**Visual Analytics Interaction log pattern Visualization   
supporting Recover and Reuse**

Youngtaek Kim(김영택), Yieun Kim (김이은)

[ytaek.kim@hcil.snu.ac.kr](mailto:ytaek.kim@hcil.snu.ac.kr), [yekim@hcil.snu.ac.kr](mailto:yekim@hcil.snu.ac.kr), [yuhoon.ki@samsung.com](mailto:yuhoon.ki@samsung.com)

Abstract

Visual Analytics(VA) allows people to analyze data when the don't know exactly what questions they need to ask in advance. In case of well-defined questions, purely computational techniques and proper algorithm will be enough. However, if the problem is ill specified, they explore raw data in various and repetitive way until good questions and answers are found. VA tools such as excel, Tableu, spotfire and so on help them to work effectively. During exploring process, repetitive and tedious works occurred. For decreasing duplicate jobs, most VA tools provide several functions such as saving current status or templates. But as far as we know, these functions are limited to mainly visual encoding idioms without interactions. In this proposal, we introduce interactions tracking and reproducing techniques.

**Keywords**: interaction log, visualization, recover and reuse

**Index Terms**: ~~K.6.1 [Management of Computing and Information Systems]: Project and People Management—Life Cycle; K.7.m [The Computing Profession]: Miscellaneous—Ethics~~

# Introduction

Sharing and reusing each person’s insight and reasoning process obtained from Visual Analytics (VA) is regarded quite difficult job. Clear and well-defined problems can be solved by purely computational techniques and sharing the analyzed results is also possible. However, in case of ill-defined problems or not knowing what questions required to be asked in advance, enabling reasoning process during Visual Analytics to be recovered and reused is necessary. Existing VA Tools such as Excel, Spotfire, Tableau and so on provide history functions but it is just recall of linear history of manipulating the tools. In this proposal, our contributions are as below. - Define practical and reusable visual encoding/interaction logging pattern - Visualize captured patterns and provide recover & replace interfaces with a simple prototype application.

SWE 도메인에서의 데이터는 이러이러한데, 무제가 이리하다. .그래서 이걸 쓰겠따.

# Related Works

[1] introduces five interrelated states in the research of visual analytics provenance: perceive, capture, encode, recover and reuse. In “Perceive” stage, the understanding of how information is perceived by the user. “Capture” stage means tracking and logging not only literally a user’s linear sequence of actions but related semantic information. “Encode” refers to describing the captured provenance in predefined formats. Captured and encoded the user’s provenance enables to be “recovered” in the same manner and also “reused” by reapplying it to a new data or domain. In the implementation of this project, we treat mainly “recover” and “reuse” stage due to the lack of time and resource. Regarding “Capture” and “Encode”, we will suppose that users’ interaction logging data exists and then manipulate it manually. As the result, the visualization and interactions of recovering and reusing stage will be developed as a rapid prototype of this project

.

GraphTrail [2], which is an interactive visualization for analyzing multivariate and heterogeneous networks, provides functions that capture users’ interactions and integrates this history directly. Captured users’ interaction history from a drag-and-drop action to overall exploration enables to recall the insight of analyzed result. However, the captured history provides ordered exploration path with pre-defined user interactions but not encoded or abstracted interaction pattern. M. Pohl et al. [3] analyzed interaction sequence. They categorized user activities: select, explore, reconfigure, encode, abstract/elaborate, filter and connect. By the analysis of log files and thinking aloud method interaction sequence logs are extracted with transitional probabilities.

# Interaction Logging and Encoding

In this section, we describe how to capture users’ interaction logs and encode them to a specific format supporting recover and reuse. Interaction log data used here is manually gathered or created regarding limited time duration of this project.

## Interaction logs

Analyzing software engineering data descripted in introduction, three participants (two senior developers, one junior developer) used commercial visual analytic tools (Spotfire, Tableau). In order to capture their interaction logs during analysis, the participants were encouraged to think aloud and write down their interactions on notes manually. The thinking aloud method was mainly intended to provide separation of each interaction and memorize serial interactions. An example of interaction logs written on note is figure 1.



1. Two boxes. One filled with confetti.

## Encoding interaction logs

Manually written interaction logs are necessary to be encoded as specific format which our visualization system is able to recognize. Instead of converting logs to the format directly, we first convert the logs to intermediate representation(IR) as modern compiler technique does [4]. The advantages of using IR are as follows.

* Common abstractions for diversity of the way each participant writing interaction logs
* Extensibility for coping with different visual analytic tools.
* Applying different encoding scheme for recover & reuse according to domain or purpose (in this project, we use only one scheme)

The structure of IR is depicted in figure 2. Basically, interaction category refers to Tamara’s visualization interaction idiom [5]. IR is converted to final format for visualization according to encoding scheme. Details of the format is described next section.



1. Two boxes. One filled with confetti.

After converting logs to IRs, we encoded it to a structured format for visualization. Whole reasoning process can be lengthy and complex, hence it is necessary to restructure sequential IRs as two level hierarchy: anchor and details. We first mark an anchor on the IR when meaningful change occurred as follows:

* Creation or alternation of kind of chart.
* Data or range of data.

(For example, when an IR indicates that a bar chart is changed to a line chart or specific range of data is filtered, it goes to an anchor.) Then IRs, which are not anchors, are aggregated to previous anchor sequentially as figure 3.



1. Two boxes. One filled with confetti.
2. 큰 단위로 한번 묶는다. (끊고)
3. Interaction 간의 path가 있는 것처럼 anchor 간에도 path가 있겠지.

anchor and details

Plain interaction log를 계층적 구조로 변경 -> 왜? … 일단 길고.

Va의 reasoning process를 잘 이해하기 위해 anchor, detail interactino으로 구분. 요약의 기능도 보유

Anchor, detail로 나누는 기준: 화면에 보여지는 데이터의 변경이 있을경우, chart의 종류가 아예 달라지는 경우. 예를 들어…

Log를 1차적 IR로 변형한 이후에,

### Subsection Two

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exercit­ation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulpu-tate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blan-dit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exercit­ation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulpu-tate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blan-dit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi.

# reuse recover

# Discussion

Future work으로 interaction ?”] bg gtttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttlog IR을 분석하는 방법을 빅데이터로 할 수도 있을 듯l;.

Rule을 추가할 수도 있고, /.

Cross project로 해볼 수도 있을 것 같다.

# Conclusion

# Conclusion

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exercit­ation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulpu-tate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blan-dit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exercit­ation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulpu-tate velit esse molestie consequat.

References

1. C. North et al., “Analytic Provenance: Process + Interaction + Insight,” 29th Annu. CHI Conf. Hum. Factors Comput. Syst. CHI 2011, pp. 33–36, 2011.
2. Dunne, C., Henry Riche, N., Lee, B., Metoyer, R., & Robertson, G. (2012). GraphTrail: analyzing large multivariate, heterogeneous networks while supporting exploration history. CHI, Human Factors in Computing Systems, 1663-1663-1672–1672.
3. Pohl, M., Wallner, G., & Kriglstein, S. (2016). Using lag-sequential analysis for understanding interaction sequences in visualizations. International Journal of Human Computer Studies, 96, 54–66.
4. LLVM
5. Tamara