Using the FBS Ontology to Capture Semantic Design Information in Design Protocol Studies

Jeff W T Kan
Key Centre of Design Computing and Cognition, Faculty of Architecture, Design and Planning, University of Sydney
kwaitak@arch.usyd.edu.au

John S Gero
Krasnow Institute for Advanced Study and Volgenau School of Information Technology and Engineering, George Mason University, and University of Technology, Sydney
john@johngero.com

Abstract
This chapter presents a method to capture semantic information from design protocols. We report on a preliminary study that analyses a design protocol by using the FBS ontology and derives processes within this ontological framework by employing linkography. The usefulness of this method is examined by applying it to the Engineering 1 protocol (E1) as a case study. The original 1990 FBS ontology captures 66% meaningful processes of all the derived processes, while the situated FBS ontology captures 92% meaningful processes of all the derived processes. Further coding analysis may improve this percentage. The session is characterized, according to the ontology, by the high percentage of behaviour reformulation, followed by structure reformulation, and analysis.

1. Introduction

The Delft 1994 workshop (Cross et al 1996) showed there is a variety of coding schemes that can be applied to study the same design protocol; these help to gain a rich understanding of different aspect of designing. However, the diversity and uniqueness of each scheme makes it hard to reuse and compare protocol results either qualitatively or quantitatively. The motivation behind this work is to develop a general coding scheme that yields high quality, uniform results, that maps well to the behaviour of designers, produces a deeper understanding of design thinking and activities and can be applied across protocols independently of the domain and the number of participants. The general coding scheme is based on an ontology of the domain of designing and as a consequence is not an ad hoc development specific to a unique protocol but one that can be used uniformly across design protocols independently of the specific design activity being studied and unrelated to the number of participants in a design team.

This chapter explores the use of the FBS ontology as a general coding scheme to study designing. Its aim is to capture semantic information from design protocols. This semantic information can then be utilized: 1) to explore different aspect of designing according to the focus of interest; and 2) to locate different types of design transformation processes.

1.1 FBS Ontology

In this section a brief summary of the FBS framework with its relation to design and design creativity is presented. The FBS ontology framework (Gero 1990) models designing in terms of three basic classes of variables: function, behaviour, and structure. 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 purposes or teleology; the behaviour (B) of that object is how it achieves its functions and is either derived (Bs) or expected (Be) from the structure, where structure (S) is the elements of an object and their relationships. A design description is never transformed directly from the function but undergoes a series of processes among the FBS variables. These processes include: a formulation which transform functions into a set of expected behaviours; a synthesis, wherein a structure is proposed that is likely to exhibit the expected behaviour; an analysis of the structure produces its derived behaviour; an evaluation process acts between the expected behaviour and the behaviour derived from structure; and documentation, which produces the design description. Based on the structure there are three types of reformulation, where new variables are introduced: reformulation of structure, reformulation of expected behaviour, and reformulation of function. Reformulation of function is relatively rare, as it changes or redefines the design problem. Figure 1 shows the relationships among the eight transformation processes and the three basic classes of variables.

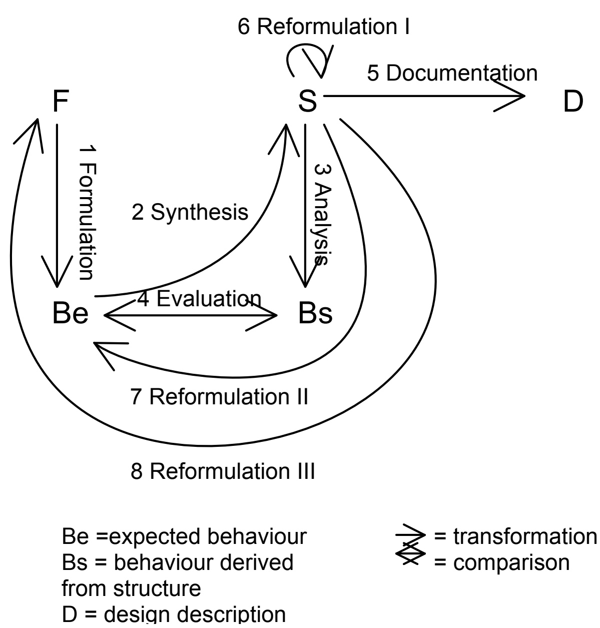


Figure 1. The FBS ontology of designing (after Gero 1990)

These eight processes are claimed as the fundamental processes for designing. For example, analogical reasoning—reported to play an important role in creative designing (Gero and Maher 1991, Goldschmidt 2001)—is consider as part of the reformulation processes. Analogies can be of function, behaviour or structure.

2. DTRS7 Engineering 1 Protocol

This particular engineering session was selected because of its content. Judged by watching the recording the architectural sessions were mainly presenting and communicating with clients; relative to the engineering sessions they contained fewer design activities. Although both engineering sessions concerned creating a new thermal printing pen, the session studied involved generating novel ideas based on analogies. The other session (E2) related more to generating ideas for the usage and control.

This engineering protocol can be divided into two episodes; the first one concerned the problem of keeping the print head in contact with the media and the optimum angle to the media, despite wobbly arm moment. The second episode dealt with protecting the print head from abusive use and overheating. In the first episode participants were asked to generate ideas from available products that follow a contour. There were seven participants involved in this session, in Gericke et al's chapter (this volume) they depicted their disciplinary area and in Adams et al's chapter (this volume) they had a detailed analysis of their roles. Several products were mentioned such as: a sledge, snowboard, wind surfboard, shaver, snow mobile, train, and slicer. Other concepts such as wheels, spirit level, and laser leveler were also discussed. Loosely related to those analogies, a few shapes, such as mouse-type pen, were proposed. Besides product behaviour, user behaviour was also considered. The outcome, sketches and drawings, were documented and analyzed in Medeiros and Gomes's chapter (this volume). Most of the analogies were observed in the first episode so it was selected for analysis in this chapter.

3. Coding Scheme

The coding scheme consists of the FBS classes – five codes (F, Be, Bs, S and D as in Figure 1) and two additional codes: requirement (R) and others (O) that did not fit within these codes. In the Gero's (1990) FBS computational model, designing was assumed to start with function. Later in the Gero and Kannengiesser's (2004) situated FBS framework (a cognitive model), designing was viewed to start with requirements The R code is used because in protocol studies the designing activities start with requirements instead of function. Below are some examples from the protocol for each code.

Example of requirements:

"quite important is its about the thermal-incli- inclis ( ) pen" (E1, 43)

"design a-a prototype" (E1, 56-57)

Examples of function:

"that's the standard plain thermal paper err and then it can draw" (E1, 54)

Examples of expected behaviour:

"either atoms or line types" (E1, 55)

"we can print thermo reactive dyes onto media substrates" (E1, 68)

Examples of derived behaviour:

"it'll be about fifty percent more expensive" (E1, 199)

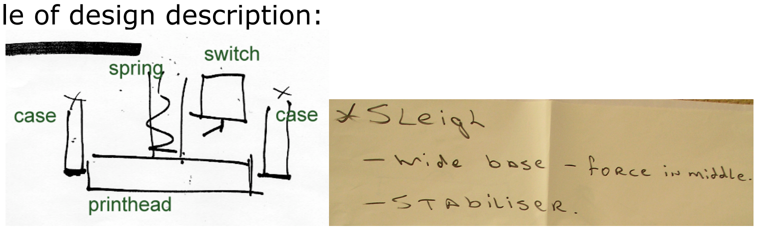
"if you lift an optical mouse slightly off the page you'll see the pattern it creates" (E1, 672 674)

Examples of structure:

“…sledge” (E1, 137)

“show the relative size of the pen if you've got an example” (E1, 171)

Example of design description:



Examples of others:

“yeah we'll come to that in a minute” (E1, 737)

3.1 Coding Segments

The protocol is segmented strictly according to the ontology—each segment contains only one class of FBS code. The segmenting and coding are done simultaneously by discerning whether an action or utterance expresses the FBS aspect of designing or concerns the requirements or others. If an utterance contains more than one class it will be further divided. This also applies to the ‘O’ and ‘R’ codes. Drawing and writing actions are also considered as segments of structure. By doing this, the segments will not have a fixed duration and are usually very short, typically with a magnitude of a few seconds.

The two episodes are about the same length, 57 minutes each. This study will use only the first episode as an illustration. We disregard the first 5 minutes as it involves the management of the meeting rather than the design process. It contains seven segments and the majority of them are about the rules for brainstorming. The protocol was coded twice by the same coder with a ten days separation and then self-arbitrated using the Delphi method proposed by Gero and Mc Neill (1998). The agreement of two codings is over 86%. Within the 52 minutes in the first episode we coded 475 segments. The average segment length is 6.5 seconds. Of the segments 448 segments have FBS codes; those “others” coded segments (27) consist mostly of jokes or communications that are not related to the design process or the resulting artefact. Figure 2 shows the percentages in each of the FBS categories in relation to the processes of the FBS ontology. However, this does not give us the distributions of the design processes. The requirement and function occupy only about 5% of the protocol. The highest percentages are in the structure and behaviour classes. In this protocol these high percentages are due to the frequent use of analogies with other products and situations.

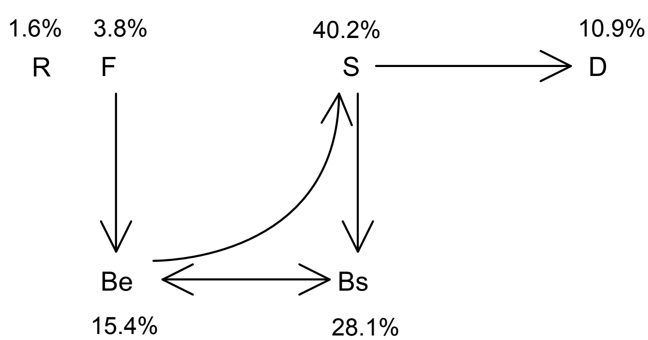
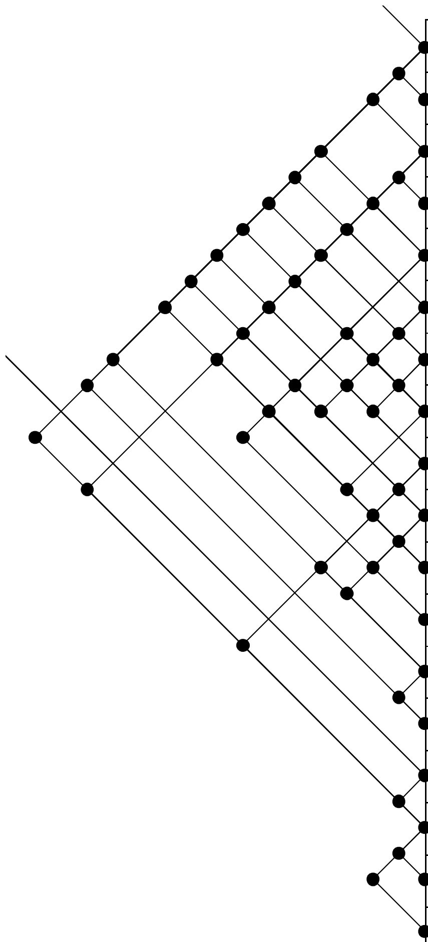


Figure 2. Percentages of each code

3.2 Linking the segments

The connections between the segments, independent of the code, are discerned by using Goldschmidt's (1990) linkography technique. Figure 3 presents an example of the extracted coded protocol (E1 137-157) together with the constructed linkograph.



38	S	J: I ended up with the + hold on +sledge
39	D	A: the sledge excellent so what did that generate then? (write: sledge)
40	Bs	J: the sledge manages to keep level by having quite a wide base
41	D	A: (write: wide base)
42	Bs	J: a main force in the middle
43	S	J: unlike the set of skis
44	S	J: where quite narrow and
45	Bs	J: you go up on an edge when you're turning
46	S	J: the sledge is er quite broad
47	Bs	J: and then you have the weight right in the middle
48	Bs	so they manage to keep both runners on the snow
49	D	A: (write: force in middle)
50	Be	J: a sledge or a snowboar- a skis or snowboard
51	S	A: some guiders almost down the side of this
52	Be	J: the easiest way to keep the pen at a right angle would be
53	S	J: to have a set of stabilisers on it based on the idea of a sledge
54	S	A: stabilisers +++like a bicycle yeah that's a good
55	D	A: (write: stabiliser)

Figure 3. Rotated linkograph in relation to the protocols, in the table column 1 is the segment number, column 2 is the code and column 3 is the transcribed protocol.

In this chunk, a cluster of links, two participants were involved, the moderator (A) and a mechanical engineer (J). The focus of the discussion was "other products or situations where a product needs to follow a contour". J suggested an object (structure) – "sledge" (segment 38) – and continued to explain the behaviour of the sledge: how it maintains contact or level on the snow (segment 40 and 48). The sledge was compared with a set of skis (segment 43) in terms of the structure (segment 44) and behaviour (segments 45, 47 and 48). The coding of segment 50 can be controversial; it was coded as expected behaviour (Be) as we interpreted J was borrowing the behaviour of the analogised objects and targeting those to be the expected behaviour of the designed object. Finally, the structure of stabilisers (segment 53) was suggested. Segment 39 was linked to segment 38 because the "write sledge" action was a response to the initiation and suggestion of the "sledge" in segment 38. J started explaining in segment 40 why a sledge was a proposed candidate for solution so segment 38 and 40 was linked. By examining the relationship of a segment with those preceding segments a linkograph was constructed.

Figure 4 shows a larger part of the linkograph of this session that includes the above chunk. Other chunks were also labelled. These clusters were distinguished by the visual inspection of link density. In another study, Kan and Gero (2008) used statistical method to detect the clusters. With the linkograph, we can trace the structure of reasoning (Goldschmidt 1990). The strength of an idea can be compared quantitatively either by using the critical move measurement (Goldschmidt 1990) or the entropic measurement (Kan and Gero 2007). These were not examined here because the focus of this study concerns combining ontological coding with linkography to capture semantic information from the protocol.

There are 1,010 links among the 475 segments, so on average each segment has about 2.1 links. However, some segments have many more links than others. Table 1 compares the distribution of the codes of the segments with the occurrences of codes in the links. Compared to the coded segments it can be observed that the codes in the links decrease for documentation, have a moderate decrease in behaviour derived from structure, and a slight decrease in function. The requirement has increased, and there is an increase in expected behaviour and structure as well. This implies the expected requirement, behaviour, and structure segments in this session are more influential. The statistics of 1990 FBS coding, as for Gero and McNeill (1998) and McNeill et al (1998), do not reflect this.

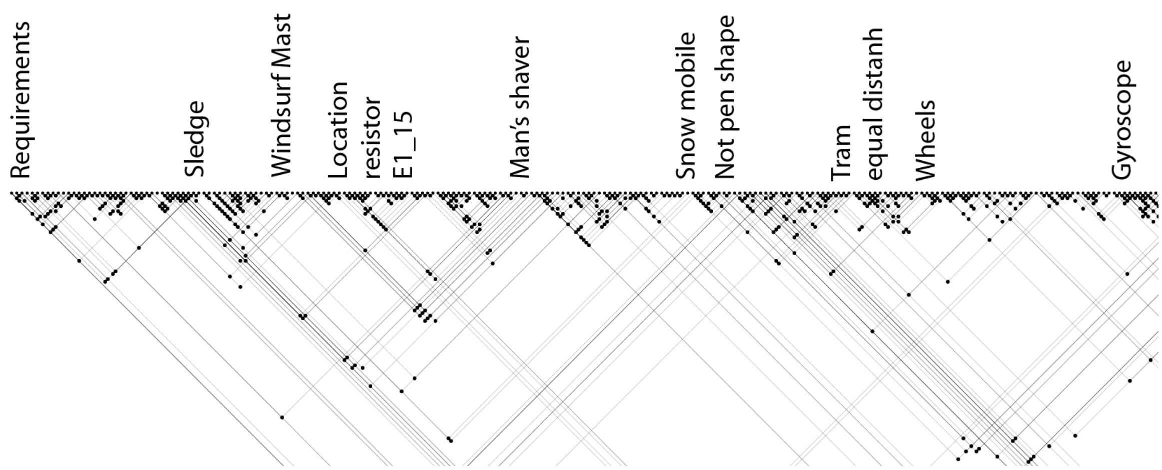


Figure 4. Part of the linkograph of the segmented protocol

Table 1. Comparing the distribution of codes in segments and links

Code	Segment		Links	
R	7	1.6%	36	1.7%
F	17	3.8%	56	2.6%
Bs	126	28.1%	504	23.8%
Be	69	15.4%	396	18.7%
S	180	40.2%	936	44.3%
D	49	10.9%	187	8.8%
Total	445	100%	2110	100%

4. Deriving FBS processes from coded segments and links

In the following analysis, we use the symbol ">" to denote the link or the transformation between the nth and the (n+i)th segments. For example, consider the first segment linking to the two subsequence segments in Figure 3, S>D was used to represent the link between segments 38 and 39. S>Bs was used to represent the link between segments 38 and 40, Figure 5 illustrates this example. S>D can be seen as the documentation process (transformation from structure to design description) and the S>Bs as the analysis process (transformation structure to behaviour) according to the ontology.

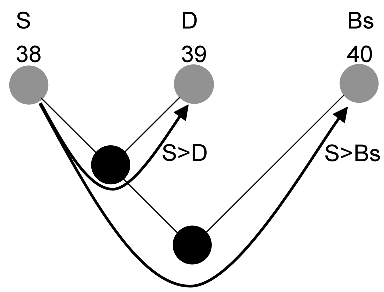


Figure 5. Deriving transformation processes from linkograph

The 981 links were viewed as design processes. The linkograph become a network of transformation processes. There are 6 categories of FBS codes, excluding O, so there will be 36 types of possible transformations. However, according to the FBS ontology many of those processes have no meaning. For example no instance of R>D was recorded. However there was a record of the F>S process which the framework does not permit. Table 2 shows the FBS related processes derived from the links of the coded segments. There were 30 types of FBS processes recorded; those FBS processes in the framework are represented in Table 3. The results are comparable to Badke-Schaub et al's results (this volume) about the frequency of cognitive acts in meeting 1 although they measured the whole session. The distribution of activities corresponds roughly. They have about 2.5 % of "Problem definition" that can be compare to the 1.4% of "Formulation"; their 30% "Analysis & Evaluation" can be compared to our 22.1% (12.2%+9.9%); their 6.5% "New ideas" is higher than our 3.4% of Reformulation II and III (structural reformulations do not usually contribute to new ideas but in this session there are various structural type of analogies).

Table 2. Percentages of all the processes derived from codes and links

Requirement		Function		Behaviour		Expected Behaviour		Structure		Description	
R>Bs	1.3	F>Bs	0.2	Bs>Bs	8.4	Be>Bs	4.9	S>Bs	12.2	D>Bs	0.5
R>Be	0.2	F>Be	1.1	Bs>Be	5.0	Be>Be	6.9	S>Be	3.3	D>Be	0.9
R>F	0.1	F>F	1.4	Bs>D	0.9	Be>D	1.8	S>D	5.3	D>D	1.4
R>R	0.3	F>R	0.1	Bs>F	0.5	Be>F	0.1	S>F	0.1	D>S	5.4
R>S	1.0	F>S	0.1	Bs>S	2.7	Be>S	6.9	S>R	0.1		
								S>S	26.7		

Table 3. Percentages of the 8 FBS processes

Processes		Occurrence	Percentage
Formulation	R>F,F>Be	14	1.4
Synthesis	Be>S	68	6.9
Analysis	S>Bs	120	12.2
Documentation	S>D	52	5.3
Evaluation	Be<>Bs	48(Be>Bs), 49(Bs>Be) 97	9.9
Reformulation I	S>S	262	26.7
Reformulation II	S>Be	32	3.3
Reformulation II	S>F	1	0.1
	Total	646	65.9

In this episode, the reformulations were mostly of structure and behaviour. The sledge example in Figure 3 contains the reformulation of structure (S>S), from the structure of a sledge to the structure of a set of stabilizers like those in a bicycle, segment 38 to 53 and 53 to 54.

Other examples of structure reformulations were: making analogies with other products, for example wind surfboard mast and man's shaver; considering the thermal pen in the shape of other things instead of a pen. Examples of behaviour reformulations were: using a universal joint to keep the angle; using springs to keep it level; suggesting the locations of resistors (E1 190) prompted the responses of the cost (E1 195, 199).

The reformulation of function was rare which reflected the nature of this session – mechanical brainstorming for ideas to keep the thermal pen in contact with the media at a correct angle. Some of the functional aspects were deliberately not dealt with. For example in E1, 622 to 623 "could we sorry could we actually see what they're doing I mean are they drawing pictures or making invitations or Christmas cards or-" as in E1, 624 "erm ( ) we're going to try to deal with that a fair bit on Monday".

The 1990 FBS ontology covers more than half of the processes derived from the links of the coded segments. What are the other processes (34.1%)? Does that reflect the deficiency of this ontology? Some of the most frequent processes not counted in Table 3 are: Bs>Bs (8.5%), Be>Be (6.9%), D>S (5.4%), S>Be (3.3%), Bs>S (2.7%), and Be>D (1.8%).

Reviewing the protocol, in the case of Bs>S, the large scale of the granularity fails to pick up the Be in the Bs>S processes. If the granularity were finer, there should be an expected behaviour before the structure code. Using the example in Figure 3, segment 40 (E1, 141): "the sledge manages to keep level by having quite a wide base" was coded as Bs because it analyzes an existing product to get the "keep level" behaviour. This segment was linked to segment 53 (E1, 153-156): "the easiest way to keep the pen at a right angle would be to have a set of stabilizers on it based on the idea of a sledge" which was coded as structure because it proposed a structure, "a set of stabilizers". The idea of sledge, the behaviour of "keep level", was translated to expected behaviour of "at a right angle" (balance?) which leads to the structure of "a set of stabilizers".

The F>F and Be>Be can be viewed as reflections of function and behaviour in many cases. An example of Be>Be happened when they discussed the shape of the designed object does not need to resemble a pen. The moderator suggested in E1 358 that "…something else that gets pulled behind it for example" and an engineer responded in E1 370, "…they'll do is move the lump around" were linked and both segments were coded as expected behaviour (Be).

The D>S is the interpretation of depicted structure. The Bs>Bs usually is a result of further analysis, for example in Figure 3 link between segments 40 and 42 and links between segments 42 and 48 were further analysis of the action and reaction of the force (weight). Sometimes the Be>D transformation was the depiction of behaviour but the FBS ontology does not distinguish depiction of behaviour from depiction of structure. Using E1 370 as an example again, "…they'll do is move the lump around" was linked to the following segment where the moderator was writing down "(move lump)", which is a depiction of behaviour. These transformations are meaningful processes resulting from the interactions among members and artifacts. In order to capture these, we turn to the situated FBS framework (Gero and Kannengiesser 2004) and recode this episode.

5. Situated FBS Coding

The situated FBS framework or ontology makes use of new concepts: the notion of situated cognition developed by Clancey (1997); the idea of constructive memory based on Dewey's (1896) and Bartlett's (1932) work; and the observation of designing as an "interaction of making and seeing" by Schon and Wiggins (1992). Gero and Kannengiesser (2004) developed these ideas further and integrated them into the FBS ontology to form the situated FBS framework by introducing interactions among three worlds – the external, interpreted, and expected worlds. An agent or human interacts and understands the external world through his interpretation of the external world to form memories of his interpreted world. In order to change the external world (the act of designing) he "focuses" to transform experiences to produce the expected world before taking action in the external world. The FBS class variables reside in these three worlds. The superscript ^{e} indicates the variables are of external world, the superscript ^{i} signifies the variables are of internal world and low-case suffix e represents variables are of expected world. In this framework the original eight processes are increased to twenty to allow for these additional activities, Figure 6.

Table 4 relates the twenty situated FBS processes to the original eight processes (Gero and Kannengiesser 2004). Of particular interest are the formulation and reformulation processes in this framework. The formulation process involves: the interpretation of requirements (R) in terms of F^i, B^i, and S^i representations (processes 1, 2 and 3); reflecting, based on experience, on those representations (processes 4, 5 and 6); focusing on subsets on these internalized requirements (processes 7, 8 and 9); and process 10 that corresponds to the original formulation in the FBS framework. The focusing and reflecting (processes 4 to 9) appear in all the three types of reformulations. Reformulations II and III are not limited to be driven by structure alone but also by external representation of function (F^e, process 20) and behaviour (B^e, process 19) as well. Process 11, transforming expected behaviour (Be^i) to expected structure (Se^i) is the synthesis process; the analysis process involves interpreting the structure (process 13) and deriving the behaviour from structure (process 14). The evaluation process (process 15) is comparing the expected behaviour (Be^i) with the interpreted behaviour (B^i).

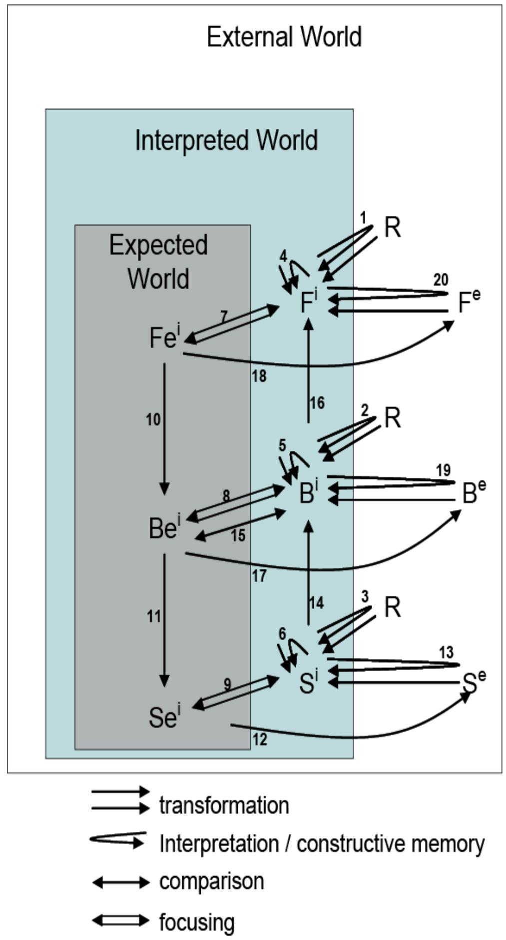


Figure 6. The situated FBS ontology (Gero and Kannengiesser 2004)

Table 4. Relating the 20 situated FBS processes to the original 8 FBS processes (Gero and Kannengiesser 2004)

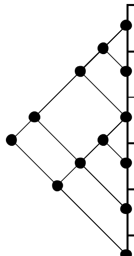
8 FBS processes	situated FBS processes
Formulation	1, 2, 3, 4, 5, 6, 7, 8, 9, 10
Synthesis	11
Analysis	13, 14
Evaluation	15
Documentation	12, 17, 18
Reformulation I	6, 9, 13
Reformulation II	5, 8, 14, 19
Reformulation III	4, 7, 16, 20

With the introduction of new classes of variables the new coding scheme is shown in Table 5. For ease of comparison, the segments have not been refined so the total numbers of segments and total number of links remains the same. As will be seen this has some unexpected consequences.

Table 5. Coding categories correspond to situated FBS ontology

R:	requirements derived from the given brief
F^i:	interpreted function either derived from requirements or ascribing meaning to depicted structure
Fe:	external representation of function, usually in terms of written words (not seen in this session)
Fe^i:	expected function resulted from focusing on F^i
B^i:	interpreted behaviour from depicted structure or requirements
Be:	external representation of behaviour, usually in terms of written words
Be^i:	expected behaviour derived from expected function or interpreted behaviour which result from requirement or interpreted structure
S^i:	interpreted structure either from external structure or from requirement
Se:	depiction that indicates structure
Se^i:	expected structure sometime without depiction

The code was coded twice by the same coder and then self-arbitrated using the Delphi method. The agreement of codes is about 67%. Figure 7 shows the recoding from segments 38 to 43. Segment 38 was coded as S^i because J was showing a picture of a sledge and was about to draw an analogy with the structure of the sledge. The main activity in segment 39 was writing down the word "sledge". It was treated as the documentation of structure as the word "sledge" denoted the object, so it was coded as S^e. Segment 40 was coded as B^i because it interpreted behaviour ("keep level") of the sledge. Segment 41 was coded as the depiction concerns the structural aspect ("wide base") of the object. Segment 42, similar to segment 40, involves the interpretation of behavioural aspect ("a main force in the middle") of the object, so the B^i code was assigned. Segment 43 concerns another object "skis" which was coded as S^i.



38	S^i	J: I ended up with the + hold on +sledge
39	S^e	A: the sledge excellent (write: sledge) so what did that generate then?
40	B^i	J: the sledge manages to keep level by having quite a wide base
41	S^e	A: (write: wide base)
42	B^i	J: a main force in the middle
43	S^i	J: unlike the set of skis

Figure 7. Examples of situated FBS coding, column one is the segment number and column two is the code.

Table 7 shows the distributions of codes in the segments and links and their percentages. There is no documentation of function. Table 7 indicates that expected behaviour, expected structure, and interpreted structure are more influential than they appear in the segments. While the interpreted function and behaviour, the expected function, and the depiction of behaviour are of less important than they appear in the segments.

Using the method depicted in section 4 Figure 5, the situated FBS processes were derived. For clarity and ease of analysis the reflection categories of processes were separated from the formulation and reformulation processes so that there is no overlapping of processes in any of the categories. Aggregating those meaningful processes into the basic eight design processes gives us a 92% coverage of all the derived processes as shown in Table 8. Compared to the original FBS there is an increase in the capture of the reformulations. The increase is most noticeable for Reformulation II (behaviour) for this protocol.

Table 7. Comparing the distribution of codes in segments and links

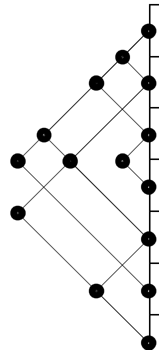
	Segments		Links	
R	7	1.6%	36	1.7%
F^i	8	1.8%	18	0.8%
Fe	0	0.0%	0	0.0%
Fe^i	9	2.0%	38	1.8%
B^i	125	27.9%	493	23.3%
B^e	13	2.9%	42	2.0%
Be^i	69	15.4%	396	18.7%
S^i	98	21.9%	485	22.9%
S^e	36	8.0%	145	6.9%
Se^i	83	18.5%	462	21.8%
Total	448	100.0%	2126	100.0%

Table 8. Percentages of situated FBS processes

8 FBS processes	Situated FBS processes	Percentages
Formulation	1, 2, 3, 10	3.4%
Synthesis	11	7.5%
Analysis	14	13.4%
Evaluation	15	4.6%
Documentation	12, 17, 18	8.1%
Reformulation I	6, 9, 13	31.7%
Reformulation II	5, 8, 19	21.6%
Reformulation III	4, 7, 16, 20	1.5%
Total		91.9%

There are 10 codes, so the possible combinations of processes will be 100. Fifty types of processes were recorded; 2.5 times more than the predetermined 20 situated processes. These occurred, in most of the cases, because of the granularity was too large to pick up the code(s) in between the segments. The derived process from 39 to 40 is S^i>S^e which is not one of the 20 processes. Segment 39 should be divided into three segments with "the sledge excellent" coded as Se^i, "so what did that generate then?" coded as either Be^i or Se^i and "(write sledge)" coded as S^e. This has been taken into account when grouping those 52 processes types of processes into the 20 situated FBS processes.

Again what are the other processes (8.1%)? The remaining eight percents contain processes like B^i>S^e and B^i>S^i. Figure 7 contains both examples, the derived process from segment 40 to 41 is an example of B^i>S^e. The processes from links between segments 42>43 and 40>43 are examples of B^i>S^i. In the first round of coding, segment 40 "the sledge manages to keep level by having quite a wide base" was code as B^i; in the second round it had been coded as Si. The final arbitrated code was B^i. It should contain two parts – the behaviour part of "keep level" and the structural part “wide base”. Segment 43 “unlike the set of skis” was also one of those disagreed code (B^i and S^i). The final coded was S^i but by carefully examining the context the analogy of “unlike” was both structural and behavioural; the structural analogy was “wide base” against “narrow”, the behaviour analogy was “force in the middle” versus on one leg. Figure 8 illustrates a proposed refinement of the segments from segment 40 to 43 together with the codes and linkograph. The first column in the table contains the segments with an alphabetic suffix added to those subdivided segments. The links were updated so that they corresponded to the processes of the ontology.



40a	S^i	J: the sledge by having quite a wide base
40b	B^i	J: manages to keep level
41a	Se^i	A:[interpret and expecting the “wide base” structure of a sledge]
41b	S^e	A: (write: wide base)
42	B^i	J: a main force in the middle
43a	S^i	J: unlike the set of skis [in terms of structure]
43b	B^i	J: unlike the set of skis [in terms of behaviour]

Figure 8. Re-segmenting the protocol with a finer grain

The missing processes (8.1%) were caused by the lack of experience in using this coding method; this includes choosing the correct granularity and making appropriate links. In this case some of the segments require a finer grain than was used. Further analysis and refinement is likely to resolve these “missing” processes as exemplified above.

6. Discussion

The FBS ontology denotes fundamental processes of designing that are general enough to embrace almost all design situations. As an ontology the ontological variables are disjoint with each other. Unlike most coding schemes, supported by available protocol analysis software, which allow overlapping of codes the ontological approach requires precise discernment of one code per segment. This clear distinction converts the protocol into unambiguous segments; it quantifies the amount of effort spent in relation to function, behaviour, or structure. The links not only provide a structural view of the processes but also locate the dominant codes and the frequency of each design transformation process. The nested representation of links, the linkograph, together with the FBS coded segments provide an opportunity to look into the design protocol not in a linear manner but as a network of processes. The study of the interaction among the FBS classes and processes may help to deepen the understanding of designing.

The use of the FBS ontology has been able to capture the design process semantics of this protocol. Of particular interest is that formulation/reformulation is the largest activity in terms of events and that the vast majority of reformulation is concerned with behaviour and structure. This maps well to our qualitative understanding of this session—generating novel ideas by analogy—and provides a quantitative measure of it. Despite the immaturity in the application of this FBS-based coding scheme it still accounts for 92% of all designing activities in this protocol.

The FBS-based coding scheme, subsumed in a more comprehensive scheme, has previously been used with individual designers (Gero and McNeill 1998). The method presented in this study had been used to investigate two architects designing face-to-face as compare to designing over the internet with 3D world (Kan and Gero 2008a). The distribution of the eight FBS processes of the face-to-face session and this session exhibit similar patterns, while the 3D world session looks very different. The results presented in this paper demonstrate that it can be used with a team of designers. The coding scheme does not require that any particular number of designers be involved. It is not limited to any particular communication channel.

Many coding schemes have been developed for use with design protocols. All such schemes are based on particular views of the activity of designing. Many of these schemes are unique to the data to which they are applied. This limits the applicability of the results obtained. Where more general codings have been attempted they still lack sufficient generality to allow them to be re-used in widely varying circumstances. It is claimed that the use of the FBS ontology and the situated FBS ontology provides a generally applicable coding basis that does not depend on any particular circumstance associated with any unique protocol.

Acknowledgements

This research is supported by an International Postgraduate Research Scholarship, University of Sydney.

References

Bartlett, F.C. (1932). Remembering: A Study in Experimental and Social Psychology. Cambridge University Press, Cambridge.

Clancey, W.J. (1997). Situated Cognition: On Human Knowledge and Computer Representations. Cambridge University Press, Cambridge.

Cross, N., Christiaans, H., & Dorst, K. (1996). Introduction: the Delft protocols workshop. In *Analysing Design Activity*, (Cross, N., et al., Eds.), pp. 1-14. John Wiley & Son, Chichester.

Dewey, J. (1896). The reflex arc concept in psychology. *Psychology Review 3*, 357-370.

Dewey, J. (1934). Art as Experience. Capricorn Books, New York.

Gero, J.S. (1990). Design prototypes: a knowledge representation schema for design. *AI Magazine 11(4)*, 26-36.

Gero, J.S., & Kannengiesser, U. (2004). The situated Function-Behaviour-Structure framework. *Design Studies* 25(4), 373-391.

Gero, J.S., & Maher M.L. (1991). Mutation and analogy to support creativity in computer-aided design. In *CAAD Futurees '91,* (Schmitt G.N. Ed.), pp. 241-249. ETH, Zurich

Gero, J.S., & McNeill, T. (1998). An approach to the analysis design protocols. *Design Studies 19(1)*, 21-61.

Goldschmidt, G. (1990). Linkography: assessing design productivity. In *Cyberbetics and System '90*, (Trappl, R., Ed.), pp. 291-298. World Scientific, Singapore.

Goldschmidt, G. (2001). Visual analogy. In *Cognition in Design Education,* (Eastman C., et al., Eds.), pp. 199-219, Elsevier, UK.

Kan, W.T., & Gero, J.S. (2007).Comparing entropy measures of idea links in design protocols, *AIEDAM 21(4)*, 367-377

Kan, W.T., & Gero, J.S. (2008). Acquiring information from linkography in protocol studies of designers, *Design Studies* (to appear)

Kan, W.T., & Gero, J.S. (2008a). A pilot study of using linkography and FBS ontology to study design protocols, *CAADRIA'08* (to appear)

McNeill, T., Gero, J.S., & Warren, J. (1998). Understanding conceptual electronic design using protocol analysis. *Research in Engineering Design 10(3)*, 129-140.

Schon, D.A., & Wiggins, G. (1992). Kinds of seeing and their functions in designing. *Design Studies 13(2)*, 135-156.