

MVC-based Graphical User Interfaces using reactive programming

Bruno Rocha Pereira - 529512

October 16, 2015

1 Introduction

Since the title of this specific subject regroups different subtopics with different related literatures, the subtopics' litterature will be dealt with in different paragraphs. However some of them have already been linked together in articles.

2 MVC

The first part of the main complete topic is the study of the MVC framework.

The initial document written about the MVC model was "Thing-Model-View-Editor" [Reenskaug, 1979b]. Reenskaug presents in this paper what is going to be transformed into the MVC model that is currently known and used. It is a must-read to start studying the MVC topic since the article deals with its roots. His idea is clarified in his next paper [Reenskaug, 1979a], in which he gives precise definitions of the terms Model, View, Controller.

This concept is vastly enriched in another article published almost ten years later [Krasner and Pope, 1988] that gives a practical application of the MVC paradigm using an object-oriented programming language. This cookbook also provides an explicit link between MVC and the usage of a user interface, which is also a part of the subject studied here. Reenskaug, also wrote a more recent article giving a topical overview over the MVC framework [Reenskaug, 2003].

The article "A comparison of model view controller and model view presenter" [Qureshi and Sabir, 2014] has to be read to determine the technical advantages and disadvantages of the MVC pattern.

"Structures and interactions"'s interest [Nowack, 1999] is that the inside of the architecture is explained and compared to other architectural patterns.

A few applications of the concept of MVC may help study it. The book "The Definitive Guide to Java Swing" [Zukowski, 2005] can be looked into to see an example of it in an generally know language.

The lack of recent articles about MVC denotates that MVC research is currently not in active research.

3 Reactive Programming

The study of reactive programming starts with a historical research about the first articles written linked with the reactive logic.

In order to understand and really master what reactive programming is, a mastery of data-flow programming is compulsory. "First version of a data flow procedure language" [Dennis, 1974] was the starting article about the data-flow concept, therefore being an interesting article to start with. It introduces the basic ideas of what a dataflow-based language should be like. In addition to that, a look into "A history of data-flow languages" [Whiting et al., 1994] provides information about the whole history of data-flow languages and their evolution, hence presenting to the reader the mistakes not to reiterate.

Reactive programming can also be seen as a pipeline implementation. Therefore, researching articles concerning the research on data pipelines is necessary. The first articles were mainly written or contributed to by Andreas Buja. "A data viewer for multivariate data" [Buja et al., 1987] and "Elements

of a viewing pipeline for data analysis” [Buja et al., 1988] were the first two articles to describe the implementation of data as contained in pipelines for real-time graphics. ”The plumbing of interactive graphics” is a more recent one, that uses data pipelines for interactive graphics.

Reactive programming knowledge has been summarized recently in the article ”A Survey on Reactive Programming” [Bainomugisha et al., 2013]. This article is useful since it regroups the current knowledge about reactive programming and can give a quick overview on the subject.

A link between user interfaces and reactive programming can be found in a recent article called ”Reactive Programming for Interactive Graphics”. [Xie et al., 2014]. This article is one of the most important for this research since it combines every subtopic our main topic can be divided into : MVC, GUI and reactive programming.

Zhanyong Wan and Paul Hudak have written several articles about FRP(Fonctionnal Reactive Programming) starting from the basics with ”Functional reactive programming from first principles” [Wan and Hudak, 2000]. It explains what reactive programming is, using functional programming languages. An advantage that this article has over other articles is that it gives a limit that FRP has. ”Event-Driven FRP” [Wan et al., 2002] was another article written by these two authors. As its name indicates, this one is more focused on events. It has to be considered since events is an important concept in interactive user interfaces. The two authors have also contributed to ”Real-time FRP” [Wan et al., 2001]. Their aim in it is to accelerate FRP programs to allow them to run in RT environments.

Another article combining FRP and GUI is ”Elm: Concurrent FRP for Functional GUIs” [Czaplicki, 2012]. It uses a new language called *Elm* to propose a new approach to easy reactive GUI with functional programming languages. Elm is now one of the main FRP still in development, which makes this article important to consider. Its author adds more content to it in another interesting article called ”Asynchronous functional reactive programming for GUIs” [Czaplicki and Chong, 2013]. An application of *Elm*, with code included is shown in the article [Krautmann and Kindermann, 2015]. Eventhough it’s applied to making games, the logic stays the same for any software and the fact that this article is recent contributes to demonstrate that FRP is still an active field of research.

References

- [Bainomugisha et al., 2013] Bainomugisha, E., Carreton, A., Cutsem, T., Mostinckx, S., and Meuter, W. (2013). A survey on reactive programming. *ACM Computing Surveys (CSUR)*, 45(4):1–34.
- [Buja et al., 1988] Buja, A., Asimov, D., Hurley, C., and McDonald, J. A. (1988). Elements of a viewing pipeline for data analysis. *Dynamic graphics for statistics*, pages 277–308.
- [Buja et al., 1987] Buja, A., Hurley, C., and MCDONALD, J. (1987). A data viewer for multivariate data. In *Colorado State Univ, Computer Science and Statistics. Proceedings of the 18 th Symposium on the Interface p 171-174(SEE N 89-13901 05-60)*.
- [Czaplicki, 2012] Czaplicki, E. (2012). Elm: Concurrent frp for functional guis. *Senior thesis, Harvard University*.
- [Czaplicki and Chong, 2013] Czaplicki, E. and Chong, S. (2013). Asynchronous functional reactive programming for guis. In *ACM SIGPLAN Notices*, volume 48, pages 411–422. ACM.
- [Dennis, 1974] Dennis, J. (1974). First version of a data flow procedure language. In Robinet, B., editor, *Programming Symposium*, volume 19 of *Lecture Notes in Computer Science*, pages 362–376. Springer Berlin Heidelberg.
- [Krautmann and Kindermann, 2015] Krautmann, D. and Kindermann, P. (2015). Functional reactive programming and its application in functional game programming.
- [Krasner and Pope, 1988] Krasner, G. E. and Pope, S. T. (1988). A cookbook for using the model-view controller user interface paradigm in smalltalk-80. *J. Object Oriented Program.*, 1(3):26–49.
- [Nowack, 1999] Nowack, P. (1999). Structures and interactions.
- [Qureshi and Sabir, 2014] Qureshi, M. R. J. and Sabir, F. (2014). A comparison of model view controller and model view presenter. *CoRR*, abs/1408.5786.

- [Reenskaug, 1979a] Reenskaug, T. (1979a). Models-views-controllers. *Technical note, Xerox PARC*, 32(55):6–2.
- [Reenskaug, 1979b] Reenskaug, T. (1979b). Thing-model-view-editor—an example from a planning system. *technical note, Xerox Parc*.
- [Reenskaug, 2003] Reenskaug, T. (2003). The model-view-controller (mvc) its past and present. *University of Oslo Draft*.
- [Wan and Hudak, 2000] Wan, Z. and Hudak, P. (2000). Functional reactive programming from first principles. *SIGPLAN Not.*, 35(5):242–252.
- [Wan et al., 2001] Wan, Z., Taha, W., and Hudak, P. (2001). Real-time frp. In *ACM SIGPLAN Notices*, volume 36, pages 146–156. ACM.
- [Wan et al., 2002] Wan, Z., Taha, W., and Hudak, P. (2002). Event-driven frp. In *Practical Aspects of Declarative Languages*, pages 155–172. Springer.
- [Whiting et al., 1994] Whiting, P. G. et al. (1994). A history of data-flow languages. *Annals of the History of Computing, IEEE*, 16(4):38–59.
- [Xie et al., 2014] Xie, Y., Hofmann, H., and Cheng, X. (2014). Reactive programming for interactive graphics. *CoRR*, abs/1409.7256.
- [Zukowski, 2005] Zukowski, J. (2005). The model-view-controller architecture. In *The Definitive Guide to Java Swing*, pages 59–65. Apress.