* The topic of this video is related to our course material, which is the importance of complexity in software architecture and its design. In last week’s lecture about modern web architectures, there is an interesting fact when introducing different versions of web architecture: in web architecture 1.5+ and web architecture 2.0, complexity is mentioned in both versions and it is defined as a disadvantage. In web architecture 1.5+ the complexity is due to the configuration of framework and in 2.0 it is because of the programming requirement. However, in web architecture 2.0+, complexity is mentioned as an advantage. It is because of the robust framework and the characteristics of JavaScript. Therefore, this talk about complexity has good contents for me to understand the background and things in the lecture.
* Although the speaker used a lot of examples to enhance the credibility of the facts that produce complexity, such as quotation and video in invisible changes, code examples in low level concurrency and imperative style, etc, he didn’t provide any solution for reducing complexity. For example, he could use some real applications as examples to tell the audiences if those applications lack cohesion, if not, how do they manage to do it, what can I learn from it, etc. Another example could be the code examples in low level concurrency and imperative style, he could use those codes as examples, refactor the code or talk about the changes of the framework, in order to tell the audiences how to deal with the complexity produced by these two facts.
* I like the words: “We should learn to deal with complexity and have the wisdom to minimize it”. First it defines complexity as an inevitable thing that exists with all software design, there is no such architecture that has perfect complexity so all we can do is to minimize it. Secondly, it states an expectation that fits the design rule high cohesion and loose coupling, which is to minimize the complexity of the design. Thirdly, the word wisdom tells us it is human’s job - no matter designer or programmer, to minimize the complexity: because the software cannot deal with it during the development and execution. This is a meaningful summary that concludes the situation and human duties.
* Enjoyed it, and I think it complemented the course. It is an interesting video that helps me understand what complexity is, where it comes from and how to understand the complexity as disadvantage and advantage mentioned in the lecture notes.