

MVC Dance: Connecting Software development and Corporeality from Agile Process and Pattern Language perspectives

Hanyuda Eiti
Mamezou, Co., Ltd.
hanyuda@mamezou.com

Abstract

This paper describes the author's view on software development in the future from agile process and pattern language perspective. It places much value on interactions among stakeholders so we need a tool for consensus building on architecture through workshop. We propose a MVC dance and other pattern dances for architecture understanding and educational purposes.

1. Current Software Development methods

This paper focuses the agile software process and compares it to a pattern language concept proposed in architecture and city planning field. UML(Unified Modeling Language), Object-Oriented design and Java are not enough. Software developments has difficulties in totally different areas:

- Users don't know what they really want as a system.
- System's requirements are likely to change.
- We have to design and verify open systems on a heterogeneous middlewares, platforms and technologies.

2. Agile Process and Pattern Language

To solve these problems, agile development process is propose. Agile process insists following points:

- A) Individuals and interactions over *processes and tools*.
- B) Working software over *comprehensive documentation*.
- C) Customer collaboration over *contract negotiation*.
- D) Responding to change over *following a plan*.

On the other hand, software pattern concept is now widely recognized effective in designing object-oriented software architecture. We have noticed the similarity between agile process and pattern language concept in that: "iterative and lifelike model based verification." At the same time, the weaknesses of agile process, requirement acquisition and consensus building, are covered by pattern language workshops. To promote the requirement acquisition and consensus

building processes, workshop participants have to be able to feel the target system architecture through their corporealities.

3. Mvc Dance and other pattern dances

So, we propose pattern dances adequate for understanding the structure and dynamics of the software architecture through participants' bodies movements. The dance is called MVC(standing for Model-View-Controller) dance, which is played by several dancers who play each architecture component. We can learn the collaboration patterns among components from playing and viewing the dance.

MVC pattern is an application of Observer and Mediator patterns, which separate the model objects from their UI and other decoration objects so that developer can easily extend and maintain the pure domain model and other layers independently. It means there goes 2 different plays: M's and VC's on a stage.

A single dancer plays a model and a VC pair dances together which is attached to the model separately and dynamically no matter who other VC are dancing aside.

This style of dance is applicable to many other design patterns and software architectures: *factory, singleton, facade, adopter, mediator, observer, proxy, decorator(wrapper)* and others. We have also verified that the pattern dance is a good educational method for beginners to learn object-oriented software design and architecture through their body-plays.

4. References

- [1] "Let's dance MVC together," @IT, 2003. (in Japanese) <http://www.atmarkit.co.jp/news/200305/24/pattern.html>
- [2] Alistair Cockburn, *Agile Software Development*, Addison-Wesley, 2001.
- [3] Christopher Alexander et al, *A Pattern Language: Towns, Buildings, Construction*. Oxford University Press, 1977.