

# CSB를 통한 PaaS 구조 설계를 위한 어플리케이션 체계 연구

지태현\*

\*아이엔소프트 UBICOM 연구소 소장

e-mail:chidae2000@in-soft.co.kr

## A Study on Application management structure for PaaS through Cloud Service Brokerage

Tae-Hyun Chi\*

\*UBICOM Rearch Lab of IN-Soft Corp.

### 요 약

클라우드에 대한 범용적인 관리를 위해서 사용되는 CSB(Cloud Service Brokerage) 기술을 기반으로 실제 사용자들이 이용하는 서비스를 제공하는 어플리케이션을 제공하기 위한 모델로 PaaS(Platform as a Service)이 도입이 되어야 한다. 이러한 구조 제공을 위해서는 CSB 서비스의 특성과 Application 구성 요소에 대한 관리 체계에 대한 접근을 통해서 다양한 클라우드에서 호환성을 보장할 수 있어야 한다. 본 논문에서는 기존에 활용되는 도구들의 제약 사항과 함께 클라우드 기반 마켓플레이스를 통해 다양한 IT 서비스를 생산 및 유통할 수 있는 구조에 대해서 고찰하고자 한다.

### 1. 서론

클라우드 서비스의 활용이 보편화되고 있으며, 이러한 보편화가 IaaS(Infrastructure as a Service)의 인프라 활용 측면에서 PaaS(Platform as a Service)<sup>[1]</sup>의 체계로 발전이 이루어지고 있다. 클라우드 기술이 활성화 되면서 도입기에서부터 제기되었던 S/W 호환성(Interoperability)과 이동성(Mobility)에 대한 이슈가 부각이 되고 있으며, 이러한 부분을 해결함으로써 클라우드를 통한 서비스의 보편화와 대중화를 달성할 수 있다.

### 2. 클라우드 기반 서비스 제공 요구사항

클라우드를 기반으로 서비스를 제공하기 위해서는 클라우드 자원에 대한 동적인 자원 관리 구조를 기반으로 서비스 운영에 필요한 제반 S/W에 대한 배치 및 제어가 이루어져야 한다. 특히 이러한 구조를 다양한 사업자의 클라우드에서 제공하기 위해서는 CSB(Cloud Service Brokerage)<sup>[2]</sup> 구조를 통해서 인프라에 대한 제어가 가능하다.

이러한 다양한 사업자의 클라우드에서 서비스를 제공하기 위한 필요성은 IT 서비스의 글로벌화 및 Latency 기반의 서비스들의 등장에 기반하고 있다. 예를 들면 자율주행 자동차의 경우, 0.1초의 차이로 인해서 사고가 날수 있기 때문에 이러한 서비스를 제공하기 위해서는 근거리 서비스 제공이 필수적이며 이러한 서비스를 제공하는 클라우드 인프라를 활용할 수 밖에 없다.

클라우드 서비스의 장점은 클라우드 가상화 기술에서 제공

하는 이미지(또는 템플릿)을 기반으로 하는 재사용성에 있었다. 하지만 이렇게 다양한 클라우드를 활용해야 하는 입장에서 단일 사업자가 제공하는 이미지 기반의 재사용성은 그 한계가 명확하다.

최근에 각광받고 있는 유사한 기술로 컨테이너(Container) 기반 기술이 이러한 부분을 대체할 수 있을 것으로 보고 있으나 기반 기술이 어플리케이션 이미지 관리 체계를 통해서 이루어지기 때문에 호환성 문제는 해결할 수 있지만 그 근본적인 접근 방법으로는 한계가 있다.

클라우드에서의 서비스는 기본적인 가상화 자원에 대한 제어어를 통해서 서비스가 필요로 하는 운영 환경을 구성하고 이를 용량에 따라서 확장할 수 있는 IaC(Infrastructure as Code)<sup>[3]</sup> 기반의 관리 체계와 함께 동적인 소프트웨어 라이프 사이클 지원 체계가 필요하다.

본 논문에서는 인프라 관리를 위한 CSB 기술을 활용해서 PaaS 서비스에 필요한 인프라를 제공하고 이러한 기술 기반으로 PaaS 서비스에 필요한 어플리케이션 관리 체계를 구성 방안을 제안하고자 한다.

### 3. CSB(Cloud Brokerage Service) 제공 기능

CSB 기술은 이질적인 클라우드 서비스를 중계자를 통해서 중계함으로써 사용자 측면에서 이질적인 클라우드를 하나의 클라우드처럼 이용하는 것을 가능하게 하는 서비스이다. 이러한 서비스를 통해서 앞서 2절에서 이야기하는 클라우드 기반 서비스 요구 사항에 대한 인프라 요구 사항을 충족시킬 수 있다.

CSB 기술은 이러한 목표를 달성하기 위해서, 이질적인 클라우드의 자원 속성을 메타 정보로 속성화하고 속성화된 정보를 통해서 개별 클라우드에 대한 자원에 대한 제어 기능을 제공한다.

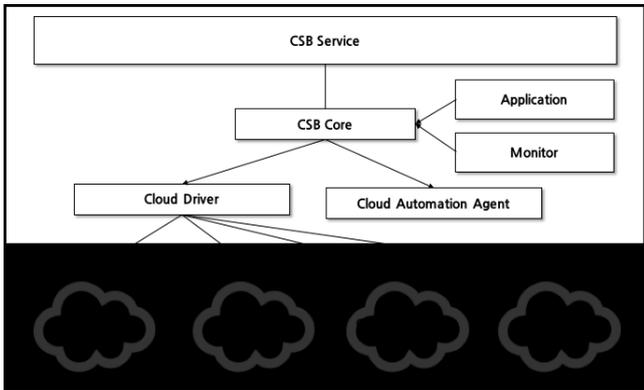


그림 1. CSB 서비스 제공 개념도

이러한 CSB의 동작 방식이 중요한 것은, CSB에서 제공하는 메타 정보 체계를 통해서 클라우드 기반 서비스가 인프라 종속성없이 동작이 가능하기 때문이다. 본 논문의 기반이 되고 있는 연구 과제에서는 이러한 부분을 해소하기 위한 논리 모델이 적용되었으며, 이러한 논리 모델을 통해서 서비스 라이프 사이클에 대한 자원 제어 기능 제공

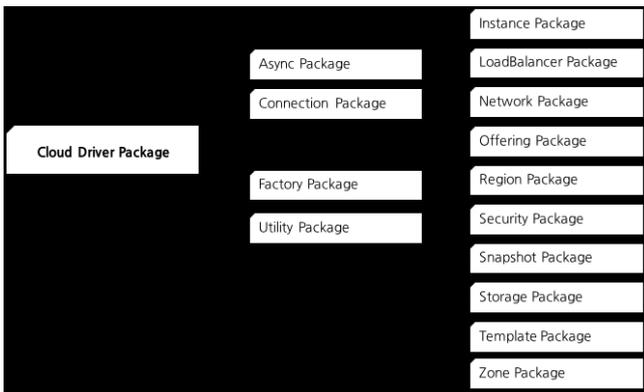


그림 2. CSB에서의 자원 정보 모델링 예시

이러한 자원 모델링 정보를 통해서 CSB 서비스를 통한 클라우드 종속성 문제를 해결할 수 있다.

#### 4. PaaS(Platform as a Service) 구성 요소

PaaS는 사용자들이 손쉽게 서비스를 개발하고 운영할 수 있는 소프트웨어 요소를 제공하는 서비스로 사용자들이 개발한 소프트웨어의 배포뿐만 아니라 활용적인 측면의 지원 기능이 제공된다.

이러한 PaaS 구성 요소로 본 논문에서는 3가지 필수 구성 요소를 제시한다.

- 자체적으로 설명 가능한 어플리케이션
- 어플리케이션간의 연계 구성을 위한 워크플로우
- 클라우드 자원과 연계된 서비스 카탈로그

이러한 필수 구성 모델을 통해서 클라우드 기반의 서비스 제공에 필요한 PaaS 서비스를 제공할 수 있다. 각각의 구성 요소에 대한 정의는 다음과 같다.

##### 4.1. 자체적으로 설명 가능한 어플리케이션

어플리케이션은 이미 Container와 같은 기술을 통해서 이미지화하여 배포되는 기술이 널리 확산되고 있다. 하지만 이러한 기술의 단점은 실제 어플리케이션이 동작되고 제공하는 기능에 대한 파악을 위해서는 전문적인 학습이 필요하다. 단적으로 Container에 대한 대표적인 오픈 소스 툴인 Docker Container<sup>[4]</sup>의 경우에도 어플리케이션에 대한 바이너리를 압축해서 배포하는 체계는 잘 제공하고 있지만, 실제 이러한 어플리케이션을 활용하기 위해서는 DevOps 체계가 수립된 고도로 전문화된 인력이 필요하기 때문에 현실적으로 보편적으로 활용되기 어렵다.

이러한 부분을 해결하기 위해서는 다음의 항목들에 대한 추가적인 관리 체계의 정의가 필수적이다.

##### 1. 어플리케이션의 동작 및 제어를 위한 관리 체계

이는 DevOps의 Operations에 해당하는 기능으로 어플리케이션의 동작 방식에 대한 정의 및 관리가 됨으로써 개발자 중심의 어플리케이션 운영 환경 제공.

##### 2. 어플리케이션이 제공하는 기능에 대한 표출을 위한 메니페스트(Manifest) 기능

어플리케이션의 업그레이드에 따라서 어플리케이션이 제공하는 기능을 명세화한 메니페스트 기능을 제공함으로써 기능에 대한 명시성과 이용 환경에 따른 지원 명세를 통해 개방성과 호환성 지원

##### 3. 버전 관리

어플리케이션의 기능 변경에 따른 버전 관리를 위한 바이너리 및 메니페스트 파일에 대한 형상 관리를 통해 하위 버전에 대한 호환성 및 최신 버전에 대한 기능 제공

##### 4. 의존성 관리

어플리케이션의 구성을 위한 의존성을 관리함으로써 어플리케이션 배포 및 운영에 필요한 환경을 효율적으로 관리

##### 4.2. 어플리케이션간의 연계 구성을 위한 워크플로우

어플리케이션은 자체적으로도 실행이 가능하지만, 어플리케이션간의 연계(Association)를 통해서 보다 다양한 기능을 제공할 수 있다. 대표적으로 하둡(Hadoop) 에코의 경우에는 다양한 오픈 소스 솔루션을 통해서 분산 파일 기능 및 분산 처리 기능을 제공하고 있다.

하둡 에코와 같이 어플리케이션의 다양성을 확장하기 위해서는 어플리케이션간의 연계를 통한 활용 방법에 대한 “정

의 기능”이 필수적이며, 이러한 기능을 통해서 사용자는 어플리케이션간의 명확한 연계와 어플리케이션의 배치를 통해 클라우드 환경에서 S/W 기반의 생태계의 구성이 가능하다. 이러한 생태계의 구성을 위해서는 이러한 “정의 기능”을 보다 효과적으로 활용할 수 있도록 다음과 같은 기능이 필요하다.

1. 어플리케이션에 대한 실행 체계에 대한 명시화를 UI/UX를 파악할 수 있도록 직관적으로 제공함으로써, 어플리케이션의 기능 변경 또는 실행에 따른 영향도를 명확하게 파악
- 2.
3. 어플리케이션 관리에 필요한 패치 및 설정 등의 중복된 프로세스 실행 및 운영 관리 기능을 공통 컴포넌트로 정의하고 이를 독립적인 어플리케이션 컴포넌트와 연계하여 관리하도록 함으로써 불필요한 어플리케이션의 중단 방지 및 관리 효율화 달성
4. 클라우드 자원과 연계된 어플리케이션 서비스 구성 기능을 통해 사용자가 활용 가능한 서비스 카탈로그 명세화하고 필요에 따라서 서비스 카탈로그를 통해 필요한 서비스를 생성하고 운영함으로써 비용 절감 효과 달성
5. 명세화된 어플리케이션 “정의 기능”에 대한 테스트 환경 제공을 통해 DevOps 환경에 맞는 개발 지원 기능 제공

#### 4.3. 클라우드 자원과 연계된 서비스 카탈로그

사용자가 이용 가능한 서비스를 생성하기 위해서는 어플리케이션과 클라우드 자원에 대한 연계 정보를 기반으로 서비스 구성이 가능한 카탈로그를 구성한다.

이러한 카탈로그는 단일 자원 모델로 구성될 수 있지만, 복수개의 자원 모델을 통해서도 구성이 될 수 있으며 이러한 구성이 실제 클라우드를 활용하는 일반적인 모델이다.

복수개의 자원 모델로 카탈로그를 구성하는 경우, 어플리케이션의 배치에 따른 서비스 운영 체계의 구축을 지원하기 위해서는 4.2에서 설명한 워크플로우의 정의가 필수적이다. 서비스 카탈로그 기반의 관리 모델에서는 자원의 변경/생성은 자원의 상태 변경과 어플리케이션의 제어가 동시에 이루어져야 하는 상황을 유발하며, 이러한 동시성을 보장하기 위해서는 자원 상태에 대한 모니터링과 어플리케이션 제어를 위한 워크플로우의 연계가 이루어져야 한다.

결론적으로 서비스 카탈로그가 가져야 하는 필수 기능은 다음과 같다.

1. 자원 모델, 어플리케이션, 워크플로우 체계를 통합하여 관리하기 위한 메타 정보 관리 체계
2. 자원의 상태 변경을 모니터링하고 상태 변경에 따른 워크플로우 실행을 위한 자원 모델 관리 기능
3. 서비스 용량에 따른 자원 모델 정의 및 자원 모델에 효과적으로 어플리케이션을 배치하기 위한 관리 체계

4. 어플리케이션이 상이한 클라우드로 이동하거나 자원 변경 및 스케일 아웃(Scale Out)을 통해 실제 클라우드 자원의 변화에 영향받지 않는 일관된(Seamless) 서비스 무중단 환경 지원

#### 5. CSB와 PaaS를 활용한 클라우드 마켓플레이스

다양한 클라우드 사업자들이 클라우드 시장의 주도권을 잡기 위해서 공격적으로 클라우드 서비스에 투자를 확대하고 있다. 국내에서도 기존 클라우드 서비스 사업자 뿐만 아니라 포탈 기업들을 중심으로 클라우드 서비스에 대한 투자가 확대되고 있다.

클라우드 서비스는 인프라 서비스에서 다양한 응용 기술을 융합하고 신 서비스를 보다 쉽게 개발하고 확산할 수 있도록 PaaS 기술에 집중하고 있다.

이러한 시장 상황과 기술 발전 방향을 보다 효과적으로 수용하기 위한 기술이 바로 클라우드간의 통합을 위한 CSB 기술과 이를 기반으로 다양한 소프트웨어 플랫폼(PaaS)을 확산할 수 있는 기반으로 역할을 할 수 있는 것이 클라우드 기반 S/W 마켓플레이스이다.

본 논문에서는 CSB 기반 클라우드 마켓플레이스를 통해서 다음과 같은 효과가 기대된다.

1. 국내외 다양한 클라우드 서비스를 비용과 지역적인 특성에 맞추어서 서비스를 전개함으로써 경제성 확보
2. CSB를 통해서 국내외 클라우드 인프라를 통해서 다양한 소프트웨어를 글로벌로 손쉽게 전개
3. 마켓플레이스를 통해 PaaS 기반 컴포넌트 및 오픈 소스 기반을 확대하여 다양한 소프트웨어 창작 및 융합 서비스 기반 확산

클라우드 마켓플레이스 기술은 국내에서는 아직 초보적인 단계이지만 위에서 언급한 효과를 기반으로 해외 유수의 대기업에서는 이미 실증 단계를 넘어서 상용화 단계로 진입하고 있다.

#### 6. 결론

국내외적으로 클라우드는 향후 IT 서비스의 근간이 될 것임을 누구도 부인할 수 없다. 이미 클라우드 서비스는 해외 유수의 사업자에 의해 시장이 잠식되고 있으며, 국내 사업자들은 틈새 시장을 바라보며 시장에 접근하고 있지만 기술을 선도하지 못하는 상황에서는 한계가 분명하다.

최근 Container 기술을 통해서 PaaS에 대한 붐이 일고 있지만, 이는 PaaS 기술에 대한 해외 솔루션 기업들의 왜곡된 마케팅에 동조된 혼란 상황으로 PaaS 기술을 소프트웨어 관점에서 연구하는 부분이 부족한 것이 사실이다.

소프트웨어 기반의 서비스 운영 환경(as-a-Service)를 보다 현실적/산업적으로 제공하기 위해서는 소프트웨어 개발 프로세스 및 활용에 대한 본질에 이해와 연구가 필요하다.

클라우드의 가능성은 무궁무진하지만 해외 기술을 쫓아가는 것으로는 국내 IT 생태계의 생존을 보장하기 어려운 상황이 도래하고 있다. 오픈 소스와 국내 IT 환경을 분석하여 우리 현실에 맞는 운영 환경을 정립해야 향후 클라우드 서비스 생태계에 적극적으로 대응할 수 있는 기반이 조성된다. 이를 위해서는 클라우드 소프트웨어의 유통 환경인 마켓플레이스의 도입과 확대가 필요하며, 이를 통해 소프트웨어 제작 및 유통 환경을 보장함으로써 다양한 IT 서비스가 확산으로 클라우드 시대의 경쟁력을 확보/유지할 수 있다.

### 참고문헌

- [1] “Platform as a service” from Wikipedia,  
“[https://en.wikipedia.org/wiki/Platform\\_as\\_a\\_service](https://en.wikipedia.org/wiki/Platform_as_a_service)”
- [2] “Cloud broker” from Wikipedia,  
“[https://en.wikipedia.org/wiki/Cloud\\_broker](https://en.wikipedia.org/wiki/Cloud_broker)”
- [3] Kief Morris “Infrastructure as Code: Managing Servers in the Cloud” 1st Ed. O'REILLY, Martin  
Fowler's Infrastructure as Code  
“<https://martinfowler.com/bliki/InfrastructureAsCode.html>”
- [4] What is a Container ?  
“<https://www.docker.com/what-container>” from Docker Inc.