

Preliminary Findings from Coding Student Design Journals

Durward K. Sobek, II
Montana State University

Abstract

Since Fall 2000, mechanical engineering students at Montana State University have been required to keep design journals of their senior design projects. We have now accumulated over 70 journals on 21 design projects. We developed a coding scheme to code the journal data by design activity (problem definition, idea generation, engineering analysis, and design refinement), design level (concept, system, detail), planning, and formal reporting. The scheme was then used to code approximately one-third of the journal data collected to date.

This paper will first describe the coding scheme and its development. It will then report some preliminary findings from the journal coding. Specifically, we will show that journal data can be used to produce time profiles of design activity over the course of the projects, estimates of the proportion of time spent in activity associated with the different codes, and comparisons of student processes to “good” design processes as documented in the literature. These findings should be of interest to design educators interested in assessing design processes. Finally, the paper will posit a number of hypotheses that arise from the data, for future consideration.

1. Introduction

Design is one of the quintessential characteristics of the practicing engineer. It is perfectly appropriate, then, for it to hold a prominent position in engineering education—most engineering programs in this country culminate in a significant design project as the capstone of the degree program. Also, ABET places special emphasis on design in its accreditation evaluation criteria.¹

The activities that typically fall under the category “design” consist of analysis activities, that is, making some determination about an existing idea or solution, and synthesis activities—generating a new idea to address an identified problem. While a good deal of research has looked at design, and there is much we know about good design and good design processes, there is still much we do not understand about the synthesis process. Therefore, it seems if we want to help aspiring engineers become proficient designers, it behooves us to delve into the human synthesis process, to really get at the basic fundamentals of what enables synthesis, what hinders it, and what tools and skills are requisite.

In 2000, I embarked on a project funded by the National Science Foundation to in part address some of these questions. The idea was to study the design processes students use in their senior

capstone projects in detail, compare them to project outcomes, and hopefully learn something about what synthesis processes tend to lead to better design outcomes.

Researchers have used a number of methods to study design processes. Bucciarelli² directly observed engineers in the course their work. Atman and Bursic,³ Cross, et al.,⁴ and others have used protocol analysis—subjects work on a design while talking aloud, and the investigators audio or video tape the activity, then analyze transcripts of the tapes. Such analysis enables a fine-grained look at design activity. Other approaches are retrospective. An investigator can build a case study through interviews of design participants about the project, or piece the project together from design documentation, or ask the design participant to document how the project progressed, or use some combination.⁵ Waldron and Waldron⁶ offer a “depositional method” that combines protocol analysis and interviews—the researcher takes “depositions” from the design participants at specific junctures in the design process.

Each of these approaches has its difficulties. Direct observation is time intensive, so one’s sample is necessarily limited. Additionally, direct observation seems difficult to conduct a) when design participants can meet just about anywhere and at all hours, and b) without influencing the student interaction, especially if the observer is a professor. Protocol analysis is equally difficult. The feasibility of recording design activity over the 15-week semester is questionable, and the tapes that are made must be transcribed, a time-consuming task in itself. For these reasons, protocol analysis is best used for short-duration design tasks (e.g., 4 hours or less) accomplished in a laboratory setting. Using retrospective approaches with students, especially seniors, is difficult because they tend to graduate and leave; plus these approaches typically do not yield data at the level of detail desired for this study.

So I decided to take a different approach. I would have the students keep design journals as part of their course grade, and collect them at the end of the semester. Engineering journals (or engineering/design notebooks) were once standard practice in professional practice and education, but in recent years the practice has waned as digital technologies have provided new ways to represent and store engineering information. Fortunately, the ME faculty at Montana State University were sympathetic to journaling, and agreed to re-institute them if I could provide support. To date, I know of no other study that uses journals to investigate design processes (although certainly design journals are in use in many design courses⁷⁻⁹), so this is new territory.

We’ve collected journals for three semesters (Fall 2000, Spring 2001, and Fall 2001) and now have over 70 journals on 21 projects. The Fall 2000 journals were of low quality for reasons explained in a companion paper,¹⁰ so only the Spring 2001 journals have been analyzed. This paper reports the coding scheme used to analyze the journals along with some preliminary results, and discusses future directions and limitations of using journals to study design processes.

2. Course Background

As mentioned earlier, design journals have been required in ME 404, the senior design capstone course, for the last 3 semesters. It is a 4-credit, one-semester (15 weeks) course. An instructor

facilitates the course, meeting with the class once per week to cover course logistics and communicate deadlines and reporting requirements. The students are assigned to teams of 3-4, each team working on a different project. Most projects are sponsored by outside organizations. Each team meets weekly with a faculty advisor (who could be the course instructor). Typical of many senior design courses, each team must interact with a client to define his needs, devise a solution to meet those needs, and deliver a product (written report, set of engineering drawings and specifications, oral report, and sometimes a hardware prototype) by semester's end.

Journals constitute 15% of each student's grade for the course. The journal grade is the only individual component of the grade (the remaining 85% are group grades). To increase the quality of journals, the students are required submit journals periodically for a "journal check" roughly five times over the semester. The journal content since the last check is evaluated using a rubric.⁸ Students are given feedback in the form of written and oral general exhortations to address deficiencies, but care is taken not to direct the students to record specific kinds of information or to record it in certain ways (e.g., "this entry seems sparse for a two-hour meeting" rather than "take more notes"). The only format requirements placed on the students are to: date each entry, record the start and stop times of each entry, and start each day on a fresh page. We also ask them to use ink, use contiguous pages and refrain from stapling computer printouts into the journals.

3. Coding Scheme and Development

One of the hypotheses in this stream of work is that process affects outcome. To better understand this connection, it would be useful to have a way to characterize (and eventually model) a design process. Authors in the area of engineering design often characterize the design process through multiple lenses. Two of the more common are to view the process as series of overlapping phases (e.g., Ulrich and Eppinger's generic product development process¹¹ has 6 phases: problem definition, concept design, system-level design, detail design, test and verification, and production) and as a set of activities (e.g., Atman and Bursic³ identify 9 design activities for their protocol analysis: gathering information, generating ideas, modeling, evaluation, and so forth).

One of the research questions we want to address with this course of inquiry is how do designers (in this case, student designers) reason at different levels of abstraction. The above two lenses seem useful to this work as the phase viewpoint highlights the fact that design participants work at different levels of abstraction throughout the project, while the design activity viewpoint highlights what participants actually do as they design. We decided to combine the two lenses into a double-coding scheme rather than focus on one or the other. But to do so meant simplifying the frameworks so that the number of codes is manageable (9x6=54 codes seemed overwhelming!), even at the expense of losing some fidelity along each dimension. The double-coding would then enable us to look closely at what design activities students are engaged when working at different levels of abstraction.

The coding scheme development began with rough definitions for the level of design at which the

student designer is working and design activity. The design levels map pretty closely to Ulrich and Eppinger's generic process,⁹ although several other design texts have similar breakdowns. The design activities are more or less a regrouping of the design steps in Atman and Bursic.³

The codes are summarized in Figure 1. The coding scheme categorizes journal entries into three levels of design (concept, system, and detailed design levels) to capture high, medium, and low levels of abstraction, and four types of design activity that capture the heart of the engineering problem-solving process (problem definition, idea generation, engineering design analysis, and design refinement or iteration). Design levels and activities are put in a matrix so that each design-related journal entry receives two codes. Two additional designations describe non-design-related activity (project management and project delivery).

| | Concept Design | System-Level Design | Detailed Design |
|-----------------------------|-----------------------|----------------------------|------------------------|
| Problem Definition | C/PD | S/PD | D/PD |
| Idea Generation | C/IG | S/IG | D/IG |
| Engineering Analysis | C/EA | S/EA | D/EA |
| Design Refinement | C/DR | S/DR | D/DR |
| Project Management | PM | | |
| Delivery | RW, PP | | |

FIGURE 1: CODING MATRIX

To refine the code definitions, the research team consisting of the author and two undergraduate research assistants coded excerpts from several journals independently. We then discussed our discrepancies to arrive at a common understanding of each code. Next, we coded larger chunks of journals, each taking a different student's journal, then traded and critically evaluated one another's coding. Again, where we had disagreement we discussed until there was consensus. Through this process, we arrived fairly concrete definitions for each code (described below) with lists of the types of journal entries that would most likely fall under that code.

The design level codes differentiate three levels of design as defined in Figure 2. Concept design occurs when problems or sub-problems are addressed with new ideas or approaches. Some activities at the concept level include: identifying customer needs, establishing the target specifications of the problem, and generating, testing, and selecting concepts. Activities in system-level design address the definition and configuration of subsystems and their interfaces (i.e., system architecture). Detail design activities look at individual components in the subsystems and focus on quantifying specific features required to achieve a particular concept. These activities include defining part geometry, specifying tolerances or dimensions, and material selection.

The level of design for a journal entry was determined through a series of questions. The first question is, "What problem does this entry address?" This was established by examining the

context of the entry, both within the journal itself and by cross-referencing the other design journals. Once the problem is determined, the next question is, “Does this problem address the definition or configuration of systems or interfaces?” That is, decide if the problem is at the system-level of design. If not, then we determined whether the problem was more detailed or conceptual in nature.

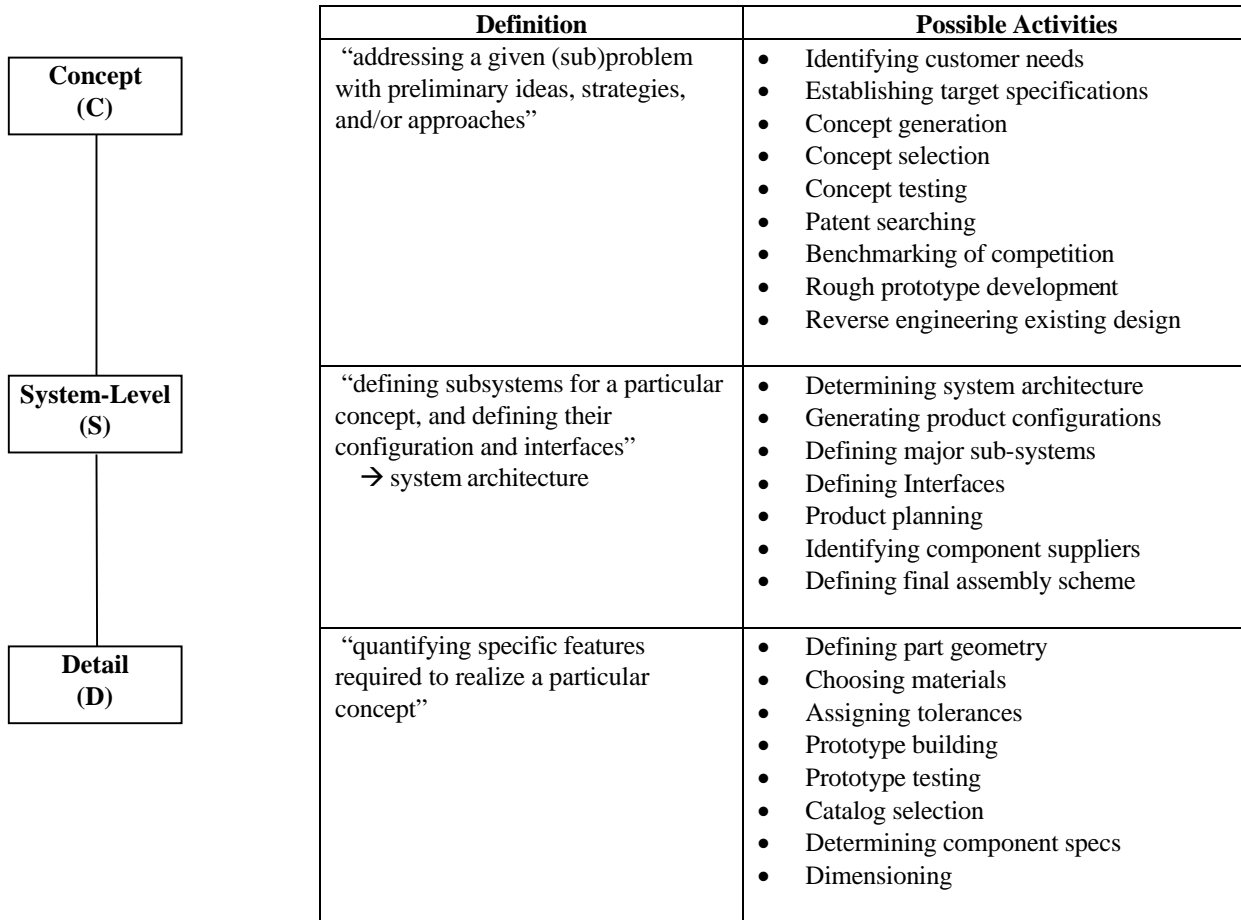


FIGURE 2: DESIGN LEVEL CODE DEFINITIONS

The coding scheme also differentiates four categories of design activity. Figure 3 defines each term and gives examples of possible activities found in design journals that would indicate that design activity.

As with design level designation, determining which activity designation to give a journal entry can be difficult in practice. The “possible activities” listed are likely to fall under that code but not guaranteed, depending on the context in which the activity occurs and the purpose of that activity. The first step in determining the design activity is checking the context of the entry to determine the purpose of the entry. Once this has been established, the definitions can usually be applied directly.

| Designation | Definition | Possible Activities/Indicators |
|---------------------------|--|--|
| Problem Definition (PD) | “gathering and synthesizing information to better understand a problem or design idea” | <ul style="list-style-type: none"> • Researching data/ideas to determine design specifications • Creating/Determining parameters • Defining design requirements • Gathering background information • Stating a problem • Identifying deliverables • Identifying need |
| Idea Generation (IG) | “qualitatively different approach(es) to a recognized problem” | <ul style="list-style-type: none"> • Brainstorming • Listing alternatives (ideas) • Looking for <u>ideas</u> via internet, patents, etc. (i.e. “What’s out there?”) • Recording “breakthrough” ideas • “What if...?” • “How about...?” • Using CAD, PROE, or SolidWorks for: <ul style="list-style-type: none"> → Preliminary design models to better communicate an idea |
| Engineering Analysis (EA) | “evaluation of existing design/idea(s)” | <ul style="list-style-type: none"> • Mathematical modeling • Using decision matrices • Listing pros/cons of ideas • Examining a design with analytical diagrams • Testing (actual, theoretical) • Graphical modeling for: <ul style="list-style-type: none"> → interference testing → dimensional analysis → calculating density, volume, etc. |
| Design Refinement (DR) | “modifying or adding detail to existing design/idea” | <ul style="list-style-type: none"> • Drawing completed sketches of a design (including tolerances, dimensions, etc.) • Studying a design to determine functionality • Changing existing features • Adjusting/refining specs • Troubleshooting of design • Using CAD, PROE, SolidWorks for: <ul style="list-style-type: none"> → Detailed modeling |

FIGURE 3: DESIGN ACTIVITIES CODES AND DEFINITIONS

Finally, the coding scheme designates codes for “non-design” activities associated with project management and project delivery. Project management (PM) covers project planning and progress evaluation, including: scheduling, class meetings, identifying tasks, and reporting project status. The delivery category is for activities associated with interim and final report writing (RW) and final presentation preparation and delivery (PP).

4. Coding and Process Measurement

The two undergraduates research assistants took the lead, each coding journals from a different team. Before getting into the journals, they read and studied the team's final written report. Once the senior researcher was satisfied they had a good understanding of the overall project, they coded all 3 or 4 journals for the team simultaneously, one day at a time. So, for example, they would read and code team member A's Jan. 31 entries, then team member B's Jan. 31 entries, and so on before moving to any team member's Feb. 1 entries. This way they could cross-reference journal entries, have a better picture of all that was happening at a given point in the project, and code each event consistently.

Once the journals were coded, the research assistants went back through journals and assigned time values to each code from the time stamps recorded. If a given entry had more than one code designation, they allotted time in proportion to the amount of space taken up on the page. Journals were gridlined, which facilitated the time measurement. Thus, for example, if a one-page journal entry has a start time of 1:00 and end time of 2:00, and 4 grid blocks are taken up with problem definition (PD) activity and 2 blocks are occupied with idea generation (IG), the PD is assigned .67 hours and IG gets .33 hours for that entry. And if all that activity concerned conceptual level design issues, then the C code received 1.0 hours (*note*: all design-related activities were double-coded). Codes and times were manually recorded directly in the journals, in a color of ink different than used in the journal.

Often journal entries for the same event in two or more journals disagreed, either in content or in time. We developed a set of rules for dealing with such discrepancies so as to treat each case consistently.

Once the research assistants had coded and calculated times, the lead researcher (myself) carefully reviewed the coding. Like the research assistants, I went through all 3-4 journals for a team simultaneously, one day at a time. I kept a running log of all errors found and any disagreements I had with the coding on a separate sheet of paper. The research assistant and I then discussed each item until we reached agreement, and made corrections to the coding as necessary.

Once we had reached mutual satisfaction with the accuracy of the coding, the times for each code were tallied in an Excel spreadsheet by week, and aggregated to the project level. We could then say for a particular project team that they spent, for example, 24.5 engineering hours in week 6 working on the midterm report. The following section reports the early fruits of our efforts.

5. Results and Discussion

We now report samples of the journal coding results. A paper that compares processes across different projects will be forthcoming. The first bit of analysis we can do is to determine the proportion of time (based on engineering hours) that a student team spends on design activity versus project management, report writing, and presentation preparation/delivery activities. Figure 4 displays the breakdown for one design team (4 student members). We can see that of the 617 engineering hours logged by the team in their journals, only 40% of the time the team is

*Proceedings of the 2002 American Society for Engineering Education Annual Conference & Exposition
Copyright © 2002, American Society for Engineering Education*

actually working on the design problem (that is, doing “design activities” like generating ideas, creating drawings, doing analysis work, searching for vendors, and so on). Roughly 20% of their time is spent managing their project, and the remaining 40% was spent on reporting activities.

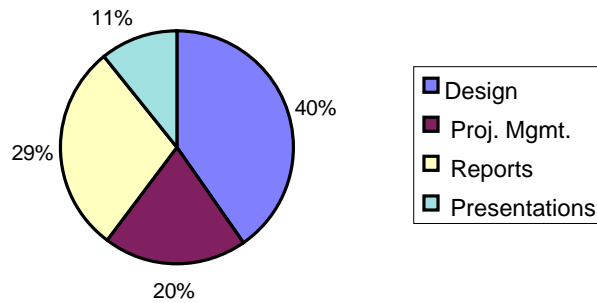


FIGURE 4: BREAKDOWN OF ENGINEERING HOURS FOR ILX PROJECT

Next, we can see how those activities break out over the semester week by week, as shown in Figure 5 (we combine presentation preparation and report writing in this chart). Note that week 8 is spring break. The first thing that may stick out is that effort is not uniform across the semester. Some weeks have light activity, while in other weeks team members can put in 20 or more hours each. From this diagram we see that report writing is concentrated in the weeks leading up to their midterm report deadline in week 7, and to the final report/presentation deadlines in week 15. This team actually spread their reporting activities out over six weeks or so, unlike some other teams (usually poorer performers) that concentrated report writing into shorter time frames. As one might expect, while the team is writing and preparing oral presentations, little design activity occurs.^a We can also see a large spurt of design activity in week 5, leading up to the midterm reports. This is probably due to an emphasis by the instructor that he wanted to see concepts generated, a final concept selected, and significant design progress on the final design in the midterm report. Project management activity is pretty constant over the semester, but interestingly seems to increase slightly in the second half.

^a It is possible that the act of writing has a transformative effect on students’ understanding of their designs/solutions; however, we could find no evidence of this in our data.

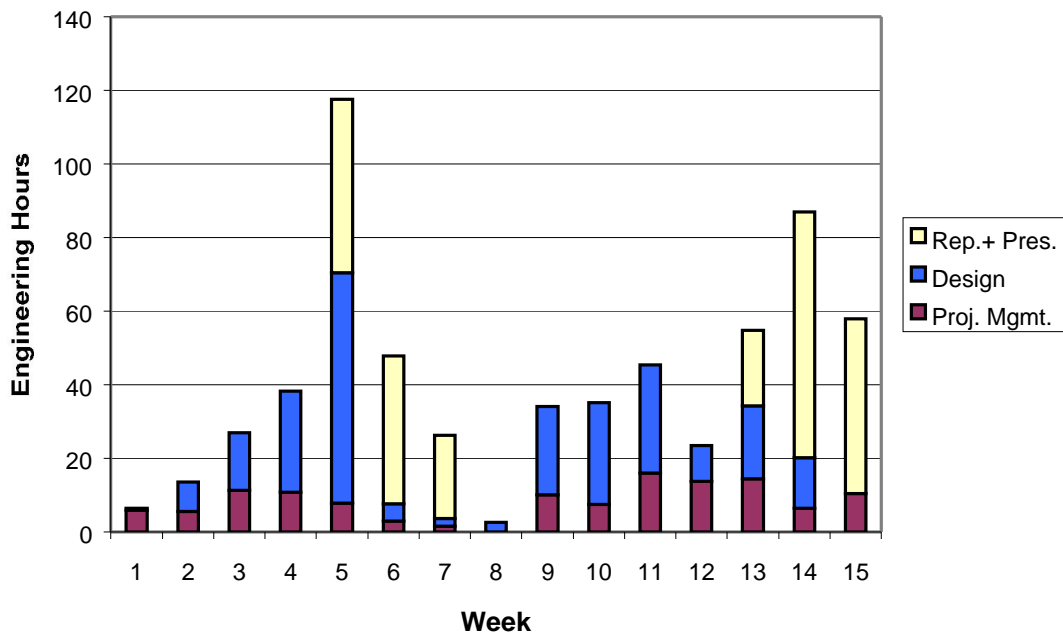


FIGURE 5: TOTAL ACTIVITY PROFILE BY SEMESTER WEEK (ILX TEAM)

Now we can take just the design activity (the blue bars in Figure 5) and break it out by design level and design activity, as shown in Figures 6 and 7 respectively. As one would hope to see, the team starts working at the conceptual level, then starts working on system-level issues, and then moves to detail design tasks. This progression is consistent with design theory literature.⁹ Additionally, the chart shows that the transitions between design stages are not abrupt—one stage phases out while the team moves on to the next. Also interesting to note, the team spent almost no time on design activity in 5 weeks of the 15-week semester (weeks 1, 6-8, and 15).

Inspection of design activity shows a perhaps more interesting story. First, note that problem definition activity occurs at some level throughout the project. In every week that there is design activity, the team members spend at least some time on problem definition. The spurt of problem definition activity in week 5, the week they did concept selection, is because a) they had to develop selection criteria, and b) once they decided on a concept they had to develop a set of specifications specific to that concept. Second, design refinement (or iteration) begins quite early, by week 4, and continues through week 13. Third, idea generation is localized to the first few weeks, virtually dying out once the concept has been identified. Interestingly, the small spurt of idea generation in weeks 9-10 occurred because the team ran into a snag with their concept, so they had to come up with an idea to overcome it. Fourth, very little engineering analysis occurs at the concept level. Most of it comes after the team is committed to a single design alternative. If that's the case, then the team has potentially decided upon a design at the concept level, before doing much feasibility analysis.

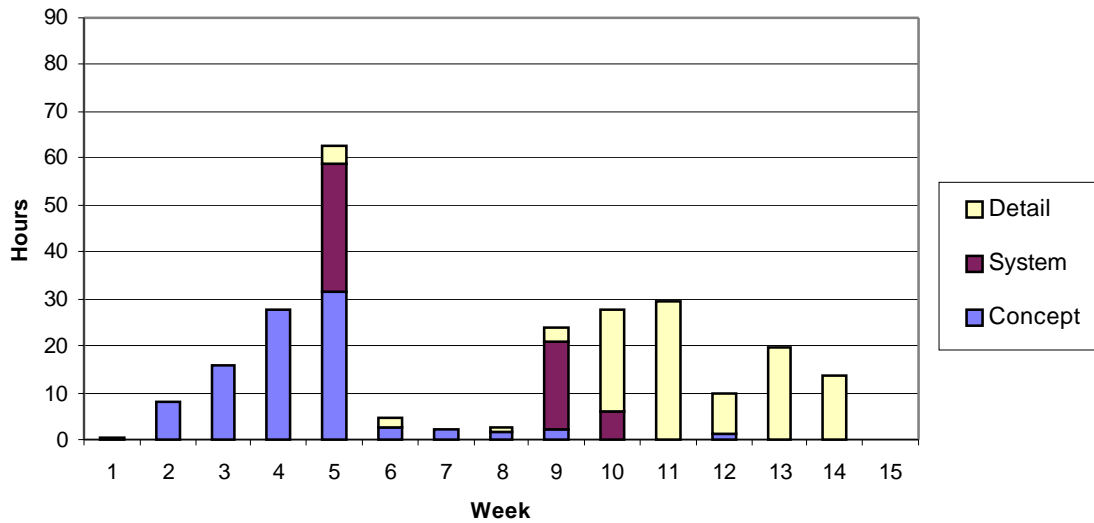


FIGURE 6: DESIGN ACTIVITY BY LEVEL OF DESIGN (ILX TEAM)

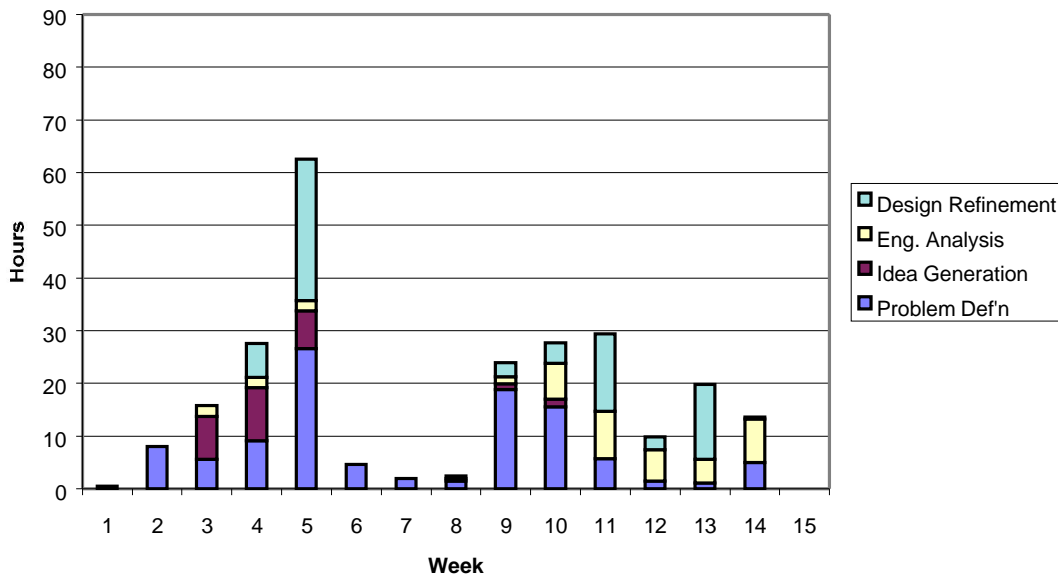


FIGURE 7: DESIGN ACTIVITY BY TYPE OF ACTIVITY (ILX TEAM)

So this analysis shows that we can gain some insight into the design process used by the student team. The next step in our analysis plan is to compare characteristics across a number of projects, correlate these characteristics to project outcomes such as design quality, design creativity, and

hours spent, and hopefully learn something about how our students are approaching design problems, and how they can do better. In order to do this, however, we will first need to establish reliability measures to ensure that the data are comparable.

Of course, this approach is not without its problems. The most glaring issue, perhaps, is that the journal records are only a partial record of the student's thoughts and activities. Journal quality varies, but even the best journals capture only a portion of what actually transpired, and may actually be biased toward capturing certain kinds of information while discarding others. A second issue is accuracy. How do we know that the students are not fabricating a record? We can compare journals to see if they corroborate, but the bottom line is we must trust the students to give us an honest record. Third, the times are rough approximations at best. Not only do we have little cross-verification of times recorded, using area proportion to allocate times across codes (when there are multiple codes) may be a poor approximation in many cases.

To validate the data recorded in the journals requires significant more effort than we've been able to garner for the project. Perhaps the best way to validate would be to perform in-depth observation of a subset of the sample, then compare the observer's record to the students' journals to determine just how complete/accurate the journal records are. We may have opportunity to do this next fall with the addition of 2-3 graduate students to the research team.

Finally, the analysis to date focuses exclusively on activities. We have not yet tackled the deeper cognitive issues that are the main motivations of this work. Worse, it's not clear that the journal data really capture the "true" story. A student may record that the team decided to go with Idea B, but we usually don't know how much debate there was, what the arguments pro and con were, what the overriding rationale was. We may, through more training and effort, be able to capture this kind of data through journaling, but that remains to be seen.

6. Conclusion

By way of conclusion, we have developed a coding scheme to evaluate student design journals, and have coded several sets of journals. We have demonstrated that the coding scheme can give us a fairly rich description of the design process at the activity level, which has potentially interesting insights. It sets the stage for larger scale analysis across multiple projects. However, the data captured are at best an incomplete picture of what really goes on, and at worst may not be all that accurate. Still, the approach seems a valid way to gain insight into our students' design processes as they work independently.

Acknowledgements

This work was sponsored by National Science Foundation grant #REC-9984484. Many thanks to the ME 404 instructors, Drs. Vic Cundy and Michael Wells, the ME 404 advisors, and ME 404 students for their help and cooperation. Also, thank you to Robert Lowis and Seth Partain for their diligent efforts on this project.

References

1. www.abet.org
2. Bucciarelli, Louis L., *Designing Engineers*, 1994
3. Atman, Cynthia J. and Karen M. Bursic, "Verbal Protocol Analysis as a Method to Document Engineering Student Design Processes," *Journal of Engineering Education*, vol. 87, no. 2, April 1998; pp. 121-131.
4. Cross, Nigel, Henri Christiaans, and Kees Dorst (eds.), *Analysing Design Activity*, John Wiley and Sons, Chichester, UK, 1996.
5. Sobek, II, Durward K., *Principles that Shape Product Development Systems: A Toyota-Chrysler Comparison*, Ph.D. dissertation, The University of Michigan, 1997.
6. Waldron, M. B. and K. J. Waldron, "Methods of Studying Mechanical Design," in M. Waldron and K. Waldron (eds.), *Mechanical Design: Theory and Methodology*, Springer-Verlag New York, Inc., 1996, pp. 21-34.
7. Mehalik, Matthew M., Larry G. Richards, and Michael E. Gorman, "Turning Students into Inventors and Entrepreneurs: The Continuing Evolution of a Course on Invention and Design," *Proceedings of the 1999 Frontiers in Education Conference*, San Juan, Puerto Rico.
8. Burrows, V. A., B. McNeill, N. F. Hubble, and L. Bellamy, "Statistical Evidence for Enhanced Learning of Content through Reflective Journal Writing," *Journal of Engineering Education*, vol. 90, no. 4, Oct. 2001, pp. 661-668.
9. Korgel, B. A., "Nurturing Faculty-Student Dialogue, Deep Learning and Creativity through Journal Writing Exercises," *Journal of Engineering Education*, vol. 91, no. 1, Jan. 2002, pp. 139-146.
10. Sobek, II, Durward K., "Use of Journals to Evaluate Student Design Processes," submitted to the 2002 American Society of Engineering Education Conference.
11. Ulrich, Karl and Steven Eppinger, *Product Design and Development*, 2000.

DURWARD K. SOBEK, II

Durward Sobek is currently Assistant Professor of Industrial and Management Engineering at Montana State University. He holds a B.A. degree in Engineering Science from Dartmouth College, and the M.S. and Ph.D. degrees in Industrial and Operations Engineering from The University of Michigan. His current research interests are in the areas of new product development, engineering design education, and health care. Please visit his web site for additional information: <http://www.coe.montana.edu/ie/faculty/sobek>