

모바일 스튜디오: 클라우드로의 효율적인 오프로딩 기법 연구

김유식*, 박용성**, 윤찬현*
 *한국과학기술원 전기및전자공학과
 **한국과학기술원 그리드미들웨어연구센터
 e-mail : {yusiky326, miracle0318, chyoun}@kaist.ac.kr

A Study on Efficient Methods of Offloading to Cloud in Mobile Studio

Yusik Kim*, Yong-Sung Park**, Chan-Hyun Youn*
 *Dept. of Electrical Engineering, KAIST
 **GRID Middleware Research Center, KAIST

요 약

모바일 클라우드 컴퓨팅 기술이 대두 됨에 따라, 모바일 단말기의 이동성과 클라우드의 컴퓨팅 파워 및 스토리지 자원의 장점을 모두 취합하는 어플리케이션 개발이 가능해졌다. 모바일 어플리케이션 사용자에게 클라우드의 존재가 마치 보이지 않는 것처럼 끊임 없이 작은 오버헤드로 전송을 하는 아키텍처를 제안하고자 한다.

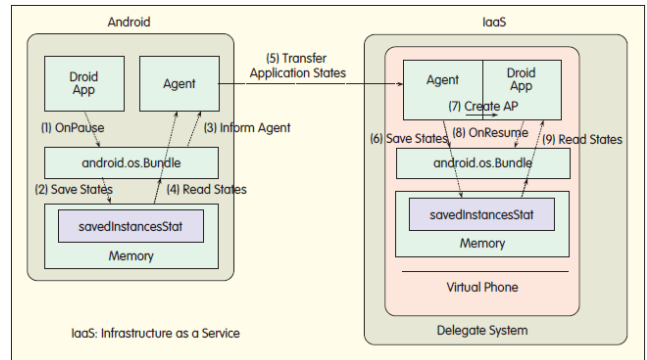
1. 서론

애플의 아이폰 보급을 기점으로 스마트폰의 보급이 시작된 후, 모바일 기기의 진화가 급속도로 이루어졌다. 다양한 모바일 단말의 등장 이 이루어지면서, 단말의 상시적인 온라인 접속이 일반화되면서, 자연스럽게 온라인 네트워크 접속 상태가 전제되어야 하는 클라우드 컴퓨팅이 업계의 화두로 부상하게 됐다. 모바일 기기와 클라우드 컴퓨팅 기술이 만나 모바일 클라우드 컴퓨팅 기술이 자연스럽게 등장하였다.

모바일 클라우드 컴퓨팅이란, 데이터 저장과 데이터 처리가 모바일 장치 밖에서 이뤄지는 인프라스트럭처를 이야기 한다. 모바일 클라우드 응용 소프트웨어들은 컴퓨팅 파워와 데이터 저장을 모바일 기기에서 벗어나 클라우드에서 이뤄지게 하며, 스마트폰 유저뿐만 아니라, 모든 모바일 가입자들에게 적용된다.[1]

모바일 클라우드 컴퓨팅을 활용하면, 모바일 기기의 최대 장점인 이동성을 그대로 유지하면서, 하드웨어적인 한계로 하지 못했던 힘든 작업을 클라우드로 오프로딩(offloading) 함으로써 가능하게 할 수 있다. 하지만, 오프로딩을 하는데 있어서, 고려해야 될 부분이 많다. 기존에 있는 어플리케이션들은 오프로딩을 고려하여 다시 개발되어야 하며, 네트워크 연결이 불안정할 때도 동작이 가능할 수 있어야 한다. 또한, 개인적인 정보의 유출 문제 등이 오프로딩 과정 중에 생길 수 있다. 이에 본 논문에서는 기존에 나와 있는 오프로딩 기법 사례 연구를 소개하고, 이들의 문제점을 극복한 새로운 프레임워크를 제안하고자 한다.

2. 모바일 어플리케이션의 특성을 살린 오프로딩



(그림 1) Shih-Hao Hung 등이 제안한 프레임워크[2]

Shih-Hao Hung 등이 제안한 클라우드 기반 가상실행환경[1] 은 안드로이드의 onPause()메소드의 활용을 극대화하는 방법을 채택한 프레임워크다. 메모리 등 리소스가 부족한 모바일 기기에서 동작하는 안드로이드 OS 는 메모리가 부족한 경우에 백그라운드 프로세스를 종료시키기도 한다. 따라서, 안드로이드 OS 에서 동작하는 모든 프로세스는 백그라운드 프로세스로 전환될 경우, 어플리케이션 상태 데이터(application state data)를 저장하게 된다. Suspend 를 요청받은 프로세스는 onPause() 메소드를 실행하게 되고, 아직 저장되지 않은 주요 정보들을 영구적으로 저장하게 된다. 이후, Suspend 된 프로세스는 OS 에 의해 종료가 될 수도 있지만, 사용자에게 의해 onResume()메소드를 통해

재실행될 수도 있다. OS 에 종료가 되어도, 영구적으로 저장된 어플리케이션 상태 데이터를 통해 상태를 복구할 수 있다.

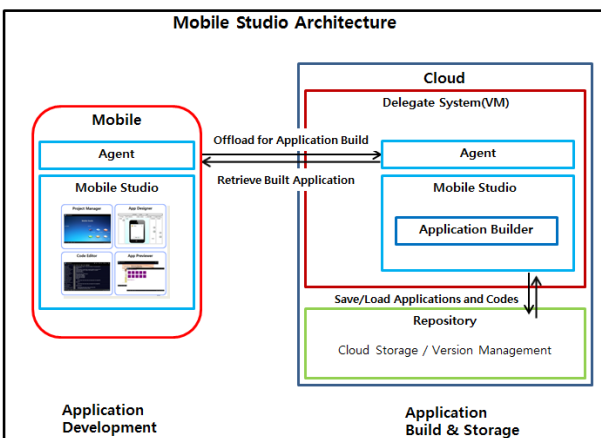
Suspend 될 때 저장되는 어플리케이션 상태 데이터를 클라우드에 보냄으로써, 클라우드 상에 구동중인 가상실행환경에서 이어서 실행하는 것이 이 프레임워크의 핵심이다.

이러한 오프로딩의 장점은 첫째, 안드로이드 어플리케이션의 경우 자연스럽게 포함된 onPause()메커니즘을 활용함으로써, 기존의 어플리케이션들을 처음부터 다시 개발할 필요가 없는 편의성이다. 둘째, 저장되는 어플리케이션 상태 데이터의 경우, 개발자가 직접 그 양을 최소화 할 수도 있으며, 암호화를 통해 개인적인 정보 유출을 막는 것도 가능하다. 셋째, 모바일 어플리케이션 사용자들은, 클라우드가 일을 처리해주는 동안 모바일 단말기로 다른 일을 볼 수 있어서 마치 모바일 단말기가 클라우드에서 제공하는 컴퓨팅 파워 및 스토리지 기능을 갖고 있는 듯한 투명성을 제공할 수 있다.

하지만, 어려움들도 있다. 어플리케이션 개발은 편해졌지만, 반대로 클라우드 서비스를 제공하는데 있어서는, 다양한 단말기와 여러 OS 들을 구동하는 가상실행환경을 개발이 선행되어야 한다. 이것을 가능하게 하는 것은 OS 와 어플리케이션들의 표준화 작업이 필요한 이에 맞춰 개발 및 동작 되어야 한다. 가상실행환경의 단말기와 실제 단말기는 미러링을 통하여 동기화가 이루어져야 하며, 지속적인 미러링은 많은 트래픽을 유발하므로, 이를 고려한 동기화 정책과 프로토콜이 필요하다. 또한, 모바일 기기에서 간단하게 처리할 수 있는 일들도 일일이 클라우드로 전송해서 처리함으로써, 불필요한 오버헤드가 생길 수도 있다.

3. Suspend/Resume 을 적용한 모바일 스튜디오

모바일 스튜디오란, 언제 어디서든 모바일 단말기만 있으면, 어플리케이션 개발이 가능한 개발 환경이다. 하이브리드 앱 기술을 이용하여 단일 소스로 이기종 OS 를 지원하는 것을 지향한다.



(그림 2) 모바일 스튜디오의 구조: Mobile Phone 에서 개발된 어플리케이션 코드가, 클라우드로 끊임 없이 오프로딩되어 빌드 및 스토리지 된다.

모바일 단말기는 일반적으로 컴퓨팅 파워나 배터리 부족 등이 단점으로 꼽힌다. 따라서, 어플리케이션의 빌드와 코드 스토리지, 버전 관리 및 배포 등은 클라우드에서 실행되는 구조를 제안한다. (그림 1)에서 제안한 프레임워크는, 가상실행환경을 통해, 복제된 환경 속에서 동일한 프로그램이 두 군데에서 실행되는 것이지만, 모바일 스튜디오의 경우 모바일 단말기와 클라우드간의 역할이 명확히 나뉘는 경우에는 적합하지 않다.

모바일 기기에서 동작하는 모바일 스튜디오는 개발자가 코드를 편집 후 저장을 한 코드가 있는 상태에서 suspend 하게 된다면, Agent 프로세스가 이를 감지하고 클라우드로 전송하게 된다. 이 때, 전송 오버헤드와 현재 단말기의 상태를 보고, 모바일 단말기 내부에서 빌드가 가능하다면, 모바일 단말기 역시 하나의 리소스로서 일을 일부 또는 전체를 분담할 수 있도록 한다.

클라우드는 스토리지 서비스를 통해 버전 관리를 제공해주어야 하므로, 모바일 기기에서 업데이트 되는 정보와 동기화가 이루어져야 한다. 모바일 기기가 suspend 상태가 된다면, Agent 간의 통신을 통해 동기화가 시작된다. 완성된 상태의 코드라면 컴파일 후 모바일 단말기로 전송이 가능하도록 준비해 놓는다.

4. 결론 및 제언

본 논문에서는 모바일 클라우드 컴퓨팅에 핵심적인 기술인 오프로딩을 모바일 어플리케이션 사용자뿐만 아니라, 개발자 입장에서 효율적인 클라우드 아키텍처를 설계하였다. 모바일 단말기가 가지고 있는 컴퓨팅 파워의 한계를 극복하려면, 클라우드에서 제공하는 인프라스트럭처를 이용해야 한다. 하지만, 이는 어플리케이션을 개발하는 입장에서는, 기존에 있는 어플리케이션조차 새로 다시 만들어야 하는 번거로움이 생긴다. 이에 suspend 와 resume 을 자연스럽게 실행하는 모바일 어플리케이션의 특징을 살려, 많은 컴퓨팅 자원이 요구되는 일들을 모바일 단말기에서 suspend 되는 동안 클라우드로 이전하고, suspend 시 자연스럽게 저장하는 어플리케이션 상태 데이터만을 전송하여 오버헤드를 줄인다면, 개발자에게도 편의성을 제공하며 빠르고 끊임 없이 동작하는 모바일 어플리케이션이 가능하다.

앞으로, 전송 오버헤드를 줄이는 방법과 동기화에 필요한 프로토콜 등의 연구가 더 필요하다.

참고문헌

- [1] <http://www.mobilecloudcomputingforum.com>
- [2] Shih-Hao Hung, Tei-Wei- Kuo, Chi-Seng Shih, Jeng-Peng Shieh, Chen-Pang Lee, Che-Wei Chang, and Jie-WenWei. "A Cloud-Based Virtualized Execution Environment for Mobile Applications"
- [3] Hoang T. Dinh, Chonho Lee, Dusit Niyato and Ping Wang. A Survey of Mobile Cloud Computing: Architecture, Applications, and Approaches, 2011