Capture and Integration of Experience from Automotive Manufacturing

A thesis submitted for the degree
of Master of Philosophy of
the Australian National University

Jeremy Ingle Smith

September, 2008

Declaration

This thesis contains no material which has been accepted for the award of any other degree or diploma in any university. To the best of the author's knowledge and belief, it contains no material previously published or written by another person, except where due reference is made in the text.

Jeremy Ingle Smith
April, 2007

Abstract

Stamping operations within automotive manufacturing organisations are reliant on experienced personnel and skilled artisans to ensure that the design of a component is effectively and efficiently reproduced during production. This includes trial and error alterations in tryout and problem-solving during production.

However, with shortening lead times to meet customer demands and a loss of experienced personnel through out-sourcing, right-sizing and a more flexible work force, the experience required to overcome these concerns is being lost by manufacturing organisations. This results in longer lead times, a decrease in quality and an increase in tryout and maintenance costs. This knowledge loss is particularly significant on the shop floor, where tacit experience is vital during problem-solving.

Based on a detailed domain analysis, a set of design principles for a system to capture this experience was established. A software application to satisify these principles was developed using a bottom-up approach and a multi-disciplinary team. It was implemented in an automotive stamping facility and the results of a two year case study analysed and evaluated.

It was found that the capture interface developed which paralled the natural problem-solving approach of its users was particularly useful for capturing experience in manufacturing operations. The system was used to ensure problems arising during die design and tryout are not repeated and improvements incorporated into the product lifecycle to help decrease lead time and rework costs.

Acknowledgements

There are so many people to thank and acknowledge in a work such as this that has covered a long period, with significant involvement from our industry partners and so much time on-site. I will make an attempt here to cover as many as possible.

To all those in Melbourne and Geelong who helped with the development and implementation of the system, by providing their time, comments, feedback and assistance. In particular those who took on board the ideas of the project and championed the systems development, especially Trevor Bennion, Ron Imase and Graeme Jacobi.

To everyone in the Engineering Department for making it such a great place to work and for all the help during my research. There are too many to mention, but for anyone who has distracted me on the kicker table, over a pint, or on the sporting field, thanks. To mention by name, Rob for all his professional and personal advice, Victor for all his time and efforts in ensuring this work could succeed, and Matt for his help and time. Finally to Mick for his ongoing support, advice, encouragement, trust, and for continuing to provide opportunities.

To all those who provided wonderful company, and so often a place to stay, in Melbourne and Geelong particularly Liz and Paul, Mozza, Kate and Pete, Siria and Bernie, and P. You all helped to make my time and work in Melbourne and Geelong more enjoyable than it probably should have been, and it will always be remembered and appreciated.

To my fellow CABities, cheers. To old friends and new, thanks to all...

Finally, but not lastly, for all the patience, understanding and support of my family, it is very much appreciated, particularly during the many times away.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Over countless generations craftsmen and artisans who worked with metal were held in the highest regard by their communities and peers. Evolved from those early days, sheet metal forming or stamping, has today become one of the most widely used manufacturing processes. It provides a relatively inexpensive process well suited to the mass production of thin metal sheet components (Tang, D.-B. et al., 2001; Cheok, B.T. and Nee, A.Y.C., 1998). By forming, or stamping, a metal blank between two dies the metal takes on the shape of the dies. Its application and use have grown significantly over the last 100 years due to its suitability for the production of components and panels for the automotive industry and other mass produced consumer markets.

Despite recent advances in equipment, processes, and computer-based aids, the production of a quality stamped component still remains an experience-based process, particularly in the key areas of tool design and die try-out (Blümel, K. W. et al., 1988; Dinda, S. et al., 1981; Keeler, S. P., 1977; Thomas, W. and Altan, T., 1998; Leake, D. B. et al., 1999a; Thomas, W. and Altan, T., 1999). In the last 20 years, numerous computer-aided tools have been developed to aid the part and die designer in such tasks, particularly Computer Aided Design (CAD) tools and more recently Finite-Element (FE) simulation packages ((Thomas, W. and Altan, T., 1998). However, the shop floor areas that represent the end-users of designs where the dies are made, tested, and used have commonly been neglected in such developments. The skilled artisan is still required during these activities to complete the transformation of the design for a die into a tool that can correctly and repeatedly produce the required component over possibly years and hundred-of-thousands of hits (Keeler, S. P., 1977; Stine, P. A. and Keeler, S. P., 1990).

The knowledge they apply during this process is built-up through apprenticeships and years of accumulated experiences, learnt from numerous mistakes and trial and error situations that particularly occur during die design and tryout (Aboutour, T. et al., 1992; Huang, S. H. et al., 2001; Liu, H. et al., 2002; Stine, P. A. and Keeler, S. P., 1990; Thomas, W. and Altan, T., 1998; Tang, D.-B. et al., 2001; Thomas, W. and Altan, T., 1999). It is here that concerns with designs become apparent and need to be overcome (Leake, D. B. et al., 1999a). This knowledge often takes the form of informal tacit knowledge from experienced shop floor employees and usually remains only with the individual (Leake, D. B. et al., 1999a; Huang, S. H. et al., 2001; McEvily, S. K. and Chakravarthy, B., 2002). If such knowledge is lost, inevitably there is a repetition of mistakes, a decrease in quality, an increase in scrap and additional time is required to fine-tune or rework tools. Knowledge loss can result from experienced personnel retiring, voluntary separations, corporate right-sizing, out-sourcing of tasks or simply not capturing the experience gained from day-to-day problem-solving activities (Ndlela, L. T. and du Toit, A. S. A., 2001).

Knowledge loss of this type is recognised as a significant problem affecting common manufacturing processes, including sheet metal forming (Leake, D. B. and Wilson, D. C., 1999; Chapman, C. B. and Pinfold, M., 2001; Tang, D.-B. et al., 2001). Retaining and making available the knowledge represented by the employees within a manufacturing organisation is seen as the way to maintain a competitive advantage in the current manufacturing industry (Dutta, S., 1997; Noh, J. B. et al., 2000; Damodaran, L. and Olphert, W., 2000; Wagner, K. and Aslanidis, S., 2001; Ndlela, L. T. and du Toit, A. S. A., 2001; McEvily, S. K. and Chakravarthy, B., 2002). However, this experience has often been difficult to elicit, record and reuse, due to its type and the manufacturing shop floor environment where it is generated. Shortening lead-times have compounded these issues by decreasing the amount of time in which information and knowledge can be recorded either during the new product developments or post-launch reviews (Affuso, T., 2005).

1.2 Research Aims

Capturing this experience in a structured manner would lead to the construction of an organisation-wide experience base which could be utilised and accessed by all activities within an organisation. If the informal, tacit experiences from the shop floor were captured they could be used, not only by future generations of toolmakers, but be fed back upstream into the design areas to provide comment on the success or otherwise of the manufacturability of the completed designs (Stine, P. A. and Keeler, S. P., 1990). In addition this has the potential to reduce die costs and shorten lead-times (Tang, D.-B. et al., 2001; Fakun, D. and Greenough, R. M., 2002).

In the context of stamping, a framework and system is required that will capture the experience, particularly tacit knowledge, from individuals on the shop floor who implement and trial die designs. The aim of this research is to design and develop a software system to capture and integrate this information into the design process, to aid its reuse for future design and tryout activities and to contribute to the construction of a corporate knowledge base. Such a system must effectively capture experience from the manufacturing shop floor to be utilised in future problem solving or design activities, to reduce the number and more effectively overcome issues occuring on the shop floor in tryout and production, to improve quality and shorten lead-times.

1.3 Scope and Objectives

This study aims to design, develop, implement and evaluate a software system to capture experience from shop floor activities for automotive sheet-metal forming. The system needs to capture information and experience that is otherwise not recorded or is lost when experienced personnel leave, to assist problem-solving and die design in order to improve product quality, manufacturing productivity and lead-times. To achieve these aims, this thesis will consider the following activities:

- requirements identification
- domain analysis
- identification of design principles
- high-level system design
- software development
- an implementation case study at a single site

A number of different fields need to be considered and examined, including:

- the manufacturing product lifecycle
- sheet metal forming
- the shop floor environment
- knowledge acquistion, and management tools and techniques
- software implementation and acceptance

Detailed non-technical and management strategies to encourage on-site use during the implementation phase are not covered here, but rather the focus is on strategies incorporated into the design and development of the system.

An action-based research approach was utilised, with the researcher spending time embedded on-site during the research work. This was to understand the context and detail requirements.

1.4 Outline

The remainder of this thesis describes the research undertaken. Chapter two provides a background to the work and an overview of sheet metal forming, the plant environment and the product lifecycle. It examines published work in stamping and other manufacturing processes including their focus and limitations, to be used as the base for requirements identification and domain analysis activities.

The approach taken for the design and development of the system is outlined in chapter three, in addition to a discussion of work on software development methodologies. This provides the background for the software development activities.

The design principles and high-level system design framework for the software system, based on research and the domain analysis, are developed in chapter four.

A case study describing the development and implementation of a system encapsulating the design principles and framework for an automotive stamping plant is presented in chapter five.

Results from the case study, an evaluation of the system, and lessons learnt from both the development and implementation are given in chapter six.

Chapter seven then summarises the work, outlines potential extensions and applications to other industries and re-examines the research aims.

Chapter 2

Background

An outline of stamping, the product lifecycle and the shop floor are described in order to highlight and identify the causes of knowledge loss. A brief description of data, information and knowledge is presented, leading into a discussion of existing knowledge-based systems, and previous work aimed at overcoming knowledge loss and capturing and managing corporate knowledge. The limitations of these when applied to the stamping shop floor are stressed in addition to non-technical considerations that influence and impact upon the development and introduction of new technology.

2.1 Stamping

Forming is a subset of net-shape manufacturing, of which metal stamping is a widely used process for producing cost-effective metal components. A material is plastically deformed using a set of dies or tools, to impart the final, or near-final, shape of the component. Stamping is used to form different types of products for automotive, aerospace, marine, home and packaging applications.

It is beyond the scope of this work to provide a detailed description of the mechanics of stamping, but a summary is given to highlight its importance and complexity.

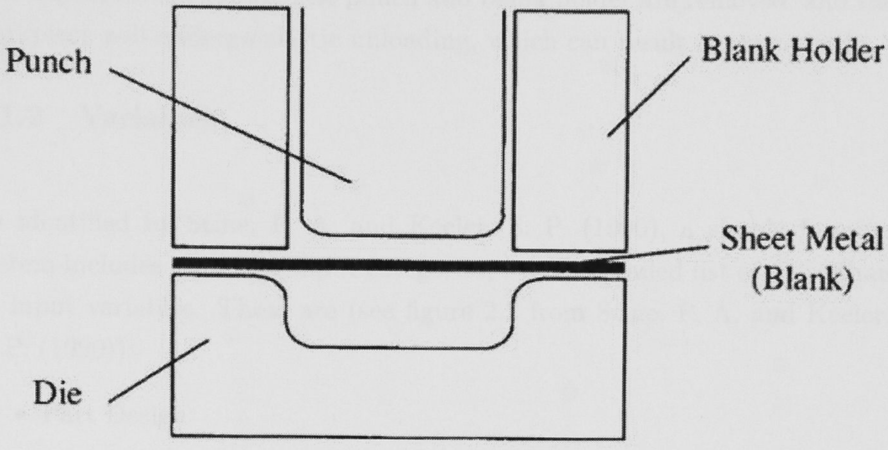

Figure 2.1: Components for a typical stamping operation

2.1.1 Stamping System

Within the automotive industry, deep drawing, stamping and shearing are among the most common operations although a broad range of techniques, collectively termed stamping, will be considered here. A typical stamping system consists of the following components (see figure 2.1, based on Cao, J. and Boyce, M. C. (1997)):

Die the cavity containing the shape of the desired component

Punch to match the die cavity

Blank Holder or binder, which holds the workpiece (blank) in place

Sheet Metal or blank, the metal workpiece being stamped

To form a completed component, the blank is placed between the die and the punch. As the drawing operation starts, the blank holder and the punch are lowered. The blank is restrained by a frictional force between the die and the blank holder. The punch continues to move downwards to draw the blank and plastically deform the blank into the required shaped (Cao, J.

and Boyce, M. C., 1997). The punch and blank holder are removed, and the workpiece will undergo elastic unloading, which can result in springback.

2.1.2 Variables

As identified by Stine, P. A. and Keeler, S. P. (1990), a simple forming system includes six major input categories, from a detailed list of more than 35 input variables. These are (see figure 2.2 from Stine, P. A. and Keeler, S. P. (1990)):

- Part Design
- Die Design
- Material
- Lubricant
- Press
- Operator

These inputs combine within the forming operation to generate a finished stamped component. During a forming operation, a number of problems can arise. Common concerns initially involve splitting or buckling, which result respectively from either too much restraining force or not enough. Other problems can result in deviation from expected dimensional tolerances, which become significant during subsequent use and assembly, and surface quality issues, which are particularly important for parts which are visible to the consumer (Zhang, Z. T. et al., 1999). Springback can occur when blank holder pressure is released, and this can also affect the dimensional accuracy of a component. Costly tryout and a reliance on experts and artisans within sheet metal design and manufacture in the automotive industry still occurs due to the complexity and number of variables involved and the balance that must be achieved between form and function.

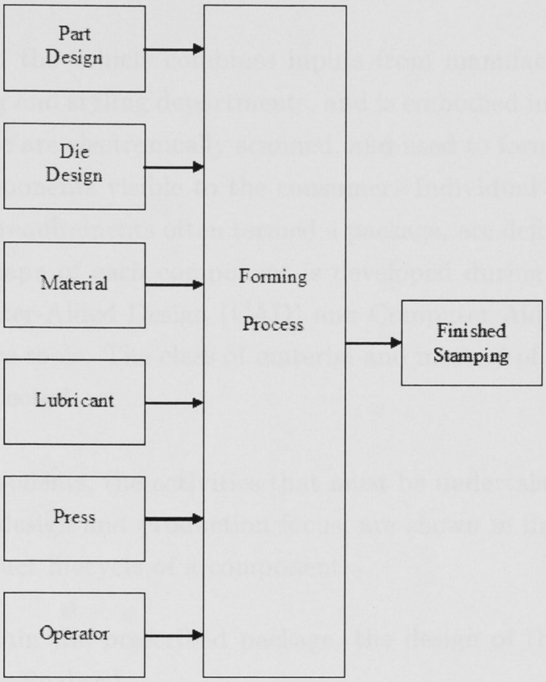

Figure 2.2: The forming system

2.2 The Product Lifecycle

The design and manufacture of sheet metal components require a number of activities. Within the automotive industry, these are often focused on the launch of a new vehicle model program. As identified by Vinarcik, E. J. (1999) a typical automotive product development cycle consists of:

1. Research and Development
2. Concept or Advanced Engineering
3. Product Design
4. Process Design
5. Process Planning
6. Product Launch
7. Production

The final shape of the vehicle combines inputs from manufacturing, engineering, marketing and styling departments, and is embodied in one or more clay models. These are electronically scanned, and used to form the basis of the shape for components visible to the consumer. Individual components, each with a set of requirements often termed a package, are defined from the clay. The final shape of each component is developed during product design using Computer-Aided Design (CAD) and Computer-Aided Engineering (CAE) software tools. The class of material and method of manufacture for each part is selected.

For stamped components, the activities that must be undertaken for manufacturing, from a design and production focus, are shown in figure 2.3, and make up the product lifecycle of a component.

Part Design within the prescribed package, the design of the individual component is finalised

Process Design the sequence of operations, to make each part is developed, in addition to initial strain targets

Tool Design the individual die set for each operation is designed

Die Build each die set is constructed

Die Tryout completed tooling undergoes tryout to ensure it meets its requirements and tolerances. Alterations may be made to dies to ensure individual parts are of the required quality

Production tools are passed into production for ongoing use

Die Maintenance ongoing maintenance and further modifications of dies to ensure quality and extend their lifetime

These activities can be divided into two. Part, Process and Die Design activities make up the design portion, and are generally performed by engineers, with a high level of computer literacy, using CAD and CAE tools. FE simulation software systems are also used during die design to assess

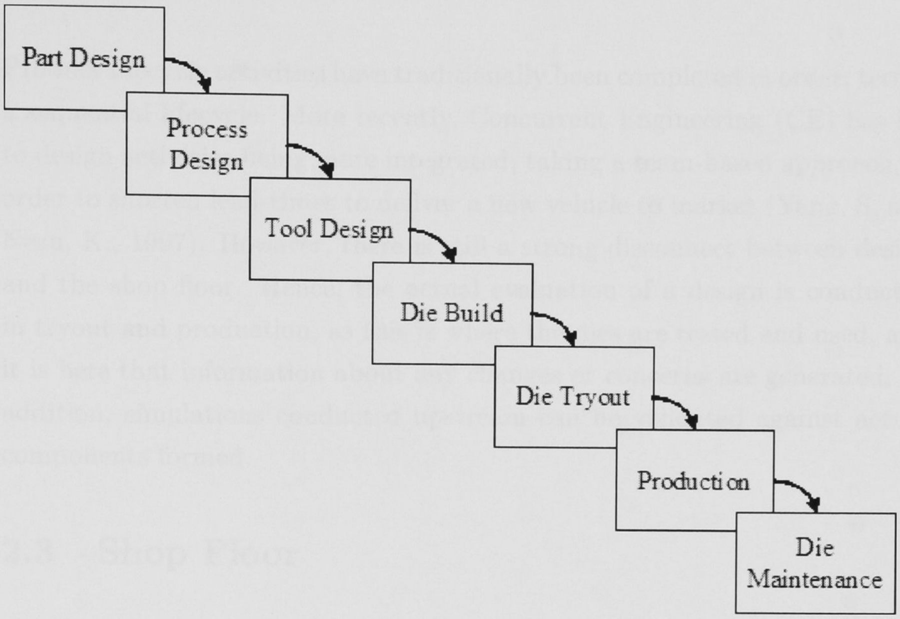

Figure 2.3: Manufacturing product lifecycle activities

potential production issues. Production consists of the remaining activities, and are undertaken on the shop floor in an industrial environment.

The time spent on each activity can vary from part to part and organisation to organisation. Cao, J. and Boyce, M. C. (1997) state that on average 25% of the die manufacturing time is spent on tryout, using a trial and error approach to fine-tune the tools to ensure the completed component is free of defects. Further time may then be spent in production to ensure acceptable parts can be produced at required rate, as factors of die and press conditions and material and lubrication variations become significant (Zhang, Z. T. et al., 1999). Die design itself takes 20% of the complete lead-time within the product lifecycle for an individual component (Fallböhmer, P. et al., 1996). As a product or part moves through its lifecycle, the number of noise factors or uncontrollable factors that can cause problems increases, while the number of variables that can be altered, the control factors, decreases (Vinarcik, E. J., 1999). Importantly, the cost to overcome problems increases as the product moves through the activities. Typically, the cost to correct a problem increases by a factor of ten for each activity from concept where the error is identified (Kalpakjian, S. and Schmid, S., 2006).

Product lifecycle activities have traditionally been completed in order, termed a sequential lifecycle. More recently, Concurrent Engineering (CE) has led to design activities being more integrated, taking a team-based approach, in order to shorten lead-times to deliver a new vehicle to market (Yang, S. and Nezu, K., 1997). However, there is still a strong disconnect between design and the shop floor. Hence, the actual evaluation of a design is conducted in tryout and production, as this is where the dies are tested and used, and it is here that information about any changes or concerns are generated. In addition, simulations conducted upstream can be validated against actual components formed.

2.3 Shop Floor

As described previously, product lifecycle activities for sheet metal can be broadly grouped into design and production, where the latter is carried out in shop floor areas. This environment and its approach to overcoming design and production issues are described in the following sections.

2.3.1 Environment

Design areas of stamping organisations are characterised by salaried staff, often with engineering backgrounds and high computer literacy, using CAD/ CAE packages and increasingly FE software tools. In contrast, the manufacturing shop floor is an industrial environment and has large numbers of shift workers with trade backgrounds and lower computer literacy (Huang, S. H. et al., 2001; Pantano, V. et al., 2002). Shop floor employees are often long-term employees of an organisation who have accumulated experience and learnt from numerous mistakes and trial and error situations (Huang, S. H. et al., 2001). This experience is often with previous dies and components and is employed when overcoming quality or process concerns that arise during production with new dies.

2.3.2 Problem-Solving

As the inputs into the forming process are continually changing due to economic, marketing and engineering factors, new issues which need to be resolved are constantly occurring. As listed by Stine, P. A. and Keeler, S. P. (1990) and Vinarcik, E. J. (1999), the problems identified on the shop floor during the forming process in tryout, production and maintenance have traditionally been overcome by a number of methods:

- trial and error, by manually altering the shape of a die and punch until the required part is formed correctly
- evaluation of postmortem measurements, typically of strain distributions and forming limit diagrams (FLDs) in order to determine an indicative forming severity (Keeler, S. P., 1988; Zhang, Z. T. et al., 1999)
- experience from previous similar problems
- process measurements, such as signature analysis during a press stroke
- prescribed problem-solving methodologies, such as 8D and the Shewhart Cycle (Vinarcik, E. J., 1999)

Finite Element (FE) simulations are now commonplace to assist designers to predict formability of designs (Thomas, W. and Altan, T., 1998). However, there is often no formalised feedback from the shop floor back to design areas concerning the accuracy of models. Once in tryout and production there are further variables to consider, such as press and material variability, lubrication and process control (Thomas, W. and Altan, T., 1998; Zhang, Z. T. et al., 1999). At the same time the number of variables which can be changed to overcome a problem decrease. This relationship between variables that can be controlled and those that impact on the process is shown over a typical product lifecycle in figure 2.4 (from Vinarcik, E. J. (1999)).

Hence, as the part lifecycle continues, the more an individual's experiences with previous issues are utilised to overcome concerns that arise during

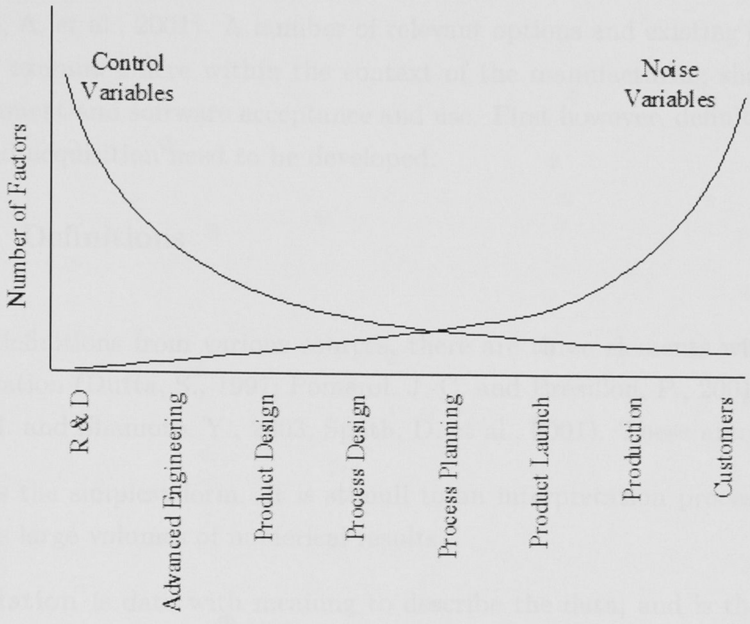

Figure 2.4: Input noise and control variables during the product lifecycle

the development of new components. Experienced, long-term shop floor personnel are vital to problem-solving, applying their informal, tacit-based knowledge to overcome issues. When such experienced personnel leave, their experience is lost with them, along with the potential for problem-solving. It is this knowledge loss which needs to be addressed.

2.4 Overcoming the Loss – Knowledge Management

Knowledge Management (KM) tools and techniques can be used to address issues of knowledge loss and ensure that an organisation makes maximum use of the experience of its employees (Lau, H. C. W. et al., 2003; Wagner, K. and Aslanidis, S., 2001; Stine, P. A. and Keeler, S. P., 1990). A number of tools, techniques and methodologies have been developed and applied to overcome concerns with knowledge loss in stamping and other similar environments. To be effective a KM system must address the twin questions of how to capture and then represent the knowledge within an organisation

(Cañas, A. et al., 2001). A number of relevant options and existing systems will be examined here within the context of the manufacturing shop floor environment and software acceptance and use. First however, definitions for KM and acquisition need to be developed.

2.4.1 Definitions

Using definitions from various sources, there are three elements within an organisation (Dutta, S., 1997; Pomerol, J.-C. and Brézillon, P., 2001; Yoshioka, M. and Shamoto, Y., 2003; Spath, D. et al., 2001). These are:

data is the simplest form. It is stimuli to an interpretation process, such as large volumes of numerical results

information is data with meaning to describe the data, and is the input into knowledge-based decision-making processes

knowledge is the ability to utilise information for decision-making

Within these broad definitions, there are different types of information and knowledge (Pomerol, J.-C. and Brézillon, P., 2001; Wagner, K. and Aslanidis, S., 2001):

know-how is knowledge used for operations or behaviour, often expressed in routines or unconscious actions

procedural knowledege is expressed by rules or sequences of activities, such as if-then rules

surface knowledge can be represented by practical rules which can be expressed by experts on a given task

explicit knowledge can be well described and expressed, and hence can be recorded, structured and standardised, and shared relatively easily

tacit or implicit knowledge is highly personal, individual knowledge which can be difficult to formalise or express, and often formed though the experience of circumstances or situations

Using the definitions from Wagner, K. and Aslanidis, S. (2001) and Dutta, S. (1997), the information and knowledge within a company can be further defined:

company-wide information static content typically administered centrally, for example design folders and static html (web) pages

company-wide communication the decentralisation and dynamic provision of information, especially regarding its processing, such as employee networks

company-wide cooperation integration of applications and workflows

company-wide knowledge networking a combination of personal, context- and activity-based provision of knowledge

2.4.2 Acquisition and Representation

As highlighted by Noh, J. B. et al. (2000) and Lapré, M. A. and Van Wassenhove, L. N. (2001), tacit knowledge is a source of sustainable competitive advantage, but this is also the most difficult to formalise. The more integrated and the closer to 'knowledge' (as opposed to information or data) this experience, the more value an organisation will be able to derive from it. This is also called the 'knowledge maturity' of an organisation (Wagner, K. and Aslanidis, S., 2001).

Two extremes have been identified for acquisition and representation tasks (Leake, D. B. and Wilson, D. C., 1999; Cañas, A. et al., 2001). The first concentrates on capturing information in a structured and standardised form, to achieve detailed knowledge representation. This allows for greater automated use of the knowledge in processing and reasoning activities, but requires significant time and training to enable a user to record their knowledge. At the other extreme knowledge acquisition is as straight forward as possible, in its simplest form consisting of free text descriptions. This removes the burden on the user entering new information, but can reduce the reusability of the knowledge acquired as the underlying principles and

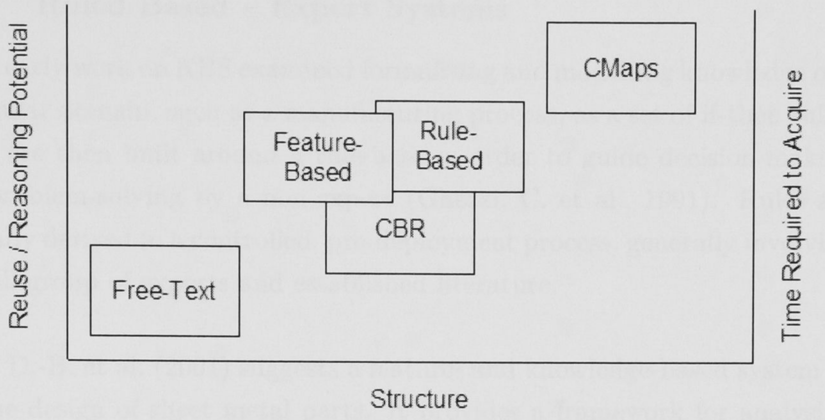

Figure 2.5: Manufacturing product lifecycle activities

concepts may not be captured (Cañas, A. et al., 2001). These extremes, and their approaches, are shown in figure 2.5.

There have been many studies which have developed KM tools and systems on this scale to acquire knowledge and overcome knowledge loss in stamping and other related manufacturing processes. However, few have focused on the critical areas of the shop floor. Work describing existing systems and techniques for part and tool design and production will be examined, covering not only stamping and other net-shape manufacturing processes but other industries.

2.5 Knowledge-Based Systems

Knowledge-Based Systems (KBS) are software applications designed to make use of domain specific knowledge to solve problems and support decision-making processes (Dutta, S., 1997). These can include expert systems, corporate databases, organisational memory systems, information storage and retrieval tools and a range of other systems, the most relevant of which for this study are described below.

2.5.1 Ruled Based – Expert Systems

Much early work on KBS examined formalising and modelling knowledge of a particular domain, such as a manufacturing process, as a set of if-then rules. Tools are then built around a rule-base in order to guide decision-making and problem-solving by a non-expert (Ghezzi, C. et al., 1991). Rules are typically derived in a controlled, pre-deployment process, generally involving a small group of experts and established literature.

Tang, D.-B. et al. (2001) suggests a feature- and knowledge-based system to aid the design of sheet metal parts. It provides a framework for analysing and describing components as a set of features, which then interact. Each feature is associated with a particular operation. Such knowledge is classified into two groups:

evaluation knowledge is the set of rules and guidelines used for evaluation

control knowledge responsible for planning and monitoring of evaluations

Both types are represented by production rules, expressed in antecedent-consequent or if-then form. Formability is evaluated by part shape analysis and cost estimation, both based on the feature representations defined. Evaluation knowledge is derived from published literature in addition to interviews with relevant experts.

A system to assist preliminary part and process design was developed in Yang, S. and Nezu, K. (1997). This approach, and the system to support it, focused on support for concurrent design through an integrated design system combining numeric simulations, graphical visualisation and knowledge representations. This applied an inverse modelling approach to give an initial indication of forming feasibility during preliminary design. The results of application-specific simulation were also incorporated into the knowledge base, as knowledge relating to component and process design. Around these were production rules, in the form of condition-action pairs (if-then rules). These represent action-oriented knowledge, and are based on empirical rules sourced from literature, industrial practice and experimentation. These are applied to part models in order to assist a designer assess the feasibility of a design through the results of similar work.

Other systems have been developed for forming process sequence design. Alberti, N. et al. (1991) completed a system to combine manufacturing and production rules with FE simulations for forward extrusion. Arvind, A. et al. (1999) outlined the development of a CAE system to assist with the process sequence design of axisymmetric drawn parts. A system to acquire process knowledge by combining neural networks and rule-based knowledge for impression-die forging process design was developed by Huang, S. H. et al. (2001).

2.5.2 Geometric and Feature Based

Another form of KBS for manufacturing is based on detailed descriptions of the individual features of a component. Knowledge and rules for similar features and concepts across components can then be compiled and reused, particularly during design. Such representations are also often termed 'product models'.

Imamura, S. et al. (1988) described the use of geometric representations based on a G-rep (Glueing representation) data model for a product model. The product model represents a concept model that describes all the information of a product which is required during design and production activities.

G-rep is a data model of solid objects and is defined as the glueing of two and a half dimensional components, which are defined as the linearly and vertically swept plane figures. The model consists of:

- part blocks
- component blocks
- geometry blocks
- geometry primitives and other auxiliary data blocks

A part block describes the part shape and is comprised of component blocks.

Position and orientation are descibed by 3D coordinates. A component block consists of variables, other components and geometric primitives. The latter actually represent basic objects such as circles or lines. It is then up to the designer to express their 'intention' using the G-Rep model and the method described.

Other classification systems proposed for sheet metal components include Greska, W. et al. (1997). This uses Fourier transforms and neural networks to identify and classify features within a part and using those to identify similarities amongst parts. Chen, Y.-M. and Wei, C.-L. (1997) present a framework to support concurrent engineering design for net-shape manufacturing, with die casting process design used as an example. This uses feature-based representations to assist designers, and includes encoding production rules within a knowledge base which are then applied to the feature descriptions of a component design.

Leake, D. B. and Wilson, D. C. (1999) used Concept Maps (CMaps) to represent an individual's cognitive structures by developing external representations of concepts and propositions related to a component's form and function. A CMap is a grapical representation of a set of concepts constructed so that interrelationships among them are evident. It can be used to describe the structure of a complete product for example. These can be annotated with text, images and diagrams at the level of individual concepts. Individual concepts are linked to other related concepts through one or two way links, each labelled with a description of the relationship (Cañas, A. et al., 2001).

CMaps can be constructed by an individual and through their construction, the individual's thought processes are identified in the concepts, their linkage and relationships. New CMaps can be constructed directly or by using another CMap as a base. Cañas, A. et al. (2001) described the use of CMaps for a domain covering the exploration of Mars while Leake, D. B. and Wilson, D. C. (1999) presented the use of CMaps to descibe the construction of a commercial aircraft.

Taking CMaps further Wagner, K. and Aslanidis, S. (2001) described Knowledge Maps (or Cognitive Maps) for the identification and codification of relevant company or individual knowledge. These models provide a basis covering the structure and requirements for the capture, storage and application of knowledge within an organisation. In this work, knowledge maps were created within a module of a larger KM system to utilise implicit and explicit knowledge experience and know-how within the product development process.

2.5.3 Engineering and Decision Support Systems

A range of KBS have been developed to capture and present information and knowledge to support engineering and decision-making processes across a number of manufacturing applications to aid various aspects of design. Examples include Shi, F. et al. (2003) for gating design in injection moulding, Su, D. and Wakelam, M. (1998) for mechnical power transmission system design and Mares, E. L. and Sokolowski, J. H. (2003) for monitoring and quality control processes for metal casting.

Hindman, C. and Ousterhout, K. B. (1998) highlighted the importance of manufacturing design in sheet metal forming and outlined the framework for a Virtual Design System (VDS), a web-based application to assist designers through the sequence of activities required to match part features to manufacturing processes and operations. The system would contain the sequence of steps required to finalise the design of sheet metal components. Designers would interact with the system, and provide part and material information in addition to CAD files. The application would contain rules and equations characterising the stamping system. CAD files would be analysed to check the dimensions of the component to ensure it could be formed without problems. According to Hindman, C. and Ousterhout, K. B. (1998) information would be compiled from a multitude of sources.

Lessons Learnt (LL) are another form of KBS. These support organisational LL processes and implement a KM approach for experience-based working knowledge that can be applied to benefit specific processes (Weber, R. et al., 2001). As described by Weber, R. et al. (2001) LL systems are designed to preserve an organisation's memory that can be lost when experts become unavailable. They should capture and provide lessons, based upon individual experiences, that can benefit employees when they encounter similar situations.

Hong, J. et al. (1996) proposed a personal electronic notebook to capture an individual's reflection-based engineering design information. Using a web-based application, the information would be available for sharing and reuse, to support a collaborative engineering design environment.

A software tool to allow design engineers to record their Design Rationale (DR) for the aerospace design domain was developed and tested by Bracewell, R. H. et al. (2004). This focused on providing a tool that designers could use as the design process proceeds, not just retrospectively once a design was complete. This allowed product definitions and design ideas to be recorded and evaluated and the rationale behind decisions to accept or reject to be recorded, all in a visual graphical form. Links to other sources of informations, such as CAD and FE models and files, could be included.

2.5.4 Case-Based Reasoning

More recent KM work has considered Case-Based Reasoning (CBR) methodologies, where past cases recorded in a case-base are used to identify a solution to a new problem (Kim, K.-J. and Han, I., 2001). This approach aims to parallel an individual's thinking, where past memories and experiences are continually used and adapted to new situations. CBR has three main activities (Rudolph, S. and Hertkorn, P., 2001):

acquisition where cases are captured, classified and stored

retrieval allows existing cases to be searched and extracted

adaption where existing cases can be modified or adapted to the current context or situation

In Leake, D. B. et al. (1999b) a number of general principles for incorporating a case-based support system into the design environment are developed:

- **seamless interaction**, the system should parallel the feasibility engineer's problem-solving approach.

- **just-in-time retrieval**, the system must pro-actively return relevant information to the user, in order to simplify the request process.

- **integration with other knowledge sources**, including prior cases, supplementary information and lessons appropriate to the current task.

- **integration across tasks**, to support current reasoning tasks, but also downstream activities.

- **experience capture**, so that each processing task contributes new cases.

As stressed by Cunningham, P. and Bonzano, A. (1999) many CBR systems involve little or no adaption, they are simply retrieval systems where solutions returned are used intact or adaption undertaken by the user. In these systems the most significant engineering issues are knowledge acquisition and representation. CBR closely reseambles LL methods, but whereas LL systems can benefit an organisation in a number of ways, depending on the immediate lesson context, CBR systems are used to accomplish a specific task.

As with other techniques, CBR has been applied to a wide range of industries, particularly those which are experienced based, such as manufacturing. Leake, D. B. et al. (1999b) describes a case-based design tool to support feasibility design assessment for sheet metal components. This is a software tool to provide a unified framework for collecting, storing and accessing design information. Information is stored using a feature-vector case representation, that allows different types of information to be stored with the one case, in order to help classification, retrieval and potential reuse. Existing feasibility assessments from an automotive organisation recorded on paper were used to develop a library of seed cases. A number of different searches, dependent on the current task context, were available to search and present cases to users.

This is an example of the CBR cognitive model which models human experience as a set of individual cases. The CBR cycle in this application consists of the following actitivies (Lei, Y. et al., 2001; Lee, K. S. and Luo, C., 2002):

determine features to characterise the input

retrieval of similar cases from the case-base

adaption where existing cases can be modified or adapted to the current context or situation

evaluation to retain the new solution if developed and used

Lei, Y. et al. (2001) developed a system to assist with cold forging process planning. Designs are stored as cases, represented using a feature-based scheme, to allow similar designs to be identified and adapted. A CBR design environment to assist die-casting design was developed by Lee, K. S. and Luo, C. (2002). Again, previous designs are recorded in a case library, to be retrieved and adapted to assist with the design of new dies. Mejasson, P. et al. (2000) used a CBR system to assist design and material engineers in the submarine cable industry. The system was used to retrieve similar designs based upon qualitative component and material descriptions. In this instance, adaption was left solely to the engineer. A system to assist casting design engineers by referencing similar previous designs was developed by Mileman, T. et al. (2002). This used a CBR retrieval method to present 3D casting shapes using a similarity measure based upon the shape of previous designs.

CBR can also be used in conjunction with other methods to assist with capture, maintenance, retrieval and adaption. Kim, K.-J. and Han, I. (2001) used a Genetic Algorithm (GA) to assist with classification and indexing. The GA determines the representation of each case within the domain. Lau, H. C. W. et al. (2003) considered the integration of a CBR knowledge model with other information data models including relational databases, data warehousing and OnLine Analytical Processing (OLAP). This is an exercise which must be completed to map the required data model of a system into a model that can be developed in a DataBase Management System (DBMS) in order to implement a CBR support system. Park, M.-K. et al. (1998) completed this using an Entity Relationship (ER) model.

Cañas, A. et al. (2001) developed a combined approach, using CMap knowledge acquisition and representation with CBR retrieval techniques. The latter uses the knowledge within the stored models and contextual information gathered from the experts' navigaton through them. New maps can be created by experts and linked to existing ones, and critiques of existing ones recorded. Retrieval and comparison of CMaps is first based on keyword comparisons and then a similarity metric of two maps calculated. Automatic and proactive retrieval of portions and complete CMaps is also performed. A library of over 150 CMaps covering the exploration of Mars was developed and tested.

A similar framework was presented by Leake, D. B. and Wilson, D. C. (1999) for use in aerospace design. This was again based on the need to overcome knowledge loss, which was a significant concern at NASA. The framework allowed cases to be added incrementally, and was designed to provide support and leverage a designer's knowledge rather than trying to replace it.

A CBR system for problem-solving and training new engineers in the production of colour display tubes was developed by Park, M.-K. et al. (1998). In order to build a case library for the domain, a collecting module was developed, with cases stored in a database. Retrieval was achieved through the calculation of a similarity score, which weighted the similarity of features and symptoms of the new problem and existing cases in the case-base.

Nedeß, Chr. and Jacob, U. (1997) described a framework for quality control loops for the design and production of injection moulds. These combined the outcomes from design activities with a diagnostic system on the production shop floor. A case-base of data from the diagnostic system was used to supplement previous designs during design reuse.

Beyond manufacturing CBR has found a wide range of applications. Mechitov, A. I. et al. (1995) developed a CBR system that outlined the importance of this activity within KBS to assist with medical diagnosis with particular emphasis on knowledge acquisition. In this work a knowledge base of cases was built up through a knowledge acquisition block which directs expert interviews to captured cases which are checked for consistency and classified.

A CBR system for aircraft conflict resolution for use by air traffc controllers was developed using an iterative development process through a series of prototypes (Cunningham, P. and Bonzano, A., 1999). The final system with over 1400 cases was tested in a controlled environment to evaluate the number of 'good' versus 'bad' solutions returned for a new situation. A k-Nearest Neighbour case retrieval process was used, with effectively no adaption being performed by the system, although in some circumstances solutions were aggregated.

In a similar approach Noh, J. B. et al. (2000) and Cañas, A. et al. (2001) used Cognitive Maps (CM) to capture, formalise and store tacit knowledge and then a CBR process to retrieve and adapt similar CMs. This was applied to a credit analysis domain example. CMs are similar to CMaps, and represent the cause-effect relationships which exist among elements of a given domain.

2.5.5 Comparison and Limitations

As stressed earlier, the key information lost from the shop floor is tacit, informal experience gained through years of problem-solving. Knowledge-Based Engineering (KBE) tools often do not include shop floor activities, and hence do not capture data or input from these areas. CBR offers a good opportunity and in these systems knowledge acquisition becomes a critical factor. For manufacturing this must be at source which suggests a significant involvement from shop floor areas in addition to design activities (Mikler, J. et al., 1999). However, to be useful a significant case library must be built-up although as seen with Leake, D. B. et al. (1999b) this can be achieved with information from existing sources and lessons.

Even with concurrent engineering as discussed in Suri, Rajiv et al. (1998) and Yang, S. and Nezu, K. (1997), where design activities are combined, the distinction between these activities and the shop floor and tryout still exist. Chen, Y.-M. and Wei, C.-L. (1997) outline three reasons why computer-based tools have not been able to support concurrent engineering. These are:

1. their stand-alone functionality

2. the lack of capability to support interactive decision-making

3. a reliance on designer experience

Knowledge acquisition is seen as the bottle-neck in knowledge and intelligent systems (Deslandres, V. and Pierreval, H., 1997; Xia, Q. and Rao, M., 1999). This is particularly the case for acquiring manufacturing knowledge in terms of if-then rules (Huang, S. H. et al., 2001). Further, extracting experience from shop floor experts has not been successful as such experience is often implicit and cannot be readily captured and codified (Huang, S. H. et al., 2001). Expert and rule-based systems require a rule-base to be set up and maintained, as do feature-based systems, which focus primarily only on part design.

Most expert systems can only solve problems in narrow domains, as they provide no means to rebuild their application-specific knowledge bases (Yang, S. and Nezu, K., 1997). Hindman, C. and Ousterhout, K. B. (1998) do not describe in detail where information will be sourced, although commonly used equations will be pre-programmed. However, these will be static, and for ideal situations only. A new set of rules has to be collated and entered for each new situation. Further, the system only focuses on design activities, it does not extend onto the shop floor. This is also the case with Tang, D.-B. et al. (2001), where a set of detailed rules must first be established. As discussed in Nedeß, Chr. and Jacob, U. (1997) it is easier and less time-consuming to maintain a small heuristic similarity knowledge base and a large scale case-base than a large, interconnected rule base. Finally, it is virtually impossible to consider every possible situation that needs to be included(Mechitov, A. I. et al., 1995).

It is a similar situation for feature-based systems, which often require the use of a rule-base or each part to be described in detail in terms of its individual elements and their relationships. Even with the generic LL system described by Weber, R. et al. (2001), each individual lesson is typically verified for consistency, correctness and relevance by a team of experts. Again this can introduce a significant bottleneck in terms of assembling and relying on a group consenus.

Compared with knowledge- or rule-based reasoning, CBR has a number of benefits (Lee, K. S. and Luo, C., 2002):

- it more closely represents human reasoning
- a case library can be constructed more readily than a rule library
- it supports more effective transfer and explanation of knowledge
- is useful where domain knowledge is difficult to acquire

CBR is more appropriate for small data sets, which is likely to be the case when acquiring experiences from the shop floor (Gonzalez, R. and Kamrani, A., 2001). As discussed by Cunningham, P. and Bonzano, A. (1999) in situations where determining key features of knowledge representation is more straight-forward than the task of developing a reasoning mechanism, CBR is more effective than rule-based systems. However, as the acquisition of cases and selection of features becomes more dominant, the advantages of CBR are less evident. This does not mean that cases are easy to extract from users. Any system still has to be able to draw out experiences and cases in an efficient and appropriate manner. A further advantage of retrieving and presenting previous actual cases to a user is that they can be more convincing than a solution developed through a rule-based derivation process (Nedeß, Chr. and Jacob, U., 1997).

Knowledge-acquisition is typically a bottle-neck within KM, with two extremes, one focused on detailed, structured and hence time-consuming capture and representation, such as geometric or feature representations, the other on low-cost acquisition at the expense of more difficult reasoning. A system that tends towards the latter with a strong focus and emphasis on knowledge capture is required.

A system which captures experience as and when it is acquired is required, in order to address concerns with knowledge acquisition. This means it must have significant involvement from not only design areas, but particularly from the manufacturing shop floor, the areas where the success or otherwise of a design is identified and observed, and often corrected. Of the work surveyed here, only Nedeß, Chr. and Jacob, U. (1997) had clear and formalised links to the shop floor environment, although the information captured was numeric process data rather than experience from personal. However, in manufacturing as in other industries, KM is not only concerned with knowledge, but with three main elements namely people, processes and tools (Lewis, K., 2002).

2.6 Non-Technical Considerations

In addition to the limitations and discussions outlined above, when considering the implementation and use of a KM system more than only technicial issues of representation, storage, and use must be addressed (Damodaran, L. and Olphert, W., 2000). Requirements relating to the personnel, environment and context in which a system will be used must be identified (Lewis, K., 2002). These requirements need to be included from the very outset of the development of a system and included in the project methodology in order to increase the acceptance, use, and value of the system.

Through a post-implementation review of user interviews and experiences of an electronic information management system, Damodaran, L. and Olphert, W. (2000) summarise the four main causes of under-utilisation of the implemented system as:

- inadequancies of the technology
- lack of user-friendliness
- existing high workload leading to a lack of capacity for new tasks
- failure to institutionalise the system by not creating the appropriate culture and supporting practices

The first two items are related to the design of the system and can be addressed through further refinement and development. However, overcoming these concerns can still result in a low up-take. A culture of trust was deemed critical for the successful implemenation of the system. This can be strongly affected by concerns of job security if knowledge sharing is seen to reduce the value of an individual to the organisation. The implementation of a new system can also be accompanied by a realignment of power, status and working habits. If a new technology cannot be aligned to established values of a group, it can result in resistance to use (Cooper, R. B., 1994; Pantano, V. et al., 2002). Lapré, M. A. and Van Wassenhove, L. N. (2001) identified a lack of management buy-in and inter-departmental problems as factors that may have impeded learning and knowledge transfer of productivity improvement processes between related factories within an organisation. The same concerns can impact on technology transfer and implementation of new KBS.

Other barriers to problem-solving, specific to the automotive industry, were identified by Vinarcik, E. J. (1999). These were classified into three main groups:

Cultural barriers generated by personal philosophies and perceptions

Tactical concerns related to problem-solving

Strategic concerns beyond the problem-solving team, related to mangement

The computer skills of shop floor employees, combined with cultural and organisation influences can result in a strong resistence to change and low adoption rates for new technologies (Cooper, R. B., 1994; Lapré, M. A. and Van Wassenhove, L. N., 2001). To integrate presentation with the user's problem-solving activities, KM systems must provide their information in a manner that parallels the user's reasoning process (Leake, D. B. et al., 1999b).

As outlined by Leake, D. B. and Wilson, D. C. (1999), to encourage acceptance of systems in applications such as design and manufacturing, they must leave the user in control. Software systems and tools must also be intuitive to their users and relate to their natural problem-solving and decision-making approach and other commonly used computer-based tools, such as

Windows applications or internet browsers (Bracewell, R. H. et al., 2004). Access to the system must also be quick and easy, otherwise use may decrease as additional time is required from the user (Bracewell, R. H. et al., 2004) .

2.7 Conclusion

Within automotive stamping, capturing experiences and providing feedback is vital to shorten lead-times, improve quality and overcome knowledge loss. From the discussion presented in this chapter, it can be seen that a gap exists in capturing and intregrating the typically tacit manufacturing experience from shop floor activities. This is related to the environment, production pressures and the background and culture of shop floor employees which can produce resistance to the introduction and use of KM systems. However, by considering and including a number of technical and non-technical requirements during design and development, a software system can be developed to bridge this gap. The next chapter outlines the approach taken in this study to achieve this aim.

Chapter 3

Methodolgy

The need for a knowledge management system to capture experience currently being lost from the manufacturing shop floor is clear. The approach and activities for the development of a new system is described in this chapter. This also required the development and adaptation of software development methodologies that could achieve acceptance, achieve buy-in and avoid departmental barriers.

3.1 Research Approach

In order to fully understand the research environment, users' needs and design principles, a participatory, action-based research methodology was adopted. This also assists to achieve the required buy-in identified as critical to potential acceptance and use.

3.2 Project Approach

A review of the literature of manufacturing systems and software development enabled a specific approach to the development of the intended application to be defined (Leake, D. B. et al., 1999b; Cunningham, P. and Bonzano, A., 1999; Saiedian, H. and Dale, R., 2000; Lapré, M. A. and Van Wassenhove, L. N., 2001; Bracewell, R. H. et al., 2004). This was to be in-plant, using a rapid prototyping methodology and an inter-disciplinary team embracing engineering, IT and trade backgrounds. From this concept, a detailed set of activities and their sequence was defined (see figure 3.1).

While the researcher was involved in all stages, the domain analysis, design requirements and framework, analysis of results and evaluation and recommendations were completed solely by the researcher and form the basis of this thesis.

Before any research or technical work could commence, buy-in and support from plant management had to be secured. Any study of this kind requires a high-level of industry and user involvement, which can only be achieved with this support. This is essential in order to gain access to the plant and its personnel, and to ensure any applications developed will be able to be tested. Such support is secured by highlighting and detailing the concepts and outcomes of the system through a series of meetings and presentations with management from a number of different levels. A key element is to identify a plant champion from management who will support the work and provide assistance and resources when required.

Once plant buy-in has been secured, a domain analysis must be completed in order to gain a detailed understanding of the tasks and context the application will support and the experience generated that can be captured and utilised. The outcomes are a set of high-level design principles and requirements which the final developed system must satisfy. These are also used as the basis for the underlying data model of the datastore where data and information will be recorded.

From these initial requirements, development of the actual system can commence. A number of options are available for the software development required, and these are outlined in the section 3.4.

3.3 Development Team

As outlined by Saiedian, H. and Dale, R. (2000), regardless of the methodology used it is the team that undertakes the development that will have most impact on a successful implementation. A small, decentralised, in-

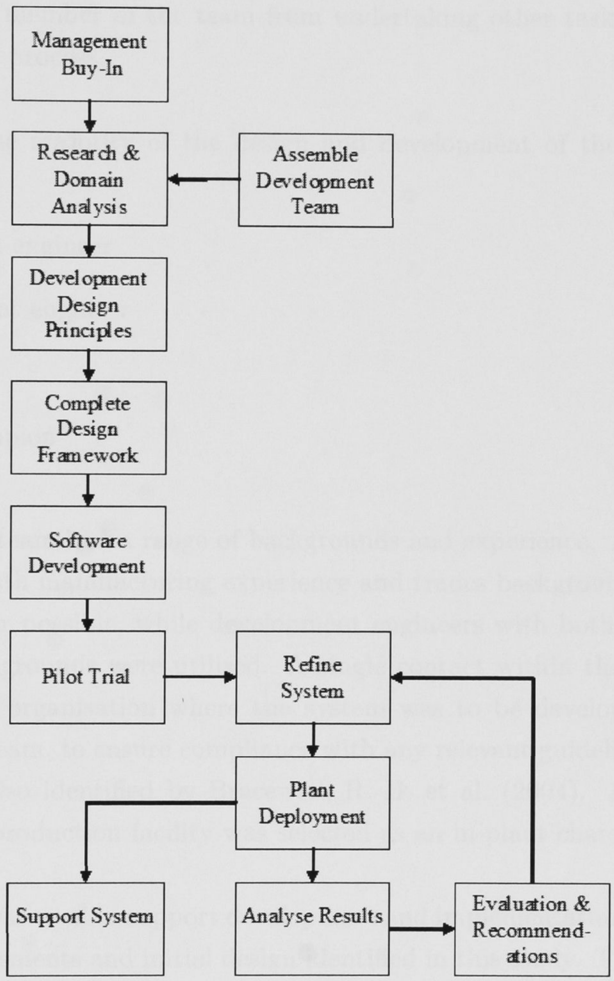

Figure 3.1: Approach and activities for the development of the software system

terdisciplinary team was selected for the software design and development. A distinction was made between application and development engineers, the latter being primarily focused on software design and coding, while the former were associated with interactions with users, including discussions, demonstrations, training and implementation support. However, this did not exclude any member of the team from undertaking other tasks during the development process.

The team for the majority of the design and development of the system comprised:

- application engineer
- development engineer
- IT contact
- plant champion

Members of the team had a range of backgrounds and experience. Application engineers with manufacturing experience and trades backgrounds were selected wherever possible, while development engineers with both IT and engineering backgrounds were utilised. A single contact within the IT department of the organisation where the system was to be developed was included in the team, to ensure compliance with any relevant guidelines and procedures, as also identified by Bracewell, R. H. et al. (2004). A senior manager in the production facility was selected as an in-plant champion.

The same team was used to support development and implementation, building on the requirements and initial design identified in this study. The team operated a virtual development model, working at both the research institute and the industrial facility.

3.4 Software Development

As with product lifecycles previously described, software engineering has a set of activities that must be undertaken to complete development of a new

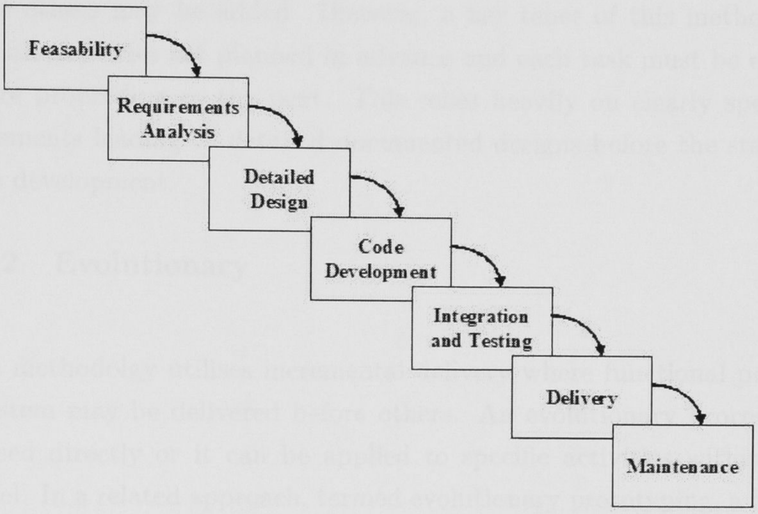

Figure 3.2: Linear software development model

application. A number of well researched software development methodologies have been described by Ghezzi, C. et al. (1991); Elliott, J. J. (2000); Hull, M. E. C. et al. (2002), although most can be categorised into one of four common models:

- sequential or linear, also known as the Waterfall model
- incremental, or evolutionary
- transformation
- risk-based

These are described in the following sections.

3.4.1 Linear

Software development follows a sequential linear model. Outputs from one activity, such as documents or code, are used as the inputs to the next (see figure 3.2).

Within the scope of a specific project, not all activities may be required, while others may be added. However, a key tenet of this methodology is that all activities are planned in advance and each task must be completed before proceeding to the next. This relies heavily on clearly specified requirements leading to detailed documented designs before the start of any code development.

3.4.2 Evolutionary

This methodolgy utilises incremental delivery where functional portions of a system may be delivered before others. An evolutionary process can be utilised directly or it can be applied to specific activities within another model. In a related approach, termed evolutionary prototyping, a prototype system is progressively transformed into the final application through extensive testing and user involvement during development (Ghezzi, C. et al., 1991).

This methodology can be particularly effective for applications that rely heavily on a user interface or do not have clearly defined requirements at the start of the development lifecycle. For these situations, prototypes can be rapidly developed and demonstrated to potential users for feedback and comments, to guide the appearance and functionality of the final application. The Spiral Model, proposed by Boehm, B. W. (1988), is a specific methodology which allows for a number of iterations of the development process with more detailed prototypes developed at each stage.

3.4.3 Transformation

A top-down approach is employed in the transformation model. Specifications and requirements are progressively decomposed into sub-systems and more fine-grained components. Each of these are then independently completed and implemented. This methodology relies on formal requirements modeling to identify and describe the functionality of the final application. As with the linear approach, a clear definition of the final requirements of the system is necessary as early in the development process as possible.

3.4.4 Risk-Based

As described by Jiang, J. and Klein, G. (2000), software engineering can be considered a process of risk identification and management. Various methodologies have been developed to identify and then control the risks associated with software development (Procaccino, J. D. et al., 2002). However, these techniques are typically applied to other methodologies described here, such as the linear or transformation models, rather than a distinct approach.

3.4.5 Comparison

Software development processes, such as the linear or transformation approaches follow a highly sequential set of activities. They start with a detailed set of requirements which are used as the base for all further design and development. However, requirements are often not well understood or specified, and complete designs cannot be documented early in the process (Cunningham, P. and Bonzano, A., 1999). In such projects an evolutionary approach is more appropriate, as requirements and functionality can be detailed and finalised as prototypes of the application are progressively developed.

The software process required for this work needs to be inclusive and capture input from prospective users, as this can effect use and perception of the final application. As highlighted by Damodaran, L. and Olphert, W. (2000) the uptake of new systems can be enhanced by creating a technology pull from users. A rapid prototyping process where users can see, use and provide input into early versions, allows this to occur and can create ownership of the new system. This can help overcome group cultural concerns as raised by Cooper, R. B. (1994) that could otherwise result in active resistance to the use of a new application.

A rapid prototyping approach also enables significant input into user interface design. A lack of sensitivity to user needs can result in a user perception of a lack of usefulness or ease of use, or that the system contains incorrect, irrelevant or unreliable information (West, A. A. et al., 1999).

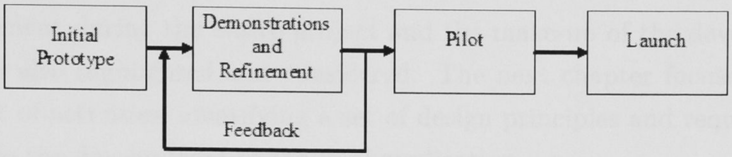

Figure 3.3: Software development process and activities

3.5 Development Methodology

To further refine the approach in figure 3.1 a bottom-up evolutionary rapid prototyping development methodology was selected to complete the software development activity. Based upon the previous discussion, this was selected for a number of reasons:

- the lack of formal requirements from the manufacturing organisation
- the need to achieve operator buy-in and ownership of the system
- the emphasis on user interactions with the system

The activities undertaken and their flow is shown in figure 3.3. The rapid prototyping phase is used to elicit comments and feedback from potential users. Once the prototype was considered appropriate a pilot trial of the system would be conducted. This would be used to capture data and information and analyses and assess use of the system over a period of time. Similar approaches were used by Cunningham, P. and Bonzano, A. (1999) and Bracewell, R. H. et al. (2004) with good outcomes in terms of the functionality and acceptance of the applications developed.

3.6 Conclusion

An approach for the design and implementation of a new software system to acquire and integrate information and experience from the manufacturing shop floor is described. A rapid prototyping software development methodology was incorporated in order to build ownership of the final system to assist with deployment and use. The importance of management buy-in and involvement during the entire project and the make-up of the development team is also highlighted and considered. The next chapter focuses on the first set of activities, identifying a set of design principles and requirements to guide the development of the final application.

Chapter 4

Design Principles

A set of design principles for a system that can acquire experience from the automotive manufacturing shop floor efficiently and effectively were developed and are described in this chapter. These are based on a domain analysis, combing outcomes from the investigation of previous work, relevant background research and an in-plant study.

4.1 Introduction

In order to identify a set of requirements and the high-level design for the development of a new system, a number of different sources and activities must be utilised. The detailed examination of background and existing work described in chapter two was undertaken to identify non-technical requirements. This was complemented by a detailed domain analysis at a large (by Australian standards) automotive stamping facility to examine existing work processes and the shop floor environment in order to identify key requirements and elements to be included in any future design and development. The outcomes of these activities were combined to develop a set of application-independent design principles which the system must encapsulate to ensure acceptance and use. Finally, a design framework for the application was detailed.

4.2 Requirements

A set of requirements were identified from background research and an on-site domain analysis. The former was used to generate an initial list of non-technical requirements, while the domain analysis was used to provide a better understanding of the shop floor environment and validate important areas identified during research.

4.2.1 Non-Technical Requirements

By combining the review of published work and research of automotive stamping a set of relevant initial requirements to be included were identified. Those described for KM systems identified by Damodaran, L. and Olphert, W. (2000) include:

- top-driven, with clear management support
- fast path implementation
- integration with other company initiatives
- institutionalised into normal work practices
- add value to individuals and teams
- relate to specific business needs
- easy and quick to use
- appropriate levels of training and support
- populated with a critical amount of data
- built upon a culture of trust

A number of these can be incorporated into the functionality of the system being developed, such as ease of use and relationship to business needs, while others need to be considered during development or implementation. Critically, a system cannot be developed and implemented by only considering the technical aspects. The importance of trust and integration within existing organisational strategies was also highlighted by Mikler, J. et al. (1999) while management buy-in and interdepartmental barriers were listed as impediments to organisational learning and technology transfer in, and between, factories. The software development methodology utilised for a project can help to ensure the problems are minimised.

4.2.2 Domain Analysis

A domain analysis for the automotive stamping facility was completed, combining resesearch and observations from industry. This consisted of a number of week-long site visits, where integration with plant business units and operations and initial discussions with plant personnel were conducted. From this, a number of important principles relating to the work practices of shop floor areas (including tool build, tryout, production and maintenance) were identified:

- component based
- visual and spatial
- experience based
- time focused
- departmental barriers

Component Based

Manufacturing is driven by the production of parts. When faced with overcoming problems, operators and engineers refer to the part being manufactured. Issues may be related to an individual tool or process, but are identified and related to the part being produced. Problem-solving is focused on correcting concerns for a part to ensure its production targets can be meet.

Visual and Spatial

Operators and engineers are visually based in their thought processes. Dis-

Figure 4.1: Panels with information added by plant personnel

cussions on tool or part changes would be conducted around the die or component, using hand gestures to describe material flows and strains, often drawing directly onto a scrapped or damaged part (see figure 4.1) or if not near-by, a sketch of the part would almost always be made.

Experience Based

Alterations during problem-solving are often made based upon on experience with similar parts or tools. This often takes the form of

> "We had a similar problem on the tool for a previous model (vehicle)."

Following from the component driven principle, such experiences are typically recalled on a component basis.

Time Focused

Production must meet set targets to ensure enough parts are made during each shift. As a consequence, any activitiy not directly related to manufacturing parts is considered non-essential. Hence, data collection is typically perceived as an overhead and will not be collected if not seen as directly helping the production of parts.

Departmental Barriers

Strong departmental barriers were identified within the organisation. This led to communication difficulties, particularly during collaborative problem-solving involving the shop floor. This was also apparent when incorporating improvements and lessons for future model developments, as shop floor areas did not feel they were receiving feedback on issues raised in their areas.

4.3 Design Principles

By combining the non-technical requirements and outcomes of the domain analysis a set of key design principles were identified which must be included in the design of a system to capture shop floor manufacturing experience. These were:

task-based capture and retrieval

common structure for storage

integration with work practices, business systems and IT infrastructure

experience capture method which parallels the natural thinking approach

support only not replacing decision making

4.3.1 Task-Based

Each stage of the product lifecycle is carried out by one or more departments within an organisation. In turn, each department has different requirements for the information they generate and use, the resources available to them and the background and experience of their personel. As discussed by Leake, D. B. et al. (1999a), systems should be designed to complement the user's reasoning process during their current task context. This means ensuring functions are available to a user relating to the activity they are performing, rather than having different activities restricted to using the same functionality.

4.3.2 Common Structure

Although functionality needs to be task-based, experience and information should be stored in a single logical data base, using the same basic structure. This simplifies the data model design and allows information from different activities to be accessed and integrated during searching and retrieval.

4.3.3 Integration

As highlighted by Leake, D. B. et al. (1999b); Lee, K. S. and Luo, C. (2002); Weber, R. et al. (2001) a KM tool must be supported by, or support, daily business practices of the organisation. A key tenet is to achieve management buy-in, as management must drive the use of a system. A related need is to provide management reports and functions which assist use and aid allocation of resources in conjunction with the system. Further, if a system is not integrated there is a strong likelihood of it becoming purely a static repository of information. As discussed by Bracewell, R. H. et al. (2004) a system must also meet all relevant IT and security requirements, so that it is integrated into the organisation's IT support and maintenance practices.

4.3.4 Experience Capture

Experience should be captured in an unobtrusive manner, as a consequence of using the system, in a timely fashion (Leake, D. B. et al., 1999b). This should parallel the natural thinking and problem-solving of the current task. During the domain analysis, when asked to describe a problem, shop floor personel would invariably draw a sketch or highlight the concern on the actual part or tool, and so a visually-based system was regarded as a key requirement, and this must be integrated into the design.

Experience should be captured as cases, one case per learning opportunity, and stored in the central database. Experience should also be captured at the source where it is generated and by the person who generated it.

4.3.5 Support Only

The system should act only as a support tool. This has been identified as a significant requirement to ensure acceptance of decision-support and problem-solving systems (Leake, D. B. and Wilson, D. C., 1999; Nedeß, Chr. and Jacob, U., 1997). Presenting actual experience as an aid will result in more acceptance of the results, as they are based upon actual situations within an organisation rather than a solution generated by a more generic expert system or rule base. As suggested by Leake, D. B. and Wilson, D. C. (1999), this principle will also assist in users gaining trust of the system which, as identified by Damodaran, L. and Olphert, W. (2000), is a key element to increase acceptance and uptake of a new system, and hence the amount and quality of experience captured.

4.4 Design Framework

From the design principles developed an initial framework for the system was constructed. This was used as the starting point for system development, to be progressively incorporated through ongoing prototype development.

Based upon the design principles, a CBR-style system was selected as a model for the system. Experience and information is stored as individual cases within a single logical case-base, accessed by a web-based user interface. Experts often find CBR systems to be a more accessible method for capturing and sharing knowledge as it parallels the natural problem-solving approach (Cañas, A. et al., 2001).

With the requirements and principles previously outlined, a number of key elements of a design framework were defined:

- Modularity
- Acquisition
- Data Model
- Retrieval

As seen in a number of studies, the process of capturing and representing knowledge can be decoupled from retrieval and reasoning mechanisms (Leake, D. B. and Wilson, D. C., 1999; Cunningham, P. and Bonzano, A., 1999; Noh, J. B. et al., 2000; Cañas, A. et al., 2001). These are represented here by the Data Model and Retrieval elements of the system respectively.

4.4.1 Modularity

While all experience captured needs to be stored centrally with a common structure, the method of input or output needs to be task-based and complement existing work practices and systems. In order to achieve this a modular design was adopted. Different modules can be developed to suit specific tasks, to provide tailored screens and functionality. This also allows additional functionality to be incorporated at a later stage.

A number of CBR systems, such as those described by Park, M.-K. et al. (1998) and Leake, D. B. and Wilson, D. C. (1999), include elements, or modules, which allow cases to be entered by users, in order to build-up a case-base rather than predefine specific cases. A modular approach was also used for the system developed by Wagner, K. and Aslanidis, S. (2001).

4.4.2 Acquistion

Experience acquisition must parallel the natural problem-solving approach. As identified in the domain analysis, this is component and visual-based. Case acquisition should make use of images of problems and resolutions wherever possible. In Leake, D. B. and Wilson, D. C. (1999) visual representations were especially useful. This was also found during the evaluation

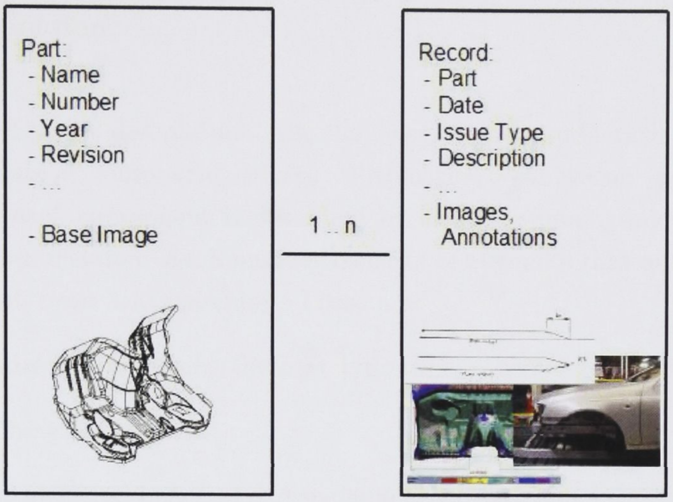

Figure 4.2: Data model with relationship between parts and records

of a number of user-interface designs used in information retrieval systems, where graphical and visual interfaces were more effective than text or list based designs (Hu, P. J.-H. et al., 1999).

To help ensure timely experience collection, as required in production areas, access to the system must be as close to the location of work as possible. Computers on press lines and try-out areas are required, and the system must be fully networked, such as web-based. This also simplifies deployment and maintenance to help satisfy the integration and knowledge acquisition design principles, and helps to overcome departmental barriers by providing feedback to all areas on the status and progress of issues raised.

4.4.3 Data Model

All information and experience entered will be stored within a single logical case-base, or database. In order to aid searching and retrieval, cases are classified when first entered. The data model consists of two major elements, part information and record information, allowing for a number of different types of experience to be recorded. These elements, and their relationship, are illustrated in figure 4.2.

Part Information

As identified in the domain analysis, the design and manufacturing process in this domain is component driven. Within the organisation used in the case study, each component is described by a part number, unique across the entire organisation. Each number consists of elements that are used for classification, reuse and searching. These are:

Base Number representing the part type such as a *front door outer*

Model such as *sedan* or *ute*

Year the model, and hence part, was launched

Engineering Level or revision, for any major or minor changes made to the initial part design

Part Name which contains key descriptors such as *fender* or *outer*

Base Image a CAD screen shot or digital camera image of the overall manufactured part

Part information is supplement by generic parts, such as *All Parts* or *All Robots* and process, such as *Design* or *Measurement*. These allow process, equipment and more generic experiences to be captured and integrated.

Record Information

Individual experiences or cases captured by the system are termed a record, and represent cases in the case-base. Every record is associated with a specific part, which can include generic parts or processes. Any record entered contains the following fields:

Part/Process selected from the parts stored in the database

Date automatically generated from the day the record is entered

Operator captured when the user logs on

Concern Type selected from a predefined list of common one or two word issues stored in the system

Concern Description a free text field allowing the symptoms and possible causes of a concern or improvement to be entered

Suggested or Recommended Action a free text field which may include any actual or temporary actions already completed

Other Action another free text field where an indication of whether the action entered was successful or implemented

Concern Location a set of coordinates on the base image of the part describing the location of the concern

Review Timing a stage-gate milestone timing describing when the record should be reviewed and incorporated during the product lifecycle

Status of the record within the system, such as *initiated*, *actioned* or *closed*, to assist tracking

Images of the actual problem, which can include photos, sketches or screen shots

Annotations recorded on images, such as a limited amount of text, circles, squares and lines

As discussed previously, images supplemented by a number of free-text fields should facilitate more timely case acqusition than text-fields only, and also provide a more detailed and understandable descrption.

4.4.4 Retrieval

Based upon the activities the system will support, there are two main forms of reuse, one associated with the shop floor, the other with design areas. On the shop floor, operators will be faced with production or quality concerns, and will be seeking a resolution based upon previous similar problems. In this case, the concern, part and process become key fields to search. In design areas, operators and engineers will typically be focused on specific sets of parts, and will want a complete part history of cases from surrogate parts. In accordance with the context-based design principle, separate searches and reports need to support these different requirements. However, both are based on a simple nearest neighbour search, generated across relevant fields from the part and record data models.

More general reports are also required to assist management to drive use of the system throughout the organisation. These need to inform management and supervisors of the number and stage of cases within the system to enable them to track use and allocate resources. Metrics and reports are calculated directly from the information and cases stored.

4.5 Conclusion

In this chapter an application independent set of design principles for a system to acquire and integrate experience from the manufacturing shop floor are described. A detailed design framework for an application for the automotive stamping industry was then outlined. In the next chapter, a case study outlining the development and implementation of a system encapsulating these principles and framework is presented.

Chapter 5

Case Study

This chapter outlines the development and implementation of a new application encapsulating the design principles identified, using the methodology developed to assist stamping operations within an automotive manufacturer. The environment in which the system was implemented, the deployment methodology, significant prototypes developed and application of the final system are described.

5.1 Environment

The system was developed and implemented in the same organisation that was used for the domain analysis. This was the Australian arm of a large multi-national automotive organisation, with an annual output of around 65,000 vehicles. At the time of the study, the organisation was employing over 4,800 personnel in Australia, with approximately 1000 in the stamping plant. All activities in the product lifecycle for two vehicle models and a number of derivatives were completed within Australia, from concept through to production. Die design and build, tryout, production and some sub-assembly were completed on the same site, a long established facility which produces over 400 stamped structural and chassis components, with the majority of tooling at the time made locally in the toolroom.

The stamping plant had a strong ingrained hierarchical structure, with a number of layers of senior and middle management above supervisors and hourly personnel. There was also a clear distinction between the shop floor, with a high number of hourly personnel with a trade background, and the design areas staffed mainly by salaried employees. The company had recently launched a new vehicle model that was not received as well as expected. This had led to a number of changes, including the retirement of a number of experienced personnel. For a full description of the organisational structure and culture see Pantano, V. (2004).

The organisation had a well established computer network, with an organisation wide intranet and a number of PCs on the shop floor. However, at the start of the study many shop floor employees had only minimal computer skills.

5.2 Software Development

The software development approach outlined in chapter three was applied at the site and is described in the following sections.

5.2.1 Management Buy-In

Although a bottom-up approach was employed, achieving management buy-in was absolutely necessary for development to proceed, to gain access to resources and potential users and for a commitment to ongoing use of the system. Initially, this buy-in was for the broad objectives of the project rather than the exact functionality of the final system. This enabled the adopted software development process to be used by allowing the research and development team on-site when necessary and to seek input and feedback from users, in a bottom-up manner and to allow ownership by potential users as development continued.

5.2.2 Initial Prototype Development

Once support from plant management was secured initial discussions were held with a small number of experienced shop floor personnel. This had a number of goals:

- determine existing work practices and methodologies
- identify problems with existing tools or processes
- ensure key members of the plant were included from the outset of development

Discussions with six employees were conducted, typically in a session of 20-30 minutes duration. During this period, time was also spent by the researcher observing the day-to-day operation of the plant, to gain familiarity with the stamping process and the types of experiences generated that needed to be captured.

From these discussions and observations, an initial prototype was developed, called Version 1. This had example functionality only, it could not record or store new data. This was used to encapsulate some of the key concepts to be incorporated in the system. It was developed in Matlab (Mathworks, 2007), an interpretative language, meaning changes and alterations could be made rapidly without the need to compile and deploy the entire system, but the system could only operate on a single computer. This version included images of a representative problem, which could be annotated to describe concerns recognised on the shop floor.

5.2.3 Demonstration and Refinement

Over a six-month period Version 1 was demonstrated and used as a base for further development. During this period three distinct demonstration periods were held. Each of these lasted approximately a week and took place in a training room onsite within the plant. This enabled plant employees to view demonstrations of the system at a time of their convenience. To allow for ease of demonstrations and open dialogue, small groups of two to four people were encouraged, in sessions of typically 15-20 minutes. Two members of the development team were present during these sessions, one demonstrating the system, the other asking questions and taking notes. The

Figure 5.1: Screen shot from prototype Version 2

prototype system provided a focus for the discussions and allowed potential users to comment on what was demonstrated.

After each demonstration period feedback and comments were collated and the prototype further developed and refined. This included changes to the interface and appearance, increasing the functionality of the system and expanding those parts seen as most beneficial, while maintaining the key requirements and objectives. Such demonstrations were provided to 44 employees from all areas of the plant, including both salaried and shift workers, involved in design, production and maintenance.

At the end of this phase, Version 2 of the system was completed (see figure 5.1). This was used during the pilot activities, and was developed using tcl/tk, another interpretative language (tcl/tk, 2007). It further emphasised the visual aspects of the system and incorporated the concept of a record. It also encapsulated the principles behind CBR where each experience entered is captured and stored as an individual record. A record is raised by an operator to:

- capture a lesson they have learnt
- record an improvement they want to see incorporated in future designs or developments
- current issues or concerns they are facing that may require a change or alteration in a different area or they are not sure how to solve

These were characterised by a number of different record types, to be used depending on the users' context and roles, namely:

Model Improvements (MI)

Lessons Learnt (LL)

Design or Production Issues (DI)

The acquisition of these records must parallel the natural problem-solving approach of users. This is necessary to make the description of their work:

- less time-consuming to record
- elicit as much information as possible
- capture as accurate a description as possible

To achieve this, when entering a record the user follows a number of steps:

1. selects the part
2. selects the record type
3. selects relevant images that have been taken or are available
4. marks up the images with annotations, notes, dimensions or sketches
5. selects from predefined options, including a concern type, to classify the record
6. supplements the images with free-text comments

Selecting a record type, part and classifiers when first entered enables records to be categorised and stored in a manner to aid retrieval. Once the initial record has been entered, other information or work related to the record, such as resolutions to issues or if an improvement was incorporated or not, is entered alongside the initial information. In this way, a complete history of each experience is built-up and stored.

The strong emphasis placed on using images during acqusition was based on the natural problem-solving approach of personnel observed in the plant, which was primarly visually and spatially oriented. Images can include photos of parts and tools, exported as 2-D screen-shots of CAD model or FE simulation results, or sketches. This capability satisfied a large number of requirements of the system. It allowed more accurate descriptions of any experiences to be recorded, and in a shorter space of time. As with many CAD and FE packages, the user interface was also icon-driven (as shown in figure 5.1).

A data record was entered in a flat text file on the PC where the system was installed, one file per record, with separate files to record other data including part numbers, operator identifiers and list options. Functionality to enter records was supplemented by administration operations to enter and edit part details, user access and options.

5.2.4 Pilot

During the demonstration and refinement stage, a clear need for the software system being developed was identified. A demand had been created from within the plant to use the system to record lessons learnt and problems with tooling during new launch programs. In order to test and assess the system in more detail and generate further ownership, a small pilot trial of the system was conducted. At this stage the prototype was functionality complete in terms of user interface capabilities, although data storage and communication were not finalised, versions could only operate independently as stand alone instances. In the interim, data from different instances could be merged and redistributed manually.

Figure 5.2: Pilot installation on stand alone PC in tryout area

Two key departments of the plant, die design and tryout, were selected to host versions of the prototype. Dedicated PCs were set-up (see figure 5.2) and a small number of people trained in the use of the system. The prototype was used over a three month period to record lessons learnt in design and issues occuring with tooling in tryout.

Evaluating Version 2 at the end of the pilot, the functionality and user interface of the system were identified as being particularly suited for use within the stamping environment. As expected, the limited number of PCs and lack of communication severly limited integration and use of the information recorded. However, the demonstration and pilot activities had developed a strong sense of ownership from potential users of the system, across departmental barriers, and a pull from the bottom-up for the system to be used more widely within the stamping community.

5.3 Deployment

A few months after the completion of the pilot plant management decided to proceed with a full implementation of the system which was to be used to support stamping operations on a plant-wide basis. This took Version 2 as the base for the user interface and functionality of the system and concentrated on the communication and data handling elements of the system. This development, and the implementation that followed were performed in the same manner using the core of the initial development team, including the researcher. The system was then implemented across stamping activities to support a new model launch program.

5.3.1 Final System

Version 3, the final version, was web-based, allowing ease of deployment, maintenance and accessibility (see figure 5.3). The system could be accessed from any networked computer within the organisation, and enabled employees from any area of the organisation to be involved in problem-solving and review. Information was entered in the manner used for Version 2, as specific experiences associated with an individual part. This was supplemented by generic parts, which represented processes, procedures and equipment, allowing experiences with these to also be recorded. Defined feedback channels were introduced to help with issue tracking and resolution.

The data model was implemented using a central SQL-compliant database management system (DMBS) (Sethi, I. K., 2001). Such systems are used in virtually all large organisations, and satisfy the design principle of integration by using existing technology and infrastructure. More reports were included to help reuse of experience, tracking and management of the experience captured. A link was made to the organisation's master parts list, to ensure new parts were available in the system as soon as they were developed and did not have to be entered manually. See Appendix A for more details of the design and functionality of the final version.

5.3.2 Roll-Out

A strategy was developed for the plant wide roll-out of the application. This was staged, starting with the key areas of tryout and die design, the same areas used during the pilot study, then progressively to other areas over the

Figure 5.3: Screen shot of the final version of the system

following six months. At the end of the roll-out personnel from the following areas had been trained and were using the system for various applications:

- **Product Development**, including part and process design
- **Stamping and Structures - Assembly**, including assembly, gauge, jigs and fixtures design
- **Stamping and Structures - Stamping**, die and process design
- **Tool Build** including die tryout
- **Press Shop** production
- **Sub-Assembly** of sub-systems of stamped components

As the system was web-based, technical deployment was relatively straightforward. The system and database were installed centrally by the relevant IT group and a unique web url address provided, at which point it could be accessed by any networked PC within the organisation.

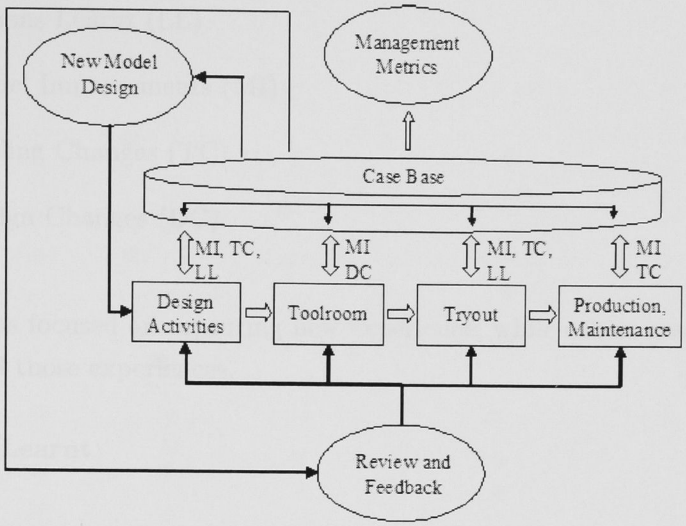

Figure 5.4: Implementation of the system

Training was conducted by the development team, in conjunction with the plant's local training department. Each training session, of up to eight people (limited by the number of computers available in the training room), was of two hours duration, and was mixed to ensure a middle or senior manager was present in each session. This was to demonstrate commitment to the system. Relevant user manuals and support documentation were developed by the development team.

One of the most important factors identified during the research was the need for continuing management support. This was also observed by Procaccino, J. D. et al. (2002), where the committed sponsor is one of the two key factors required for successfully implementation of a software development project. During the system launch, plant champions were selected to assist deployment and display a commitment from the organisation to the system and its objectives.

5.3.3 Application

The system was deployed to support a number of different business processes in use within the plant (see figure 5.4), including:

- Lessons Learnt (LL)
- Model Improvements (MI)
- Tooling Changes (TC)
- Design Changes (DC)

Some areas focused on capturing new experience, while others assessed and integrated those experiences.

Lessons Learnt

As described at the start of this chapter, the organisation where the system was implemented had been through a less than successful model launch. A number of changes were made to the vehicle, including new materials and styling, both of which impacted significantly on stamping design and manufacture. As part of the launch process, a number of key lessons learnt relating to stamping had been identified and were captured for future programs. These had previously been created and stored in a simple excel spreadsheet, but the developed system was seen as an enhanced and more appropriate way to capture and integrate these in the product development lifecycle.

Model Improvements

The system replaced an existing paper-based new Model Improvement, or MI, process which was intended as a means for any employee in stamping to raise suggestions or proposals for changes or improvements to be incorporated in upcoming new vehicle model launches. Tracking improvements was difficult, and a complete absence of feedback had resulted in a lack of involvement from the shop floor. Allowing MI's to be captured, tracked and integrated was a key focus of the new system. Each MI represented a single case in the case base. This also meant the system had the business process required to support its use already in place, another key requirement.

Tooling and Design Changes

Issues with tooling, and the actions used to overcome them, in die tryout and production were captured using the system. Previously, much of this information had not been recorded, or only the final action for a change recorded. The system allowed the symptons and then all actions attempted to overcome the problem to be recorded. This is significant as it captures the intent behind a change, not only the final result.

Review and Integration

The organisation had a well-defined product lifecycle, with stage gate reviews at which specific elements of the design of a component had to be finalised and frozen before continuing to the next activity. To ensure experience captured in the system was integrated within this process, each case captured had a Review Timing assigned, which was the stage at which the case should be reviewed and potentially incorporated. A search of the system for a given stage gate was then included as part of the checklist for each stage gate, to ensure lessons, improvements or experiences were examined and included where appropriate and in accordance with the key design principles of integration and support.

5.4 Operation

Following the system's deployment, its operation was supported across the identified areas. Roles to support its use are outlined here, along with an example of use.

5.4.1 Use

As discussed previously the system was used across a number of areas. For each record, there were four distinct roles:

- **Initiator**, enter a new record

- **Controller**, initial review of a new record for clarity and appropriateness
- **Actioner**, examine the record in detail and identify action to be undertaken or implemented
- **Approver**, approve the action to be undertaken or recommended

Operationally, engineers or team supervision's typically acted as initiators and actioners, while senior management undertook controller and approver roles. The same person could have multiple roles within a single record, for example the initiator could be the same as the actioner.

Initiators were given discrection over when to capture a new record or experience within the specific business processes identified in the previous section, based upon the context of their work. Broad guidelines covering when to enter a new record were covered during training.

Once captured, each record was reviewed by a controller. This initial level of review examined the record for appropriateness, relevance and detail. If lacking, it could be returned to the initiator to be edited. If deemed appropriate, the record was assigned to an individual, the actioner, to assess and develop specific actions or recommendations, or review a change already made for wider implementation. This allowed the controller to determine who was the most appropriate person, this could be the initiator, another person from the same area or an individual in a different area.

The actioner's role was to examine the initial record and determine if further action was required, and if so what, or to review the change or improvement captured. In this way both the initial concern or suggestion was captured alongside additional action taken, to provide a complete history. The actioner would enter the recommendation or action into the system, which would then be approved. This provided management the opportunity to review and assess changes and recommendations for other factors such as cost and complexity. If not approved, the reasons why had to be justified and entered, to allow feedback to those involved with the record.

Within each role the database could be searched and interrogated to find solutions to previous problems. Searching was conducted using a number of fields from the data model:

- Part Name or Number
- Record Date
- Record Concern Type
- Record Description
- Record Timing

Matching cases were returned and any reuse or adaption of an existing solution for a new problem was left to the user.

5.4.2 Example

A particular issue had arisen during tryout from the preceeding model which affected part quality. The problem had been corrected by changing the blank dimensions and one of the dies in tryout. Marked-up panels from the experience had been kept and the operators involved were still working in the area. The initial concern and the change made were entered into the system as a Lesson Learnt, where if the same design was used again the part and die designers could incorporate the change made during tryout to ensure the same issue was not repeated.

The record was entered by the tryout supervisor against the individual part on which the problem occured, using photographs of the marked-up panel. The initial concern was recorded, along with the change that was made, and how it overcame the issue. The was reviewed by the controller, who was a manager in the Stamping and Structures - Stamping area. It was assigned to an engineer within that area who reviewed the record and recommended that the lesson should be incorporated for future product developments. A review timing was appended to the record, which indicated at which stage during the next product development cycle the lesson should be reviewed. The entire record, including both the initial lesson and the recommendation and timing, was reviewed by the approver, who in this instance was the same manager within Stamping and Structures - Stamping, who approved the recommendation. This was in effect saying the change made during tryout should be incorporated into future surrogates of the part, to ensure the same issue does not occur again during tryout.

During each stage, the original initiator was informed of progress and actions taken. The lesson will then be incorporated by searching the database through two keys fields, the individual part and the reviewing timing. This ensures the lesson is incorporated for not only the correct part but during the correct stage of the product lifecycle.

5.5 Conclusion

A case study covering the development and implementation of a software system for use in an automotive stamping plant is described in this chapter. This used an on-site, multi-disciplinary approach to develop a complete system through a number of prototypes specifically aimed at engaging the shop floor. Results from the use of the system, and a discussion of its benefits and limitations, are presented in the next chapter.

Chapter 6

Results and Discussion

Results from two years use of the system during the case study are presented and discussed in this chapter, together with an evaluation of the benefits and limitations identified. Observations and lessons learnt from the study that could be applied to similar manufacturing processes are outlined.

6.1 Results

Following the implementation and deployment described in chapter five, the system was used to support stamping design and production activities across two sites. This was driven by the organisation, supported by a subset of the initial development team including the researcher. The amount of time, personnel and areas involved with each activity during the development and deployment are presented in table 6.1. The researcher was involved in all these activities.

After two years of use following the initial deployment in die design and tryout, a snap-shot of the case-base was taken and analysed. The number of cases captured from each department, effectively representing each major activity during the product lifecycle, are shown in figure 6.1.

Figure 6.1: Number of records entered by department

6.2 Analysis and Discussion

Over the two year period of the case study a total of 354 individual cases were captured, representing almost three and a half per week. This was across five main areas illustrated in figure 6.1.

The fields in the record type data model were used to analyse the cases and generate metrics to quantify their focus and classification. The record fields used for analysis discussed in section 4.4.3 are listed again here:

Part/Process selected from the parts stored in the database

Date automatically generated from the day the record is entered

Operator captured when the user logs on

Concern Type selected from a predefined list of common one or two word issues stored in the system

Concern Description a free text field allowing the symptoms and possible causes of a concern or improvement to be entered

Activity	Involvement	Time (months)
Initial Discussions	6 employees	1
Version 1 Development		3
Demonstrations and Version 2 Development	44 employees	6
Version 2 Pilot	2 Areas	3
Final System Development		6
Deployment	6 areas	3
Case Study Use	6 areas	24

Table 6.1: Activity duration and involvement during the applications development and case study

Suggested or Recommended Action a free text field which may include any actual or temporary actions already completed

Other Action another free text field where an indication of whether the action entered was successful or implemented

Concern Location a set of coordinates on the base image of the part describing the location of the concern

Review Timing a stage gate milestone timing describing when the record should be reviewed and incorporated during the product lifecycle

Status of the record within the system, such as *initiated*, *actioned* or *closed*, to assist tracking

Images of the actual problem, which can include photos, sketches or screen shots

Annotations recorded on images, such as a limited amount of text, circles, squares and lines

The text and classifiers for each case were examined in detailed to generate the metrics in table 6.2. It was found that sigificant amounts of classification information can be extracted from the various free text fields in the record data model, specifically the *concern*, *feature* and *operation* of each case.

For the concern type field the operator would select an alternative from a list of predefined options before entering any text (see Appendix A for

Result	Number	Percentage
Total Records Entered	354	100%
Entered Under Part Number	68	19%
Mentions Part in Text Fields	135	38%
Case Related to Part Design	237	67%
Generic Concern	313	88%
Non-'Other' Concern Type	67	19%
Mentions Concern in Text Fields	231	65%
Concern Record in Type Field and/or Text Field	273	77%
Mentions Feature in Text Field	277	78%
Mentions Operation in Text Field	129	36%
Suggested Solution or Recommended Action Provided	306	86%

Table 6.2: Results of analysis of the case base after the first two years of use

the options avaiable). These options were generated from discussions with quality and shop floor personnel. This was supplemented by an *Other* option which the user could select. In only 19% of cases did the user select an alternative beyond *Other* (a non-Other option) from the list of concerns specified in the system. However, 65% of the cases made direct reference to a single word or short phase description of the concern within the text fields. In some cases, a word entered was available in the predefined list of options, although that option was not selected as the concern type.

A total 77% of cases contained a single word or short phase description of the concern the case related to in either the dedicated concern type field or the concern description free text field. This would suggest it would be effective to maintain an extensive list of concerns not displayed to the user to search through cases for classification and retrieval rather than display an excessively long list for the user to select from. This would also allow concern types with more than one name to be associated. For example, *Tears* and *Tear-Outs* could be associated within the application to refer to the same concern for the purpose of searching and retrieval, independent of which was entered by the user.

A similar conclusion can be seen from the part field. Only 19% of cases were classified under a specific, individual part number although double that number made reference to the part within at least one of the free text fields. The majority of cases were classified under one of the generic parts, representing processes or equipment (see Appendix A for a list of the generic parts available).

As a further potential classifier, 78% of cases made reference to one or more features in the free text fields of each case. These are typically geometric features such as *Holes* or *Trim Lines*.

Based upon the above results, it would be possible to extract and map different classifier information from the free text fields. This includes:

- features to concerns
- features to operations
- features to parts
- operations to parts

For example, for an individual part it would be possible to search across all cases and extract the features and operations entered by users. As features and operations are present across different parts, other cases with the same features or operations for different parts could be identified and retrieved to assist with design and problem-solving. The location of a concern on a base image for a case can be used to associate the location of the features described with a specific location on a part. This has the potential to construct simple feature maps for individual parts through use of the system rather than these having to be developed and entered manually. This would remove the set-up emphasis required for feature-based systems described by Chen, Y.-M. and Wei, C.-L. (1997); Cheok, B.T. and Nee, A.Y.C. (1998); Tang, D.-B. et al. (2001). Using these various mappings it may even be possible to construct relatively simple CMaps related to the design and manufacture of a part based upon the case-base.

Case reuse was also analysed. Broadly there were five key review stage gates during the new vehicle product development cycle. A relevant review timing can be set for each case, either by the operator who entered the initial case or someone assigned to assess the concern or improvement. This represents the

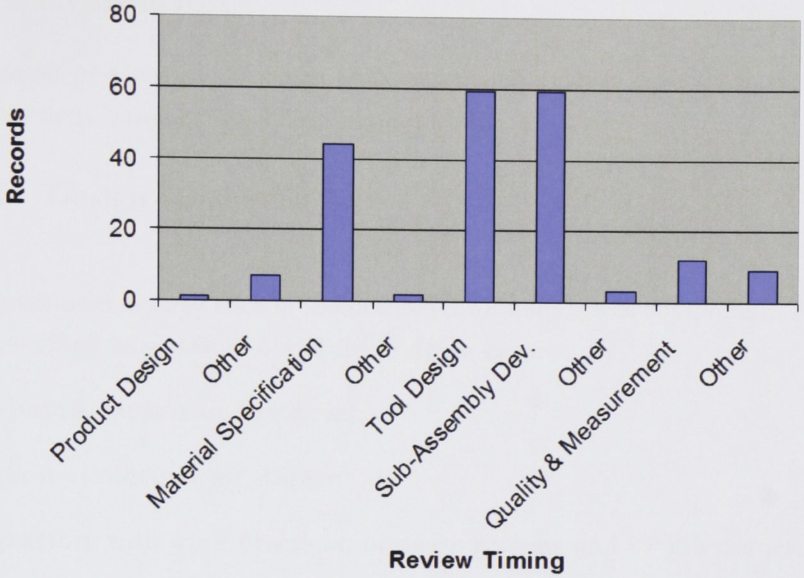

Figure 6.2: Number of records at each review timing

stage gate when the case should be reviewed for inclusion in a new vehicle or to ensure a concern is not repeated. The number of cases for each review timing are shown in figure 6.2. Within a development cycle, each review has a specific date, although an *Other* option was provided by the application which allowed a specific date to be set. By placing cases with *Other* between the dates for the formal stage gates, the review timings in figure 6.2 are in sequential order over a period of approximately two years.

As can be seen by comparing figures 6.1 and 6.2 very few cases impact directly on product (part) design. Rather, the majority of cases captured should be reviewed and potentially incorporated during material selection and detailed die and tool design, which includes gauge and fixture design. The bulk of the cases captured were potential improvements or lessons learnt, often to be reused within the activity where they were generated but during the next vehicle development process.

6.3 Evaluation

A number of areas of work will be reviewed to evaluate the system and its development based upon the discussion above.

6.3.1 Design Principles

An assessment against each of the design requirements was conducted. These are described in section 4.3 and listed again here:

task-based capture and retrieval

common structure for storage

integration with work practices, business systems and IT infrastructure

experience capture method which parallels the natural thinking approach

support only not replacing decision making

Task-Based

As seen from figure 6.1, records were captured across a number of areas. Incorporating different functions within modules of the system (outlined in Appendix A) enabled task-based case capture and retrieval across these various activities. However, in situations where administrative operators were assigned to enter cases, this resulted in an increase in incorrect classifications, as the operator was not familiar with the necessary detail.

From a review of the case-base, a number were incorrectly classified as generic parts rather than a specific part. This was likely the result of a lack of training, where operators were not directed to enter cases against individual parts. Further, despite the best efforts of the development team, in some situations administrative operators were assigned to enter records without having been involved in the initial concern or problem. These cases were often mis-classified as the operator did not have detailed first-hand experience.

Common Structure

Providing a common structure for cases in the form of the Records data model enabled different types of cases to be captured, including Model Improvements, Design or Tooling Changes and Lessons Learnt. In addition it allowed searching to take place across all records to summarise all relevant types of information.

Integration

In the first instance, integration with the organisation's IT infrastructure enabled the system to be deployed and its accessibility increased. Without this the system would not have be used in any significant way. Integration with an existing business process, where a module within the system replaced a paper-based system, also proved beneficial as a new process did not have to be developed and deployed, rather the system was seen as an improved way of managing the process.

Cases captured during the case study are reviewed and applied during the new vehicle model development process (see figure 6.2). By appending a stage gate review timing to each case, a report allows cases to be extracted that are relevant for a given timing. These are then reviewed as part of the process for that milestone, and if relevant incorporated or acted upon.

Experience Capture

The process of acquiring new cases, with a predominantly visual focus, was well received by users and seen as an appropriate way to capture tacit experience. Capturing experience from individual situations also paralled the natural problem solving approach and existing methods for remembering experiences and previous concerns.

Support Only

To achieve this, the system must be well integrated with existing business processes. Integrating with existing business processes at the organisation where the case study was conducted proved successful. The system was providing information into an existing decision making process. Linking retrieval and reuse with stage gate review timings was also invaluable as decisions were made and finalised at these times based upon information from numerous sources.

6.3.2 Development Methodology

One of the two key requirements raised by Procaccino, J. D. et al. (2002) is the level of confidence that users have in the development team. This is related to the problem identified by Saiedian, H. and Dale, R. (2000) that projects often fail as developers do not fully understand or address the requirements of the user and their environment. A lack of understanding of the application and context for which software is being developed will decrease the confidence users have in the development team and hence increase the risk of an unsuccessful implementation. By assembling a development team with engineering, manufacturing and software skills, and conducting significant reviews and testing within the plant, the development team were able to build up knowledge of the plant and its operations. Further, the development team was seen within the plant, which helped gain the confidence of potential users, which was the first step towards gaining confidence in the system.

Through the rapid prototyping and pilot activities (listed in table 6.1), potential users were able to have significant input into the appearance and functionality of the final system. This ensured the system was filling an actual need for employees within the organisation. It also created a strong sense of ownership of the system. The project did not suffer from departmental barriers, as the team was external and not associated with a single department or area.

6.3.3 System Benefits

The main benefit from use of the system was the capture of experience and potential improvements in a form that could be reviewed and understood by employees not involved in the initial concern. It was found the two key elements to aid this were the visual elements and the review timing for each case. This use of a review timing means cases can be continually reused during the development of any new vehicle.

6.3.4 Limitations

As identified, the development methodology lead to an increase in use and acceptance of the final application. However, this approach could only be employeed as a new system was being developed. A different approach would be required for implementing an existing system, which is examined in more detail in Pantano, V. et al. (2002). Having a development team that spends extended periods on-site during deployment and initial use can quickly become costly (see table 6.1).

To be used on an ongoing basis, knowledge-based systems may need to be supported by other cultural and organisational changes, depending on the scope and signifiance of the new technology (Cooper, R. B., 1994; Pantano, V. et al., 2002). Management strategies to consider such impacts and assist implementation and use are discussed in detail by Pantano, V. (2004).

Although individual cases can be entered in a relatively short period of time, it still takes time to build up the case-base. Replacing elements of existing systems and business processes, as was done during the case study, can help decrease the time required.

6.3.5 Observations and Lessons Learnt

Throughout the design, development and use of the system a number of lessons were learnt for future similar developments or implementations. First and foremost, the use of an interdisciplinary team, in terms of both skills and backgrounds, was particularly beneficial. It allowed discussions and meetings to be tailored and to use the person with the most experience for a given situation. In-plant development and support was also highly valuable, as it gave potential users the view that the team, and hence the software, was there to help them. The approach described in chapter three proved successful, and has the potential to be applied in other applications.

During the course of the study, they was a dramatic change in computer use and acceptance. For example, early training sessions often consisted of how to use a digital camera and a mouse, while towards the end virtually all users were initially comfortable with a computer.

One of the most important factors identified during the research was the need for continuing management support. This was also observed by Procaccino, J. D. et al. (2002) who stressed the presence of a committed sponsor as one of the two key factors required for success of a software development project. During the launch, plant champions were selected to assist implementation of the system and displayed a commitment from the organisation in the system's use. Involvement and input from the local IT groups were also particularly important, to ensure integration, maintenance and compilance.

As expected, the use of administrative operators for entry in some areas lead to misclassification and less detailed case descriptions. Wherever possible, such involvement should be discouraged.

6.4 Conclusion

This chapter provided an evaluation of the system over a two year case study. Benefits, limitations and lessons learnt from the development and deployment process were highlighted. The next chapter will outline final conclusions from the study as a whole.

Chapter 7

Conclusions and Future Work

This chapter presents the final conclusions from the study summarising the key research aims and outcomes from each chapter. Potential further work and extensions to the system and its value to other applications are also outlined.

7.1 Summary

Chapter 1 outlined the motivation for the study, to remedy the significant issue of knowledge loss from the manufacturing shop floor. This is often in the form of implicit tacit experience built-up over years of trial and error situations which is used to overcome issues with designs and tooling.

Chapter 2 identified a clear gap in capturing and integrating this experiential knowledge from the manufacturing shop floor which is related to a number of technical and non-technical requirements.

Chapter 3 presented an approach to the development of a system to capture this experience, utilising a bottom-up rapid prototyping methodology, undertaken by a multi-disciplinary team to assist user buy-in and acceptance.

Chapter 4 detailed a set of design principles that a system to capture and integrate manufacturing experience from the shop floor must satisfy. A framework for a specific application encapsulating these principles for the automotive stamping industry was outlined.

Chapter 5 detailed a case study covering the development and implementation of a new software system aimed specifically at capturing and integrating manufacturing experience from the shop floor, in order to overcome knolwedge loss, decrease lead time and improve part quality.

Chapter 6 analysed the results of the case study after two years of use. It demonstrated that the capture interface developed, which paralled the natural problem-solving approach, resulted in more detailed descriptions of experience, without an unduly significant increase in time to capture them. During the case study an experience base was built up for reuse during future product development activities, to reuse lessons learnt and incorporate potential new improvements.

7.2 Conclusions

The loss of important experiential knowledge from the manufacturing shop floor during the product development lifecycle was identified and highlighted. In response, the study identified and developed a set of design principles which a software system must encapsulate in order to capture and integrate key experience from these activities. A system that encapsulated these aims was developed and implemented at the stamping plant of the Australia arm of a multi-national automotive manufacturer as a case study in order to assess its effectiveness.

It was found that the design principles identified, and their inclusion in the system developed, resulted in more efficient capture of experience by paralleling the natural problem-solving and thinking approach of its users. This allowed capture to be completed in a relative timely manner, while still providing the ability for reuse and extraction from the case-base created. The software development methodology used also generated a demand from its potential users and created a sense of ownership, which assisted implementation and use of the system.

It was found that many of the experiences captured related to improvements and lessons learnt. This were utlised during future product development lifecycles, to incorporate improvements and reduce repeated concerns. Both can contribute to a decrease in product development lead-time, in addition to storing experiences for future reuse by the organisation.

This study confirmed the importance of ongoing management support in addition to technical requirements. Including reports and metrics for management activities was benefical to support use of the system beyond its initial development and implementation.

7.3 Future Work

Future work in terms of both new functionality of system and applications in other industries have been identified.

7.3.1 Application Extensions

From the analysis of the results it was established that classifiers could be extracted from free text fields. More advanced text searching algorithms could be introduced to extract these details, such as those described in Ruiz-Sánchez, J. M. et al. (2003). More recent work has also examined folknomics to generate and compare classifiers rather than using explicit lists of classifier options (Sinclair, J., 2006). Implicit relationships between components, concerns, features and operations could be identified from cases and used to automatically generate simple concept maps, which could then be presented to users for modification or used in the training of new operators or engineers.

Another area for work is the inclusion of 3D models, particularly CAD geometries and the results of FE simulation work. Combining the case-base with geometric feature recognition and extraction such as the method described in Ferrari, D. et al. (2006) would allow CAD models of a new tool to be uploaded into the system and key geometric features identified. Features could then be used to automatically search through existing cases and present relevant experience, improvements or problems to the designer and to ensure previous mistakes with a new design are not repeated. This would enable the system to become predictive, rather than only reactive and be used to identify potential problems rather than present solutions to previous concerns.

7.3.2 Further Applications

The system developed in this study remains in use by the organisation involved in the case study. Other applications within that organisation have been recognised and such systems implemented.

The principles identified and the system development could be applied to similar manufacturing industries. The issues of knowledge loss and acquiring experience and feedback from the shop floor are common to most manufacturing organisations. With more mobile work-forces and the need for more flexible and responsive product development these issues are becoming increasingly significant. Presentations of this research to other industries, including steel making, forging, casting, general assembly and non-automotive stamping, have indicated the system developed here could have value elsewhere.

The visually based capture interface developed, which was found to be particular effective, could be incorporated into other systems to assist case or experience capture, whilst employing other methodogies for storage and retrieval.

Appendix A

Application Description

The system developed was separated into functional modules, listed in table A.1. Within each of these was a specific set of related functions which the user selected to enter or retrieve information or details, depending upon their exact task. The functions for each module are shown in figure A.1. Each user has an unique identifier which they enter when they log on to the system and this allows access to functions to be controlled for individual users.

Module
Model Improvement (MI)
Issues
Noticeboard
Reporting
Summary
Administration

Table A.1: System modules

Generic Name
Blanks
Designs
Dies
Doors
Parts
Processes
Tolerances

Table A.2: Generic parts used to supplement individual components

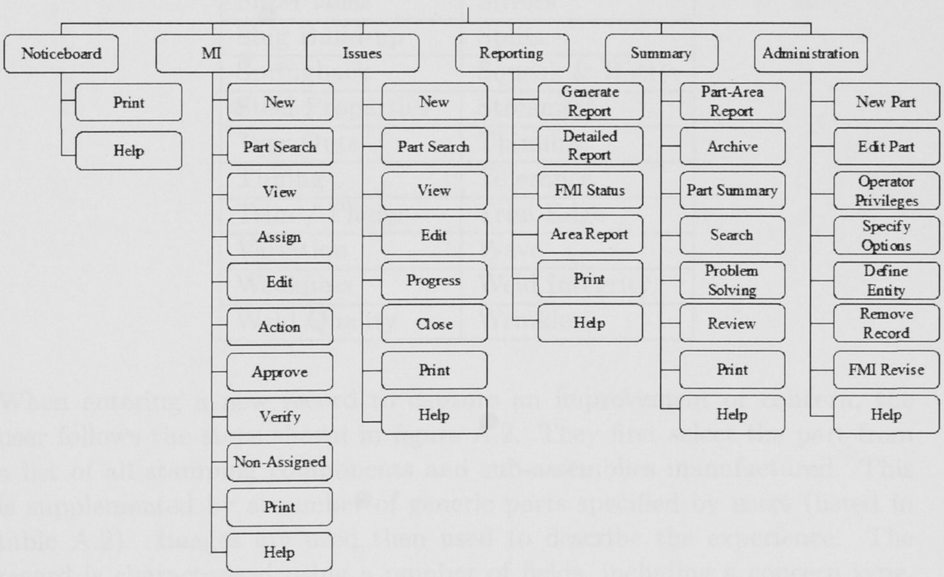

Figure A.1: Functions available within each module

Option	Option
Bedding	Bleed Out
Broken Finger	Broken Punch
Buckles	Burrs
CAD	Closing Effort
Co-ordination	Contamination
Craftsmanship	Damage
Design	Dunnage
Flange Clearance	Flushness
Fouling	Gauge
Gripper	High
Interference	Lifter
Location Pins	Loose Metal
Low	Measurement
Metal Folds	Misalignment
Misfeed	Mislocation
Overcrown	Pimples
Poor Margin	Repeatability
Safety	Scores
Scrap Build-up	Sensor
Slip Planes	Slivers
Slug Build-up	Splits
Springback	Squeak & Rattle
Steel Properties	Streamers
Tear Outs	Thinning
Timing	Tolerance
Trim / Flange	Trim Edge
Variation	Wave
Weakness	Weld Integrity
Weld Quality	Wrinkles

When entering a new record to capture an improvement or concern, the user follows the steps shown in figure A.2. They first select the part from a list of all stamping components and sub-assemblies manufactured. This is supplemented by a number of generic parts specified by users (listed in table A.2). Images are used then used to describe the experience. The record is characterised using a number of fields, including a concern type, selected from a pre-defined list developed by quality and key plant personnel (the options are shown in table A.3). Finally text descriptions are typed to supplement the other information entered.

Table A.3: Concern type options

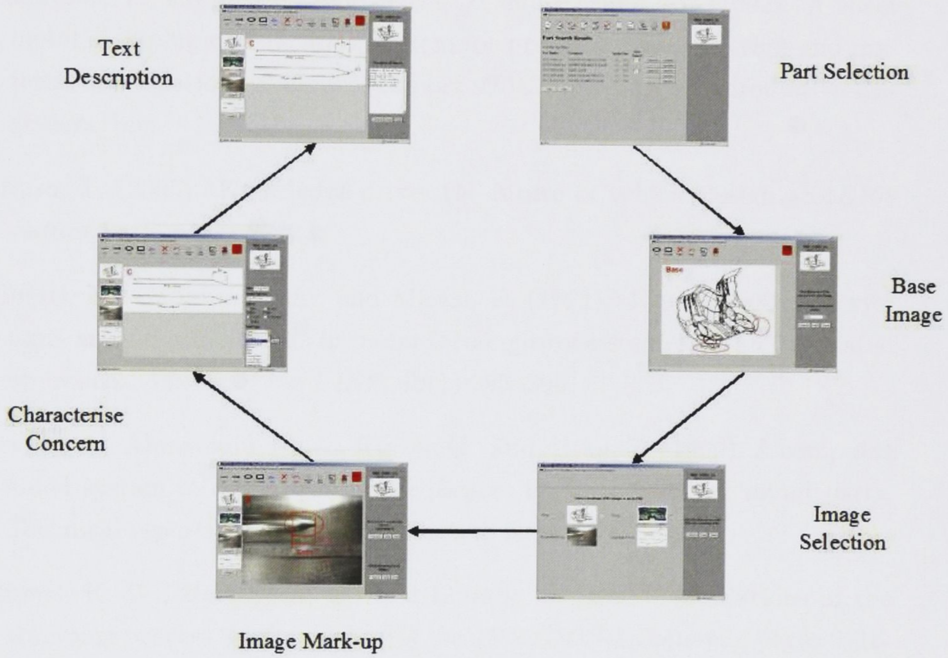

Figure A.2: Major steps in entering a new record

Bibliography

Aboutour, T., Hook, T., Kolodziejski, J., and Karima, M. (1992). A sheet metal stamping classification system for product/process design and material specification. Technical Report 920430, Society of Automotive Engineers, Inc.

Affuso, T. (2005). Knowledge drives the future of vehicle design. *SAE100 Future Look*, pages 22–23.

Alberti, N., Cannizzaro, L., and Micari, F. (1991). Knowledge-based systems and f.e. simulations in metal-forming processes design an integrated approach. *Annals of the CIRP*, 40(1):295–298.

Arvind, A., Ahmetoglu, M. A., Kinzel, G., and Altan, T. (1999). A computer aided system to design process sequences for deep drawing round parts. Technical report, Society of Automotive Engineers, Inc.

Blümel, K. W., Hartmann, G., and Lübeck, P. (1988). Vairiations of the stamping process. In *Proceeding of the 15th IDDRG Congress*, pages 9–16.

Boehm, B. W. (1988). A spiral model of software development and enhancement. *IEEE Computer*, 21(5):61–72.

Bracewell, R. H., Ahmed, S., and Wallace, K. M. (2004). Dred and design folders, a way of capturing, storing and passing on, knowledge generated during design projects. In *Proceedings of DETC'04 ASME 2004 Design Engineering Technical Conferences and Computers and Information in Engineering Conference, September 28-October 2, 2004, Salt Lake City, Utah, USA*, number DETC2004-57165, Salt Lake City. ASME.

Cañas, A., Leake, D. B., and Maguitman, A. (2001). Combining concept mapping with cbr: Towards experience-based support for knowledge modeling. Technical report, American Association for Artificial Intelligence.

Cao, J. and Boyce, M. C. (1997). A predictive tool for delaying wrinkling and tearing failures in sheet metal forming. *Transactions of the ASME*, 119:354–365.

Chapman, C. B. and Pinfold, M. (2001). The application of a knowledge based engineering approach to the rapid design and analysis of an automotive structure. *Advances in Engineering Software*, 32:903–912.

Chen, Y.-M. and Wei, C.-L. (1997). Computer-aided feature-based design for net shape manufacturing. *Computer Integrated Manufacturing Systems*, 10(2):147–164.

Cheok, B.T. and Nee, A.Y.C. (1998). Trends and developments in the automation of design and manufacture of tools for metal stampings. *Journal of Materials Processing Technology*, 75:240–252.

Cooper, R. B. (1994). The inertial impact of culture on it implementation. *Information and Management*, 27:17–31.

Cunningham, P. and Bonzano, A. (1999). Knowledge engineering issues in developing a case-based reasoning application. *Knowledge-Based Systems*, 12:371–379.

Damodaran, L. and Olphert, W. (2000). Barriers and facilitators to the use of knowledge management systems. *Behaviour & Information Technology*, 19(6):405–413.

Deslandres, V. and Pierreval, H. (1997). Knowledge acquisition issues in the design of decision support systems in quality control. *European Journal of Operational Research*, 103:296–311.

Dinda, S., Keeler, S. P., James, K. F., and Stine, P. A. (1981). How to use circle grid analysis for die tryout. Technical report, American Society for Metals.

Dutta, S. (1997). Strategies for implementing knowledge-based systems. *IEEE Transactions on Engineering Management*, 44(1):79–90.

Elliott, J. J. (2000). Design of a product-focused customer-oriented process. *Information and Software Technology*, 42:973–981.

Fakun, D. and Greenough, R. M. (2002). User-interface design heuristics for developing usabel industrial hypermedia applications. *Human Factors and Ergonomics in Manufacturing*, 12(12):127–149.

Fallböhmer, P., Altan, T, Tönshoff, T., and Nakagawa, T. (1996). Survey of the dies and mold manufacturing industry. *Journal of Materials Processing Technology*, 59:158–168.

Ferrari, D., Rolfe, B., West, G., and Cardew-Hall, M. (2006). Free-form feature representation and comparison. In *Proceedings of IDETC/CIE 2006 ASME 2006 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, September 10-13, 2006, Philadelphia, Pennsylvania, USA*, number DETC2006-99066, Philadelphia. ASME.

Ghezzi, C., Jazayeri, M., and Mandrioli, D. (1991). *Fundamentals of Software Engineering*. Prentice-Hall, USA.

Gonzalez, R. and Kamrani, A. (2001). A survey of methodologies and techniques for data mining and intelligent data discovery. In Braha, D., editor, *Data Mining for Design and Manufacturing*, pages 41–60. Kluwer Academic Publishers.

Greska, W., Franke, V., and Geiger, M. (1997). Classification problems in manufacturing of sheet metal parts. *Computers in Industry*, 33:17–30.

Hindman, C. and Ousterhout, K. B. (1998). A virtual design system for sheet metal forming. *Journal of Materials Processing Technology*, 84:107–116.

Hong, J., Toye, G., and Leifer, L. J. (1996). Engineering design notebook for sharing and reuse. *Computers in Industry*, 29:27–35.

Hu, P. J.-H., Ma, P.-C., and Chau, P. Y. K. (1999). Evaluation of use interface designs for information retrieval systems: a computer-based experiment. *Design Support Systems*, 27:125–143.

Huang, S. H., Xing, H., and Wang, G. (2001). Intelligent classification of the drop hammer forming process method. *The International Journal of Advanced Manufacturing Technology*, 18:89–97.

Hull, M. E. C., Taylor, P. S., Hanna, J. R. P., and Millar, R. J. (2002). Software development processes – an assessment. *Information and Software Technology*, 44:1–12.

Imamura, S., Kojima, T., Sekiguchi, H., and Inoue, K. (1988). A study on the object oriented product model – representation of geometry and dimension. *Annals of the CIRP*, 37(2):127–130.

Jiang, J. and Klein, G. (2000). Software development risks to project effectiveness. *Journal of Systems and Software*, 52:3–10.

Kalpakjian, S. and Schmid, S. (2006). *Manufacturing Engineering and Technology*. Pearson Education South Asia Pte Ltd, fifth edition.

Keeler, S. P. (1977). Sheet metal stamping technology - need for fundamental understanding. *Journal of Materials Processing Technology*, pages 3–18.

Keeler, S. P. (1988). Fifty years of sheet metal formability - has science replaced the art? Technical report, Society of Automotive Engineers, Inc.

Kim, K.-J. and Han, I. (2001). Maintaining case-based reasoning systems using a genertic algorithms approach. *Expert Systems with Applications*, 21:139–145.

Lapré, M. A. and Van Wassenhove, L. N. (2001). Creating and transferring knowledge for productivity improvement in factories. *Management Science*, 47(10):1311–1325. INFORMS 2001.

Lau, H. C. W., Wong, C. W. Y., Hui, I. K., and Pun, K. F. (2003). Design and implementation of an integrated knowledge system. *Knowledge-Based Systems*, 16:69–76.

Leake, D. B., Birnbam, L., Hammond, K., Marlow, C., and Yang, H. (1999a). Integrating information resources: A case study of engineering design support. In K.-D. Althoff, R. B. and Branting, L. K., editors, *Proceedings of ICCBR-99*, pages 482–496. Springer-Verlag Berlin.

Leake, D. B., Birnhaum, L., Hammond, K., Marlow, C., and Yang, H. (1999b). Integrating diverse information resources in a case-based design environment. *Engineering Applications of Artificial Intelligence*, 12:705–716.

Leake, D. B. and Wilson, D. C. (1999). Combining cbr with interactive knowledge acquisition, manipulation and reuse. In K.-D. Althoff, R. B. and Branting, L. K., editors, *Proceedings of ICCBR-99*, pages 203–217. Springer-Verlag Berlin.

Lee, K. S. and Luo, C. (2002). Application of case-based reasoning in die-casting die design. *The International Journal of Advanced Manufacturing Technology*, 20:284–295.

Lei, Y., Peng, Y., and Puan, X. (2001). Applying case-based reasoning to cold forging process planning. *Journal of Materials Processing Technology*, 112:12–16.

Lewis, K. (2002). Holistic km at the department of health. *Knowledge Management*, 6(4):16–19.

Liu, H., Yuan, J. L., and Jin, J. (2002). Visualisation simulations for a cold press die. *Journal of Materials Processing Technology*, 129:321–325.

Mares, E. L. and Sokolowski, J. H. (2003). Design, integration and evaluation of an artifical intelligence-based control system for the improvement of the monitoring and quality control process in the manufacturing of metal casting compomnents. In Palade, V., H. and Jain, L. C., editors, *KES 2003*, pages 80–86. Springer-Verlag Berlin.

Mathworks (2007). www.mathworks.com/products/matlab/index.html?ref=pfo. Last Accessed, March 30, 2007.

McEvily, S. K. and Chakravarthy, B. (2002). The persistence of knowledge-based advantage: An empirical test for product performance and technological knowledge. *Strategic Management Journal*, 23:285–2003.

Mechitov, A. I., Moshkovich, H. M., Olson, D. L., and Killingsworth, B. (1995). Knowledge acquisition tool for case-based reasoning systems. *Expert Systems with Applications*, 9:201–212.

Mejasson, P., Petridis, M., Knight, B., Soper, A., and Norman, P. (2000). Intelligent design assistant (ida): a case base reasoning system for material and design. *Materials & Design*, 22:163–170.

Mikler, J., Høadeby, H, Kjellberg, A, and Sohlenius, G (1999). Towards profitable persisten manufacturing. human factors in overcoming disturbances in production systems. *The International Journal of Advanced Manufacturing Technology*, 15:749–756.

Mileman, T., Knight, B., Petridis, M., Cowell, D., and Ewer, J. (2002). Case-based retrieval of 3-dimensional shapes for the design of metal castings. *Journal of Intelligent Manufacturing*, 13:39–45.

Ndlela, L. T. and du Toit, A. S. A. (2001). Establishing a knowledge management programme for competitive advantage in an enterprise. *International Journal of Information Management*, 21:151–165.

Nedeß, Chr. and Jacob, U. (1997). A case-based reasoning approach towards learning from experience connecting design and shop floor. *Computers in Industry*, 33:127–137.

Noh, J. B., Lee, K. C., Kim, J. K., Lee, J. K., and Kim, S. H. (2000). A case-based reasoning approach to cognitive map-driven tacit knowledge management. *Expert Systems with Applications*, 19:249–259.

Pantano, V. (2004). *An Interdisciplinary Model of International Technology Transfer*. PhD thesis, The Australian National University.

Pantano, V., Cardew-Hall, M. J., and Hodgson, P. D. (2002). Technology diffusion and organisational culture: Preliminary findings from the implementation if a knowledge management system. In *IEEE IEMC-2002, Conference Proceedings*, pages 1–6. IEEE.

Park, M.-K., Lee, I., and Shon, K.-M. (1998). Using case based reasoning for problem solving in a complex production process. *Expert Systems with Applications*, 15:69–75.

Pomerol, J.-C. and Brézillon, P. (2001). About some relationships between knowledge and context. In et al., V. A., editor, *Proceedings of CONTEXT 2001*, pages 461–464. Springer-Verlag Berlin.

Procaccino, J. D., Verner, J. M., Overmyer, S. P., and Darter, M. E. (2002). Case study: factors for early prediction of software development success. *Information and Software Technology*, 44:53–62.

Rudolph, S. and Hertkorn, P. (2001). Data mining in scentific data. In Braha, D., editor, *Data Mining for Design and Manufacturing*, pages 61–86. Kluwer Academic Publishers.

Ruiz-Sánchez, J. M., Valencia-García, R., Fernández-Breis, J. T., Martínez-Béjar, R., and Compton, P (2003). An approach for incremental knowledge acquisition from text. *Expert Systems with Applications*, 25:77–86.

Saiedian, H. and Dale, R. (2000). Requirements engineering: making the connection between the software developer and customer. *Information and Software Technology*, 42:419–428.

Sethi, I. K. (2001). Data mining: An introduction. In Braha, D., editor, *Data Mining for Design and Manufacturing*, pages 1–40. Kluwer Academic Publishers.

Shi, F., Lou, Z. L., Lu, J. G., and Zhang, Y. Q (2003). An improved rough set approach to design of gating scheme for injection moulding. *The International Journal of Advanced Manufacturing Technology*, 21:662–668.

Sinclair, J. (2006). Categorisation in knowledge contexts. Mid-term review, The Australian National University, Department of Engineering.

Spath, D., Sternemann, K.-H., Selinger, G., and Ruoff, E. (2001). Context sensitive provision of enterprise information. *Annals of the German Academic Socity for Production Engineering*, 8(1):73–76.

Stine, P. A. and Keeler, S. P. (1990). Sheet metal formability – comparsions between computer modeling and production case histories. Technical Report 900276, Society of Automotive Engineers, Inc.

Su, D. and Wakelam, M. (1998). Intelligent hybrid system for integration in design and manufacture. *Journal of Materials Processing Technology*, 76:23–28.

Suri, Rajiv, Otto, Kevin N., and Van Buren, Rachel A. (1998). Models to integrate sheet-metal part and process design. In *31st CIRP International Seminar on Manufacturing Systems*, pages 1–6. Annals of the CIRP.

Tang, D.-B., Zheng, L., and Li, Z.-Z. (2001). An intelligent feature-based design for stamping system. *The International Journal of Advanced Manufacturing Technology*, 18:193–200.

tcl/tk (2007). www.tcl.tk. Last Accessed, March 30, 2007.

Thomas, W. and Altan, T. (1998). Application of computer modelling in part, die, and process design for manufacturing of automotive stampings. *Steel Research*, 69(4+5):181–187.

Thomas, W. and Altan, T. (1999). Implementing computer simulation into the concept to product process. Technical Report 1999-01-1003, Society of Automotive Engineers, Inc.

Vinarcik, E. J. (1999). The reality of problem solving. Technical Report 1999-01-1626, Society of Automotive Engineers, Inc.

Wagner, K. and Aslanidis, S. (2001). Utilisation of experience and know-how in the product development process. Technical Paper Series 2001-01-3436, Society of Automotive Engineers, Inc. Copyright SAE International and Messe Dusseldorf.

Weber, R., Aha, D. W., and Becerra-Fernandez, I. (2001). Intelligent lessons learned systems. *Expert Systems with Applications*, 17:17–34.

West, A. A., Bowen, B. A., Monfared, R. P., and Hodgson, A. (1999). User-responsive interface generation for manufacturing systems: a theoretical basis. Technical report.

Xia, Q. and Rao, M. (1999). Knowledge architecture and system design for intelligent operation support systems. *Expert Systems with Applications*, 17:115–127.

Yang, S. and Nezu, K. (1997). A knowledge-based approach to concurrent design of sheet metal forming product and process and its preliminary implementation. In *Proceedings of DETC'97 ASME Design Engineering Technical Conferences, September 14-17, 1997, Sacramento, California*, number DETC97/CIE-4278, Sacramento. ASME.

Yoshioka, M. and Shamoto, Y. (2003). Knowledge management system for problem solving - integration of document information and formalised knowledge -. In *Proceedings of DETC'03 ASME Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Chicago, Illinois, USA, September 2-6, 2003*, number DETC2003/CIE-48217, Chicago. ASME.

Zhang, Z. T., Monette, D. G., Monroe, R. J., and Dailey, B. L. (1999). Establishment of stamping process windows. Technical Paper Series 1999-01-0687, Society of Automotive Engineers, Inc. Reprinted From: Advances in Sheet Metal Forming (SP-1435).