# A Computer-Based Design Reuse System

Tamer M. M. Shahin[1] and S. Sivaloganathan[2]

[1]United Arab Emirates University, Tamer.Shahin@uaeu.ac.ae
[2]Brunel University, s.sivaloganathan@brunel.ac.uk

**ABSTRACT**

Structuring past designs and storing them in computer archives for future use with ease are two essential components of a Design Reuse System. This paper presents a methodology that structures and stores mechanical designs at their conceptual, embodiment and detailed stages for reuse. Its computer implementation is then described to illustrate the power of the system. Each Stage of the Design Process can be searched for and retrieved. The designs can then be identified to be strong or weak. This will subsequently enable a designer whether or not to reuse a design within a new product.

**Keywords:** Computer-Based Design, Design Reuse, Design Methodology, CAD.

## 1. INTRODUCTION

Reuse of successful past designs and designers' experiences has been employed for many years now. Research shows that designers prefer to use the concepts and lessons of past designs, [2] especially when the task undertaken is complex. In the past, when engineers were trained as designers, they were encouraged to maintain a logbook to record successful designs of subsystems that could be reused. This cuts down design time and hence cost and also increases the reliability of new products. But due to the inherent non-structured nature of the reuse, gaps often arise whereby reuse could have been better exploited. Substantial research has now been carried to enable the efficient utilisation of Design Reuse, but an overview [11] has revealed that these 'modules' of research need a design reuse system that integrates within a systematic design process. This will provide the means of retrieving design information either wholly or partially, resulting in an efficient system. This paper presents such a system of which its aims can be summarised as follows:

a) Develop a methodology to structure and store each stage of mechanical designs in a holistic manner.
b) Implement this methodology as a computer-based system that allows for the storage and subsequent retrieval of such designs.

In order to achieve these aims, the following objectives were set out and pursued:

1. Establish the design model that is to be the underlying system for Design Reuse.
2. Establish a method to store and represent each stage of the design process from Concept through to detail within the underlying Design model
3. Develop a technique to identify good or bad designs for future reference
4. Establish suitable methods of transferring this developed methodology and representing it within a computer-based system
5. Develop the computer-based system
6. The remainder of this paper explains how the two stated aims have been achieved through individual sections describing the above-mentioned objectives.

## 2. STRUCTURING OF THE DESIGNS – THE DESIGN MODEL.

For a design reuse system to be effective, it should provide assistance to the designers in the form of information appropriate to the design stage he/she is in. This indicates a strong relationship between the design process and the format of the past design information. Design models formalise the problem solving activity in the transformation of an abstract set of requirements into definitions of physically realisable systems. Hence for effective design reuse, the design model the designer uses and the design model supporting the structuring of the past designs should be comparable.

The choice of design model therefore dictates, to some extent, the type of knowledge needed just as the choice of a representation determines how it can be used [1,4].
Systematic design advocates the use of 'Design Methods' in association with a design model. However, designers will quickly point out that different products require different approaches to design. To introduce such flexibility, Product Modelling has recently been introduced. This is essentially the reservoir of information concerning the lifecycle of a product. It has two components namely (a) the data concerned and (b) the process chains. Analysing design models, design methods and product modelling together reveals that in a proper design system:

- The design models represent the minimum prescribed path
- The system contains a variety of design methods
- The activities described by the process chain consists of (a) the minimum activities defined by the design model and (b) the design methods used
- The process chain can be unique to each design while following the design model in general.

Design Function Deployment [3,10] is such a design system that addresses these issues. Figure 1 illustrates the structure of the system. In Level 0, the user defines the various design methods (design tools) he/she is expecting to use during the design process. Level 1 represents the design model as the minimum prescribed path. Level 2 houses the various Design Methods and Level 3 houses the supporting databases, rule bases and knowledge bases. Design Reuse can be a design method housed in level 2 and can be used as a tool at different stages of the design process.

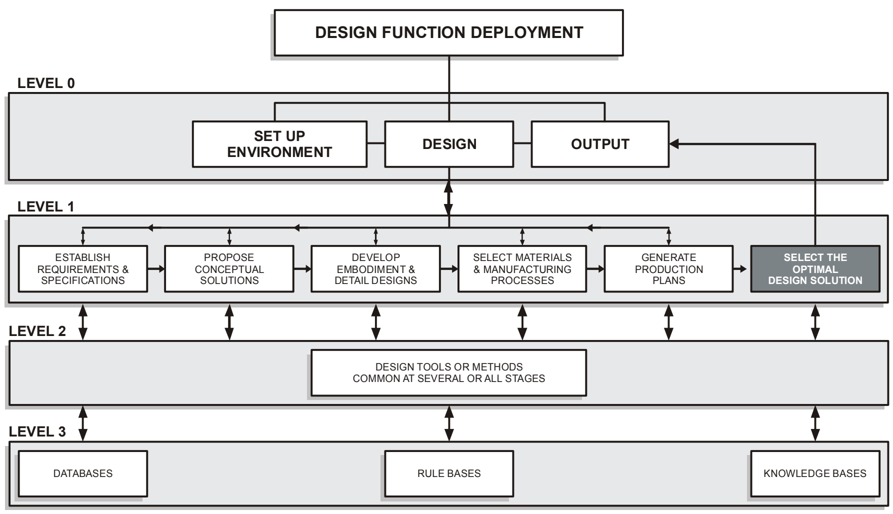


Fig. 1. Design Function Deployment.

Design Reuse would be housed as a Design method within Level 2. It can hence be accessed at any stage of the Design process and at anytime during a particular process chain. In this way, Design Reuse can be used as a supporting tool without constraining the designer by imposing the past designs while permitting the beneficial aspects for exploitation.
The design model underlying Design Function Deployment (DFD) has six stages (Level 1 in figure 1). It starts with the identification of customer needs and the derivation of the specifications from these needs. It then proceeds with the establishment of the conceptual solutions in stage 2, embodiments and detailed designs in stage 3, materials and associated manufacturing processes in stage 4, and process plans in stage 5. Each of these stages has a feedback loop to any of the prior stages. Finally, stage 6 is the selection of an appropriate solution. Design Reuse was found to be relevant in the first three stages (From Concept through to Detailed Design) and will be the only consideration from the Design Model in this paper.

## 3. REPRESENTING EACH STAGE OF THE DESIGN PROCESS

It is vital that design information can be retrieved and reused either wholly or partially. For this reason, among others, it is necessary to realise an efficient method of representing each stage of the design process. The first three stages of the Design model discussed in the previous section can be categorised to expect the following outcomes:

- **Customer Requirements**: This is essentially a wish list that a customer would like to see in a given product. These requirements would be represented as a prioritised list. e.g. A chair that is comfortable to sit on.
- **Product Concept**: This defines what the product will do in technical terms. The design is stored in a manner that avoids the use of possible design solutions or outcomes (solution neutral). This is done by having a prioritised list of Specifications or Functions. A review carried out on Function Analysis [4] suggested that the solution-neutral form of functions should rely on a formal prescribed language. They are usually sub-divided in a tree structure containing Primary, Secondary and Tertiary Specifications. This is simply a refinement and grouping of specifications beginning with a vague definition at the primary level until achieving accurate specifications at Tertiary Level. These levels can be optional. A typical specification is represented as a verb/noun phrase function e.g. Provide seat area.
- **Solution Concept**: This is essentially the conceptual solution or outcome that the designer has decided as optimal in addressing the Product Concept. This is represented in the form of Subsystems, whereby each subsystem can represent a particular technology in addressing a set of specifications.
- **Embodiments**: It is widely known that a given concept can be implemented using different parts. Embodiments are essentially parts trees that form the parts and subassemblies of the product.
- **Detailed Design**: This is where the storage of the part details (form or geometric details) is kept. This is carried out using Geometric Models

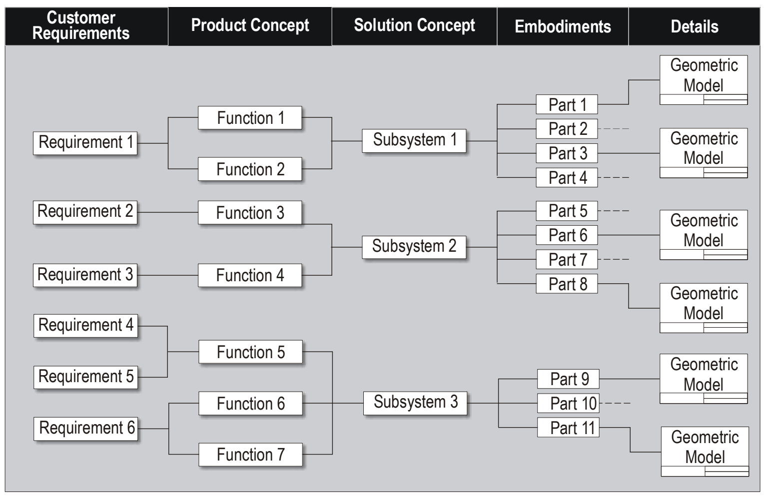


Fig. 2. Representation of the Design Outcomes.

Figure 2 presents a generic structure of the Design Outcomes of a product. Having established these outcomes, it is important to choose suitable Data Models for efficient representation. Figure 3 presents each of these Data Models. Chart-Based Representations for the Conceptual and Embodiment Designs were identified in order to incorporate ratings. (Discussed in more detail in the following section).

Feature-Based Design [6] was deemed the only contemporary geometric modelling technique that contained sufficient flexibility to cater for changes. However on its own, it is not yet suitable for radical topological changes which might be needed with Design Reuse. Hence the 'Generative methodology' introduced by Shahin [7] has been utilised.

This method is based around the Constructive Solid Geometry (CSG) representation technique, which stores and builds a complex solid model by performing Boolean operations on basic solid primitives. The Generative Methodology takes this process one step further by using a technique called 'Parametric CSG' [7]. A solid model is created. Using Interactive Feature Recognition, Features are then extracted from the model and stored as CSG trees. This CSG tree is then modified by the introduction of parameters and geometric constraints to it. On modification of the design, the designer enters in values for the parameters using a user-friendly front-end menu. These values are then substituted into the parametric CSG tree and the solid model is re-generated. The advantage in this method is that it can be extremely flexible since rules can be stored within these parametric CSG trees. This not only leads to simple modifications in the parameters, but also entire changes of topology, by including different optional paths in the parametric CSG tree are also possible.

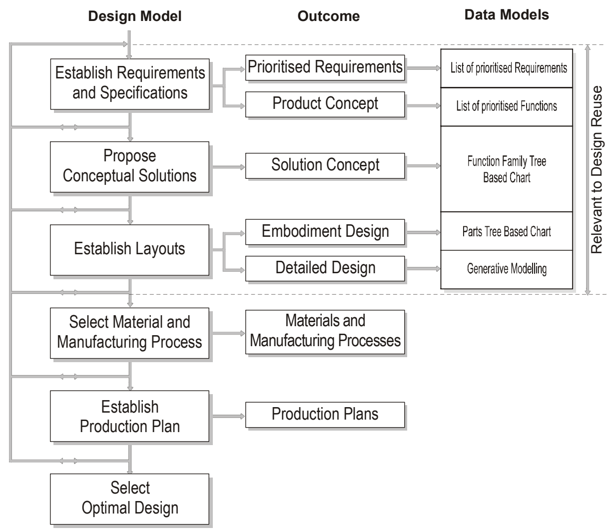


Fig. 3. The Structure of the DFD Design Model, Outcomes and Data Models.

## 4. IDENTIFYING GOOD/BAD DESIGNS

It is evident that what is needed is not only a system that can store successful past designs, but it is just as important to store the ones that were not so successful so as to avoid them in the future. Therefore a method is needed to help the designer in deciding whether or not to reuse a past design. Hence a ratings technique has been employed, which is a feature of DFD. This enables a designer to record how well or badly a particular solution has performed to a given problem. This can be at any stage of the design process. These ratings can then be used at the reuse stage in deciding whether or not to incorporate a past design feature within a new product. For Design Reuse, the ratings are stored within two charts – The Conceptual and Embodiment charts. They use the same rating method employed within QFD [8] as this is an accepted and established technique.

### 4.1 Conceptual Chart

Sivaloganathan et al [10] state that an adequate representation of Conceptual Design provides:

a) An overall list of functions (specifications) performed by the product
b) A list of subsystems that constitute the overall product
c) A list of functions performed by each subsystem
d) A description of the shape of the product if appropriate
e) An optional importance ratings of the functions, and
f) An optional 'measure of achievement' to indicate whether the function is provided well by the concept or not. A chart-based method [10] was developed as part of the program to store conceptual designs. A typical chart is given in Figure 2.

The Product Concept and Solution Concept from the Design Model form the Conceptual Design Stage. They are separated in order to allow for keeping say, the Product Concept the same, whilst changing the Solution. Figure 4 shows how this is arranged to form the Conceptual Design Chart. The Tertiary Design Specifications are allocated Importance Ratings (how important is each specification to the project). Ratings are then allocated to how different Design Subsystems (Solutions) meet relevant Design Specifications (only the relevant ones need to be addressed). The Absolute Importance Rating for each Design Subsystem is then calculated by the sum of the product of the matrix rating and its corresponding Specification Importance Rating in each column. A Relative Importance Rating is then calculated by normalising and rounding the Absolute Importance Ratings on a scale of 1 to 9. A Conceptual Sketch is also included to help visualising the product. This establishes the conceptual design and rates how well or badly each Subsystem addresses the Product Specifications.

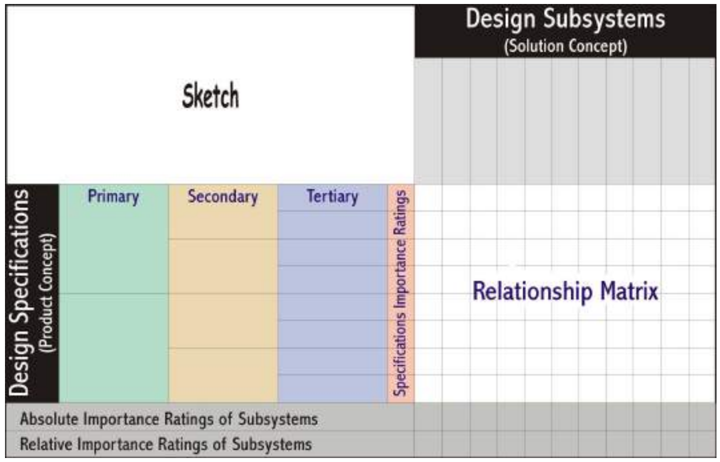


Fig. 4. The Conceptual Chart.

### 4.2 Embodiment Design

The Embodiment Design Stage carries through the successful Design Subsystems and presents subassemblies and parts to each Subsystem. Again, representing this in the form of a chart helps efficient storage and the rating of how well or badly each part addresses the relevant Design Subsystem. Figure 5 illustrates how this is done. The Design Subsystems carry forward their Relative Importance Ratings calculated in the Conceptual Design Chart. Ratings are then allocated to how different Parts meet relevant Design Subsystems (only the relevant ones need to be addressed). The Absolute Importance Rating for each Part is then calculated by the sum of the product of the matrix rating and its corresponding Subsystem Relative Importance Rating in each column. A Relative Importance Rating is then calculated by normalising and rounding the Absolute Importance Ratings on a scale of 1 to 9. An Embodiment Sketch is also included to help visualising the product. This establishes the Embodiment design and rates how well or badly each part addresses the Product's Design Subsystems.

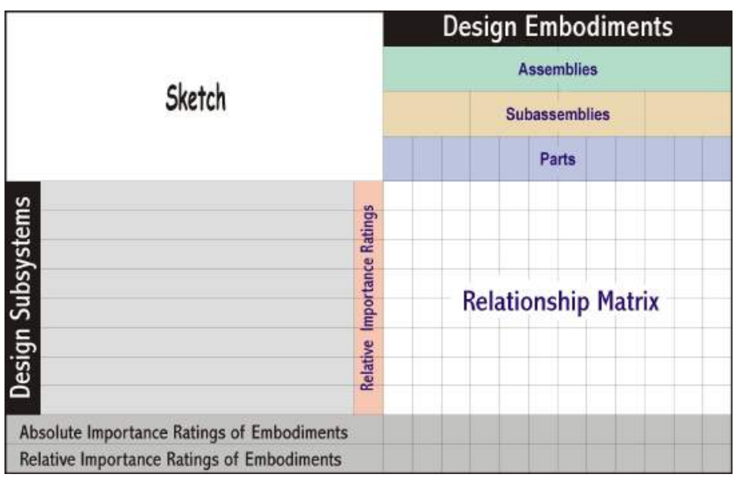


Fig. 5. The Embodiment Chart.

## 5. DESIGN REUSE

Once a method for structuring designs has been established, exploitation of this material is needed to enable its reuse. For Design Reuse to be successful, three main goals become dominant for a successful implementation:

1. Structuring the Information within the proposed Design Model
2. Partitioning Past Designs for storage

3. Developing a software database package that allows for the easy storage and retrieval of past designs.

In order to do this, establishing how the Design Reuse Structure and respective Data Models are to be reused will help in perceiving efficient methods of storage for easy retrieval.

### 5.1 Reuse at the Conceptual Stage

It has been established earlier in this paper that the Product Concept is represented as Functions and the Solution Concept is represented in terms of Subsystems. Both these are combined in one Conceptual Design Chart which also includes measurements on how well the subsystems achieve the desired functions of a product.

Both Functions and Subsystems can be reused together or independently when designing a new product at the conceptual stage. Figure 6 illustrates this using a generic diagram. This figure shows three potentially different products with reuse employed.

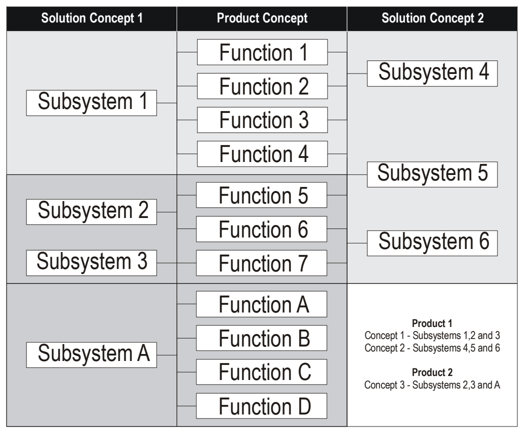


Fig. 6. Reuse of Conceptual Designs.

Product 1 (e.g. a plane) has a set of design specifications outlined by Functions 1 – 7 (The Product Concept). Functions 1 – 4 are addressed by a conceptual solution in the form of Subsystem 1. Subsystem 2 addresses functions 5 and 6 and subsystem 3 addresses function7. Subsystems 1–3 form Solution Concept 1, which is a particular conceptual design or Product in its own right (e.g. a jet plane).

Subsystems 4-6 present a different method of satisfying the same set of specifications, by possibly using different technologies (e.g. a propeller plane). By reusing the same Product Concept, it has been possible to provide for two different concepts.

Taking this one step further, it is possible to reuse some of the same Functions, say 5-7, whilst adding a new set of design specifications, as outlined by Functions A-D. This creates Product Concept 2. As Functions 5-7 were addressed well in the past by Subsystems 2 and 3, these solution concepts will be reused as well. However, the new set of design specifications requires new conceptual solutions and is addressed by Subsystem A. Hence, Subsystems 2,3 and A create a conceptual solution for an entirely new product (e.g. a Helicopter).

Therefore the facility provided by the Reuse system is to show the function structures of one solution concept so that another concept or product can be developed without losing any freedom in creativity

### 5.2 Reuse at the Embodiment Stage

A similar scenario as described for the conceptual stage can be adopted at the Embodiment stage. However this time the set of 'Specifications' will be the Subsystems and the 'Solutions' will be the parts tree. Hence it becomes possible to reuse different parts to address the same subsystem as well as reusing the same parts in different subsystems. The ratings within the Embodiment chart mentioned in section 4.2 again play a vital role in the identification of good or bad embodiments.

### 5.3 Reuse at the Detailed Stage

At this stage, the part would have already been defined during Embodiment design and so all that needs to change would be the details such as geometric parameters as well as non-geometric parameters such as material property. It is also important to be aware of the inherent geometric constraints that can arise with such modification. The Generative methodology adopted here caters for all these issues and hence all that is needed is an interface and link to this implementation.

### 5.4 Partitioning Past Designs for Reuse

In the majority of situations, past designs can only be studied by looking at the product and its detailed design in previous drawings. The objective of analysis is to facilitate the extraction of good functional subsystems for reuse in new designs. A four-stepped process (named Design Interpretation by the authors) was established for this purpose:

1. Initially the designer describes the process,
2. From this process, parts are grouped together and their subsystems are established.
3. From these subsystems and the process, the functions of the subsystems are defined
4. This will lead to identifying the specifications of the product.

## 6. THE COMPUTER-BASED DESIGN REUSE SOFTWARE

Several important issues needed to be considered when transforming a proposed methodology or system to a computer program. Importantly, it should not be a direct transfer as this may, and usually will be confusing to the user/designer. A spreadsheet, for example, is not how a person would usually perform manual calculations. However, it should not also be completely different so as to alienate the designer and make it a burden to use. The application should therefore try to implement the proposed methodology, whilst attempting to remain loyal to how a designer would 'design' a product. The previous section discussed the structuring and partitioning of design information for subsequent reuse. This section describes its computer-based implementation. Such a program achieves the following:

1. Allows input of designs in a manner consistent with the thought process of a designer.
2. Stores new designs under the proposed structure within the Design Model.
3. Allows the partitioning of existing past designs and subsequent storage as mentioned above
4. Allows for easy retrieval of past designs at any stage of the design process
5. Provides an indication as to the success of each stage of the design to help in the decision of reusability.
6. Provides a means for easy reusability and modification of past designs to suit a new one.

The remainder of this section initially describes the development of the program as a whole and then explains how each of the above points has been addressed within the software.

### 6.1 Developing the Design Reuse Program

It was evident that what was required was a program that was based around a database. Such a database required careful consideration and structuring to achieve efficient storage and retrieval. It was decided to base the database around a relational table structure so as to integrate easily with other databases in the designer's environment. Each piece of information is stored in a separate table and then relationships and links are created to avoid repeated information and inefficiency. This also allows for greater flexibility. Although the program used MS Access to generate the tables and store the information, all the links and queries were stored within the Design Reuse Program using the Structure Query Language (SQL) to aid easy portability to other databases. This means that the all the data structures can hence be integrated with a company's existing database and linked further down to information such as suppliers and customers.

The search facility is available at any stage of the design process and is searchable by any data model output, such as the Specifications, Functions or Embodiments. Search results will display where the search criteria have been used before as well as relevant ratings. It is possible to view further down or up the design process to see how this item is related throughout the whole process. It is then possible to import any piece of the information or all of it into the current working project. For ease of retrieval, information is also exported to HTML files for viewing over a company's Intranet. The Database Management System accomplishes the following:

- Structures and Stores new and old designs using the underlying Design model described earlier, i.e. Product Concept, Solution Concept, Embodiments and Detailed Designs.
- Retrieving, viewing and modifying any stage of the design in an efficient manner
- Reusing any item of a design into another design.

### 6.2 Inputting of information

In order to capture information whilst a designer is in the process of designing a new product, this process should be seamless to the designer. Sivaloganathan et al. [9] show through a practical case-study that when designing a new product and that not all the information is immediately available to the designer, he/she will revert to traditional-based

design. This includes brainstorming, sketching and using general experience. However, once this information becomes readily available, the designer is keen to store this information in a structured fashion, so that he/she has a proper record of the design for reporting and reuse. This has been provided for the designer in the following ways:
A Notebook environment has been developed for capturing the designer's thought process. This becomes a reference point for the designer and is similar to a traditional designer's 'logbook'. The information is stored in a database in the relevant places. Figure 7 illustrates an example of this environment. Each tab for the notebook represents a different stage in the design process. There are several pages under each stage to enable the input of sketches, comments and ideas. Also available to the designer is a database search facility to view previous past designs for possible reuse.

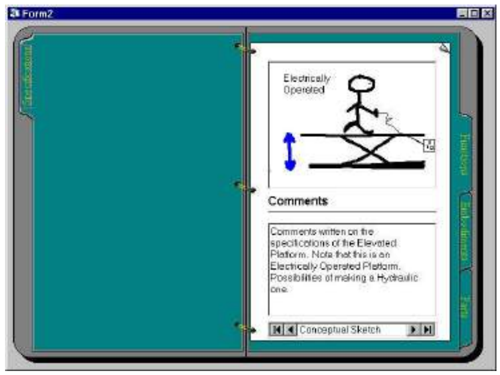


Fig. 7. Capturing Early Design Information.

### 6.3 Structuring the information

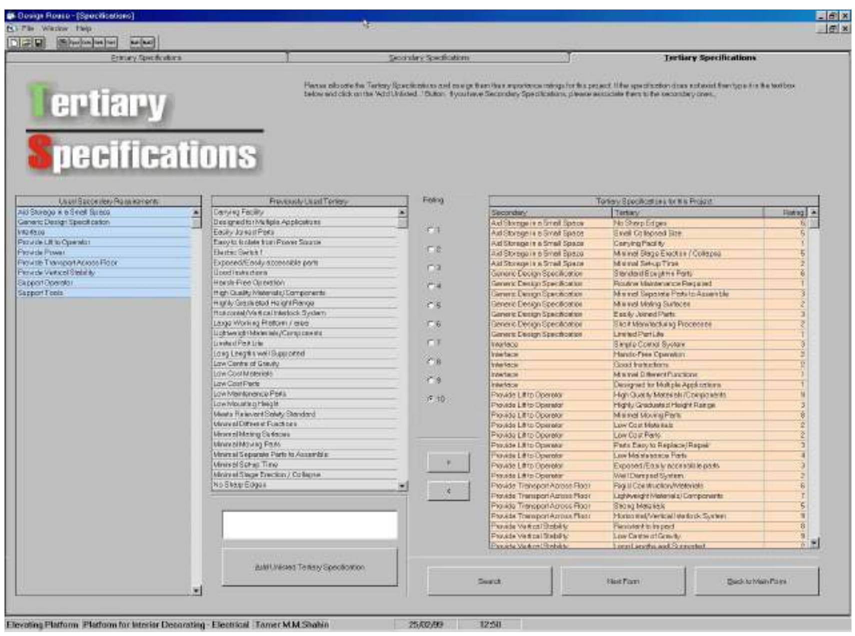


Fig. 8. Inputting Requirements and Stage 1 Chart .

Once the designer is ready to start structuring information for the design, it is possible to access at any time, screens to feed in the specifications of the product. The designer will be presented an opportunity to refine the specifications to precise engineering specifications. An importance rating is then asked from the designer by selecting one of the listed options. It was intentionally decided not to present the designer with any charts at this stage, as it tended to confuse him/her. The forms are presented in a list fashion and it is possible to pick previous specifications that have been used before as they are always presented in a separate list box. Each stage of the design process is presented in a similar fashion, so as to keep consistency and hence familiarity. Figure 8 shows example screenshots of specifications input. Once each stage has been finished, an option for entering ratings for each 'row' and 'column' of the corresponding chart is presented. This is not presented in a table format as it can become confusing and tedious for the designer when designing complex products. Two list boxes are available, one for the rows and the other for the columns. The designer selects the relevant row and respective columns and selects a rating. After the designer has fed this information, a chart is then created, showing a sketch of the relevant stage along with its absolute and relative importance ratings. Figure 9 contains an example conceptual chart.

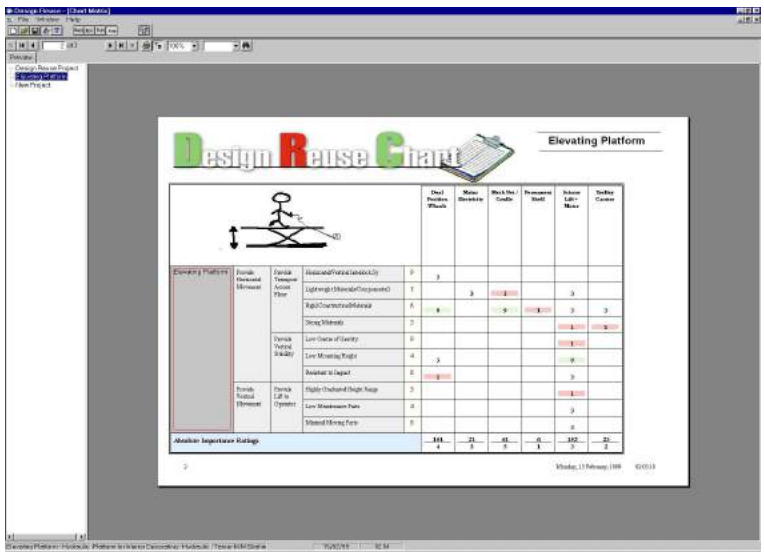


Fig. 9. Example Stage 1 – Conceptual Design.

### 6.4 Partitioning past information

As mentioned in section 5.4, a four-stepped process named Design Interpretation was established for this purpose. This process is achieved by starting with the detailed drawings and the part definitions. The software input process then essentially in reverse to the process described in the previous section, using a step by step process. Again information is stored in the database in the relevant places. This enables each stage to be accessed later on when designing a new product.

### 6.5 Reusing Design information

For ease of viewing and retrieval, the Conceptual and Embodiment Charts are generated 'on-the-fly' and information can also exported to HTML files for viewing over a company's Intranet.

A search facility is available at any stage of the design process and the database is searchable by any data model output (Specifications, Functions, Embodiments or Part). Search results will display where the search criteria have been used before as well as their relevant ratings. It is also possible to view further down or up the design process to see how this item is related throughout the entire process. It is then possible to import any piece of the information or all of it into another design. The program also keeps track of the work the designer is carrying out and will automatically search new information being entered for similar past designs. It then points this out to the user in a 'wizard' form and displays the relevant project, date, design information and ratings, with the option of importing this information into the current project. A 'Reuse' Switchboard, which allows the user to reuse any part of past designs into the current design, is also available.

The designs are colour-coded by their ratings for easy visualisation and identification of good or bad designs (e.g. good designs in green, bad in red)

Finally an interface to a Feature-Based Modelling package (in this case SolidWorks) has been implemented, which links to the correct geometric models of relevant parts. However, the Generative methodology is implemented within the

Design Reuse software and controls the Feature-Based Model to allow for advanced topological changes. The interface can be created for any Geometric modelling package running under Windows 98/NT, such as Pro/Engineer. Figure 10 shows an example of the interface at Detailed Design Stage.


Fig. 10. Detailed Design Reuse using Feature-Based Modelling.

## 7. CONCLUSIONS

A Design Reuse System has been presented as an integral part of a systematic design process. A new computer-based application has been created to cater for this, bearing in mind the designer. Each Stage of the Design Process can be search for and retrieved. Selected Information can then be used (whether good or bad) in a new project.

This has been done by (a) establishing DFD as the design model that is to be the underlying system for Design Reuse, (b) establishing prioritised Functions to represent the Product Concept, Function Family Based Chart to represent the Solution Concept, Parts Tree-based Chart to represent Embodiments and Generative Feature-Based Modelling to represent the Detailed Design Stages (c) Used ratings technique inherent within DFD as a technique to identify good or bad designs for future reference, and (d) Developed a stand-alone package that utilises SQL relational datastructure as the basis for storing and retrieving past designs as stipulated within the Design Reuse System.

## 8. REFERENCES


[1] Broens R.C.J.A.M.; de Vries M.J., Classifying technological knowledge for presentation to mechanical engineering designers, *Design Studies*, Vol. 24, No. 5, 2003, pp. 457-471(15)

[2] Khadilkar D. and Stauffer L., Evaluating the Reuse of Design Information: An Experimental Case Study, ICED95, Praha, August 22 – 24, 1995

[3] Kimpton M. and Sivaloganathan S., Systematic Design of and Elevating Platform, *Design Reuse*, Professional Engineering Publishing Limited, Bury St. Edmonds and London, UK, ISBN 1 86058 132 3, 1998.

[4] King A.M. and Sivaloganathan S., Development Of A Methodology For Using Function Analysis In Flexible Design Strategies, *IMechE Proceedings Part B, Journal of Engineering Manufacture*, 1998

[5] Mary Lou Maher et al. *Case-Based Reasoning in Design*, Lawrence Erlbaum Assoc, June 1995

[6] Shah J.J and Mantyla M., *Parametric and Feature-Based CAD/CAM : Concepts, Techniques, and Applications,* John Wiley & Sons; ISBN: 0471002143, October 1995

[7] Shahin T.M.M., Automation of Feature-Based Modelling and Finite Element Analysis for Optimal Design, PhD Thesis, Brunel University, December, 1996.

[8] Sivaloganathan et al. Quality Function Deployment – State of the Art Survey, CEERA Journal, 1996

[9] Sivaloganathan S., Shahin T.M.M., Cross, M. Lawrence M., ‘A Hybrid Systematic And Conventional Approach For The Design And Development Of A Product: A Case Study’, Accepted for Publication in Design Studies.

[10] Sivaloganathan S.; Andrews P. T. J.; Shahin T. M. M. , Design function deployment: a tutorial introduction , *Journal of Engineering Design*, Vol. 12, No. 1, 2001, pp. 59-74(16)

[11] Sivaloganathan S.; Shahin T. M, Design Reuse: An Overview, *Proceedings of the I MECH E Part B Journal of Engineering Manufacture*, Vol. 213, No. 7, 1999, pp. 641-654(14)