Using a knowledge management approach to support quality costing

Steve Eldridge and Mohammed Balubaid
Manchester Engineering Doctorate Centre, The University of Manchester, Manchester, UK, and
Kevin D. Barber
School of Management, The University of Bradford, Bradford, UK

Abstract

Purpose – The purpose of this article is to examine the difficulties associated with quality costing and propose a solution based upon the use of knowledge management techniques.

Design/methodology/approach – A widely available software tool is used to create a quality costing ontology based upon the prevention appraisal failure classification of quality costs. This ontology is used for the collection, processing, sharing and use of quality cost-related knowledge. The ontology was evaluated using case study data and compared with conventional approaches to quality costing.

Findings – The quality costing ontology is easier and more efficient than conventional quality costing methods. It has greater capability in terms of the analysis and use of quality costing knowledge and overcomes the barriers to quality costing due to poor understanding and awareness.

Research limitations/implications – The quality costing ontology provides a platform for researchers to investigate quality cost behaviour within a well-structured environment. The use of alternative classifications to prevention appraisal failure in the ontology need to be investigated further.

Practical implications – Traditionally, training and education have been used to rectify poor understanding and awareness of quality costs but with limited success. The quality cost ontology provides an alternative solution that uses knowledge management (KM) technology and is based on information systems.

Originality/value – The authors could find no research or published paper that has discussed the role of KM in quality costing.

Keywords Quality costs, Knowledge management

Paper type Research paper

Introduction

There is widespread acceptance that quality is an important strategic competence and a key competitive weapon that cannot be ignored by any organisation. The implementation of a quality improvement process can result in substantial cost savings and higher revenues and, consequently, organisations invest considerable sums of money every year on implementing and sustaining quality programmes and initiatives. One of the most important techniques in quality management is quality costing. Quality costing is seen as a means of helping companies to reduce manufacturing costs by identifying excessive cost and non-value adding activities (e.g. Carson, 1986; Israeli and Fisher, 1995; Johnson and Kleiner, 1993). Ignoring it can make

International Journal of Quality & Reliability Management
Vol. 23 No. 1, 2006
pp. 81-101

0265-671X
DOI 10.1108/02656710610637569

goods and services more expensive, which affect competitiveness, salaries, jobs and standard of living.

Despite an increased awareness of the importance and benefits of quality and quality costs, concerted efforts to improve quality and reduce the costs of quality have not always gone well. Several surveys (e.g. Superville and Gupta, 2001; Kumar *et al.*, 1998; Lascelles and Dale, 1990) of different organisations reveal that few of the respondents calculated their quality costs. The most surprising results are revealed by a Government Accounting Office (GAO) study reported by Superville and Gupta (2001) which finds that only one in four finalists of the prestigious Malcolm Baldrige National Quality Award calculated their quality costs. On the other hand, quality costs are very large; a recent study by Dale and Plunkett (1999) claims that quality-related costs commonly range from 5 to 25 per cent of a company's annual sales turnover. These various studies raise the question of why these companies are not calculating quality costs.

Following a review of the quality costing literature (e.g. Machowski and Dale, 1998; Kumar *et al.*, 1998; Dale and Plunkett, 1999; Roden and Dale, 2000), it appears that there are a number of practical difficulties with quality costing and that these difficulties are not specific to one industry sector or size of organisation. For example, it is clearly important for organisations that use a quality costing system to have a clear understanding of the concept of quality costing and remove the confusion which exists at all levels of the organisational hierarchy over the terms used in quality costing. Also, it can be argued that the reason why many organisations do not collect quality costs is the lack of understanding and awareness of the concept of quality costing. In summary, it appears that organisations do not collect quality costs because of the following five difficulties:

(1) lack of understanding and/or awareness of the concept and principles of quality costing among the management team;

(2) company culture;

(3) an acute lack of information and data;

(4) the confusion between the levels of the organisational hierarchy over the terms used in quality costing; and

(5) inefficiency of the accounting information system, which prevents firms from providing quality cost data.

Traditional solutions to some of these difficulties (e.g. points "1", "2" and "4") have focused on training. However, research has shown that training was, at best, a short-lived solution to the problem of lack of understanding of the quality costing concept. For example, in the examination of a company's quality costing knowledge carried out by Machowski and Dale (1998), they found that although all levels of management including supervisors had been exposed to quality management training, which outlined quality costing concepts such as the PAF model, some confusion and lack of understanding still existed. Machowski and Dale (1998) go on to make two observations about the training:

> First it does not appear to have had a substantial impact on those who attended the training course, and second, it was not filtered down by managers and supervisors to other employees within the organisation.

Also, training can be expensive and smaller companies may not be able to afford it. Finally, the training solution does not overcome the systems-based or "hard" issues (e.g. points "3", "4" and "5") associated with quality costing in management information systems. The ineffectiveness of training alone justifies the need for the provision of useful tools that can support quality costing and relieve the confusion and understanding problems that arise in organisations.

On the other hand, embedding quality costing within the organisation's established management information system provides an opportunity to address points "1", "2" and "3" in the summary of difficulties using a systems solution. Standardising on a common language and terminology using a knowledge classification/ontology-based approach would address point "4" whilst efficient system design would address point "5". These potential solutions all lie within the realm of knowledge management. Consequently, knowledge management could provide a solution for quality costing that is technology-based rather than based on a social model using training and educational approaches. This paper examines how knowledge management can help to support quality costing.

Knowledge management

Knowledge management (KM) has become a critical and popular subject of discussion in the business and academic literature in the last decade according to Bhatt (2001). Most research (e.g. Davenport *et al.*, 1998; Prusak, 2001; Lee and Hong, 2002) agrees that KM is an important and necessary component for organisations to survive and as a competitive advantage. In addition, KM is now accepted and is widely practiced and utilised in many world-class organisations where its usage has realised benefits.

Davenport *et al.* (1998) suggest that knowledge is information combined with experience, context, interpretation, and reflection. In the literature there are many different approaches to defining knowledge and these offer several different perspectives on the nature of knowledge. However, most authors accept that knowledge within an organisation falls into two categories, namely, explicit knowledge and tacit knowledge. Duffy (2000) describes tacit knowledge as residing in the human mind, behaviour, and perception. It evolves from people's interactions and requires skill and practice. He describes explicit knowledge as documented and public knowledge and this is what can be captured and shared through information technology. It can be expressed in words and numbers and shared in the form of data, scientific formulae, specifications and manuals. It is stored in databases and used and accessed and used by employees in the company as described by Civi (2000). Nonaka (1994) proposes four modes of transferring knowledge:

(1) Socialisation (tacit to tacit), through coaching and on-the-job training.

(2) Internalisation (explicit to tacit), learning from the analysis of explicit knowledge.

(3) Externalisation (tacit to explicit), the articulation of tacit knowledge into procedures or reports that attempt to document experience in context.

(4) Combination (explicit to explicit), the combination several elements of explicit knowledge into summary reports.

From the perspective of quality costing, the key knowledge conversion processes would be externalisation tacit to explicit and combination explicit to explicit. An integral feature of these conversion processes would be the use of an ontology to standardise the language and terminology of quality costing and capture knowledge. According to Neches *et al.* (1991):

> Ontology is an important discipline that has a huge potential to improve information organisation, management and understanding. It has a crucial role to play in enabling content-based access, interoperability and communications. In the literature ontologies have been proposed as a specification mechanism to enhance knowledge sharing and reuse across different applications.

The difficulty in differentiating between data, information and knowledge has caused confusion among researchers and is the reason that has caused many companies to regard knowledge management as synonymous with information management or document management (e.g. Gupta, 2000; Liebowitz, 2001). Bhatt (2001) argues that defining data, information, and knowledge is difficult, and only through a user's perspective can one distinguish between data, information, and knowledge. In general, data are considered as raw facts, information is regarded as an organised set of data, and knowledge is perceived as meaningful information. Within quality costing, there are instances or activities that incur quality costs (i.e. data), there are reports (i.e. information) and there are investigations analysis and actions (i.e. knowledge). Despite the apparent similarities in approach, the authors could find no research or published paper that has discussed the role of KM in quality costing or linked KM to quality costing.

Research method

The technologies used in KM can be divided into four main parts:

(1) Technologies for knowledge gathering.

(2) Technologies for knowledge transformation and processing.

(3) Technologies for knowledge sharing and use.

(4) Technology for knowledge lifecycle maintenance.

For the application to quality costing, the initial focus was placed on the technologies for gathering, transformation and processing, sharing and use. As mentioned earlier, this requires the development of an appropriate ontology. The foundation for the approach proposed in this paper uses an established software tool in order to enable the standardisation of language and efficient system design. Protégé-2000 was chosen for the application to quality costing. Protégé-2000 is described by Grosso *et al.* (1999) as an ontology-development and knowledge-acquisition software tool that is designed to make it easier for experts to maintain and edit knowledge bases. It allows the user to:

- Construct specific ontologies (i.e. to define classes and class hierarchy, relationships between classes and properties of these relationships).
- Generate a default form for acquiring instances of, for example, quality cost-related activities.
- Enter domain knowledge. Protégé can be also viewed as a knowledge-acquisition tool that can be used to acquire instances of quality cost-related activities.

Protégé has been designed by Stanford University's Medical Informatics Section to assist "software developers in creating and maintaining explicit domain models, and in incorporating those models directly into program code". According to Eriksson *et al.* (1995), the Protégé methodology, to which the tool belongs, allows system builders to construct software systems from modular components, including reusable frameworks for assembling domain models that can be applied to other organisations.

Before going into the detail of ontology development, the main reasons for developing an ontology are presented below:

- *To share common understanding of the structure of information among people or software agents*. This is one of the more common goals in developing ontologies (see Musen, 1992; Gruber, 1993).
- *To enable reuse of domain knowledge*. This was one of the driving forces behind the recent surge in ontology research.
- *To make domain assumptions explicit*. Making explicit domain assumptions underlying an implementation makes it possible to change these assumptions easily if our knowledge about the domain changes (see McGuinness and Noy, 2001).
- *To separate domain knowledge from the operational knowledge*. This is another common use of ontologies. A task of configuring a product (e.g. a quality cost report) from its components (e.g. transactions in the management information system) according to a required specification can be described and programmed independently of the products and components themselves (see McGuinness and Wright, 1998).
- *To analyse domain knowledge*. Formal analysis of terms is extremely valuable when both attempting to reuse existing ontologies and extending them (see McGuinness and Noy, 2001).

Using KM terminology, our "domain" is quality costing. The main reasons for developing a "quality costing ontology" are sharing a common understanding among people or software agents, separating domain knowledge from other knowledge and analysing domain knowledge.

It is important to understand that often an ontology of the domain is not a goal in itself. Developing an ontology is similar to defining a set of data and their structure for other programs to use. Problem-solving methods, domain-independent applications, and software agents use ontologies and knowledge bases built from ontologies as data. For example, in this paper the development of an ontology for quality costing is described. This ontology can then be used as a basis for the knowledge management solution that would support collecting quality costs.

Quality costing ontology

For the purpose of developing the quality costing ontology (QCO), the PAF model proposed by BS 6143: Part 2 (1990) was used. It is a general and a well-known standard that makes the ontology more flexible and easier to adopt in different organisations and industry sectors. There are many other different PAF models in the literature such as the model proposed by Harrington (1987) where he has compiled a list of typical cost elements, identifying 101 prevention costs, 73 appraisal costs, 139 internal failure costs and 50 external failure costs. Initially, this model looked like a valuable resource that could be used for developing the ontology. However, it is relatively old and needs considerable editing to remove inappropriate detail prior to it being used as a guide in creating the instances in the ontology. BS 6143: Part 2 has the advantage of general acceptance and it embraces Harrington's model.

The ontology was developed in four main steps:

(1) Define the aims and objectives of the ontology.

(2) Develop the class hierarchy.

(3) Defining the slots.

(4) Defining the forms and creating the instances.

The main aim of developing the ontology is to support the collection of quality costs. The ontology will cover the domain of quality costing using the PAF categorisation of quality costs. The QCO will be used in the following ways:

- to help identify quality costs to provide a scoreboard for cost control and identify opportunities for improvement;
- to help collect quality costs, by removing the difficulties associated with collecting quality costs such as: confusion because of the great amount of data involved in calculating quality costs and the lack or loss of information and data needed for calculating quality costs;
- to share common understanding of quality costing among people or software agents;
- to separate quality costing domain knowledge from the operational knowledge;
- to measure quality costs because it is an essential step for achieving competitiveness because these costs are strongly related to the company's annual revenue;
- to track, organise and analyse quality costs; and
- to be used in a knowledge management system that supports quality costing.

From a user's perspective, the QCO should explicitly identify whether an activity is quality cost-related or not. Having identified a quality cost-related activity, the QCO should define whether the activity should be classified as prevention, appraisal or failure. In order to prioritise quality improvement activities, the QCO should be able to identify easily and efficiently those quality costs that are high and those which are relatively low. Furthermore, both the location of where those costs are incurred and the source of the costs is valuable information for those carrying out the quality improvement activities. Consequently, it could be expected that the organisation's quality manager would have the prime responsibility for maintaining and using the ontology but its output would be available to all in the organisation who are involved in making improvements to business process.

The next step in developing the QCO is to define the class hierarchy. In order to do this, it is useful to write down a list of all quality costs-related terms that are useful in statements about quality costs. For the QCO, all the cost element of prevention, appraisal and failure from BS 6143: Part 2 are considered in the list. In addition, some more terms need to be included in the list (name, product, process, customer, value, and location (department), cost source, labour, equipment, material, energy, overhead, and lost opportunity) in order to provide the level of detail required by the quality improvement process to support problem solving.

As shown in Figure 1, prevention, appraisal, failure, internal failure, and external failure are defined as the main classes for this ontology using a top-down development method that starts with the definition of the most general concepts in the domain.

Following this, the cost elements listed in the BS6143: Part 2 PAF model are defined as sub-classes and the second level of the class hierarchy is created as shown in Figures 2 and 3.

The class hierarchy will be the main platform for the ontology where it would be used for creating the instances. Each class will have it own list of instances. After creating the classes, the slots are defined. The definition of the slots consists of two main steps. In the first step the appropriate terms are listed as slots for the classes. After that, a value type is chosen for each slot, which determines the kind of values that the slot may hold. The slots identified for the quality costing classes are shown in Figure 4.

After listing the slots for the QCO, the value type for each slot is defined with value type being dependent upon the purpose of the slot. Table I summarise the available value types.

The nature of each value type is explained below:

- *Name*. The purpose of this slot is to create a space where the user can edit a name for each instances (quality cost). So, the value type for this slot is a string, that is, a slot that takes values that are any string of alphanumeric characters, possibly including spaces.

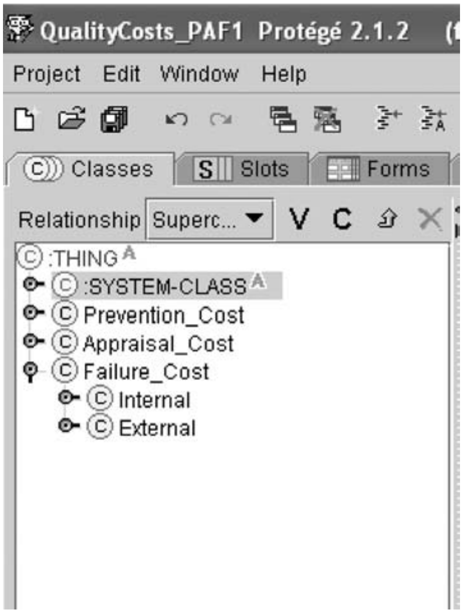

Figure 1.
The class hierarchy for quality costs

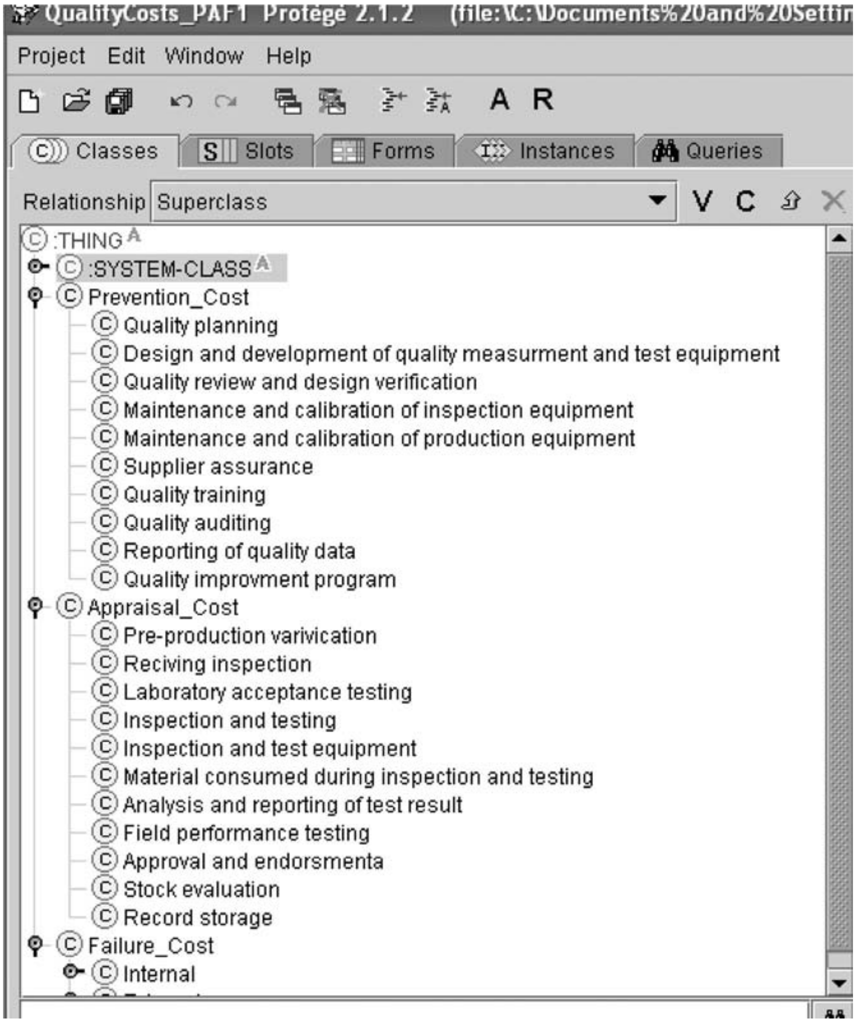

Figure 2.
The class hierarchy for quality costs, showing the prevention and appraisal classes

- *Source*. The purpose of this slot is to create a list of values that a user can chose from in order to define the source of the instance. The value type for this slot is a symbol, that is, a slot that allows creating a pre-set list of strings. The allowed values for this slot are: product, process, customer, and logistics.
- *Cost location*. The purpose of this slot is also to create a list of value for the user to chose from. So, the value type for this slot is a symbol. The allowed values for this slot are: sales and marketing, product engineering, process engineering, purchasing, quality department, human resource, and finance department. As quality costs can appear in more than one single location the cardinality of this slot is going to be multiple to allow the user to define more than one location for the instance (quality cost).

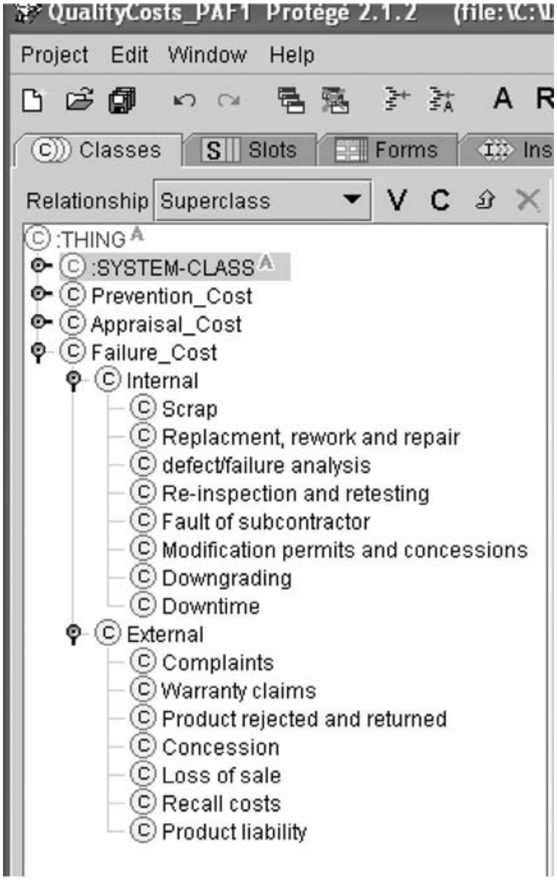

Figure 3.
The class hierarchy for failure costs

Template Slots

Name	Type	Cardinality
Name	String	single
Source	Symbol	single
Cost Location	Symbol	multiple
Equipment cost	Boolean	single
Labour cost	Boolean	single
Energy cost	Boolean	single
Overhead cost	Boolean	single
equipment cost value	Float	single
labour cost value	Float	single
energy cost value	Float	single
overhead cost value	Float	single

Figure 4.
The slots and their type in the quality costs ontology

- *Equipment cost*. The purpose of this slot is to create a flag, that the user can tick when the instance is relevant to it. So, the value type for this slot is a Boolean, where the slots are simple yes-no flags. For example, whether or not a quality cost is an equipment cost can be represented as a value of a Boolean slot: if the value is "true" ("yes") the cost is an equipment cost and if the value is "false" ("no") the cost is not.
- *Labour cost*. The value type for this slot is a Boolean.
- *Energy cost*. The value type for this slot is a Boolean.
- *Overhead cost*. The value type for this slot is a Boolean.
- *Material cost*: The value type for this slot is a Boolean.
- *Equipment cost value*. The purpose of this slot is to allow the user to enter an exact value for the cost. So this slot is float, this type of slot has numbers as values; these numbers may include a decimal point. Values of type float are stored on the system as floating point values, and are only as accurate as the system. When entering a float value for an instance, decimal point or exponential representation can be used. Also positive and negative values can be used.
- *Labour cost value*. The value for this slot is float.
- *Energy cost value*. The value for this slot is float.
- *Overhead cost value*. The value for this slot is float.
- *Material cost*. The value for this slot is a float.

Figure 5 shows how the slots appear after they are defined in the instance tab. Usually after that, the developer must determine which class each slot describes. For this ontology the slots describe all the created classes in the QCO.

The last step in developing the QCO is to create individual instances of classes in the hierarchy. Defining an individual instance of a class requires the selection of a class, the creation of an individual instance of that class and then input of the slot values.

Instances are the actual data in the quality costing knowledge base. In general, an ontology should be structured as well as possible before entering extensive instances. The reason for this is if the developer had to make changes to the class or slot structure after instances have been entered, information may be lost.

The instances tab in Protégé-2000 is used to acquire instances and is typically used by people who are neither domain experts nor knowledge engineers. For the purpose of

Table I.
Possible slot value types in Protégé-2000

Type	Description	Examples
Any	Any of the types below (logical union)	
Boolean	Logical value	True, false
Class	Class in the knowledge base	Organisation
Float	Number with a decimal point	1.0, 3.4e10, − 0.3e-3
Instance	Instance of a class in the knowledge base	Instance_00010
Integer	Whole number	1, 2, − 4
String	String of alphanumeric characters, possibly including spaces	"John Doe"
Symbol	List of values, which may not include spaces	Red, blue and green

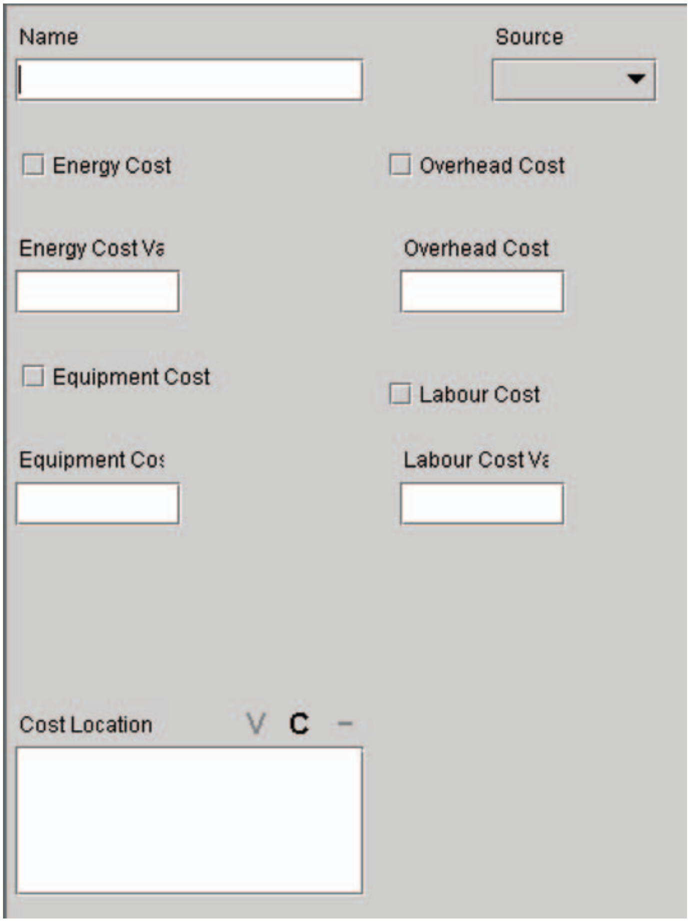

Figure 5.
The quality cost ontology slots in the instances tab

the QCO, the ontology is developed to be used by quality managers to collect and manage quality costs but other users could enter instances to describe quality costs in different departments. They enter the instances by filling out forms and Protégé-2000 generates a default form for every class. Using the forms tab a developer can customise the way the forms look and feel. The forms created for the quality costing classes at the forms tab are then used to enter instances at the instances tab. The forms tab also allows the creation of several different user interfaces for the same ontology. Consequently, the forms for knowledge acquisition can be customised for groups of users or activities. For example, a customer rejection or external failure may require input from a number of users but process scrap or internal failure may be entered automatically using the organisation's management information system.

For the instances in the QCO, a list of PAF quality costs was created by using and modifying Harrington's (1987) PAF quality list and organising it under the BS 6143 PAF model heading. Figure 6 shows an example of an instance (education) under the heading "quality training" in "prevention costs" class.

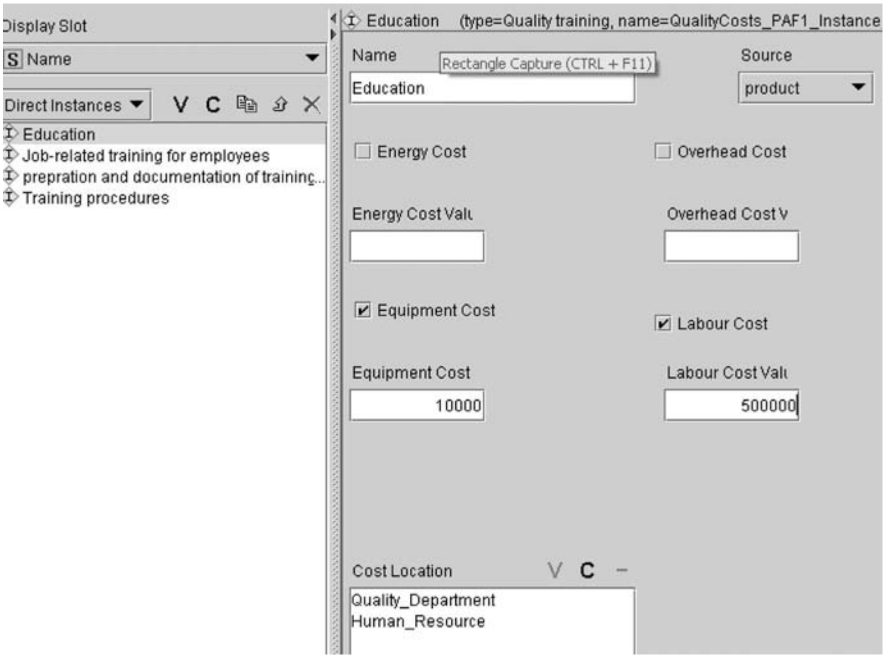

Figure 6.
The instance "education" in the quality costs ontology

An application of the QCO

Protégé-2000 allows the user to create, run, and save query reports. These reports are a way to select instances using the ontology and knowledge base using selection criteria based on the values of one or more slots. A simple query can be created in one of three ways:

(1) A class can be specified by selecting one of the slots attached to the class, and then specifying criteria based on the slot type. For example, the slot may be "source" and the value could be "logistics". Running the query will find the instances that match the criteria.

(2) A query can be created based solely on a slot, without selecting a class. For example, the slot may be "labour cost value". Running the query will find the instances that match the criteria.

(3) A query can be created based solely on a class, without selecting a slot or any criteria. For example, the class may be "prevention." Running the query will find all instances of the selected class and all of its subclasses such as "quality planning."

Consequently, the QCO has the potential to be an important tool for the quality manager to use from the perspective of resource management and measurement, analysis and improvement. In order to demonstrate its usefulness, previous case study research data has been used to create instances from which example query reports have been generated.

The case study data used is that provided by Papacanellou (1997) from research in a company that designed and manufactured electrical components and control systems for the automotive industry. This particular case was selected because it presents the quality cost data at the level of individual cost elements as identified by BS6143: Part 2 rather than aggregated at simply the prevention, appraisal and failure level. The data was generated by the company itself as part of its routine management reporting system that supports its validity. At the highest level of aggregation, the quality costs are presented using the conventional PAF categorisation. Each cost element has been derived by collecting cost data in the departments where those costs have been measured or incurred. For example, scrap costs have been derived from the quality and finance departments. It is only at this level of detail that the quality costing approach becomes specific to this organisation.

In order to demonstrate the application of the QCO, instances have been recreated in order to replicate the actual case study data as shown in Tables II-V. These instances are intended to simulate the raw data that would have been used to prepare the case study quality cost report.

The procedure for creating the instances for the QCO is illustrated in Figure 7 with the five steps being described below:

(1) Enter the name of the quality cost.

(2) Enter the value.

Prevention cost element	£
Quality planning	452,736
Design and development of quality test and control equipment	101,385
Quality review and verification of design	225,136
Calibration and maintenance	81,322
Supplier quality assurance	8,818
Quality improvement programs	718,312
Quality training	30,000
Quality auditing	26,873
Acquisition analysis and reporting of data	79,264
Total	1,993,846

Table II. Case study prevention costs

Appraisal cost element	£
Pre-production verification	393,702
Inspection and testing equipment	8,000
Receiving inspection	98,787
Field performance testing	20,310
Inspection and testing	515,342
Material consumed during inspection and testing	81,676
Approval and endorsements	16,783
Approvals and reporting of test and inspection result	287,919
Stock evaluation	3,900
Total	1,426,419

Table III. Case study appraisal costs

(3) Select the type of cost and value of this type. The QCO allows the user to select the type of cost for each quality cost entered in the ontology, there are five types to select from (energy, overhead, labour, equipment, material).
(4) Enter the location of cost. The QCO is designed to allow the user to select one or more than one location from a list of locations as seen in Figure 8.
(5) Enter the source of cost. The ontology is designed to allow the user to select one or more than one source from a list of sources as seen in Figure 9.

After fully populating the QCO with instances, it now becomes possible to run quality cost query reports. For example, if the user wanted to identify those quality cost elements that were greater than, say, £100k then the "total value" slot can be used to design the query. The input form is shown in Figure 10.

The results of running such a query are shown in Figure 11 and can be easily verified against the original data.

It may be important to the quality manager to identify the level of quality costs incurred in the quality department. In this case, a query can be designed using the cost location slot that would generate the output shown in Figure 12.

It is at this level of detail that the QCO begins to demonstrate its superiority over conventional methods of quality costing. One further example is included which features the "source" slot. With the QCO, it is possible to identify those quality costs that have the supplier as their source and list them as shown in Figure 13.

It would be extremely difficult to generate reports at this level of detail with the conventional methods of quality costing used in the preparation of the case study quality cost report.

Table IV.
Case study internal failure costs

Internal failure cost element	£
Scrap	115,731
Rework/replacement/repair	483,498
Troubleshooting/defect/failure analysis	292,295
Re-inspection and retesting	600,013
Fault of supplier	45,559
Modification/permits and concessions	66,975
Downtime	496,846
Total	3,100,917

Table V.
Case study external failure cost

External failure cost element	£
Complaints	171,294
Warranty claims	157,522
Products returned and rejected	56,136
Loss of sales	5,000,000
Recall costs	75,527
Concessions	0
Product liability	26,000
Total	5,486,479

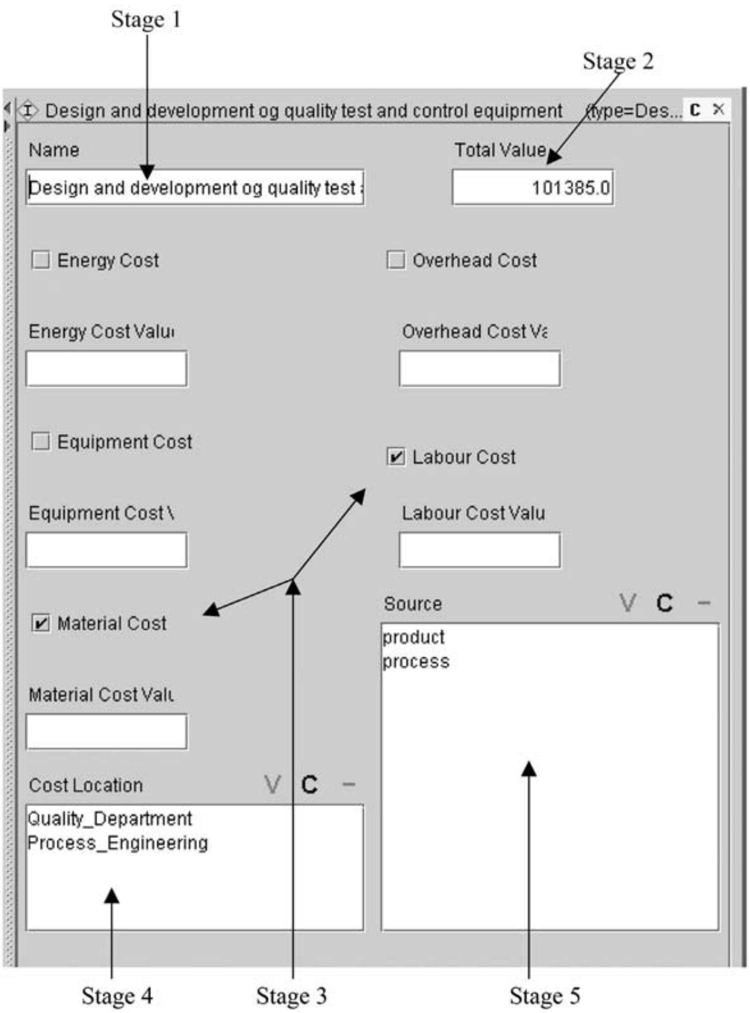

Figure 7.
The stages of entering quality cost data in the ontology (instance tab)

Discussion of findings

The application of the QCO to the case study demonstrates that the concept of an ontology for quality costing is feasible. The structure of the QCO was developed independently of this application but the QCO was able to replicate the reporting of quality costs and quality cost elements provided in the case study company. The QCO also provided additional reporting and analysis capabilities, such as an analysis of quality costs by source, that are both easy to define and efficient to use. The QCO offers an organisation the facility to run reports very quickly and to obtain a better view of its quality costs than was possible previously. This reporting facility is being further improved by the development of a simple interface to allow the ontology software to take data directly from databases commonly used by organisations.

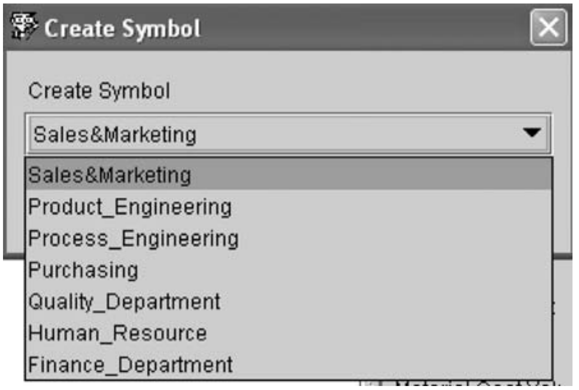

Figure 8.
Locations list

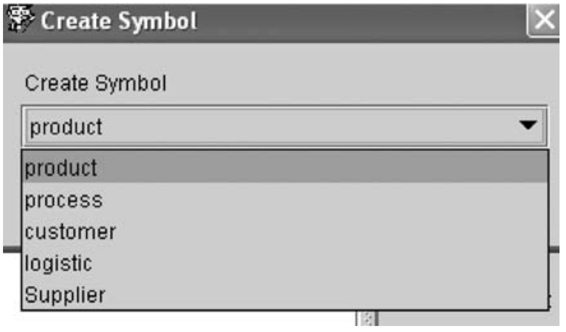

Figure 9.
Sources list

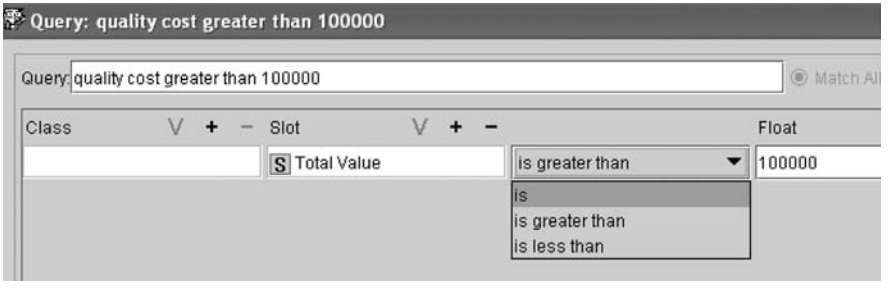

Figure 10.
Query design form

The strength of the QCO is its requirement for quality cost-related activities to be quantitatively and qualitatively described using a standard form of language and terminology. Having established this structure or framework, quality cost incidents that would conventionally be recorded using a variety of data collection methods can now be recorded using the QCO's instance forms. Consequently, the reporting and analysis of quality cost data becomes much easier and more efficient. It is accepted that the manual creation of instances will still be required for some activities (e.g. quality planning meetings) but other routine activities could be expected to create instances

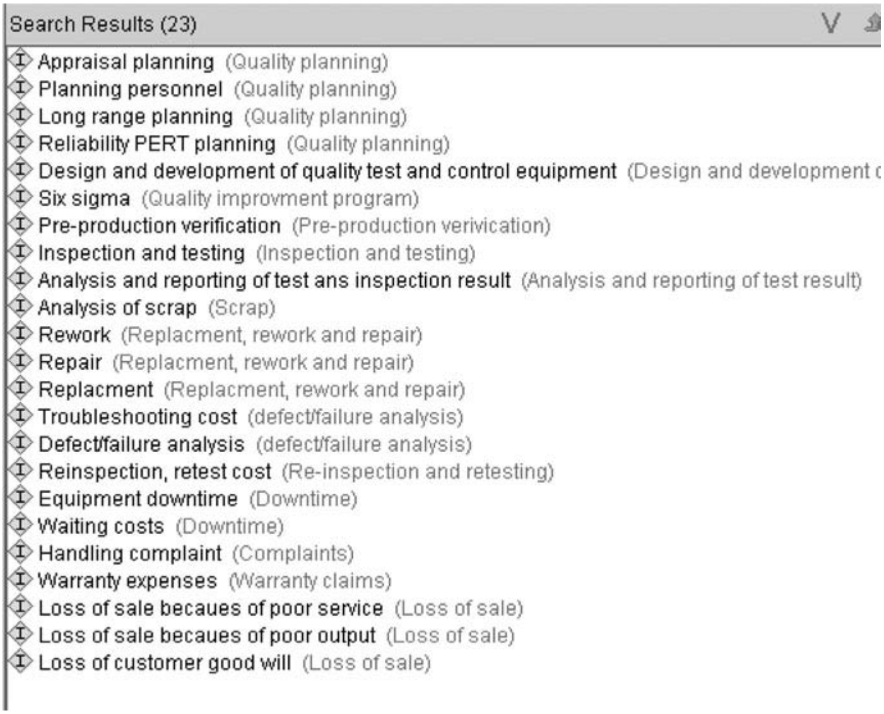

Figure 11.
Quality cost elements greater than £100k

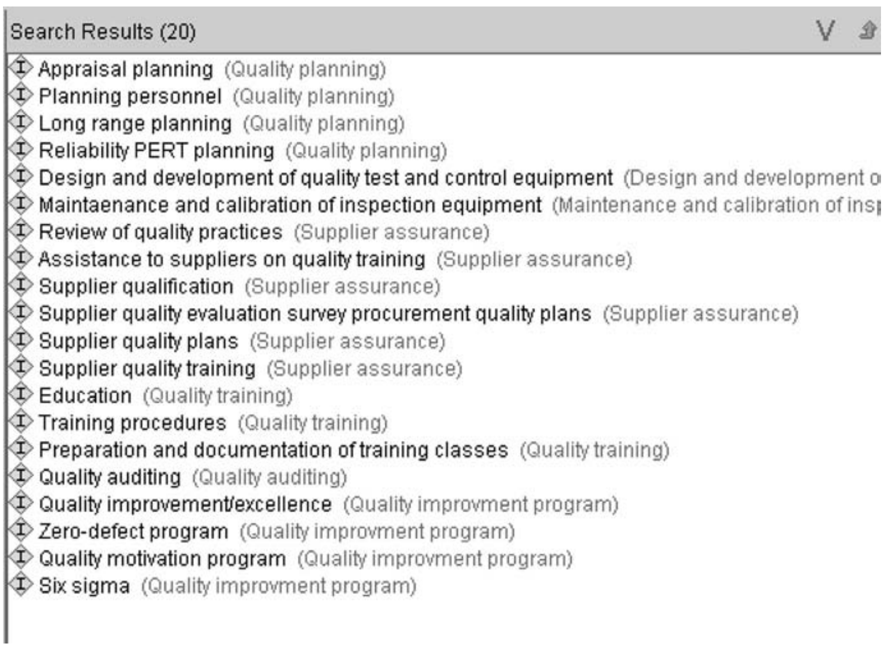

Figure 12.
Quality cost elements in the quality department

Figure 13.
Quality cost elements with supplier as the source

automatically via the organisation's existing management information system (e.g. for internal production scrap). Running the QCO in parallel with an organisation's existing quality costing system would be an interesting research case study to demonstrate how using the QCO can reduce the costs associated with running a quality costing system.

The issues of awareness and understanding of the concepts of quality costing are addressed right at the outset of developing the QCO and, as such, the QCO captures and preserves this high level of awareness and understanding for all to use again and again. Consequently, the implementation of the QCO in an organisation would require, initially, the support of a cross-functional team led by the quality manager who would, subsequently, control and maintain the QCO as part of the organisation's documented quality management system.

The QCO can be seen as a relatively "hard", prescriptive systems solution for the difficulties associated with quality costing. Its use of the PAF model could be seen as an advantage because of its widespread use by practitioners. However, other models do exist, such as the price of non-conformance/ price of conformance categorisation proposed by Crosby (1984), and for the QCO to achieve wider application it would need to encompass this process-based classification as well. At this stage, there is confidence that this can be achieved within the QCO because, at the quality cost instance level, both classifications are sharing the same data pool.

Conclusion

The QCO was developed using the PAF categorisation and was applied to the case study data of Papacanellou (1997). This experience suggests that the QCO offers an alternative approach to the collection and reporting of quality costs using the conventional PAF categorisation. One of the main reasons for developing any ontology is the ability to reuse the knowledge in other applications. From the perspective of the QCO, it now needs to be evaluated using alternative classifications of quality costs such and with other organisations including those in the service sector. It is believed that the QCO is generic for implementation in any manufacturing organisation but, having already established a quality costs class hierarchy, it is relatively easy to customise the data collection and reporting formats to an individual organisation's requirements. However, implementation in a service organisation would present significant research questions. The principles used for ontology development could be used to establish the extent of the commonality of terminology and data structures for quality costing between a manufacturing organisation and, for example, a bank or a hospital. When these issues are investigated then it may become possible to claim, from a KM perspective, that the QCO can provide "reusable" domain knowledge.

Another reason for the development of the QCO was that it would provide a more efficient system for the collection and reporting of quality costs when compared with the conventional approaches such as the department-based reporting used in the case study. The strength of the QCO is its use of "instances" to capture quality cost-related events. The instance forms have a very flexible design such that they could be used to replace existing quality reporting documentation. Consequently, quality cost collection becomes automatic within the operation of a quality management system. In other words, the QCO could be implemented "instead of" rather than "as well as" other quality reporting systems with the implication that the cost to the organisation of running a quality costing system becomes much reduced. However, to realise this potential, further research is needed to integrate the QCO's data capture and reporting capabilities with other management information software tools.

The QCO was developed as a "hard" or systems-based solution to what is often viewed as an education or training issue (i.e. lack of understanding or awareness of quality costing). Implementing the QCO imposes a standardised terminology and does not allow confusion to exist at the data collection or data reporting interfaces. The QCO is a method that can be used to embed quality costing knowledge within an organisation. However, of course, training solutions and the QCO are not mutually exclusive. Indeed, it is very possible that the QCO could be used to underpin training programmes in quality improvement techniques, including quality costing, because of its ability to demonstrate the financial consequences of a quality-related action using real company data. The use of the QCO as a training aid with "what if?" capability could be very valuable to an organisation that is just starting to implement a quality improvement programme as well as for those organisations that are experienced quality costing practitioners. Again, further research into these opportunities is needed.

The work carried at this stage and the opportunities for further research described in this paper do suggest that the application of KM approaches in the field of quality management could be a very productive area for both researchers and practitioners. Implementing and sustaining quality improvement programmes over the long term is a challenge for all organisations not least because of the impact of issues such as understanding and awareness, culture and information systems. The QCO is just one approach that could be integrated into an organisation's KM system for quality management. Other facets that remain to be investigated include KM approaches to the capture and sharing of knowledge in the areas of customer satisfaction, process improvement and supplier performance.

References

Bhatt, G.F. (2001), "Knowledge management in organisations: examining the interaction between technologies, techniques, and people", *Journal of Knowledge Management*, Vol. 5 No. 1, pp. 68-75.

Carson, J.K. (1986), "Quality costing – a practical approach", *International Journal of Quality & Reliability Management*, Vol. 3 No. 1, pp. 54-67.

Civi, E. (2000), "Knowledge management as a competitive asset: a review", *Marketing Intelligence & Planning*, Vol. 18 No. 4, pp. 166-74.

Crosby, P.B. (1984), *Quality Without Tears*, McGraw-Hill, New York, NY.

Dale, B.G. and Plunkett, J. (1999), *Quality Costing*, 3rd ed., Gower, Aldershot.

Davenport, T.H., De Long, W. and Beers, M.C. (1998), "Successful knowledge management projects", *Sloan Management Review*, Vol. 39 No. 2, pp. 43-57.

Duffy, J. (2000), "Knowledge management: to be or not to be?", *Information Management Journal*, Vol. 122 No. 15, pp. 33-5.

Eriksson, H., Shahar, Y., Tu, S.W., Puerta, A.R. and Musen, M.A. (1995), "Task modelling with reusable problem-solving methods", *Artificial Intelligence*, Vol. 79 No. 1, pp. 293-326.

Grosso, W.E., Eriksson, H., Fergerson, R.W., Gennari, J.H., Tu, S.W. and Musen, M.A. (1999), "Knowledge modelling at the millennium (The design and evolution of Protégé-2000)", *Proceedings of the Twelfth Banff Workshop on Knowledge Acquisition, Modeling, and Management, Banff, Alberta*.

Gruber, T.R. (1993), "A translation approach to portable ontology specification", *Knowledge Acquisition*, Vol. 5 No. 1, pp. 199-220.

Gupta, A. (2000), "Enterprise resource planning: the emerging organisational value systems", *Industrial Management and Data Systems*, Vol. 100 No. 3, pp. 114-8.

Harrington, H.J. (1987), *Poor-Quality Cost*, ASQC Quality Press, Milwaukee, WI.

Israeli, A. and Fisher, B. (1995), "Cutting quality cost", *Quality Progress*, Vol. 24 No. 1, pp. 46-9.

Johnson, R.D. and Kleiner, B.H. (1993), "Does higher quality means higher cost?", *International Journal of Quality & Reliability Management*, Vol. 10 No. 1, pp. 86-100.

Kumar, K., Shah, R. and Fitzroy, P.T. (1998), "A review of quality cost surveys", *Total Quality Management*, Vol. 9 No. 6, pp. 479-86.

Lascelles, D.M. and Dale, B.G. (1990), "Examining the barriers to supplier development", *The International Journal of Quality & Reliability Management*, Vol. 7 No. 2, p. 46.

Lee, S.M. and Hong, S. (2002), "An enterprise-wide knowledge management system infrastructure", *Industrial Management & Data Systems*, Vol. 102 No. 1, pp. 17-25.

Liebowitz, J. (2001), "Lessons learned in developing knowledge management strategies for government", *KM World*, January.

McGuinness, D.L. and Noy, N.F. (2001), *Ontology Development 101: A Guide to Creating Your First Ontology*, Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880, Stanford, CA, March, available at: www-protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html

McGuinness, D.L. and Wright, J. (1998), "Conceptual modelling for configuration: a description logic-based approach", *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, Vol. 12 No. 4, pp. 333-44.

Machowski, F. and Dale, B.G. (1998), "Quality costing: an examination of knowledge, attitudes and perceptions", *Quality Management Journal*, Vol. 5 No. 3, pp. 84-95.

Musen, M.A. (1992), "Dimensions of knowledge sharing and reuse", *Computers and Biomedical Research*, Vol. 25 No. 1, pp. 435-67.

Neches, R., Fikes, R., Fiinin, T., Gruber, T., Senator, T. and Swartout, W. (1991), "Enabling technology for knowledge sharing", *AI Magazine*, Vol. 12 No. 1, pp. 36-56.

Nonaka, I. (1994), "A dynamic theory of organisational knowledge creation", *Organisational Science*, Vol. 5 No. 1, pp. 14-37.

Papacanellou, D. (1997), "Quality costing simulation: quality costs a strategic perspective", MSc dissertation, UMIST, Manchester.

Prusak, L. (2001), "Where did knowledge management come from?", *IBM System Journal*, Vol. 40 No. 4, pp. 1002-7.

Roden, S. and Dale, B. (2000), "Understanding the language of quality costing", *The TQM Magazine*, Vol. 12 No. 3, pp. 179-85.

Superville, C.R. and Gupta, S. (2001), "Issues in modelling, monitoring and managing quality costs", *Total Quality Management*, Vol. 13 No. 6, pp. 419-24.

About the authors

Steve Eldridge, the corresponding author, is based at the Manchester Engineering Doctorate Centre, The University of Manchester, Sackville Street Building, Sackville Street, Manchester M60 1QD, United Kingdom. Tel: +44 1613064161; E-mail: steve.eldridge@manchester.ac.uk

Mohammed Balubaid is based at the Manchester Engineering Doctorate Centre, The University of Manchester, Sackville Street Building, Sackville Street, Manchester M60 1QD, United Kingdom.

Kevin D. Barber is based at the School of Management, The University of Bradford, Emm Lane, Bradford BD9 4JL, United Kingdom.