INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED11
15 - 18 AUGUST 2011, TECHNICAL UNIVERSITY OF DENMARK

THE EVOLUTION OF INFORMATION WHILE BUILDING CROSS-DOMAIN MODELS OF A DESIGN: A VIDEO EXPERIMENT

Naveed Ahmad, David C. Wynn and P. John Clarkson
Engineering Design Centre, University of Cambridge

ABSTRACT
Models capturing the connectivity between different domains of a design, e.g. between components and functions, can provide a tool for tracing and analysing aspects of that design. In this paper, video experiments are used to explore the role of cross-domain modelling in building up information about a design. The experiments highlight that cross-domain modelling can be a useful tool to create and structure design information. Findings suggest that consideration of multiple domains encourages discussion during modelling, helps identify design aspects that might otherwise be overlooked, and can help promote consideration of alternative design options.

Keywords: Cross-domain models, design process, information structure, video experiment

1 INTRODUCTION

There has been a recent increase in the research literature reporting applications of cross-domain models to support different aspects of product development, such as analysing change propagation in the design process [10]. However, building cross-domain models, especially including information from the design process, can be difficult because it requires consideration of different sorts of information from different domains. To our knowledge, no study has been reported which analyses how design information is built up while constructing these cross-domain models.

Testing a design modelling method in a true industrial setting can be difficult for various reasons, including: all the necessary information is not available in one place; no one person knows everything about a product; and experts are often specialised in a certain area. This paper presents two video-recorded experiments conducted to explore the role of cross-domain modelling in building design information. The experiments study cross-domain modelling as a framework for information acquisition, and give some insights into a modeller's thinking while building the model. Moreover, the experiments also highlight some of the ways in which building a cross-domain model can assist the creation and evolution of design information.

Through the video experiments this paper considers the following questions:

- *What role can cross-domain modelling play in creation and evolution of design information?*
- *(How) does the information captured in one layer (domain) of a cross-domain model help in building/modelling design information in the connected layer(s)?*
- *(How) can cross-domain modelling help modellers improve their understanding of a design?*

The paper is structured as follows. Section 2 explains the methodology for the experiments, which are based on observing two groups of research students building cross-domain models using a particular modelling notation. Section 3 describes the notation which was used and explains the rationale for each of the domains included. Section 4 discusses how the participants approached the modelling exercise in each of the two experiments. Section 5 analyses the experiment results in greater detail. Section 6 summarizes the main findings and draws conclusions.

2 METHODOLOGY

Two semi-controlled experiments were conducted to observe design research students disassembling an everyday consumer product and constructing a cross-domain model of that design using a given modelling framework and software tool. The experiments were recorded using a video camera and the resulting footage was analysed.

Similar methodology has been used previously by a number of different researchers (e.g., [6, 7]). One of the most highly-cited examples discusses a situation in which a group of researchers were invited to observe a video of designers working on a product [6]. This proved helpful in understanding different design issues and in considering the interaction between the designer and the product. It also helped in understanding how designers work on different aspects of the product and how design learning can be increased. We aimed to take a similar approach to understand issues related to building cross-domain models of a design.

2.1 Experiment setup

Because the objectives of the experiment stated in Section 1 are quite open-ended, it was decided to organise the experiment to be similarly open-ended and to capture as much information about the modelling process as possible. The data could then be reviewed to search for relevant insights. The experiment was thus set up as follows:

- **Basic experiment setup and method of recording information**: Modellers were grouped into pairs, so that their discussions might make their thought processes more explicit. Each group was given a (physical) product and asked to perform the decomposition and modelling task explained below. The experiments were video-recorded for later analysis. The video captured the disassembly of the device, the reasoning about its parts and functionality, and the interaction of the modellers with each other and with the emerging model. Review of these recordings allowed the questions posed in Section 1 to be considered.
- **Selecting a product to model**: It was necessary to select a product which could be disassembled, understood and modelled on a reasonable level of detail in a limited time. The product also needed to be complex enough to justify modelling the interactions between the different domains of design information, such as functions, components, etc. A vacuum cleaner was selected to fulfil these criteria. Within the authors' research group, several researchers had already analysed this type of device for different research projects. A group of experiment participants with similar levels of understanding of the product's functionality was thus readily available. The specific vacuum cleaner which was used.
- **Selecting the experiment participants:** Two groups each comprising two researchers familiar with vacuum cleaner designs participated in the analysis. The two groups worked independently to make a cross domain model of the vacuum cleaner.
- **Describing the modelling task to the participants:** Both the groups were given a set of requirements for the vacuum cleaner, divided into four categories: general requirements, functional requirements, performance requirements and usability requirements. They were also provided with the vacuum cleaner itself. The participants were asked to take the product apart and construct a cross-domain model to represent four different "layers" of information concerning its design: a requirement layer (as mentioned some requirements were already provided, although the modellers could choose to change, supplement or arrange them differently), a function layer, a component layer and the "design process" layer. For each of these layers, the participants were asked to identify the elements in the given domain (for instance, the components comprising the device) as well as the connections or flows between them. They were also asked to create connections between elements modelled on different layers, for instance to capture the mapping between the device's functions and its components. The definitions of elements and relationships which the participants were asked to model, as well as the rationale for including them in the experiment, are described in Section 3.
- **Setting a time limit for the modelling:** Each group was allotted up to five hours to analyse the product and complete the cross-domain model. Setting a time limit was important for two reasons: 1) the participants needed to find time to work on the experiments; and 2) the time limit helped the groups to decide on the level of detail of the analysis. This was important since, without some external limit on modelling scope, different modellers might choose to represent the same design at very different levels of detail. The time limit and similar levels of modelling detail ultimately facilitated comparison between the results of the two groups.

3 THE CROSS-DOMAIN MODELLING APPROACH

The modelling notation used for the experiments is called the Information Structure Framework (ISF). Full details of the ISF are explained in detail in our related article [1]. In this section, we briefly describe the approach and the rationale for using it in the experiments.

The ISF approach provides a notation for structuring information that describes four domains of a given design: 1) requirements, 2) functions, 3) components and 4) the detailed design process. Figure 1 shows the interconnections between domains in the approach, and Figure 2 shows an example model of a product. For the experiment, the participants were provided with a specialised software tool for constructing such models, based on the Cambridge Advanced Modeller [11]. This tool allowed the participants using interchangeable diagram and Design Structure Matrix (DSM) views.

3.1 Rationale for the domains and relationships included in the approach

The original purpose of the ISF approach was to help modellers structure the information required to assess the impact of a design change. A tool was developed allowing an ISF model of a given product to be used to trace from the point of change initiation through the layers and relationships [1]. Ultimately, this tool helps identify the set of design tasks which need revisiting to complete a given change. With this purpose in mind, the set of domains and relationships included in the framework were determined from a review of the literature on how change can propagate. In particular:

- **Rationale for including a requirement domain.** According to a survey conducted by Deubzer *et al*., (2006), requirement changes constitute 44% of all change requests [2]. Furthermore, requirement traceability systems are well-established in many organisations and they are used to track changes initiated by a change in requirements. Including a requirements domain in the modelling framework allows a change in requirements to be traced through to the functions and components of the product.
- **Rationale for including a function domain**. Functions have been used extensively along with the components/subsystems at the conceptual design stage, mostly to generate different concepts [3,4]. Following Pahl and Beitz' [5] argument that building a function structure as part of the design process allows a clear definition of subsystems of the product, including a function domain in the cross-domain modelling framework was intended to help build understanding of the relationships between different components/subsystems of the product. In terms of change propagation, the function domain provides a means for tracing changes in requirements through to the components that implement them.
- **Rationale for including a component domain**. The interactions of component/subsystems in a product have been used extensively by many authors considering change propagation in design [10]. The component domain also provides a relatively concrete grounding for a design model, since a product can be disassembled to identify its constituent parts.
- **Rationale for including a process domain.** The ability to trace a change through to the tasks required to implement it can help develop realistic estimates of the effort required to implement that change. In the Information Structure Framework, the detailed design process domain consists of a network of design activities linked by the parameters that they consider and affect. This network can be used to identify the tasks that need rework, because the rework caused in a task due to changes in its input parameter(s) might propagate through the process flow network to require rework of downstream tasks.

The modelling framework takes a layered approach to structuring the information required to manage a change process. Each of the four layers provides a data repository for information describing the corresponding domain. The layers also contain links within themselves and to other layers, as shown in Table 1 and Figure 1.

Table 1: Interactions within and across domains

Within domains	Across domains
Not allowed	Requirements *<map to>* functions
Functions *<interact with>* functions	Functions *<map to>* components
Components *<connect to>* components	Components *<are described by>* parameters
Parameters *<are affected by>* design activities, and vice-versa (directional information flows)	*Not allowed*

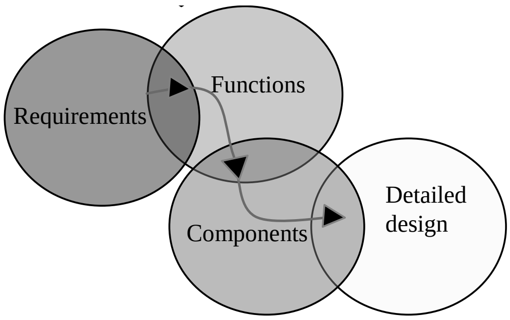

Figure 1: Connections across domains that are allowed in the modelling framework

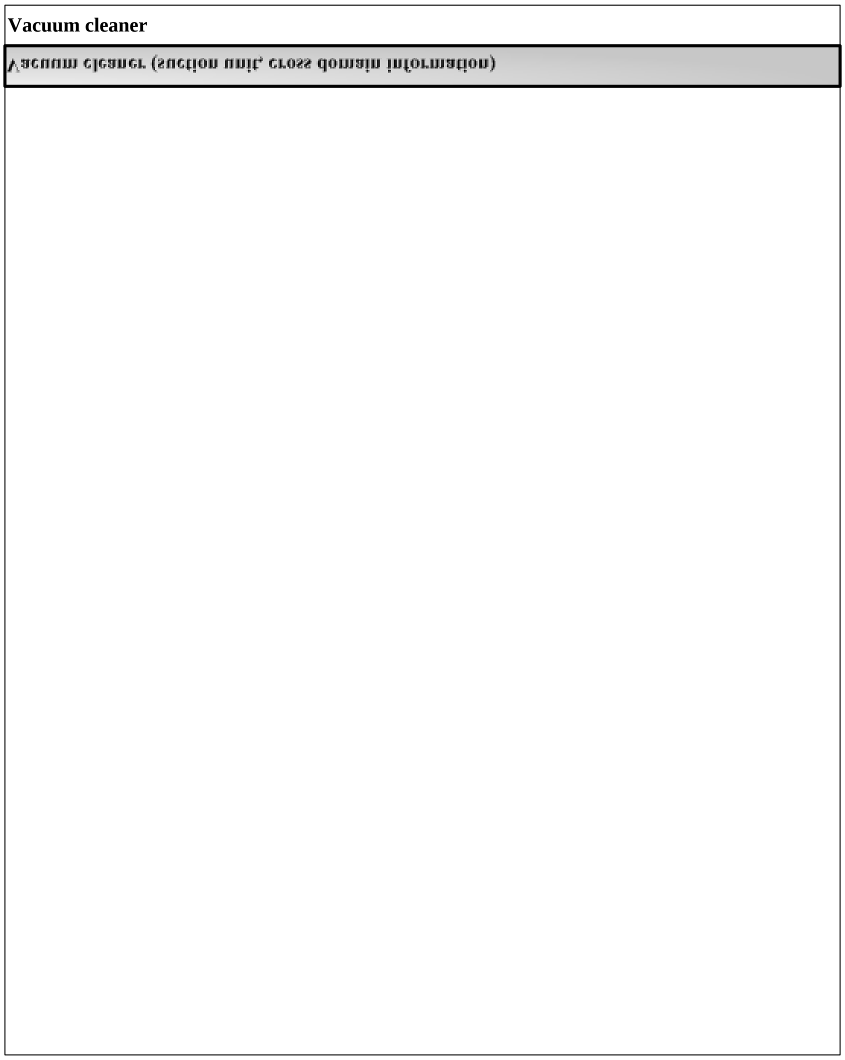

Figure 2: An example of the modelling framework applied to represent requirements, functions, components and a possible design process for a vacuum cleaner suction unit

4 DESCRIPTION OF THE MODELLING PROCESS IN EACH EXPERIMENT

Group 1 completed the modelling experiment in 4 hours and 35 minutes, whereas group 2 completed the exercise in 4 hours and 5 minutes. The modelling processes undertaken by each of the two groups who performed the experiment were studied by analysing the video recordings, and are described below.

Figure 3: Frames from the video recordings, showing Group 1 (left) and Group 2 (right)

4.1 Group 1

Group 1 spent almost 40 minutes at the start of the experiment discussing the product requirements and arranging them into different categories. During the first 30 minutes of that time, there was minimal interaction between the two team-mates as they studied the product description on their own. For the next ten minutes they discussed various requirement-related issues including: use of standard components; manufacturability; the need to cut the cost down; and aesthetic requirements.

The participants in Group 1 started work on the function structure of the product 40 minutes into the experiment. They used the requirements as the basis for constructing the function layer of their model, following a process similar to Pahl and Beitz's systematic design process's *original design* [5], with which both participants were familiar. There was a lot of confusion regarding how to organise the function structure. This led to different solutions being proposed to control the flow power of the device:

- "*…control flow power either by dissipating energy, where a mass can be put in the dust containment chamber*", or…
- "*…control flow power by reducing energy, where there can be a hole in the pipe which can help in reducing the energy*", or…
- "*…control flow power by sending an electrical signal.*"

The modellers took almost 55 minutes to complete the function structure model.

A hierarchical approach was taken when modelling the component layer. The participants divided the vacuum cleaner into main systems and then further decomposed these systems into subsystems/components. They noted that:

- "*…functions are mapped to subsystems and subsystems are connected to each other as well, thus allowing tracing changes in requirements to components.*"

While decomposing the product, the modellers also referred to the information in the requirement and the function layer. This suggested one advantage of using a cross-domain modelling approach, namely that the information in one domain helps to build up that in related domains. In total, it took 35 minutes to break down the product into different subsystems/components and connect them together in the model.

The participants started work on the detailed design process layer 2 hours and 10 minutes into the experiment. This was modelled as an iterative process in which each part would be separately designed. Five main groups of tasks were identified:

- **Air system design tasks**: These focused on design to handle power requirements. The model of the vacuum cleaner included an input (electrical) power and an output power in the form of suction. Comparing the input and the desired output power, tasks to calculate the size of the impeller, motor and rough estimates of flow losses were considered. Tasks for calculating losses were modelled, to consider losses in the transformations from electrical energy to torque, torque to air flow and all the air flow losses. This was part of the first iteration in the detailed design. The modellers then directed the calculated output power (a parameter) through an "aerodynamic design" workflow to determine the flow and size of the bag, pipe and filter.
- **Design for manufacture**: This was modelled as a sub-process which applies to all the subsystems in the components layer.
- **Ergonomics design**: These tasks included consideration of design issues such as: centre of mass

of the vacuum cleaner; the best place for the handle; and the strength and hardness of the wheels.

- **Overall performance testing**: The modellers represented performance testing as an iterative loop undertaken for each component. The rationale here was that each component needs to be tested in different ways, and if problems were revealed the design would need to be reconsidered. They discussed an example of strength testing of the wheels, which should resist a fall from a certain height. They considered that failure of this test would lead to iteration of the wheel design.
- **Iterations to cut cost and complexity**: Other iterations included in the detailed design process layer included looking for ways to reduce cost of manufacture or to reduce the complexity of each subsystem.

Creating the activities and the design parameters in the detailed design process layer took the modellers 1 hour and 35 minutes. Finally, cross-domain connections between the components/subsystems and the design parameters were made. This final step was completed in 20 minutes.

4.2 Group 2

The second group spent 20 minutes at the start of the session to understand the product requirements given to them. The group acknowledged that:

- *"It is a realistic set of requirements that may be given to a designer / modeller"*

One issue discussed by the modellers while studying the product description and looking at the vacuum cleaner was:

- *"…which parts to acquire from suppliers?"*

The modellers chose to assume that the suction unit is sourced from a supplier. They spent 40 minutes to model the function structure of the product. Similarly to the first group, the requirements were used as the basis to develop a function structure. The process of developing the function structure helped in building understanding of the product. For example, the modellers realised that 'vibration and noise' should be included in the function structure as a function related to operation of a vacuum cleaner, although this was not clearly stated as one of the requirements. Without an explicit function layer in the model, this information might have been lost, even if the modellers were able to identify the issue. In other words, building the function structure appeared to help modellers in identifying and recording otherwise-implicit issues.

The modellers spent 15 more minutes to develop the cross domain links from requirements to functions. These fed back to improving the function structure. In particular, the modellers realized while attempting to connect the requirement – "Machine will be operable in Europe and USA" – that the function structure needed to be changed to accommodate it.

Moving on to the component layer, in comparison to group 1 the participants in group 2 did not adopt a top-down approach to decompose the device. Instead they identified the lowest-level subsystems directly, a task which took almost 15 minutes. After creating a list of subsystems, this group used a DSM to consider and create the different possible connections between them–instead of drawing a connectivity diagram such as that shown in Figure 2. It took only 15 minutes to finish creating connections in the component layer using this approach. The modellers then started making cross-domain links between functions and components, which took them almost 30 minutes. This highlighted additional components which had been previously overlooked: for instance, while connecting the functions *'Filter clean air from dirt' and 'contain dirt'*, they realised that they had missed a component necessary to partly implement these functions. A new component, '*bag*', was included in the component layer, which was subsequently validated again.

2 hours and 15 minutes into the exercise, the participants started modelling the detailed design process of the vacuum cleaner. They discussed that the process could be described in different ways. For instance, a tree-like structure could be used, in which for every subsystem a set of specifications should be created for the suppliers. Such a description would be similar to a Systems Engineering "Vee" model, in which overall requirements are broken down to subsystem requirements that are validated and then re-integrated. Alternatively, the same process could be described using another structure. For example the process could be divided into: 1) mechanical layout of the parts in the vacuum and the CAD modelling; 2) appearance and aesthetics of the outside and; 3) electronics. It was also mentioned that:

"[The] detailed design process also depends on whether you are designing from scratch or you are modifying an existing design".

In the end, the modellers settled on a hybrid description combining elements of their different ideas. For each subsystem design they used a tree-like organisation to describe the process, in which once each subsystem is designed it is checked for any design errors and then fixed if necessary. The overall design was guided by electronics design, mechnical design, styling and aerodynamics. In total, 1 hour and 55 minutes were spent to deliberate about and model the detailed design process. This was by far the most time the group spent in any single layer of the ISF model.

The final step was to make the cross-domain links between the component layer and the detailed design process layer. This was completed in 25 minutes.

4.3 Issues raised by the participants about the cross-domain modelling framework

A number of concerns were voiced by the modellers about the cross-domain modelling framework:

- Requirements were not allowed to be directly connected to components. However, some requirements were identified which did not easily map onto functions, but which nevertheless were thought to affect the implementation of the product. The original reason for restricting the type of connections across domains was to reduce the complexity of modelling by providing guidance for expected connections. The experiments suggested this may be too restrictive.
- The modellers did not enjoy making the function structure of the product; yet it was evident that it helped to increase the participants' understanding of the vacuum design. They also appreciated that consideration of functions helped arrive at a more complete component/subsystem model.
- Deciding on the level of detail for modelling proved difficult. As a result, when decomposing the product into subsystems, group 1 decomposed the vacuum cleaner into 27 components and group 2 decomposed it into 19 components. Finding a level of detail and suitable structure for the detailed design process layer seemed the most challenging.
- There was a lot of confusion when connecting the subsystems to the design parameters, especially considering whether there should be a link from a subsystem to all related design parameters, or just one link at the highest level where the concept of subsystem is defined.

4.4 Summary

The experiments confirmed that constructing a cross-domain model helped the participants in creating and structuring information about the vacuum cleaner design. The modelling process also led to an increased understanding of the product, and led the participants to explore different design possibilities. A number of issues were raised, especially considering how the level of detail of a model could be chosen and regarding the benefits and challenges of function modelling. Overall, the participants expressed satisfaction with the models they had created:

- *"It is a good representation of vacuum design."*
- *"It seems like a realistic design process."*

5 ANALYSIS OF THE MODELLING PROCESS IN EACH EXPERIMENT

The video recording of each design experiment was divided into 15-minute segments and analysed to study the evolution of the design information over time, the referral of modellers to different domains as the information evolved, and the interaction between team-mates during different stages of the two exercises.

5.1 Evolution of the design information over time

Figure 4 shows the evolution of each model over time for group 1 and group 2. It highlights the time spent by the modellers on each part of the cross domain model. The dashed lines indicate the times at which each successive layer was completed.

The figure shows that each group of modellers spent a different amount of time working on each layer of the model. One thing that stands out in these plots is the time spent on the detailed design process, which shows the complexity of modelling the detailed design process of a product. The vagueness of modelling the detailed design is also highlighted by the fact that both the groups came up with two different detailed design processes.

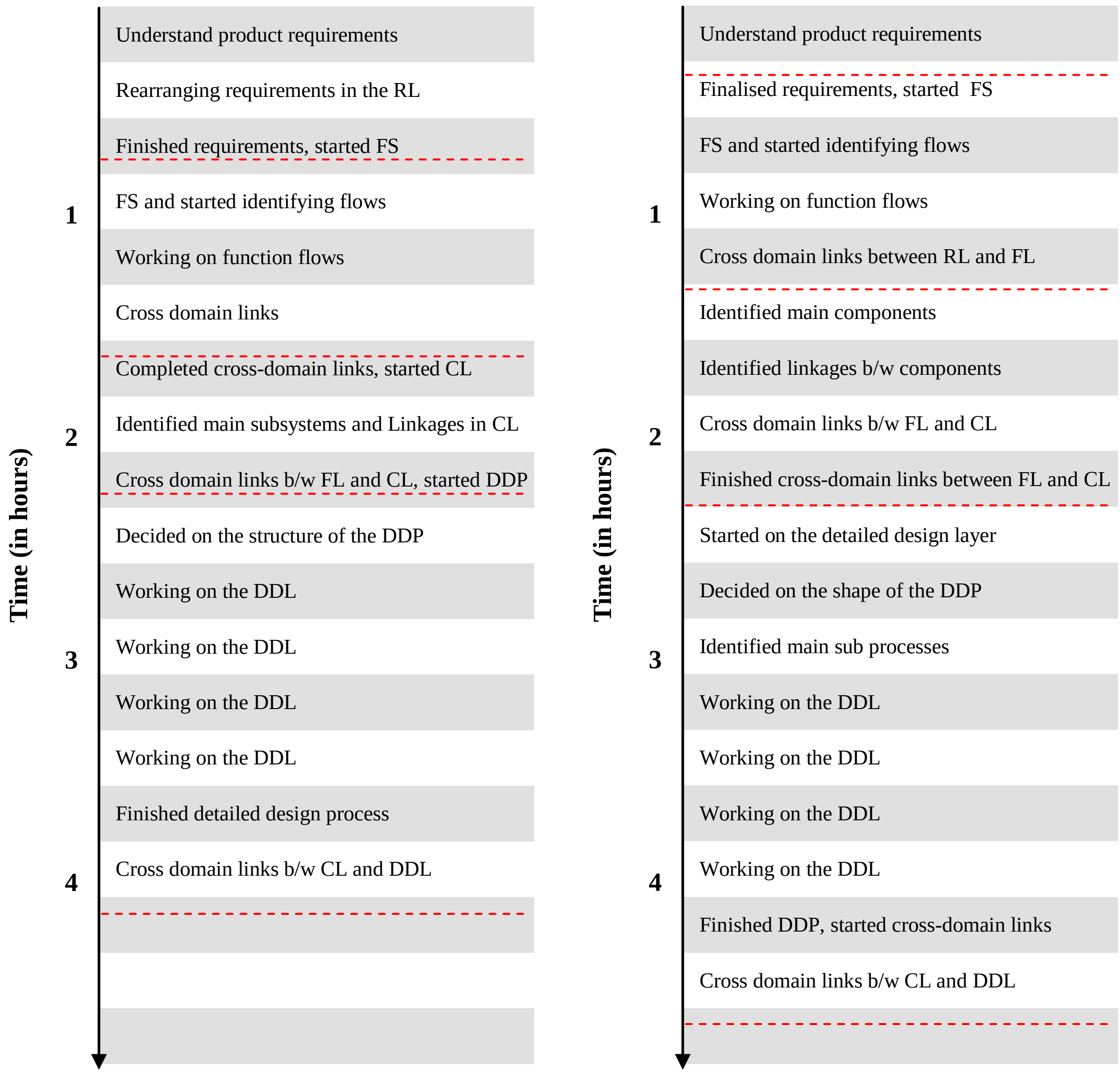

Figure 4: Plot of design process evolution against time: Group 1 (left); Group 2 (right)

5.2 Switching between different layers during modelling

Although the modellers largely progressed between the four domains in the sequence shown in Figure 1, they also referred back and forth between the layers while modelling. In some cases, viewing a layer which had earlier been completed highlighted a problem with the layer currently in focus. Figure 5 shows the number of times the modellers changed something significant as a result within each 15-minute segment of the experiments. The figure does not show how often the modellers switched between domains to compare information or make cross-domain connections. These operations were repeated numerous times by both groups during the exercises.

For instance, Figure 5 (left) shows that the requirement layer provided insights that on two occasions led to modifications in other layers (comparison to Figure 4(left) shows that on both occasions this occurred while making cross-domain connections involving the function layer). Several such incidents were discussed in Section 4.

In summary, Figures 4 and 5 clearly indicate the influence of previously-created domains as the model building progressed. This suggests that it would not have been possible to capture the same information in a single-domain model, since these observations and corrections would not have been prompted. Using a cross-domain framework such as the ISF can thus help in developing a more complete model.

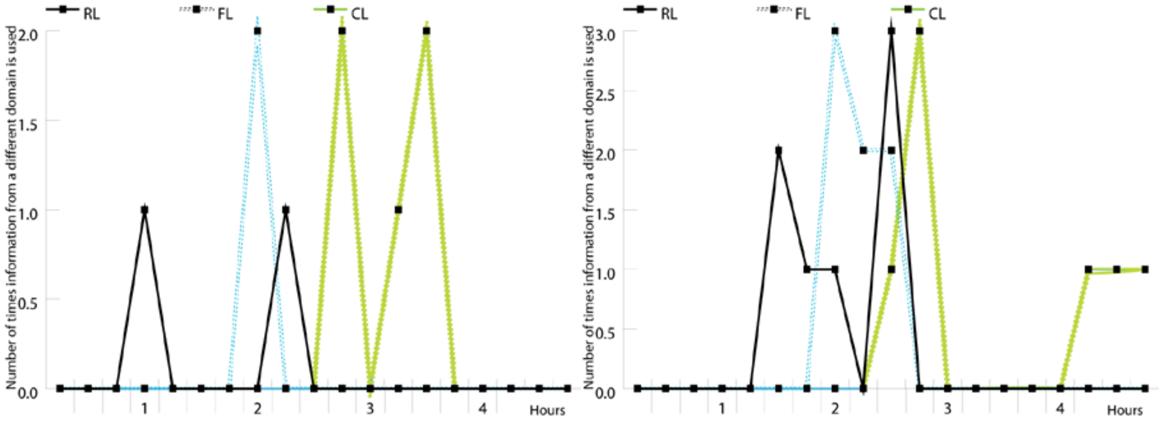

Figure 5: Using information in a domain while focusing on another domain: Group 1 (left), Group 2 (right)

5.3 Interactions between participants during each experiment

During the experiments, the modellers often discussed issues related to the emerging model. Figure 6 show the number of interactions between modellers in each 15-minute segment, for both the groups. The plots show different approaches adopted by the two groups. Group 1 spent a lot of time considering and rearranging the requirements, reflected in the large number of interactions early in the experiment (Figure 6). In comparison Group 2, spent relatively little time discussing requirements (Figure 6).

It is interesting to see that both the groups had more interactions when they started the experiment and they tend to decrease towards the end, in both cases with the exception of a sharp increase when starting the detailed design layer. At the same time, Figure 5 shows that the influence of other layers also increased. This behaviour was due to the need for discussion and correction in two activities: 1) making cross-domain links from the function layer to the component layer; and 2) deciding on the structure of the detailed design process layer. This highlights the ambiguity of modelling the detailed design process, at least in the experimental context. Once the structure of the detailed design process was agreed upon, the number of interactions decreased. In both the cases one of the modellers took ownership for completing the detailed design layer.

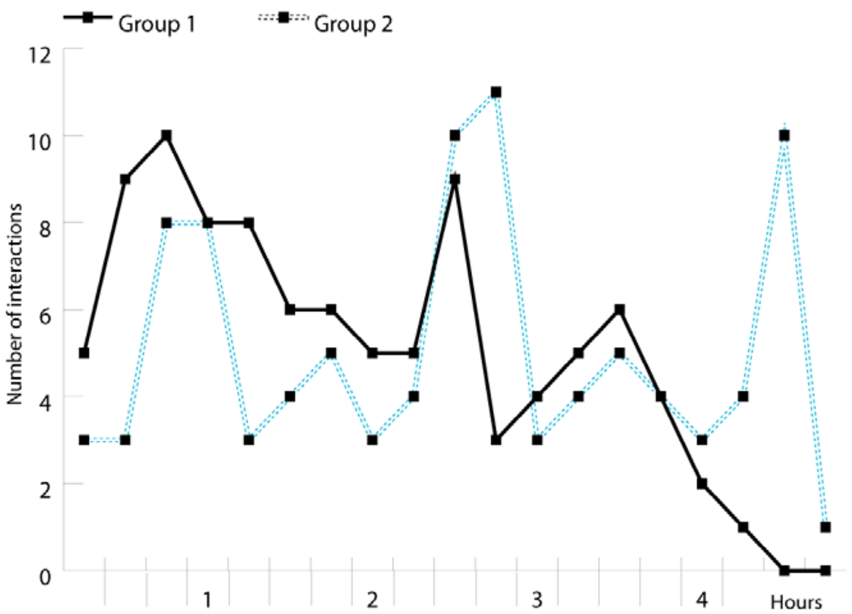

Figure 6: Number of interactions between team-mates in 15 min. intervals

5.4 Comparison of models created by the two groups

The models made by both the groups were quite similar in a number of ways, yet are difficult to compare directly due to the different assumptions and structuring approaches used in their construction. One measure of similarity is the number of elements or connections in each layer as modelled by both the groups. Table2 shows the number of elements in the function layer and components layer as modelled by group 1 and group 2. These two layers were similar in scope across the two groups. Requirements are not included in this table since the groups were provided with a list of requirements at the start of the modelling exercise, and the design process layers remained incomplete at the end of the modelling exercises and were too different to make such a comparison meaningful.

Table 2: Number of elements in each layer for both groups

Layer	Group 1	Group 2
Function layer	19	27
Component layer	17	23

6 DISCUSSION AND CONCLUSIONS

6.1 Summary of findings

To recap, the video experiments clearly showed that cross-domain modelling helps in incrementally building and refining design information. The modelling process also helps in increasing modellers' understanding of the product. Another finding of the paper is that it is possible to make quite detailed cross-domain models in a reasonable amount of time. In an industry setting, such models could be constructed once and revisited to gain insight into later versions of the product.

Most of the interactions between modellers took place when making links across different domains, which highlights the role of cross-domain modelling can play in questioning assumptions while analysing a design. The experiments showed modellers' dislike for modelling the function layer of the product, but also clearly suggested that this layer was useful in identifying issues and prompting thought. Future research could investigate the reasons behind the dislike of function modelling, and what methods could be applied or developed to facilitate modellers in constructing this layer.

One of the main findings of the literature review reported in [10] is the recent increase in the number of academic papers proposing methods that use cross-domain models to manage engineering changes. This paper contributes insights to this literature by highlighting how such models can facilitate creation of comprehensive product descriptions, which should provide a good foundation for analysis.

6.2 Advantages and limitations of the research method

Conducting an experiment in a semi-controlled environment provided control over various aspects thought to be essential for understanding the construction of cross-domain models. For instance, in our experiments the modellers were told to complete all domains of the model even if they omitted some detail from individual layers. This allowed exploration of the importance of each layer in the creation and evolution of information about the design. The semi-controlled environment also made it possible to video record the whole experiment, providing rich data for later analysis. For instance, it was possible to determine the proportion of time that modellers spent working on each domain of the models they constructed.

On the other hand, several limitations to the methodology are apparent. Modelling conducted by design researchers is likely to be different in many respects to that conducted in a real industrial setting. For instance, the researchers who participated in the experiments do not have the same experience of the product as its designers would have. The short time allowed for the experiments might have discouraged the modellers from adding extra detail to a given domain, and thus the resulting model is likely to differ from one constructed in an uncontrolled setting by the same individuals. Nevertheless, the experiments provide several insights into the modelling process. Further research could be undertaken to evaluate and possibly generalise the findings.

6.3 Main conclusions

To recap, three research questions were set out at the start of the paper. These are re-stated below and insights gained through the experiments are summarised.

- *What role can cross-domain modelling play in creation and evolution of design information?*
  The experiments showed that a cross-domain modelling framework capturing requirements, functions, components, parameters and design tasks (the Information Structure Framework) provides a useful way to create and structure design information, as well as to progressively build understanding of design issues. The experiments also highlighted some limitations of the framework that could be considered in future research, such as providing guidance to help structure the design process descriptions. Finally, the experiments showed that it is possible to make quite detailed cross-domain models of a moderately-complex product in reasonable time.
- *(How) does the information captured in one layer (domain) of a cross-domain model help in building/modelling design information in the connected layer(s)?*
  The experiments revealed many instances where modellers used information they had created in one layer to identify and resolve issues on another layer. This showed that the progressive construction of a model covering multiple domains can help create a more complete and potentially more rigorous description of a design that a single-domain model alone.
- *(How) can cross-domain modelling help modellers improve their understanding of a design?*
  The network of information that the participants developed while creating their multi-domain models provided a rich picture highlighting many design issues and prompting them to consider and discuss different design options. These findings suggest that such models are not only a way to record information, but could also provide a useful tool for analysing a design.

REFERENCES

[1] Ahmad, N., Wynn, D.C., Clarkson, P.J., Development and Evaluation of a Tool to Estimate the Impact of Design Change, *International Design Conference (DESIGN 10)*, Dubrovnik, Croatia, 2010

[2] Deubzer, F., Kreimeyer, M., Lindemann, U., Exploring Strategies in Change Management – Current Status and Activity Benchmark, *International Design Conference (DESIGN 06)*, Dubrovnik – Croatia, 2006, pp815-822

[3] Gupta, S., Okudan, G.E., Framework for Generating Modularized Product Designs with Assembly and Variety Considerations, *Proc. of ASME IDETC/CIE 2008*, Brooklyn, Newyork, USA, 2008

[4] Forslund, K., Kero, T., Söderberg, R., Appearance FMEA – A Method for Appearance Quality Evaluation of Early Design Concepts, *Proc. of ASME IDETC/CIE 2009*, San Diego, California, USA, 2009

[5] Pahl, G., Beitz, W., *Engineering design: A systematic approach*, 1995 (Springer)

[6] Cross, N., Christiaans, H. and Dorst, K., *Analysing Design Activity*, 1996 (Wiley)

[7] Leifer, L., *Evaluating product-based-learning education*, 1995 http://www-cdr.stanford.edu/~leifer/publications/Osaka95/Osaka95.html

[8] Bucciarelli L.L., *Designing Engineers*, 1994 (MIT Press, Cambridge MA)

[9] Eckert, C.M., Clarkson, P.J. and Zanker, W., Change and Customization in Complex Engineering Domains, *Research in Engineering Design*, Vol. 15, 2004, pp1-21

[10] Ahmad, N., Wynn, D.C. and Clarkson, P.J., Information models used to manage changes: A review of literature 2005-2010, 2011, *Proc. International Conference on Engineering Design (ICED'11)*, Denmark, 2011

[11] Wynn, D.C., Wyatt, D.F., Nair, S.M.T. and Clarkson, P.J., An introduction to the Cambridge Advanced Modeller, *1st International Conference on Modelling and Management of Engineering Processes (MMEP 2010)*, Cambridge, United Kingdom, 2010

Contact: Naveed Ahmad
University of Cambridge
Engineering Design Center
Department of Engineering
Cambridge, CB2 1PZ
United Kingdom
Tel: Int +44 1223 748564
Fax: Int +441223 332662
Email: na315@cam.ac.uk
URL: http://www-edc.eng.cam.ac.uk/people/na315.html