

A Discursive Approach to Understanding Dependencies between Design Acts

Ömer Akin, PhD, AIA

Professor of Architecture
Carnegie Mellon University, Pittsburgh PA, USA
omerakin@cmu.edu

Olaitan Awomolo, MS in AECM

PhD candidate, School of Architecture
Carnegie Mellon University, Pittsburgh PA, USA
oawomolo@andrew.cmu.edu

Abstract: *Design activity has a significant discursive component. Based on the well-established Hierarchical Task Analysis Method, which links human behavior to syntactic analysis and the hypothesis that design as a problem solving activity can be characterized by its discursive space, we attempt to identify dependencies between design features, defined by codifications based on pairings of nouns (tasks) and verbs (actions) in protocol data. This enables their integration into a complete solution, within a team design setting. Using the Service Learning dataset provided on the Purdue University Research Repository and focusing on the protocols corresponding to the requirement specification, preliminary design and design development stages of design delivery, we carried out macro-, midi- and micro-level analyses. In the macro-level data analysis, statistical tests showed significant correlation between major and minor nouns (tasks). In the midi-level we established similarities between the occurrence of nouns and verbs in protocols. We also observed that certain nouns were more prevalent during specific design stages. In the micro-level data analysis, we found correlations between nouns. Overall, the results show that design actions are anchored around a central task and discursive data can provide significant insight into the integration of successive design actions.*

Keywords: design integration, discursive analysis, codification taxonomy, requirement analysis, solution re-composition, team design

1. Background

1.1. Brief History

The quality and volume of research on design thinking is substantial. This interest grew out of early work on “design methodology” which almost exclusively considered *prescriptive* models of design. (Jones, 1963; Archer, 1979). The earliest works on *descriptive* accounts of design cognition were published in the late 1960s. Eastman is credited with conducting the first protocol analysis study of designers (1968). Simon is credited with developing the theory and methodology for the scientific study of design (1969). Almost concurrently, Schön had been developing an alternative approach to modeling design thinking based on a pragmatist framework (1983) borrowed from John Dewey’s theory of inquiry (1938).

The next generation of researchers planted the seeds for a mosaic of approaches to design thinking (Cross, 1984; Akin, 1986, 1998; Laurel, 2003). These approaches are responsible for the emergence of a plethora of models and methods used to investigate design thinking today. Significant contributions to this development came about through the special attention paid to cognition and computation (Gero, 1998), sketching and representations of the workflow in design protocols (Goldschmidt, 1997), analytical models of sketching in design (Suwa, 1997; Oxman; 1997) in the Deft Protocol Workshops [DTRS2] (Cross et.al., 1996) and Analyzing Design Meetings [DTRS7] (McDonnell and Lloyd, 2010).

Current research in the area of design thinking can be considered through a handful of basic questions. What is design? What is its connection to cognition? What methods are commonly used in design cognition? And, what models and constructs accurately capture the essence of design thinking research?

1.2. Design Thinking Research

Design has been characterized as an ill-defined process that aims “to devise courses of action in order to change the present conditions into desired ones” (Simon, 1983). This is a *positivist* point of view that considers design as problem solving and as the act of solving a Cartesian task (Akin 2006). Alternatively, it is seen as *reflection in action* where design is regarded as an evolutionary dialectic between action and reflection on the action, both in the design studio and the design office (Schön, 1983). This bifurcation in the philosophical basis of design is no different from the distinctions between alternative considerations of knowledge in the natural sciences – i.e., analytical and finite models versus synthetic and indeterministic ones.

Cognitive Psychology revolutionized the way we understand how humans think, learn, and behave. Scientists redefined the way they explained human processes through *behaviorism* or *specific actions* versus *cognition* or *mental representations*. Whereas the commonly used

techniques, such as reaction time studies, recall tasks, eye-fixation studies, recognition tasks, and ethnographic observations serve the field well-enough, protocol analysis has become one of the most powerful approaches to gathering rich and reliable data particularly in design fields.

In his own words, Smith describes what happens in a design space depicted in a design protocol as: “cognitive structures and processes moving between mental states, i.e., from an initial problem state to the goal or solution state” (Smith, 1995). The *state space formalism* is one of the most powerful metaphors to help represent, model, predict and control design outcomes. Within this domain of exploratory behavior the designer is perceived by the researcher as one building higher level complex processes upon primitive operations and using well-defined methods and techniques, like means-ends-analysis, hill climbing, heuristic problem solving, and so on. It is in the middle ground where these aggregated techniques meet the primitive operations that provide remarkable evidence about the nature of solo design (Gero, 1998; Goldschmidt, 2014) and team design (Stempfle, 2002).

Other paradigms of design have been proposed to explain phenomena uniquely, if not idiosyncratically, as “individual styles and representations” (Eisentraut and Gunther, 1997), “problem space and indeterminism” (Goldschmidt, 1997), “linguistic modelling” (McDonnell, 1997), “episodic data [knowledge] modelling” (Visser, 1996), and “SBF modelling” (Goel, Rugaber & Vattam, 2009)

1.3. Design as a Discursive Activity

The theoretical basis of the research in this study is that design has a significant discursive activity component and *protocol data* is the perfect medium to investigate its nature and role in solving design problems. Whilst, a preponderance of the protocol analysis research to date has focused on the investigation of strategies, knowledge, sketching, problem solving, problem discovery, creativity, team design, analogical reasoning, invariants between disciplines, role playing, and testing the validity of a variety of design models, studies of the discursive aspects of design behavior have also been prevalent and this will be our focus.

During the DTRS-2 Delft Protocols Workshop, Radcliffe used the Non-Numerical Unstructured Data Indexing Searching and Theorizing (NUD.IST) approach to analyse design protocols. His approach consists of (1) recording the design ideas based on the transcripts of protocol, (2) cross-linking the ideas, (3) constructing theories, and (4) generating detailed reports on statistical analysis. He reported his findings in terms of four knowledge types: declarative, procedural, situational, and strategic. The hypotheses

articulated in this study constitute a sound premise for discursive analysis of design protocols:

(Hypothesis 1) Design acts can be characterized by analyzing their discursive space

(Hypothesis 2) Argumentative acts contribute to the problem solving approach

(Hypothesis 3) Speech acts are linked to the character of each design step.

In the end, the paper concludes that “the design problem solving process can be characterized by the location of the discursive space” (Radcliffe, 1996). Researchers who conducted design protocol analysis for the proceedings of the DTRS-4 (Suwa, et.al., 1997) dissect protocols into types of actions (looking, drawing, perceiving, deriving meanings, and higher cognitive functions). This is an all-inclusive and well-developed construct used in design protocol analysis. It provides a top view of the cognitive processes as well as primitives of design action (Akin 1986).

In the publication edited by McDonnell and Lloyd (2009) that include papers presented at DTRS-7, two authors report findings on *linguistic analysis of design protocols*. Luck utilizes an Ethno Methodology and Conversational Analysis (EM/CA) technique (Luck, 2009), while Glock “investigates episodes and some aspects of design conversation” regarding design as a social, interactive, interpretative process (Glock, 2009). Luck states that talk is the equivalent of action in design, since the design space is cognitive in nature and it is navigated through thoughts or their equivalents in speech. Reification of design moves into the design state space produces models of design thinking workflows. Luck proposes an analysis method that regards these acts as sequence- rather than semantic-phenomenon. Only what is spoken matters, not the interpretation of the minds that have produced these utterances (Luck, 2009). Glocks’ analyses designing in terms of contexts and frames and “reconstructs how participants interactively construct meaning in the design process and to describe practices they employ in the process” (Glock, 2009)

At the expense of partially digressing from the domain of design, let us consider a well-developed, verified, and useful method that maps the world of human behaviour into syntactic analysis.

“Hierarchical task analysis (HTA) is a core ergonomics approach with a pedigree of over 30 years continuous use. ... It has been used for a range of applications, including interface design and evaluation, allocation of function, job aid design, error prediction, and workload assessment. Ergonomists are still developing new ways of using HTA which has assured the continued use of the approach for the foreseeable future.” (Stanton, 2005)

This view of the world of human enacted tasks, codifying, understanding, evaluating and manipulating them is one of the basic motivations in design as well. How far can we go into understanding design action, through a deterministic, syntactic, and analytical approach to

protocol data analysis? How much of the nature of design cognition can we capture by considering sequences of design activity defined by Verb-Noun-Pairs that correspond to each statement uttered in a protocol? We intend to demonstrate through our research the effectiveness of Verb-Noun-Pair analysis in design protocols.

2. Our Approach

2.1. Theoretical Framework

In this paper, discursive material found in design protocols will be analysed and described. As is the case in several of the sources cited earlier, design utterances are treated as reified data in the state space of sequential design thinking actions. We regard protocol transcriptions and their codification as the evidence that represent these sequential design acts. Like Radcliffe we assume that design activities “can be characterized by analysing their discursive space” (Radcliffe, 1996). Like Luck we regard this data as a sequential, temporal phenomenon (Luck, 2009). Intentionally, we do not elaborate our analysis and conclusions with added layers of semantic, interpretive, and visual-graphic data.

It is our contention that design protocols viewed in this rather “mechanistic” way will reveal patterns, or foot, hand and fingerprints that can help uncover new constructs about design activity workflow dependencies and integration behaviors in team design situations. This we believe will be feasible and yield even more robust results than that of codification and interpolations efforts, however meticulous, encumbered by the introduction of complex and sometimes ambiguous interpretation of data semiotics and design sketches. Thus our expectations, hypothesis if you will, are as follows:

[H1]: Design acts have dependencies when viewed in sequence highlighting moves that are pre- or post-requisites for other moves. This can be verified through statistical analysis of data sequences (Pearl, 2009).

[H2]: In design, features are decomposed into sub- or subservient features that are developed independently and then synthesized into wholes. This can be shown through episodic analysis of discursive data as was done in two previous studies (Akin, 1996; Akin, 2007).

Finally, the underlying assumption of our entire approach to protocol analysis in this study is expressed by the following axiom:

[Axiom 1]: Utterances codified as verb-noun pairings retain the essence of the discursive representation (Stanton, 2005) mined here from design protocol data. This is all-inclusive of the linguistic representations sufficient to investigate both H1 and H2.

2.2. Methods of Analysis

From all of the datasets provided on the Purdue University Research Repository, we decided to focus our attention on the Service Learning dataset consisting of eight protocol folders: 1-Partner Debrief #1, 2-Brainstorm Review, 3-Partner Debrief #2, 4-Advisor Debrief #1, 5-Advisor Debrief #2, 6-Partner Review, 7-Advisor Debrief #3, and 8-Final Review. The protocol folders followed a group of Service Learning students through their design task, which was to come up with a treehouse design for a camp for children with special needs. Apart from providing a connection to nature and access to an existing “zipline,” the design also had to incorporate the safety and special access needs of the campers. As the protocol segments in the Service learning dataset begin from the initial partner debrief meeting and go to the final review meeting, they provide the most comprehensive design process flow information among all of the DTRS-10 protocols.

After going through all the protocols in the dataset, we decided to codify and analyse the data in Protocol-1 Partner Debrief #1 (P-1), Protocol-2 Brainstorm Review (P-2), and Protocol-3 Partner Debrief #2 (P-3), only, as they appeared to be most representative of the three typical steps in design delivery workflow - requirement specification, preliminary design, and design development, respectively.

Our analysis has several major steps: codification, macro-level data analysis, mid-level data analysis, micro-level data analysis, investigation of process workflow dependencies, and investigation of design integration strategies.

Codification

We developed an *a priori* taxonomy for coding protocol data. This codification system was based on utterances, represented as Verb-Noun-Pairings that described a specific design activity. Nouns were defined as design tasks to be performed which could be tangible design objects or object requirements and verbs were the actions performed on these tasks. The latest evolution of this taxonomy is shown in Table 1.

This taxonomy went through several iterations as a function of the coder matching results we obtained. Two experienced coders [C1 and C2]ⁱ completed this step for all three protocols (P-1, P-2, and P-3) and went through several refinement cycles. Table 2 shows three of these cycles in succession. Each protocol was then broken up into utterance segments with each utterance segment assigned a verb-noun code according to the taxonomy. Utterances were either phrases or sentences that held a specific idea identified by a Verb-Noun-Pair.

Table 1. Final Noun-Verb Taxonomy used for codifying all three Protocols (P-1, P-2 and P-3)

| NOUN CLASSES | | s – Physical Environment | |
|--|-----------------------------------|--|--|
| | | s1: | Site description |
| do - Design Objects | | s2: | Site protection: tree protection |
| do1: | Ramp objects | s3: | Site characteristics: features on site |
| do2: | Deck objects | so - Specific-Object Requirement | |
| do3: | Treehouse objects | so1: | Treehouse requirements |
| do4: | Sensory/activity objects | so2: | Zip line requirements |
| do5: | Water feature objects | so3: | Ramp requirements |
| gp - General-Performance Requirement | | so4: | Wheelchair requirements |
| gp1: | Safety requirements | so5: | Observation deck |
| gp2: | Nature experience requirements | so6: | Water: waterslide, hot tub |
| gp3: | Activity requirements | so7: | Tire swing |
| gp4: | Cost | so8: | Fire-ring, fireplace |
| gp5: | Character of camp champ | so9: | Skylights |
| go - General-Object Requirement | | sp - Specific Performance Requirement | |
| go1: | Dimension requirements | sp1: | Protection requirements |
| go2: | Camper requirements | sp2: | Security/ Access requirements |
| go3: | Electricity requirements | sp3: | Structural requirements s-performance |
| i - Information or Statement | | u- People within facility excluding design team | |
| i1: | Introduce design team | u1: | Users-kids |
| i2: | Camp information: Camp operations | u2: | Users-staff |
| i3: | Process/meeting information | u3: | User-activity |
| i4: | Precedent information | VERB CLASSES` | |
| i6: | Mechanics of the requirements | I: | Introduce – make a general declaration |
| i7: | Mechanics of the design | Q: | Question – pose a question |
| ro – Requirement Options | | A: | Answer – respond to a Q |
| ro2: | Zip line options | C: | Confirm – affirm the correctness of |
| ro4: | Ramp options | E: | Elaborate – add new information |
| ro5: | Activity options | D: | Disqualify or delete information |
| NOUN CLASSES – continued on the right | | M: | Modify information |

Table 2. Consistency in Coding

| Data Coders | 1. V and N match | 2. V X-or N match | 3. Any match [1+2] | 4. No-match | 5. Total |
|-----------------------------|------------------|-------------------|--------------------|--------------|----------|
| 3/15/2014 Comparison | | | | | |
| C1 | 17 | 45 | 62 | 148 | 210 |
| C2 | 17 | 45 | 62 | 21 | 83 |
| % [C1 vs. C2] | %21 vs. %8 | %54 vs. %21 | %75 vs. %30 | %25 vs. % 70 | |
| 3/25/2014 Comparison | | | | | |
| C1 | 44 | 90 | 134 | 77 | 211 |
| C2 | 44 | 90 | 134 | 52 | 186 |
| % [C1 vs. C2] | %21 vs %24 | %43 vs %48 | %64 vs. %72 | %36 vs %28 | |
| 3/31/2014 Comparison | | | | | |

| | | | | | |
|--------------|------------|------------|-------------|------------|-----|
| C1 | 65 | 100 | 165 | 46 | 211 |
| C2 | 65 | 100 | 165 | 90 | 255 |
| % [C1 vs. C] | %31 vs %25 | %47 vs %39 | %78 vs. %65 | %22 vs %35 | |

After the first cycle we realised that the failure to code all utterance segments resulted in a high ‘no-match’ category. After the second cycle, we realised that there were differences in the use of the taxonomy by the coders: confusing the [E] explain-verb – to provide more information on an idea, with the [I] introduce-verb – to bring up a new idea; incorrect use of the [Q] question-verb – to pose a question and the [L]-link-verb – to connect with a different idea; and nesting verb codes such as [Q] question-verb and [A] answer-verb within one another. When these anomalies were corrected by both coders we achieved match between coders at %65-78. This range is due to the variance in the total number of utterance segments coded by each coder.

For the sake of consistency, one of the coders [C2] continued to code the remaining protocols (P-2, and P-3) with validation provided by the other coder [C1]. In this more advanced design stage, we discovered that subjects began to talk about *design objects* (included in Table 1). Further analysis steps we foresaw during this stage fits well within the *functional task analysis* approach (Adams, 2010) as adapted to discursive analysis in this study.

This codification and the rich results we mined here from the discursive aspects of the protocol data are based on [Axiom 1] that purports, utterances codified as Verb-Noun-Pairings “retain the essence of the discursive representation” (Stanton, 2005). This is all-inclusive of the linguistic representations sufficient to investigate [H1] and [H2].

Macro-level Data Analysis

In this analysis stage, we used descriptive statistics and means testing to ascertain at once the general stochastic parameters of the coded data and that our codification has captured the distinctions between the three protocols we included in our research scope. Through means testing we demonstrated that P-1 is distinct from the other two with a preponderance of dominant nouns and verbs characteristic of requirement specification ([I] introduce, [so] specific object requirement); and in turn the other two protocols P-2, and P-3 are also distinct from each other with a preponderance of dominant nouns and verbs ([I] introduce, [do] design object) and ([E] explain [C] confirm, [do] design object), characteristic of preliminary design and design development stages, respectively.

In this analysis stage, we also select the most prominent and detailed episodes that involved multiple or pairwise integration between the *major* noun-class and the other *minor* noun-classes to be analysed in the midi-and micro- level analyses stages.

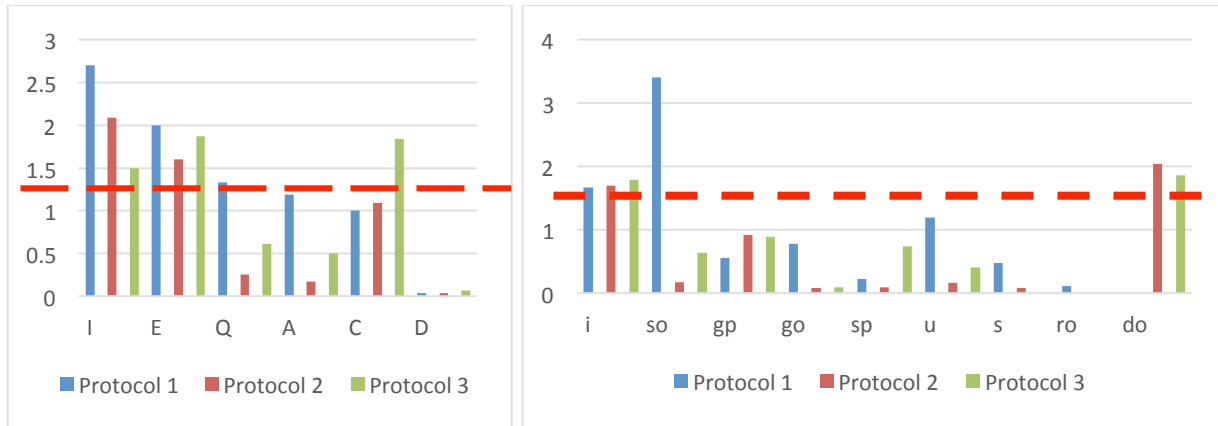


Figure 1. Frequency of Verbs (actions) and Noun Classes (tasks) per protocol

In aggregate, the Verbs that were dominant throughout the three protocols were: [I] introduce and [E] explain in P-1 (requirement specification) and P-2 (preliminary design), and [I] introduce, [E] explain and [D] disqualify in P-3 (design development) as shown in Figure 1. The aggregate frequency of the noun-classes showed that [i] information and [so] specific object requirement were dominant in P-1 (requirement specification) and [i] information and [do] design object were in P-2 (preliminary design) and P-3 (design development).

For each analysed protocol the dominant noun-class differed. For Protocol-1 [i] information, [so] specific object requirement and [sp] specific performance requirement were dominant; in Protocol-2, [i] information, [gp] general performance requirement and [do] design objects dominated, and in Protocol-3, [i] information, [gp] general performance requirement and [do] design objects were prevalent. In other words, i (information) was the consistent dominant noun-class in all three protocols. On the other hand, [gp] general performance requirement and [do] design objects, a pair of requirement-design nouns both essential for design actions, are the signatures for the preliminary and development design protocols (P-2 and P-3). Specification related discourse, subsumed in the [so] specific object requirement and [sp] specific performance requirement noun-classes, fit perfectly with the requirement specification protocol (P-1).

Midi-level data analysis

We ran analyses of variance (ANOVA) to see which Verb-Noun-Pair classes were statistically different between the three protocols. The results, in Table 3, show that for the [i] information, [gp] general performance requirement and [do] design noun classes, with significance values above 5% level ($p = 0.989$, $p = 0.700$ and $p = 0.058$, respectively), there was no statistically significant difference between the protocols while the rest of the noun classes, with significance values below 5% were different between protocols. For [I] introduce, [E] explain, [C] confirm and [D] decline verb classes, as significance values were

above 5% ($p = .092$, $p = .669$, $p = .051$, and $p = .486$, respectively), there was no statistically significant difference between protocols. In other words, verb-classes [I] introduce, [E] explain, [C] confirm and [D] played significant roles in each of the protocols.

We then carried out a multiple comparison analysis to identify which of the three protocols were responsible for the observed similarities and differences between verb- and noun-classes. The Tukey post-hoc test provided the detailed analysis we needed for the noun-classes and the verb-classes. The results for nouns are shown in Table 3 and those for verbs in Table 4. For the nouns [so] specific object requirement, [go] general object requirement, [u] users, [ro] requirement options and [do] design options, we found a difference between P-1 and P-2 and P-1 and P-3, but no difference between P-2 and P-3. This is primarily due to the fact that P-1 specifies initial requirements and does not deal with design objects, while P-2 and P-3 are deeply engaged in design objects; *namely ramp, deck, treehouse, sensory activity, and water feature objects*. The [sp] specific performance requirement noun-class [including the nouns, *protection, security-access, and structural requirements*] accounted for some differences between P-2 and P-3 but no difference between P-1 and P-3 and P-1 and P-2.

We attribute this to the fact that specification of requirements was further refined while advancing from preliminary design (P-2) to design development (P-3). Finally, for noun-classes [i] information and [gp] general performance requirements, there were no differences between any of the protocols, as information and general performance issues would be naturally prevalent in all.

Table 3. Means Testing of Noun Classes with Results from the Tukey Post Hoc test on the right

| | | Sum of Squares | df | Mean Square | F | Significance |
|----|-------------------|----------------|-----|-------------|--------|--------------|
| i | Between Protocols | .453 | 2 | .226 | .012 | .989 |
| | Within Protocols | 4637.765 | 236 | 19.652 | | |
| | Total | 4638.218 | 238 | | | |
| so | Between Protocols | 238.010 | 2 | 119.005 | 21.172 | .000 |
| | Within Protocols | 1326.534 | 236 | 5.621 | | |
| | Total | 1564.544 | 238 | | | |
| gp | Between Protocols | 2.977 | 2 | 1.489 | .357 | .700 |
| | Within Protocols | 984.738 | 236 | 4.173 | | |
| | Total | 987.715 | 238 | | | |
| go | Between Protocols | 11.658 | 2 | 5.829 | 4.943 | .008 |
| | Within Protocols | 278.300 | 236 | 1.179 | | |
| | Total | 289.958 | 238 | | | |
| sp | Between Protocols | 20.060 | 2 | 10.030 | 3.894 | .022 |
| | Within Protocols | 607.848 | 236 | 2.576 | | |
| | Total | 627.908 | 238 | | | |
| u | Between Protocols | 24.143 | 2 | 12.071 | 6.109 | .003 |
| | Within Protocols | 466.334 | 236 | 1.976 | | |
| | Total | 490.477 | 238 | | | |
| s | Between Protocols | 4.658 | 2 | 2.329 | 6.126 | .003 |
| | Within Protocols | 89.727 | 236 | .380 | | |
| | Total | 94.385 | 238 | | | |
| ro | Between Protocols | .296 | 2 | .148 | 4.026 | .019 |
| | Within Protocols | 8.667 | 236 | .037 | | |
| | Total | 8.962 | 238 | | | |
| do | Between Protocols | 95.040 | 2 | 47.520 | 2.877 | .058 |
| | Within Protocols | 3897.395 | 236 | 16.514 | | |
| | Total | 3992.435 | 238 | | | |

| Noun-class | Significance between protocols | | | |
|------------|--------------------------------|-------|-------|-------|
| | Protocols | 1 | 2 | 3 |
| i | Protocols | 1 | 2 | 3 |
| | 1 | 1 | 0.999 | 0.992 |
| | 2 | 0.999 | 1 | 0.99 |
| | 3 | 0.992 | 0.99 | 1 |
| so | Protocols | 1 | 2 | 3 |
| | 1 | 1 | 0 | 0 |
| | 2 | 0 | 1 | 0.359 |
| | 3 | 0 | 0.359 | 1 |
| gp | Protocols | 1 | 2 | 3 |
| | 1 | 1 | 0.679 | 0.756 |
| | 2 | 0.679 | 1 | 0.995 |
| | 3 | 0.756 | 0.995 | 1 |
| go | Protocols | 1 | 2 | 3 |
| | 1 | 1 | 0.007 | 0.015 |
| | 2 | 0.007 | 1 | 0.999 |
| | 3 | 0.015 | 0.999 | 1 |
| sp | Protocols | 1 | 2 | 3 |
| | 1 | 1 | 0.92 | 0.326 |
| | 2 | 0.92 | 1 | 0.016 |
| | 3 | 0.326 | 0.016 | 1 |
| u | Protocols | 1 | 2 | 3 |
| | 1 | 1 | 0.002 | 0.437 |
| | 2 | 0.002 | 1 | 0.043 |
| | 3 | 0.437 | 0.043 | 1 |
| s | Protocols | 1 | 2 | 3 |
| | 1 | 1 | 0.007 | 0.002 |
| | 2 | 0.007 | 1 | 0.617 |
| | 3 | 0.002 | 0.617 | 1 |
| ro | Protocols | 1 | 2 | 3 |
| | 1 | 1 | 0.017 | 0.03 |
| | 2 | 0.017 | 1 | 1.00 |
| | 3 | 0.03 | 1.00 | 1 |
| do | Protocols | 1 | 2 | 3 |
| | 1 | 1 | 0.047 | 0.11 |
| | 2 | 0.047 | 1 | 0.952 |
| | 3 | 0.11 | 0.952 | 1 |

For the verb-classes, [Q] question, [A] answer, [I] introduce, [E] explain, [C] confirm and [D] disqualify, there were no significant differences between their use in the three protocols, which means that, throughout the three protocols, the verb (action) classes were basic enough to pair up with all of the noun-classes. This was predicted at the time the taxonomy was developed. However, the difference between [Q] question and [A] answer verb-classes was surprising. We expected that they would go together through thick and thin; but this was not the case. -

If a question [Q] is asked for a certain noun class, say [so] specific object requirement, the answer provided could include not just the [so] specific object requirement noun class, but could go on to include other noun classes as [do] design object and [u] users. For example, the excerpt below was in response to a [do] design object question

So, I guess that first idea was kind of like, ah, I was thinking of, of a ramp when we talked about, um,...like using it kind of like as viewpoints. So I just kind of had like a little extra area like where like the ramp does branch off ...and it's like a separate deck. And so like if people do like just like wanna go up there just to stand on it, they're not like blocking traffic, ...like with their even if it is like an extra-wide ramp. So that was just one idea.

The answer provided then went on to describe the design object itself and introduce specific requirements for a view and physical dimensions.

Table 4. Means Testing of Noun Classes with Results from the Tukey Post Hoc test on the right

| | | Sum of Squares | df | Mean Square | F | Sig. |
|---|-------------------|----------------|-----|-------------|--------|------|
| I | Between Protocols | 32.023 | 2 | 16.011 | 2.408 | .092 |
| | Within Protocols | 1568.939 | 236 | 6.648 | | |
| | Total | 1600.962 | 238 | | | |
| E | Between Protocols | 5.811 | 2 | 2.906 | .402 | .669 |
| | Within Protocols | 1703.963 | 236 | 7.220 | | |
| | Total | 1709.774 | 238 | | | |
| Q | Between Protocols | 28.206 | 2 | 14.103 | 11.910 | .000 |
| | Within Protocols | 279.459 | 236 | 1.184 | | |
| | Total | 307.665 | 238 | | | |
| A | Between Protocols | 24.834 | 2 | 12.417 | 17.493 | .000 |
| | Within Protocols | 167.518 | 236 | .710 | | |
| | Total | 192.351 | 238 | | | |
| C | Between Protocols | 29.228 | 2 | 14.614 | 3.007 | .051 |
| | Within Protocols | 1147.081 | 236 | 4.861 | | |
| | Total | 1176.310 | 238 | | | |
| D | Between Protocols | .064 | 2 | .032 | .724 | .486 |
| | Within Protocols | 10.430 | 236 | .044 | | |
| | Total | 10.494 | 238 | | | |
| M | Between Protocols | .000 | 2 | .000 | . | . |
| | Within Protocols | .000 | 236 | .000 | | |
| | Total | .000 | 238 | | | |

| Verbs | Significance between protocols | | | |
|-------|--------------------------------|-------|-------|-------|
| | Protocol | 1 | 2 | 3 |
| I | Protocol | 1 | 2 | 3 |
| | 1 | 1 | 0.496 | 0.1 |
| | 2 | 0.496 | 1 | 0.26 |
| | 3 | 0.1 | 0.26 | 1 |
| E | Protocol | 1 | 2 | 3 |
| | 1 | 1 | 0.757 | 0.976 |
| | 2 | 0.757 | 1 | 0.766 |
| | 3 | 0.976 | 0.766 | 1 |
| Q | Protocol | 1 | 2 | 3 |
| | 1 | 1 | 0 | 0.011 |
| | 2 | 0 | 1 | 0.062 |
| | 3 | 0.011 | 0.062 | 1 |
| A | Protocol | 1 | 2 | 3 |
| | 1 | 1 | 0 | 0.001 |
| | 2 | 0 | 1 | 0.021 |
| | 3 | 0.001 | 0.021 | 1 |
| C | Protocol | 1 | 2 | 3 |
| | 1 | 1 | 0.979 | 0.212 |
| | 2 | 0.979 | 1 | 0.053 |
| | 3 | 0.212 | 0.053 | 1 |
| D | Protocol | 1 | 2 | 3 |
| | 1 | 1 | 0.999 | 0.751 |
| | 2 | 0.999 | 1 | 0.466 |
| | 3 | 0.751 | 0.466 | 1 |

Micro-level data analysis

At this data analysis level, we get a good, up-close look at the architecture of the design process. To achieve this, we decided to further parse the coded protocol data into smaller modules, called episodes. Episodes were defined as sequences of utterances that represent a dominant coded token, in this case the noun-classes. Based on the results obtained in the midi-level analysis, where most of the verb categories showed no significant difference between protocols, we decided to forego a breakdown based on verb-classes since they did

not show dominant characteristics correlated with the three protocols. We decomposed the protocols into series of episodes based on the episode definition criteria shown in Table 5. This strategy yielded a total of 239 episodes, 72 in P-1, 156 in P-2, and 70 in P-3. Figures 2, 3 and 4 show the number of noun-classes per episode.¹

Table 5. Rules of Episode Definition

| Rules of Episode Definition | |
|-----------------------------|---|
| 1. | An episode must have a major noun-class [more than 50% among all noun-classes it contains]. |
| 2. | An episode must start with an instance of the major noun-class |
| 3. | An episode cannot contain a minor noun-class more numerous than 50% of the major noun-class. |
| 4. | An episode cannot end with a sequence of noun-class instances that constitute a new episode. |
| 5. | A sequence of noun-class instance consists of two or more consecutive ones in the same class. |

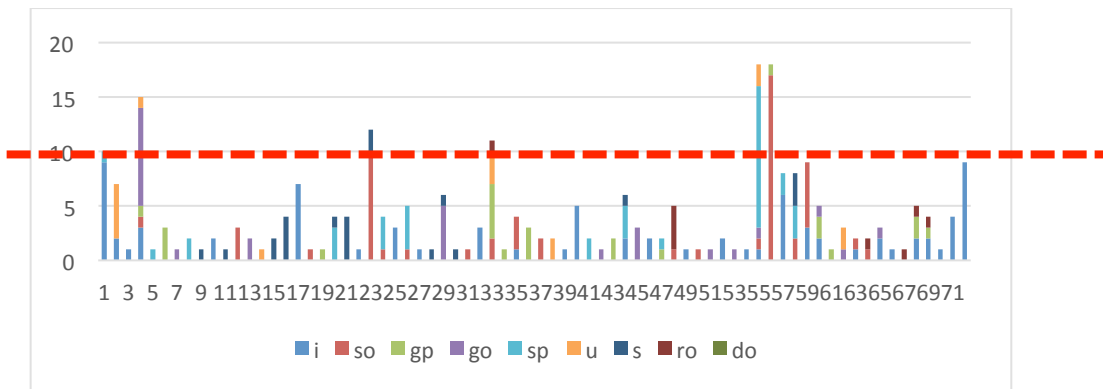


Figure 2. Frequency graph showing noun-classes per episode Partner Debrief Session

¹ The role these noun-classes play in bridging between the three episodes was considered. Noun-classes **i** and **gp** have means that are not significantly different between all three protocol pairs. Noun-classes **so**, **go**, **ro**, **s** and **do** are so for protocols -2 and -3, only; finally, **u** for protocols -1 and -3, and **s** for -2 and -3.

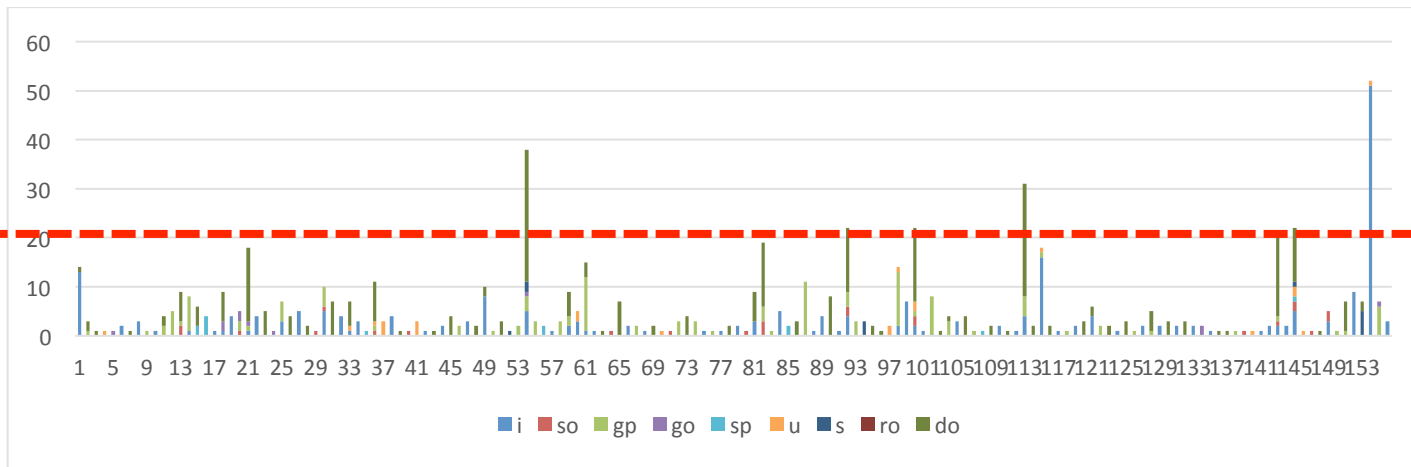


Figure 3. Frequency graph showing noun-classes per episode in the 02 Partner Debrief Session

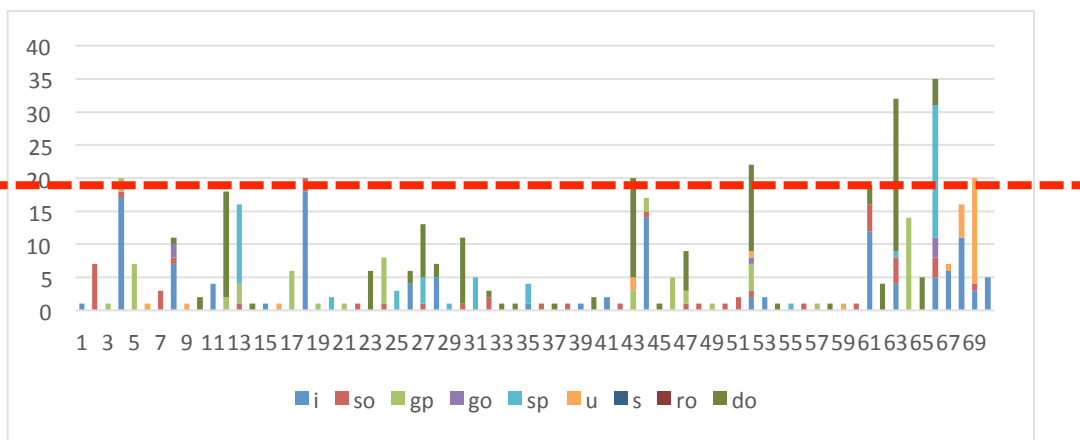


Figure 4. Frequency graph showing noun-classes per episode in the 03 Partner Debrief Session

When an episode contains multiple noun-classes with one dominant noun class and several minor, supporting noun classes, it becomes a candidate for exemplifying *design integration*. In earlier research studies (Akin, 2001, 2009 [DTRS7]) design protocol data contained episodes with evidence of design integration for several physical features, like “form,” “construction details,” “cost,” “ecology,” and “site navigation.” By virtue of the fact that various combinations of these features were included in singular episode modules, they were processed concurrently. We use this as the basis for identifying design integration patterns. Thus, in cases where an episode contains *major* noun-classes, the possibility of integration between the *major* noun-class and several *minor* noun-classes is highly likely.

We used an arbitrary criterion of about 50% of the highest frequency to select the top 5 episodes as episodes we would study up close. These criteria are shown as horizontal lines in Figure 2, Figure 3 and Figure 4. For P-1 we picked episodes numbered 4, 23, 33, 55, and 57. For P-2 we picked episodes numbered 54, 92, 100, 113, and 154. For P-3 we picked

episodes numbered 52, 61, 63, 66 and 69. The top five episodes from the three protocols are shown in Table 8. Each episode contains multiple noun-classes, IDs and frequencies of which are shown in the columns of each episode. Note that noun-classes with the highest percentage of occurrence are the major noun-classes for that episode and are highlighted.

Table 6. Frequency Tables showing the noun classes in the major episodes per protocol

| Episode Analysis | | | | | | | | | | | | | | |
|------------------------|-----------|------------|------------------------|-----------|------------|-------------------------|-----------|------------|-------------------------|-----------|------------|-------------------------|-----------|------------|
| P-1: EPISODE 5 | | | P-1: EPISODE 23 | | | P-1: EPISODE 33 | | | P-1: EPISODE 55 | | | P-1: EPISODE 57 | | |
| A4 | f | % | A23 | f | % | A33 | f | % | A55 | f | % | A57 | f | % |
| i | 2 | 12.5 | s | 1 | 8 | u | 2 | 12.5 | gp | 1 | 3 | u | 2 | 6 |
| so | 10 | 62.5 | sp | 1 | 8 | i | 13 | 81.25 | ro | 3 | 9 | go | 1 | 3 |
| gp | 3 | 18.75 | so | 9 | 76 | sp | 1 | 6.25 | so | 19 | 70 | so | 26 | 73 |
| u | 1 | 6.25 | u | 1 | 8 | Σ | 16 | 100 | i | 3 | 9 | 1 | 4 | 12 |
| Σ | 16 | 100 | Σ | 12 | 100 | | | | go | 3 | 9 | gp | 2 | 6 |
| | | | | | | | | | Σ | 29 | 100 | Σ | 35 | 100 |
| P-2: EPISODE 54 | | | P-2: EPISODE 92 | | | P-2: EPISODE 100 | | | P-2: EPISODE 113 | | | P-2: EPISODE 154 | | |
| B54 | f | % | B92 | f | % | B100 | f | % | B113 | f | % | B154 | f | % |
| gp | 6 | 12.5 | gp | 4 | 12 | gp | 3 | 14 | gp | 1 | 5 | sp | 1 | 5 |
| go | 1 | 2 | do | 25 | 74 | i | 15 | 71 | do | 13 | 62 | go | 2 | 9 |
| do | 31 | 65 | i | 5 | 14 | u | 1 | 5 | i | 6 | 28 | so | 2 | 9 |
| u | 1 | 2 | Σ | 34 | 100 | do | 2 | 10 | so | 1 | 5 | do | 11 | 50 |
| s | 3 | 6 | | | | Σ | 21 | 100 | Σ | 21 | 100 | u | 2 | 9 |
| i | 6 | 12.5 | | | | | | | | | | s | 1 | 5 |
| Σ | 48 | 100 | | | | | | | | | | i | 3 | 13 |
| | | | | | | | | | | | | Σ | 22 | 100 |
| P-3: EPISODE 52 | | | P-3: EPISODE 61 | | | P-3: EPISODE 63 | | | P-3: EPISODE 66 | | | P-3: EPISODE 69 | | |
| C52 | f | % | C61 | f | % | C63 | f | % | C66 | f | % | C69 | f | % |
| go | 1 | 5 | so | 4 | 19 | do | 25 | 73 | do | 25 | 73 | u | 16 | 80 |
| gp | 4 | 18 | i | 12 | 57 | i | 4 | 12 | i | 4 | 12 | i | 3 | 15 |
| do | 13 | 59 | do | 5 | 24 | sp | 1 | 3 | sp | 1 | 3 | so | 1 | 5 |
| i | 2 | 9 | Σ | 21 | 100 | so | 4 | 12 | so | 4 | 12 | Σ | 20 | 100 |
| so | 1 | 5 | | | | Σ | 34 | 100 | Σ | 34 | 100 | | | |
| u | 1 | 4 | | | | | | | | | | | | |
| Σ | 22 | 100 | | | | | | | | | | | | |

With this kind of fine grain data analysis, we can see if these coded tokens describe “design acts that have dependencies ... highlighting moves that are pre- or post-requisites for other moves” [H1]; as well as, describe the blow by blow process of integrating multiple design problem aspects, in our case, the noun-classes [H2].

Dependency Analysis [Hypothesis 1]

To test “design acts that have dependencies ... highlighting moves that are pre- or post-requisites for other moves” [H1], we constructed the null hypothesis [h0]: the dominant noun-class is dependent on the supporting noun-classes within each episode; and the alternative hypothesis [h1]: the dominant noun-class is not dependent on the supporting noun-classes within each episode. To test [h0], we ran a Multivariate Regression Analysis of the overall intra-protocol dependencies between noun-classes. The regression analysis showed significant R-square values between all major and minor noun-classes in each episode (Table 7). This suggests that for the episodes included in the analysis the major noun-classes were dependent on the minor noun-classes.

Table 7. Regression Analysis (R-square values) between major and minor noun-classes in each episode

| Protocol-1 [P-1]: Requirement Specification | | | | | |
|---|--|--|-------|-------|---------------|
| Episode | Dependent Variable [major noun-class] | Independent Variable [minor noun-classes] | R | R. sq | Adjusted R.sq |
| 4 | so | u, gp, i | 0.893 | 0.798 | 0.646 |
| 23 | so | u, sp, s | 0.933 | 0.871 | 0.677 |
| 33 | i | sp, u | 0.919 | 0.844 | 0.687 |
| 55 | so | go, gp, i, ro | 0.696 | 0.484 | 0.255 |
| 57 | so | gp, go, u, i | 0.727 | 0.529 | 0.214 |
| Protocol-2 [P-2]: Preliminary Design | | | | | |
| Episode | Dependent Variable [major noun-class] | Independent Variable [minor noun-classes] | R | R. sq | Adjusted R.sq |
| 54 | do | i, u, go, s, gp | 0.778 | 0.605 | 0.506 |
| 92 | do | i, gp | 0.682 | 0.465 | 0.376 |
| 100 | i | do, u, gp | 0.636 | 0.405 | -0.042 |
| 113 | do | so, gp, i | 0.657 | 0.432 | 0.006 |
| 154 | do | i, s, u, gp, sp, so | 0.822 | 0.676 | 0.432 |
| Protocol-3 [P-3]: Design Development | | | | | |
| Episode | Dependent Variable [major noun-class] | Independent Variable [minor noun-classes] | R | R. sq | Adjusted R.sq |
| 52 | do | u, so, go, i, gp | 0.715 | 0.512 | -0.098 |
| 61 | i | do, so | 0.703 | 0.494 | 0.292 |
| 63 | do | so, i, sp | 0.801 | 0.642 | 0.489 |
| 66 | sp | go, so, do, i | 0.537 | 0.288 | -0.067 |
| 69 | u | so, i | 0.73 | 0.532 | 0.298 |

To determine the extent of dependency between individual noun-classes, we estimated Correlation Coefficients between them (Table 8). These revealed detailed information about the extent of dependency between individual noun-classes. The correlation coefficients shown in Table 8 clearly echo the results of the R-squared test. For example in P-1- Episode 23, the major noun-class, [so] specific object requirements is significantly

correlated with all three minor noun-classes: [s] site, [sp] specific performance requirements, and [u] user requirements (Table 8 Episode-23). In P-2- Episode 92 where design decisions are being made on the basis of requirements specifications developed in P-1, the major noun-class, [do] design objects is significantly correlated with the [gp] general performance requirements minor noun-class. Finally, in P-3 - Episode 61 we observe how major noun-class, [do] design objects is significantly correlated with the minor noun-class, [i] information, since this is the design development protocol and information interacting with design objects is all that is needed.

Table 8. Correlation Coefficients between major and minor noun-classes in each episode

| Protocol-1 [P-1]: Requirement Specification | | | | |
|---|--------|--------|--------|--------|
| Episode 23 | s | sp | so | u |
| s | 1 | -0.2 | -0.417 | -0.2 |
| sp | -0.2 | 1 | -0.417 | -0.2 |
| so | -0.417 | -0.417 | 1 | -0.417 |
| u | -0.2 | -0.2 | -0.417 | 1 |
| Protocol-2 [P-2]: Preliminary Design | | | | |
| Episode 92 | gp | do | i | |
| gp | 1 | -0.407 | -0.337 | |
| do | -0.407 | 1 | -0.378 | |
| i | -0.337 | -0.378 | 1 | |
| Protocol-3 [P-3]: Design Development | | | | |
| Episode 61 | so | i | do | |
| so | 1 | -0.32 | -0.365 | |
| i | -0.32 | 1 | -0.466 | |
| do | -0.365 | -0.466 | 1 | |

An inclusive view of all significant correlations between noun-classes for the three protocols and the cumulative of all three is included in Figures 5, 6, 7 and 8. In P-1, (Figure 5) correlations [i]-[so], [i]-[gp], [i]-[go], [i]-[u], [so]-[gp], [so]-[go], [so]-[u], [gp]-[go], [gp]-[u], and [sp]-[u] are in the upper half of the range for all correlations. Here we see the characteristic dominance of performance specifications and information based noun-classes, for P-1. In P-2, (Figure 6) correlations [i]-[gp], [i]-[u], [i]-[do], [gp]-[do], and [u]-[do] occupy the upper half of the range for all correlations. This is characteristic dominance of the collaboration of information specifications and design in the design development related noun-classes, in P-2. Finally, information and specific object requirement [i]-[so], information and design objects [i]-[do], and specific object requirement and design objects [so]-[do] are in the upper half of the correlations in design development protocol, P-3.

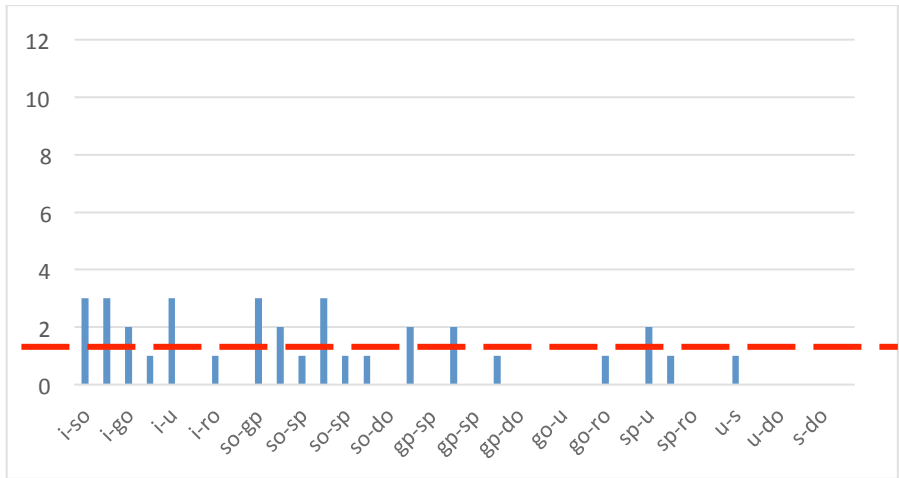


Figure 5. Frequency graph showing correlation entries per noun-class for Protocol-1

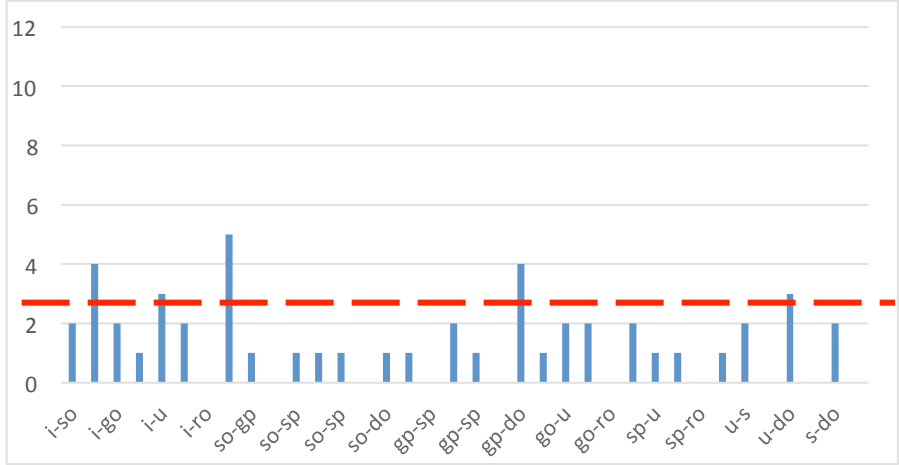


Figure 6. Frequency graph showing correlation entries per noun-class for Protocol-2

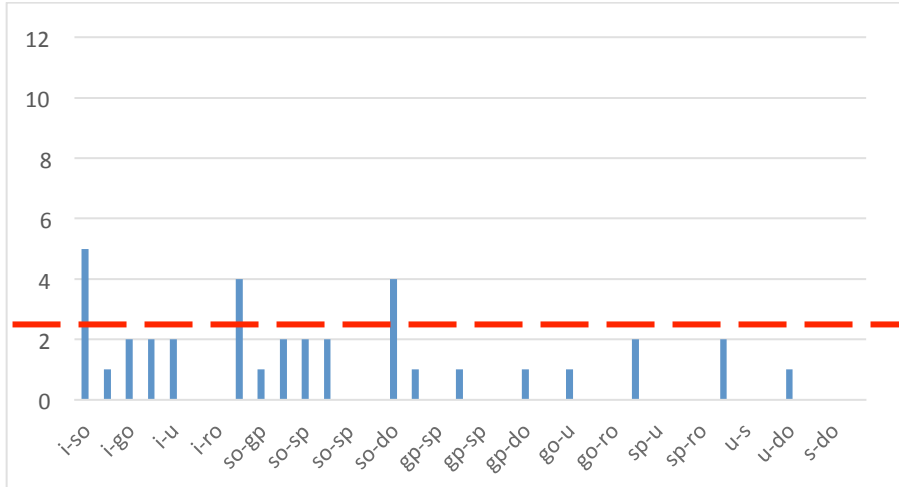


Figure 7. Frequency graph showing correlation entries per noun-class for Protocol-3

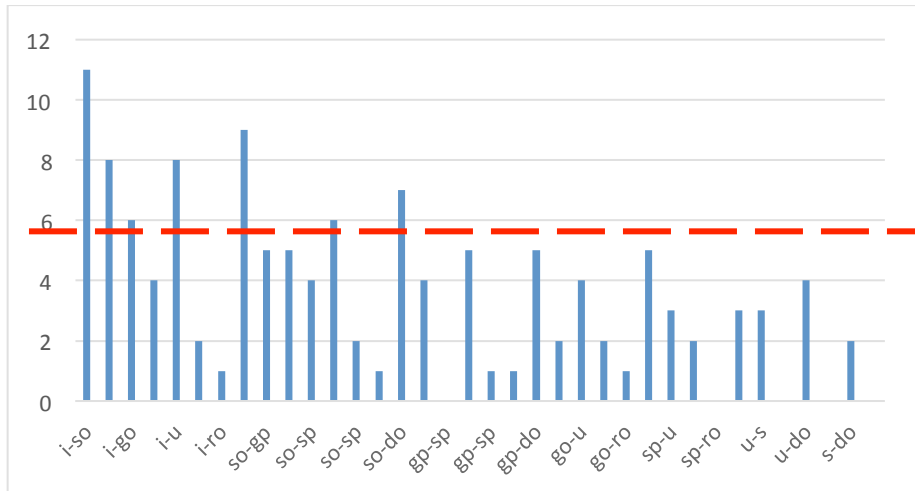


Figure 8. Frequency graph showing cumulative correlation entries per noun-class

Design Integration Analysis [Hypothesis 2]

Out of the 15 protocol episodes we studied in detail we chose P-2- Episode 92as the illustrative example for discourse-based design integration. The top half of Table 9 shows the raw data, it's coding and corresponding noun tokens. Here we see the subjects (in the first column) discussing the design features to be included in the design of the treehouse [do.1]. Information from the earlier discussions in the process meetings and precedents are integrated [i.3] and [i.4]. The main requirement used deal with nature experiences [gp.2]. Through this process of integrating diverse sources of knowledge the tree house is embellished with bird-feeders, portholes/retractable roof features, among others.

The Median Analysis graph shows how control of design, despite the participation of multiple “designers” is anchored around the major noun-class [do] deign objects and the arrows indicate intermittent shifts in focus that illustrate its integration with the other minor noun-classes (information, and general performance specification). The micro level analysis part of the table shows in specific all of the design features that arise from the interaction of these noun-classes:

- ship-like tree house
- observation deck
- a crow's nest
- retractable roof
- flipped or port-hole window
- bird feeders
- viewing through the window
- flat portholes

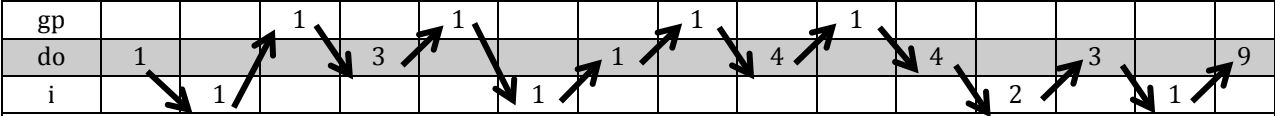
- bubble-inward portholes with feeders
- dome shaped with transparent base roof
- a rotating roof
- open a section of the house
- regulate sunlight

Table 9. Micro Analysis of the Design Integration Process in EPISODE-92

| Protocol Transcription and Assigned Codes | | | |
|---|--|---------|-------------------------|
| Subject | Discourse [utterance] | Code | Noun Tokens |
| M1 | Yeah, the main tree house area, and like maybe – like when we were talking with the kids with like the cards, a lot of the things they liked about like that one ship tree house, or like the other one, so they have like a place to – | I(do.3) | treehouse_objects |
| | | I(i.4) | precedent_information |
| | | I(gp.2) | nature_exp_requirements |
| 0:37:00 | <i>timestamp</i> | | |
| | go up high if they wanted to and look out. So maybe like having an observation – like an obso- – like an – observation deck like up high or whatever, they could go to. | I(do.2) | deck_objects |
| | Like a crow's nest, but like having it like close in with glass, even on the top so they can see out like all the sides. | E(do.3) | treehouse_objects |
| N1 | That'd be cool. | C(do.3) | treehouse_objects |
| M1 | Um, and then my second one is kind of like nature, but like now I kinda feel like it's – I don't know. Just the way I was thinking at the time is having like a retractable roof on like a pulley system from like down below where they can pull the roof open to stargaze at night, but if the window gets bad, they can like undo it more and like flip it back where it was. And then the last one is just having like port hole windows, like at different levels where like everybody at different heights could see out – | I(gp.2) | nature_exp_requirements |
| | | I(i.3) | process_meeting_info |
| | | I(do.3) | treehouse_objects |
| | | I(gp.2) | nature_exp_requirements |
| | | E(do.3) | treehouse_objects |
| | | E(do.3) | treehouse_objects |
| N1 | I like that. | C(do.3) | treehouse_objects |
| M1 | – like different ways. And maybe even like attach them to bird feeders, too, though, so they can have like different specified areas to see just nature up close. | I(gp.2) | nature_exp_requirements |
| C1 | And you could have those even like outside like you said, but open – | E(do.3) | treehouse_objects |
| M1 | Yeah. Like they're really neat. They, they come in bird feeders. They make like a nest that like you can just see the inside and it's closed on the outside, so like they're not like nervous about what's going on – | C(do.3) | treehouse_objects |
| | | E(do.3) | treehouse_objects |
| 0:38:00 | <i>timestamp</i> | | |
| | outside your window, just see. | E(do.3) | treehouse_objects |
| J1 | I have – | I(i.3) | process_meeting_info |
| S1 | Sorry. I, I had some ideas to add onto that like could make _____. And so just flat portholes, you can make them bubble inward, and then you can put a feeder on the inside of the bubble, and | I(i.3) | process_meeting_info |
| | | I(do.3) | treehouse_objects |
| | | E(do.3) | treehouse_objects |

| | | | |
|-----------|---|---------|----------------------|
| | then hopefully, the birds would come in and it would be inside of the tree house and – | | |
| <i>N1</i> | That's cool. I like that. | C(do.3) | treehouse_objects |
| <i>J1</i> | I had two ideas. The first one is like the top is like, ah, dome shape roof, and on top, it's like a one section is ch-, ah, like, actually, the base is totally transparent, and on top of that, has three sections, which is solid. So the roof is rotating to change the transparency of the, the roof. | I(i.3) | process_meeting_info |
| | | I(do.3) | treehouse_objects |
| | | E(do.3) | treehouse_objects |
| <i>M1</i> | You mean like an open one? | Q(do.3) | treehouse_objects |
| <i>J1</i> | Yeah, like – | A(do.3) | treehouse_objects |
| <i>M1</i> | Like you could rotate – | E(do.3) | treehouse_objects |
| <i>J1</i> | Changing the rope, opening section of the house. | E(do.3) | treehouse_objects |
| <i>M1</i> | Cool. | C(do.3) | treehouse_objects |
| <i>J1</i> | And, and also they can block it if they want sunshine to come in. And this one is just for the ramp side here, which can have some like branches cover around – | E(do.3) | treehouse_objects |
| | | I(do.1) | treehouse_objects |

Median Level Analysis



intermittently the focus of the process going back to: general performance requirement [gp]; and basic information [i] with specific purposes expressed as “a place to go up and look out”; “emulating nature”; “stargazing at night”;

Micro Level Analysis

Dominated by treehouse objects [do.3] with one or two visits to ramp objects [do.1] and deck objects [do.2]. These reveal a transformation of the design from-to: ship tree house → observation deck → a crow’s nest → retractable roof → flip the window back → port-hole windows → bird feeders → view outside your window → flat portholes → make windows bubble inward with feeder → dome shaped roof → with transparent base → rotating roof → open section of the house → regulate sunlight.

Taken as a whole the Maxi-, Midi-, and Micro-analysis results are expected to provide the right evidence for gaining insight into how the nature of coded utterances (nouns and verbs) reveal the dependencies between these utterances [H1] and design integration moves [H2] viewed solely from the standpoint of discursive data. We hope that the description provided above shows the consistent pre- and post-connectivity between each verb- and noun-class and can lead to a more palpable (less abstract and quantitative) understanding of the design integration process.

3. Results and Discussion

Initially, we tried to develop a reliable coding method with two experienced coders. After several tries the consistency between the coders increased to 65%. When we took into account the variance in deciding to code an utterance or not, this accuracy increased to 78%. We consider this an adequate level of accuracy between coders. To improve the consistency of coding between protocol segments to nearly 100% we relied on the coding of one coder in conducting our subsequent analysis.

Based on [Axiom 1] we defined utterances codified as verb-noun pairings, and stated that they “retain the essence of discursive representation” (Stanton, 2005) in design protocols. The codification and the rich results we obtained from the discursive aspects of the protocol data are all-inclusive of the linguistic representations we consider *sufficient* to investigate our hypotheses [H1] and [H2].

We focused our research on three protocol segments. Primarily because of the diversity of design process flows they represent. Protocol-1 P-1 is distinct from the other two with a preponderance of dominant nouns and verbs characteristic of *requirement specification*; and in turn the other two protocols P-2, and P-3 are also distinct from each other with a preponderance of dominant nouns and verbs characteristic of *preliminary design* and *design development*, respectively.

When considered in aggregate, the Verbs that were dominant throughout the three protocols are: [I] introduce and [E] explain in P-1 and P-2, and [I] introduce and [E] explain and [D] disqualify in P-3. The aggregate frequency of the noun-classes showed that [i] information and [so] specific object requirement were dominant in P-1 while [i] information and [do] design objects were in P-2 and P-3.

When we disaggregated the data we found that the role of the noun-verb classes was further refined. In other words, [i] information was the consistent dominant noun-class in all three protocols. On the other hand, [gp] general performance requirement and [do] design objects, a pair of requirement-design nouns both essential for design actions, provided the signature discourse for the preliminary and development design protocols P-2 and P-3. Specification related discourse, subsumed in the [so] specific object requirement and [sp] specific performance requirement noun-classes, fit perfectly with the requirement specification protocol P-1.

We tested statistical significance of the differences between the three protocols in terms of verb- and noun-classes. Analysis of Variance (ANOVA) showed that noun-classes [i]information, [gp] general performance requirement, and [do] design object played significant roles in each of the protocols, while, verb-classes [I] introduce, [E] explain, [C]

confirm and [D] disqualify played significant roles in each of the protocols, eliminating [M] modify, [Q] question and [A] answer from contention.

The multiple comparison Tukey Post-Hoc analyses showed which of the specific protocols were responsible for the similarities and differences of the verb-, noun-classes. Results indicated that P-1 specifies requirements and does not deal with design objects, while P-2 and P-3 are deeply engaged in design objects; namely ramp, deck, treehouse, sensory activity, and water feature objects. The [sp] specific performance requirements noun-class [including the nouns, protection, security-access, and structural requirements] accounted for some differences between P-2 and P-3 but no difference between P-1 and P-3 and P-1 and P-2, owing to the fact that specification of requirements was further refined while advancing from preliminary design P-2 to design development P-3. Finally, for noun-classes [i] information and [gp] general performance requirement, there were no differences between any of the protocols, as information and general performance issues would be naturally prevalent in all. For all verb-classes there were no significant differences between their use in the three protocols, which means that, throughout the three protocols, the verb (action) utterances were basic enough to pair up with all of the noun-classes.

Our decomposition of each protocol segment into smaller modules called *episodes* showed that when an episode contained *major* noun-classes (>50% of all noun-classes in the episode), the possibility of integration between the *major* noun-class and several *minor* noun-classes presented the best context within which to investigate the two hypotheses offered in this paper:

- “design acts have dependencies ... [which] highlight moves that have pre- or post-requisites for other design moves” [H1]
- describing the blow by blow process of integrating multiple design problem aspects, the noun-classes, into a comprehensive solution [H2]

For episodes with major noun-classes, the results of the regression analysis showed that these were dependent on the minor noun-classes. Estimates of Correlation Coefficient between the major and minor noun-classes echoed the results of the R-squared test. The correlations between the noun-classes provided a signature for the roles of each protocol segment in the overall work flow of the tasks being performed: namely requirement specification, preliminary design and design development.

In the Median Level Analysis the data shows that the control of design, despite the participation of multiple “designers” is anchored around the major noun-class (design objects) and the flow between these noun-classes indicates intermittent shifts in focus that

return to the major noun-class. This is a signature narrative that illustrates integration of physical design features (noun-classes).

In the Micro Level Analysis the data shows myriad finer grain design features generated by the design integration episode, such as, ship-like tree house, observation deck, a crow's nest, retractable roof, flipped or port-hole window, bird feeders, viewing through the window, flat portholes, bubble-inward portholes with feeders, dome shaped with transparent base roof, a rotating roof, open a section of the house, and regulating sunlight.

Viewed solely from the standpoint of discursive data, the entirety of the Maxi-, Midi-, and Micro- Level Analyses results provide the right evidence for gaining insight into how the nature of coded tokens (noun- and verb-) reveal the dependencies between these coded tokens [H1] and generation and integration of physical design features [H2]. We expect that the descriptions provided above lead to a more palpable (less abstract and quantitative) understanding of the dependencies in verb-noun codifications and the design integration process.

4. Conclusions and Future Contribution

We demonstrated in this paper that limiting protocol evidence to discursive data was not limiting, on the contrary it enabled us to apply a rich set of analytical techniques that were applicable to purely syntactic codifications. This freed us from difficult interpretations of visual-graphic data, semantic analysis and complex speculation of what it all means. Just as human acts are easier to encode than their intentions, discursive syntax is more reliable and less ambiguous than semantics.

There have been remarkably successful models of parsing the complexity of design protocols into semantic entities that can be represented with clarity, like the *Linkograph* representation which shows logic in design actions (Goldschmidt, 2014), the *Function-Behaviour-Structure* (FBS) ontology that models design as specific activities (Gero, 2002) and more recently the *Knowledge Handling Notation* that captures the dynamics of knowledge in design conversation (Scheer et al, 2014). In all of these methods, the final representation is based on abstractions of the data that have to be symbolically acquired, i.e., learned. In the Verb-Noun Analysis (VNA) method we used, the tokens of the taxonomy consist of natural language symbols. Thus it is easier to learn, understand, or decipher. We can readily see the dependencies between these symbols without overlaying a network of new symbolic relationships.

One conclusion we reached is Verb-Classes were not specific enough for any dependent relations – not even for question-answer sequences. Another one was that major noun-

classes (those that dominate a given episode) were dependent on the minor noun-classes. In turn this dependency was instrumental in understanding how design objects became anchors for all other noun-classes, particularly in the later stages of the design workflow process. These conclusions were also mirrored in the work of Georgiev and Tauru (2014) who explored the polysemy of nouns in design activity and found having a multiplicity of nouns in a design idea led to the generation of successful and creative design ideas (Georgiev and Tauru, 2014). While the focus of design-flow moved away from the major noun-class, it always returned, which also accounts for the dominance of the major noun-class within the episode.

Finally, we observe the following best practices for design in the office or in the studio:

- in design, anchoring the flow of actions around a single, major noun-class
- In design, keeping the focus on discursive information is vital, private solo design activities should be discouraged in lieu of team, conversational sessions
- In team design, concurrent and co-located activities by multiple participants increases the chances of multiple design object (physical feature) integration.

5. References

Akin, Ö. (1986, 1989) *Psychology of Architectural Design* Pion Limited, London

Akin, Ö, and C. Akin (1998) "On the process of creativity in puzzles, inventions and designs" in *Automation in Construction*, 7, pp. 123-138

Akin, O. (2001) "Variants in Design Cognition" in *Knowing and Learning to Design* edited by C. Eastman, M McCracken, and W, Newstetter, Elsevier, pp. Akin, 2001 105-124.

Akin, O. (2009) "Variants and Invariants of Design Cognition" in *About Designing: Analysing Design Meetings* edited by Janet McDonnell and Peter Lloyd, Taylor and Francis, London,

Archer, Leonard Bruce (1979) "Whatever Became of Design Methodology?" *Design Studies* 1, 1,17-20

Cross, Nigel (1984) *Developments in Design Methodology*, Chichester, UK: John Wiley & Sons

Cross, N., Christiaans, H., and Dorst, K. (eds.) (1996) *Analysing Design Activity*, John Wiley & Sons, Chichester, UK

Cross, Nigel (2006) *Designerly Ways of Knowing*. London: Springer

Dewey, John (1938) *Logic: The Theory of Inquiry*, Henry Holt and Co. N.Y.

Jones, J C and D G Thornley (eds) (1963) *Conference on Design Methods*, Oxford, UK: Pergamon Press

- Eastman, C. (1968) "Explorations of the cognitive processes in design" Research Report, Department of Computer Science, Carnegie-Mellon University
- Eisentraut, Renate. (1997) "Individual styles of problem solving and their relation to representations in the design process." *Design Studies* 18, no. 4: 369-383.
- Georgiev, G. V. and Tauru, T. (2014) "Polysemy in Design Review Conversations" Design Thinking Research Symposium 10, Purdue, IN.
- Gero, J. S. (1998) "Research in design computing: an artificial intelligence framework," in X. Huang, S. Yang and H. Wu (eds), *International Conference on Artificial Intelligence for Engineering*, HUST Press, Wuhan, China, pp.5-12
- Gero J.S. and Kannengiesser U. (2002) "The situated function-behaviour-structure framework", *Artificial Intelligence in Design '02*, Kluwer Academic Publishers, Dordrecht, pp. 89–104
- Goel, A. K., Rugaber, S., & Vattam, S. (2009). Structure, behavior, and function of complex systems: The structure, behavior, and function modeling language. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 23(01), 23-35.
- Goldschmidt, Gabriela (2014) *Linkography: Unfolding the Design Process*, MIT Press Ltd
- Goldschmidt, Gabriela (1997) "Capturing indeterminism: representation in the design problem space," in *Design Studies*, Volume 18, Issue 4, pages 441-455
- Howard, Thomas J., Stephen J. Culley, and Elies Dekoninck. (2008) "Describing the creative design process by the integration of engineering design and cognitive psychology literature." *Design studies* 29, no. 2: 160-180.
- Luck, Rachael. (2009) "'Does this compromise your design?' Interactionally producing a design concept in talk." *CoDesign* 5, no. 1: 21-34
- Laurel, B. (2003) *Design Research: Methods and Perspectives*, Cambridge: M.I.T. Press
- McDonnell, Janet. (1997) "Descriptive models for interpreting design." *Design Studies* 18, no. 4: 457-473.
- Oxman, Rivka (1997) "Design by re-representation a model of visual reasoning in design" in *Design Studies*, pp. 329-347
- Pearl, J. (2009) "Understanding propensity scores". *Causality: Models, Reasoning, and Inference* (Second ed) New York: Cambridge University Press
- Radcliffe, D. (1996) "Concurrency of actions, ideas and knowledge displays within a design team." *Analysing design activity*: 343-364.
- Scheer, A., Menning, A., Von Helldorf, E., Rhinow, H. and Nicolai, C. (2014) "The Knowledge Handling Notation: building an interface to enable design conversation diagnosis" Design Thinking Research Symposium 10, Purdue, IN.
- Schön, D (1983) *The reflective practitioner: How professionals think in action*, NY: Basic
- Simon, Herbert Alexander (1969) *The Sciences of the Artificial*, Cambridge: M.I.T. Press

Suwa, Masaki and Tversky, B. (1997) "How do architects interact with their design sketches in exploring design ideas?" Proceedings of 4th Australasian Cognitive Science. Conference '97 Newcastle, Australia.

Stanton, Neville A. (2009) "Hierarchical task analysis: Developments, applications, and extensions." *Applied ergonomics* 37, no. 1: 55-79.

Stempfle, Joachim, and Petra Badke-Schaub. (2002) "Thinking in design teams-an analysis of team communication." *Design studies* 23, no. 5: 473-496.

Visser, Willemien. (1996) "Two functions of analogical reasoning in design: a cognitive-psychology approach." *Design Studies* 17, no. 4: 417-434.

Biographies

Ömer Akin is a Professor in the School of Architecture at Carnegie Mellon University, USA. He holds a PhD in Architecture, a Masters in Architecture in Environmental Systems, a Masters in Architecture and a Bachelors in Architecture.

Olaitan Awomolo is a PhD student at Carnegie Mellon University

ⁱ The coders were the authors of the paper Omer Akin [C1} and Olaitan Awomolo {C2}