

# Understanding the Nature of Software Design Processes

Kumiyo Nakakoji<sup>1,2</sup>

Yasuhiro Yamamoto<sup>1</sup>

Nobuto Matsubara<sup>2</sup>

<sup>1</sup>RCAST, University of Tokyo  
University of Tokyo  
Tokyo, Japan

{kumiyo, yxy}@kid.rcast.u-tokyo.ac.jp

<sup>2</sup>SRA Key Technology Laboratory Inc.,

Tokyo, Japan  
matubara@sra.co.jp

**Abstract**—The goal of our research is to understand software design processes. The approach toward the goal is twofold: by designing and building a process viewing tool and by conducting in-depth qualitative analyses of the study materials. This paper first briefly describes the tool we have built for viewing software design process data, including video, transcripts and notes. We then list a set of characteristics we have found interesting in each of the three studies: Adobe, Amberpoint, and Anonymous. We conclude the paper by discussing the current findings, implications, and future plans.

**Keywords**—software design processes; visualizing process; tool support; observational studies

## I. INTRODUCTION

A software design process involves the design of vocabulary elements (concepts as well as object names and their relations, which turn into class names and method names constituting a program), temporal and logical flows of events (user interactions, event transactions and controls), and visual elements (ER diagrams and graphical user interface layout).

The three types of elements depend on each other and coevolve during the design process. For instance, an object name (such as “intersection” in the Adobe video) identified as an important concept is used to verbally describe stepwise transactions to be designed, which may then help designers to bring in another term (such as “cops”) helping them understand and share the abstract role of the “intersection.” Designers may sketch a graphical user interface to depict essential objects, and go through a scenario of how a user would interact with the system under design, which may help the designers revise the temporal sequence of system transactions.

Different media are used as a means to drive such design processes. Verbal communications, gestures, and white board drawings, which were recorded in this design study, can be regarded as the primary media.

What interests us most is how such different types of design elements evolve over time through what representational media. Textual terms are used not only for representing design concepts but also for building program structures (as object names in object-oriented languages like Java). Diagrams are used both for describing logical and temporal flows as well as visual elements of user interfaces. It is quintessential to analyze such interplay between textual and diagrammatic representations, which evolve over time.

In this Studying Professional Software Design project, we are interested in understanding software design processes: *what is the nature of software design processes?* Such an understanding would help us better design an analytical framework for further studying software design processes, would inform us in designing tools and environments for software design, and would guide us in teaching software design.

Our approach is twofold toward this goal: by designing and building a process viewing tool and by conducting in-depth qualitative analyses of the study materials. This paper first briefly describes the tool we have built in viewing the software design process data, including video, transcripts and notes. We then list a set of characteristics we have found interesting in each of the three studies: Adobe, Amberpoint, and Anonymous. We conclude the paper by discussing the current findings and their implications, and future plans.

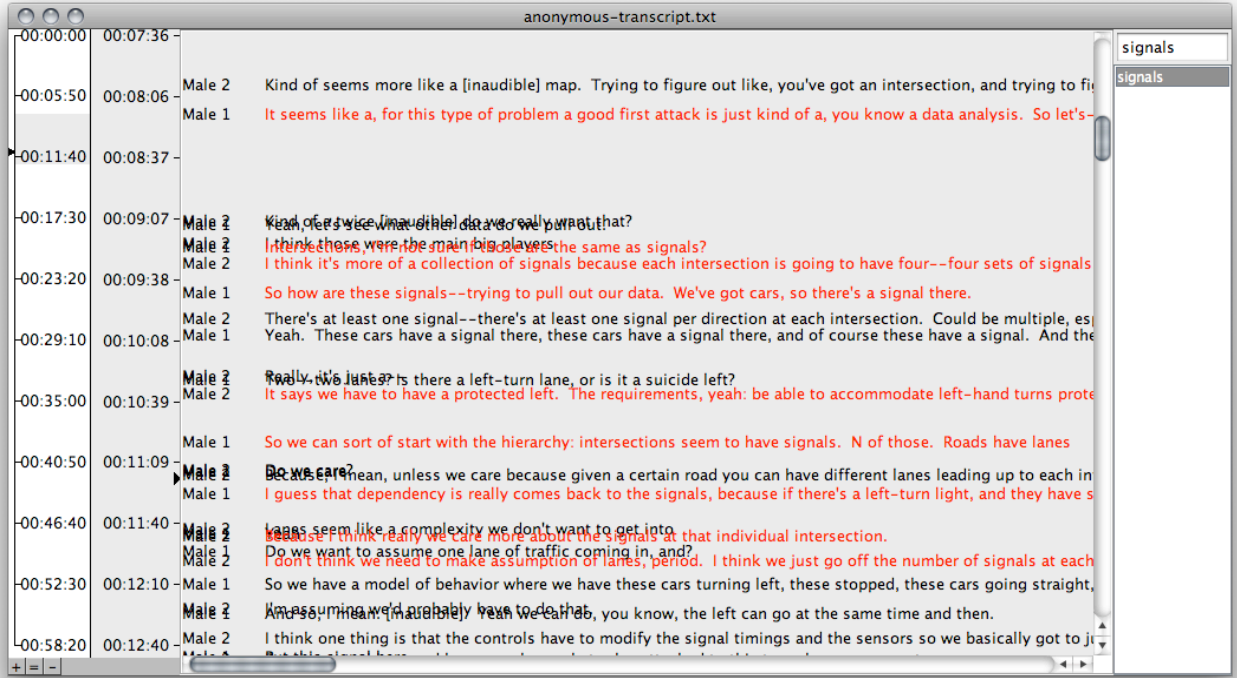
## II. TOOLS FOR VIEWING THE DATA

The tool we have built (currently called CProVA: Creative PROcess Viewer and Annotator) consists of three components: the Transcript Viewer for the transcript data, the Movie Viewer for the movie data, and the Note Viewer for the notes taken by an individual analyst (Figure 1). The three components are integrated through the timestamps of each data.

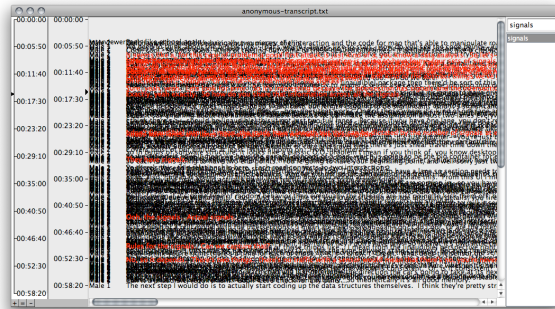
### A. The Transcript Viewer

The Transcript Viewer (Figures 1(a) and 1(b)) has two vertical timeline panes on the left, the transcript pane in the middle, and the search pane on the right. The left-most timeline shows the timeline of the entire recorded session, starting from the top and ending at the bottom. A user can zoom in and out over the timeline by pressing the “+” or “-” buttons located in the bottom. The presently focused area is displayed with the gray background. The second-left timeline bar shows the focused time period (Figure 1(a)).

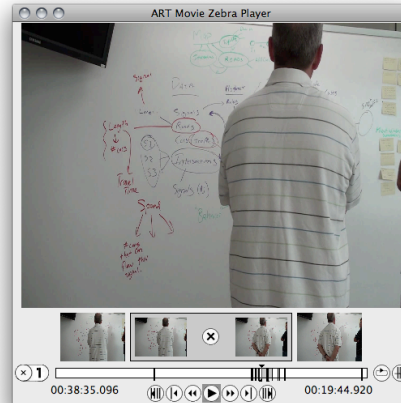
The transcript pane displays text transcripts within the focused time period. Each sentence is displayed in one line (no wrapping), and the location of the sentence is determined with the timestamp of the sentence. Thereby, more sentences are displayed overlapping each other when study participants densely spoke and no sentences are displayed when no conversations were carried out.



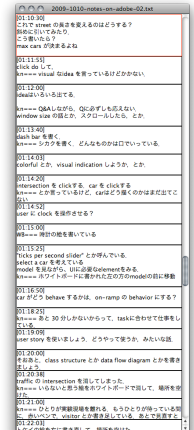
(a)



(b)



(c)



(d)

Figure 1: CProVA, which consists of the Transcript Viewer (a,b), the Movie Viewer (c), and the Note Viewer (d)

A user can search for a phrase by typing in a window in the top of the search pane. The system incrementally identifies matching words to the queried phrase extracted from the entire transcripts, and lists them below the query window. When choosing one or more matching words, all the sentences that have the selected matching words are displayed in red in the transcript pane. For instance, in Figure 1(a), the transcript pane displays the transcripts between 00:07:36 and 00:12:40 of the Anonymous group, and red sentences are the ones that have a word “signals.” From this visualization, one may say that during this period of time the participants quite often mentioned “signals” as there are many sentences in red. As another example, Figure 1(b) shows the entire transcript of the

Anonymous group and shows where “signals” were uttered. One may say that they talked about “signals” more often in the first half of the process than in the second half of the process (as red lines are more dense in the upper part of the pane).

### B. The Movie Viewer

With the Movie Viewer (Figure 1(c)), a user can play, pause, fast forward a movie, as well as change the playing speed. The user may specify one or more segments of the movie (indicated with a gray bar area of the control bar located in the bottom) to play only the selected segments.

When a user selects a sentence in the Transcript Viewer by shift-clicking on the sentence in its transcript pane, the Movie Viewer jumps to the frame with the corresponding timestamp. Thus, by clicking on a transcribed sentence in the Transcript Viewer, one can watch the corresponding video segment uttering the transcribed sentence in the Movie Viewer.

When a user selects one or more matching words in the Transcript Viewer and the system locates the multiple sentences among the transcript data that have the matching words, the Movie Viewer also identifies the video segments that have the same set of timestamps. By selecting the selected-segment-play-only option, a user then can watch all the video segments that have the utterances that have the selected matching words in a sequential manner.

### C. The Note Viewer

The Note Viewer displays notes taken by a researcher while watching videos and reading transcripts if associated with the corresponding timestamps. When the user selects a sentence in the Transcript Viewer, the Note Viewer displays the corresponding note having the same timestamp.

## III. RESULTS OF OBSERVATIONAL STUDIES

We have conducted in-depth analyses of the study materials. This section lists a set of characteristics that we have found quite interesting in each of the data of the three groups. It is important to note that the goal here is *not* to characterize each of the three design processes. Rather, our interest is in finding interesting characteristics of software design processes. Because each of the three teams consists of professional software designers, we believe that such characteristics would lead us to better understand the nature of software design processes. In what follows, the findings are listed in the chronological order for each of the three design studies. Timestamps of the video material in the form of “(hh:mm:ss)” are provided where appropriate.

### A. The ADOBE Study

#### 1) *some terms are recorded on the whiteboard and grow into objects*

M and M2 started by working on “basic data structures” (00:05:56). During the conversation, terms that seem to play an important role in the model were mentioned, and M2 started writing them down in the left side of the whiteboard, such as, “intersection,” “car,” and “time.” Other terms that also seem important, such as “queues” (00:08:28), were mentioned but were not written down on the whiteboard then.

#### 2) *a diagram to understand the world to model*

After a few minutes talking about the model, M2 wrote a picture of an intersection on the whiteboard (00:06:41). This picture was not used so much as an image of the map visualization but as an object-to-think-with in understanding the role and behavior of an intersection, and its relationships with cars, lanes, and signals.

#### 3) *sectioning of the whiteboard evolves over time*

When M mentioned the “controller,” (00:09:37), M2 went back to the place on the whiteboard where he wrote down a list

of terms, and put “model” on top of the list and underlined it (00:10:05). Immediately after that, he wrote down “controller” in the space below and underlined it. Now the whiteboard had two areas, one labeled “model,” and the other labeled “controllers.” The initial list of terms became a part of the “model.”

#### 4) *the role of diagram evolves as more contexts are added*

M2 initially drew the picture of an intersection (i.e., crossing roads) when talking about a simple intersection, cars, traffic lights, and clock (00:06:41). When talking about the controller (00:09:37), M2 labeled the original intersection with “T0” and then draw another picture of an intersection on the right side of the original one, and labeled it with “T1” (00:11:29) to see “how pictures will change from one moment of time to another and think about how the software would create those pictures maybe” (00:09:57).

#### 5) *annotations to describe the term written on the whiteboard*

M2 wrote down “clock controller” under the label “controllers,” and boxed the words. Then M2 drew an arrow and wrote “send ‘tick’ event to model” (00:13:40). This seems to become the description of what “clock controller” would do.

#### 6) *a term not mentioned in the given specification grows into an important concept*

M mentioned “cop” for the first time at (00:14:06) by saying “It’s almost like you need to traffic cop, right, traffic cop controller.” The term “cop,” which was not used in the given problem description, served as a main player in modeling an intersection throughout the design process, but was not immediately written down on the whiteboard then.

After M mentioned the term, M2 drew a larger square around the model and its components on the whiteboard (00:14:16). He added a box underneath the square he had just drawn as if he was ready to add “external” entry to the “model.” M2 wrote “Event (e)” in the new box by M2 saying “So, something happens as an external controller is ticking the model, is that what your saying?” (00:15:07).

Then, M2 wrote down “cop” on the whiteboard near the “model” label (00:15:15), then M said, “Well, I don’t know, is the cop really a model or is that a controller. A cop is kind of controlling the state of the model” (00:15:18). M2 then immediately erased it (00:15:26). He then deleted the square frame surrounding the model field. After pondering for a few seconds, M2 started boxing each of “intersection,” “car” and “time” by saying, “these are objects, right?” (00:15:29). He deleted “Event” that he kept unboxed.

When M2 wrote “cop” again on the whiteboard (00:16:37), he wrote “responds to time changes” and “changing state of model” while verbalizing them. He then wrote a square around the “cop” and its description, and asked M if M thought it was reasonable.

This resulted in the list of boxed terms under the model field, and “cops” had become a boxed object under the “model” category.

*7) grouping labels may not remain throughout the process for the sake of affording more spaces*

At time 00:18:30, M2 erased the “model” label and wrote “road” where “model” was on the whiteboard. By adding properties on to the “road,” M2 deleted “road” and wrote it again in an upper position on the whiteboard so that it has more space below to write more properties (00:24:05).

*8) occasionally going back to the problem specification*

When they were talking about how “push” and “pull” were done by “cop,” M wondered, “Well it does say that something that the user could control” (00:25:32) and they both went back to the problem description sheet to better understand how the simulation is controlled by a user (i.e., student) (00:25:47).

*9) more sectioning of the whiteboard emerges as the design proceeds*

When M and M2 agreed to work on “cop rules” (00:33:53), M2 moved to the right side of the whiteboard, and wrote “rules,” underlined it, and then put “1” underneath. Now the whiteboard was divided into three sections: the left side lists boxed object names, the central section shows two intersection pictures, and the right side shows “cop rules.”

*10) using pseudo-code expressions on the whiteboard*

When M2 wrote down each rule for the “cop rules,” he wrote it in quite detail by using a pseudo programming language like notation (00:34:32). For instance, “1 car per tick passes if car is the head of the road” when “car == straight and I.light == greenlight R.pop” (00:36:15). When he wrote “I.light=green” on the whiteboard, what he was actually saying was “And green, and intersection I.light.” What he uttered was transformed into a programming language like expression when being written down on a whiteboard.

He then put “(1)” in the head of the line he just wrote by saying “So that’s case one, that’s case one. Case two would be...” (00:36:06) and put “(2)” in the new line.

*11) re-appropriating the diagrams on the whiteboard*

When the two were discussing about the two intersections next to each other (00:37:55), they started pointing to the two intersection pictures drawn previously (00:09:57). These two intersections originally represented the same intersection at different times, T0 and T1 (see (4) above). Now they use the two intersections as if they are two adjacent intersections. M2 drew dotted lines to connect the two intersections at 00:40:33. The labels “T0” and “T1” remained, but they became obsolete.

*12) seemingly important terms were sometimes not immediately recorded on the whiteboard*

The notions of “interior” road was first mentioned at 00:40:25. The “interior” and “onramp” properties were added to the “road” on the whiteboard sometime later at 00:42:55.

*13) the power of simplified diagrams*

M drew a simplified model (a box with inward and outward arrows in the four directions) of two interactions below the original two interaction pictures on the whiteboard (00:44:33). Immediately after drawing the simple model, they went back to the instruction and made sure that “you’ve got an intersection object which basically has four pointers to other intersection, possible intersections” (00:44:33).

*14) sharing the growing importance of the “cop” object*

At time 01:00:28, M said, “This cop is starting to sound very—I mean I can already see the length of the source file for the cop” with the gesture expressing the long lines of code. They both seemed to have agreed that the notion of “cop” now became the major part of the program.

*15) discussing implementation strategies*

Starting at the time around 01:01:09, M2 talked about the implementation strategies, by saying, “That should be—you don’t put the logic in the object, you have a traversal class that knows how to traverse, or the traverse is handled by a separate traverser object but then in hands-off control, based on the type of the object you’re traversing then that could be a way to kind of—because you’re talking about a kind of combination of a rule and an action, really depends on two things.”

*16) topic-change triggers thinking about erasing whiteboard drawings*

When they started talking about sketching user interfaces, they first tried to find things that they would erase on the whiteboard, but then decided would not, and started sketching a GUI window in the small space in the bottom of the whiteboard (01:04:46).

When they later changed the topic to discuss a user story, they erased the pictures of the two intersections on the whiteboard to create a space (01:19:09).

*17) gestures of interacting with interfaces drawn on the whiteboard*

While sketching a graphical user interface, M kept talking about how a user interacts with the system, such as, “So once you drag this one here you have to specify how that one’s connected, right? So as you drag, as you drag each intersection on it will snap and say okay I’m connected to here and to here, and it snaps. And here, and what not. And so then drag it and the it automatically connects and as you move this back and forth it increases the capacity for this road so now it” (01:05:45). While doing so, M made gestures pointing and touching the GUI window drawn on the whiteboard.

*18) a number of GUI ideas were not recorded on the whiteboard*

When they discussed about the user interface for the system, a number of ideas such as below were verbally described, but they were never written down on the whiteboard. “Right so this is now a UI piece and the dot, brings up your editor with frequency of, your frequency value. Maybe if you select a car, I mean a road, it allows you to edit. I wonder if we make the capacity of the geometry and you could sort of simulate a highway or something. ...” (01:11:52).

*19) rectifying the whiteboard contents*

When M2 stepped away from the room for a short break (01:21:03), M added “visitor” as a label to the “rule/actions” and “objects/rules” that had been written previously using a red pen. M used the time to rectify some writings and drawings on the whiteboard.

*20) redrawing/rewriting objects on the whiteboard to move them*

M copied a picture of the clock toward right to have more space (01:22:03).

*21) using natural language expressions to describe a user story*

When talking about a user story, M wrote in natural languages “as a user: - I want to ...” (01:22:10).

*22) discussions on user interfaces and those on models are intertwined*

When talking about “queuing and de-queuing of cars” for visualizing cars in the user interface (01:25:34), they started talking about the notion of “queue” and went back to talk about the model, such as “Would that be a property of the queue or a property of the car?” (01:26:44). The subsequent conversation for the several minutes kept going back and forth between two topics: the user interface visualization and the model.

*23) seeking for more appropriate expressions in writing a user story*

While working on a user story, M kept writing natural language sentences by verbalizing, for instance, “I want to evaluate success of each simulation” (01:31:56). M then pondered what expression he should use by saying, “... average capacity on the road or something? Is it capacity or average -- umm or umm.. capacity means like...” (01:32:04). M2 then said, “Well capacity is a good one to have because then you can evaluate wait time per, for a given capacity” (01:32:31). M kept the expression “average capacity” on the whiteboard.

*24) ephemeral drawings on the whiteboard (1)*

M drew a graph diagram when explaining the relationship between “capacity” and “wait time” (01:33:30). Then he erased the graph when he finished his explanation (01:34:02).

*25) suggesting more appropriate expressions in writing a user story*

When M wrote “I want to control the aspects of the intersection” on the whiteboard (01:35:20), M2 rephrased it by saying “I guess aspects of parameters of the simulation, really” (01:35:28). M changed the writing into “I want to control the parameters of simulation” (01:35:50).

*26) using a pseudo-code expression to describe the system*

In summarizing what they had designed, M2 wrote in a programming language like notation, “main( ) { n = new network, c = new clock. c.run(n) }” and said this is “at a very gross level” (01:37:50).

*27) referring to what had been written on the whiteboard as a checklist*

When started drawing an UML diagram, M2 erased the original intersection drawings, the UI sketches, and the UI stories that they had written in the middle of the whiteboard (01:40:00). The “model” written on the left side, and the “rules” written on the right side were kept on the whiteboard.

Then, when M2 was drawing the UML model (01:41:00-01:52:45(end)), both M and M2 kept referring to the objects listed under the model on the left side of the whiteboard.

*28) ephemeral drawings on the whiteboard (2)*

When M2 asked if “all the opposing lanes always the same state” (01:48:25), M drew a picture of a simplified intersection in a corner space trying to illustrate the real situation, and described that “it (what M2 said) is a simplified assumption” (01:48:29). M then immediately erased the intersection that he had just drawn.

*B. The AMBERPOINT Study*

*1) starting by drawing GUI*

They started by drawing a GUI sketch for “laying out the roads” and “visual map” (00:07:44). While sketching a GUI window, Jim verbalized what the role of this window would be, such as “to kind of tell you what your settings are for the individual intersections, and what kind of effect, like how much is the traffic backup at this light, or what's the average wait-time at this light” (00:08:53).

*2) asking for more appropriate expressions when naming objects*

When Jim started listing up “objects that they need to deal with” (00:09:20) on the left side of the whiteboard, he asked Ania, “are we going to call them streets or roads?” Ania responded by saying, “I guess roads, intersections” (00:09:31). Jim then wrote down “roads” on the whiteboard.

*3) Asking for more objects to list versus asking for more things to draw*

Ania asked a question “What else are we going to be drawing?” (00:09:33). Jim posed a question, “Are there other objects?” (00:10:52). Ania said, “Okay, so that's another thing we want to be able to specify in drawing” (00:11:23). They seem to be saying the same thing but stated differently; one in terms of objects to list, and the other in terms of things to draw in a GUI window.

*4) labeling things drawn in a GUI window so that one can talk about them*

When they talked about signals and left-hand turns, Jim labeled each horizontal road with “A,” “B,” “C,” and “D,” and each vertical road with “1st,” “2nd,” and “3rd” (00:12:06). Jim started calling an intersection by using the labels he put, for instance, “at the corner of A and 1st, there's a north approach” (00:12:31).

*5) detailing the content of a GUI window*

Jim started drawing how the “summary area” would look by writing down what information that they would like to see in the area (00:14:00). He listed all the roads names and intersections that were drawn on the left side map (00:14:50).

*6) not so much layout but the content may matter when sketching the GUI window*

When Jim added the “speed” column in the drawing of the “summary area” window, the section did not fit within the drawn square area and stuck out from the area toward right; but he left it there (00:14:50).

*7) asking for expressions in naming an instance of an object*

When the notion of “leg” emerged (00:15:46), Ania first asked, “what would be an easy way to name each leg” (00:15:51). Jim said, “if this is A1 and this is A2, the leg would

be A1 to A2” (00:16:01) and Jim drew “A1->A2” on the left side of whiteboard under the list of objects (00:16:20).

8) *better expressions are sought for not when writing it as an object name but when drawing the name within a GUI window*

Jim added “legs” above “A1->A2” in the left side of the whiteboard, and added “legs” to the interface summary area on the right side of the whiteboard. But then, he did not like the term “legs” and changed it to “blocks” on the left side of the whiteboard (00:16:30). He also changed “legs” to “blocks” on the right side beneath the summary area, drew a window-like square around the “blocks” that he just wrote, and listed possible block names, such as A1->A2, A1->B2, and so on (00:16:40).

9) *emerging sectioning of the whiteboard*

Now the whiteboard consisted of two parts. The thin left area listed objects and examples, and the remaining part consisted of the visual map area and the summary window area, which listed intersections, streets, and blocks, each with a complete list of roads displayed in the map area (00:17:33).

10) *showing gestures to describe the user’s interaction with the system (1)*

When discussing how a student would interact with the system, Ania used gestures interacting with the whiteboard, and told possible interaction steps, by saying, “so there is some sort of drawing pallet right, that says okay I have this thing I drag something, I’m drawing a road and I call it something and I draw and I call it B ...” (00:20:10). Ania used the object listed on the left side of the whiteboard as the palette, although they were the list of objects and not drawn as the palette.

11) *making an assumption about the given requirements*

As Jim said the following, the two have decided to rely on an assumption that there is a software package that provides the traffic simulation functionality: “I think we’re going to have to rely on professor E for creating the details about the theory of how that works. I mean, we might have to code whatever her understanding is of traffic light theory into it, right? But we don’t have enough information here. There might be packages out there, I don’t know.” (00:22:46).

12) *thinking of the client (Professor E) when making an assumption to simplify the model*

Talking about assuming the speed of each car, they have decided to assume that all the cars move with the same maximum speed by Jim saying, “We’ll talk to Professor E about it” (00:24:56).

13) *showing gestures to describe the user’s interaction with the system (2)*

When Jim was saying, “So the user can change where the intersections are, the distance between them by moving them around,” he moved his both hands and showed gesture expanding the road drawn in the user interface window on the whiteboard.

14) *becoming aware of a problem by walking through a user’s interaction with the currently drawn interface*

While Jim was talking about a student’s walking through interacting with the current user interface and how the information in the summary area would change (00:25:35), he had become aware of a possible problem with the summary window by saying that “So its funny because some of these things are editable and some of them just kind of give you details about how your edits made a change” and said, “so I think it’s a little confusing” (00:26:25).

15) *more contexts are added to a diagram later in the process*

At 28:17:00, Jim added the North-up sign above the map drawn in the GUI window sketched on the whiteboard.

16) *drawing in details*

In sketching out the pop-up window for an intersection to specify the signal behavior, they discussed details such as “2 minutes” for the default cycle duration of a signal (00:29:59), and “on average 40 seconds because with the green arrow” (00:30:42), and “So it would be 50 seconds at the top of the cycle followed by, I don’t know, 10 seconds of yellow, followed by how about 55 here” (00:31:00). Jim used a green pen for representing a green light, a red pen for a red light and so on.

17) *articulating design principles*

By realizing how complex the setting of the signals would be (“Gosh, who knew that this was so complex” (00:32:14)), Jim talked about design principles by saying, “Yeah, so the form needs to be flexible but easy; easy to fill out for a simple case and flexible enough that you could make it pretty complicated” (00:32:52).

18) *redrawing the area to afford more spaces for a GUI window*

Jim enlarged the pop-up window for the intersection toward right since the area got quite cramped with its contents. (00:33:40).

19) *becoming aware of a problem by articulating details*

Jim continued listing up all possible patterns of signals in the pop-up window by first finishing the N/S options and then the E/W options (00:34:06). While listing up, he had become aware that the signal has the state when both directions are red (00:34:30).

20) *not minding redrawing the details of the GUI window*

In response to the discussion on how a user would specify the duration of the cycle a signal, Jim erased all the inside area of the pop-up window where he had spent several minutes for listing up the conditions, and started re-writing them (00:36:30).

21) *redrawing the GUI details would help one come up with a new idea*

When started another attempt by saying, “So there’s 120 seconds we want broken into increments” in specifying the duration of a cycle of a signal, Jim came up with the idea of the “timeline” (00:37:26) representation.

22) *sketching a GUI window on whiteboard may not afford the intended resolution*

He put “5 sec” as an increment and started putting dots in a timeline drawn in the popup window, but as it became so small,

he changed to 10 sec increment. (00:37:50). It was not clear if this was a design decision or just a way to accommodate with his hand-drawing.

*23) paying attention to colors when sketching a GUI window on whiteboard*

While drawing the timeline, he used a green colored pen for a green light, a red pen for a red light, and a yellow pen for a yellow light. He asked for another color by saying, "that represents green arrow if we need to. Left-turn is orange, I don't know." He then got a purple pen (00:39:58).

*24) the whiteboard drawing as an important artifact*

At 00:48:35, Jim wanted to take a picture of the whiteboard. Jim said, "But I don't want to (erase it), that's always one of my concerns of writing on a whiteboard is that I've got some great ideas on here but now I need to reshuffle a little bit, and so I would just pull out a camera" (00:48:47). He actually did not erase anything right away. Taking the picture of the whiteboard was not necessarily for erasing something on the whiteboard but was for him for the purpose of recording the snapshot of the important artifact that he had been working on.

*25) wanting to ask the goal of the system to the client*

At 00:52:38, Ania posed a question by asking what the goal of the simulation is. They both came up with a number of possible answers to the question, but they both wanted to talk to the client, Professor E, by saying, "I think we have to talk to Professor E again, because she didn't specify the most important aspect of it, which is how do you know that your lighting system is working to its best potential?" (00:55:16).

*26) worrying about the next version of the system*

Ania demonstrated a way to design for future by saying, "I don't know if Professor E really wants that level of detail but, you know, we should think about that because in the next version of the product we may want to incorporate that" (01:02:10).

*27) planning how to section and use the whiteboard*

At 01:06:15, Jim planned for how to further use the whiteboard by saying, "Let's build like a small UI over here that we can show to the developers, and then we have some room to do like an ER diagram over here." He then decided to erase the summary window area because "It's kind of sketchy" (Jim) (01:06:20) and "it doesn't mean anything right now" (Ania) (01:06:30).

*28) revising "modes" when redrawing GUI windows*

Jim was redrawing a map window and a window for setting up the intersection, and was about to draw a window for car/traffic configuration while saying that these were "editing and setting modes." Ania then wanted to have "the map drawing as one thing" but the intersection specification and the car specification "as a part of the simulation" (01:18:49). The two modes now became "building map" and "building simulation" (01:21:35).

*29) using different part of the whiteboard to check the GUI window drawings*

Jim used the "set up" points he wrote in the top of the whiteboard to check if the interface drawings covered all (01:37:00).

*30) writing down as an object uncovers not-so-clear concept*

When Jim draw a boxed "Approach" on the whiteboard in the ER diagram, Ania asked a question, "what is approach?" (01:39:01). Jim then explained that each intersection has four approaches (01:39:03). Jim actually wrote "intersection approach" for the first time on the whiteboard very early in the process (00:12:45) and used the term for several times afterward, but never got questioned until this moment.

*C. The ANONYMOUS Study*

*1) start by sharing understanding about the task*

Male 1 first wanted to hear Male 2's summary on the given problem description. (00:05:29)

*2) sharing the envisioned process*

Male 1 claimed that this is going to be a "pretty classic software design issue" (00:07:10). He tried to describe a process that he would like to take by saying that they have to think about UI and "underlying data structures supporting this" (00:07:30).

*3) starting with "data" names*

Male 1 first wrote "Data" on the whiteboard, and underlined it. He then added "signals," "roads," and "cars (traffic)" (00:08:59).

*4) draw a diagram to understand the world to model*

Male 1 drew a picture of an intersection (i.e., crossing roads) and adds visual marks corresponding to the components of "Data" that he had just wrote, such as cars and signals, while talking about how they were related to one another (00:01:00).

*5) lines represent hierarchies*

In talking about the "hierarchy," Male 1 draw a line between "interaction" and "signal" indicating that there is a hierarchical relationship between the two.

*6) a term to be avoided to simplify a model but kept emerging*

When Male 1 drew a line between "roads" and "lane" by saying, "Roads have lanes" (00:11:01), Male 2 asked if they cared about details such as lanes. Then Male 2 concluded by saying, "I don't think we need to make assumption of lanes, period" (00:12:11). The notion of "lane," however, kept appearing in the subsequent discussions during the design process (see (11), (12) and (19), for instance) until the time when they decided to view lanes as "paths towards left, straight and right" (00:49:08).

*7) using a diagram representation of the world to talk about a model for the world*

They add arrows and little marks on the intersection diagram they had drawn when talking about Data. As discussing, they wrote "Abstract," underlined it, wrote "rules" beneath "Abstract" (00:13:00), and drew a line between rules and signals. As they talk about the objects listed under "Data" and "Abstracts" and their relationships by adding lines, they tried to understand the world of traffic system as Male 1 said, "So we kind of tinkering around with this real-world model" (00:14:41).



*8) articulating a meta-process*

Male 1 summarized what they had just done, and commented on the process by saying, "I like jumping into details and then coming up and saying okay now what do we really need here, what are we seeing here" (00:14:51).

*9) sharing the key aspect of the design*

Male 1 stated, "designing intersection is kind of the key thing" (00:17:04).

*10) growing diagram to represent more complex aspect of the world*

When talking about the notion of the "length" of the "road," Male 1 drew another intersection on the right of the initial intersection and connected them with the in-between road to illustrate the length of the road segment between the intersections (00:19:15).

*11) incrementally refining the "data" objects on the whiteboard*

When finished discussing about the length of the road in front of the pictures of the two intersections on the whiteboard, Male 1 walked toward the left of the whiteboard where "Roads" is listed as "Data." He then annotated "Lane" connected to "Roads" with a cross in red by saying, "So we've decided that lanes, maybe not so significant but length is going to be significant" (00:19:38) (see (6) for more on "lanes"). He continued saying, "And it's significant because it holds a number of cars" (00:19:40), as he drew a red line starting from "Roads" and wrote "Length" and "# of cars."

*12) fostering an assumption that would simplify the model*

Male 1 still wondered about "lanes" by saying, "I'm still wondering about the left-turn lane" (00:20:21). After describing why having lanes affect the complexity of the problem, he claimed to have "a protected left" to "go with the minimum" (00:21:03) in taking the simplest assumption. The two continued discussing how having lanes affects the traffic simulation for a few more minutes. Male 1 then expressed his concern by saying, "I hate to bog down in that detail." Male 1 wanted to make a simplifying assumption. They kept talking about what assumption would make the simplest model.

*13) using the content drawn on the whiteboard in reflecting on the design process*

The two kept looking at the whiteboard and remained silent for about a minute (00:28:30).

*14) using a different color pen for the whiteboard in a different phase of the process*

When they started thinking about "container," Male 1 wrote "map" and underlined it with a green pen in the top part of the whiteboard (00:30:50). He then circled "interactions" and "roads" connected from "map" (00:31:07) also in green. Male 1 kept using a green pen and drew two circles and a connecting line, and labeled them "I1," "I2," and "P1," respectively (00:33:50). He then said, "how do you define the behavior of the traffic" (00:34:30). He then wrote "behavior" in the bottom part of the whiteboard in green as if it would serve as a reminder to rethink later the things written in green (00:34:50).

*15) reflection happens when looking at what has been being written on the whiteboard; immediate reaction might change after a few seconds pondering*

After talking about making "roads" own traffics, Male 1 said, "so the road is an object" (00:36:08) and wrote as operations "add cards" in green next to "roads" (00:36:25). Male 2 initially nodded for a few seconds expressing his agreement. But then, Male 2 murmured, "Would it add a car" (00:36:28). Immediately after that Male 1 echoed with Male 2 by saying, "Yeah I'm trying to get--how do we (add car)?" (00:36:29).

*16) growing a diagram by adding labels and objects to depict the concept of focus*

When Male 2 was talking about how input traffic would flow cars into the road, he wrote "input traffic" and drew a circle on the right side of the intersection picture Male 1 previously had drawn. Then Male 2 drew an arrow from the circle into a road, which was a part of the intersection (00:36:40). Male 1 then wrote "R1" as a label to the road, and he used "R1" in his saying, "lets say there's I'm thinking the box they're using they click, they're going to define a flow into R1 and then they're going to click go" (00:36:55).

*17) using a metaphor to describe complex behavior*

When wondering what would be "start" and "stop" of simulation, Male 1 used "faucet" as a metaphor to describe the behavior of the controller as in the following "I'd imagine some guy turning on a faucet. When the sink's full, turn it off, let it drain down, or tell me there's more room I'll turn it on again" (00:41:35). This metaphor was a comprehensive one and Male 2 completely agreed with Male 1 by saying, "Perfect okay, yeah that makes sense" (00:41:58).

*18) labeling diagram to talk about objects*

When discussing how an intersection behaves with roads, Male 1 labeled the four roads of the intersection he initially drew with "R1," "R2," "R3," and "R4." He then used the labels to describe the behavior of the intersection such as in "The intersection is going to say, okay my light is green, R1--I mean it's green for thirty seconds--R1 give me ten cars, we can calculate that, and put them in R3" (00:46:56). He had already written "R1" on a different road of the adjacent intersection at 00:36:55 so there now existed two R1's in the single drawing.

*19) resolving the issue of "lanes"*

When Male 1 went back to where he wrote "roads" under "Data," he adds "s" to "lane" connected to "roads" to make "lanes" (00:48:59). He then wrote "L/S/R" near the "lanes" with Male 2 saying, "More paths, kind of" (00:49:08). This way they seemed to have agreed that lanes are like "paths" for each of the direction (left, straight, right) (see (6), (11) and (12)).

*20) writing down a not-so-sure-about object on the whiteboard so that they will come back later*

When they went back to working on the "rules" of signals under "Abstract," Male 2 decided to write "left/straight/right" (00:51:25) under "time," "sensor," and "red/yellow/green" after wondering a little bit. Male 1 agreed with what Male 2 had just done by saying, "Yeah let's put it in there and we'll capture that" (00:51:17).



21) *text objects written on the whiteboard are transformed into different objects by adding a prefix.*

When they were re-examining “rules” of signals, they discussed the relation between “time” and “red/yellow/green,” both of which had been written under “rules” (00:51:32). Male 2 said, “there should be a time component attached to each one of these things” by pointing to “red/yellow/green.” Then Male 1 added “t-” to each of “red/yellow/green” to create “t-red/t-yellow/t-green” by saying, “there’s a time for each of those. t-R, t-Yellow, t-Green” (00:52:09). Thus, “red/yellow/green,” which was originally written to mean something to do with rules related to the three options, had become to mean the three time durations for the red, yellow and green lights when each was prefixed with “t-.”

22) *going back to the problem description*

Male 1 brought up the topic of “sensor,” which had also been written beneath “rules.” They both went back to the problem description to see what was written about the sensor.

23) *reflecting on what they have written on the whiteboard*

Male 1 claimed, “this is the model which we build the end user application” (00:56:55).

24) *abstracting out the user interaction*

In talking about an end-user application, they both agreed to have a “drag and drop environment” (00:57:06) and use “a classic drawing package interface” (00:57:28). They did not draw any GUI window or a map visualization on the whiteboard.

#### IV. DISCUSSIONS

The three teams being studied demonstrate a wide variety of activities. It is not the approaches they have taken were different, but what they have designed and aimed to design were different; that is, the aspects of software design they have worked on were different. The Amberpoint team worked on the interaction design of the system, but not on the simulation engine. The Anonymous team worked on the underlying substrate models of the simulation mechanism, but little on how a user would interact with the system. The Adobe team worked on the both, ranging from simulation engines to user stories; but the level of details they worked on were different from those of the other two teams. The very variety of the three teams may be the reflections of the wide range of activities software design involves, and of the different levels of details that need to be covered by software design.

We compared the names of the objects that each design team presented in their reporting sessions. Some terms overlap, but some are unique, such as “Cop” by the Adobe team, “Configuration” by the Amberpoint team, and the “Input” and “Output” by the Anonymous team. We used our CProVA tool to visualize where each of such unique terms was mentioned during their design processes and found no consistent patterns.

The Amberpoint team displayed a great deal of concerns on understanding the goal of the system with respect to the client's purpose. The phrases “professor” and “professor E” were mentioned only by the Amberpoint team and not by the other two teams. In contrast, the term “user” was used only once by

the Anonymous team (in “end-user application”), which was quite distinctive comparing to the other two teams.

The Anonymous team spent a large amount of their time in modeling the “real” world and making assumptions that would simplify the model. Given the problem requirements that the client wants to have “traffic simulation,” they spent a long time discussing how the traffic operates in the real world, and how to come up with a model that is simple to handle but fulfills the requirements. Their struggle in dealing with “lanes” of “road” typifies the issue. We may argue that such assumptions should have been articulated in the requirement in the first place, but this in fact seems to be quite typical. What they tried in the study seemed to us not only analyzing the requirements but “designing” the requirements.

The Amberpoint team demonstrated the use of the whiteboard as a medium to represent the artifacts that they have designed. As Ania pointed out in the video, Jim seems to have strong expertise in using whiteboard. He often planned which part of the whiteboard to use for what purposes. When he sketched GUI windows in the study, he almost always enumerated all the possible elements and seldom omitted details. He did not seem to mind redrawing different versions of the GUI ideas again and again.

The Adobe team and the Anonymous team demonstrated the effective use of metaphors in understanding the complex behavior of an “intersection.” The Adobe team used the term “cop” to illustrate how an intersection handles cars flowing from a road to another. The notion of “cop” became the object in their design. The Anonymous team used the term “faucet” to understand the incoming flows of traffic into a road. The term “faucet” did not become an object name, but this metaphor was helpful for the both members to understand what “Add Cars” operation would do to a “Road.”

In analyzing the data we obtained in this study, we spent a long time in analyzing what has been written and drawn on the whiteboard by carefully watching the video data. Each stroke was generated or removed on the whiteboard at a certain time in a certain context. We have found how the diagrams drawn on the whiteboard evolve over time, which is not necessarily consecutive but intermittent, for instance, by adding labels to verbally refer to some parts of the diagram, or by adding more graphic representations to re-appropriate it for the purpose that is different from the original. It would have been a rich resource of data if the timestamped stroke data on the whiteboard had been captured. We would extend the CProVA tool by incorporating the stroke-viewing tool such as ART019, which allows a user to interact with stroke data through their locations and timestamps.

A set of findings we have listed in the previous section seems to address a wide range of potentially interesting research issues. The lists are related to vocabulary elements, temporal and logical flows of events, and visual elements. The lists involve the uses of a variety of representations, including terms and phrases, gestures, layout, colors, diagrams and marks, as well as the evolutions of such representations over the course of a design process. A next step in our study would be to classify these findings and identify dimensions that would help us characterize the nature of software design processes.