reported that the SystemsHouse replacement system was not entirely trouble-free either.

## Warranty Project

### Defects

The project did not proceed beyond the analysis stage, therefore defects were not recorded.

### Stakeholder Views

The project was perceived as a valuable exercise in clarification of Warranty activities and information needs. Jack and Malcolm certainly benefitted from the transfer of analytical skills which enabled East Factory to lead the corporate re-organisation of problem management and to retain a local Warranty function. The project may have achieved more if it had been better supported, if regular deliverables had been provided and if "analysis paralysis" had been avoided.

**Recycling Project**

Defects

The evolutionary nature of the Recycling project and the limited use of data flow diagrams makes function point analysis difficult. Therefore, there are no IS size figures available for this project. However, a defect count is possible, based on the PRF system, which was used to record defects and changes for the period from 1 3/93 to 30/7/93. The defect count for this period is as follows:

| | Priority | | | |
|---|---|---|---|---|
| | **High** | **Medium** | **Low** | **Total** |
| **Defects** | 6 | 2 | 0 | 8 |

**Table 7.5** Recycling Project Defect Summary

The count for the five month period is approximately half that of the New Model system for a six month period (15 defects). However, the count of high priority defects is very similar (6 versus 7). The Recycling IS may therefore be viewed as having a similar defect quality as the New Model system. Furthermore, the two systems had similar functionality[15] (and therefore similar function point size?) If this is the case, the Recycling system could also be claimed to be of "high quality" in comparison to the defect densities reported in the literature.

## Stakeholder Views

The stakeholder views report an extremely mixed response, depending largely on user involvement and understanding of IS development. Geoff and Robert were very satisfied with the quality of information provided, although reservations were expressed about usability, on-site support and training. The other four stakeholders were all very dissatisfied with the IS (all IS *product* scores of less than -1 in the UIS survey), although only one was a regular user of the IS (Jim). Jim may have been influenced by Ronnie's continual criticism of the IS (they worked closely together on vehicle stripping; Ronnie as stripper, Jim as IS operator). Furthermore, Jim was not as experienced or as capable a user as Robert and was therefore more worried about usability problems. Chris, like Ronnie, was a volatile character who appeared to be influenced by criticism of the IS by Ronnie. However, he had some cause for complaint since the materials reports were delayed by the chaos surrounding the project and the part-material relationship was poorly supported until the July modification, almost four months after Chris completed the UIS questionnaire. Finally, Terry admitted that his views were perceptions. The unreliability of his

[15] They both provided screen-based add/change/lookup and delete facilities and various management reports, although the reporting algorithms and use of graphics in the Recycling IS represented a higher level of complexity than the New Model system.

perception was shown clearly by his view that response time was poor whereas Robert, a regular user, stated that response time was "very good"! Unfortunately, Terry was also a powerful stakeholder, therefore his "perceived" negative views carried considerable weight, especially in terms of their impact on the author's and Mary's morale.

### 7.3.2 Quality Systems Quickly?

The research question that motivated this study was "Do CASE tools enable quality systems to be developed quickly?". In this section the preceding productivity and quality analysis is used to answer this question.

The New Model project delivered a small to medium-sized system very quickly in terms of programming productivity prior to implementation. The system had a very low defect density compared to the other systems cited in the literature, but suffered from several serious defects. Nonetheless, the users were generally satisfied with the system's functionality but were very dissatisfied with the lack of on-site support and the resulting inflexibility of the system when new reports were required. These factors, plus the availability of a substitute system at the South Factory meant that the New Model system was phased out after six months of operation. To answer the research question, it could be claimed that a quality system was delivered quickly, but that the system context was ignored and system maintainability was poor. This left the author exposed to changing requirements with inadequate IT tools to cope with demand and inadequate conceptual tools to enable the situation to be understood better.

The Warranty project, although not leading to system implementation, was arguably the most successful of the three projects. The users were very satisfied with the quality of analysis and with the skills transferred. The project contributed substantially to local IS development by Jack and Malcolm and to the corporate re-organisation of problem management. As with the New Model project, context was largely ignored, which led to a lack of commitment on behalf of Warranty management and a failure to integrate the Research Centre/ToolVendor team into the local IS development and the company-wide problem management review activities. Analysis gradually ground to a halt, partly encouraged by upper CASE tool "box filling" and a general lack of direction not remedied by the structured techniques being used. In conclusion, the project could be claimed to have delivered a quality "model" slowly. The failure to address the context of the work frustrated the author and the Consultant but may well have suited Jack and Simon.

In contrast to the Warranty project, the Recycling project was chaotic and acrimonious. The author's keenness to adopt an evolutionary development approach to avoid the "analysis paralysis" of the Warranty project was to prove costly in terms of morale and workload. The Recycling IS was a source of much criticism during the project, but the defect analysis shows that the defect count was low, and, indeed, the defects were not as serious as some of the defects arising during the New Model project. The stakeholder views were strikingly polarised: Geoff and Robert were generally very satisfied with the IS, whereas Jim, Chris, Ronnie and Terry were very dissatisfied. The major complaints concerned usability, training and lack of on-site support. The quality of information provided was highly praised by Robert and Geoff, and heavily criticised by the rest (although Chris became a supporter of the system once his material report became operational in July). Furthermore, Jim became much happier with usability and understanding once the mouse-driven version of QuickStore was installed in July. Unfortunately Ronnie and Terry remained critical until to end.

Here was an example of a "mixed" quality system; delivered in time to support Geoff and Robert but somewhat late for Chris and too late for the project management. As with the two previous projects, the context of IS development was ignored, which meant that the author and Mary were left to fight against a rising tide of requirements with inadequate training or resources and with little understanding or support from the users.

## From Context-Free to Context-Rich

The three studies show that it is very difficult to deliver a "quality system quickly" if the context surrounding the technology is ignored. The analysis could end here. The productivity and quality metrics (P + Q) and stakeholder views (S) have been used to address the research question and some tentative answers have been given. But in many ways the answers only serve to stimulate a whole new set of questions concerning why the context was not addressed and what should be done to improve matters. It may be useful therefore to provide a definition of context at this point.

Sauer's (1993, p. 71) definition is both broad and covers many of the concerns raised by the CarMaker studies; context is seen as consisting of:

- cognitive limitations
- technical process (e.g. IS development methodology and tools used)
- environment

  (suppliers, customers, competitors, technology, interest groups, regulators, culture and institutions)
- politics
- structure

  (both of the project organisation and of the host organisation)
- history.

This is where the richer lessons learned come into play. The next section locates each of the lessons in the extant literature to show where the lessons add to what is known about the context of IS development, and where new issues have been identified that have yet to be adequately addressed.

### 7.3.3 Grounding the Lessons Learned

This section represents the final stage of grounding the CarMaker research work. A novel approach has been taken to identifying where contributions have been made by using a matrix to map the lessons learned onto the literature and in so doing enabling gaps in knowledge to be readily identified. Lesson 21 (research approach) has been omitted from the matrix because the research approach has been discussed already, in Section 7.2.

## The Contribution Matrix

|  | How Addressed in the Literature |  |  |
|---|---|---|---|
| **Action Research Lesson** | **CASE Literature** | **IS Development Literature** | **Comments** |
| Category 1: Politics |  |  |  |
| 1. Use and abuse of power is evident in CASE-based systems development | Kemerer (1989) (ignore implementation politics at your peril), Land *et al.* (1992) ("champion" needed), Orlikowski (1990) (Beta management authorised tool development), Orlikowski (1993) (SCC: managers happy with tools, some users not) | Avison and Wood-Harper (1990) (lesson 4: the political dimension is important), Boland and Day (1989) (power struggle between two managers), Galliers (1985) (senior management support crucial), Kling (1987) (web model acknowledges resource dependency), Kling and Iacono (1984) (promote "organisational politics" metaphor in PRINTCO study), Necco *et al.* (1987) (senior management support vital), Newman and Noble (1990) ("coercion" in appealing to Vice-President in SIS study), Orlikowski (1992) (structuration theory: meaning, power and norms), Sauer (1993) (support management key, politics as context variable), Walsham (1993) ("political system" metaphor and Structuration Framework) | A rarely addressed issue in the CASE literature; a central issue in the holistic IS literature |
| 2. Conflict occurs both between developers and users and between different user groups | Land *et al.* (1992) (conflicting views of IPSE implementation: managers and users), Orlikowski (1990) (Beta battle between "technical consultants" and "functional consultants"), Orlikowski (1993) (SCC (Beta) battles and PCC battles between IT dept  and user depts.), Stobart *et al.* (1991) (user analyst communication a key problem) | Avison and Wood Harper (1990) (lesson 3: methodology not a "guarantor of truth"), Boland and Day (1989) (ISD "crosses paths"), Kling (1987) (web models acknowledge social relations), Markus (1983) (FIS study: head office versus divisions), Newman and Noble (1990) ("conflict of interests" may be inherent), Newman and Robey (1992) ("acceptance", "equivocation" and "rejection" stages of process model), Orlikowski (1992) (structuration theory: meaning, power, norms), Sauer (1993) ("politics" a context variable), Walsham (1993) (formal methods may not lead to shared vision) | Again, inadequately addressed in CASE literature; widely acknowledged in holistic ISD literature |

|  | How Addressed in the Literature |  |  |
|---|---|---|---|
| Action Research Lesson | CASE Literature | IS Development Literature | Comments |
| Category 2: History |  |  |  |
| 3. History is important and influences stakeholder behaviour | Huff (1992) (culture change key to CASE success), Land *et al.* (1992) (culture important), Orlikowski (1993) (established ISD methodology helps CASE implementation), Thompson (1993) (CASE unsuccessful in "immature" organisations; beware bad commitments) | Avison and Wood-Harper (lesson 7: methodology is interpreted by users and analysts according to culture, education and background), Kling (1987) (web models acknowledge "history of commitments"), Newman and Robey (1992) ("antecedent conditions" important), Orlikowski (1992) (structuration theory recognises context process relationship), Sauer (1993) ("history" as context variable), Walsham (1993) (structuration theory again, plus senior management need CBIS experience) | Huff cites "culture change" as key to CASE success, Thompson warns of bad commitments, the other writers view IS development as embedded in an historical context |
| Category 3: Technology |  |  |  |
| 4. Upper CASE tools used in background, relationship with users unchanged; poor tool integration | Aaen *et al.* (1992) (poor integration), Price Waterhouse (1989) (23% perceived integration a problem), Prokit (1991) (poor integration), Wijers and van Dort (1990) (58% state role of end user unchanged) | Orlikowski (1992) ("duality": IS developed away from user workplace) | A general agreement that CASE tools do not change the developer-user relationship and integration with other tools is generally poor |

|  | How Addressed in the Literature |  |  |
|---|---|---|---|
| Action Research Lesson | CASE Literature | IS Development Literature | Comments |
| 5. Upper CASE tool usability adequate but lower CASE usability poor and documentation poor (both upper and lower CASE) | Aaen *et al.* (1992) (u CASE liked by developers), Aaen and Sorensen (1991) (tools rigid with limited functionality), Bendure (1991) (developers liked u/CASE), Burkhard (1990) (analysts liked u/CASE), Coad and Yourdon (1991) (poor HCI/limited functionality), Gryczan and Kautz (1990) (work-arounds required), Howard and Rai (1993) (diagrammers are most used feature), Land *et al.* (1992) (poor IPSE usability), McGinnes (1994) (poor usability), Norman and Nunamaker (1989) (u/CASE provided good support for DFDs and data dictionaries), Orlikowski (1990) (work-arounds required), Orlikowski (1993) (developers liked tools:PCC), Prokit (1991) (additional work required in 4GL), Software Management (1992) (can't modify generated code), Stobart *et al.* (poor documentation) | Kling and Iacono (1989) (usability of CBIS is key CSF), Necco *et al.* (1987) (u CASE tools highly regarded), Orlikowski (1992) ("duality": usability poor if IS developed away from workplace) | Overall, upper CASE tools appear to be popular with developers but are criticised by researchers. Lower CASE tools rarely mentioned apart from Prokit (1991) which showed that experienced PRO-IV users did not have a problem |
| 6. Lower CASE tools are inflexible, some are difficult for end users to use, others have insufficiently powerful query languages | not addressed | not addressed | Lower CASE tools 4GLs often assumed to promote end-user computing through flexible query languages, the action research presented here questions this view |
| 7. Lower CASE tools have bugs | Only addressed by Aaen and Sorensen (1991) ("error prone" tools) | not addressed | Bugs in FourthGen, and to a lesser extent in QuickStore (CarMaker studies), greatly increased developer workload and reduced user confidence in the systems |

|  | How Addressed in the Literature |  |  |
|---|---|---|---|
| **Action Research Lesson** | **CASE Literature** | **IS Development Literature** | **Comments** |
| Category 4: Cognitive Limitations |  |  |  |
| 8. Requirements analysis is problematical | Avison *et al.* (1992) ("messy world", reductionism can hide real problem), Land *et al.* (1992) (wisdom of "hindsight"), Thompson (1993) (beware bad commitments) | Avison and Wood-Harper (1990) (lesson 6: sometimes methodology gives insufficient guidance), Boland and Day (1989) (changing requirements), Galliers (1985) (over-emphasis on "design" aspects), Kling (1987) ("impoverished vocabulary" of discrete-entity models), Mouakket *et al.* (1994) (requirements changed post-implementation), Newman and Noble (1990) ("Garbage Can" model), Sauer (1993) (cognitive limitations), Walsham (1993) ("narrowing" aspects of IS methods) | Another issue rarely addressed in the CASE literature, but acknowledged in the IS literature. |
| 9. Outside events can have major consequences | not addressed | Kling (1987) (PRINTCO study: difficulties in hiring IS staff), Orlikowski (1992) (structuration theory: context-process linkage), Sauer (1993) ("environment" as context variable), Walsham (1993) (structuration theory: context-process linkage) | CASE literature assumes ISD is unaffected by the outside world |
| 10. Substitute systems exist and may divert stakeholder energy; linkage to business strategy important | Burkhard (1990) (link to business strategy key CSF), Land *et al.* (1992) (IPSE needs to fit strategy; wisdom of "hindsight"), Siltanen (1990) (SISP maturity key CASE CSF) | Avison and Fitzgerald (1988) (importance of business-ISD linkage), Avison and Wood Harper (1990) (lesson 5: responsible participation is contingent, cite a "hidden" IS example), Kling (1987) (PRINTCO study: substitute end user systems appeared), Newman and Noble (1990) (heads of departments did not foresee the implications of their IS purchase decision; garbage can model), Sauer (1993) ("cognitive limitations" as context variable; competing influences on supporters) | In general, CASE literature does not recognise cognitive limitations or strategy linkage |
| 11. Prototyping is helpful but not a panacea | Bendure (1991) (prototyping successful), Kievit *and Martin (1989) (75% of IS staff "satisfied or* very satisfied" with prototyping) | Avison and Fitzgerald (1988) (prototyping a promising *future development*), Necco *et al.* (1987) (IT staff very satisfied), Newman and Noble (1990) (users don't know what they need until they see it) | Little addressed in CASE literature; most writers assume prototyping valuable but do not identify limitations |

|  | How Addressed in the Literature |  |  |
|---|---|---|---|
| **Action Research Lesson** | **CASE Literature** | **IS Development Literature** | **Comments** |
| 12. Software is "unknowable" | not addressed | Kling (1987) (PRINTCO study: complexity of MRP system not appreciated), Sauer (1993) ("cognitive limitations" context variable) | Little addressed in either literature |
| Category 5: Professionalism/ ISD methodology |  |  |  |
| 13. Problem solving is often *ad hoc*; techniques have variable value | Bendure (1991) (users found diagrams confusing plus "analysis paralysis"), Burkhard (1990) (70% used methods but structured methods had negative performance impact), Gavin and Little (1994) (successful structured analysis), Kievitt and Martin (1989) (83% of IS staff "satisfied/very satisfied" with DFDs), McGinnes (1994) (logical models (e.g. DFDs) don't inspire users), Smolander (1990) (few Finnish organisations use methods), Software Management (1992) ("analysis paralysis" due to "gold plating") | Avison and Wood-Harper (1990) (lesson 2: waterfall model is inappropriate), Banker *et al.* (1991) (structured methods had negative impact on maintenance productivity), Mouakket et al. (1994) (DFDs poor communication tool; several stages of SSADM skipped), Necco *et al.* (1987) (DFDs popular), Sauer (1993) ("systematic" versus "*ad hoc*" problem solving; always flaws) | Some conflicting views expressed regarding the efficacy of structured techniques. Developers seem to like them, but users find them confusing and "analysis paralysis" can occur. |
| 14. IS implementation and support stressful and resource-intensive | not addressed | Kling (1987) (web models recognise importance of available infrastructure; PRINTCO study shows how things start going wrong after implementation), Newman and Robey (1992) (process model can show where resources needed e.g. to "unfreeze" prior to change), Sauer (1993) (project organisation becomes more powerful once IS operational, but if IS flawed this can lead to trouble), Walsham (1993) (Processing Company study: analysts report post implementation chaos) | There is no mention in either literature of "scapegoating" of IS staff due to the tangible nature of the IS as opposed to the intangible nature of "requirements". The dangers of evolutionary IS development, particularly resource implications, are not explored. |

| | How Addressed in the Literature | | |
|---|---|---|---|
| Action Research Lesson | CASE Literature | IS Development Literature | Comments |
| 15. Inadequate training of all stakeholders; expectations of CASE and IS in general too high | Aaen *et al.* (1992) (CASE new to many organisations), Aaen (1993) (experienced firms had better opinion of CASE), Burkhard (1990) (many organisations new to CASE), Coad and Yourdon (1991) (CASE over-hyped), Dimbleby and Lo (1992) (tools and techniques widely used), Glasson *et al.* (1992) (CASE over-hyped), Hayley and Lyman (1990) (CASE new to many organisations; CASE cannot compensate for human shortcomings), Howard and Rai (1993) (most organisations "explorers": low diffusion and low infusion), Jones and Arnett (1993) (DFDs, DDs, 4GLs widely used), Kemerer (1992) (learning curve important), Land *et al.* (1992) (IPSEs are new technology; expectation control critical), Low and Jeffery (1991) (CASE experience important) | Avison and Wood-Harper (1990) (lesson 1: methodology takes time to learn), Banker *et al.* (1991) (IS staff capability had positive impact on productivity), Boehm (1981) (personnel team capability a major productivity driver), Kling (1987) (web models recognise "infrastructure available"; discrete-entity models "overstate" leverage of tools and techniques), Necco *et al.* (1987) (DFDs and DDs widely used), Walsham (1993) (training requires adequate resources, must address both technological and organisational issues) | Adequate training and the availability of experienced staff are generally recognised as important. Some confusion as to whether CASE is widely-used. Most surveys show that organisations have less than 3 years experience of tool use, and that the tools are not widely diffused (notably Howard and Rai's recent large-scale survey of US practice). However, some state that techniques such as DFDs and tools such as data dictionaries and 4GLs are widely used. This may be due to confusion in terminology over what constitutes "CASE". |
| 15. (cont.) | Price Waterhouse (1989) ("embryonic stage"; 34% perceive hype as a problem), Prokit (1991) (experienced PRO-IV users capable of rapid maintenance), Smolander (1990) (use currently "experimental"), Statland (1989) (CASE not yet widely used; 39% perceived no CASE benefits), Stobart *et al.* (1991) (CASE not yet widely used), Wijers and van Dort (1990) (CASE new to organisations), Wynekoop and Conger (1991) (many researchers refuse to accept that tools may not help) | | Most commentators are in agreement that CASE has been over-hyped and that this had led to unrealistic expectations. |
| Category 6: Culture/ Commitment | | | |

|  | How Addressed in the Literature |  |  |
|---|---|---|---|
| Action Research Lesson | CASE Literature | IS Development Literature | Comments |
| 16. User commitment is difficult to obtain/culture gap | Burkhard (1990) (user involvement is a CSF), Gavin and Little (1994) (user commitment and involvement obtained successfully), Land *et al.* (1992) (IPSE users less committed than management), McChesney and Glass (1993) (SSADM weak on stimulating user involvement) | Avison and Wood-Harper (1990) (lesson 5: responsible participation is contingent), Boland and Day (1989) (systems thinking different to users' normal thinking), Kling (1987) (web models recognise "social relations"), Kling and Iacono (1989) (institutionalisation of CBIS depends on stable social structures), Necco *et al.* (1987) (user involvement is no. 1 CSF), Newman and Noble (1990) ("organisational validity" of IS is a key factor), Sauer (1993) (support management vital, loss of support can occur due to negative evaluation of IS), Walsham (1993) (participation of all stakeholders "essential") | Clearly user involvement is very important, however, many of the studies show that it is contingent on factors such as power, interests and shared language. |
| Category 7: IS/CASE Assessment |  |  |  |
| 17. Assessment is subjective and can be misleading | CASE is generally assessed via IT staff "perception":<br>Aaen *et al.* (1992), Burkhard (1990), Doke (1989), Glasson *et al.* (1992), Hayley and Lyman (1990), Jones (1992), Kievit and Martin (1989), Norman and Nunamaker (1989), Orlikowski (1990), Price Waterhouse (1989), Siltanen (1990), Smolander *et al.* (1990), *Software Management* (1992), Statland (1989), Stobart *et al.* (1991), Wijers and van Dort (1990)<br>Both IT staff and user perceptions reported:<br>Bendure (1991), Gavin and Little (1994), Katz (1990), Orlikowski (1993) | IT staff perceptions:<br>Necco *et al.* (1987)<br>Other comments:<br>Avison and Wood-Harper (1990) (IS assessment is subjective/stakeholder-dependent), Cusumano and Kemerer (1990) (few managers collect quantified software data), Sauer (1993) (different supporters/stakeholders have different views) | Only 9 out of the 98 publications reviewed from the CASE, IS Development and IS Assessment literatures reported quantified assessment results. The vast majority of assessments were based on IT staff perceptions. |

|  | How Addressed in the Literature |  |  |
|---|---|---|---|
| Action Research Lesson | CASE Literature | IS Development Literature | Comments |
| 17. (cont.) | Coupe (1994) (identifies subjective nature of CASE assessment and the rarity of user views), Chikofsky *et al.* (1992) (no widely accepted method for tool assessment), Kemerer (1989) (no valid tool benefits research; quality depends on perspective) |  |  |
| 18. Analyst's interpretation may differ from others | Most publications state only perception of IT staff, do not compare with user perceptions.<br>Burkhard (1990) (users had less favourable views of CASE than analysts), Coupe (1994) (users rarely heard) | not addressed | A little addressed issue, showing the dangers of listening only to the perceptions of one stakeholder group |
| Category 8: Noise |  |  |  |
| 19 Noise drowns good ideas | not addressed | not addressed | This "emergent property" of ISD appears to have been overlooked |
| Category 9: Emotion |  |  |  |
| 20. Developers are not politically adept | Avison *et al.* (1992) (analysts struggle with politics) | Boland and Day (1989) (inexperienced analyst is initially excited but soon becomes weary of politics), Kling (1987) (recognises that developers can be subjected to inappropriate criticism if discrete entity view adopted e.g. WWMCCS case study), Newman and Nobel (1990) (key factors include "climate of trust"), Sauer (1993) (IS staff often unable to cope with politics), Walsham (1993) (quotes Zuboff re importance of developers' feelings and morale for sustained commitment) | The human dimension is barely recognised in the CASE literature |

**Table 7.6** The Contribution Matrix

### 7.3.4 Summary of Contribution

The results of the contribution matrix can be classified under the six headings shown in the table below. Where there were many publications that supported a lesson, the lesson is classed as having substantial support from the literature. Where there is disagreement in the literature and between the literature and the lesson, the lesson is classed under the "conflict" heading. Where the lesson has been barely addressed by the literature, the classification reflects this.

| Contribution | Lesson number |
|---|---|
| 1. Substantial support in CASE literature | 4, 13, 15, 17 |
| 2. Some conflict with CASE literature | 5, 13, 15 |
| 3. Not addressed or little addressed by CASE literature | 1, 2, 3, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16, 18, 19, 20 |
| 4. Substantial support in IS development literature | 1, 2, 3, 8, 9, 10, 11, 13, 14, 15, 16, 17, 20 |
| 5. Some conflict with IS development literature | 17 |
| 6. Not addressed or little addressed by IS development literature | 12, 14, 18, 19 |

**Table 7.7** Summary of Contribution

Note: The CASE technology lessons (lessons 4, 5, 6 and 7) are pertinent to the CASE literature but not to the IS development literature, they have therefore been omitted from the assessment of IS development "contributions" (contributions 4, 5 and 6 in above table).

**Discussion of Contribution**

There is substantial support in the CASE literature for lessons 4 (upper CASE tool popular with analysts but used in the background and poor tool integration), part of 13 ("analysis paralysis" can occur), part of 15 (CASE has been over-sold and learning curve important) and 17 (CASE assessment is subjective). However, there are conflicting views on lesson 5 (upper CASE tool usability), part of 13 (efficacy of structured techniques) and part of 15 (extent of CASE usage, diffusion and experience).

However, the majority of the lessons arising from the CarMaker research were not addressed in the CASE literature, showing that there are significant areas where a contribution to CASE knowledge and awareness can be made. The missing lessons are lessons 1 (use and abuse of power is evident in CASE-based systems development), 2 (conflict occurs both between developers and users and between different user groups), 3 (history is important and influences stakeholder behaviour), 6 (lower CASE tools are inflexible, some are too difficult for end users to use, others have insufficiently powerful query languages for complex programming tasks), 7 (lower CASE tools have bugs), 8 (requirements analysis is problematical), 9 (outside events can have major implications), 10 (substitute systems exist and may divert stakeholder energies; linkage of IS development to business strategy is important), 11 (prototyping is helpful but not a panacea), 12 (software is essentially "unknowable"), part of 13 (problem solving is often *ad hoc*), 14 (IS implementation and support is stressful and resource-intensive), 16 (user commitment is often difficult to obtain), 18 (the analyst's interpretation of CASE benefits may differ from the views of other stakeholders), 19 (noise drowns good ideas) and 20 (developers are not politically adept).

In contrast to the CASE literature, most of the lessons were supported by the IS development literature. The only area of conflict was with lesson 17 (IS assessment is subjective and can be misleading), where objective (defect-based) metrics dominate the software engineering literature (Boehm, 1981, Gilb, 1988, Fenton, 1991, AMI, 1992) whereas more subjective (user satisfaction-based) measures can be found in the information systems literature (for example Bailey and Pearson, 1983, Ives *et al.*, 1983 and QA Forum, 1989). Whilst the review of the IS development literature is by no means comprehensive, four lessons appear not to have been addressed: lesson 12 (software is "unknowable"), lesson 14 (evolutionary IS development can require more resources than expected and can lead to "scapegoating" of the developers), lesson 18 (the analyst's interpretation of IS success may differ from the other stakeholder groups) and lesson 19 (noise drowns good ideas).

The implications of these results are discussed in the next chapter.

## 7.4 Limitations of the Research

The most evident limitation of this work is that the CarMaker studies were undertaken with limited knowledge of the CASE, IS development, IS assessment and IS research methodology literature. The author had a first degree and a masters degree in Computer Science and had worked for several years as a commercial programmer prior to becoming an academic. Therefore the principles of structured systems development and CASE were familiar, but not the holistic IS literature that called into question the scientific approach to systems development (for example, Checkland, 1981, Kling, 1987 and Avison and Fitzgerald, 1988). The author was not familiar with developments in IS research methodology (exemplified by the work of Nissen *et al.*, 1991, Galliers, 1991 and Wood-Harper, 1992), nor with either the "scientific" IS assessment/software metrics literature (for example Boehm, 1981, Gilb, 1988, Fenton, 1991 and AMI, 1992) or the questioning of the scientific assessment school (Bjorn-Andersen and Davis, 1988).

Given this lack of knowledge at the start, the research relied on the practitioner motivation of the Consultant and a perception by the Consultant and the author that CASE hype was soon to unleash a backlash against the tools which would force tool vendors to substantiate their claims. Furthermore, CarMaker were interested in exploring CASE potential, particularly if there was an added bonus of free consultancy and low-cost system development. The research was therefore well-grounded in practice if not in theory.

The lack of theory to guide data collection meant that enormous quantities of data were collected, much of which could have been discarded if a single theory were being tested. However, the research programme evolved from an initial narrow, technological focus into a broader organisational study as the importance of context became apparent. Fortunately, the vast quantity of documents and material collected enabled the author to revisit the earlier work and to construct a story that described the interweaving of technology and organisation over time, a more relevant story than one of either technology or organisation alone. With hindsight, a rich theory such as Walsham's (1993) structuration framework or Sauer's (1993) innovation model may have provided sufficient breadth to enable the full story to have been told whilst focusing and reducing data collection, but the author was not aware of these models at the time; indeed neither text had been published in 1990.

The subsequent review of the literature (Chapters 6 and 7) shows that the practitioner "gut feel" was accurate. Wynekoop and Conger (1991) show that there is a need for detailed action research and case studies that evaluate tool impact using a variety of data sources and Coupe (1994) laments the absence of IS user and manager views in the CASE literature and identifies a need to balance subjective IT staff perceptions with objective metrics. The CarMaker studies addressed all of these concerns.

Action research may be criticised because the work entails intervention by the researcher into the subject being studied (Avison and Wood-Harper, 1991). How can unbiased results be obtained from someone who is so closely involved in creating the work being studied? No one likes to admit that their efforts have been unsuccessful, and there is a temptation to omit or disguise problematical events. Furthermore, the researcher may avoid asking awkward questions when he or she may be implicated in the answer. In answer to these criticisms, it is hoped that the honesty of the research presented here is evident to the reader. Both success and failure have been described in sometimes painful detail; I have tried to avoid hiding any important incidents. Action research is research in "technicolour"; the researcher can be exposed to a tremendous range of emotions as work progresses and it is important not to allow these emotions to influence the conduct of the research. This only occurred once during the four years of work described here. I had intended to conduct a user satisfaction survey of the Recycling project around Christmas 1992. Two weeks before Christmas the project erupted and the database became the subject of acrimonious and continual criticism for several weeks. This was a very unpleasant experience and I decided not to give the database critics more ammunition, so postponed the survey until the hostilities had subsided. The survey was therefore not initiated until March, and an opportunity to record user views at the height of the battle was lost.

Critics of action research also question the generalisability of results (Avison and Wood-Harper, 1991). The same criticism can be levelled at any small-sample research including case studies and laboratory experiments. This criticism misses the central purpose of action research, which is to create opportunities for detailed, first-hand research in a realistic context in order to explore new ideas and technologies, such as CASE. There have been numerous surveys of CASE practice in recent years, all using different instruments, all asking different questions and many returning very small samples (for example, Stobart *et al.*, 1991). Furthermore, the surveys would appear not to be asking a number of important questions, for example whether the IS context is properly addressed; whether there are differences of opinion amongst stakeholders as to the efficacy of the tools; whether the true potential of IT to simulate and animate is being exploited (or whether the tools are simply glorified drawing packages); whether the generated systems are sufficiently usable, flexible and maintainable; whether the tools simply perpetuate the status quo and fail to address the culture gap between users and developers (Grindley, 1991). None of these questions have been asked by the CASE surveys reviewed to date. This would indicate an urgent need for more detailed examination of the phenomena under study before further surveys are attempted. In conclusion, the preceding discussion shows that, when compared to the reality of generalisable research, the potential contribution of action research and grounded theory is clearly significant.

One contribution that this work has made is to make use of both qualitative and quantitative measures of CASE impact. Unfortunately, the only project to include a full set of quantitative metrics was the New Model project. The function point analysis, defect density analysis and productivity analysis could not be carried out for the Warranty project because an IS was not developed, nor for the Recycling project because the evolutionary nature of development plus the limited use of data flow diagrams made function point analysis difficult. Nonetheless, the author's subjective views plus those of the other stakeholders enabled a comprehensive assessment of both projects to be made (see Section 7.3).

The development of the Contribution Matrix (Section 7.3.3) as a technique for concisely identifying where a contribution has been made greatly aided the author in organising his thoughts. However, it is difficult to map from the richness of a publication to the limited scope of an individual lesson. Clearly the mapping is subjective, and another researcher may map differently in some instances. But this happens in any literature review, and the Contribution Matrix offers the advantage of clearly illustrating the lessons that have been inadequately addressed in the literature.

The literature reviewed here has been accumulated over the course of the study. It does not represent a focused search (cf. Wynekoop and Conger, 1991), but is, instead, an eclectic selection of work that the author assesses to be important. The range of CASE literature reviewed is nonetheless large (80 publications compared to Wynekoop and Conger's 40), and up to date (Wynekoop and Conger reviewed the limited period of 1988 and 1989). The British Computer Society CASE conference papers (Spurr and Layzell, 1990 and 1992) have not been reviewed in the same detail as the rest of the CASE literature and were added to balance the evaluative and empirical nature of the author's initial selection. Furthermore, the IS development and IS assessment literatures have not been as extensively reviewed as the CASE literature since, given the wealth of literature on CASE alone, it was necessary to constrain the survey somehow. The author acknowledges that a number of important works on IS development have not been included in the review, particulary "normative writings" such as Davis (1982), Brooks (1987), Lyytinen and Hirschheim (1987), Hirshheim and Klein (1989) and Humphrey (1989), but believes that the work that has been reviewed (for example, Kling, 1987, Avison and Fitzgerald, 1988, Avison and Wood-Harper, 1990, Newman and Robey, 1992, Orlikowski, 1992 and 1993, Sauer, 1993 and Walsham, 1993) has provided ample support for the CarMaker results and a source of inspiration for further work.

The research could be criticised for focusing on small-scale system development being undertaken by relatively inexperienced developers. Would the same issues arise if the study had been undertaken in a large system development project and by experienced developers? Land *et al.* (1992) address this question when looking back at four large organisations' mixed fortunes in IPSE implementation:

> *"In some ways, the results of this study may be regarded as obvious or to conform to common sense. Would not any sensible project manager take account of all the factors discussed above and avoid the mistakes revealed b) the study? ... It is more likely that the adverse factors are obvious only in hindsight ... "* (p. 80)

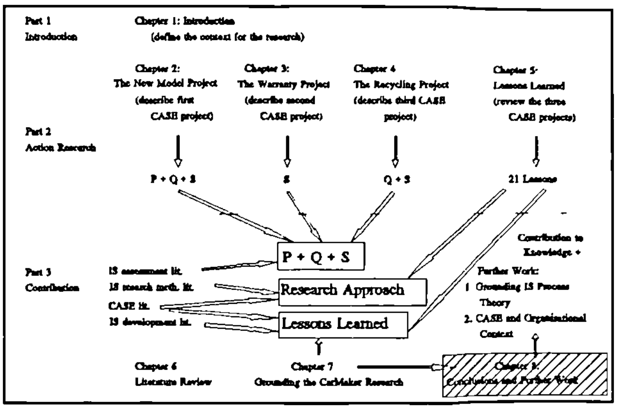
The high rate of IS failure cited by Lyytinen and Hirschheim (1987) in their review of the IS literature and Sauer's (1993) fatalistic look into the future show that problems are likely to bedevil all sizes of projects and all levels of experience for some time to come. In fact, size is a major productivity and quality driver, large systems cost more and are more prone to failure! (Boehm, 1981 and Cusumano and Kemerer, 1990).

## Conclusion to Chapter 7

The discussion of limitations concludes Chapter 7. All of the evidence from the CarMaker research has now been presented and grounded in the appropriate literature. Much has been learned by the author and a number of contributions to CASE knowledge have been identified. Chapter 8 summarises these contributions and looks forward to further CASE research.

# Chapter 8

# Conclusions and Further Work



## 8.1 Introduction

In this final chapter the "grounded practice" from Chapter 7 is summarised and presented as a set of contributions to CASE and information systems knowledge. The research described here has led to a number of findings that will stimulate further work. Two proposals for further research are presented in the second part of this chapter; the first addresses the need for better theory to support CASE and IS development practice, the second proposes a new direction for CASE technology development that addresses the need for stakeholders to visualise and understand better the contextual issues surrounding IS development.

# 8.2 Conclusions

## 8.2.1 Overview

The research reported here represents a significant contribution to CASE knowledge and a detailed first-hand account of how systems development happens in practice and how CASE tools contribute to the process. The author has learned a great deal about CASE technology over the four year period of the research, but more importantly, the exposure to organisational problem-solving has shown that a narrow focus on technology is insufficient to guarantee systems development success. Wynekoop and Conger called for CASE *action research* in 1991, following their review of the CASE literature that showed that research to date had focused almost entirely on tool building and normative writings; typically either practitioners rationalising on CASE experience after the event or academics reviewing the literature or proposing new approaches. There were few case studies, and most of those used obscure tools or techniques. Orlikowski's (1988 and 1990) work stood out as the only example of rigorous and relevant case study research at that time. There was little evaluative work, raising the concern that numerous tools were being built with little understanding of how effective the tools were in practice; in other words, the CASE community was not *learning* from experience of use. Furthermore, there had been no action research. Clearly the stage was set for a programme of in-depth research that used mainstream tools and techniques and that evaluated tool impact using mainstream assessment methods. The research described here addresses this need.

At about the same time as Wynekoop and Conger were undertaking their literature review, the author and the Consultant were formulating a tool evaluation programme at the Research Centre. We did not know of Wynekoop and Conger's work (indeed, it had yet to be published), but we were aware of a backlash forming against CASE "hype" and therefore of the need to justify tool purchase by providing objective evidence of benefits. The Consultant's initial proposal to the Research Centre represented a timely window of opportunity, enabling academics and practitioners to work together to address these issues. The fact that the research was initiated by a tool vendor's marketing concerns rather than by an academic's literature review is unimportant now. The key point is that the work was motivated by a genuine research problem that was of concern to both practitioner and academic communities.

There was no theory to guide data collection. Instead, we were guided by "common sense" which assumed that data on effort, stakeholder participation, activity descriptions and problems/thoughts would form a rich resource for subsequent analysis. So it has proved. As the author's awareness of software metrics developed, more sophisticated data was collected by way of function point analysis and user satisfaction surveys. This helped to ground data collection in mainstream techniques that are becoming increasingly popular in practice. Indeed, the software metrics literature is immature, as is data collection in practice. In comparison to much of the CASE impact data reported in the literature (for example Norman and Nunamaker, 1989, Burkhard, 1990, Wijers and van Dort, 1990, Stobart *et al.*, 1991, Aaen *et al.*, 1992 and Glasson *et al.*, 1992), our metrics are extensive and rigorous. The meticulous collection of project documents and the extensive recording of thoughts and issues in the project diaries has enabled the focus of the research to change over the four years, from an initial emphasis on technology and metrics and describing the tool features that were useful and those that were not, to a far broader analysis of the organisational context in which the technology was being used. A more constrained data collection approach or the rigid following of a pre-defined theory may well have denied us the opportunity to change direction in order to tell the more important story.

What emerges now is a set of very impressive productivity and quality metrics which show that lower CASE tools enable systems to be programmed quickly with few defects in comparison with the studies reported in the literature. Of course there are a number of concerns regarding metrics collection, interpretation and conversion, since no two studies used the same productivity or quality metrics! Even so, a strong case could be made that CASE tools represent a very worthwhile investment if we were to take the metrics at face value. A tool vendor would no doubt happily use the metrics presented in this thesis as evidence of tool success. However, a strength of the hybrid research approach adopted is that we can go behind the figures to explore multiple perspectives and to look at the events that occurred and account for differing points of view. On closer inspection it can be seen that the stakeholders (managers and end users) had less sanguine views of what had taken place. Certainly benefits accrued from all three projects, but none of the projects left a long-term legacy in terms of information systems delivered. Furthermore, the author was left feeling frustrated and weary at the inadequacies of the tools and techniques to support him in his endeavour to improve organisational effectiveness.

The message that emerges from these mixed opinions and contradictory metrics is that systems development is not a discrete activity, taking place in isolation from the organisational milieu that surrounds it. Whilst we can measure our success in terms of narrow, technically-focused metrics and proclaim our achievements in sales brochures and the press, the reality is that these metrics only tell a small part of the story, and are not very helpful in identifying or encouraging good practice. Ultimately, it is the relationship between the system and its host organisation that determines whether the system will be useful. Therefore, the system developers have, in some way, to become part of that organisation; they cannot operate in isolation, unaware of developments elsewhere that will impact their work. All three CarMaker studies suffered from a weak integration of developers and hosts. The most telling criticism of the CASE tools and techniques used is that they did not help the developers to bridge that gap; in fact, in many ways the tools encouraged the developers to focus on the technology and to avoid confronting the bigger organisational issues.

The importance of this message for the CASE community cannot be overstated. The literature reviewed in Chapters 6 and 7 shows a clear divide between CASE knowledge and IS development knowledge. Whilst the CASE literature has matured a little since Wynekoop and Conger's review in 1991, notably to increase the use of survey research and evaluative studies, there remains an almost complete vacuum in terms of recognising that CASE initiatives must be embedded within the larger socio-political organisational picture (Orlikowski, 1990 and 1993 and Land *et al.*, 1992 are notable exceptions). In contrast, the IS development community is grappling with the "scientific" versus "systemic" division (Avison and Fitzgerald, 1988). Whilst the scientific paradigm still holds sway, there is an active movement to argue the case for more holistic thinking about information technology and organisations (for example Kling, 1987, Avison and Wood-Harper, 1990, Orlikowski, 1992, Sauer, 1993 and Walsham, 1993). Kling's (1987) description of the discrete-entity model is uncomfortably close to the dominant mindset of the stakeholders in the CarMaker studies. Ironically, the author can thank the struggles described in this thesis for shifting his own mindset away from discrete-entity thinking and towards web thinking; working in an environment that is "a-historical, a-contextual, and assumes adequate resourcing" (Kling, 1987, p. 307) teaches some hard lessons. As Kling and Iacono (1989) observe, it is very difficult to institutionalize a CBIS if the social context is ignored.

### Discrete-entity thinking: some CarMaker quotes

To emphasise this point further, the most senior CarMaker manager in each project inadvertently provided a quote that seems to summarise their view of the CASE-based development work and illustrates the prevalence of discrete-entity thinking encountered in the research:

The Sponsor (New Model project, from interview transcript, Appendix 2.3):

> *The Sponsor viewed the system as "suffering from its own success" in that as more areas began to use the system, there was demand for a greater range of reports than had originally been envisaged.*

The Warranty Director (Warranty project, from interview transcript, Appendix 3.2)

> *"Again on a personal note, I am more inclined to support an 'Information Development Project' than a 'Systems Development Project'. Words, I know, but it felt like the latter."*

Terry (Recycling project, from interview transcript, Appendix 4.3)

> *He completed the questionnaire by stating that the IS "... seems to have been the single largest problem on the project".*

### 8.2.2 Contribution to Knowledge

The major contribution of this thesis is to present a detailed argument for a change in CASE thinking away from the discrete-entity *weltanschauung* towards a web-like *weltanschauung*. Kling's (1987) terms have been used here, but clearly the work of any of the holistic writers described in this thesis could form a starting point for dramatically improving the contribution of CASE to organisational well-being. For instance, Orlikowski (1992) and Walsham (1993) utilise structuration theory which emphasises social context and social process and Sauer (1993) provides a very broad-based definition of context that has similarities to Kling's web model. A programme of future research that addresses this goal is outlined in the next section.

In addition to this major contribution a number of further contributions have been made by this work. The contributions have been divided into *process* contributions which emphasise the strengths of the research process and *product* contributions which are the major results of the research.

Process Contributions

1. A rare CASE action research study has been presented (Gavin and Little, 1994 is the only other CASE action research study identified by the author)[16]. This helps to counterbalance the continued domination of tool building, normative writing and, more recently, surveys based on IT staff "perception" in the CASE literature.

2. The views of IT specialists, users and managers have been presented. In comparison, the CASE literature is dominated by IT specialists views, an issue also highlighted by Coupe (1994).

3. The study represents a rare example of quantitative measurement of both productivity and quality. The literature survey showed that few studies have attempted to quantify CASE impact, and those that have have generally concentrated on productivity impact and not investigated quality.

[16] Whilst recognising that both studies could be criticised as not being pure action research since neither study made use of theory to guide data collection.

4. Mainstream tools and techniques (for example DFDs, ERMs, repositories and prototyping) and mainstream performance metrics (defect analysis, function point analysis and a UIS survey) have been used to ensure that the work is relevant to a wide audience (cf. Wynekoop and Conger's criticism of the obscure case study research presented in the CASE literature).

5. The research programme spanned a long period of time (four years), enabling the full consequences of CASE-based IS development to be recorded and the full systems life-cycle to be observed from analysis, through design. programming, implementation, support and maintenance, to final abandonment in the case of the New Model project. In contrast, many studies cover a shorter period of time and are therefore denied the opportunity of seeing the story unfold. For example, Gavin and Little (1994) focus on the analysis stage, Boland and Day (1989) describe one year of IS development, Orlikowski (1988 and 1990) describes eight months, Land *et al.* (1992) six months, Mouakket *et al.* (1994) four months. Certainly some of these studies also investigated historical data, but the *observation* of events was constrained to the periods specified above.

6. The research programme covered three studies, this is a larger group than many of the studies identified in the case study/action research literature. Whilst the purpose of the work is not to generalise the results, the common issues which were evident in all three studies increase confidence in the relevance of the results.

7. There was a high degree of continuity between the three studies: the same upper CASE tool was used throughout, lower CASE tools based on relational databases were used throughout, the lead developer (the author) was the same in all three projects, the host organisation, CarMaker, remained constant, the format of the productivity and quality data collected was the same for all three projects (apart from the move away from structured interviews to a UIS survey for the third project). The high level of continuity adds to the comparability of results.

Product Contributions

8. A highly-detailed account has been presented of the process of systems development and tool use as it evolves over time, helpmg to explain why actions were taken and decisions made and "how analysts analyse" (Wynekoop and Conger, 1991, p. 314). Action research is immediate, events can be captured the moment they occur, not via an interview with a third party some time after the event when the importance of the event is either forgotten or post-rationalised. A danger with action research is that the author may disguise or omit problematical events if they bring into question the author's competence in the practice of systems development. It is hoped that the honesty of the stories is evidenced by the combination of success and failure described in this thesis.

9. The work addresses both technology (the "small" font sections of the "stories") and organisation (the "normal" font sections) thereby explaining why certain tool features were useful and the impact of tool limitations.

10. The results represent a triangulation (cf. Orlikowski, 1993) of the author's subjective account (the stories and lessons); other stakeholder views (interviews and UIS survey)[17]; quantified metrics (productivity and defects); and the CASE and IS literature results. This showed that the author's subjective views were very much in accord with the holistic IS development literature; identified the potentially misleading nature of "objective" metrics; highlighted the discrete-entity thinking of the stakeholders and showed clearly that there is a significant gap in CASE knowledge pertaining to more holistic thinking and research. Furthermore, a wide variety of documents were consulted in constructing the stories, including minutes of meetings, memos, whiteboard printouts, progress reports, timesheets, the project diary, print-outs from the ISs, DFDs, ERMs and the IS software itself.

11. The work represents a rigorous investigation which grounds practice (the CarMaker studies) in theory by examining a very large number of CASE publications (80 in total), and a significant number of IS development publications. In contrast, Wynekoop and Conger (1991) examined 40 CASE publications presented over a two year period (1988 and 1989). The CASE literature has expanded greatly since Wynekoop and Conger's survey,

[17] Stakeholders were sent copies of the interview transcripts and asked to amend them if they so wished. Several amendments were received and incorporated. The transcripts have therefore been endorsed by the stakeholders.

therefore the update to their taxonomy presented in Section 7.2 represents a significant contribution to the codification of CASE knowledge. Each of the three elements of the CarMaker research: the research approach, the productivity and quality results and the lessons learned were separately grounded in the appropriate literature in Chapter 7 (sections 7.2, 7.3.1 and 7.3.3 respectively). Therefore the work is well-grounded and can be seen to build upon existing knowledge.

12. The work has highlighted the immature and contradictory state of the software metrics literature. Virtually all of the studies that included metrics used different units of measure; there was no agreement for example on how many hours constitute a work-month; and Dreger's (1989) "large" IS of 800 to 1000 function points was much smaller than the mean sizes for the US and Japanese systems studied by Cusumano and Kemerer (1990) and the applications generated by the Application Software Factory (Swanson *et al.*, 1991).

13. The "Contribution Matrix" contributes to IS research methodology by consolidating a diverse set of publications into a single concise table that enables the contribution of a particular piece of work (the lessons learned in this case) to be readily ascertained. The technique may be criticised for eliminating richness (cf. Avison and Fitzgerald's, 1988 decision not to use a table to compare ISD methodologies), but when used in conjunction with an extensive narrative, such as the lessons learned rubric in Chapter 5, can be a useful presentation device.

14. One abiding memory of the stress of systems support in an under-resourced environment is the realisation that projects tend to be very pleasant and relaxed until the system goes live; then the pressure mounts. It is worth noting that many of the "political" writers, for example Markus (1983), Kling (1987), Newman and Robey (1992), Orlikowski (1993), Sauer (1993) and Walsham (1993) describe post-implementation activities, whereas many of the CASE success stories focus on the pre-implementation stage (Gavin and Little, 1994 for example). In other words, talking about changing peoples' work is relatively painless, actually doing something about it can hurt.

15. Following on this theme, the "scientific" systems analysis approach adopted in the CarMaker studies did not address *context* (Avison and Fitzgerald, 1988). This failure caused no pain until the systems went live, when the developers became scapegoats and the users were presented with sub-optimal solutions to their problems. The scientific approach can be costly.

16. Further evidence of the importance of ontological debate concerning the nature of reality (Avison and Fitzgerald, 1988), particularly the form of data which is to be stored in the IS, is contributed by the wrangle over whether the database should be "part-oriented" or "process-oriented" in the Recycling project. This argument lay at the heart of the difficulties the developers had in creating a stable data model during the first half of the project. Terry's frustration at the stalemate became evident when he terminated one heated meeting by stating that it was obvious, all we had to do was store data on parts and processes!

The list of contributions concludes Section 8.2; proposals for further work follow.

The last word is reserved for the Epilogue.

## 8.3 Further Work

Two areas of further work arise from this study. The first area revisits the CarMaker studies and addresses the question "Can the CarMaker studies contribute to the development of IS theory?", specifically, can the three studies be used to ground some of the context-rich process models of IS development such as those proposed by Kling (1987), Newman and Robey (1992), Orlikowski (1992), Walsham (1993) and Sauer (1993). And can these models be used prescriptively as well as descriptively? If so, their value to practitioners will be greatly enhanced.

The second area is an entirely new programme of research that addresses the question "Can recent developments in information technology, such as multimedia, enable CASE tools to be developed that encourage stakeholders to address contextual issues?" These two areas are discussed in more detail in the following sections.

### 8.3.1 Grounding IS Process Theory

The need for both researchers and practitioners to have a better understanding of the key elements and relationships within the IS environment has been amply demonstrated by the CarMaker studies. It has been shown that the lessons learned from these studies echo the thoughts of a number of researchers writing in the IS development literature. Some of these researchers are developing process models that attempt to explain the causes of IS success and failure and, in so doing, aim to improve practice by offering practitioners an alternative conceptual model to the discrete-entity models which are currently predominant. In this section Sauer's (1993) innovation model is discussed briefly to indicate the kind of work proposed.

### The Innovation Model

Sauer (1993) views the IS process as a process of innovation in which the project organisation attempts to solve a "problem" for its supporters. Figure 8.1 represents an amalgamation of Sauer's ideas.

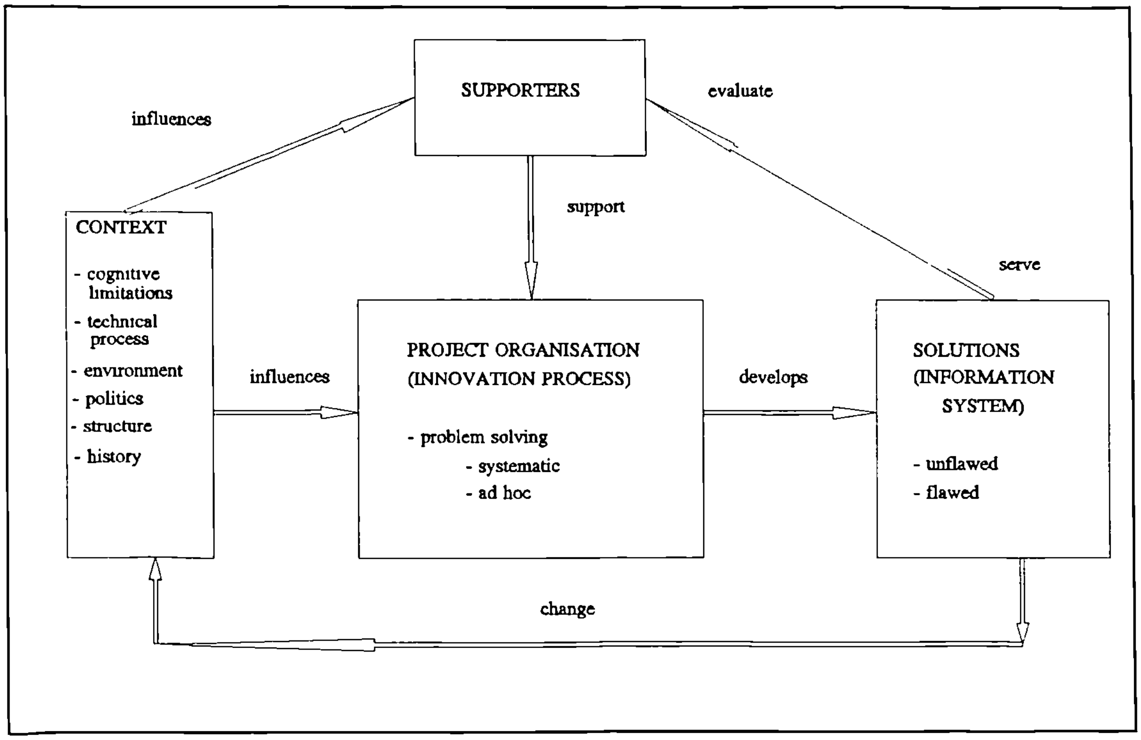


**Figure 8.1** The IS Innovation Process (adapted from Sauer, 1993, p. 71)

The solution to the problem is an information system that hopefully will serve the supporters' needs. The model is a web model (Kling, 1987) in that system development is not seen as a straightforward rational process in which all the factors are known and the supporters' requirements clearly defined and agreed. Instead, the process takes place within a web of stakeholders, technology and context which are all subject to change as time goes by. The project organisation applies problem solving approaches that will address part of the problem context. Since the context is extremely rich, it is unlikely that any problem solving approach will ever fully capture the contextual influences. Therefore, all problem solving, by definition is flawed. This means that the product of the problem solving, the information system, will also be flawed. Whether the flaws are significant or not will depend on the evaluation of the system by the supporters. It is clearly conceivable that a system with bugs will be tolerated as long as the bugs do not conflict with the objectives of the key supporters. Similarly, a bug-free system that satisfies less influential supporters but damages the interests of powerful supporters will lead a precarious existence. In his conclusions Sauer identifies a number of areas warranting further work based on this model:

| Areas Warranting Further Research | Practices Warranting Development |
|---|---|
| Innovation process | Practice of politics |
| Relation of IS to supporters | Support management |
| Flow of support | Management of evaluation processes |
| Management of support | Project abandonment |
| Use of problem-solving mechanisms | Post-project reviews |
| Evaluation processes | |
| Project organisation power | |
| Historical context of IS in host organisation | |

**Table 8.1** Areas for Further Research and Development

(after Sauer, 1993, p. 330)

A major contribution of Sauer's work has been to investigate why information systems fail from an organisational perspective; the result is a realistic account of IS practice supported by new and insightful theory. The first proposal for further work following on from the CarMaker CASE studies has the same aim, to build upon a realistic, context-rich account of CASE practice and emerging process theories of IS development, such as Sauer's. Reviewing Sauer's suggestions for further work, it can be seen that a better understanding of organisational context is key, particularly where context influences supporters. The same message emerged from the CarMaker studies. Sauer's most controversial suggestion is that if sufficient support is not forthcoming, the project organisation should propose that the project be abandoned. A worthy objective for IS theory research would be to provide stakeholders with sufficiently rich conceptual models so that they can appreciate the need for abandonment sooner rather than later; or, more optimistically, so that they can identify project risks and prescribe remedial action before it's too late.

### 8.3.2 CASE and Organisational Context

The tools used in the CarMaker studies had a major limitation, they supported analytical techniques, such as dataflow diagramming and entity-relationship modelling which had no mechanism for representing the socio-political organisational context in which development was taking place. The tensions evident between the different organisations; CarMaker, the Research Centre, ToolVendor and, in the third study, Recyclate Ltd., were rarely acknowledged and never adequately resolved. The desperate lack of resources for IS support and maintenance which made the New Model and Recycling projects particulary wearying for the developers was not predicted nor remedied; hidden agendas and substitute systems initiatives remained under cover but influenced stakeholder behaviour in a way that was difficult for the developers to counter. With reference to this issue Checkland (1981) concedes that politics will always move one step beyond analysis, and that a hidden agenda can remain hidden despite attempts to expose it but this should not stop the development of alternative approaches to analysis since the inadequacy of the "scientific" approach clearly calls for a new way of thinking.

Checkland's solution has been to develop the Soft Systems Methodology (SSM) and Avison and Wood-Harper (1990) have integrated some of the SSM techniques into a contingent IS development methodology called *Multiview*. One of the central techniques is rich picture diagramming, where a "cartoon" of the problem situation is drawn with contributions from the stakeholders. The rich picture depicts tangible objects such as buildings, machinery and people, as well as intangibles, such as hopes, fears and frustrations. Currently rich pictures are static, there is no way of bringing them to life or animating them. This is unfortunate, because many of the contextual issues that influence a project are best explored by modelling the *dynamics* of human interaction. This is where emerging multimedia technology offers considerable potential. If videos of the problem situation could be recorded and combined with the static rich picture symbols, the vividness of real experience could be preserved. Furthermore, a fully integrated multimedia CASE tool could combine video, rich pictures, and conventional simulation software to enable the context to be explored via a number of complimentary techniques. The role of IT is central here in animating and simulating the situation, enabling the full potential of CASE tools to be realised instead of the limited role currently played by the glorified drawing packages.

Admittedly, the downstream integration issues that currently bedevil CASE technology may well remain, but a multimedia upper CASE tool would at least contribute to an informed analysis of context prior to code generation or package selection. This work has much in common with the work of Avison *et al.* (1992) and their *Get Rich Quick!* Soft Systems Methodology tool. Discussions with a vendor of a multimedia tool and suppliers of simulation tools are currently taking place to provide a foundation for the proposed research. Furthermore, this work links with the first proposal for further work in that IS process models such as Sauer's model could be used to provide a conceptual framework for action research using the multimedia tool. Finally, the combination of theory and technology which addresses organisational context promises to produce a powerful approach for exploring strategic initiatives, such as Business Process Re-engineering (Hammer and Champy, 1993), as well as smaller-scale IS developments.

# *Epilogue*

Two quotations were presented in the Prologue to this thesis ... the first recognised that IS development is highly problematical; the second regretted the failure of theory to inform practice.

Having spent four years wrestling with the problems of information systems development using practitioner methods I can't help but think that perhaps the two quotations are related ...

# *Bibliography*

The thesis has been prepared in accordance with the guidelines specified in the University of Warwick Graduate School publication *Guide to Examinations for Higher Degrees by Research (September 1993 Revision).*

## 1. Full Bibliography in Author Name Order

Aaen I. 1993. CASE User Satisfaction - Impact Evaluations in User Organizations, unpublished.

Aaen I. 1994. CASE as a Maturity Lever - Institutionalization via CASE, unpublished.

Aaen I., Siltanen A., Sorensen C. and Tahvanainen V-P. 1992. A Tale of Two Countries: CASE Experiences and Expectations, in *The Impact of Computer Technologies on Information Systems Development*, K.E.Kendall, K.Lyytinen and J. DeGross (eds.), IFIP Transactions, North-Holland.

Aaen I. and Sorensen C. 1991. A CASE of Great Expectations, *Scandinavian Journal of Information Systems*, 3(1):3-23.

Alexander J. and Hicks K. 1992. ISTEL Applications Architecture, in K. Spurr and P. Layzell (1992), *op cit.*

AMI. 1992. *The AMI Handbook*, Centre for Systems and Software Engineering, South Bank University.

Avison D.E. and Fitzgerald G. 1988. *Information Systems Development Methodologies, Techniques and Tools*, Blackwell Scientific, Oxford.

Avison D.E. and Wood-Harper A.T. 1990. *Multiview: An Exploration in Information Systems Development*, Blackwell Scientific, Oxford.

Avison D.E. and Wood-Harper A.T. 1991. Information Systems Development Research: An Exploration of Ideas in Practice, *The Computer Journal*, 34(2):98-112.

Avison D.E., Golder P.A. and Shah H.U. 1992. Towards an SSM Toolkit: Rich Picture Diagramming, *European Journal of Information Systems*, 1(6):397-407.

Baba I. 1990. A Standard for SSADM CASE Support, in K.Spurr and P.Layzell (1990), *op cit.*

Bailey J.E. and Pearson S.W. 1983. Development of a Tool for Measuring and Analyzing Computer User Satisfaction, *Management Science*, 29(5):530-545.

Banker R.D., Datar S.M. and Kemerer C.F. 1991. A Model to Evaluate Variables Impacting the Productivity of Software Maintenance Projects, *Management Science*, 37(1):1-18.

Banker R.D. and Kauffman R.J. 1991. Reuse and Productivity in Integrated Computer-Aided Software Engineering: An Empirical Study, *MIS Quarterly* September:375-401.

Barki H. and Hartwick J. 1994. Measuring User Participation, User Involvement. and User Attitude, *MIS Quarterly*, March:59-79.

Barn B.S. 1992. User Interface Development: Our Experience with HP Interface Architect, in K. Spurr and P. Layzell (1992), *op cit.*

Baroudi J.J. and Orlikowski W.J. 1988. A Short-Form Measure of User Information Satisfaction: A Psychometric Evaluation and Notes on Use, *Journal of Management Information Systems*, 4(4):44-59.

Beeby M., Parkinson J.S. and Hamilton V. 1992. Integrated Support for Real-Time Control Systems, in K. Spurr and P. Layzell (1992), *op cit.*

Benbasat I., Goldstein D.K. and Mead M. 1987. The Case Research Strategy in Studies of Information Systems, *MIS Quarterly*, September:368-386.

Bendure C.O. 1991. A Case Study on CASE, *Information Systems Management*, Fall 1991:50-56.

Bjorn-Andersen N. and Davis G.B. (eds). 1988. *Information Systems Assessment: Issues and Challenges*, Proceedings of the IFIP WG 8.2 Working Conference on IS Assessment, Noordwijkerhout, The Netherlands, 27-29 August, 1986, North-Holland, Amsterdam.

Boehm B.W. 1981. *Software Engineering Economics*, Prentice Hall, Englewood Cliffs, New Jersey.

Boehm B.W. 1988. A Spiral Model of Software Development and Enhancement, *IEEE Computer*, May:61-72.

Boland R.J. and Day W.F. 1989. The Experience of System Design: A Hermeneutic of Organizational Action, *Scandinavian Journal of Management*, 5(2):87-104.

Boland R.J. and Hirschheim R.A. 1987. *Critical Issues in Information Systems Research*, Wiley, Chichester.

Brooks F.P. 1987. No Silver Bullet: Essence and Accidents of Software Engineering, *IEEE Computer*, April:10-19.

Burkhard D.L. 1990. Examination of Factors Contributing to the Success of the Implementation of Computer-Aided Software Engineering Technology, in *Proceedings of the 23rd Annual Hawaii International Conference on System Sciences Vol IV* (R.H. Sprague, Jr. (ed.)), IEEE.

Budgen D. and Marashi M. 1990. Knowledge Use in Software Design, in K. Spurr and P. Layzell (1990), *op cit.*

Checkland P. 1981. *Systems Thinking, Systems Practice*, Wiley, Chichester.

Chikofsky E.J., Martin D.E. and Chang H. 1992. Assessing the State of Tools Assessment, *IEEE Software*, May:18-20.

Coad P. and Yourdon E. 1991. *Object-Oriented Analysis*, Prentice-Hall, Englewood Cliffs, New Jersey.

Coupe R.T. 1994. A Critique of the Methods for Measuring the Impact of CASE Software, *European Journal of Information Systems*, 3(1):28-36.

Cotterman W.W. and Senn J.A. (eds). 1992. *Challenges and Strategies for Research in Systems Development*, Wiley, Chichester.

Cusumano M.A. and Kemerer C.F. 1990. A Quantitative Analysis of U.S. and Japanese Practice and Performance in Software Development, *Management Science*, 36(11):1384-1406.

Davis G.B. 1982. Strategies for Information Requirements Definition, in R.D. Galliers (1987), *op cit*.

Davy C. 1990. Using CASE to Control a Large Data Analysis Project, in K. Spurr and P. Layzell (1990), *op cit*.

Dimbleby J. and Lo B. 1992. Tools and Techniques used in Information Systems Development: A Measure of Maturity, in *Proceedings of 3rd Australian Conference on Information Systems*, University of Wollongong, 5-8 October 1992.

Doke E.R. 1989. Application Development Productivity Tools: A Survey of Importance and Popularity, *Journal of Information Systems Management*, Winter:37-42.

Dreger J.B. 1989. *Function Point Analysis*, Prentice-Hall, Englewood Cliffs, New Jersey.

Eisenhardt K.M. 1989. *Building Theories from Case Study Research*, Academy of Management Review, 14(4):532-550.

Fairbairn D. 1990. The Opportunity of the Decade, in K. Spurr and P. Layzell (1990), *op cit*.

Fenton N.E. 1991. *Software Metrics: A Rigorous Approach*, Chapman & Hall, London.

Galliers R.D. 1985. In Search of a Paradigm for Information Systems Research, in *Research Methods in Information Systems*, E. Mumford, R. Hirshheim, G. Fitzgerald and A.T. Wood-Harper (eds.), Proceedings of the IFIP WG 8.2 Colloquium, Manchester Business School, 1-3 September 1984.

Galliers R.D. 1985. An Approach to Information Needs Analysis, in R.D. Galliers (1987), *op cit.*

Galliers R.D. (ed.) 1987. *Information Analysis: Selected Readings*, Addison-Wesley, Sydney.

Galliers R.D. 1991. Choosing Appropriate Information Systems Research Approaches: A Revised Taxonomy, in H.-E. Nissen, H.K. Klein and R. Hirschheim (1991), *op cit.*

Galliers R.D. and Land F.F. 1987. Choosing Appropriate Information Systems Research Methodologies, *Communications of the ACM*, 30(11):900-902.

Gavin C.J. and Little D. 1994. Application of CASE Within Manufacturing Industry, *Software Engineering Journal*, 9(4):140-152.

Gee D.M., Worrall B.P. and Henderson W.D. 1992. Real-time CASE Tools: A Review of Current Tools and Future Prospects, in K.Spurr and P.Layzell (1992), *op cit.*

Gilb T. 1988. *Principles of Software Engineering Management*, Addison Wesley, Wokingham.

Glasson B., Quaddus M. and Pneumatikos E. 1992. CASE: in Search of a Consensus View, in *Proceedings of 3rd Australian Conference on Information Systems*, University of Wollongong, 5-8 October 1992.

Grindley K. 1991. *Managing IT at Board Level: The Hidden Agenda Exposed*, Pitman, London.

Gryczan G. and Kautz K. 1990. A Comparative Case Study of Prototyping Tools - Experiences and Conclusions, in *Proceedings of 1990 IEEE International Conference on Computer Systems and Software Engineering*, (8-10 May, Tel-Aviv, Israel).

Haine P. 1990. CASE Support for the Whole Life Cycle, in K. Spurr and P.Layzell (1990), *op cit.*

Haine P. 1992. Second Generation CASE: Can it be Justified?, in K.Spurr and P.Layzell (1992), *op cit.*

Hammer M. and Champy, J. 1993. *Reengineering the Corporation: A Manifesto for Business Revolution*, Nicholas Brealey.

Hayley K.J. and Lyman H.T. 1990. The Realities of CASE, *Journal of Information Systems Management*, Summer:18-23.

Hirshheim R. and Klein H.K. 1989. Four Paradigms of Information Systems Development, *Communications of the ACM*, 32(10):1199-1216.

Holloway S. and Bidgood T. 1991. *CASE Handbook for Information Managers*, Avebury Technical, Aldershot.

Howard G.S. and Rai A. 1993. Promise and Problems: CASE Usage in the US, *Journal of Information Technology*, vol 8:65-73.

Huff C.C. 1992. Elements of a Realistic CASE Tool Adoption Budget, *Communications of the ACM*, 35(4):45-54.

Huff C., Smith D., Stepien-Oakes K. and Morris E. 1992. *Proceedings of the CASE Adoption Workshop*, Technical Report CMU/SEI-91-TR-14, Software Engineering Institute, Carnegie-Mellon University, Pittsburgh, USA.

Humphrey W.S. 1989. *Managing the Software Process*, Addison-Wesley.

Ip S., Cheung L. C-Y. and Holden T. 1992. A Knowledge-Based Requirements Engineering Assistant, in K. Spurr and P. Layzell (1992), *op cit.*

ISO. 1986. *ISO-8402: Quality Vocabulary*, International Standardisation Institute.

Ives B., Olson M.H. and Baroudi J.J. 1983. The Measurement of User Information Satisfaction, *Communications of the ACM*, 26(10):785-793.

Jackson M. 1990. CASE Tools and Development Methods, in K.Spurr and P.Layzell (1990), *op cit.*

Jones C. 1989. Metric with Muscle, *Systems Development*, August:1-3.

Jones P. 1992. A Suitable CASE for Development, *Informatics*, November:62-63.

Jones M.C. and Arnett K.P. 1993. Current Practices in Management Information Systems, *Information & Management*, 24:61-69.

Kaplan B. and Duchon D. 1988. Combining Qualitative and Quantitative Methods in Information Systems Research: A Case Study, *MIS Quarterly*, December:570-586.

Katz R.L. 1990. Business/Enterprise Modeling, *IBM Systems Journal*, 29(4):509-525.

Keen P.G.W. 1991. Relevance and Rigour in Information Systems Research: Improving Quality, Confidence, Cohesion and Impact, in H.-E. Nissen, H.K. Klein and R. Hirschheim (1991), *op cit.*

Kemerer C.F. 1989. An Agenda for Research in the Managerial Evaluation of Computer-Aided Software Engineering (CASE) Tool Impacts, in *Proceedings of 22nd Annual Hawaii International Conference on Systems Sciences*, vol II (B.D. Shriver (ed.)), 3-6 January 1989.

Kemerer C.F. 1992. How the Learning Curve Affects CASE Tool Adoption, *IEEE Software*, May:23-28.

Kievit K. and Martin M. 1989. Systems Analysis Tools - Who's Using Them?, *Journal of Systems Management*, July:26-30.

King S.F. 1992. The Quality Gap: A Case Study in Information System Development Quality and Productivity using CASE Tools, in K. Spurr and P. Layzell (1992), *op cit.*

King S.F. and Galliers R.D. 1994. Modelling the CASE Process: Empirical Issues and Future Directions, *Information and Software Technology*, 36(10):587-596.

Kling R. 1987. Defining the Boundaries of Computing Across Complex Organizations, in R.J. Boland and R.A. Hirschheim (1987), *op cit.*

Kling R. and Iacono S. 1984. The Control of Information Systems Developments after Implementation, *Communications of the ACM*, 27(12):1218-1226.

Kling R. and Iacono S. 1989. The Institutional Character of Computerized Information Systems, *Office: Technology and People*, 5(1):7-28.

Land F.F., Le Quesne P.N. and Wijegunaratne I. 1992. Technology Transfer: Organizational Factors Affecting Implementation - Some Preliminary Findings, in W.W. Cotterman and J.A. Senn (1992), *op cit.*

Law D. and Naeem T. 1992. DESMET - Determining an Evaluation Methodology for Software Methods and Tools, in K.Spurr and P.Layzell (1992), *op cit.*

Lloyd-Williams M. and Benyon-Davies P. 1992. Knowledge Based CASE Tools for Database Design, in K.Spurr and P.Layzell (1992), *op cit.*

Low G.C. and Jeffery D.R. 1991. Software Development Productivity and Back-End CASE Tools, *Information and Software Technology*, 33(9):616-621.

Lyytinen K. 1990. *SYTI Project Summary*, Department of Computer Science, University of Jyvaskyla, Finland.

Lyytinen K. and Hirschheim R. 1987. Information Systems Failures - a Survey and Classification of the Empirical Literature, *Oxford Surveys in Information Technology*, vol 4:257-309.

Lyytinen K., Smolander K. and Tahvanainen V-P. 1989. Modelling CASE Environments in Systems Development, in *Proceedings of CASE89*, SISU, Stockholm, May.

Markus M.L. 1983. Power, Politics and MIS Implementation, *Communications of the ACM*, 26(6):430-444.

McChesney I.R. and Glass D. 1993. Post-Implementation Management of CASE Methodology, *European Journal of Information Systems*, 2(3):201-209.

McGinnes S. 1994. CASE Support for Collaborative Modelling: Re-engineering Conceptual Modelling Techniques to Exploit the Potential of CASE Tools, *Software Engineering Journal*, 9(4):183-189.

Milan J. and Munt C.E. 1990. The ETHOS Toolset and Methodology, in K. Spurr and P. Layzell (1990), *op cit.*

Mouakket S., Sillince J.A.A. and Fretwell-Downing F.A. 1994. Information Requirements Determination in the Software Industry: A Case Study, *European Journal of Information Systems*, 3(2):101-111.

Necco C.R., Gordon C.L. and Tsai N.W. 1987. Systems Analysis and Design: Current Practices, *MIS Quarterly*, December:460-476.

Newman M. and Noble F. 1990. User Involvement as an Interaction Process: A Case Study, *Information Systems Research*, 1(1):89-113.

Newman M. and Robey D. 1992. A Social Process Model of User-Analyst Relationships, *MIS Quarterly*, June:249-266.

Nissen H-E, Klein H.K. and Hirschheim R. (eds). 1991. *Information Systems Research: Contemporary Approaches & Emergent Traditions*, Proceedings of the IFIP TC8/WG 8.2 Working Conference on the Information Systems Research Arena of the 90's: Challenges, Perceptions, and Alternative Approaches, Copenhagen, Denmark, 14-16 December, 1990, North-Holland, Amsterdam.

Norman R.J. and Nunamaker J.F. 1989. CASE Productivity Perceptions of Software Engineering Professionals, *Communications of the ACM*, 32(9):1102-1108.

Olle T.W. 1992. A Comparative Review of the ISO IRDS the IBM Repository and the ECMA PCTE as a Vehicle for CASE Tools, in K.Spurr and P.Layzell (1992), *op cit.*

Orlikowski W.J. 1988. CASE Tools and the IS Workplace: Some Findings from Empirical Research, in *Proceedings of the 1988 ACM SIGCPR Conference on the Management of Information Systems Personnel*, p. 88-97.

Orlikowski W.J. 1990. Division Among the Ranks: The Social Implications of CASE Tools for Systems Developers, in *Proceedings of the 10th International Conference on Information Systems*, Boston, Ma., December 1989. (J.I. DeGross *et al.* (eds.))

Orlikowski W.J. 1992. The Duality of Technology: Rethinking the Concept of Technology in Organizations, *Organization Science*, 3(3):398-427.

Orlikowski W.J. 1993. CASE Tools as Organizational Change: Investigating Incremental and Radical Changes in Systems Development, *MIS Quarterly*, September:309-340.

Orlikowski W.J. and Baroudi J.J. 1991. Studying Information Technology in Organizations: Research Approaches and Assumptions, *Information Systems Research*, 2(1):1-28.

Parkinson J. 1990. Making CASE Work, in K.Spurr and P.Layzell (1990), *op cit.*

Piggott S.A., Hart M.M., Smith P.A. and Witcomb L.E. 1990. Using a CASE Tool to Develop Integrated Business Systems for Implementation Across a Distributed Hardware Network, in K. Spurr and P. Layzell (1990), *op cit.*

Price Waterhouse. 1989. *Information Technology Review 1989/90*, Price Waterhouse, London.

Prokit. 1991. Tried-and-True Prokit Falls Short, *Computerworld*, January:32.

QA Forum. 1989. *User Satisfaction Surveys*, Quality Assurance Forum, Middlesex.

Rego H.B. and Lima J.R. 1990. A Tool for Automating Facts Analysis, in K. Spurr and P. Layzell (1990), *op cit.*

Reynolds W. and Kalra C. 1992. The Business v. IS: A CASE of Breach of Promise, in K.Spurr and P.Layzell (1992), *op cit.*

Richards N.L. 1990. AD/Cycle - A New Official Standard?, in K.Spurr and P.Layzell (1990), *op cit.*

Robinson K. 1992. Putting the SE into CASE, in K.Spurr and P.Layzell (1992), *op cit.*

Sabanis N. and Darlison A. 1992. Reverse Engineering Data using an Integrated Toolset, in K.Spurr and P.Layzell (1992), *op cit.*

Sanders P. and Short K. 1992. Declarative Analysis in Information Engineering, in K.Spurr and P.Layzell (1992), *op cit.*

Sauer C. 1993. *Why Information Systems Fail: A Case Study Approach*, Alfred Waller, Henley-on-Thames.

Siltanen A. 1990. The Impact of CASE Tools on IS Management, in K.Spurr and P.Layzell (1990), *op cit.*

Smolander K., Marttiin P., Lyytinen K. and Tahvanainen V-P. 1990a. MetaEdit - A Flexible Graphical Environment for Methodology Modelling, in *Proceedings of CAiSE 91*, Springer-Verlag, Berlin.

Smolander K., Tahvanainen V-P and Lyytinen K. 1990b. How to Combine Tools and Methods in Practice - A Field Study, in *Advanced Information Systems Engineering*, Lecture Notes in Computer Science no. 436, Springer-Verlag, Berlin.

Software Management. 1992. *CASE: The Business Benefits*, A Software Management Special Report, Process Communications Ltd, London.

Spurr K. and Layzell P. (eds.) 1990. *CASE on Trial*, Wiley, Chichester.

Spurr K. and Layzell P. (eds.) 1992. *CASE - Current Practice, Future Prospects*, Wiley, Chichester.

Stapleton J. and Williams S. 1992. Experiences of Introducing a CASE Tool into Undergraduate Teaching of Systems Analysis and Design, in K.Spurr and P.Layzell (1992), *op cit.*

Statland N. 1989. Payoffs Down the Pike: A CASE Study, *Datamation*, April:32, 33 and 52.

Steer K. 1990. Methodologies and Tools for Parallel Processing, in K.Spurr and P.Layzell (1990), *op cit.*

Stobart S.C., Thompson J.B. and Smith P. 1991. Use, Problems, Benefits and Future Direction of Computer-Aided Software Engineering in the United Kingdom, *Information and Software Technology*, 33(9):629-636.

Strange G. 1990. International CASE Tool Selection and Evaluation, in K.Spurr and P.Layzell (1990), *op cit.*

Strauss A. and Corbin J. 1990. *Basics of Qualitative Research: Grounded Theory Procedures and Techniques*, Sage.

Swanson E.B. 1987. Information Systems: Necessary Foundations, in *Intellectual Foundations for Information Professionals*, H.K. Achleitner (ed.), Columbia University Press, New York.

Swanson K., McComb D., Smith J. and McCubbrey D. 1991. The Application Software Factory: Applying Total Quality Techniques to Systems Development, *MIS Quarterly*, December:566-579.

Symons C.R. 1991. *Software Sizing and Estimating: Mk II FPA*, Wiley, Chichester.

Thompson K. 1990. *The CASE Market: A Review*, The Institute of Software Engineering, Belfast.

Thompson K. 1993. CASE, Software Process Maturity and the Capability Maturity Model, IEE Colloquium on *CASE: Towards Software Process Maturity*, IEE, London, 22 November (digest no. 1993/222).

Walsham G. 1993. *Interpreting Information Systems in Organizations*, Wiley, Chichester.

Ward P.T. 1990. The Use of Current-Generation CASE Tools in Object-Oriented Analysis and Design, in K. Spurr and P. Layzell (1990), *op cit.*

Wijers G.M. and van Dort H.E. 1990. Experiences with the use of CASE-Tools in the Netherlands, in *Proceedings of CAiSE 90, the Second Nordic Conference in*

*Advanced Information Systems Engineering*, Springer-Verlag Lecture Notes in Computer Science, vol 436, Springer-Verlag, Berlin.

Wood-Harper A.T. 1992. Viewpoint: Action Research, *Journal of Information Systems*, 2:235-236.

Wynekoop J.L. and Conger S.A. 1991. A Review of Computer Aided Software Engineering Research Methods, in H-E Nissen, H.K. Klein and R. Hirschheim (1991), *op cit.*

Yellen R.E. 1990. Systems Analysts Performance using CASE versus Manual Methods, in *Proceedings of 23rd Annual Hawaii International Conference on System Sciences*, vol IV (R.H. Sprague, Jr. (ed.)).

# 2. CASE Bibliography in Reference Number Order

(Used in update of Wynekoop and Conger's (1991) CASE classification presented in Table 7.1)

## 2.1 Author's Initial Collection

(emphasising empirical work and tool evaluation)

[1] Aaen I. 1994.

[2] Aaen I. 1993.

[3] Aaen I., Siltanen A., Sorensen C. and Tahvanainen V-P. 1992.

[4] Aaen I. and Sorensen C. 1991.

[5] Avison D.E., Golder P.A. and Shah H.U. 1992.

[6] Banker R.D. and Kauffman R.J. 1991.

[7] Bendure C.O. 1991.

[8] Burkhard D.L. 1990.

[9] Chikofsky E.J., Martin D.E. and Chang H. 1992.

[10] Coad P. and Yourdon E. 1991.

[11] Coupe R.T. 1994.

[12] Dimbleby J. and Lo B. 1992.

[13] Doke E.R. 1989.

[14] Gavin C.J. and Little D. 1994.

[15] Glasson B., Quaddus M. and Pneumatikos E. 1992.

[16] Gryczan G. and Kautz K. 1990.

[17] Hayley K.J. and Lyman H.T. 1990.

[18] Howard G.S. and Rai A. 1993.

[19] Huff C.C. 1992.

[20] Huff C., Smith D., Stepien-Oakes K. and Morris E. 1992.

[21] Jones P. 1992.

[22] Jones M.C. and Arnett K.P. 1993.

[23] Katz R.L. 1990.

[24] Kemerer C.F. 1992.

[25] Kemerer C.F. 1989.

[26] Kievit K. and Martin M. 1989.

[27] Land F.F., Le Quesne P.N. and Wijegunaratne I. 1992.

[28] Low G.C. and Jeffery D.R. 1991.

[29] Lyytinen K., Smolander K. and Tahvanainen V-P. 1989.

[30] Lyytinen K. 1990.

[31] McChesney I.R. and Glass D. 1993.

[32] McGinnes S. 1994.

[33] Orlikowski W.J. 1993.

[34] Price Waterhouse. 1989.

[35] Prokit. 1991.

[36] Siltanen A. 1990.

[37] Smolander K., Marttiin P., Lyytinen K. and Tahvanainen V-P. 1990.

[38] Smolander K., Tahvanainen V-P and Lyytinen K. 1990.

[39] Software Management. 1992.

[40] Statland N. 1989.

[41] Stobart S.C., Thompson J.B. and Smith P. 1991.

[42] Swanson K., McComb D., Smith J. and McCubbrey D. 1991.

[43] Thompson K. 1993.

[44] Wijers G.M. and van Dort H.E. 1990.

[45] Wynekoop J.L. and Conger S.A. 1991.

[46] Yellen R.E. 1990.

## 2.2 BCS CASE Conference 1990

[51] Fairbairn D. 1990.

[52] Davy C. 1990.

[53] Haine P. 1990.

[54] Piggott S.A., Hart M.M., Smith P.A. and Witcomb L.E. 1990.

[55] Milan J. and Munt C.E. 1990.

[56] Rego H.B. and Lima J.R. 1990.

[57] Strange G. 1990.

[58] Jackson M. 1990.

[59] Ward P.T. 1990.

[60] Steer K. 1990.

[61] Richards N.L. 1990.

[62] Budgen D. and Marashi M. 1990.

[64] Baba I. 1990.

[65] Parkinson J. 1990.

## 2.3 BCS CASE Conference 1992

[66] Robinson K. 1992.

[67] Barn B.S. 1992.

[69] Stapleton J. and Williams S. 1992.

[70] Gee D.M., Worrall B.P. and Henderson W.D. 1992.

[71] Reynolds W. and Kalra C. 1992.

[72] Beeby M., Parkinson J.S. and Hamilton V. 1992.

[73] Sabanis N. and Darlison A. 1992.

[74] Haine P. 1992.

[75] Olle T.W. 1992.

[76] Law D. and Naeem T. 1992.

[77] Sanders P. and Short K. 1992.

[78] Lloyd-Williams M. and Benyon-Davies P. 1992.

[79] Ip S., Cheung L. C-Y. and Holden T. 1992.

[80] Alexander J. and Hicks K. 1992.

# *Appendices*

# Appendix 1

## Abstracts to Publications

## Arising from the Research

1.1 King S.F. 1992. The Quality Gap: A Case Study in Information System Development Quality and Productivity using CASE Tools, *CASE: Current Practice, Future Prospects*, Wiley, pp. 35-54.

**Abstract**

The ISO definition of quality states that a product must satisfy the customer's sated or implied needs. These needs change over time. A simple graphical indicator is introduced to illustrate the danger of failing to match system deliverables to changing requirements. This indicator is called *"The Quality Gap"*. The quality of an information system developed for the CarMaker car manufacturing group using CASE technology is assessed. The CASE tools used were Analyser and FourthGen from ToolVendor Ltd. Three quality techniques are applied to the project: Gilb's Attribute Specification, Function Point Analysis and a User Satisfaction Survey. The results show that a *functionally-correct* system was developed but was not updated in line with changing user requirements. In other words, a *Quality Gap* appeared. Development productivity is also assessed by comparing actual development effort with an estimate for an equivalent development in COBOL. The results show a six-

fold productivity improvement over the estimated COBOL figure. The reasons for the difference are discussed.

1.2 King S.F. and Galliers R.D. 1994. Modelling the CASE Process: Empirical Issues and Future Directions, *Information and Software Technology*, 36(10):587-596.

**Abstract**

Many current models of information system development originated before the widespread use of Computer Aided Systems Engineering (CASE) tools. The aim of CASE is to automate part of the development process, thus increasing productivity and quality. In this paper, we assess the validity of four significant models of system development against the experiences of three CASE-based development projects. A valid model is a necessary but not sufficient basis for process improvement. The adequacy of the models in addressing the problems encountered in the projects is assessed, and a broader model of information system development is proposed, encompassing *strategic planning*, *organizational learning* and the reconciliation of *alternative viewpoints*.

# *Appendix 2*

# *New Model Project Data*

## 2.1. New Model Effort, Participation and Tool Usage

| Effort | | | Staff Participation | | | | | | | Tool Usage | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Effort (staff hours) | | Research Centre | | West Factory | | South Factory | | | Tool usage | | |
| Week commencing | Project Staff | Users | Author | Consultant | David | Tom | Sponsor | Andre w | Stuart | Analyser | Fourth-Gen | Events |
| 1. 19/3/90 | 18.5 | 1 | * | * | | | | | | * | | Meeting with Bill Green at the Design Centre |
| 2. 26/3 | 27.5 | 13 | * | * | * | | * | | | * | | Various meetings |
| 3. 2/4 | 26 | 5 | * | * | * | * | | | | * | | |
| 4. 9/4 | 0 | 0 | | | | | | | | | | Author on holiday 6/4 90 to 22/4/90 |
| 5. 16/4 | 10 | 0 | | * | | | | | | * | | |
| 6. 23/4 | 45.5 | 2 | * | * | | * | | | | * | | |
| 7. 30/4 | 36.5 | 1.5 | * | * | | | | | | * | * | One meeting |
| 8. 7/5 | 23 | 4 | * | * | * | * | | | | * | * | One meeting |

| Effort | | | Staff Participation | | | | | | | Tool Usage | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Effort (staff hours) | | Research Centre | | West Factory | | South Factory | | | Tool usage | | |
| Week commencing | Project Staff | Users | Author | Consultant | David | Tom | Sponsor | Andrew | Stuart | Analyser | Fourth-Gen | Events |
| 9. 14/5 | 29.5 | 10 | * | * | * | * | | * | | * | * | Analyser "prototype" demonstration |
| 10. 21/5 | 29.5 | 0 | * | * | | | | | | * | * | Main FourthGen work starts |
| 11. 28/5 | 39 | 0 | * | * | | | | | | * | * | |
| 12. 4/6 | 40 | 0 | * | * | | | | | | * | * | |
| 13. 11/6 | 0 | 0 | | | | | | | | | | |
| 14. 18/6 | 46 | 0 | * | * | | | | | | * | * | |
| 15. 25/6 | 34.5 | 0 | * | * | | | | | | * | * | |
| 16. 2/7 | 24 | 18 | * | | * | * | | * | * | * | * | FourthGen New Model system demonstration |
| 17. 9/7 | 22 | 0 | * | * | | | | | | * | * | |

| Effort | | | Staff Participation | | | | | | | Tool Usage | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Effort (staff hours) | | Research Centre | | West Factory | | South Factory | | | Tool usage | | |
| Week commencing | Project Staff | Users | Author | Consultant | David | Tom | Sponsor | Andre w | Stuart | Analyser | Fourth-Gen | Events |
| 18. 16/7 | 25 | 0 | * | * | | | | | | * | * | Demonstra tions to CarMaker IT strategy mgt. |
| 19. 23/7 | 17 | 0 | * | | | | | | | * | | |
| 20. 30/7 | 0 | 0 | | | | | | | | | | |
| 21. 6/8 | 8 | 10 | * | | * | * | | | | * | | Storage capacity meeting |
| 22. 13/8 | 0 | 0 | | | | | | | | | | |
| 23. 20/8 | 0 | 0 | | | | | | | | | | |
| 24. 27/8 | 0 | 0 | | | | | | | | | | |
| 25. 3/9 | 0 | 0 | | | | | | | | | | |
| 26. 10/9 | 0 | 0 | | | | | | | | | | |
| 27. 17/9 | 0 | 0 | | | | | | | | | | |
| 28. 24/9 | 1 | 0 | * | | | | | | | | | |

| Effort | | | Staff Participation | | | | | | | Tool Usage | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Effort<br>(staff hours) | | Research Centre | | West Factory | | South Factory | | | Tool usage | | |
| Week commencing | Project Staff | Users | Author | Consultant | David | Tom | Sponsor | Andre w | Stuart | Analyser | Fourth-Gen | Events |
| 29. 1 10 | 7 | 15 | * | * | * | | * | * | | * | * | "Sales" demonstrat ion at Research Centre: decision to purchase made by the Sponsor |
| 30. 8/10 | 13.5 | 0 | * | | | | | | | * | * | |
| 31. 15/10 | 7 | 0 | * | | | | | | | * | * | |
| 32. 22/10 | 0 | 0 | | | | | | | | | | |
| 33. 29/10 | 0 | 0 | | | | | | | | | | |
| 34. 5/11 | 0 | 0 | | | | | | | | | | |
| 35. 12/11 | 7 | 4 | * | | * | | | | | | * | Start FourthGen VAX installation |
| 36. 19/11 | 10 | 0 | * | | | | | | | | * | |
| 37. 26/11 | 6 | 0 | * | | | | | | | * | * | Writing user manual |

| Effort | | | Staff Participation | | | | | | | Tool Usage | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Effort (staff hours) | | Research Centre | | West Factory | | South Factory | | | Tool usage | | |
| Week commencing | Project Staff | Users | Author | Consultant | David | Tom | Sponsor | Andre w | Stuart | Analyser | Fourth-Gen | Events |
| 38. 3/12 | 25 | 8 | * | | | | | * | | | * | South Factory training |
| 39. 10/12 | 2 | 2 | * | | | | | | | | * | |
| 40. 17/12 | 2 | 4 | * | | * | * | | | | | * | West Factory training |
| 41. 24/12 | 0 | 0 | | | | | | | | | | CHRIST MAS |
| 42. 31/12 | 0 | 0 | | | | | | | | | | |
| 43. 7/1/91 | 0 | 0 | | | | | | | | | | |
| 44. 14/1 | 3 | 0 | * | | | | | | | | * | Restructur ing database |
| 45. 21/1 | 18 | 0 | * | | | | | | | | * | |
| 46. 28/1 | 8 | 2 | * | | * | | | | | | * | David helping with VAX installation and net-working |
| 47. 4/2 | 10 | 5 | * | | * | | | | | | * | |
| 48. 11/2 | 6 | 0 | * | | | | | | | | * | |

| Effort | | | Staff Participation | | | | | | | Tool Usage | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Effort<br>(staff hours) | | Research Centre | | West Factory | | South Factory | | | Tool usage | | |
| Week commencing | Project Staff | Users | Author | Consultant | David | Tom | Sponsor | Andre w | Stuart | Analyser | Fourth-Gen | Events |
| 49. 18/2 | 0 | 0 | | | | | | | | | | |
| 50. 25/2 | 0 | 0 | | | | | | | | | | |
| 51. 4/3 | 3 | 0 | * | | | | | | | | * | The author responds to Andrew's FAX |
| 52. 11/3 | 12 | 3 | * | | * | | | | | | * | |
| 53. 18/3 | 11 | 9 | * | | | | | * | | | * | Visit to South Factory |
| 54. 25/3 | 0 | 0 | | | | | | | | | | |
| 55. 1/4 | 11 | 3 | * | | | | | * | * | | * | Visit to South Factory |
| 56. 8/4 | 0 | 0 | | | | | | | | | | |
| 57. 15/4 | 12 | 2 | * | | | | | | | | * | Visit to South Factory |

| Effort | | | Staff Participation | | | | | | | Tool Usage | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Effort (staff hours) | | Research Centre | | West Factory | | South Factory | | | Tool usage | | |
| Week commencing | Project Staff | Users | Author | Consultant | David | Tom | Sponsor | Andre w | Stuart | Analyser | Fourth-Gen | Events |
| 58. 22/4 | 3 | 0 | * | | | | | | | | * | Working at CarMaker Design Centre |
| 59. 29/4 | 0 | 0 | | | | | | | | | | |
| 60. 6/5 | 0 | 0 | | | | | | | | | | |
| 61. 13/5 | 7 | 0 | * | | | | | | | | * | Working at CarMaker Design Centre |

## 2.2 New Model Defect and Change Analysis

The analysis starts after the requirements determination report was published showing the initial "core" system. Any changes or defects reported after that date are shown in the analysis.

| Defect/ change number | Date raised | Description | Defect or Change | Priority (H=High; M=Med; L=Low) |
|---|---|---|---|---|
| 1 | 11/6/90 | Add "white" PSR status for problems that have been certified as eliminated. | C | L |
| 2 | 29/6 90 | Value of last auto-repeated key field redisplayed prior to addition of next record in scrolling window. Confuses operator who thinks data has been recorded twice. | D | L |
| 3 | | Difficult to successfully delete records in some scrolling windows. | D | M |
| 4 | | Users request some PSR field values to be carried over, with override facility, to next record entered to save rekeying: PSR number, cell number, project id, variant, build phase. | C | M |
| 5 | | PSR field "company responsible" make mandatory input. | C | L |
| 6 | | Increase PSR field "brief problem description" to 55 characters. | C | L |
| 7 | 3/7/90 | Add "IR/PIR number" field to PSR input screen. | C | L |
| 8 | | Allow user to input "ALL" for SDV number. | C | L |
| 9 | | Add launch team data to report DU10. | C | L |
| 10 | | Add report DU20 as first stage of multi stage report DU18. | C | L |
| 11 | | Alter sequence of launch team field headings on all reports | C | L |
| 12 | | Set default value to "ALL" for all fields on report starter screens. | C | L |
| 13 | | Create file DB17 for rapid calculation of PSR totals. | D | M |

| Defect/ change number | Date raised | Description | Defect or Change | Priority (H=High; M=Med; L=Low) |
|---|---|---|---|---|
| 14 | | Allow "root" of VPG number to be entered as search string (eg. "1105") | C | L |
| 15 | | Add *Problem Incidence by PCA Number* report. | C | L |
| 16 | | Change first part of Part Number field from alpha to alphanumeric. | C | L |
| 17 | 2/10/90 | Add referenial integrity warning messages. | D | H |
| 18 | | Provide transparent logon facility. | C | M |
| 19 | | Validate Part Number and Part Description against a master file. | C | L |
| 20 | | Add Target Resolution Date to PSR screen. | C | L |
| 21 | | Add Date of Last Signal Change to PSR screen. | C | L |
| 22 | | Extend record selection options on screen DU2 to include Signal Colour and days since last signal change. | C | L |
| 23 | | Provide windowing facility to lookup Launch Teams, Fault Codes, SDVs, VPGs and Part Numbers when entering PSRs. | C | L |
| 24 | | Provide a report listing PSRs by Part Number. | C | L |
| 25 | | Provide facility to allocate a problem across several different vehicle derivatives. | C | L |
| 26 | 3/10/90 | Increase Model Derivative field from 5 to 10 characters. | C | L |
| 27 | | Rename Model Year field to "Programme" and increase to 4 characters. | C | L |
| 28 | 5/12/90 | Use of line 24 for screen functions on the PC causes the VAX terminals to scroll. | D | L |
| 29 | | Set default screen mode to "lookup" to avoid low security operators being automatically logged off when entering a function. Originally set to "change". | D | M |

| Defect/ change number | Date raised | Description | Defect or Change | Priority (H=High; M=Med; L=Low) |
|---|---|---|---|---|
| 30 | | Add "Quit" option to main menu to avoid "security violation" if Escape key hit accidentally. | D | M |
| 31 | | System error message "018 - ERROR IN WRITING TO FILE - DB9" displayed when attempting to change PSR status in function DU2. | D | H |
| 32 | 11/12/90 | Set PSR fields resolution priority, signal, target resolution date and company responsible to auto-repeat with override. | C | L |
| 33 | | Provide "user-friendly" printer control screen to enable page layout to be specified (landscape or portrait) and output redirected to file or screen. | C | M |
| 34 | | Allow a PSR to be assigned to a launch team for information purposes only (signified by blank signal colour). | C | L |
| 35 | 20/12/90 | Display "selecting records" message in function DU2 to tell user that something is happening (slow selection). | C | L |
| 36 | | PSR numbers are not unique across the system, only within a project. Therefore need to prefix PSR number with project number to make unique. Requires significant reorganisation of database and means West Factory can't use system until fixed in case of clash with South Factory New Model numbers. | D | M |
| 37 | | Extend project identification to include the four fields: Projet Number, Programme, Variant and Build Phase. | D | M |
| 38 | | Link Resolution Priority to the PSR, not to the Launch Team. | C | L |
| 39 | | Make SDV number unique within a Programme rather than a Project. | C | M |
| 40 | | Increase length of Programme field to 10 characters and Derivative to 20 characters. | C | L |
| 41 | | Set exit from main menu to return user to VMS, not to the FourthGen logon screen. | C | M |

| Defect/ change number | Date raised | Description | Defect or Change | Priority (H=High; M=Med; L=Low) |
|---|---|---|---|---|
| 42 | 17/1/91 | Enable wildcard selection of PSR numbers e.g. PSR Number = BS* for all PSRs allocated to Bill Smith. | C | M |
| 43 | | The three problem incidence reports (DU12, DU14 and DU16) print PSRs that do not match the launch team and signal colour record selection criteria due to a bug in the deselection logic in FourthGen. The author attempted a complicated fix involving scratch files which failed to work. | D | M |
| 44 | 18/1/91 | Choice of printers to include HP laser printer as well as dot-matrix. | C | L |
| 45 | 24/1/91 | File variable help messages cannot be displayed. FourthGen bug. | D | L |
| 46 | 28/2/91 | Deletion of launch team in DU4 (Launch Team Maintenance function) does not lead to deletion in DB17 (PSR totals file), therefore deleted launch team continues to be printed in the PSR totals reports and screen (DU22). | D | M |
| 47 | | PSR totals incorrect in file DB17, leading to incorrect totals printed against launch teams. These are key reports that are used to control the problem resolution process. | D | H |
| 48 | 13/3/91 | Allow "ALL" to be entered as selection criteria in screens and reports that select on Programme, Build Phase or Project. | C | L |
| 49 | | Add new report based on the manual report given to the author by Stuart at South Factory. This "simple" report required three FourthGen report functions to implement and still didn't work! | C | M |
| 50 | | Add update function to DU22 to automatically reset PSR totals in file DB17 before report is run (attempt to overcome incorrect calculation of PSR totals). Problem still persisted after this fix. | D | H |
| 51 | | Modify PSR Maintenance function (DU1) to ensure null signal cannot be entered. This was thought to be the cause of the PSR totals problem. Problem persisted after this fix! | D | H |

| Defect/ change number | Date raised | Description | Defect or Change | Priority (H=High; M=Med; L=Low) |
|---|---|---|---|---|
| 52 | | Investigate sending report output to file in order to export PSR data to the Warranty system in the format requested by SystemsHouse. | C | L |
| 53 | 18/3/91 | Spurious launch team details retrieved from file when going from Add mode to Change mode in DU1 (PSR Maintenance screen). File DB9 stores launch team details, this is same file for which the "ERROR - 018" message was displayed (see defect 31). | D | H |
| 54 | | South Factory request facility to enter logical expression as record selection criteria for reports e.g. Select all PSRs where launch team # "TRIM" and signal # "W". FourthGen does not provide a query language, therefore a limited facility had to be hard-coded by the author. A tortuous and not very successful solution. | C | M |
| 55 | | Add new report: PSR status by Problem Owner. | C | L |
| 56 | | Add new field to Brief Problem Analysis report: PCA Number. | C | L |
| 57 | | Add new reports: PSRs Past Target Resolution Date and PCAs Past "Into PCC Date". | C | L |
| 58 | | Require facility to split launch teams into sub-launch teams and assign PSRs to sub-launch teams as well as to launch teams. | C | M |
| 59 | 4/4/91 | Add "END OF REPORT" banner to all reports. | C | L |
| 60 | | Allow three ways of selecting PSR by signal value:<br>1. Enter single signal value.<br>2. Enter range of signal values.<br>3. Enter "ALL". | C | M |
| 61 | | Add "Grand Totals" line to the PSR Totals report (DU22). | C | L |
| 62 | | Screen fails to scroll to show more records when reports directed to screen on VT340 terminals at South Factory. Problem only occurs on some reports. | D | H |

| Defect/ change number | Date raised | Description | Defect or Change | Priority (H=High; M=Med; L=Low) |
|---|---|---|---|---|
| 63 | | For all reports: print count of records printed at end of report. | C | L |
| 64 | | Modify report heading text on DU18. | C | L |
| 65 | | Suppress printing of certain launch teams in report DU18. | C | M |
| 66 | 18/4/91 | Allow "*" to be entered to denote "NOT SIGNAL" e.g. "* R" means list all signals apart from Red. | C | M |
| 67 | | Allow entry of blank signal value (space) to enable PSRs posted to launch teams for information purposes only to be listed. | C | L |
| 68 | 14/5/91 | Develop new PSR totals report and associated functions to overcome incorrect calculation of PSR totals. The report uses file DB9 instead of the summary file DB17 and entails writing seven new functions (DU33 to DU39). The work was never completed. | D | H |

## 2.3 New Model Stakeholder Views: Interview Transcripts

1. The Sponsor - The CarMaker New Model Centre Manager and Project Sponsor (interviewed 25/7/91).

The Sponsor described the pre-production problem management situation at the time of the start of the New Model system project. Problems were being handled by three different areas: CarMaker's Design Centre Vehicle Test, Test Track and the New Model Centre, West Factory. Each area used a different method for recording and managing problems. The method used by the New Model Centre had been developed during the new Family Car project. This approach had been very successful for the Family Car project and the Sponsor was keen to involve other areas in the New Model Centre system in time for the start of the Executive Car project at South Factory. At that time, the South Factory New Model system was paper-based and was seen as considerably inferior to the computer-based West Factory system. The Sponsor therefore viewed this project as an opportunity to extend the West Factory New Model system to embrace all pre-production problem management activities at both West Factory and South Factory.

The FourthGen New Model system was delivered in time to provide useful support to the Executive Car project. The Sponsor viewed the system as "suffering from its own success" in that as more areas began to use the system, there was demand for a greater range of reports than had originally been envisaged. It was intended that a member of the West Factory New Model team provide end-user support for the system. However, due to the company re-organisation, the West Factory operation separated from the South Factory operation and support became no longer available. The lack of local system support meant that new reports could not be created without some delay and the users at South Factory became frustrated with the apparent inflexibility of the system.

In the meantime the Executive Car project had progressed to the stage where forward considerations of problem control post-volume were being explored with South Factory Warranty Group. Warranty had developed their own post-volume problem tracking system in conjunction with SystemsHouse. The system had been refined by a member of the Warranty team and was being used successfully at South Factory. A decision was made to combine the two systems. The issue of flexibility was seen as crucial here. Since the Warranty system was supported locally by both a member of the Warranty staff and SystemsHouse, whilst the New Model system had no established local support, the decision was made to use the Warranty system as the basis for the combined system. The separate New Model system was therefore being phased out as the Executive Car approached launch.

The Sponsor saw the major problem with the project as being the lack of local support for the New Model system once it was installed at South Factory. The time pressure imposed by the tight schedule for the Executive Car project meant that South Factory staff could not be made available for training in system support. More frequent on-site support could have been provided by members of the Research Centre/ToolVendor development team but it was recognised that South Factory would have had to become self-sufficient eventually. The need to combine the New Model and Warranty systems could not have been envisaged by the Sponsor at the start of the project since at that time his attention was completely focused on pre-launch activities. The Sponsor was aware of the South Factory Warranty system but did not think it appropriate for New Model management at that time.

With regard to future problem management system development, the Sponsor stated that many of the successful developments arising from the Executive Car project would be carried forward to the next model project at South Factory. The Sponsor was therefore keen that the CarMaker IT Strategy team learn from the Executive Car problem management experience by talking to the staff involved in the project. This had yet to happen.

### 2. Stuart - The South Factory New Model Manager (25/7/91).

Stuart described the need for an improved New Model system. The spreadsheet-based system developed at West Factory was seen as very difficult to use, slow and storing minimal data. Stuart was looking for a more flexible system that would provide a wide variety of management reports for all stages of pre-production build from D02 to Volume plus 90 days (the Warranty cut-over point).

The New Model system delivered by this project did provide a greater variety of reports and was easier to use. However, the number of reports required by different users was far greater than Stuart had envisaged. The lack of local support for the new system meant that new reports could not be developed sufficiently quickly to meet this demand. Stuart became frustrated by the lack of flexibility of the system. The ideal reporting system was seen as one where the user could specify a report containing any number of attributes from the database selected on any combination of attributes.

The need for a very flexible New Model system was accentuated by the time pressure of the Executive Car project. There wasn't time for the New Model staff at South Factory to develop the skills to support a new system, therefore either the development of new reports had to be very easy or on-site system support had to be provided. In contrast, the South Factory Warranty problem management system was supported locally and had been developed over a period of time to provide an extensive range of reports for Warranty. A decision was made to combine the two systems into a single system to unify pre- and post-production problem management. Stuart decided to invest in extending the Warranty system to include pre-production problem data rather than extend the New Model system to include post-production problem data. The main reason for this decision was the availability of local system support.

Stuart was not involved in the initial specification of the New Model system and was surprised that the West Factory staff who were involved had not emphasised the need to interface to the Warranty system since the need for an interface must have become evident in the Family Car project. The scope of the project should therefore have been extended to include the Warranty interface.

Stuart believed that the correct issues and objectives had been established for the project and found the visits to the Research Centre to review the prototype most useful in bringing together New Model staff from both West Factory and South Factory.

### 3. Andrew - The South Factory New Model Engineer (late July 1991)

Andrew had no experience of problem management before the Executive Car project. He assumed (admittedly naively) that the FourthGen New Model system was like a package that could be installed with little thought to maintenance and support. The system had been a victim of its own success as the extensive range of standard reports led to users wanting their own set of "customised" reports. Although he had not used the spreadsheet-based New Model system, he was familiar with the underlying spreadsheet product. He believed that it was easier to customise reports using the spreadsheet system, for example logic strings could be entered for record selection, however FourthGen response time was considerably faster than the spreadsheet. The FourthGen New Model screen had a different layout to the paper PSR form (due to the need to place repeating groups at the end of the screen). This slowed down data entry as the operators had to scan the form to locate the data. The extensive data validation facilities were liked; in contrast the ApplicationMaster system provided no data validation apart from date format.

Undoubtedly the lack of local support had caused problems, but the SystemsHouse ApplicationMaster system had problems too. Like the FourthGen system, it provided a range of standard reports which the users couldn't alter. Unlike the FourthGen system SystemsHouse staff were available at South Factory to make alterations (at a price). The SystemsHouse New Model/Warranty system was likely to go company-wide in 1992, with the 1993 model year New Car project at East Factory using it as well as the CarMaker divisions at West Factory and South Factory. West Factory Warranty already used a reduced form of the South Factory Warranty ApplicationMaster system. However, SystemsHouse support was not trouble-free; the analyst assigned to the system was also assigned to other projects and was often not available. The system had not been ready for the Quality Proving build phase and suffered from bugs, therefore the FourthGen system was used in parallel for longer than expected which had drained resources and resulted in a two-week moratorium on issuing reports to users. The SystemsHouse system response time was about the same as FourthGen, but PSR data was displayed across two screens instead of one which slowed down viewing. A further reason why the Warranty system was preferred to the FourthGen system was the archive of several years of problem data held in the Warranty system. This was used by Warranty staff who were not keen to transfer the data to a different system.

The Executive Car Chief Engineer who visited the Research Centre to view the FourthGen prototype had been keen to embrace the latest technology and wanted graphics included in the FourthGen system. Neither the FourthGen system, nor the ApplicationMaster system satisfied this wish. An organisational problem had arisen due to the Interior Trim launch team progressing PSRs locally and not feeding the current status back to PSR Control. This had meant that the progress reports generated by the system were out of date, but was not the fault of the FourthGen system.

Andrew found the visits to the Research Centre very useful in improving his understanding of the system. He thought that it had been right to let the West Factory staff drive the project at the start since the South Factory staff had no experience of the spreadsheet New Model system. However, there was no consultation with PSR Control's main customers - the Joint Engineering Teams, which represented all of the launch teams. These people were key users of the reports but were not involved in the development process. The original objectives were correct at the time but had changed subsequently. They were somewhat "woolly", but so were the objectives put forward by SystemsHouse for the ApplicationMaster-based replacement system.

Andrew had given a very brief presentation on the FourthGen system to a group of users at the start of operation but had not explained the systems capabilities and limitations. User expectations were therefore not controlled. The system should have been demonstrated to all Chief Engineers in order to gain greater support. Certainly a South Factory-based CarMaker IT person should have been identified to support the FourthGen system. Andrew was disappointed that the CarMaker IT Strategy team had shown no interest in the project and felt that they could learn from the South Factory experience.

### 4. Stan - The South Factory New Model System Operator (late July 1991).

Stan felt that the FourthGen system was "getting there" by the end. The system had many good points, for instance extensive data validation that reduced mistakes and the system provided a "great" range of standard reports. Stan felt that the greater range of data held by the system had led to a greater demand for reports ("the more you put in, the more they want out"). Furthermore, these reports may only be required once or twice. The type of reports required could not be envisaged at the start. However, the system was inflexible and the screen layout unhelpful at times, for example the Problem Status field in the middle of the PSR screen would often be the only change on the screen, therefore the user would have to tab through half a screen full of data before making the single keystroke change. The lack of logical expressions for record selection had been a major drawback; this facility was available in the spreadsheet package in use at South Factory and was much needed.

South Factory staff were under tremendous time pressure to prepare for the Executive Car launch. In fact, the Quality Proving build date had been brought forward. There was certainly no time for system training or for working around system defects, such as the printer paging and screen scrolling problems. Furthermore, the volume of PSRs had been far greater than expected. Time was lost because the FourthGen system could not spool printer output therefore data entry had to wait until the report was printed. In comparison, the ApplicationMaster system supported spooling.

Finally, Stan felt the new ApplicationMaster system had benefitted greatly from the lessons learned and facilities provided by the FourthGen system.

# *Appendix 3*

# *Warranty Project Data*

## 3.1 Warranty Effort, Participation and Tool Usage

| Effort | | | Staff Participation | | | | | | | Tool Usage | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | Effort (staff hours) | | Research Centre | | | Warranty East Factory | | | | Tool usage | |
| Week commencing | Project staff | Users | Author | Consult-ant | Paul | Sponsor | Jack | Malcolm | Others | Analyser | Events |
| 1. 14/5/90 | 3 | 5 | | * | * | * | * | | the Research Manager | | Meeting at East Factory |
| 2. 21/5 | 7 | 13 | * | * | * | * | * | * | the Warranty Director, Tony, Pete | | Meeting with the Warranty Director |
| 3. 28/5 | 10 | 0 | * | * | * | | | | | * | |
| 4. 4/6 | 15 | 15 | * | * | * | * | * | * | | * | DFD feedback at Warranty |
| 5. 11/6 | 3 | 23 | | * | | | * | * | | | East Factory meeting |
| 6. 18/6 | 17.5 | 40 | | * | * | * | * | * | | * | Two Warranty meetings |
| 7. 25/6 | 23 | 21 | * | * | * | | * | * | the Research Manager | * | One Warranty meeting |

| Effort | | | Staff Participation | | | | | | | Tool Usage | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | Effort (staff hours) | | Research Centre | | | Warranty East Factory | | | | Tool usage | |
| Week commencing | Project staff | Users | Author | Consult-ant | Paul | Sponsor | Jack | Malcolm | Others | Analyser | Events |
| 8. 2/7 | 25 | 5 | | * | * | * | * | * | the Research Manager, the Warranty Director | * | Meeting with the Warranty Director |
| 9 9/7 | 0 | 0 | | | | | | | | | |
| 10. 16/7 | 2.5 | 2.5 | | * | | | | * | | * | Draft Requirements Phase |
| 11. 23/7 | 11.5 | 6.5 | | | * | | | * | the Research Manager | * | |
| 12. 30/7 | 3.5 | 0 | | | * | | | | | * | |
| 13. 6/8 | 0 | 0 | | | | | | | | | |
| 14. 13/8 | 16 | 10 | * | * | | | * | * | the Research Manager, Data Analyst | * | Warranty and Data Analyst meetings |
| 15. 20/8 | 7 | 3 | * | | | | | * | the Research Manager | * | v3 "Current System" created |
| 16. 27/8 | 8 | 5 | * | | | | | * | | * | |
| 17. 3/9 | 8 | 8 | * | * | | | * | * | | * | |

| Effort | | | Staff Participation | | | | | | | Tool Usage | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | Effort (staff hours) | | Research Centre | | | Warranty East Factory | | | | Tool usage | |
| Week commencing | Project staff | Users | Author | Consult-ant | Paul | Sponsor | Jack | Malcolm | Others | Analyser | Events |
| 18. 10 9 | 7 | 6.5 | * | * | | * | * | * | | * | Warranty meeting |
| 19. 17/9 | 2 | 1 | * | | | | * | | | * | |
| 20. 24 9 | 7 | 1 | * | * | | | | | Simon | * | |
| 21. 1/10 | 4 | 12 | * | * | | * | * | * | Pete + 2 engineers | | First "lunchtime" meeting at Warranty |
| 22. 8/10 | 6 | 11 | * | | | * | * | * | 2 engineers | * | Second "lunchtime" meeting at Warranty |
| 23. 15/10 | 4 | 14 | * | * | | * | * | * | Pete + 3 engineers | | Third "lunchtime" meeting at Warranty |
| 24. 22/10 | 0 | 0 | | | | | | | | | |
| 25. 29/10 | 1 | 0 | * | | | | | | | * | |
| 26. 5/11 | 0 | 0 | | | | | | | | | |
| 27. 12/11 | 18 | 84 | * | * | * | * | * | * | Pete, Tony + 9 engineers | | JAD session at ToolVendor |
| 28. 19/11 | 0 | 0 | | | | | | | | | |
| 29. 26/11 | 0 | 0 | | | | | | | | | |

| Effort | | | Staff Participation | | | | | | | Tool Usage | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | Effort<br>(staff hours) | | Research Centre | | | Warranty East Factory | | | | Tool usage | |
| Week commencing | Project staff | Users | Author | Consult-ant | Paul | Sponsor | Jack | Malcolm | Others | Analyser | Events |
| 30. 3/12 | 5 | 0 | * | * | | | | | | * | |
| 31. 10/12 | 18 | 0 | * | | | | | | | * | |
| 32. 17/12 | 18 | 6 | * | * | | | * | | | * | Final meeting at Research Centre |
| 33. 24/12 | 0 | 0 | | | | | | | | | CHRISTMAS |
| 34. 31/12 | 0 | 0 | | | | | | | | | |
| 35. 6/1/91 | 0 | 0 | | | | | | | | | |
| 36. 13/1 | 3 | 0 | | * | | | | | | * | |

## 3.2 Warranty Stakeholder Views: Interview Transcripts

1. The Sponsor - The East Factory Warranty Manager and Project Sponsor (interviewed 22/3/91).

The Sponsor explained his original expectations of the project by reference to the situation in the Warranty Group when the project was started in May 1990. The department, having been recently set up, was in a totally reactive mode. There were regular changes of direction, focusing attention first on one information source, then another e.g. Warranty Claims, 1200 Mile Audit. The department was looking for an infrastructure, a coherent way of working. He hoped to establish a system (not necessarily computer-based) which could collate the available problem information. The approach suggested by the Research Centre and ToolVendor of using a modelling tool to assist discussion of the business seemed to represent a useful opportunity.

The project produced significant benefits by providing a clear picture of how the organisation worked, in a form which everyone could understand. This was a great help later in the year when a major corporate reorganisation provoked a comprehensive review of all Warranty and Service activities. The modellıng undertaken during this project gave the East Factory group greater clarity of thinking about their role than was the case with their counterparts from other areas of the company. East Factory therefore became the leaders in discussions which led to recommendations for changing the Warranty/Service structure across the whole company. Jack used the DFDs from this project to drive those discussions.

At a more personal level, there were great benefits to the members of staff who had attended the ToolVendor analytical techniques course (Jack and Malcolm) and then had experience in using the techniques during the project. Jack and Malcolm now felt very much at home with the techniques and used them as a matter of course during discussions within the department.

In considering how the project might have been better run, the Sponsor felt that there were a number of clear lessons to be learned. There was insufficient user commitment to the project and the need for continuous effort from them had not been understood. The project was regarded as a low priority, background activity, and this led to the elapsed time expanding. The reason for the low priority was that the project had not been adequately sold to the departmental director and management team, who therefore did not understand the project sufficiently to give it the support it deserved. The Sponsor and Tony could not agree on how problems were to be managed. Guidance from the Warranty Director was needed to resolve the disagreement, but was not forthcoming. The Sponsor also felt that there was a culture problem in manufacturing industry in being reluctant to commit time and effort to thinking out how the business should be run: there was an inherent preference for operating in a reactive way to the current situation. Warranty was therefore perceived as "administration" that did not add value to the product. The Warranty director did not understand the problems facing the Sponsor's staff and did not agree with the cost/benefit analysis. Furthermore the Sponsor and Jack tended to take ownership of the project to the exclusion of the other Warranty managers.

Partly because of the low priority, the appropriate staff involvement was not established in the early stages of the project. The Sponsor expected the project to involve mainly his own staff, Jack and Malcolm (since they were charged with establishing appropriate system support), with limited input from other people in the department. The other managers were initially pleased that it did not take up much of their time, but later became frustrated at their lack of involvement. They all now recognised that many of the department's procedural problems still exist.

A further difficulty was the wide divergence of expectations regarding computer support. The Director was somewhat sceptical of the benefits of computers, feeling that they represented a loss of control and were too expensive. Some managers and staff hoped for a system with a terminal on everyone's desk. Certainly Jack was keen to be at the centre of things, and wanted a computer-based solution. Others expected a computer system to be run by one person who would provide a service to everyone else. Although the project had not reached the stage of a decision in this respect, the divergence of attitudes made it difficult to focus the objectives of the project. In addition, the scope of the project was probably too tightly constrained to objectives that could be quantified by the pressure to identify a tangible cost/benefit case. Warranty had since moved away from a cost-saving mindset and were now focusing on improving service to their customers.

2. Jack - The East Factory Warranty Systems Manager (28/3/91).

Jack expected significant office automation to arise from the project, with a terminal in each Warranty work area. He identified many benefits from the project. The techniques learned were "tremendously useful" and had enabled the systems staff to think clearly about the Warranty information requirements. By developing the data flow diagrams Jack had identified how confused things were, and had become aware of areas of duplication which could be addressed. This enabled Jack to focus his own local system development efforts on areas that would deliver the most immediate benefits. Finally, the DFDs had been a great help in clarifying requirements in the departmental re-organisation which saw Warranty merge with the Service operation.

The project had encountered difficulties for a number of reasons. Ideally the project should have been established on a more committed basis but this would probably have prevented the project from being approved in the first place. Two of the Warranty product managers, Pete and Tony should have been involved right from the start instead of later as their support was vital for project success. The real problem: that of identifying a product problem had been overshadowed by the focus on cost-benefit analysis at the start of the project. As for the method adopted, too little time was spent modelling the current system. The research team tried to move too quickly to modelling the new system. Jack had hoped initially that the CASE tool would be used by the end users, but had not had the time to learn how to use the tool himself. Finally, the project was eventually overtaken by the major company-wide re-organisation.

### 3. Malcolm - The East Factory Warranty Engineer (4/7/91).

Malcolm described the original aims and objectives of the project as being correctly identified and still relevant today. There was still a need for better problem recording, quantification and resolution. The scope of the project and the players involved were also correctly identified. The accuracy of the original analysis had enabled the data flow diagrams to be used as the basis for a model of the new Problem Information Control Centre (PICC) which was set up after Christmas.

The emphasis on user involvement encouraged by the analytical techniques used was viewed as most beneficial. Malcolm believed that the DFDs were self-explanatory and were a major aid to better communication between users and system developers. However, the CASE tool seemed to encourage the users to "fill in all the boxes" leading to too detailed analysis. Similarly the JAD session had been too detailed and had placed insufficient emphasis on the top-level view and what was to be delivered.

The slow progress of the project was caused by several factors, the primary ones being the low priority placed on the project in comparison to the day to day operations of Warranty; and the lack of clearly visible deliverables with which to gain the support of senior management. The DFDs were seen as useful communication devices but were not sufficiently tangible to excite senior management interest. A prototype system would have helped retain enthusiasm. Malcolm noted that considerable progress was achieved only when the re-organisation of Warranty problem control centre into the PICC raised the profile of the project and put pressure on the Sponsor, Malcolm and Jack to deliver a system quickly. Until that point, the project was viewed as a background activity to be fitted in between the more important operational activities. On a personal note, Malcolm found the application of the techniques varied between the three analysts involved (the author, the Consultant and Paul). This was somewhat confusing. Early training in the analytical techniques would have reduced Malcolm's confusion and would have helped Malcolm make a more informed contribution to the analysis process.

A formal steering body would not have helped and would have detracted from the "real work". Recent experience with setting up the PICC had showed a lack of management commitment to a steering body.

To summarise, Malcolm suggested the following improvements to the project:

1. Total commitment to the project from all involved i.e. Warranty, the Research Centre and ToolVendor.

2. An agreed schedule of meetings with clear objectives for each meeting.

3. Tangible deliverables to be produced at regular intervals.

4. Each deliverable to be "signed off" by a senior manager so that accountability can be maintained and progress made.

5. Details of data flows, data stores and processes not to be documented until the high-level model is agreed and signed-off.

6. Early training on analytical techniques and the deliverables expected is essential for the key players to make a full contribution to the analysis process (Malcolm was sent on ToolVendor analytical techniques course too late in project).

4. Simon - The CarMaker IT Strategist (24/7/91)

Simon described the company-wide quality initiative which aimed to develop a unified approach to handling quality problems at all stages of the vehicle life-cycle. The initiative had two strands: a pro-active strand and a reactive strand. The pro-active strand aimed to eliminate problems at the design stage. The reactive strand aimed to manage better problems that occurred at the later stages of the life-cycle, such as customer-reported faults in vehicles. Feedback of the problem details from the later stages to the design stage was seen as vital in order to avoid re-introducing problems in subsequent designs. A major issue at present was that each separate business unit (South Factory, West Factory and East Factory) had its own method for handling problems. These methods did not embrace the complete vehicle life-cycle and there was insufficient feedback of problem details from the later stages to the design stage.

The company-wide quality initiative came to focus on the Warranty area several months after the East Factory/Research Centre/ToolVendor project had commenced. Simon therefore viewed the East Factory project as a useful detailed case study from which to gain a better understanding of how problems were handled at present. The process and data models from the East Factory project were subsequently used by

Simon in discussions with CarMaker management over the change to a unified problem management approach.

The project issues and objectives were seen by Simon as appropriate for the short-term needs of the East Factory Warranty management, namely to understand better their hectic business. However, the scope could have been extended usefully to encompass the complete life-cycle of a problem. This would have broadened the study to include other functional areas such as Service and Product Engineering. The failure to take this broader view was seen as surprising since a member of the East Factory Warranty team (Jack) was actively involved in the cross-functional quality initiative and might have been expected to communicate details of the initiative to the project team. In general, the project was seen as working at too low a level, dealing with the mechanics of the system rather than strategic issues. This was due to the lack of senior management input. Simon summarised this issue by stating that the project was insufficiently embedded in the overall IT Strategy work taking place and lacked an owner with the "right form of business need".

### 5. The Warranty Director - The East Factory Warranty Director (6/8/91).

The Warranty Director described the problems with information provision when he took over East Factory Warranty some two years earlier. At that time engineers were working into the early hours of the morning to provide the information necessary to manage problems. The Warranty Director expected this project to address this problem by improving the precision and rapidity of information available to

Warranty. To a large extent this aim had been achieved and engineers were no longer devoting excessive amounts of time to information gathering and processing.

The Warranty Director was not closely involved in the project, preferring to leave project leadership to one of the Warranty managers (the Sponsor). He regarded the Warranty systems operation as a "black box" which was expected to provide consolidated problem information from a variety of sources on demand. To this end, the Warranty and Service problem management activities were combined into a single Problem Information Control Centre (PICC) early in 1991. Since the Warranty Director was not closely involved in the project, he was not in a position to identify any impact made by the project on the merger discussions. However, following the project, the Warranty Director had noticed a considerable improvement in the ability of the East Factory systems personnel involved in the project to present information.

The Warranty Director was not surprised by the lack of involvement or commitment of the other Warranty managers to the project. He viewed this as "a fact of life" since these managers had their own, more pressing priorities and could not be expected to become heavily involved in a systems project. In view of this, the Warranty Director suggested that systems developers should be prepared to develop systems with a level of involvement in proportion to the users' other commitments, having first agreed the objectives to be achieved. Regular reviews would however be required. It was absolutely essential that systems projects provide regular deliverables and showed demonstrable benefits. A period of some six months between project inception and the delivery of an initial working system was viewed as reasonable. This should be followed by iterative improvements to the system as additional requirements arose.

On a personal note, the Warranty Director was critical of past systems projects in which he had been involved. One important system had proved unworkable and was abandoned in favour of a return to the previous manual system. He didn't know of any system that had been produced on time, but did appreciate the importance of information systems as East Factory used an MRP system heavily.

From the Warranty Director's viewpoint, the project had failed to provide sufficient tangible deliverables. The approach taken of consulting all interested parties and carefully documenting the proposed system was "too grand". The more pragmatic approach adopted by Jack and the PICC staff of delivering a number of reports that met the Warranty Director's immediate needs as the first step was seen as preferable. Similarly, the idea of a formal steering body for the project was not welcomed as this was seen as adding to project bureaucracy and was unlikely to have contributed to project success. Again, the Warranty Director cited an example of a recent CarMaker systems project where the steering body had "mushroomed" and diverted effort away from development.

In conclusion, the Warranty Director noted that East Factory had retained a visible Warranty function after the recent company-wide review of the Warranty Groups, whereas the divisions at West Factory and South Factory had absorbed the Warranty activity into the Product Supply business unit. This "says something" about Jack's systems.

The Warranty Director was sent the interview transcript subsequently for approval. He added one final comment:

> *"Again on a personal note, I am more inclined to support an 'Information Development Project' than a 'Systems Development Project'. Words, I know, but it felt like the latter."*

# *Appendix 4*

# *Recycling Project Data*

# 4.1 Recycling Project Effort, Participation and Tool Usage

The author officially joined the project on a half-time basis from 1 May 1992. He commenced detailed data collection at this date, and continued to collect effort data with the help of Mary until the end of July 1993. Collection of effort data ceased five months before the author left the Research Centre due to the author's increasing involvement in other activities and Mary's wish to be freed from the burden of detailed data collection in the author's absence.

The project staff effort figures presented below include Mary's time as a student working with the author. Geoff is counted as a user despite his early database development activities, which had ceased by 1 May 1992. For monthly progress meetings an estimate of the amount of discussion time devoted to the database has been used and multiplied by the number of attendees. Clearly many other matters were discussed at the meetings and not all attendees participated in the database discussion.

| Effort | | | Staff Participation | | | | | | | | | | Tool Usage | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Effort (staff hours) | | Database Developers | | Senior Management | | | Recyclate Ltd. Data Collectors | | | Data Users | | Tool usage | | | |
| Week commence. | Project staff | Users | Author | Mary | Terry | Accountant | Environmental Strategist | Robert | Jim | Ronnie | Geoff | Chris | Quick Store | Analyser | Number Crunch | Events |
| 1. 27 4/92 | 2 | 4 | * | | | | | | | | * | * | | | | |
| 2. 4 5 | 0 | 0 | | | | | | | | | | | | | | |
| 3. 11 5 | 1 | 11 | * | | * | * | * | * | | * | * | * | | | | Monthly meeting |
| 4 18/5 | 6 | 4 | * | * | | | | | | * | | | * | | | Database "readopted" from Geoff |
| 5 25/5 | 32 | 5 | * | * | | | | | | * | | | * | | | |
| 6. 1/6 | 32 | 6 | * | * | | | | | | * | | | * | | | |
| 7. 8/6 | 1 | 8 | * | | * | * | * | * | | | * | * | * | | | Monthly meeting |
| 8. 15/6 | 16 | 7 | * | * | | | | * | | * | | * | * | | | |
| 9. 22/6 | 7 | 2 | * | * | | | | * | | | | | * | | | |
| 10. 29/6 | 4 | 0 | * | | | | | | | | | | * | | | |
| 11 6/7 | 7 | 11 | * | | * | * | * | * | | | * | * | * | | | Monthly meeting |
| 12. 13/7 | 16 | 0 | * | * | | | | | | | | | * | | | |
| 13. 20/7 | 0 | 0 | | | | | | | | | | | | | | |
| 14. 27/7 | 4 | 2 | * | * | | | | * | | * | | | * | | | |

| Effort | | | Staff Participation | | | | | | | | | | Tool Usage | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Effort (staff hours) | | Database Developers | | Senior Management | | | Recyclate Ltd. Data Collectors | | | Data Users | | Tool usage | | | |
| Week comm ence. | Project staff | Users | Author | Mary | Terry | Accoun -tant | Environ- mental Strategist | Robert | Jim | Ron- nie | Geoff | Chris | Quick Store | Analy- ser | Number Crunch | Events |
| 15 3 8 | 11 | 1 | * | | | | | | | | | * | * | | | |
| 16 10 8 | 10 | 9 | * | | * | * | * | * | | | * | * | * | * | | Monthly meeting and Analyser ERM |
| 17 17 /8 | 0 | 0 | | | | | | | | | | | | | | the author on holiday |
| 18. 24 8 | 0 | 0 | | | | | | | | | | | | | | |
| 19. 31/8 | 0 | 0 | | | | | | | | | | | | | | |
| 20. 7/9 | 0 | 0 | | | | | | | | | | | | | | |
| 21. 14/9 | 4 | 0 | * | | | | | | | | | | * | | | |
| 22. 21/9 | 7 | 14 | * | | * | | | * | | * | | | * | | * | Database requirem ents meeting at Re-cyclate Ltd |
| 23. 28/9 | 15 | 17 | * | | * | * | * | * | | | * | * | * | | | Monthly meeting |
| 24 5/10 | 11 | 0 | * | | | | | | | | | | * | | | |
| 25. 12/10 | 0 | 0 | | | | | | | | | | | | | | |

| Effort | | | Staff Participation | | | | | | | | | | Tool Usage | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Effort (staff hours) | | Database Developers | | Senior Management | | | Recyclate Ltd. Data Collectors | | | Data Users | | Tool usage | | | |
| Week commence. | Project staff | Users | Author | Mary | Terry | Accountant | Environmental Strategist | Robert | Jim | Ronnie | Geoff | Chris | Quick Store | Analyser | Number Crunch | Events |
| 26.<br>19 10 | 6 | 0 | * | | | | | | | | | | * | | | |
| 27.<br>26 10 | 6 | 3 | * | | | | | * | | | | | * | | | |
| 28.<br>2/11 | 1 | 8 | * | | * | * | * | * | | | * | * | | | | Monthly meeting |
| 29.<br>9/11 | 31 | 1 | * | * | | | | | | | * | | * | * | * | Mary joins full-time |
| 30<br>16/11 | 45 | 7 | * | * | | | | * | | | | | * | * | * | Process modelling |
| 31.<br>23/11 | 47 | 17 | * | * | * | | | * | | | * | * | * | | | |
| 32.<br>30/11 | 36 | 16 | * | * | | | | * | | | * | | * | | | Geoff "explodes" |
| 33.<br>7/12 | 53 5 | 34.5 | * | * | | | | * | * | * | * | * | * | | | Crisis meeting at Recyclate Ltd. |
| 34.<br>14/12 | 67 | 68 | * | * | * | * | * | * | * | * | * | * | * | | | Monthly meeting; the author goes full-time |

| Effort | | | Staff Participation | | | | | | | | | | Tool Usage | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Effort (staff hours) | | Database Developers | | Senior Management | | | Recyclate Ltd. Data Collectors | | | Data Users | | Tool usage | | | |
| Week commence. | Project staff | Users | Author | Mary | Terry | Accountant | Environmental Strategist | Robert | Jim | Ronnie | Geoff | Chris | Quick Store | Analyser | Number Crunch | Events |
| 35.<br>21 12 | 27 | 73 | * | * | | | | * | * | * | * | * | * | * | | "Team" view Car4 strip |
| 36<br>28 12 | 0 | 0 | | | | | | | | | | | | | | Christmas |
| 37<br>4/1/93 | 74 | 19 | * | * | | | | * | * | * | * | | * | | * | |
| 38<br>11/1 | 81 | 46 | * | * | | | | * | * | * | * | * | * | | * | Database JAD |
| 39.<br>18/1 | 66 | 72 | * | * | * | * | * | * | * | * | * | * | * | | | Monthly meeting; Car6 #1 strip; Quick-Store vs. Number Crunch |
| 40.<br>25/1 | 66 | 81 | * | * | | | | * | * | * | * | * | * | | | |
| 41 1/2 | 48 | 30.5 | * | * | | | | * | * | * | * | | * | | | the author reverts to half-time working |
| 42 8/2 | 35.5 | 67 | * | * | * | * | * | * | * | * | * | * | * | | | Monthly meeting |

| Effort | | | Staff Participation | | | | | | | | | | Tool Usage | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Effort (staff hours) | | Database Developers | | Senior Management | | | Recyclate Ltd. Data Collectors | | | Data Users | | Tool usage | | | |
| Week commence. | Project staff | Users | Author | Mary | Terry | Accoun-tant | Environ-mental Strategist | Robert | Jim | Ron-nie | Geoff | Chris | Quick Store | Analy-ser | Number Crunch | Events |
| 43<br>15 2 | 47.5 | 12.5 | * | * | * | | | | * | * | * | | * | * | | Process model-ling to "sell" database work |
| 44<br>22 2 | 40 | 16 | | * | | | | * | | | * | * | * | | | |
| 45 1/3 | 54 | 11 | * | * | * | | | * | | | * | * | * | | * | PRF system intro-duced; start work on stripping curves |
| 46 8/3 | 47 | 31 | * | * | * | * | * | * | * | | * | * | * | * | * | Monthly meeting |
| 47<br>15/3 | 47 | 13 | * | * | | | | * | * | | * | * | * | | * | |
| 48<br>22/3 | 49 | 6 | * | * | | | | * | | | * | | * | | | |
| 49<br>29/3 | 56 5 | 17 | * | * | | | * | | | | * | * | * | | * | First "Manage ment Report-ing" meeting |
| 50 5/4 | 8 | 25 | * | | * | * | * | * | * | | * | * | * | | * | Monthly meeting |

| Effort | | | Staff Participation | | | | | | | | | | Tool Usage | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Effort (staff hours) | | Database Developers | | Senior Management | | | Recyclate Ltd. Data Collectors | | | Data Users | | Tool usage | | | |
| Week comm ence. | Project staff | Users | Author | Mary | Terry | Accoun -tant | Environ- mental Strategist | Robert | Jim | Ron- nie | Geoff | Chris | Quick Store | Analy- ser | Number Crunch | Events |
| 51. 12 4 | 25 | 7 | * | * | | | | * | | * | | * | * | | * | |
| 52 19 4 | 53.5 | 2 5 | * | * | | | | * | * | | | | * | | * | |
| 53. 26 4 | 45 5 | 17 | * | * | * | * | * | * | | | * | * | * | | * | Second "Manage ment Report- ing" meeting |
| 54. 3/5 | 24 | 1 | * | * | | | | * | | | | | * | | * | |
| 55. 10 5 | 53.5 | 41 | * | * | * | * | * | * | * | * | * | * | * | | * | Monthly meeting |
| 56. 17/5 | 58.5 | 8.5 | * | * | * | | | * | * | | * | | * | | * | |
| 57. 24/5 | 46.5 | 19 | * | * | * | | * | * | * | * | | * | * | | * | Third "Manage ment Report- ing" meeting |
| 58. 31/5 | 18 | 9 | * | | | | | * | | | * | | * | | | |
| 59. 7/6 | 29 | 12 | * | * | | | | * | * | * | * | * | * | * | | |
| 60. 14/6 | 39 | 26 | * | * | * | | | * | * | | * | * | * | | | |
| 61. 21/6 | 18 | 27 | | * | * | * | * | * | * | * | * | * | * | | | Monthly meeting |

| Effort | | | Staff Participation | | | | | | | | | | Tool Usage | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Effort (staff hours) | | Database Developers | | Senior Management | | | Recyclate Ltd. Data Collectors | | | Data Users | | Tool usage | | | |
| Week comm ence. | Project staff | Users | Author | Mary | Terry | Accoun -tant | Environ- mental Strategist | Robert | Jim | Ron- nie | Geoff | Chris | Quick Store | Analy- ser | Number Crunch | Events |
| 62. 28 6 | 43.5 | 54 | * | * | | | | * | * | * | * | * | * | | | Mater- ials meeting at Research Centre |
| 63 5/7 | 5 | 2 | * | | | | | | * | | | | * | | | |
| 64. 12/7 | 40 | 4 | * | * | | | | * | * | | * | | * | | | |
| 65 19/7 | 46.5 | 10 | * | * | | | | * | * | * | | | * | | | |
| 66 26/7 | 38 | 20 | * | * | * | * | * | * | * | | * | * | * | | * | Monthly meeting |

## 4.2 Defect and Change Request Analysis: The PRF System

The Problem Report Form (PRF) system was introduced by the author on 1 March 1993 in order to gain some control over the chaos threatening to engulf database development. The volume of change requests and defects was overwhelming the author and Mary, who were losing track of problems to be solved and becoming confused over who was working on which problem and how far resolution had progressed. The PRF form was based on the PSR form used by CarMaker to track problems arising on prototype vehicles; the system that the author had implemented at South Factory. A total of thirty problems were recorded from 1 March to 30 July, when detailed data collection for CASE research purposes ceased. A summary of the problems is presented in the following table.

| PRF number | Date raised | Description | Defect or Change | Priority (H=High; M=Medium; L=Low) |
|---|---|---|---|---|
| 1 | 1/3/93 | Restructure part name glossary to improve lookup speed and ease maintenance. | C | M |
| 2 | 1/3/93 | Minor database changes e.g. restructure user menu. | C | L |
| 3 | 1/3 93 | Replace Bin/Material data entity with two separate entities: Bin and Material. | C | M |
| 4 | 1/3 93 | Revised requirements for materials reports from Chris, using new Material entity. | C | M |
| 5 | 1/3/93 | Develop Net Process Value report for Geoff. | C | M |
| 6 | 1/3/93 | Develop list of IS operating instructions e.g. advice on how to avoid misuse of F8 (MODIFY) key. | C | M |
| 7 | 1/3/93 | Enter data backlog into QuickStore from NumberCrunch and paper records. | C | M |
| 8 | 1/3/93 | Develop stripping curves using QuickStore for report generation and NumberCrunch for graph production. | C | M |
| 9 | 1/3/93 | One of Terry's assistants to evaluate the suitability of leading Windows database as an alternative to QuickStore. | C | M |
| 10 | 1/3/93 | One of the author's MSc. students to evaluate QuickStore for Windows as alternative to QuickStore v2 (DOS). | C | M |
| 11 | 1/3/93 | Add subtotals and totals to all reports "as appropriate". Request from the Accountant. | C | L |
| 12 | 1/3/93 | Devise procedure to synchronise data collection, report generation and software development across the un-networked PCs. | C | M |
| 13 | 1/3/93 | Data import problems from NumberCrunch to QuickStore. | D | M |
| 14 | 3/3/93 | Change "materials" fields in Materials form from choice field to lookup field as number of materials exceeds 99 (max. choice limit). Bin Id remained a choice field and caused many problems due to difficulty in maintaining multiple copies of the Bin list. | C | M |
| 15 | 3/3/93 | Split the combined Car6 and Car7 database into two separate databases. | C | H |

| PRF number | Date raised | Description | Defect or Change | Priority (H=High; M=Medium; L=Low) |
|---|---|---|---|---|
| 16 | 4 3 93 | Develop utility to sort the part name glossary due to lack of facility in QuickStore to automatically view records in chosen sequence. | D | H |
| 17 | 5/3 93 | Add Deficiency and Extras forms to the Copy Trial utility. | C | M |
| 18 | 5 3 93 | Add Aggregate entity to store #3 car (optimal strip) activity times. | C | M |
| 19 | 12/3/93 | Investigate propagation of process number to level 2 operations and lower when resequencing processes. | D | H |
| 20 | 15 3/93 | Check selection criteria used in reports and form triggers for errors e.g. make sure "materials" parts are ignored in valuation reports. | D | M |
| 21 | 23/3/93 | Develop sophisticated financial analysis "what if" reports for Geoff. | C | M |
| 22 | 26/3/93 | Cascade material default bin to part records to enable above reports to work (see PRF# 21). | C | M |
| 23 | 8/4/93 | Investigate data collection response time problems at Recyclate Ltd.. | D | H |
| 24 | 8/4/93 | Further work on PRF# 23. | D | H |
| 25 | 26/4/93 | Further development of the stripping curves. | C | M |
| 26 |  | PRF Missing. |  |  |
| 27 | 5/5/93 | Various maintenance tasks to align Recyclate Ltd. databases with Research Centre versions e.g. synchronise reports supported. | C | L |
| 28 | 5/5/93 | Include level 2 and lower operations on Sequenced Worklist report. Had been missed out because process number not propagated to lower level operations (see PRF# 19). | D | H |
| 29 | 5/5/93 | Add "Wanted" flag to denote parts to be removed on #2 and #3 cars. | C | M |
| 30 | 30/7/93 | Investigate generation of duplicate records when F2 (SAVE) hit accidentally instead of F8 (MODIFY). | D | H |

## 4.3 Recycling Project Stakeholder Views: UIS Survey Responses

The author issued the User Information Satisfaction (UIS) questionnaires to the eight stakeholders during the spring of 1993. He had intended to survey user views at several stages during the project, but delayed undertaking the first survey until criticism of the database became less fierce, for fear of damaging his own and Mary's morale! User reaction to the survey was mixed, some felt it an unnecessary exercise (Robert in particular). In view of the lack of enthusiasm and the author's own involvement in other activities by the summer, a second survey was not carried out. Users were asked to answer twelve questions on the system development process and twenty-one questions on the end product, the Recycling IS. Responses were recorded using a 5-point Likert scale, with values of +3 (very satisfied), +1, 0, -1 and -3 (very dissatisfied). A question could be skipped if not applicable to the respondent. Respondents were also asked to add comments and make suggestions for improvement if they wished. Six completed questionnaires were returned. Two of the management group, the Accountant and the Environmental Strategist, did not respond despite follow-up requests from the author. A summary of the UIS scores is given below.

| Stakeholder | Date Questionnaire Completed | Mean score for the system development process (n=12) | Mean score for the IS product (n=21) |
|---|---|---|---|
| Geoff | 22/3/93 | 1.18 (n=12) | 1.65 (n=20) |
| Chris | 13/4/93 | -0.5 (6) | -3 (1) |
| Ronnie | 14/5/93 | -1.64 (11) | -1.5 (18) |
| Jim | 14/5/93 | -1.66 (12) | -1.2 (20) |
| Robert | 19/5/93 | 1.66 (12) | 0.8 (21) |
| Terry | 19/6/93 | 0 (12) | -1.1 (10) |

## Major Issues

The major issues raised by each respondent on the returned questionnaires are presented below.

1. Geoff - The Recycling Economic Model Developer (22/3/93).

Geoff felt that his participation in development and understanding of the database had improved considerably since Christmas and that the IS staff (the author and Mary) were supporting him well and their relationship improving after a "rough ride". He had little idea of the size and scope of his information requirements at the start of the project. He felt the IS was reliable and the information produced was relevant, precise and complete. Usability was "eccentric", but he felt confident in using the system and developing his own reports. Overall he was very satisfied with the IS.

2. Chris - The CarMaker Materials Engineer (13/4/93).

Chris felt the developers had not involved him sufficiently, although he had a good relationship with them. He was still unable to use the database and thought the user interface unfriendly. Undoubtedly more resources should have been directed at this "corner stone" of the project. Finally, he had derived no benefits from the IS to date.

3. Ronnie - The Recyclate Ltd. Disassembly Engineer (14/5/93).

Ronnie was not a direct user of the database, his job was to strip the cars, however he did make use of the reports and was a vociferous critic of what he saw as an unusable system. His comments were as follows. He understood well the capabilities of the IS, and felt that it failed to meet his expectations of a "20th century environment" (i.e. Windows-based). He observed that the users had difficulty entering and modifying data, making many errors. He felt the report layout was poor and the documentation non-existent. Finally, he felt he had no control over IS development although his relationship with the developers was satisfactory.

4. Jim - The Recyclate Ltd. Information System Operator (14/5/93).

Jim felt his level of participation in IS development was satisfactory, but that he didn't understand the IS capabilities. The IS was seen as very inflexible and failed to meet his expectations. His relationship with the developers was adequate but "permanent on-site support was needed" to improve the IS. The system should be upgraded to a Windows version as soon as possible and a QuickStore consultant called in to resolve the technical deficiencies. Error recovery was poor, Jim couldn't generate his own reports and documentation was non-existent.

5. Robert - The Recyclate Ltd. Special Projects Manager (19/5/93).

Robert was very satisfied with his participation and understanding of the database. He felt the system was very flexible, but that more on-site developer time was needed "... to develop an IS without a spec." Overall he was satisfied with the development process although somewhat frustrated. The information delivered by the IS was highly praised, being very accurate, precise, complete, and readily available. He now felt the response time was very good. Data entry was straightforward but data modification very difficult: "... needs a lot of thought". The IS was difficult to learn and explanation of the data model would have helped. Overall he was satisfied with the IS.

6. Terry - The CarMaker Project Manager (19/6/93).

Terry felt that everyone involved in the IS should have spent more time "up front" in order to understand the information requirements of the project. His perception of the IS was that it was inflexible, although he had not used it himself. He felt he had a very good relationship with the IS developers (!), but was surprised at the amount of resource IS development had consumed. His views on usability were again drawn from an overall perception of problems, rather than from direct usage himself. He perceived that the IS was unreliable, difficult to use and had a poor response time. He completed his questionnaire by stating that the IS

> *"... seems to have been the single largest problem on the project".*