Open Research Online

PARAMETER TRAILS

T L Flanagan, C M Eckert and P J Clarkson

Abstract

Successful communication is vital for the success of any design project. However, communication often fails, adversely affecting design process efficiency and product quality. This is particularly true for complex products. As a result, designers and managers do not understand the connections between different aspects of design and don't know where to find out more information or who to talk to.

This paper presents a new model, developed from current project planning techniques, which supports communication using parameter-specific data. It enables designers to question information, inform their colleagues pro-actively and assess the impact of changing parameter values on subsequent design tasks. Such interaction is critical in allowing designers to see how their own tasks fit into the overall product design.

Keywords: design information management, planning and workflow methodology

1 Introduction

The design of large and complex products, such as aircraft, involves the collaboration of 100s of people doing a myriad of different tasks. Individuals often have very little idea how their own tasks fit into the context of the wider product despite being experts in their own field and understanding the tasks of the people they work with frequently. They may have little idea where information is coming from or going to and who is using it in the wider product context. Failure to exchange knowledge efficiently is often a symptom of communications problems. Even within well-structured organisations, where personal and cultural factors are minimal and designers do their best to communicate effectively, information flow continues to present difficulties.

Designers generate or are provided with specifications for their tasks, which are often expressed in terms of parameters. These designers in turn generate output parameters in the form of sketches or as numeric values. As the design progresses, these parameters change to reflect new information available to the designer. Throughout this paper we will use the concept of parameter trails to refer to the route that a parameter takes through the design process and the data that influences the parameter value. Such information about parameters is vital for communications.

In most cases designers try to meet their task requirements as accurately as possible and expect others to do the same. However, they often don't know where the parameter values given to them have originated from, so they can not question them easily and in turn they do not know how the information they generate is used. This hinders negotiation between designs and leads to a waste of resources. A parameter value is often used by many different

designers who may not be aware of each other. If one of them needs to change a value, others are often not informed about it. This can cause unnecessary design effort and costly changes.

By making parameter flow explicit throughout a design process, designers can follow information throughout the process and elicit rationales for values. Design teams can negotiate about task values and other users can be proactively informed about changes. This paper describes a method, based on current models for project planning and design connectivity, which supports communications using parameter trails. In small teams communication usually works well, because designers interact informally. Our research aims to facilitate informal communication across a wider organisation, by allowing people to locate each other and find out about each other's context and tasks.

2 Our studies

During recent years, the authors have carried out four major industrial case studies, each involving approximately twenty one-hour interviews with designers and design managers. These interviews were complemented by observations within the companies and feedback from the participants on issues identified during analysis of interview recordings.

The central focus for two of these studies was communications and project planning. In the consultancy arm of a large international engineering firm, distant management and strong personal animosities amongst partly embittered team members has let to surprisingly bad communication in a small collocated organisation. In a new design project undertaken by a medium sized car manufacturer and based in a new design office [1], we observed problems arising from very formal communication paths in an organisation where individuals did not know each other from past projects. We also noted that a convoluted structure of meetings had been introduced from quality control personnel.

The remaining studies were carried out with a helicopter manufacturer and a diesel engine manufacturer [2]. Both focussed on changes to existing products as well as incremental new product development. Although the main focus was on change, these studies also demonstrated the importance of information flow and contextual understanding in complex product design, confirming results from the other studies and showing the generality of the problem.

3 Communication Problems

When building the first aeroplane, the Wright bothers had a complete overview of the entire product. Further, their design team consisted of only two people, both with similar experience and based in the same location [3]. Modern design processes for complex products such as aircraft share little commonality with that of the first aeroplane; thousands of designers, dispersed across several countries work independently or in small teams to produce a single product. These designers have a localised knowledge of the product and the process, but have very little understanding of other aspects, even on a very high level. Figure 1 shows clusters of overview for a helicopter project and illustrates that nobody retains a complete overview of the product.

Understanding complexity and supporting information flow throughout the project is a key to product success. Although human factors play a major role in communications, information flow problems also exist within companies where personality conflicts are minimal.

Product hierarchy   **Company hierarchy**
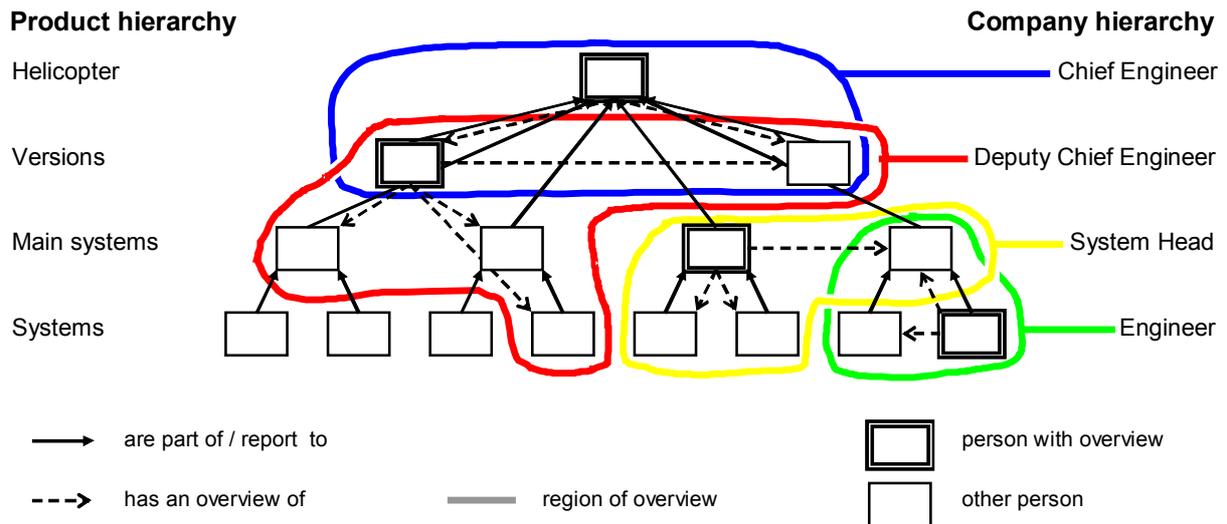
Figure 1. Overview in complex products

Much communication is both hierarchical and formal [4]; informal communications, which overcome some of these concerns, are often complicated by hierarchical company structures and discouraged by management. Designers who don't know people in other design groups have to go up through the hierarchy and down again through very few people at the top (Figure 2).
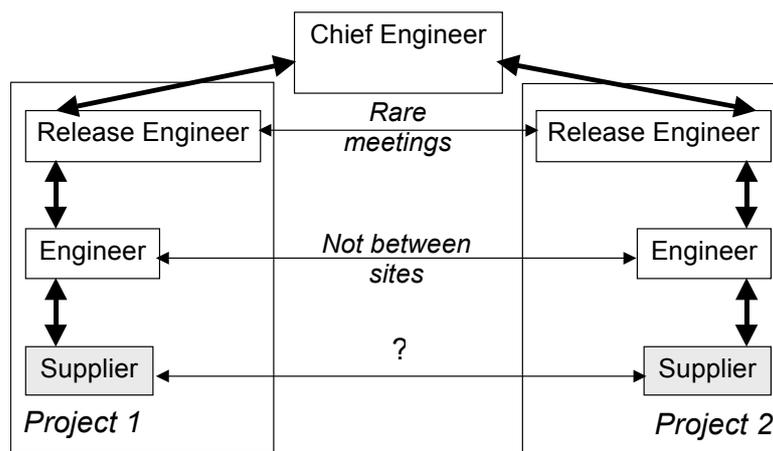
Figure 2. Hierarchical communication across projects

Past experience relating to solving similar problems in the organisation isn't accessed and mistakes are repeated. Our work focuses on identifying different communications problems, determining their causes and developing suitable solutions. Many designers fail to see their role in the context of the overall design (Figure 1). Complex products are decomposed into modules with relatively simple interactions, to minimise the complexity of design process management. However, lack of awareness of component interactions and design processes interdependency results in a number of problems: designers don't know what information they need to provide at which time; nor what information they need to request [4]. In particular we encountered the following issues:

1. **Incomplete information history.** Team members can't trace information, such as specifications and parameter values, back to the designers who are responsible for them. Hence, they cannot question these values or change pervious decisions although such changes would improve the overall product.

2. **<u>Concurrent information applications are unknown.</u>** Designers don't know who else is using the same information simultaneously or what consequences a change would have on other design aspects. Obtaining information concerning concurrent parameter use is especially difficult across organisational barriers.

3. **<u>Unknown destination for information.</u>** In many cases, designers don't know who else will be influenced by their decisions. The destination and application of information they create or influence is unknown. In consequence, designers often don't provide their colleagues with all the information they need, especially data about decisions that are provisional, or which parameter boundaries can be changed [see 5]. This results in arbitrary decisions which impose unnecessary restrictions later in the project.

The consequences of misunderstanding the information flow in design are severe. Important tasks are not given appropriate priority, resulting in unnecessary delays for others. This is especially true of small, seemingly insignificant tasks, such as ordering a component, which can have a huge impact on the project if they are not done in time. Often, designers don't get sufficient feedback on how their information is used by colleagues. Hence, they fail to identify opportunities to improve their own task performance or to develop communications with their colleagues. In addition, they are often unaware of the status of information they receive and have no way of distinguishing final values from rough estimates. Designers mistakenly assume that placeholder values [6, 7] are exact requirements and put great efforts into meeting these targets causing unnecessary delays and wasted resources. Similarly, exact values may be mistaken for placeholders resulting in a poor quality product or rework [7, 8]. These problems are especially relevant for contractors and suppliers, who are excluded from the decision making process because of confidentiality concerns. Information is consciously withheld from such parties due to company policies on data secrecy. In summary, several information context problems exist in complex products and people are lost in the design process.

Some support is available from design process models but fully appropriate models and tools at a parameter level are lacking. The development of such tools from current models of design processes is a core interest of the authors.

4 Models of Design Processes and Connectivity

Traditionally, much design research has concentrated on the development of high level generic models. While these can provide useful insights for teaching, they are often of little practical use besides providing checklists for targets during in the design process. The alternative is to use model-specific design process analysis techniques such as signposting models [9, 10, 11] or design structure matrices [12, 13].

Several design process models exist. Pahl and Beitz [14] and Dym [15] presented a staged model of the design process in terms of idea generation, conceptual design, embodiment and detailed design. Cross [16] gives an overview of this model which provides useful guidance for the development of milestones, but does not address specific design activities or product properties. An indication of necessary task sequences can be obtained from activity models (Shigley and Mischke [17] and Blessing [18], but these models don't describe a specific process. Bichlmaier [19] added the concept of generic building blocks to represent typical activities of the design and manufacturing processes. Hand-over between blocks is represented by deliverable documents. These models emphasise the links between design, manufacturing and assembly, but are of limited use due to their high level of abstraction.

The U.S. Air Force program for Integrated Computer Aided Manufacturing (ICAM) [20] resulted in the development of the Integration Definition for Function Modelling technique (IDEF0) [21] which is based on the Structured Analysis and Design Technique. IDEF0 includes both a comprehensive methodology for developing models and a definition of a graphical modelling language. It can be used to construct structured representations of functions, activities or processes within the modelled system or subject area. Currently, IDEF0 supports modelling efforts for a wide range of applications. Limitations concern maintainability issues and artificial restrictions imposed by the model structure, especially the high level of detail on which IDEF models are built. Typical IDEF models have documents as input and not individual parameters; hence, confidence levels for parameters are not available.

Design Structure Matrices (DSMs) consists of a square matrix with identical rows and columns and uses an off-diagonal entry to signify the dependency of one element on another (Figure 3). These matrices show properties of processes and can be reordered to achieve minimum iteration. DSMs have been successfully used in numerous areas by both academics and industry [22]. However, DSMs do not hold information how the tasks are connected and hence cannot be used to generate parameter trails.

4.1 Signposting

Signposting is a dynamic design process [9] model based on the task connectivity due to parameters. Output parameters from one task are used as inputs to another. The state of a parameter is indicated in terms of the subjective confidence that the designer has in its refinement, indicating a level of maturity of the parameter. A task order is implicit in the confidence values and the affect that one task has on another is determined by confidence mapping. New tasks can be added to the set of tasks at any time without interfering with the rest of the model and iteration can be modelled to a limited extent by re-running a task with different confidence values.

Initially the signposting approach was used to guide designers to the next task, by showing designers those tasks for which they had sufficient input data. Currently, the technique also supports optimum task ordering by selecting the most appropriate option from a list of available tasks [10, 11]. Figure 3 shows the routes through a design process, generated using the signposting model. But such information on task connectivity is not always sufficient to enable effective communications. Designers may also wish to query models further, to ascertain information at a parameter level. The next section describes how parameter level information can be extracted from the signposting model.
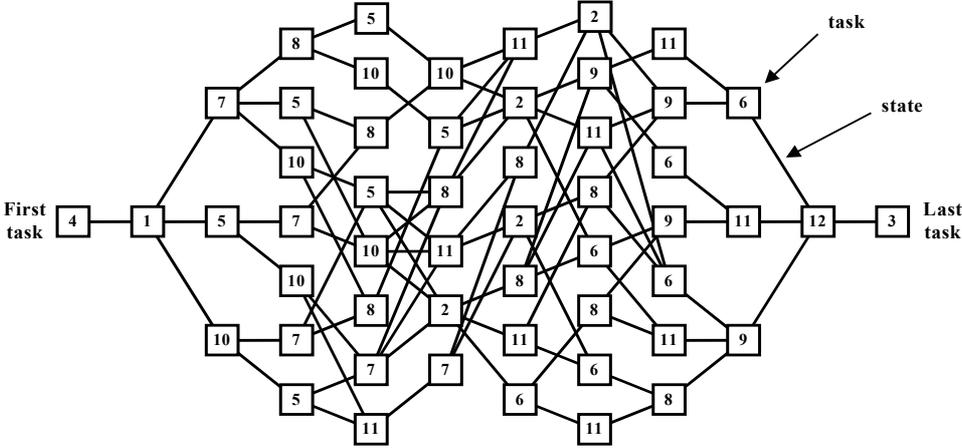

Figure 3. Task Sequence diagram

5 Parameter trails from Signposting Models

Signposting models described above are very rich in data. In contrast to the other models, they contain sufficient information to represent information flow in the form of parameter trails. This gives designers a better overall picture of the design process and the trade-offs that influence parameter values and allows them to question information they receive at any stage in the design process. They can thus find out about the status of a parameter; currently such information is difficult to obtain. In this section, different types of parameter trails are presented. The confidence level associated with the parameter is shown using colours. This allows designers to easily identify the tasks that have the greatest influence on parameter confidence.

5.1 Parameter trails for a single parameter

Understanding parameter history is critical to appreciating the significance of a specific task in the context of the overall design. It contains the answers to several key questions such as: where does this value come from, who changed it and why, where can more information about this parameter be found, what are the tolerance margins and how accurate is the value? These questions can be answered by looking at the section of a parameter trail which relates to completed tasks (Figure 4). In addition, it allows designer to see what tasks the parameter has gone through en route to the current state and to determine the maturity of the parameter in terms of confidence. For example, the designer might need to know the reason for the precise location of the fuel pipe in a diesel engine and will then want to know where the value was originally set and which tasks have changed it.
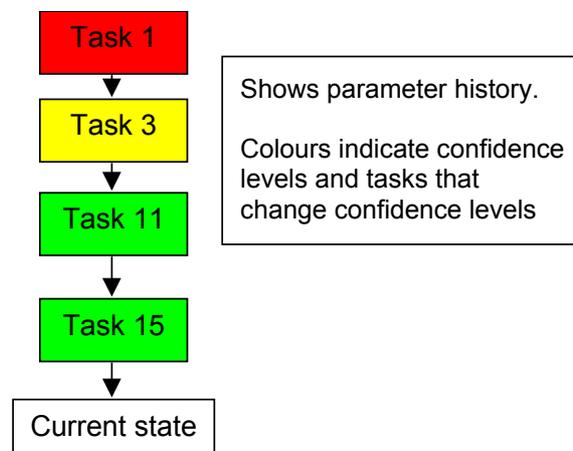

Figure 4. Bloodline - Parameter trail history

If the parameter value needs to be changed, the designer ought to know which other tasks are currently using this value; otherwise these tasks may have to be reworked. Parallel tasks which use the same parameter can be identified using parameter trails (Figure 5). This allows a designer to determine who else is using a given parameter. As well as concurrent tasks, the parameter trail shows other tasks that have already been performed and have influenced the parameter of interest en route to a parallel task. (e.g. task 7 has influenced the parameter en route to task 25 which is being carried out in parallel to the current task). For example in the design of an engine, a parameter indicating maximum fuel pressure, will be used in the design or selection of the fuel pump, the selection of the pipes and the design of the engine block and pistons.
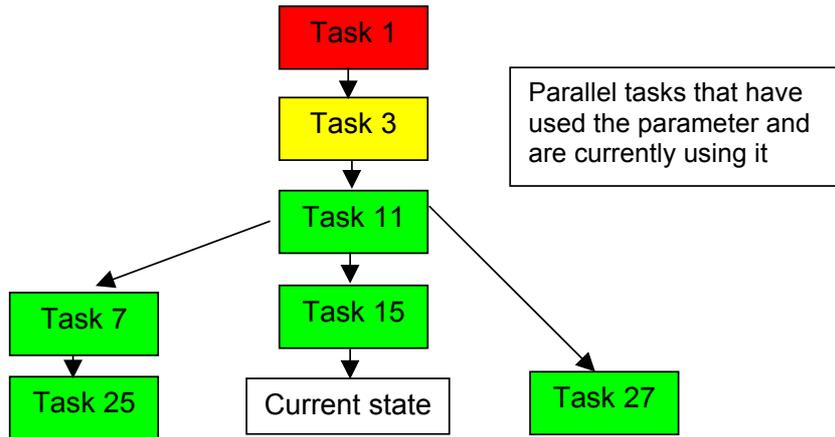
Figure 5. Relations – Concurrent task parameter trail

Designers must also consider where a parameter goes when they are finished with their task and who else will be affected by an envisaged change. Parameter trails for future tasks can be created assuming that a route through the design process has been chosen. This could be done using techniques such as signposting. Once a route through the given task model has been selected, you can identify future tasks that will use the parameter of interest. This section of the parameter trail (Figure 6) can be displayed through a tree or a list of tasks in a similar manner to past or current tasks. If the order of future tasks is unknown a list of future tasks that might use the parameter can be drawn up, but the manner in which these tasks interact, as well as temporal data such as completion dates, will be unavailable. For example, if issues arise relating to the supply of certain components, advanced warning of how other parts of the design may be affected, is extremely helpful. This is also true for planning, testing and manufacturing changes.
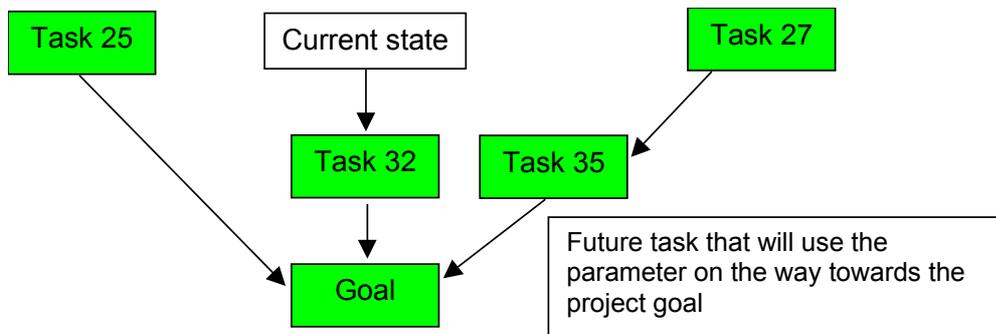

Figure 6. Offspring - Future parameter trail

5.2 Multi-parameter trails

Figures 4 to 6 have illustrated the parameter trail for a single parameter throughout the design process. But what if the designer wants to know how other parameters have influenced the value under consideration? For example a designer who wishes to change the diameter of a bolt would need to know about associated stresses and diameters through which the bolt must fit. In addition to trails relating to a single parameter, multi-parameter trails, describing parameter interaction, can also be generated (Figure 7). Such parameter family trees identify the relationship between different parameters required to perform a given task. They enable

the designer to identify other parameters required as inputs for a task where this parameter was also needed. In turn, the designer can view a similar tree for any of these parameters that he considers relevant. Multi-parameter trails allow the designer to obtain a clear overview of any given parameter in the context of the overall design. For example, if a designer is provided with a load figure for a stress analysis, it is useful to be able to trace how the load value has been calculated.
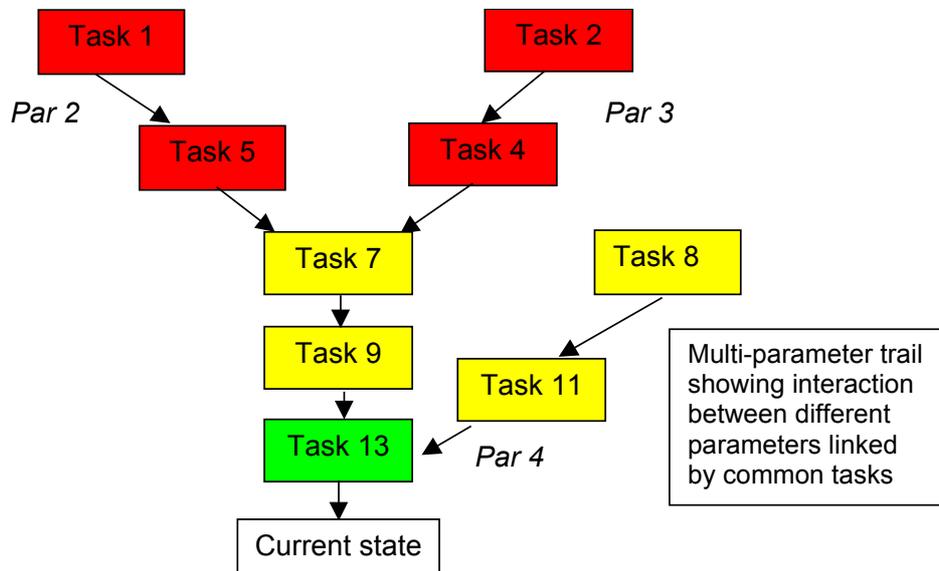

Figure 7. Ancestors – Multi-parameter family-tree

6 Future Work

Parameter trails could be automatically generated from signposting models using computer support. Such computer tools offer the potential for improved communications by enhancing clarity about the constraints and targets that should guide designing. They should be quick, simple to use and transparent (in the sense that designers can use them while thinking about the design, not the tool or the representation).

Benefits could be obtained from increasing the amount of information contained in the parameter trails. Parameter values as such don't carry information about the range of values associated with a parameter, i.e. margins, commitment, value rationale and value maturity. Such information could be included using annotations as suggested in Stacey and Eckert, in press [6]. This would increase the range of applicability of the model.

New features could be included to allow designers to query a complex model and identify important subsets of the parameter trail. For example, a designer may wish to look at all of the tasks where a parameter value changed. People-parameter trails could also be developed showing the key people involved with a given parameter. Currently, designers are often forced to consult managers in order to determine who else is involved in the design process; such questions could be answered using parameter trails. This would improve the efficiency of the design process by involving the right people at the right time and ensuring that managers and other staff are not needlessly disturbed.

7 Conclusions

Communication is complex and resulting problems are a major concern in industry. This is especially true in large organisations working across multiple sites. Many designers do not know how their own tasks fit into the overall product design and current process models to not provide the designer with sufficient information at a parameter level.

This paper introduces the concept of parameter trails which can be derived easily from design process models and cross-checked with design product models. Current design process models do not facilitate the tracking of inter-parameter influences. Parameter trails show the entire live cycle of a parameter from its generation to future use, thus allowing designers to trace the parameter through the entire design process. As a result, designers can question or annotate information and negotiate with future users.

A parameter family tree enables designers to check how a parameter has been derived, so that they know how to question the validity of a value. Our model allows designers to follow up information and engage in negotiations with the appropriate members of staff, ensuring that others are not unnecessarily bothered. The model supports communication by providing parameter-specific information.

Future work focuses on providing computer support and increasing the amount of information contained in the parameter trail by using annotations.

References

[1] Eckert C.M. and Clarkson P.J. "Connectivity as a key to supporting design", <u>Artificial Intelligence in Design</u>, Cambridge, UK, 2002, pp.479-501.

[2] Clarkson P.J., Simons C. & Eckert C.M., "Predicting change propagation in complex design", <u>Journal of Mechanical Design</u>, in press

[3] Gates B., "The Wright Brothers", <u>Time 100</u>, www.time.com/time/time100/-scientist/profile/wright.html, 2002.

[4] Eckert C.M., Clarkson, P.J. and Stacey, M., "Information flow in engineering companies: problems and their causes", <u>Proceedings of ICED '01</u>, Glasgow, UK, 2001.

[5] Stacey M.K. and Eckert C.M., "Managing uncertainty in design communication", <u>Proceedings of ICED'01</u>, Glasgow, UK, 2001.

[6] Stacey M.K. and Eckert C.M., "Against ambiguity", <u>Computer Supported Cooperative Work</u>, in press.

[7] Stacey M.K., Eckert, C.M. and McFadzean, J., "Sketch interpretation in design communication", <u>Proceedings of ICED'99</u>, Technical University of Munich, Munich, Germany, Vol. 2, pp.923-928, 1999.

[8] Eckert C.M., "The communication bottleneck in knitwear design: analysis and computing solutions", <u>Computer Supported Cooperative Work</u>, 2001.

[9] Clarkson P.J. and Hamilton J.R., "Signposting: a parameter-driven task-based model of the design process," <u>Research in Engineering Design</u>, 2000, pp18-38.

[10] Melo A.F. and Clarkson P.J., "Design process planning using a state-action model," Proceedings of ICED'01, Glasgow, UK, 2001.

[11] Jarrett J.P. and Clarkson P.J. "Improving the product by accelerating the process," Proceedings of ICED'01, Glasgow, UK, 2001.

[12] Steward D., "The design structure system: a method for managing the design of complex systems" IEEE Transactions on Engineering Management, Vol. EM-28, No. 3, 1981, pp.71–74.

[13] Eppinger S.D., Whitney D.E., Smith R.P. and Gebala D.A. "A model-based method for organizing tasks in product development", Research in Engineering Design, 1994, pp.1-13.

[14] Pahl G. and Beitz, W., Engineering Design, 2nd ed., trans. Wallace K.M., L.T.M. Blessing and F. Bauert, Springer, London, 1996.

[15] Dym C.L., "Engineering Design: A Synthesis of Views", Cambridge University Press, Cambridge, UK, 1994.

[16] Cross N., "Engineering Design Methods", John Wiley and Sons, Chichester, 1989.

[17] Shigley J.E. and Mischke C.R., "Mechanical Engineering Design", McGraw-Hill, 1989.

[18] Blessing L.T.M., "A Process-Based Approach to Computer Supported Engineering Design", PhD thesis, University of Twente, Netherlands, 1994.

[19] Bichlmaier C., "Methoden zur flexiblen Gestaltung von integrierten Entwicklungs-prozessen. München", Dissertation, TU München, 2000.

[20] Softech, Inc., "Integrated Computer-Aided Manufacturing (ICAM) Architecture Part II: Function Modelling Manual", Wright-Paterson AFB, Ohio, 1981.

[21] Marca D.A. and McGowan C.L. "IDEF0 - SADT Business Process and Enterprise Modelling", EMcGraw-Hill, 1993.

[22] Browning, T.R., "Applying the Design Structure Matrix system to decomposition and integration problems: A review and new directions", IEEE Transactions on Engineering Management. Vol. 48, pp. 292-306, 2001.

Tomas Flanagan
Engineering Design Centre
University of Cambridge
Trumpington Street
Cambridge CB2 1PZ
United Kingdom
Tel. Int +44 1223 332 673
Fax. Int +44 1223 766 963
E-mail: tlf21@eng.cam.ac.uk
URL: http://www-edc.eng.cam.ac.uk/people/tlf21.html