Reading Feedback 3

Name: Dazhi Li

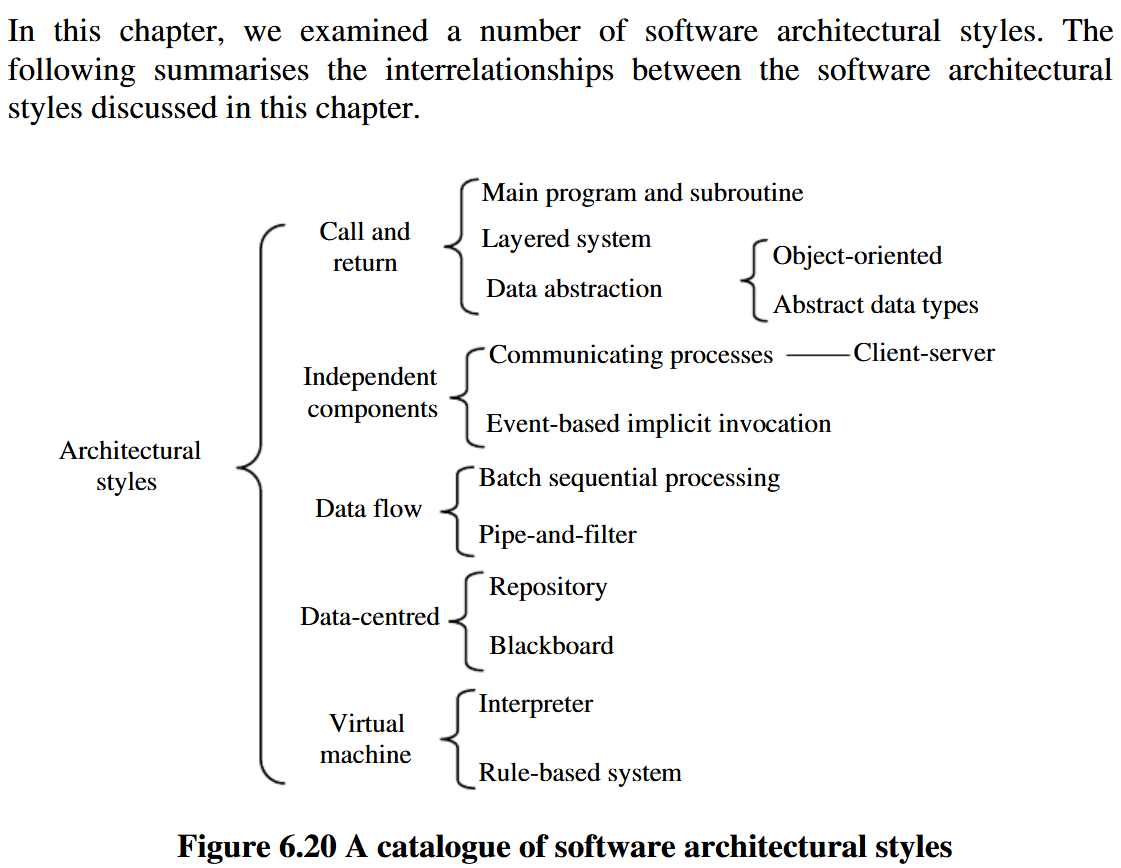
**One sentence conclusion:**

It is very clear to learn those typical architectural styles from fixed perspectives like computational model, structural patterns and properties of them. I am a little bit confused on using styles on design. It provided some metrics on comparison of different styles, but what is more important than other metrics? And chapter 15 describes interfaces very clearly from scope, representation to error handling. Moreover, I think chapter 16 is really inspiring for me as I got work experience in cloud company. Especially the creating VM instances examples. Hedged requests give me a sight on how software engineer neglects the hardware facts that different requests could be responded with different time.

# Software Design Methodology

## Chapter 6 page 168

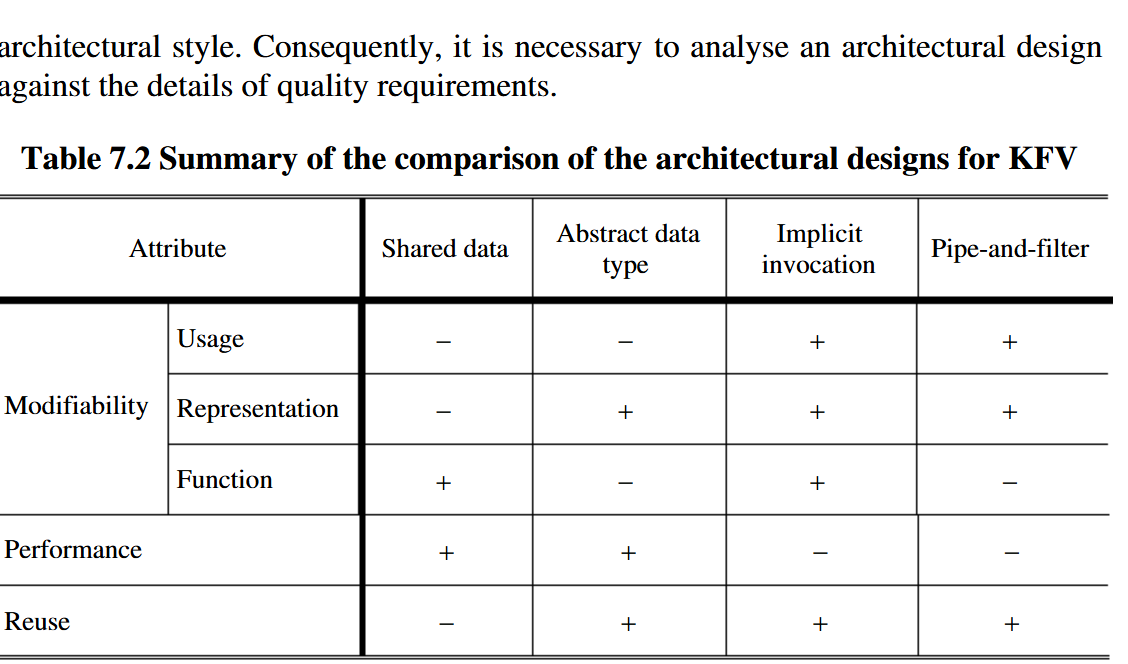
Clear



Very clear on what architectural styles I went through with specific sub-types of these styles.

## Chapter 7 page 193

Muddy

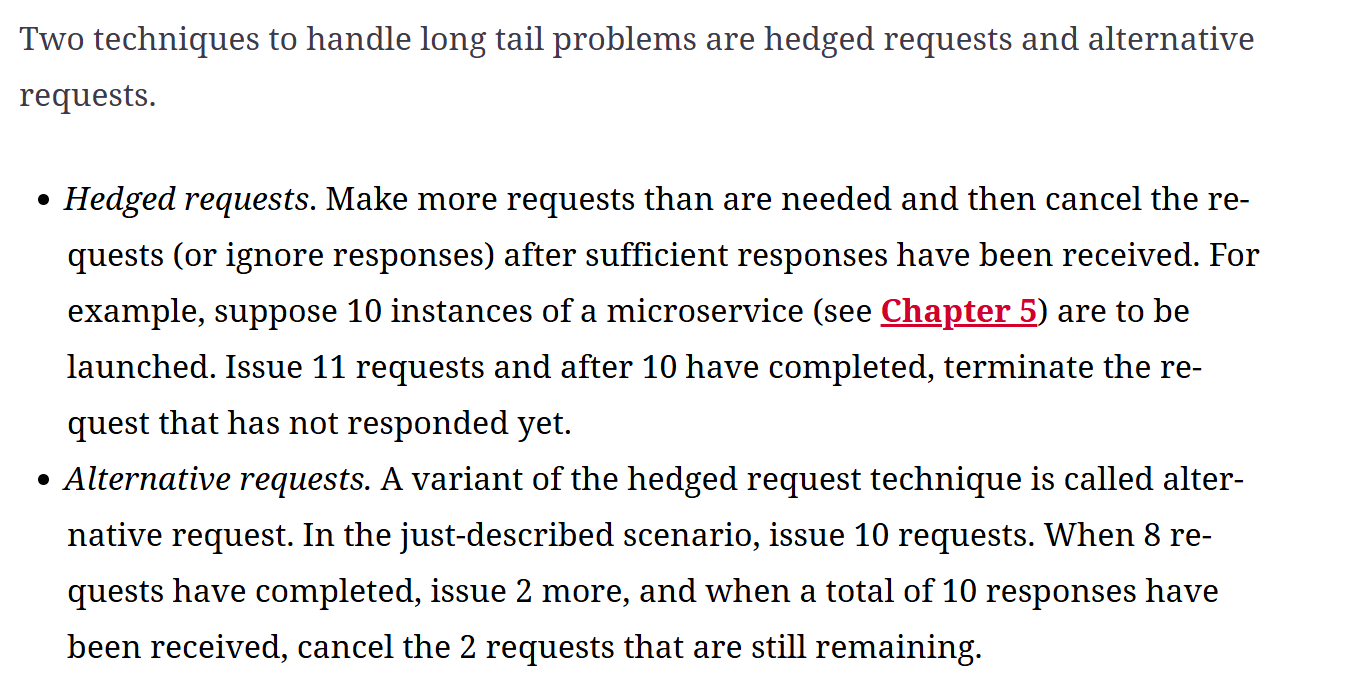


My question: What is more important among those quality attributes and how these styles influence on that?

# Software Architecture in Practice

## Chapter 17

Inspiring



There is a metric on cloud computing resources that how fast could 1000 instances be created once. This is my first time hearing the Hedged requests. It reminds me of I often neglect the hardware issues as a software engineer.