ANALYSING DESIGN PROTOCOLS: DEVELOPMENT OF METHODS AND TOOLS

John S Gero

Krasnow Institute for Advanced Study, Fairfax, VA, USA
Email: john@johngero.com

Jeff WT Kan

Taylor's University, Subang Jaya, Malaysia and Krasnow, Institute for Advanced Study, Fairfax, VA, USA
Email: kan.jeff@gmail.com

Morteza Pourmohamadi

University of Sydney, Sydney Australia and Krasnow Institute for Advanced Study, Fairfax, VA, USA
Email: mpou3922@uni.sydney.edu.au

This paper presents the development of methods and tools to analyse design protocols so that the results of different researchers studying different designers under different conditions can be compared. The foundation is an ontologically-based coding scheme utilizing the Function-Behaviour-Structure (FBS) ontology. This produces segmented and coded protocols showing design issues and processes independent of the design task or situation. Protocol measurements include tabular statistics, dynamic modeling of issues and processes, Markov models, first passage models and entropy models. These measurements are embodied in a computational tool. Examples of the use of the tool are presented.

Keywords: Design protocols, ontologically-based coding, measurement methods, analysis tool.

1. INTRODUCTION

The development of an understanding of designing requires empirical data from designers designing on which to found and test models of designing. One method of collection of that data is through the study of design. Most models of designing assume that designing is a process, rather than being some mysterious activity, and as a consequence design cognition can be studied scientifically. The most commonly used method for studying design cognition is protocol analysis [1–3]. It has produced results across a variety of domains [4–10]. However, there has been a significant impediment with the application of protocol analysis in studying designing and this has been the lack of the means to compare the results of different analyses. This has been caused by the use of ad hoc coding schemes applied by different researchers even when studying the same task. This is evidenced by the results where different researchers carried out protocol analyses over the same data set [5, 11]. Each research group used their own coding scheme and as a consequence there was no way to compare the results.

Whilst there are many ways of viewing designing the claim is made that the fundamental issues and processes involved in designing are not uniquely related to any particular design task [12–19] and can be studied independently of the design being produced. Therefore, it should be possible to produce a small number of coding schemes for protocol studies that addresses fundamental issues and processes;

Research into Design — Supporting Sustainable Product Development. *Edited by* Amaresh Chakrabarti

ISBN: 978-981-08-7721-7

coding schemes that can be used across design domains and design environments. Preferably what is needed is a principled coding scheme rather than an ad hoc one, one that is based on an ontology [20, 21] of designing.

Without agreed methods of coding and measurement the results of cognitive studies of designers carried out under different conditions by different researchers cannot be compared. If the results of research cannot be compared then the field cannot progress.

The remainder of this paper presents the protocol analysis method, one principled coding scheme for use with protocol studies and then proceeds to describe a number of methods of measuring design cognition from protocols. The final section shows the outcome of embodying these methods in a single tool that produces the results in a consistent form, allowing for comparisons across different protocols.

2. PROTOCOL ANALYSIS

Protocol analysis is a rigorous methodology for eliciting verbal reports of thought sequences as a valid source of data on thinking. It is a well-developed, validated method for the acquisition of data on thinking [1, 3, 22]. It has been used extensively in design research to assist in the development of the understanding of the cognitive behavior of designers [4, 6–11, 23–28].

The basic methodology of the protocol analysis method consists of the following sequence of tasks.

Coding development; Videoing designers; Transcription of video; Segmentation and coding; Analysis of coded protocol; Generation of linkograph; and Analysis of linkograph

3. ONTOLOGICALLY-BASED SEGMENTATION AND CODING

The coding scheme presented here is based on a general design ontology, namely, the Function-Behavior-Structure (FBS) ontology [29] and its extension, the situated Function-Behavior-Structure (sFBS) ontology [30] as a principled coding scheme. Google Scholar service indicates that these two papers, which outline the ontology, have over 900 citations between them.

The ontological variables that map on to design issues are: function, behavior, and structure plus a design description, Figure 1. Outside the direct control of the designer is an additional variable, the set of requirements, labeled R, provided by the client. In this view the goal of designing is to transform a set of functions into a set of design descriptions (D). The function (F) of a designed object is defined as its teleology; the behavior (B) of that object is either derived (Bs) or expected (Be) from the structure, where structure (S) represents the components of an object and their relationships. A design description is never transformed directly from the function but is a result of a series of processes among the FBS variables. These processes are a consequence of transforming one issue into another and include: formulation which transforms functions into a set of expected behaviors (process 1 in Figure 2); synthesis, where a structure is proposed to fulfill the expected behaviors (process 2); an analysis of the structure produces derived behavior (process 3); an evaluation process acts between the expected behavior and the behavior derived from structure (process 4); documentation, which

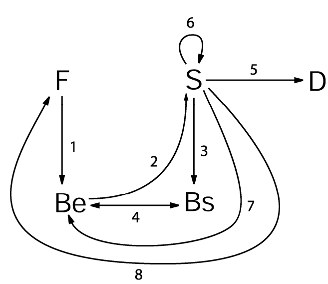

Figure 1. The FBS ontology (see text above to decode the numbers).

produces the design or partial design description (process 5). There are three types of reformulation: reformulation I–reformulation of structure (process 6), reformulation II–reformulation of expected behavior (process 7), and reformulation III—reformulation of function (process 8). Figure 1 shows the relationships among the eight transformation processes and the three basic classes of issues, which claim to be the fundamental processes for designing.

4. MEASUREMENT METHODS

The FBS based coding scheme converts a verbal design protocol into a series of segments where each segment addresses a single ontological issue. This approach also deals with the question of what makes a segment since each segment has only one code and there is only one segment per code. A consequence of this is that a segment is always determined in the consistent manner. The transitions between segments can be measured in two ways: syntactic linkograph and semantic linkograph. In a syntactic linkograph, only the adjacent segments are considered to extract the series of the processes, i.e., a transition from one issue to the issue directly following it defines a process. This is a weak model of design processes. In semantic linkograph [31] the links between segments are made on the basis of their semantic connection. A backlink from segment A to segment B defines a process BA, and depending on the issues A and B as to what design process is found [32]. For example if the issue of the originating segment is S and the issue of the backlinked segment is Be the process Be → S maps onto the design process 'synthesis'. Using issues and processes of a protocol a range of uniform analyses becomes possible.

4.1. Tabular statistics of issues and processes

There are now two source datasets from the protocol. The first is the issues coded through the FBS codes and the second is the linkographs that are the results of linking issues. The distributions of issues, syntactic processes and semantic processes are measurable statistically. These distributions measure the amount of cognitive efforts spent on each issue through the length of a design session.

The statistical analyses can be used to compare the results of one protocol with another. Despite the extensive use of statistical analysis in traditional coding schemes, the ad hoc nature of the schemes does not allow for comparisons between different cases and across different domains. On the other hand, the ontological basis of the FBS coding scheme allows for comparative studies of the statistical results obtained from different cases of design protocols.

4.2. Dynamic modeling of issues and processes

The goal of designing is to transform a set of functions into a set of design descriptions. This does not imply that designing is a linear process between these two points. Studies show that a designer goes through a variety of transitional processes to reach an acceptable solution. To capture the dynamic nature of designing, dynamic models are used for a more detailed analysis of coded protocols.

There are multiple ways to study the dynamic nature of designing. This paper presents four approaches:

fractioning; windowing; and trimming

In *fractioning* the protocol is divided into two or more sections and each section described statistically. The resulting measurements can be compared with each to determine if there are differences in behavior over a design session.

In *windowing* a window of a fixed number of segments is produced and the statistical analysis carried out for that window as it commences at the beginning of the session, with its left edge at segment 1, and then moves a single segment over with the analysis repeated. The movement of the window a single segment at a time is repeated under the window's right edge hits the last segment of the protocol. For each window position an independent calculation is carried out and assigned to the central segment of that window. Putting the result of calculations together, a dynamic model is produced that shows the changing values of the issues and processes in the course of the design session.

In *trimming* the protocol is trimmed to a specific contiguous set of segments that is the active section of the protocol under consideration. This allows any part of the protocol to be analyzed separately and then compared to any other part of the protocol.

4.3. Markov models

A Markov model [33] describes the probability of moving from one state to another in a stochastic system. In the FBS coding scheme, each issue represents the state of affairs in that segment. In cognitive studies of designing a Markov model measures the probability of one particular design issue being followed by another particular design issue. This model had been used to compare different design sessions and showed that there is a lack of evaluation (process 4) when designing in 3D virtual environment in that study [34]. The transition matrix for protocols is a 6×6 matrix with the rows and columns labeled with the six FBS issues. Every cell of the matrix shows the probability of a design issue leading to the corresponding issue. This is called a 'first order Markov model'. The dynamic Markov model can be produced using the same method of sliding a window of a fixed size along the segments.

In a more advanced analysis, not only the current state of the system, but also the previous state is taken into account in predicting the next state, i.e. the system has a memory. This is called a 'second order Markov model'. In analyzing coded design protocols, a transition from one state into another is a process. Thus, the second order Markov model of a coded protocol measures the probability of a particular issue following a process. The transition matrix of this model has eight rows, including of all eight designing processes and six columns for the issues. The syntactic and semantic modes are also applicable in the calculation of each model.

4.4. First passage models

A first passage model (sometimes called a first passage event model) [33, 34] measures the average number of segments that a designer traverses before returning to the same issue. This is a measure of the designer's focus on an issue. It can be measured within a single protocol as an overall measure or using fractioning to determine changes in behavior during a protocol. As an overall measure it can be used to compare different protocols.

4.5. Entropy models

Further understanding about the behavior of designers can be obtained by examining the information of their protocols. Here information is a measure of the potential of a system, rather than its semantic content. The measure of information is information entropy. Originally, Shannon [35] introduced entropy as a measurement for information. In Shannon's theory, the amount of information carried by a message is defined by the probability of its outcomes. The more stochastic a system is, the more informative is the definition of its state. By measuring the entropy of semantic linkographs from different design sessions, Kan and Gero [32] argued that there is a potential relation between the productivity of design activities and the entropy of their corresponding linkographs.

Using the same idea about incrementally sliding a window along the design session, the dynamic entropy of the session can be measured. The dynamic entropy only takes the structure of the linkographs into account and currently ignores the codes in the segments.

The dynamic entropy offers additional dimensions to explore to assist in understanding the behavior of designers. For example, it has been used to study fixation and commitment [36].

5. A TOOL FOR PROTOCOL ANALYSIS MEASUREMENTS

Quantitative analysis of design protocols are time consuming and as a consequence are expensive to carry out. The dependency of the traditional analysis methods on their specific coding methods has been a barrier to developing universal toolkits to perform quantitative analysis. The applicability of the FBS coding scheme and its independence from the design domain allowed us to develop a standardized measurement toolkit, called LINKOgrapher, Figure 2. LINKOgrapher carries out all of the measurement methods introduced above and generates corresponding models for each of them. This dramatically reduces the cost and time of performing such measurements and allows researchers to be more focused on the results rather than calculations.

The FBS coding scheme transforms a transcribed design session into a set of FBS-based issues and links. For each of the analysis methods described in Section 4 LINKOgrapher generates two types of output: graphic and textual. The graphic output is in pdf format and the text output is in a form that can be imported into standard spreadsheet software. LINKOgrapher currently can produce the following measurements, as well as drawing the linkograph:

tabular statistics of issues and processes; dynamic issues; dynamic processes; dynamic entropy; overall (or first order) Markov model; second order (or dynamic) Markov model; and average first passage event model.

The utility of the tool is enhanced by being able to control the window size for all dynamic calculations and separately control the fractioning and trimming. The measurement characteristics and speed of LINKOgrapher coupled with the ontological basis of the FBS coding scheme allow for an explorative approach toward quantitative studies of design protocols. This approach transcends any particular designer, design discipline or any specific characteristics of the design situation.

Figure 8 shows the comparison of the Markov models of design process in two disparate domains: architecture and software, where both protocols were analysed using this paradigm, which then makes direct comparison possible.

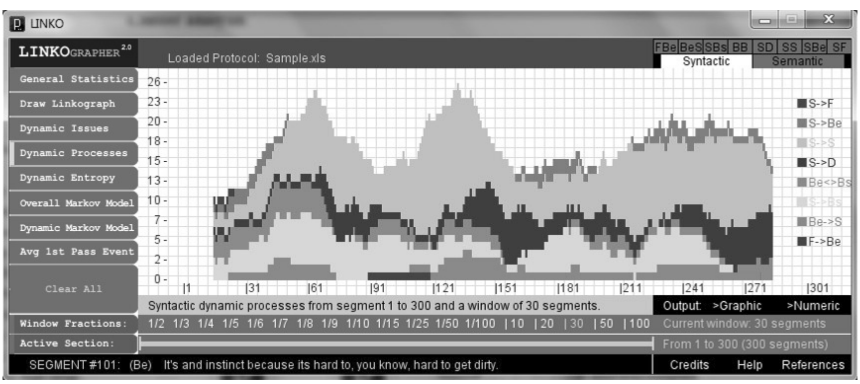

Figure 2. Screenshot of LINKOgrapher toolkit showing interface and graph of dynamic processes.

General statistics from segment 1 to 1280.

Total Segments:	1280			
Non-FBS Segments:	3 (% 0)			
Total Links:	5191			
Link Ratio:	4.06 per segment			
Overall Entropy:	0.089			

Issue Distribution (%):

R	0 (0.0)	F	47 (3.7)
Be	272 (21.3)	Bs	363 (28.4)
S	518 (40.6)	D	77 (6.0)

	Syntactic Process Distribution (%):	Semantic Process Distribution (%):
Formulation	21 (2.8)	80 (2.5)
Synthesis	97 (12.9)	388 (12.1)
Analysis	151 (20.1)	578 (18.0)
Evaluation	117 (15.6)	577 (18.0)
Documentation	41 (5.5)	108 (3.4)
Reformulation I	242 (32.2)	1076 (33.5)
Reformulation II	72 (9.6)	373 (11.6)
Reformulation III	11 (1.5)	30 (0.9)

by LINKO grapher

Figure 3. Typical tabular statistics presented as an image.

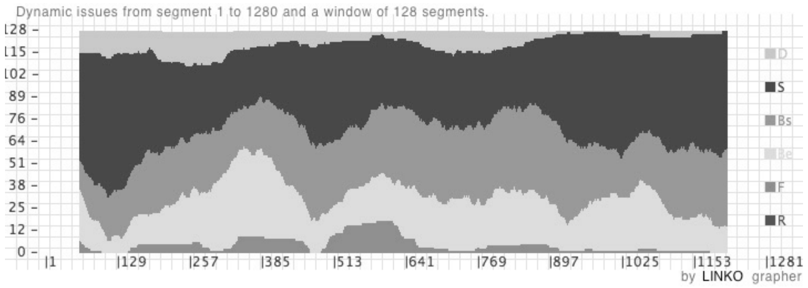

Figure 4. Dynamic issues showing the distribution of design issues across the protocol with a window 1/10th of the length of the protocol.

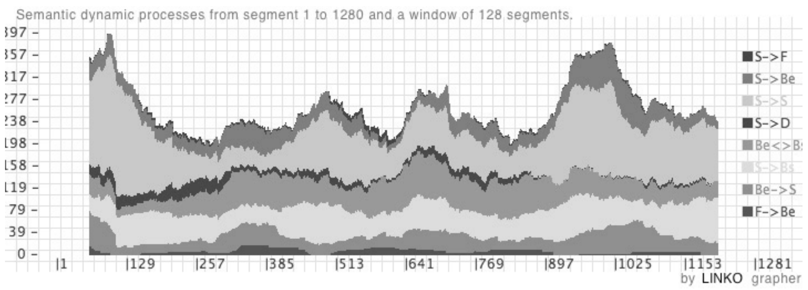

Figure 5. Dynamic processes showing the distribution of design processes across the protocol with a window 1/10th of the length of the protocol.

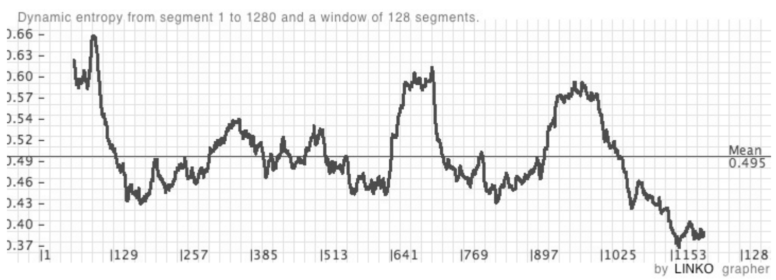

Figure 6. Dynamic entropy showing the distribution of entropy across the protocol with a window 1/10th of the length of the protocol.

6. EXEMPLARY RESULTS

Some results are presented to indicate the capabilities of the approach and the tool, Figures 3 to 6. A protocol of a large design team from industry is used as a demonstration vehicle. The protocol is over one hour long and has been coded into 1280 segments.

These results allow for the characterization of the designing behavior of this design team in such a way that the characterization can be compared with other teams and individuals.

Figure 7 shows the comparison of design issues distribution in two disparate domains: architecture and software, where both protocols were analysed using this paradigm, which then makes direct comparison possible.

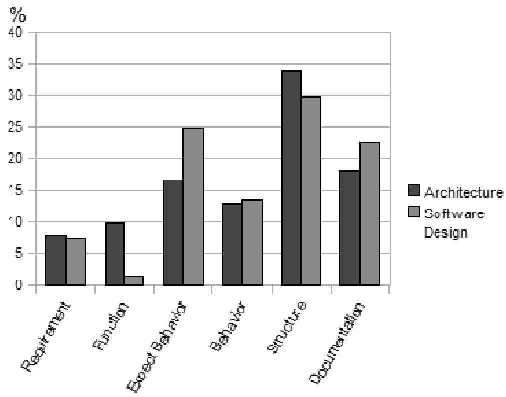

Figure 7. Comparison of the distributions of issues between architects and software designers.

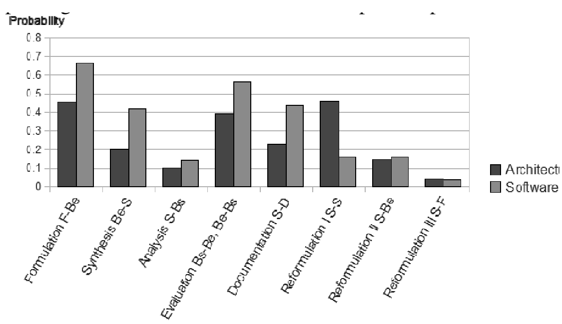

Figure 8. Comparison of the Markov models of design processes between architects and software designers.

7. CONCLUSIONS

The methods and tool presented in this paper allow for research into designing using the protocol analysis approach that covers the following designer variables:

individual designers; homogeneous design teams; heterogeneous design teams; student designers; professional designers; designers from different disciplines; co-located designers; remotely located designers; and designers using digital tools.

The use of a generally applicable coding scheme turns protocol studies from an individual research group's internal activity to a means of producing knowledge that can be transferred from one researcher to another. The particular coding scheme adopted here is but one of a number of potential general schemes that could focus on different ways of viewing the acts of designing. The argument for the adoption of this scheme lies with its ontological foundation and the widespread references to it in the research literature. As more generally applicable schemes are developed and used the field of design cognition will develop into a coherent research field.

ACKNOWLEDGMENTS

This research is based upon work supported by the National Science Foundation under Grant Nos. SBE-0750853, EEC-0934824 and CMMI-0926908. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation. Parts of this paper are drawn from the discussion paper presented at DTRS8, Sydney Australia.

REFERENCES

1. Ericsson, K. and H. Simon, “Protocol Analysis; Verbal Reports as Data”, MIT, 1993.
2. Svenson, O., “A note on think aloud protocols obtained during the choice of a home”, Psychological Laboratories, University of Stockholm, 1974.
3. Van-Someren, M., Y. Barnard and J. Sandberg, “The Think Aloud Method: A Practical Guide to Modelling Cognitive Processes”, Academic Press, 1994.
4. Adams, R. and C. Atman. “Cognitive processes in iterative design behavior”, Proceedings of the Annual Frontiers in Education Conference. San Juan, Puerto Rico, 1999.
5. Cross, N., H. Christiaans and K. Dorst, “Analysing Design Activity”, Wiley, 1996.
6. Cross, N., K. Dorst and N. Roozenburg, “Research in Design Thinking”, Delft University Press, 1992.
7. Ennis, C. and S. Gyeszly, “Protocol analysis of engineering systems design process”, Research in Engineering Design 3(1): 15–22, 1991.
8. Gero, J. and T. McNeill, “An approach to the analysis of design protocols”, Design Studies, 19(1): 21–61, 1998.
9. Kavakli, M. and J. Gero, “The structure of concurrent cognitive actions: A case study of novice and expert designers”, Design Studies, 23(1): 25–44, 2002.
10. McNeill, T., J. Gero and J. Warren, “Understanding conceptual electronic design using protocol analysis”, Research in Engineering Design, 10(3): 129–140, 1998.
11. McDonnell, J. and P. Lloyd eds. “About: Designing. Analysing Design Meetings”, CRC Press, 2009.
12. Asimov, W., “Introduction to Design”, Prentice-Hall, 1962.
13. Coyne, R. *et al.*, “Knowledge-Based Design Systems”, Addison-Wesley, 1990.
14. Dym, C., “Engineering Design: A Synthesis of Views”, Cambridge University, 1994.
15. Ertas, A. *et al.*, “Principles of Engineering Design”, CRC Press, 2008.
16. Gero, J. ed., “Artificial Intelligence in Design’91”, Heinemann, 1991.
17. Gero, J. ed., “Design Computing and Cognition”, Springer, 2008.
18. Lawson, B., “How Designers Think”, Architect Press, 2005.
19. Ullman, D., “The Mechanical Design Process”, McGraw-Hill, 1992.
20. Gruber, T., “Toward principles for the design of ontologies used for knowledge sharing”, International Journal Human-Computer Studies, 43(5–6): 907–928, 1995.
21. Uschold, M. and M. Gruninger, “Ontologies: Principles, methods and applications”, Knowledge Engineering Review 11(2): 93–136, 1996.
22. Crutcher, R., “Telling what we know: The use of verbal report methodologies in psychological research”, *Psych. Sci.* 5: 241–244, 1994.
23. Atman, C. and K. Bursic, “Documenting a process: The use of verbal protocol analysis to study engineering student design”, *Journal of Engineering Education, Special Issue on Assessment* 121–132, 1998.
24. Badke-Schaub, P. *et al.*, “Task–team — process: Assessment and anlaysis of the development of shared representations in an engineering team”, DTRS 7 Design Meeting Protocols Workshop Proceedings, London, 2007.
25. Christensen, B. and C. Schunn, “The relationship of analogical distance to analogical function and pre-inventive structure: The case of engineering design”, Memory & Cognition 35(1): 29–38, 2007.
26. Suwa, M., T. Purcell and J. Gero, “Macroscopic analysis of design processes based on a scheme for coding designer’s actions”, *Design Studies*, 19(4): 455–483, 1998.
27. Tang, H.-H. and J. Gero, “A cognitive method to measure potential creativity in designing”, Workshop 17 — Creative Systems: Approaches to Creativity in AI and Cognitive Science, ECAI-02. Lyon, 2002.
28. Purcell, T. *et al.*, “The data in design protocols: The issue of data coding, data analysis in the development of models of the design process”, in Analysing Design Activity, N. Cross, H. Christiaans and K. Dorst, Editors, John Wiley: Chichester, 225–252, 1996.
29. Gero, J., “Design prototypes: A knowledge representation schema for design”, AI Magazine, 11(4): 26–36, 1990.
30. Gero, J. and U. Kannengiesser, “The situated Function-Behavior-Structure framework”, Design Studies, 25(4): 373–39, 2004.
31. Goldschmidt, G., “Linkography”, Cyberbetics and System’90: World Scientific, 1990.
32. Kan, J. and J. Gero, “Using the FBS ontology to capture semantic design information in design protocol studies”, in About: Designing. Analysing Design Meetings, J. McDonnell and P. Lloyd, Editors, CRC Press, 213–229, 2009.
33. Kemeny, J. and J. Snell, “Finite Markov chains”, Van Nostrand, 1960.
34. Kan, J. and J. Gero, “Universal Tools to Study Human Design Activities”, in Human Behavior in Design, R. Noell Bergendahl, M., Grimheden, M., Leifer, L., Skogstad, P. and Badke-Schaub, Editors, Design Society, 9:123–134, 2009.
35. Shannon, C., “A mathematical theory of communication”, The Bell System Technical Journal, 27: 397–423, 1948.
36. Gero, J., “Fixation while designing and its measurement”, in http://mason.gmu.edu/~jgero/publications/Progress/10GeroFixationDelft.pdf, 2010.