

분산 컴퓨팅 환경을 위한 WAS 프레임워크 설계

WAS Framework Designed for Distributed Computing Environments

최웅엽*, 권은지*, 이정섭, 백서현
한양대학교*, 경희대학교, 성균관대학교

Choi ung-ueop*, Kwon eun-ji*, Lee jeong-sub,
Baek seo-hyun

Hanyang Univ.*, Kyunghee Univ.,
Sungkyunkwan Univ.

요약

최근 MVC 아키텍처를 따르는 많은 웹 서비스 프레임워크들이 제공되고 있다. 그러나 기존의 상용 프레임워크들의 대부분이 분산 컴퓨팅 환경(Distributed Computing Environment)을 지원하지 않거나 지원한다고 하더라도 기존 MVC 아키텍처보다 복잡한 구조를 가지고 있다. 따라서 우리는 기존 MVC모델을 한번 더 추상화 하는 방법인 VCC를 정의하였고, 분산 컴퓨팅 환경을 MVC 모델에 맞게 재설계 하였다. View - Control - Client(VCC)란 MVC모델에서 클라이언트가 Control의 역할까지 담당하는 구조를 말한다. 본 논문에서 제안하는 분산 환경 VCC 아키텍처를 사용하면 좀 더 효율적으로 다수의 자원을 관리할 수 있을 뿐 만 아니라 고효율의 웹 어플리케이션 서비스 프레임워크를 제작할 수 있다.

I. 서론

최근 대다수의 웹 서비스 프레임워크들은 Model-View-Controller (이하 MVC)의 디자인 패턴을 지향하고 있다 [1]. 최근 발전하여 AJAX(Asynchronous JavaScript And XML) 기반의 Rich Internet Application (이하 RIA)가 생겨났고 RIA 기술은 일반적인 웹 서비스에서는 볼 수 없었던 기능들을 만들어 웹서비스는 웹 어플리케이션으로 발전하였다[4]. 점차 웹 어플리케이션들의 기능이 많아져 분산 컴퓨팅 환경(Distributed Computing Environment)이 중요하게 되었다. 하지만 웹 서비스 프레임워크에서는 RIA 기술과 분산 컴퓨팅 환경을 고려하지 않는 경우가 많거나, 지원하더라도 복잡한 MVC 패턴 구조를 가지는 경우가 많다.

본 논문에서는 기존의 MVC 패턴의 구조를 바탕으로 분산컴퓨팅 환경을 지원하는 새로운 웹 어플리케이션 서버 프레임워크 구조인 View-Control-Client(이하 VCC)를 제안한다. 제안하는 구조는 기존의 MVC 패턴을 적용시킨 웹 서비스 프레임워크를 활용하고 이를 한 번 더 추상화 하여 분산 컴퓨팅 환경을 지원하는 웹 어플리케이션 서버 프레임워크의 기능을 가능하게 할 수 있다.

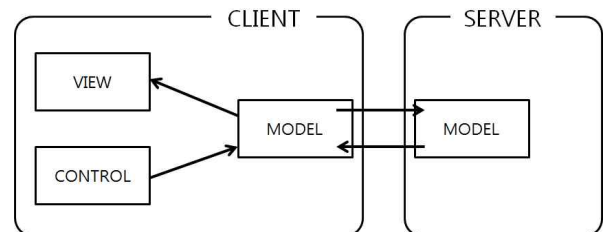
II. 기존 아키텍처 및 개선 프레임워크

1. MVC

MVC 패턴은 Model, View, Controller으로 나누어 이루어진 구조로 Model은 데이터의 입출력, View는 사용자에게 보이는 화면, Controller은 사용자가 데이터를 요

청하고 받을 수 있는 채널을 구조적으로 나눈 것이다. 이 패턴을 잘 활용하면 사용자 인터페이스와 로직을 분리하여 서로의 결합력이 낮아져 각 부분을 좀 더 손쉽게 수정 할 수 있는 장점이 있다. 이 장점을 얻기 위해서 많은 프레임워크들이 활용하고 있는 패턴이다.

[그림 1]은 기본적인 MVC 패턴에서 발전하여 서버-클라이언트에 구조에서 활용성이 높은 RIA MVC 구조를 나타낸 것이다[5]. 이는 클라이언트에 View, Control, Model을 모두 두어 서버에서의 데이터를 받아오는 작업을 최소화 시킨 것이다. 서버의 모델에서 데이터를 받아올 때에는 AJAX기반을 활용하여 비동기적으로 서버 측의 데이터를 클라이언트 측으로 옮겨 저장을 시킨다. 이 RIA MVC의 또 다른 장점으로는 클라이언트 측의 자원을 활용해서 View를 생성하고 보여줌으로써 서버의 작업량을 낮출 수 있는 것이다.



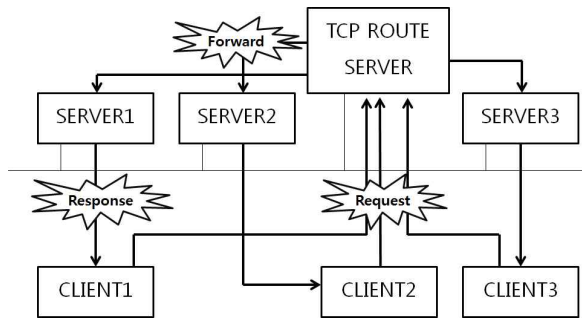
▶▶ 그림 1. RIA MVC

2. 분산 컴퓨팅 환경

분산 컴퓨팅 환경은 일반적으로 복수의 컴퓨터를 활용한 것으로 여러 대의 컴퓨터에 정보를 두고 한 대의 컴퓨터처럼 작동하는 것을 말한다.

분산 컴퓨팅 환경의 장점으로서는 작업을 여러 대의 컴퓨터로 나누어 처리를 하기 때문에 유휴 프로세싱 자원을 사용할 수 있다는 것이고, 또한 한곳에 집중된 리소스 소모를 여러 곳으로 분산하여 빠른 처리가 가능하다는 것이다. 이런 이점을 활용하면 다음 작업에 대한 준비 또한 빠르게 할 수 있다.

[그림 2]는 여러 대의 서버가 분산 컴퓨팅 환경으로 구축 되어 있으며 클라이언트가 요청하는 작업에 대한 흐름도를 나타낸 것이다. 클라이언트들이 작업 요청을 하였을 때 TCP Route Server는 요청들을 각 서버의 상태를 바탕으로 각각의 다른 서버로 나눈다. 해당하는 서버는 클라이언트의 요청을 받아서 처리를 하고, 클라이언트에게 직접 결과를 전송하는 것을 보여준다. [3]의 연구에서는 다양한 분산 컴퓨팅 환경 서버들의 구조가 제안되어 있다.



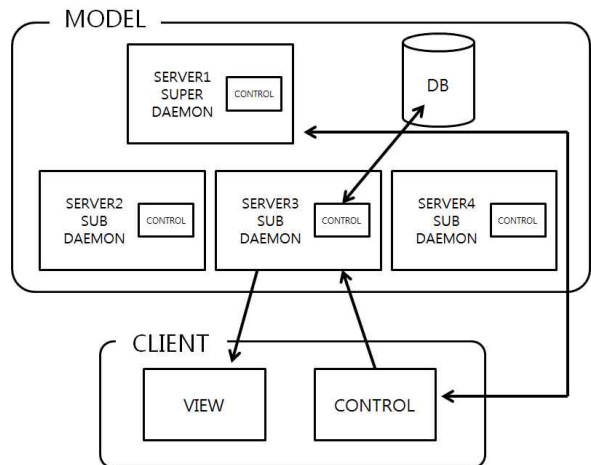
▶▶ 그림 2. 분산 컴퓨팅 환경

3. View-Control-Client(VCC)

본 논문에서 제안하는 VCC 아키텍처는 크게 기존의 웹 서비스 프레임워크를 활용하는 단계와 RIA 기술을 활용하여 분산 컴퓨팅 환경을 지원하는 과정으로 구성된다.

[그림 3]은 여러대의 서버와 1대의 관리 서버가 존재한다. 1대의 관리 서버는 Super Daemon으로 하위 여러대의 서버인 Sub Daemon을 관리하고 있으며 관리 서버를 포함한 모든 서버는 웹 서비스 프레임워크를 설치한다. 해당하는 프레임워크의 Controller를 활용하여 Database와 통신을 하고 정보들을 가져오는 방식을 선택하였다. 위와 같이 동작하는 방식의 서버 여러 대를 묶어 한 번 더 추상화 시켜 Model로 정의하였다.

그리고 Client는 View와 Control을 가진다. Control은 2가지 기능을 하게 된다. AJAX 기술을 구현한 것과 Model로부터의 데이터를 가져오는 기능을 한다. 클라이언트가 요청을 하게 되면 분산되어 있는 서버 중 먼저 Super Daemon이 존재하는 Server1로 연결되어 해당하는 Web Application이 어느 서버에서 작동이 되고 있는지를 받아 온다. 이후 요청은 Super Daemon이 알려준 서버인 Sub Daemon으로만 요청을 하게 되는 구조이다. 클라이언트의 Control은 일반적인 역할을 하면서 분산 컴퓨팅 환경이 구성된다. View는 Client에 존재하여 사용자 화면을 구성하는 기능하기 때문에 Server의 작업량을 낮춰준다.



▶▶ 그림 3. View-Control-Client

III. 결론 및 향후 계획

기존의 웹 서비스 프레임워크들을 활용한 여러 대의 서버를 묶어 Model로 정의하고 View와 Control을 Client에 두었다. 본 연구에서는 분산 컴퓨팅 환경에서의 웹 어플리케이션 서버 프레임워크에 적용할 수 있는 View-Control-Client를 제안하였고 서버측의 작업량을 줄이며 분산 컴퓨팅 환경을 활용 가능하게 하는 방안을 얻었다.

■ 참고 문헌 ■

- [1] E. Gamma, R. Helm, R. Johnson, and J. M. Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley, 1994.
- [2] Valeria Cardellini, Emiliano Casalicchio, Michele Colajanni and Philip S. Yu, "The State of the Art in Locally Distributed Web-server System," ACM Computing Surveys (CSUR), Vol. 34, pp. 263-311, 2002.
- [3] Azer Bestavros, Mark Crovella, Jun Liu and David Martin, "Distributed Packet Rewriting and its Application to Scalable Server Architectures," Proceedings of the 6th International Conference on Network Protocols, Austin Texas, pp. 290-297, 1998
- [4] Morales-Chaparro, R. Linaje, M. Preciado, J. C. Sanchez-Figueroa F, "MVC Web design patterns and Rich Internet Applications," Actas del IV Taller en Sistemas Hipermedia Colaborativos y Adaptativos, asociado a JISBD'2007. Zaragoza, 2007.
- [5] 손동혁, 정강수, 박석, "MVC 웹 프레임워크를 기반으로 하는 RIA를 위한 플러그인 프레임워크 설계 및 구현", 한국정보과학회 논문집 제37권 제2호(C), pp. 72-77, 2010.