INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED'09
24 - 27 AUGUST 2009, STANFORD UNIVERSITY, STANFORD, CA, USA

DESCRIBING INFORMATION USE IN ENGINEERING DESIGN PROCESSES USING A DIAGRAMMATIC MODEL

Khadidja Grebici[1], David C Wynn[2] and P John Clarkson[2]
(1) McGill University
(2) Engineering Design Centre, University of Cambridge

ABSTRACT

Methods and tools to support engineering design should be based on an understanding of how it is conducted in practice and how information is used in the design process. The development of better tools and methods could thus be facilitated by a better understanding of information use in design practice. This paper explores how information use in design could be captured in-situ by process participants using a diagrammatic node-link modelling tool. Through literature review we synthesise a 'language' of concepts which we propose can be used by designers to describe information use in their work. The language contains classes for describing concepts in the product, process, specification and rationale domains. It is implemented in a diagrammatic modelling tool, which we illustrate by application to describe a design process fragment as conducted in a UK manufacturing company. We argue that studying the models designers create using this tool could provide insights into information use in practice and thereby support development of better design process support approaches.

Keywords: Design information use, integrated model, design methods/approaches

1 INTRODUCTION

Engineering design involves co-ordination between multiple actors playing different roles and with expertise in different methods and disciplines. Consequently, a variety of information models are used by these actors during the design process when they manipulate different aspects of the design for different purposes. The relationships between these models are often not easy to discern because they are implicit in the flow of work through the design process and in the transactions between participants. Because of the disparate forms of information and the lack of explicit links between them, it can be difficult to explore how the collaborative design process is conducted–how the different activities and contributions interface, how the information models could be enhanced or integrated to better support design, and where there are opportunities for further investigation.

This paper discusses an approach to explore how information is used in the design process and ultimately how these information needs could be better supported. Our approach is based on a diagrammatic software tool developed to enable the in-situ capture of collaborative design processes and specifically how information is used within them. Previous work by Bracewell et al. [1] has found that such diagrammatic modelling tools are believed by designers to actually support their work by making design reasoning more explicit. This finding that designers are willing to use such tools in their daily work indicates that they are well-suited to support the investigation of how design is conducted in practice, thus avoiding some of the limitations of studying design in a simplified or experimental setting. Bracewell et al.'s work focuses on the representation of design rationale and does not focus on information usage. In this paper, we develop an approach to allow diagrammatic modelling of elements representing aspects of the design process pertinent to information use. The types of element which can be modelled were determined by examining the information requirements of some of the main design approaches found in the literature, alongside descriptive theories of how design is conducted. The resulting approach allows description of concepts in the product, process, specifications and rationale domains and the links between these elements.

We argue that studying the models that process participants create using our tool could provide a useful way to investigate how designers organise and conduct their work, and specifically how they use information within the design process. The insights resulting from such investigation could ultimately guide the development of more appropriate process support tools which are based on an enhanced understanding of designers' information needs in practice.

The paper is organised as follows. Section 2 discusses our research methodology. Section 3 discusses some key methods used during the engineering design process, representative of those whose use we wish to model. Section 4 discusses types of information required to describe design problem solving, as indicated by a representative sub-set of literature in this area. Section 5 then summarises by tabulating the information requirements of the methods and problem-solving approaches which were reviewed, alongside key concepts required to describe how these approaches are used. The set of information types and concepts identified provides the vocabulary we use to describe the design process and information use within it. Section 6 discusses how this meta-model is implemented in a diagrammatic process capture tool which allows the description of design processes in a semi-formal and intuitive way. Section 7 describes a case study of bevel gear design as conducted within a UK manufacturing company, illustrating how our approach could be used to describe design processes as conducted in practice. Section 8 compares the vocabulary of our approach to that provided by other modelling approaches in the literature which aim to capture multiple domains of the design process. Section 9 reflects upon the paper's contributions and highlights limitations and directions for future work. Section 10 concludes.

2 METHODOLOGY

As outlined above, the objective of this paper is to develop a graphical modelling language which allows expression of how information is used in the design process. To identify the concepts required in such a language, we therefore reviewed some of the main engineering design approaches, methods, models and techniques. By summarising the information requirements of these approaches, we identify a set of classes which can be used to describe how these methods are applied in context.

The review is organised to cover two complementary perspectives of design, which we argue are important to describe information use within the design process. Firstly, from a procedural point of view design can be perceived as the following of procedures, the application of design methods, and the use of tools – in other words, as the execution of tasks which require and produce information in a workflow which emerges during the design process. Secondly, design can be viewed on a lower level as the designer's reasoning about information in order to solve particular problems they encounter while performing tasks in this workflow. By basing our modelling language on 1) the types of information required to describe application of design approaches and methodologies; and 2) the types of information required to describe design reasoning, we argue that the resulting language will allow these complementary viewpoints to be described – how design procedures are organised, how they are supported by design reasoning, and how this depends upon information and assumptions.

The set of methods and design reasoning theories we discuss is not exhaustive, due to space constraints and the extensive body of relevant literature. For instance, it has been identified that 140 different methods exist in the DfX area alone [2], which forms only a small sub-set of our review's scope. However, since many of the design methods we reviewed have significant overlap in the concepts required for their describing their application, as do the theories of design reasoning we considered, we concluded that examination of a representative sub-set was sufficient for preliminary identification of the concepts required in the modelling language we developed. This conclusion was supported by the preliminary application of our approach to describe detailed design process data gathered during a case study, showing that the concepts we identified were sufficient to describe many interesting aspects of information use in practice.

A *grounded* approach [3] was used for analysis of the literature and conceptualisation of the information required to describe information use in the design process. After collecting relevant literature, an Excel spreadsheet was populated with detailed descriptions of the key information needs highlighted by each publication. Within this, the key information was highlighted using a series of codes which were extracted from the text in the spreadsheet. The codes were grouped into similar *concepts* in order to make them more workable. From these concepts, *classes* were formed. The purposes of use of the different classes were then studied and links between the concepts were thus defined and justified. In the final results, each synthesised class is documented by a set of notes or "memos" that consist of arguments describe the main concepts that form the class or are linked to it. Each memo also captures the reasons why the category was synthesised, as suggested by [3].

The literature review is described below, prior to discussing the approach which was synthesised.

3 DESCRIBING DESIGN APPROACHES AND METHODOLOGIES

In our view of the design process, methods such as QFD, DFSS, reliability analysis, robust design, DfX etc. are applied to fulfil different objectives related to the product and/or manufacturing process. Each method can be applied to meet certain design objectives, and has certain information requirements. In this section, representative methods are reviewed to allow the information required to describe their application to be identified. Since it is not possible to analyse every design method found in the literature, we focus the review as follows: a) the methods target both functional and non-functional design objectives, b) they cover different stages of the product life-cycle; and c) they are generic across industry sectors.

The discussion is organised into the following sections. Firstly, we discuss general design and design management methods. We then discuss conceptual design methods, followed by methods used during detail design to meet specific objectives. Finally, we discuss literature on design processes and engineering systems to cover the need for the graphical language to allow application of these methods to be described not only individually, but also in the context of the design process. Following discussion of design reasoning theories in Section 4, Section 5 summarises the approaches by highlighting the classes which our graphical language provides for describing their application. In the text, italicised terms indicate classes identified through the grounded approach described in Section 2.

3.1 Information requirements of general design and design management methods

This section describes five representative design and design management approaches which can be used in different stages of the process to support general, high-level objectives. The purpose of this description is to highlight the key information required to describe application of the methods.

- **DFSS (Design For Six Sigma)** is a set of approaches/techniques that aim to assist in achieving quality, reliability and performance objectives through customer-driven product design and operation. Reich [4] stated that DFSS approaches should be complemented by explicit capture of knowledge about the *design alternatives*, their *priorities*, the *success criteria* used for their assessment and the *decisions* underlying their selection. In addition, the purpose of DFSS approaches is to facilitate the flow of creativity rather than to provide traceability of solutions which are generated. Therefore, it has been pointed out in [5] that not only the information used explicitly by DFSS has to be captured but also the *rationale* behind the decision-making.
- **QFD, via the HOQ (House of Quality) matrix** starts with identifying the needs for a new product [6]. Then, the approach considers the *customer requirements* which are ranked and assessed against the *product characteristics*, which in turn are assessed against the *process control variables*. To describe HOQ elaboration activities, the review revealed that it is necessary to associate *rationale* underlying the choice of values of the *customer attributes* (e.g. the mean time to failure), the *product characteristics* or the *process control variables* explicitly captured in the QFD matrices.
- **Failure Modes and Effects Analysis (FMEA)** is an approach to identify the possible *failure*s in a system for which design attention is likely to yield the most improvement in terms of severity, frequency and detectability of failures. The *product attributes* or *functions* are listed on the left-hand side of a matrix, and the likely *failure modes*, their *effects* as well as their *causes* are listed along the side and across the top of the matrix. The *risk of failures* can then be assessed and mitigating actions identified.
- **Fault Tree Analysis (FTA)** [7] provides a picture of the interrelationships between the modes of failure identified in FMEA. It includes both fault simulation and fault tree analysis activities. Fault simulation examines what fault*s* may arise in the parts and relationships among the parts and how the faults may affect the whole system. A key pre-requisite to this activity is a model representing the *product structure* – for instance, *parts* and their *relationships*. Other information which can be used in fault simulation includes *abnormal states arising in relationships between parts* and *abnormal part states* caused by abnormal relationship states. Decision tables show the *relations* among the *failure modes*, input and output *parameters* associated with parts. Failure modes within the decision tables are considered in conjunction with *environmental conditions*.

- **Cause-effect analyses** are used to identify issues such as *failure mechanisms* or *component faults* and to analyse the *reasons* underlying their occurrence. Knowledge of the origin of problems and the potential deviations should be captured and traceable. Information requirements include the *methods* used during design, the *machines/tools* selected for *manufacturing processes*, the *material* used, the measurements and *tests* undertaken, the *environmental constraints* considered, and information about customer usage. Issues which are fed back include *failure mechanisms*, *component faults*, etc. Cause-effect analyses are often supported by visual graphs such as Fishbone diagrams or IBIS charts [1]. Such diagrams can depict *decisions*, *evidence* and the *tasks* which are performed to acquire it.

3.2 Information requirements of concept design methods

This sub-section discusses two representative design methods focused on concept generation.

- **Brainstorming.** Designers often perform brainstorming and affinity activities in order to generate meaningful groups of ideas or concepts (*principles of solution*) synthesised from the functions (*functional requirements*) [8]. They can consider the results of previous sessions including, for instance, the *criteria* and *constraints* considered and the *rationale behind decisions* regarding concept grouping. The results are used in other design methods; for instance, they could provide the *product characteristics* modelled in the QFD matrices discussed above.
- **TRIZ.** The theory of Inventive Problem Solving (TRIZ) [9] aims to assist in delivering innovation by tackling the technical contradictions in engineering design problems. More specifically, the TRIZ method consists of deriving the functions that represent technical contradictions. Firstly, from the analysis of the *customer needs*, it consists of deriving the *functions* that represent technical contradictions. Then, the mechanisms that should be innovated are identified in each *component/system/sub-system.* The technical innovation is supported through TRIZ resulting in generation of new ideas. After this, function and mechanism (technological system) decomposition diagrams can be used to establish the relationship between technical solutions and functions.

3.3 Information requirements of detail design methods to meet specific objectives

This sub-section completes the discussion of representative design methods, describing four methods used to meet specific objectives during detail design.

- **Functional cost analysis** is a technique based on breaking down a *product* into its *component parts* and then contrasting the *costs* of those parts with the *functions* being provided [10]. It is carried out in a matrix where the rows show the component/part *costs* and the columns show functions. The *manufacturing costs* of the components are isolated and put in the matrix. The high cost areas versus functions are thus highlighted. As the technique requires detailed knowledge of the design and component costs, it is usually used as a check on designs prior to manufacturing.
- **Process tolerance** activities include parts tolerance analyses and cost of tolerance assessments [11]. Tolerance analysis essentially consists of optimising the tolerance ranges of the parts (e.g. diameter tolerances) before their manufacturing and assembly process and the definition of the subsequent manufacturing and assembly operations. Cost tolerance analysis refers to estimating the expenditures needed to achieve certain level of *dimensional* and *geometrical tolerance* accuracy (including rework). Those costs are usually a function of design and *machine tools*. Thus, information required for these two analyses includes *part geometry* and *dimensions* (diameter, length and location), knowledge about their distribution (e.g. mean, standard deviation, etc.), the list of *equipment* and the associated *operations* (e.g. turning, milting, drilling, etc.), machine particulars, *machine operations* and *equipment selection*.
- **Design for Reliability.** DfR-related activities are performed throughout the design process. They include identification of reliability requirements, development of reliability targets, reliability evaluation and the comparison of test results to predictions [12]. Perhaps the most important activity is reliability evaluation, which is performed through 1) product modelling and 2) reliability analysis. The first activity consists of modelling *assemblies, parts and features*, and the second activity involves analysis of the potential *failures* and/or *faults*, e.g. using FTA/FMEA as described above.

- **Taguchi Robust Design.** The Taguchi methods are a set of methods where specified product *parameters* are thought of in terms of *target values,* which are met by controlling *(manufacturing) processes*. Any deviation from the target values would incur *costs* (or quality loss) [13]. The basic Taguchi approach is applied at the design stage, to assist selection of a system or functional design to reduce variability in performance of the manufactured product. The method leads the designer into determining optimum parameters of *system/component*, etc. having fully investigated the variability, or more specifically the sensitivity of the system specification to the causes of variability.

3.4 General information requirements for describing design processes

This section discusses general concepts of design process organisation. This allows identification of concepts required to describe the application of design methods within the process context.

- **Describing design processes.** Many authors such as Pahl and Beitz [14] propose a systematic design approach, which argues that complex problems such as the design process are best tackled in fixed steps. Design work is considered as the conversion of information. After each step, it may become necessary to upgrade or improve the results of the last. The splitting of the design process into steps ensures that the essential links between objectives, planning, implementation and checking are maintained. The information about the nature of outputs of each stage and their feedback and forward links are described. The main phases involved are: *clarification of the tasks*, *conceptual design*, *embodiment design*, and *detail design*. All these steps, as mentioned above are based on *procedural steps* or *tasks* whose execution necessitates *synthesis, analysis,* and *evaluation activities*.
- **Describing technical systems.** In engineering design literature [15], technical systems are often described in terms of *structure, features* and specific external and internal *properties*. Pahl & Beitz [14] further divide the concept of *feature* into *functional*, *physical* and *interface features*. External properties are those which can be measured from the completed physical product. Internal properties are those which can be identified in design descriptions.

4 DESCRIBING DESIGN THINKING PROCESSES

In an empirical study of the design process in industry, Hales [16] recorded and categorised the activities involved in a specific industrial design process. The categories of design process information he recorded accounted for only 47% of the information specified by Pahl & Beitz [14]; new categories of activities were needed to account for 53% of general design activities. These results have inspired additional research in the areas of design thinking and design practice, where different classification schemes of the design activities have been provided. For instance, Jagtap et al. [22] introduce a generic design model encompassing the following generic activities: *explanatory generation*, *constructive generation (synthesis)*, *evaluation*, *analysis* and *comparison*. These categories correspond to the generic types of low-level activities that account for problem definition, conceptualisation, analysis and evaluation of the design solution. Similarly, [17] [18] [19] consider the design process as a human problem-solving process.

Researchers such as [20], [17] [21] [22] and [19] argue that the design participants could be guided more effectively by considering the question-asking behind the execution of activities. Question-asking is treated as a process whose investigation could reveal how to prompt the design participants with relevant information at the right time to undertake the right activities.

A large taxonomy of types of question can be considered [23]. It is also important to consider the knowledge and rationale basis the team uses for breaking down and structuring the project into design phases, where the timing and nature of the questions can impact strongly upon the behaviour of the design participants [23]. The review undertaken by Ozgur [23] identifies 52 key questions that should be modelled around the generic engineering activities. In fact, these generic questions are asked to allow the designer (1) to explore a given activity topic or (2) to explore a series of activities of different topics.

Furthermore, additional to the categories of questions, the review highlighted that categories of subjects sought by these questions are relevant to describing how designers accomplish their activities. These categories include: *issues*, *functions*, *intended attributes*, *predicted attributes, constraints, structure and features*. Apart from the *issue* class, the latter categories are all reflected in the literature discussed in Section 3. According to [22] and [24], the categories of issues should be broader than the in-service and operation related ones (e.g. failure modes). According to the *life-cycle phase* considered and the *type of problem analysis* applied, different types of issue may become apparent.

One well-known approach for modelling these aspects of how information is used to resolve problems encountered during design is the Graphical Issue-Based Information System (gIBIS)[25], which provides a way of capturing the *issues*, *arguments* and *positions* taken by team members or individuals engaged in the engineering process. Linking and structuring the rationale in this way is often claimed to improve the understanding of the design process by the participants (e.g. by helping them articulate why a certain position has been taken) e.g. [1]. However, experience of such approaches in industry has indicated that IBIS structures tend to capture unconnected fragments of the engineering process. Thus existing IBIS tools are not sufficient to address the objectives laid out in Section 1.

5 SYNTHESISING A GRAPHICAL LANGUAGE

Sections 3 and 4 highlight that there is a wide range of methods and approaches which are used in design, as well as different theories of design reasoning. However, there is significant overlap in the concepts which are considered in each approach. Applying the analysis and clustering steps described in Section 2 resulted in the hierarchy of classes and their icons used in the graphical language. This is summarised in Table 1. There are fewer icons than classes; due to the large number of classes, those which are closely related are displayed in the same way to assist in reading the diagrams. Where multiple classes use the same icon, the modeller can select which class is intended when a given concept is modelled by using a drop-down list in the software implementation described below. The possible links between concepts of different classes are not constrained; we allow any two elements to be linked to one another using an arrow in the diagrammatic model, regardless of their classes. This offers flexibility allowing description of processes according to the modeller's preference.

Table 1. Summary of classes in the graphical language. Italics indicate abstract classes which cannot be used directly in the modelling notation.

PRODUCT DOMAIN

	6	*Product and manufacture*
⬭	6.1	**Physical phenomenon**
⬭	6.2	**Physical effect**
⬭	6.3	**Material**
	6.4	*Product structure*
⚙	6.4.1	**System**
⚙	6.4.2	**Assembly**
⚙	6.4.3	**Component**
⚙	6.4.4	**Part**
	6.5	*Product feature*
	6.5.1	*Physical feature*
⬭	6.5.1.1	**Dimension**
⬭	6.5.1.2	**Geometry**
⬭	6.5.2	**Functional feature**
⬭	6.5.3	**Interface feature**
	6.6	*Attribute(expected/predicted/observed)*
⬭	6.6.1	**Product**
⬭	6.6.2	**Process**
⬭	6.6.3	**Interface/environment**

	5	*Issue*
	5.1	*Product lifecycle*
!	5.1.1	**Development**
!	5.1.2	**Manufacturing/assy.**
	5.1.3	*In-service operation*
!	5.1.3.1	**Failure mnsm.**
!	5.1.3.2	**Failure effect**
!	5.1.4	**Disposal**
!	5.2	**Product characteristic**
!	5.3	**Functioning/fault**

SPECIFICATION DOMAIN

	7	*Specification*
	7.1	*Function*
F	7.1.1	**Elementary**
F	7.1.2	**Composite**
	7.2	*Requirement*
R	7.2.1	**Constraint**
R	7.2.2	**Criteria**
R	7.2.3	**Functional**
R	7.2.4	**Other**
$	7.3	**Cost**

PROCESS DOMAIN

	2	*Design process*
▭	2.1	**Life-cycle phase**
▥	2.2	**Task**
◆	2.3	**Decision**
	2.4	*Activity*
	2.4.1	*Analysis*
▥	2.4.1.1	**Constructive**
▥	2.4.1.2	**Explanatory**
▥	2.4.2	**Synthesis**
◆	2.4.3	**Evaluation**

	4	*Representation*
	4.1	*Domain-specific*
▦	4.1.1	**Physical structure**
▦	4.1.2	**Geometry**
▦	4.1.3	**Analysis/test/log**
▦	4.2	**Procedural**

	3	*Resource*
■	3.1	**Technology**
	3.2	*Tool*
■	3.2.1	**Machine**
■	3.2.2	**Software**
■	3.3	**Method**
■	3.4	**Human**

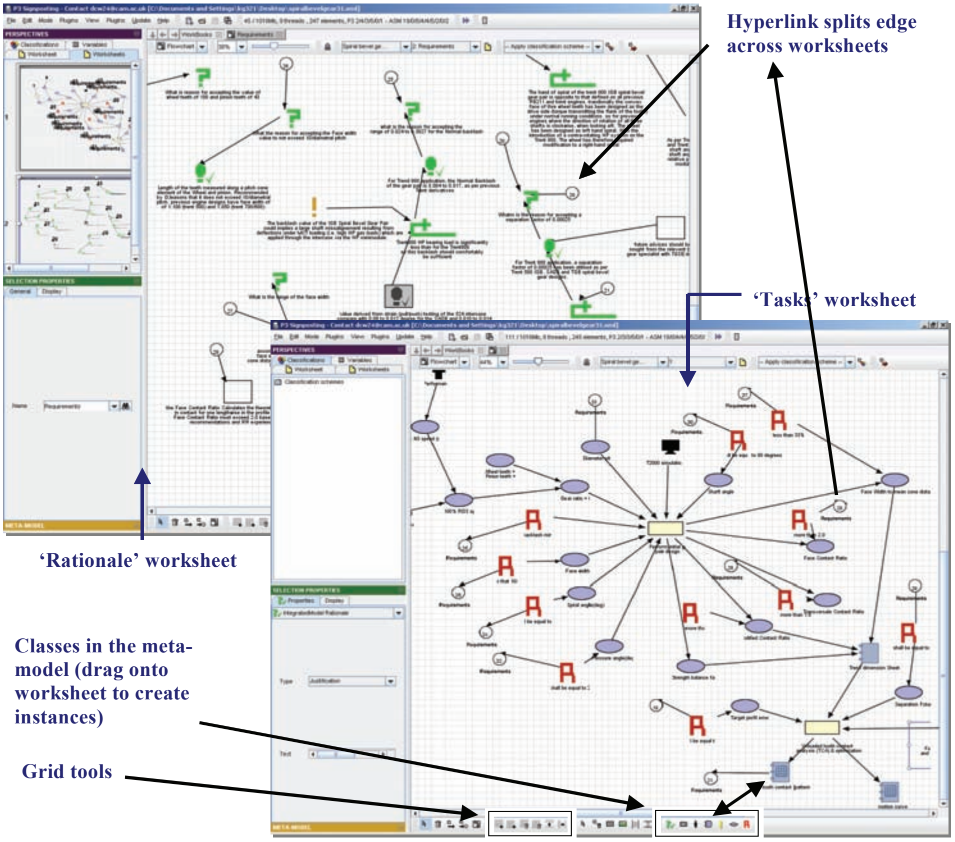
RATIONALE DOMAIN

	1	*Rationale*
🟩	1.1	**Question**
	1.2	*Factual*
🟩	1.2.1	**Past design**
🟩	1.2.2	**Current design**
	1.3	*Argument*
🟩	1.3.1	**Pro argument**
🟥	1.3.2	**Con argument**
✓	1.4	**Position**
✓	1.5	**Solution principle**
✓	1.6	**Evidence**

6 IMPLEMENTATION IN A DIAGRAMMATIC MODELLING TOOL

The tool for modelling information use in the design process was implemented as a linkage meta-model in the P3 Platform software [26]. P3 is a general-purpose diagrammatic modelling tool which can be configured for building informal or formal models of classes and their relationships. Within P3, a linkage meta-model is a formalised description of the elements and relationships allowed in a particular type of model. The linkage meta-model also configures the user interface of the tool to allow construction of models of that particular type [26].

P3 provides a number of features that are important to the application in this paper. Firstly, it is possible to split models across multiple 'worksheets' via the use of 'hyperlinks' which split arrows (small circles in Figure 1). Secondly, individual worksheets can be hierarchically decomposed into 'sub-sheets' which can be opened and closed, to as many levels as necessary. A managed layout algorithm ensures that, when sub-sheets are opened and closed, gridlines upon which elements are placed are 'stretched' such that the positioning of elements above, below, left or right of one another is maintained. Thirdly, it is possible to insert space within an existing diagram by adding vertical or horizontal gridlines, making it easy to extend a model without moving nodes individually. These features ease the development of large models which would be difficult in a general-purpose diagramming tool, but which the case study outlined below showed are necessary to capture information use in the design process.



Hyperlink splits edge across worksheets

'Tasks' worksheet

'Rationale' worksheet

Classes in the meta-model (drag onto worksheet to create instances)

Grid tools

Figure 1. The integrated knowledge model forms the basis of a diagrammatic capture tool

An example of a model created using the system is shown in Figure 1. Our approach does not enforce a particular organisation of the elements which comprise a model; in this example the model is organised into two worksheets – one focusing on the process domain and the other on the rationale domain. Each node on the diagram represents an instance of a particular class. The shape of the node indicates its class, as defined in Table 1, and the text indicates the specific detail of the instance (for example, a description of the particular task which the node represents). All classes are grouped into the six super-classes as shown in Table 1. Nodes representing classes from the meta-model are created

by dragging one of the six super-classes from the toolbar onto the worksheet. The properties and subclass of the element may then be modified. Classes may be joined together with directed edges. If necessary, edges may also be labelled to indicate the nature of the relationship between two nodes.
Figure 1 illustrates how the notation may be used. In this example, the 'tasks' worksheet shows *Tasks* and their inputs and outputs which could be different types of *Product and Manufacture* or *Representation* element. *Resources* are used to perform the tasks – e.g., *Software Resource*, *Human Resource*, etc. *Product and Manufacture* elements are used to describe any predicted or expected attribute of the product, process, project or customer. *Representation* elements indicate the information models used to perform tasks, which may be linked to the *Product and Manufacture* elements they describe, which in turn can be connected to the *Requirements* and *Constraints* that influence them. The *Rationale* elements associated with a task or *Product and Manufacture* element indicate the specific process by which the task was conducted or the value determined. Rationale processes are typically modelled to lead from *Questions* raised when considering a particular design problem to *Answers*, which are justified by the associated *Pro Arguments* and *Con Arguments*. In turn these may lead to further questions and answers. The rationale processes may be linked to other elements which further qualify them by indicating their context. *Evidence* elements may be used to indicate the ultimate foundation of rationale graphs on tests, experiments or measurements. To show the provenance of evidence, links could be created to indicate the tasks involved in tests, experiments, etc. Finally, *Issue* elements indicate contingent or contextual aspects of rationale and are used to qualify the rationale processes which lead to an answer. For instance, they may be used to indicate that evidence only supports an argument under certain circumstances.

7 ILLUSTRATIVE APPLICATION

This section illustrates the approach through a case study, in which the authors used data gathered from a UK manufacturing company to construct a model of a small fragment of a gear pair design process. The aim is to illustrate that the integrated meta-model and capture tool outlined above can be applied to model the engineering design process as conducted in practice. We therefore constructed a model by analysing some Design Definition Reports (DDRs) obtained from the case study company. The DDRs were particularly suitable for this task, as they had been created using an existing software tool known as DKC that captured semi-structured knowledge linked to geometric features, by prompting designers to answer a list of standard questions for each feature (e.g. how is this feature designed? Why is this feature here?). Two parts of the resulting model are visible in Figure 1.
The purpose of the Spiral Bevel Gear pair whose design process we examined is to transmit torque between an input and output shaft. The fragment of the design process visible in Figure 1 (Tasks worksheet) shows the initial gear pair design task (on the top-left side of the tasks sheet) which is undertaken using optimisation software that allows the definition and adjustment of the wheel teeth and the pinion teeth contact ratios, such as the face width to mean distance ratio, transverse contact ratio and modified contact ratio (*Product and Manufacture* elements in the tasks worksheet). The inputs to this task consist of the dimensions of the gear and teeth alongside other ratios and angles. By changing the set of parameters such as face width, normal backlash and shaft angle, the software used to perform this task allows the calculation of the above ratios throughout different iterations.
In order to predict the sensitivity of the contact pattern to small amounts of relative vertical and horizontal movements between the gear pair axes, the designer performs the "unloaded tooth contact analysis and optimisation" (TCA) task, which uses information produced by the first task and which is visible on the bottom-right of the tasks worksheet. The contact patterns and motion transmission are optimised by introducing small amount of surface mismatch (lengthwise, width wise or flank twist). The outputs of the TCA task are the *tooth contact plots* and the *motion curves*, which are shown as "representations" resulting from the TCA task in Figure 1. The tooth contact plots show the tooth contact patterns for the convex and concave sides as they are viewed in a test machine. The position of the transfer points on the motion graph gives an indication of the smoothness and quietness of operation and is referred to as the 'motion error'.
As the designers were entering data into the DKC tool to indicate how the bevel gear was designed and analysed, they were providing answers to questions explicitly posed as, for instance, "what is the reason for accepting the constraints (certain value or a range of values) for a given parameter". In the example diagrammatic model, we therefore created hyperlinks linking the considered element within

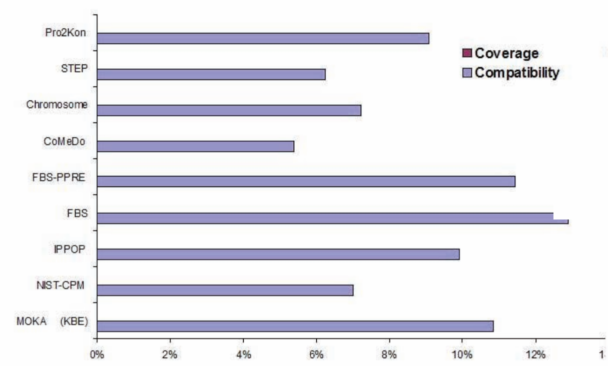
the tasks sheet such as the constraints (e.g. the constraint on the Backlash parameter, top-left of the tasks sheet in Figure 1) to a destination element in the rationale sheet (e.g the Question "what is the reason for accepting the range for the backlash parameter", on the top-left of the rationale sheet).

As mentioned previously, rationale processes modelled in the tool typically progress from a question through answers towards evidences. This is the case for the justification process regarding the choice of allowable range of the normal backlash, where the evidence is provided by testing the intercase (rationale worksheet). The issues are the other elements that document the rationale process with the potential difficulties regarding the functioning, manufacturing, assembly, etc. of the artefact. In the case of the normal backlash, the issue we modelled represents the deflection of the shaft under the MOT loads. By specifying this issue, we highlight the fact that even if the bearing load is proven to be significantly less than for a given previous design, and the backlash is consequently sufficient, the Pro Argument should be tied to the consequences of the choice of the backlash values on the incidence of the shaft deflection which inevitably implies a shaft misalignment. For instance, this contingency could be the backbone upon which the designers justify their arguments, or demonstrate how the evidence does not cover certain key assumptions in the decision-making process.

In summary, therefore, the simple example presented in this section illustrates that the proposed approach can be used to capture the design process, how different aspects of information is used in that process, and the rationale which indicates why certain decisions are made. This example cannot cover all aspects of the approach presented in prior sections. For instance, one such aspect is the ability to distinguish between different types of questions which can be asked during design. Although our illustrative example is thus limited in both scope and depth, it does show that it is possible and reasonable to capture the design process using our approach.

8 COMPARING THE APPROACH'S VOCABULARY TO STATE-OF-THE-ART

To compare the capability of our approach to capture the information needs of each objective with that of models in the design literature which attempt to capture multiple domains of the design process ("integrated models"), we took a similar approach to Wyatt et al. [2]. This is based on two simple metrics: *coverage* and *compatibility*. The former is defined as the fraction of the information needs of a design method that a given integrated model can provide. The information needs we consider are those summarised in Table 1. The *compatibility* of a model is defined as the proportion of all design methods reviewed in Section 3 for which that model provides all the required information. The formulae to calculate these metrics, as presented by [2], are shown for reference in Figure 2.



$$compatibility_{ij} = \frac{\sum_{k=1}^{D} p_{ik} u_{jk}}{\sum_{k=1}^{D} u_{jk}}$$

$$coverage\ score_i = \frac{\sum_{j=1}^{M} c_{ij}}{M}$$

$$compatibility\ score_i = \frac{\sum_{j=1}^{M} compatibility_{ij}}{M}$$

- Compatibility$_{ij}$ = the compatibility between integrated model i and objective j
- compatibility score$_i$ = the overall compatibility score of integrated model i (as plotted)
- coverage score$_i$ = the coverage score of integrated model i.

where…

- p_{ik} = 1 if data item k is provided by integrated model i, or 0 otherwise
- u_{jk} = 1 if data item k is used by objective j, or 0 otherwise
- R, M, D = the total number of {integrated models, objectives, data items} considered (9, 31 and 91 respectively)
- c_{ij} = 1 if compatibility$_{ij}$ is 1, or 0 otherwise.

Figure 2. Evaluation of other modelling approaches according to compatibility and comparability metrics, as defined by Wyatt et al. [2]

We used this approach to compare the coverage and compatibility of our representation against the nine integrated models reviewed by Heisig et al. [27]. The resulting scores, shown in Figure 2, show that our approach has significantly greater coverage of the information requirements of the design methods reviewed than these other approaches. The compatibilities shown in Figure 2 are all lower than 13%. Furthermore, the coverage score of every model is zero – indicating that no model can represent the domains of information needed to describe the application of a single design method. This is perhaps not surprising since many of the models considered were not developed for this purpose. However, it does indicate the contribution of this paper in introducing a new integrated model to address issues which were not previously considered in depth.

9 DISCUSSION AND FUTURE WORK

To recap, the modelling approach developed in this paper aims to provide a way to capture how the design process unfolds as designers undertake the design activity. The diagrammatic model and implementation is intended as a step towards answering the broader research question: How do designers derive the output from a given design task and what drives the performance requirements considered when synthesising, analysing and evaluating the solution? The tool provides one way to model how information is used during design, with respect to the design process, design methods and design reasoning. The key elements surrounding information use in design and their relationships are expressed using simple symbols and hyperlink technology, which have been shown by Bracewell et al. [1] to be intuitive to use in practice.

There are many opportunities to extend and apply the approach proposed in this paper, including:

- **Deploy the approach in a design process to evaluate and refine it.** This paper has reported on the development of the approach and its initial validation in a laboratory context. To take this research further the next step is to provide the tool to some practicing designers, either in laboratory experiments with a simple ‘toy’ design problem or in an industry setting. The empirical data gathered from this application will allow evaluation and further refinement of the ideas developed in this paper. The modelling approach on which the tool is based should also be evaluated against some basic criteria, including: completeness, redundancy, and the degree to which terms in the graphical notation conform to the conceptual model of the users [28].
- **Understanding how processes are represented in the tool.** There is an inherent difficulty describing any design process in a sequential manner as would be necessary in our notation, since it is difficult to identify whether new information/concepts/activities are created or existing ones are revisited. There are also issues regarding the different viewpoints associated with different design methods, and which might not be sufficiently compatible for description within one model. The problem of using different terminologies to describe the same concepts is also not considered in this paper. For rigorous analysis of processes captured in the tool, it will thus be necessary to understand in greater detail how designers think about their processes when using it and how this would be simplified for representation in our modelling language. It seems likely, for instance, that a different view of the process would become apparent than that revealed through a textual narrative description or through a ‘think-aloud’ protocol.
- **Evaluate whether greater formalism is needed.** The system described in this paper allows concepts and their relationships to be described in a free-form way, organised and connected according to the modeller’s preference. However, due to the relatively large number of classes available in the modelling language, the tool could be confusing to use in practice. One possible solution is to incorporate a formal ‘grammar’ within the system (for instance, requiring every evaluation activity to be connected to one or more specification objects representing the evaluation criteria). Further work is needed to explore how such structures could be formalised and whether this would enhance the usability of the approach.

10 CONCLUSIONS

The use of information in the design process is difficult to discern, because the relationships between disparate information models used during design are implicit in the flow of work between activities and the transactions between process participants. This paper has proposed that a diagrammatic modelling tool could provide a useful way of capturing design processes at the point they are conducted, and that analysis of the models created using such a tool could lead to new insights into design processes and how they could better be supported. The paper has taken the view that a representation which captures engineering design information and how it is used in the design process should provide terminology to describe the elements of information required by the major design methods used during design, as well as the design reasoning conducted to resolve problems encountered in the process. We refer to the resulting model as an ‘integrated model’, since it allows description of classes in the product, process and rationale domains. The idea is to provide a simple language allowing designers to describe the links between separate types of information used during design. The vocabulary of the integrated model was synthesised by studying the classes required to describe application of a representative set of engineering design methods. This approach was based on a view of the design process as a workflow in which such methods are applied.

The present work has introduced the vocabulary for the construction of semi-formal diagrammatic models of design activities. We still have to bring more specification to how the vocabulary can be used in practice, through application of the approach to case studies in which the tool is applied by designers to capture their design processes. Future investigations are therefore planned to better understand the rules which should be used to indicate how the information elements can be linked to one another in a meaningful way. In conclusion, we propose that the lessons that could be learnt from application of the modelling approach proposed in this paper could provide researchers with new understanding of the design process and how it could be supported in practice.

REFERENCES

[1] Bracewell, R., H. , Synthesis Based on Function- Means Trees: Schemebuilder. In *Engineering Computational Design: Understanding approaches and tools*, 2002, pp. 199-212.

[2] Wyatt, D., Wynn D., Clarkson, P.J. Comparing Representations for Product Architecture Design through Life-Cycle Evaluation Methods, In *NordPLM’09,* Gothenburg, Sweden. 2009.

[3] Glaser, B. G., 1978, *Theoretical Sensitivity: Advances in the Methodology of Grounded Theory*, Sociology Press.

[4] Reich, Y. Improving the Rationale Capture Capability of QFD. In *Engineering with Computers*, 16, 2000, pp. 236-252.

[5] Bracewell, R., H., Gourtovaia, M., 2006, Knowledge Management in Design for Six Sigma: A Review for a High Performance and Robust Systems (Hyparsys) Project, Technical Report No. CUED/C-EDC/TR134, University of Cambridge, Cambridge, Uk.

[6] Akao, Y., 1990, *Quality Function Deployment* Productivity Press, Cambridge, MA.

[7] Suzuki, K., Kameyama, Y., 1989, Application of Knowledge Engineering to Automated Fault Tree Synthesis Based on Decision Table, *Safety Engineering* 28, pp. 291-303.

[8] Osborn, A., F., 1963, *Applied Imagination: Principles and Procedures of Creative Problem Solving*, New York: NY: Charles Scribner's Son.

[9] Altshuller, G., 1988, *Creativity as an Exact Science*, New York: Gordon and Breach,

[10] Akiyama, K., 1991, *Function Analysis: Systematic Improvement of Quality and Performance* Cambridge, MA: Productivity Press.

[11] Evans, D., H. Statistical Tolerances: The State of the Art, Part I. *Journal of Quality Technology*, 6, 1971, pp. 188-195.

[12] Birolini, A., 1993, *Concurrent Engineering: Automation, Tools, and Techniques*. New York: John Wiley & Sons, Inc.

[13] Tagushi, G., 1986, *Introduction to Quality Engineering*, *Unipub/Quality Resources* NY:White Plains, New York.

[14] Pahl, G., Beitz, W., 1988, *Engineering Design. A Systematic Appraoch* Addison Wesley.

[15] Hubka, V., Eder, W. Ernest, 1984, *Theory of Technical Systems. A Total Concept Theory for Reengineering Design*, Springer-Verlag.

[16] Hales, C. *Analysis of the Engineering Design Process in Industrial Context.* Ph.D. thesis, University of Cambridge, 1987.

[17] Kuffner, T. A., Ullman D., G. The Information Requests of Mechanical Design Engineers. *Design Studies*, 12, 1990, pp. 42-50.

[18] Waldron, M., B., Waldron, K. J., Abdelhamied, K. "The Differences in Reading Schematic Drawings of Mechanisms by Expert and Naive Mechanical Designers", in *International Conference on Engineering Design*, Boston, MA, 1987, pp. 15-21.

[19] Aurisicchio, M., Bracewell, R.H., Wallace, K. M. Characterising Design Questions That Involve Reasoning ", in *International Conference on Engineering Design ICED07*, Paris, France, 2007.
[20] Gruber, T. B. C., Boose, J., Weber, J., 1991, Design Rationale Capture as Knowledge Acquisition: Tradeoffs in the Design on Interactive Tools", in *Machine Learning eighth International Workshop*, Saint Mateo, CA, 1991, pp. 3-12.
[21] Marsh, M., J. *The Capture and Utilisation of Experience in Engineering Design* Ph.D. thesis, University of Cambridge, 1997.
[22] Jagtab, S., Johnson, A., Aurisicchio, M., Wallace, K.M. Causality in in-Service Information and Design Actions", in *International Symposium on Tools and Methods of Competitive Engineering (TMCE 2008),* Kusadasi, Turkey. 2008.
[23] Ozgur, E. *Perceiving, Comprehending and Measuring Design Activity through the Questions Asked While Designing*. Ph.D. thesis, University of Stanford, 2002.
[24] Ahmed, S., Wallace, K. M. and Langdon, P. Identifying the Stategic Knowledge of Experienced Designers", in *Strategic Knowledge and Concept Formation III*, Sydney, 2001, pp. 31-54.
[25] Conklin, E. J., Begeman, M. L. GIBIS: A Hypertext Tool for Exploratory Policy Discussion", in *Proceedings of CSCW 88*, New York, 1988, pp. 140-152.
[26] Wynn, D. C. *Model-Based Approaches to Support Process Improvement in Complex Product Development*, Ph.D. thesis, University of Cambridge 2007.
[27] Heisig, P., Grebici, K., Ariyo, L., Caldwell,N. H. M., Clarkson, P. J. "Towards an Integrated Product-Process-Rationale Framework for the Product Life Cycle", in *Knowledge and Information management through life*, University of Reading 2008.
[28] Ahmed, S. Encouraging Reuse of Design Knowledge: A Method to Index Knowledge. *Design Studies*, 26, 2005, pp. 565-592.
[29] Goulding, C. *Grounded Theory: A Practical Guide for Management Business and Market Researchers*, SAGE, 1992.
[30] Mill, *Procedure for Performing a Failure Mode Effect and Criticality Analysis* Technical Report No. MIL-P-1629, United States Military, 1949.
[31] NASA. *Fault Tree Handbook with Aerospace Applications*. Technical Report No. NASA Office of Safety and Mission Assurance NASA Headquarters, Washington DC, 2002.
[32] Baya, V. *Information Handling Behavior of Designers During Conceptual Design: Three Experiments*, Ph.D. thesis, University of Stanford, 1996.

Contact: Khadidja Grebici
McGill University
Macdonald Engineering Building,
817 Sherbrooke Street West
Montreal, Quebec H3A 2K6
Email: khadidja.grebici@mail.mcgill.ca

Dr. Khadidja Grebici holds an MSc and PhD in Industrial Engineering from the Institut National Polytechnique de Grenoble. She is presently a Postdoctorate Fellow at McGill University Faculty of Engineering.
Dr. David Wynn holds an MEng in Engineering and Computing Science from the University of Oxford and a PhD in Engineering Design from the University of Cambridge. He is presently a post-doctoral Research Associate in the Cambridge University Engineering Department and a Fellow of Homerton College, Cambridge where he teaches Mechanical Engineering.
Professor John Clarkson received a BA in Electrical Sciences and a PhD in Electrical Machines from Cambridge University. He worked in industry for 7 years before being appointed Director of the Cambridge Engineering Design Centre in 1997. He is presently Professor of Engineering Design at the University of Cambridge.