

## 모바일 클라우드 환경에서 브로커를 이용한 워크플로우 관리 연구

김희재, 윤찬현,  
한국과학기술원 전기및전자공학과  
e-mail : {kim881019, chyoun}@kaist.ac.kr

### A Study on Workflow Management using Broker in Mobile Cloud Environment

Heejae Kim, Chan-Hyun Youn  
Dept. of Electrical Engineering, KAIST  
e-mail : {kim881019, chyoun}@kaist.ac.kr

#### 요 약

모바일 기기에서의 워크플로우 실행에서 기존의 클라우드 컴퓨팅 환경을 사용하는 것은 컴퓨팅 자원 제한 측면에서는 유용할 수 있지만 모바일 기기의 이동성, 배터리 제한의 제약 때문에 새로운 방법이 필요하다. 따라서 본 논문에서는 모바일 클라우드 환경에서 모바일 클라우드 워크플로우 브로커(Mobile Cloud Workflow Broker)를 이용하여 모바일 기기의 이동성과 배터리 제한의 제약을 고려하며 워크플로우를 관리하는 방법에 관하여 다룬다.

#### 1. 서론

워크플로우(Workflow)는 많은 양의 데이터를 일정 순서대로 자동화 또는 협업화 시켜 최종적으로 분석된 새로운 결과를 도출하는 개념 [1]으로써 물리, 생명과학 등 여러 분야에서 다양하게 사용되고 있다. 데이터 규모가 점점 거대해지고 있는 시점에서 클라우드 컴퓨팅(Cloud Computing)은 무한한 컴퓨팅 자원을 제공하고 탄력성을 보장할 수 있다는 장점으로 이러한 데이터를 처리하는 워크플로우를 지원할 수 있는 효과적인 방법으로 떠오르고 있다 [2].

하지만 모바일(Mobile) 기기에서 워크플로우 실행을 위해 기존의 클라우드 컴퓨팅을 사용하는 것은 컴퓨팅 자원 제한 [3] 측면에서는 유효할 수 있지만 모바일 기기의 이동성, 배터리 제한 등의 제약 [3] 때문에 기존 클라우드 컴퓨팅과는 다른 접근 방식이 필요하다.

따라서 본 논문에서는 브로커(Broker)를 이용한 모바일 클라우드 환경에서의 워크플로우 관리 방법을 다룬다. 모바일 기기들과 클라우드를 중계할 수 있는 브로커의 사용은 여러 모바일 기기의 제약조건 하에서 워크플로우 실행을 원활히 할 수 있도록 한다. 본 논문에서의 워크플로우는 그 자체의 의미와 더불어 단일 머신에서 실행하는 워크플로우의 목적은 VM을 실행할 수 있는 상태로 미리 만들어 놓기 위함이다.

#### 2. 모바일 클라우드 워크플로우 브로커(Mobile Cloud Workflow Broker)

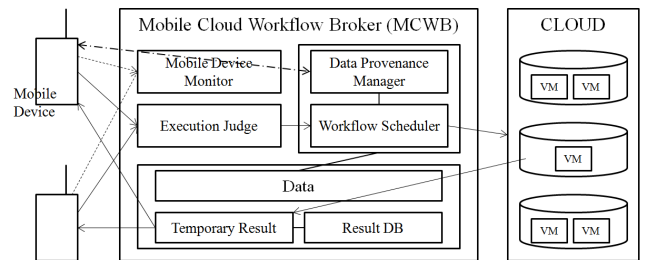


그림 1. 모바일 클라우드 워크플로우 브로커(MCWB)

모바일 클라우드 워크플로우 브로커(MCWB)는 모바일 기기들과 클라우드 사이를 중계하는 역할을 하며 그림 1과 같이 구성되어 있다.

MCWB에서 모바일 기기로부터의 워크플로우 실행 요청은 실행 판단자(Execution Judge)에게 들어오게 된다. 여기서 실행 판단자는 결과를 보내는 시점을 결정하는 정책인 결과 전송 정책(Result Transfer Policy)를 만들어 워크플로우 스케줄러(Workflow Scheduler)로 보낸다. 워크플로우 스케줄러는 결과 전송 정책을 이용하여 데이터 기원 매니저(Data Provenance Manager)와 함께 클라우드 내에서의 워크플로우 실행을 관리한다.

MCWB에서는 모바일 기기의 이동성을 지원하기 위해서 모바일 기기 모니터(Mobile Device Monitor)를 이용하여 네트워크 연결 상태를 체크하게 되며 이

것은 결과 전송 정책과 더불어 모바일 기기가 안전하게 결과를 받을 수 있도록 한다.

### 3. 모바일 클라우드에서의 워크플로우 스케줄링

#### MCWB Workflow Scheduling Algorithm

##### Execution Judge

- 1: Estimate the execution time,  $t_e$
- 2: Estimate the lifetime of mobile device,  $t_m$
- 3: if  $t_e < t_m$ ,
- 4: Immediate Result Transfer Policy is adopted
- 5: else
- 6: Waiting Result Transfer Policy is adopted

##### Workflow Scheduler

- 7: Set Battery Performance Curve, BPC
- 8: Combine  $f_b$  and selected RTP
- 9: Set Deadline Interval ( $t_d$ )
- 10: MCWB VM Allocation Algorithm
- 11: Workflow Scheduling

MCWB 에서의 워크플로우 스케줄링은 MCWB 워크플로우 스케줄링 알고리즘을 이용하여 처리된다. 워크플로우 스케줄링은 실행 판단자, 워크플로우 스케줄러에서 진행되게 된다.

실행 판단자에서는 우선 워크플로우의 총 실행 시간( $t_e$ )과 배터리 상태를 고려한 해당 모바일 기기의 남은 수명( $t_m$ )을 예측하게 된다. 만약  $t_e$  가  $t_m$  보다 작으면 워크플로우 실행 결과를 임시 결과 저장소(Temporary Result Storage)에 넣지 않고 바로 모바일 기기으로 전송하는 정책인 즉각적인 결과 전송 정책(Immediate Result Transfer Policy, IRTP)이 적용되고  $t_e$  가  $t_m$  보다 크면 워크플로우 실행 결과를 임시 결과 저장소에 넣어서 모바일 기기 모니터에서 수집된 모바일 기기의 상태를 확인한 후 모바일 기기으로 결과를 전송하는 정책인 대기 결과 전송 정책(Waiting Result Transfer Policy, WRTP)가 적용된다. 워크플로우 스케줄러는 결과 전송 정책에 따라 실제로 워크플로우 스케줄링을 진행한다. 우선 워크플로우 스케줄러는 해당 모바일 기기의 배터리 성능 커브(Battery Performance Curve, BPC)을 정한다. 또한 현재와 데드라인 사이의 간격을 데드라인 간격( $t_d$ )로 두어 그 시간까지 결과가 전송되도록 한다. 만약 WRTP 의 경우 데드라인은 해당 모바일 기기가 다시 접속하는 시간을 뜻한다. 여기서  $(f_b + RTP)/t_d$  의 값이 높을수록 클라우드에서 자원 용량이 큰 가상 머신(Virtual Machine)이 필요하다.

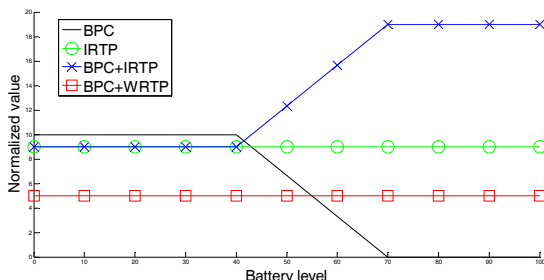


그림 2. 배터리 성능 커브와 결과 전송 정책

### 4. MCWB 가상 머신 할당 알고리즘 (MCWB VM Allocation Algorithm)

모바일 클라우드에서의 워크플로우 관리를 위한 가상 머신 할당 알고리즘은 다음과 같다.

#### MCWB VM Allocation Algorithm

- 1: Estimate each required capacity of VM,  $d_i$
- 2: Make VM Sort Plain
- 3: Sort
- 4: for all  $i$
- 5: if  $i^{\text{th}}$  job does not finish until deadline
- 6: Re-Sort based on  $t_d$
- 7: end for
- 8: VM Allocation based on the result of sort

MCWB 가상 머신 할당 알고리즘에서,  $d_i$  는 모바일 기기  $i$  의 VM 요구량으로  $d_i = \min \max(\text{VM volume of workflow execution in the certain moment})$  로 정의된다. VM Sort Plain 은 그림 3 과 같고 오른쪽 위를 향할 수록 용량이 큰 가상 머신 할당의 할당이 필요하다는 것을 의미한다. VM 소트 평면(Sort Plain)을 바탕으로 모든 사용자 요청들은 가상 머신의 필요 용량 순으로 정렬된다. 하지만 여기서 데드라인까지 처리가 되지 못하는 사용자(모바일 기기) 요청이 있을 때는 해당 요청을 재정렬하여 해당 요청을 데드라인에 끝낼 수 있도록 한다. 최종적으로 정렬된 결과는 브로커가 담당하는 가상 머신에 적용되어 각 모바일 기기들에 대하여 가상 머신 할당을 하게 된다.

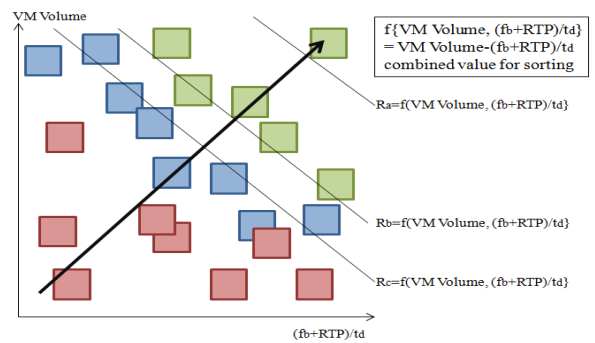


그림 3. VM 소트 평면(Sort Plain)

### 5. 데이터 기원(Data Provenance)을 이용한 워크플로우 관리

데이터 기원은 워크플로우에서 각 태스크(Task) 및 파라메타(Parameter)와 결과 데이터 간의 연관관계를 나타내는 것 [4] 으로 워크플로우에서 각 태스크(Task)의 결과들의 의존성(Dependency)을 표현한다 [5]. 본 논문에서는 그림 1 의 데이터 기원 매니저(Data Provenance Manager)가 데이터 기원을 관리한다. 워크플로우 스케줄링에서 WRTP 가 적용되고 기기가 종료되었을 때 워크플로우 실행은 요청 데이터 오류 및 부재, 중간 태스크의 부정확성 등으로 더 이상 진행되지 못하는 경우가 생기게 된다. 이 경우, 데이터 기원 매니저는 해당 모바일 기기가 다시 네트워크 접속이 된 후 워크플로우 실행에 관한 새로운

정보를 얻고 워크플로우 스케줄러와 함께 워크플로우의 실행을 계속 진행하게 된다.

## 6. 결론

본 논문에서는 워크플로우를 모바일 클라우드 환경에서 실행하기 위해 클라이언트와 서버 두 관점에서 MCWB 를 이용하여 워크플로우 실행을 원활하게 할 수 있는 방법에 관하여 다루었다. 이 방법의 사용으로 기존 클라우드에서 모바일 기기를 이용한 워크플로우 실행의 걸림돌이었던 모바일 기기의 이동성 및 배터리 제한의 제약조건을 극복할 수 있었다. 또한 앞으로 본 연구와 관련하여 작업 오프로딩(Offloading)에 대한 연구가 더 이루어져야 할 것이다.

## Acknowledgement

본 논문은 2012 년도 지식경제부의 재원으로 개인 및 기업 맞춤형 서비스를 위한 개방형 모바일 클라우드용 통합개발환경 및 이기종 단말-서버 간 협업 기술개발[2012-10039260]과 BK21 사업의 지원을 받아 수행된 연구임

## 참고문헌

- [1] J. Yu, R. Buyya, “ A Taxonomy of Scientific Workflow Systems for Grid Computing” , SIGMOD Record, vol. 34, no. 3, p. 45, 2005
- [2] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Paterson, A. Rabkin, I. Stoica, M. Zaharia, “ Above the Clouds: A Berkeley View of Cloud Computing” , Feb, 2009.
- [3] M. Satyanarayanan, “ Fundamental challenges in mobile computing” , in Proceedings of the 5<sup>th</sup> annual ACM symposium on Principles of distributed computing, pp. 1-7, May. 1996.
- [4] D. Koop, E. Santos, B. Bauer, “ Bridging Workflow and Data Provenance using Strong Links” , in Proceedings of the 22<sup>nd</sup> international conference on Scientific and statistical database management, pp. 397-415, 2010
- [5] D. Yuan, Y. Yang, X. Liu, J. Chen, “ A Cost-Effective Strategy for Intermediate Data Storage in Scientific Cloud Workflow Systems” , 2010 IEEE International Symposium on Parallel & Distributed Processing, pp. 1-12, April, 2010