

SaaS 기술 개발 동향

Technical Trend Analysis of SaaS Development

클라우드 컴퓨팅 특집

김형환 (H.H. Kim)	SW서비스연구팀 책임연구원
원희선 (H.S. Won)	SW서비스연구팀 선임연구원
오병택 (B.T. Oh)	SW서비스연구팀 선임연구원
이상민 (S.M. Lee)	SW서비스연구팀 선임연구원
박경현 (K.H. Park)	SW서비스연구팀 선임연구원
이지현 (J.H. Lee)	SW서비스연구팀 선임연구원
양경아 (K.A. Yang)	SW서비스연구팀 선임연구원
이원재 (W.J. Lee)	SW서비스연구팀 연구원
허성진 (S.J. Hur)	SW서비스연구팀 선임연구원
정문영 (M.Y. Chung)	SW서비스연구팀 선임연구원
김원영 (W.Y. Kim)	SW서비스연구팀 책임연구원
전성익 (S.I. Jun)	SW서비스연구팀 책임연구원
최 완 (W. Choi)	SW서비스연구팀 팀장

목 차

-
- I. 서론
 - II. SaaS 참조모델
 - III. SaaS 기술 개발 동향
 - IV. 결론

최근 국내 소프트웨어 시장에서 소프트웨어를 서비스로 사용하는 SaaS에 대한 관심이 부각되고 있다. 본 고에서는 SaaS의 정의와 개념을 살펴보고 핵심기능인 멀티테넌시를 구현하기 위한 참조모델을 소개한다. 또한, 해외 및 국내의 개발 사례를 알아보고 국가 플랫폼 과제의 일환으로 2009년부터 한국전자통신연구원에서 수행하는 SaaS 플랫폼 개발과제를 소개하며 앞으로의 SaaS 플랫폼 개발을 전망해 본다.

I. 서론

요즘 소프트웨어 시장의 주요 트렌드로 클라우드 컴퓨팅, SaaS, 유틸리티컴퓨팅, SOA, 웹 2.0, RIA 등을 꼽을 수 있다. 이들의 공통점은 서비스에 있으며 서비스는 시장의 주된 성장동력으로 자리매김하고 있다[1]. 이러한 시점에 소프트웨어를 서비스로 발전시킨 SaaS 기술에 대한 관심은 자연스럽다.

SaaS 기술은 가트너가 발표한 2009년 10대 전략기술 중에서 첫번째와 두번째를 차지한 가상화와 클라우드 컴퓨팅에 있어서 애플리케이션 가상화 기술에 속하며 클라우드 컴퓨팅의 핵심요소 기술이다[2]. 그리고, SOA, 웹 2.0, RIA들은 소프트웨어를 구현하고 사용, 관리하는 방식의 변화를 설명하는 개념인데 비해서 SaaS는 소프트웨어 유통방식의 변화를 설명하는 개념이다[3].

SaaS는 사용자가 필요한 소프트웨어를 인터넷을 통해 온라인 서비스로 이용할 수 있도록 하는 최신의 소프트웨어 배포 모델로 정의할 수가 있으며 [4],[5] 또한, 응용 소프트웨어를 인터넷을 통하여 다수의 사용자에게 온라인 서비스로 제공하는 기술로도 정의할 수 있다. 즉, 사용자는 인터넷을 통해 소프트웨어를 사용하고 그에 대한 비용만 지불하는 방식으로 복잡한 소프트웨어 및 하드웨어의 관리라는 부담에서 벗어날 수 있다.

SaaS 비즈니스 모델은 고객의 초기 투자비용이 거의 없고 시스템 관리 필요성도 없어진다. 비용은

매월 또는 매년 서비스 기간에 따라 정해진 금액을 지불하거나 사용량만큼 지불하는 종량제가 있다. SaaS는 패키지화된 애플리케이션을 공급하는 호스 터드 애플리케이션 관리 방식과 소프트웨어 및 각종 지원을 인터넷을 통해 다수에게 제공하는 소프트웨어 온-디맨드 방식으로 분류할 수 있다[6].

SaaS를 기존의 ASP와 비교해 보면, 사용자의 입 장에서는 소프트웨어를 인터넷을 통해 사용하는 점 에서는 차별성이 없으나 사용자를 위한 커스터마이 징을 ASP처럼 소프트웨어 공급자가 하지 않고 사용 자가 직접 할 수 있다는 점에서 차별성을 갖는다. 소 프트웨어 공급자의 입장에서는 사용자의 커스터마 이징을 메타데이터를 활용해서 지원하며 사용자들 또는 사용자 그룹으로 표현되는 테넌트들을 하나의 소프트웨어 인스턴스로 지원(single instance mul- titenant)한다는 점에서 차별성을 갖는다. 이는 커스 터마이징에 많은 비용이 들고 인스턴스를 개별적으 로 띄우기 때문에 규모의 경제를 실현하지 못했던 ASP의 단점을 해결한다.

정리하면, SaaS 소프트웨어는 다음과 같은 특징 을 갖는다[7].

- 네트워크 기반으로 접근하고 관리하는 상업적으 로 사용 가능한 소프트웨어
- 각 고객 사이트가 아닌 중앙의 위치에서 활동을 관리하며 고객은 웹을 통해 소프트웨어에 접근
- 애플리케이션 전달은 일반적으로 일대일 모델보 다는 일대다 모델에 가까우며, 여기에는 아키텍 처, 가격, 파트너링, 관리 특성포함
- 중앙화된 기능 업데이트로 패치와 업그레이드 다운로드 필요를 없앴

SaaS를 구현하는 아키텍처는 다양한 테넌트들을 고려한 용이한 환경설정(configurability), 다중 테 넌트 지원 정도(multitenant efficiency), 그리고 확 장성(scalability)의 특성들을 기준으로 그 성숙도를 다음과 같이 분류할 수 있다[8].

- Level 1: Ad Hoc/Custom

● 용 어 해 설 ●

테넌트(tenant): 여러 사용자가 속한 하나의 조직으로 회 사, 기관, 단체 등 사용자 집단

멀티테넌시(multitenancy): 개별 사용자(테넌트)의 결정 에 의하여 설정되고 표현되는 기능, 컴퓨터의 자원들을 분할하여야 하며 여러 사용자들을 하나의 애플리케이션 인스턴스로 지원하여야 함

하이퍼바이저(hypervisor): 가장 기본적인 가상화 요소로 물리적인 리소스로부터 운영체제와 애플리케이션을 분 리해주는 소프트웨어, 자체 커널을 가지고 있으며 하드 웨어에 직접 설치됨

전통적인 애플리케이션 제공 서비스인 ASP와 유사한 형태로 각 고객별로 별도의 애플리케이션 인스턴스를 제공하고 커스터마이징하므로 관리가 복잡하고 비용이 큼

• Level 2: Configurable

각 고객별로 별도의 애플리케이션 인스턴스를 제공하고 서비스 내의 설정으로 커스터마이징이 가능하지만 멀티테넌시를 지원하는 것은 아니며 공급자는 대량의 애플리케이션 인스턴스들을 지원하기 위해서 충분한 하드웨어와 저장공간을 제공해야 함

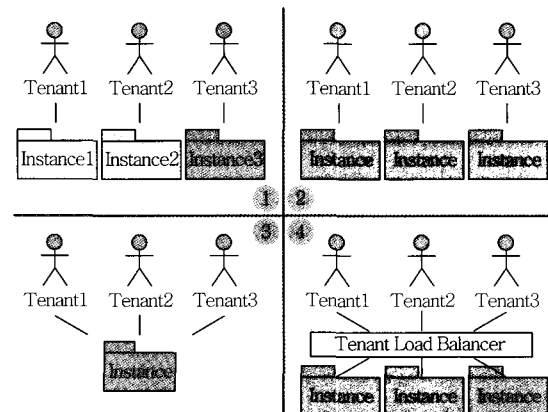
• Level 3: Configurable, Multi-Tenant-Efficient

모든 고객을 하나의 인스턴스로 지원하고 메타데이터를 통해 커스터마이징하며 멀티테넌시를 제공함. 레벨 2에 비해서 공급자는 효율적인 자원관리 가능

• Level 4: Scalable, Configurable, Multi-Tenant-Efficient

부하분산 시스템(load-balanced farm)에서 모든 고객을 지원하는 가장 성숙한 모델로 고객의 데이터를 분산 관리하며 재구성 가능한 메타데이터로 각 고객별 커스터마이징을 지원

위에서 설명한 SaaS 아키텍처의 네 가지 성숙도 모델은 (그림 1)과 같이 표현된다.



<자료>: Microsoft [8]

(그림 1) SaaS 성숙도 모델

II. SaaS 참조모델

SaaS의 멀티테넌시란 소프트웨어의 단일 인스턴스가 SaaS 공급자의 서버에서 실행되면서 다중 테넌트들을 지원하는 소프트웨어 아키텍처에 있어서의 원칙을 일컫는다[9]. 한편, 클라우드 컴퓨팅에서는 테넌트들간에 하부의 컴퓨팅 자원들을 공동으로 사용하는 것을 클라우드 컴퓨팅의 필수적인 특성으로 보는데 이렇게 논리적으로는 서로 분리되어 있는 테넌트들간의 컴퓨팅자원의 공유를 멀티테넌시라고 일컫기도 한다[10].

Gartner는 현존하는 멀티테넌시의 모델들을 (그림 2)와 같이 발표했다. 가장 왼쪽의 그림은 기업체에서 애플리케이션을 설치하여(on-premise) 운영하는 구조의 전체 스택을 설명하는 것으로 애플리케이션은 WebLogic, WebSphere, JBoss 또는 .NET 프레임워크 등으로 구축되는 애플리케이션 플랫폼에서 실행되고 애플리케이션 플랫폼은 주로 DBMS로 구축되는 데이터 플랫폼을 사용한다. 애플리케이션 플랫폼은 Linux, Windows 등과 같은 운영체제에 구축되는 시스템 인프라스트럭처를 하부에 두고 있다. 대형 기업체는 여러 개의 지사들을 위해서 중앙에 이러한 플랫폼을 구축하는데 이런 경우를 내부 클라우드(private cloud)라고 부르기도 한다.

클라우드 공급자는 특정 애플리케이션 플랫폼을 하부에 두고 작성한 애플리케이션들을 제공하는데 이때, 애플리케이션 플랫폼은 시스템 인프라스트럭처에 호스팅한 데이터 플랫폼을 사용한다. 비즈니스 애플리케이션의 아래에 위치하는 컴퓨팅 자원들은 계층(layered) 구조이기 때문에 애플리케이션의 멀티테넌시는 하단의 계층 중 어느 계층 또는 계층들에도 구현될 수 있다. (그림 2)에서, 모델 I ~ 모델 V는 각각 애플리케이션의 하부에 위치하는 컴퓨팅 자원들의 공유에 대한 다른 접근을 보여주고 있다. 모델 I은 아무것도 공유하지 않는 형태로, 사용자의 입장에서는 멀티테넌시를 경험하겠지만 각 사용자가 자신만의 전용 리소스 스택과 애플리케이션을 소유하므로 진정한 멀티테넌시는 아니다. 모델 II는

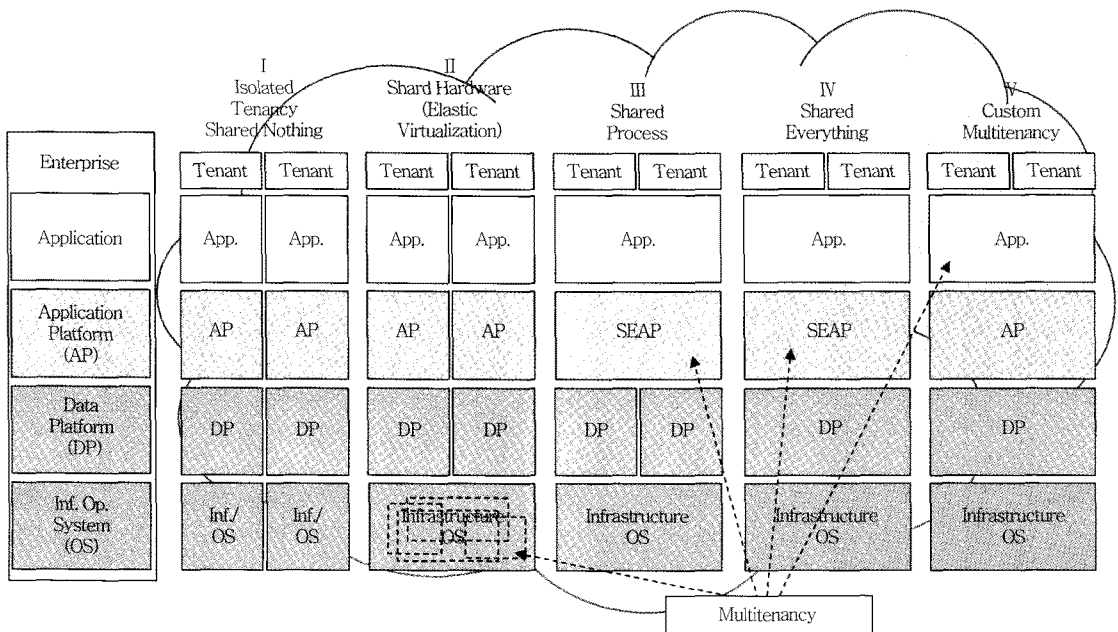
하드웨어의 공유를 통해 멀티테넌시를 제공하는 형태로, 테넌트별로 전용 리소스 스택을 갖는 점에서 모델 I 과 유사하지만 하이퍼바이저를 기반으로 한 가상화 기술을 이용하여 하부의 하드웨어를 테넌트 간에 공유하도록 하므로 멀티테넌시가 될 수 있다.

클라우드 시스템으로 구분하는 핵심기준은 공유에 있어서의 탄력성이다. 이는 애플리케이션이 필요한 경우에 공유 자원을 할당 받아서 사용하고 필요 없는 경우에는 다시 반납하는 것을 뜻한다. 이 관점에서, 가상머신들은 애플리케이션에 대한 컴퓨팅 파워의 할당 및 회수를 자동으로 실행해야 하는데 Zen이나 VMware 같은 기본적인 하이퍼바이저들은 이와 같은 자동기능이 없다. 클라우드로서의 가치가 있으려면 가상머신 플랫폼에 대한 자동 할당 및 회수 기능이 추가되어야만 한다.

하드웨어 공유 기반 멀티테넌시 모델의 주된 장점은 하위버전 호환성(backward compatibility)에 대한 비용의 절감이다. 모델 II의 접근 방식의 예로 Archer Technologies의 SmartSuite Framework, Bungee Labs의 Bungee Connect, Microsoft의 Azure Services Platform, 그리고 IBM의 Blue

Cloud를 들 수 있다.

모델 III, IV는 SaaS 플랫폼에서 멀티테넌시를 제공한다. 즉, 다중 테넌트들의 트랜잭션은 단일 애플리케이션 인스턴스 컨테이너에서 다른 테넌트들의 영역을 침범하지 않도록 지원되며 이를 위해, 컨테이너는 테넌트들에 대한 자원 할당을 관리하도록 설계되어야 한다. 이를 SEAP이라고 하며 이들 대부분은 JVM의 예와 같이 트랜잭션 수행을 위해 메타데이터를 인터프리트하는 특수한 가상머신을 사용한다. 애플리케이션의 실행에 메타데이터 기반 인터프리터를 사용하면 테넌트별 애플리케이션 커스터마이징의 생산성을 크게 높이고 다양한 커스터마이징을 할 수 있다. 클라우드 컴퓨팅에 대한 가장 큰 장벽은 테넌트들이 자신들의 비즈니스 데이터들을 외부에 두는 것을 꺼리는 것이다. 이 때문에, DBMS를 테넌트별로 운영하거나 테넌트 내에 두기도 한다. 모델 III의 예로는 Cordys Platform, Magic Software Enterprises의 uniPaaS, 그리고 Relational의 Long-Jump 등이 있다. 모델 IV는 모델 III과 같이 SEAP 기반 멀티테넌시 모델이지만 데이터플랫폼도 테넌트들에 의해 공유되도록 함으로써 공유의 관점에서 보



<자료>: Gartner, Dec. 2008.[10]

(그림 2) Gartner Reference Architecture for Multiitenancy

다 더 발전된 형태이다. 플랫폼 제공자들은 주로 제삼자(3rd party) DBMS를 사용하지만 모든 테넌트들에게 한 개의 논리적인 인스턴스를 할당하고 테넌트 데이터 격리(tenant data isolation)는 SEAP에서 관리한다. 이 모델은 자원 공유를 세밀하게 할 뿐 아니라 테넌트들, 애플리케이션들 그리고 사용자들의 진입(on-ramping)과 진출(off-ramping)-테넌트 추가 및 삭제, 애플리케이션 도입(deploy) 및 폐기(with draw), 사용자 등록 및 삭제-이 빠르고 그 비용이 저렴하다. 이 모델의 예로는 Coghead Service, Fore Soft의 dbFlex, Google App Engine, RollBase Platform as a Service, Salesforce.com의 Force.com, Microsoft의 Titan 등이 있다.

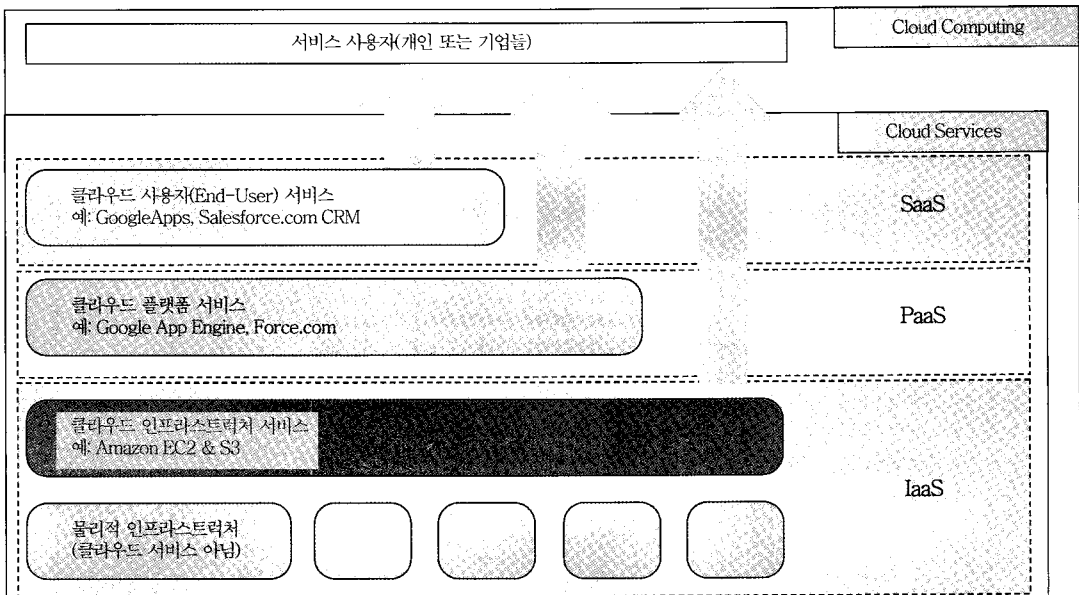
마지막으로, 모델 V는 전통적인 제삼자 플랫폼에 기존의 프로그래밍 언어로 애플리케이션 자체에 멀티테넌시를 구현-애플리케이션 코드에 멀티테넌트 격리의 원칙을 구현-한 형태이다. 비즈니스 애플리케이션에 테넌트 격리와 커스터마이징 알고리즘을 구현하려면 훌륭한 기술력이 있는 개발팀이 필요하며 광범위한 영역의 기술을 확보해야 하고 애플리케이션에 구현된 이런 시스템소프트웨어를 오랜 기간

관리해야 하는 책임 등의 비용이 매우 높다. 제삼자 SEAP가 없었던 초기에는 이 모델과 같이 애플리케이션이 멀티테넌시를 지원하는 방식이었다. 오늘날 일부 SaaS 공급자들은 먼저, 애플리케이션의 한 부분으로서 플랫폼을 개발하였고 다음으로, 비즈니스 애플리케이션 코드에서 플랫폼 코드를 추출해서 제품화 했다. 이런 방식의 공급자들로는 Archer Technologies, ForeSoft, Parallels, 그리고 Salesforce.com 등이 있다.

클라우드 컴퓨팅 서비스에서 SaaS의 위치는 (그림 3)과 같다.

IaaS는 하부의 하드웨어와 데이터 저장소 등을 서비스로 제공하고, PaaS는 이를 기반으로 애플리케이션의 실행기반을 제공하며, SaaS 서비스는 이들을 기반으로 실행되는 애플리케이션을 서비스로 제공한다.

지금까지 SaaS의 핵심인 멀티테넌시를 구현하기 위한 클라우드 컴퓨팅에서의 SaaS 참조모델과 클라우드 서비스에서의 SaaS 위치를 살펴보았다. 다음 장에서 해외 및 국내, 그리고 ETRI에서의 기술개발 동향을 소개한다.



<자료>: 디지털테일리[11]

(그림 3) 클라우드 컴퓨팅 서비스의 분류

Ⅲ. SaaS 기술 개발 동향

1. 해외 기술 개발 동향

가. Salesforce.com

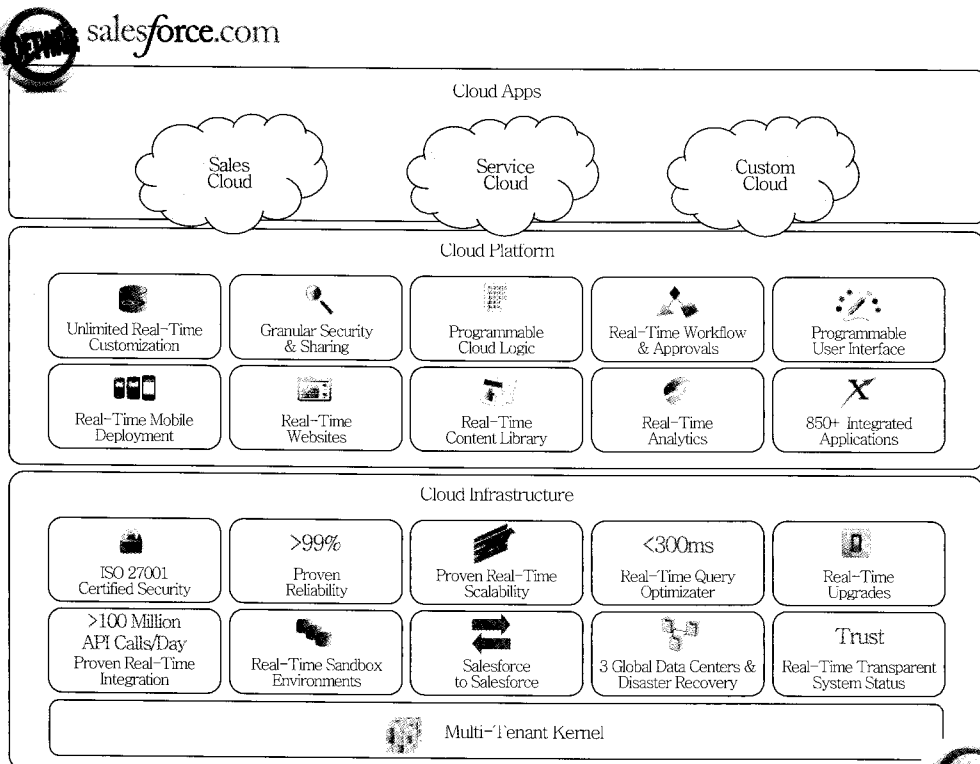
SaaS 기반 플랫폼 관련 선두업체들 중의 하나인 Salesforce.com은 올해로 창사 10주년을 맞이하였으며, CRM 솔루션으로 시작하여 지금은 클라우드 컴퓨팅 분야까지 진출하고 있다.

Salesforce.com은 최근 (그림 4)와 같이 ‘Cloud Infrastructure’를 기반으로 Force.com이라고 부르는 PaaS 플랫폼인 ‘Cloud Platform’과 그 상위에 SaaS 응용 솔루션 마켓플레이스인 ‘Sales Cloud’, 지식 기반의 고객 응대를 위한 ‘Service Cloud’, 그리고 고객이 직접 응용 솔루션을 개발하고 커스터마

이징 할 수 있도록 ‘Custom Cloud’라는 이름으로 SaaS 플랫폼과 클라우드 플랫폼 기술을 개발하여 새로운 시장을 공략하고 있다.

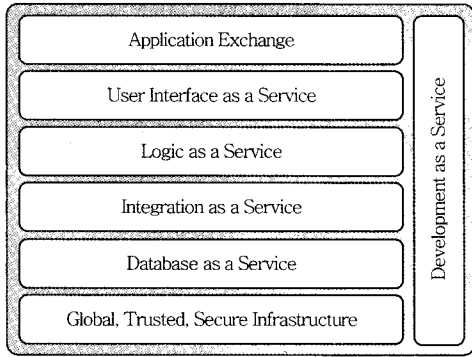
(그림 4)에서 ‘Cloud Platform’에 해당하는 Force.com 플랫폼은 (그림 5)에서와 같이 ‘Global, Trusted, Secure Infrastructure’ 스택, ‘Database as a Service’ 스택, ‘Integration as a Service’ 스택, ‘Logic as a Service’ 스택, ‘User Interface as a Service’ 스택, 그리고 ‘Application Exchange’ 스택으로 구성 되어 있다.

이들 중 가장 하단부의 ‘Global, Trusted, Secure Infrastructure’ 스택은 데이터 센터와 보안 관련 모듈로 구성되어 있고, ‘Database as a Service’ 스택은 관계형 테이블과 같은 data object들을 생성하며, 그러한 object들을 명세하기 위한 meta data를 사용할 수 있도록 지원한다. ‘Integration as a



<자료>: <http://www.salesforce.com>[12]

(그림 4) Salesforce.com 클라우드



<자료>: <http://www.salesforce.com>[13]

(그림 5) Force.com 스택과 서비스

Service' 스택은 타 애플리케이션들과의 데이터 연동이나 통합을 위해 타 시스템으로의 연동이 가능한 SOAP 웹서비스 API를 제공한다. 또한 Force.com 플랫폼은 고객사의 비즈니스 프로세스와 요구사항에 맞게 워크플로를 쉽게 생성할 수 있도록 공통적이며 재사용 가능한 컴포넌트들을 활용할 수 있는 'Logic as a Service' 스택을 가지고 있다.

유연성을 더 높이기 위해서는 Apex라는 온디맨드 프로그래밍 언어를 사용하여 가상적으로 비즈니스 로직과 기능을 포함시킬 수 있다[12].

Force.com은 플랫폼상에서 실행되는 애플리케이션의 사용자 인터페이스를 새로 만들고 커스터마이징하기 위해 'User Interface as a Service' 스택에서 두 가지 방법을 제공한다.

애플리케이션에 온라인 설정 공간을 통해 간단히 드래그앤드롭(drag-and-drop) 인터페이스를 사용하여 페이지의 레이아웃이나 데이터 필드를 변경하거나 다른 사용자를 위해 다른 뷰(view)를 생성할 수 있다. 다른 방법으로는 Force.com에 포함된 'Visualforce'라는 도구를 이용해서 개발자가 사용자 인터페이스를 생성할 수 있다. 'Application Exchange' 스택은 Force.com(PaaS 플랫폼) 위에 애플리케이션 제공자들이 응용 애플리케이션을 개발해서 운영하여 마켓플레이스를 형성할 수 있도록 지원하는 스택이다. Force.com 상에서 운용하는 Application Exchange(AppExchange.com)라는 마켓플레이스에서는 300여 개의 응용 애플리케이션을 Sales-

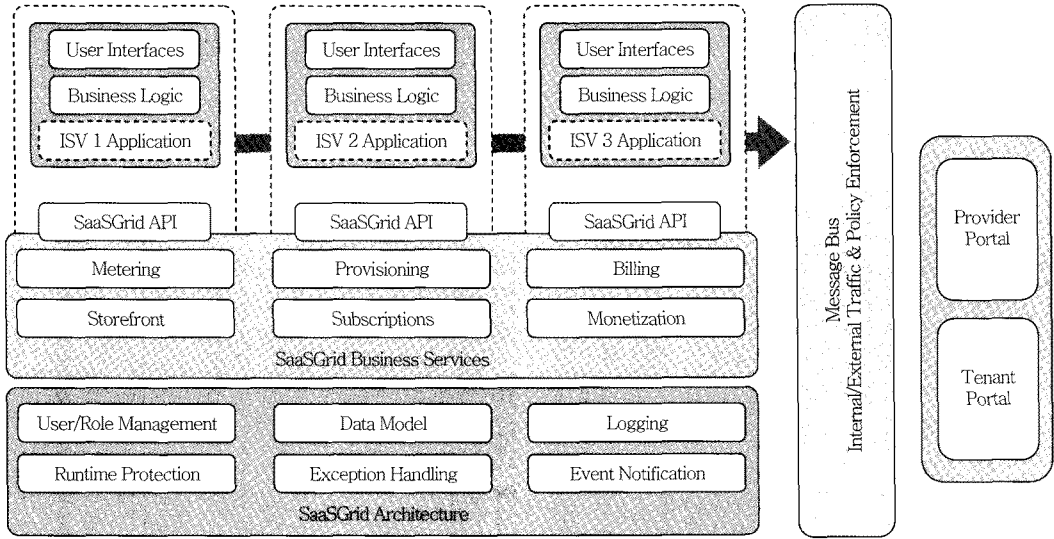
force.com이 무료로 제공하고 있으며, 500개의 애플리케이션이 애플리케이션 제공자들에 의해 유료로 운영되고 있다.

Salesforce.com은 최근 모바일 에디션 프로그램을 경량화한 스마트폰용 CRM 솔루션을 선보였는데 Black Berry를 비롯해 윈도모바일폰, 아이폰 등에서 사용할 수 있도록 하여 업무 업데이트와 일정관리, 고객 서비스 요청 검토 등의 기능이 지원된다. 또한 최근에는 'Force.com cloud infrastructure'라는 이름으로 클라우드 컴퓨팅 시장에도 진출하고 있다. 이와 더불어, Salesforce.com의 국내 파트너 업체인 (주)다우기술은 2007년 Salesforce.com과 제휴를 맺고 국내에서 본격적으로 SaaS 서비스 시장에 진출하여 Salesforce.com 솔루션의 국내 영업을 담당하고 있다.

나. Apprenda의 SaaSGrid

Apprenda에서 개발한 SaaSGrid는 ISV들을 위한 범용 SaaS 플랫폼으로서 멀티테넌트를 지원하는 애플리케이션의 개발과 배포를 지원하기 위한 메커니즘을 제공한다[14]. 소프트웨어 개발 회사에서는 SaaSGrid 애플리케이션을 개발하기 위해 Visual Studio와 같은 통합 개발 환경에서 SaaSGrid SDK를 사용하여 비즈니스 코드, 사용자 인터페이스, 데이터베이스 등을 작성한다. (그림 6)은 계층화된 SaaS-Grid의 구조를 보인 것이다.

SaaSGrid의 주요 배경요소로 SaaS API, 테넌트 포털(tenant portal), 프로바이더(providers), SaaS 애플리케이션 등이 있다. 테넌트 포털은 테넌트 계정 프로파일, 사용자 접근 통제, 구축되는 SaaSGrid 애플리케이션 정보 등과 같은 제어를 테넌트에게 제공하는 일종의 SaaSGrid 애플리케이션이다. 프로바이더는 SaaSGrid 애플리케이션을 공개하거나 업로드할 수 있는 특별한 권한을 갖는 테넌트를 의미한다. SaaS 애플리케이션은 한 개 이상의 웹 서비스, 사용자 인터페이스, 데이터베이스 스키마, SQL Script를 포함하는 계층 구조를 가진다. SaaSGrid 개발 환경은 WCF 3.0, 사용자 인터페이스는 ASP .NET 사용자



<자료>: Appenda[14]

(그림 6) Layered SaaSGrid

인터페이스, 데이터베이스 스키마는 SQL Server 2000을 사용하여 구현되었다.

SaaSGrid는 멀티테넌시, 거래에 따른 비용 정책, 시장의 변화에 따른 가격 정책 설정, 구독 관리, 사용자 관리 및 사용자들의 역할 관리, 프로비저닝 등의 기능을 지원한다[15].

1) 멀티테넌시

SaaSGrid 애플리케이션 공급자는 SaaSGrid SDK를 이용해 단일 테넌트 방식으로 SaaSGrid 애플리케이션을 작성한다. 또한 SDK를 이용해 로컬 환경에서 작성된 애플리케이션을 테스트 할 수 있다. 테스트 과정을 거친 후 SaaSGrid 애플리케이션이 SaaSGrid 클라우드 인스턴스[15] 상에 배포될 때 SaaSGrid가 자동으로 단일 인스턴스 멀티테넌트 구조를 애플리케이션에 부여한다.

2) 비용 정책

SaaSGrid는 SaaS 애플리케이션 공급자가 어떤 거래에 비용을 책정할지를 정의할 수 있는 기능을 제공하며 설정된 가격 정책에 따라 요금을 청구하고 그 비용을 애플리케이션 공급자의 은행 계좌에 예금

할 수 있도록 해준다.

3) 시장 변화에 따른 가격 정책

SaaSGrid는 애플리케이션 공급자가 시장변화에 민첩하게 대응할 수 있도록 가격 결정 기능을 제공한다. 애플리케이션 공급자는 시장의 조건이 변할 때마다 적합한 가격정책을 적용하기 위해 여러 개의 가격정책 사항을 정의해두고 웹 인터페이스를 통해 적합한 가격 정책을 선택한다.

4) 구독 관리

테넌트가 SaaS 애플리케이션에 가입하였을 때 구입주문이나 송장(invoice) 정보들을 생성한다. 또한 고객들에게 그들의 가입 사항에 대한 수정 및 갱신을 가능하게 한다.

5) 사용자 관리 및 사용자 역할 관리

테넌트는 비용을 지불하고 가입함으로써 SaaS 애플리케이션을 사용한다. 테넌트는 SaaS 플랫폼 레벨에서 최종 사용자를 생성할 수 있으며 애플리케이션에 대한 액세스 승인이나 권한 설정 등을 정의할 수 있다.

6) 프로비저닝

테넌트가 SaaS 애플리케이션에 빠르게 접근할 수 있는 방안들을 제공한다. 신규 테넌트가 생성되었을 때 네트워크 리소스 할당, 데이터베이스 생성, 계정 정보 저장과 같은 기능들을 수행한다.

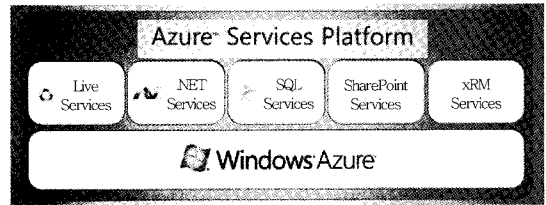
다. Microsoft의 “Software+Services”

마이크로소프트는 “소프트웨어+서비스” 전략을 내세우고 있다[16]. 이 전략 하에서 인터넷 서비스, 클라이언트 소프트웨어, 서버 소프트웨어를 모두 활용하는 솔루션을 제공한다는 계획이다. SaaS, 클라이언트 소프트웨어, 온프레미스(on-premise) 서버 배포 모델을 선택할 수 있게 함으로써 고객의 요구와 상황에 맞는 솔루션을 제공할 수 있음을 강조하고 있다. 이와 같이, 마이크로소프트는 SaaS를 제공하면서도 기존 소프트웨어 모델과의 공존을 모색하고 있다.

마이크로소프트는 Dynamics라는 이름 아래 회계, CRM, SCM 솔루션들을 출시하였다. Dynamics 제품들은 MS SQL 서버, .NET 프레임워크에 기반하여 개발되었다. Dynamics 제품들은 다른 MS 제품들과의 통합이 쉬우며, 기존 MS 제품들과 사용자 인터페이스가 비슷하다는 장점을 내세우고 있다. Dynamics 제품들은 온프레미스 형태로 설치될 수도 있고, MS 파트너에 의해 호스팅 될 수도 있으며, 마이크로소프트에 의해 직접 호스팅 될 수도 있다. 마이크로소프트에 의한 호스팅 서비스는 SaaS로 볼 수 있으며, Salesforce.com 등과 직접적인 경쟁관계를 가진다.

마이크로소프트는 웹브라우저 상에서 사용할 수 있는 웹기반 오피스 제품군을 내놓을 예정이다. 이 오피스 제품군 역시 on-premise 형태로 설치될 수도 있고, 가입(subscription) 기반으로도 제공될 예정이다. 추가적으로, 모바일 상에서 사용할 수 있는 오피스 제품군도 발표하였다.

마이크로소프트는 Azure Services Platform을 클라우드 컴퓨팅 플랫폼으로 개발하고 있다[17].



<자료>: [http://blogs.zdnet.com/microsoft/\[18\]](http://blogs.zdnet.com/microsoft/[18])

(그림 7) Azure 서비스 플랫폼

(그림 7)과 같이 Azure Services Platform은 Windows Azure, .NET Services, SQL Services, Live Services, SharePoint Services, xRM Application Services 등으로 구성된다. Windows Azure는 클라우드 컴퓨팅을 위한 윈도 기반 OS이다. .NET Services에서는 접근 제어, 서비스 버스 및 워크플로 관련 서비스들을 제공한다. SQL Services는 클라우드 컴퓨팅용 DB 서비스를 제공한다. Live Services는 데이터 접근 및 동기화 서비스를 제공한다.

마이크로소프트는 xRM 플랫폼을 PaaS 플랫폼으로 제공할 계획이다. xRM 플랫폼은 Azure Services Platform의 일부이며, Dynamics CRM 및 .NET 플랫폼을 사용하여 새로운 서비스들을 개발할 수 있을 것으로 보인다. 이는 Salesforce.com의 PaaS인 Salesforce.com과 경쟁할 것으로 예상된다.

라. Google의 AppEngine

2008년 4월 구글은 Google Application Engine(GAE)을 발표하고 서비스를 시작하였다. GAE는 구글의 검증된 인프라스트럭처를 기반으로 클라우드 컴퓨팅 패러다임을 구현하고, 전세계 개발자들을 구글 환경의 마켓플레이스로 유도하기 위한 플랫폼 비즈니스 전략 사례이다. GAE의 통합 개발 환경 하에서 서비스 개발자는 쉽게 웹 애플리케이션을 구축/배포 및 관리할 수 있으며, 구글의 다른 서비스들과 연동시키는 것이 가능하다. 특히 GAE는 수백만 명까지 증가하는 사용자를 수용할 수 있도록 시스템 자원의 자동 스케일링과 로드 밸런싱을 제공하는 레벨 4 수준의 SaaS 플랫폼 기술을 포함하고 있다. 따라서, 개발자는 다양한 서비스를 통해 증명된 구글

시스템의 가용성과 보안 서비스를 그대로 보장받으며 서버 확장을 고려할 필요 없이 구글 플랫폼에 자신의 서비스를 제공할 수 있는 것이다.

GAE의 데이터 저장 계층은 구글 시스템의 확장성을 지원하는 핵심 기술로서 구글의 분산 파일 시스템과 빅테이블로 구성되어 있다. 빅테이블은 수천 개의 컴퓨터 및 수만 개의 하드디스크로 구성되어 수백만 개의 엔티티 저장을 지원하는 초대용량 저장 기술로서 사용자에게 이들 모두를 하나의 테이블로 보이게 하는 것이 특징이다. 또한 SQL과 유사한 질의언어인 GQL을 제공하며 트랜잭션 처리 기능도 제공한다.

애플리케이션은 기반 운영체제 시스템에 대한 접근이 제한되어 로컬 파일 쓰거나 서브프로세스 생성 등의 작업이 불가능한 샌드박스 환경에서 수행된다. 이러한 애플리케이션의 고립된 실행 환경을 통해 GAE는 애플리케이션을 웹 서버의 하드웨어, 운영체제, 그리고 물리적 위치에 무관하게 여러 개의 서버에 분산시켜 수행시키는 것이 가능하다.

GAE의 시스템 구성요소는 다음과 같다.

1) 웹 서빙 인프라스트럭처

일반적인 웹 기술 지원 및 사용자의 HTTP 요청을 애플리케이션과 코드로의 자동 연결을 지원한다.

2) 파이썬 및 자바 실행 환경

파이썬 버전 2.5.2와 Django 등의 웹 개발 프레임워크를 제공하며 자바 환경으로는 JRE 6플랫폼을 제공한다. 샌드박스 제약에 위배되는 API 호출 및 C 언어 확장은 금지하고 있다. 향후 개발자 요구사항에 따라 PHP, Perl, Ruby 등의 다양한 언어 환경을 계속 추가할 예정이다.

3) 로컬 개발 환경

로컬 컴퓨팅 환경에서의 코드 개발 지원을 위한 GAE SDK를 제공하고, 이클립스 플러그인을 제공한다. SDK는 파이썬 및 자바 런타임, 데이터 스토어, 이미지, 메일, 맵캐시, URL Fetch, 사용자 API 등으로 분류된다.

4) 웹기반 관리자 콘솔

자신의 애플리케이션을 관리하고 디버깅하기 위한 통합 도구이다.

5) 데이터 스토어

질의 엔진과 트랜잭션 기능을 제공하는 분산 저장 시스템이다. 스키마가 정의되지 않으므로 애플리케이션 코드에서 데이터 구조를 정의하고 해석해야 한다.

GAE는 매달 약 5백만 페이지 뷰의 서비스를 제공할 수 있도록 최대 500MB 저장소, 200M megacycles/day의 CPU 자원, 총 10GB 대역폭까지는 무료로 제공하며, 무료 용량 이상의 서비스에 대해서만 과금된다.

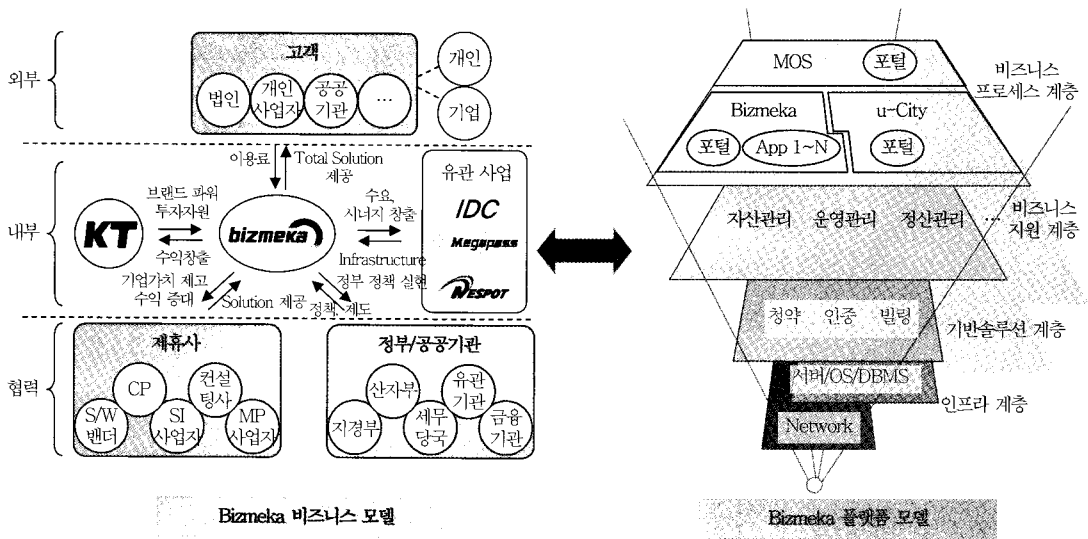
2. 국내 기술 개발 동향

가. KT의 SaaS 기술 개발 동향

KT는 기존의 인터넷망 가입자에게 필요한 서비스를 제공하는 비즈니스 모델에서 벗어나 소프트웨어가 중심인 IT 컨버전스를 주도하기 위해 변혁 중이며 이에 대한 대표적 예로 비즈메카라는 임대 플랫폼 서비스가 있다[19].

비즈메카는 KT가 중소기업 및 중소 상공인에게 필요한 e-비즈니스 솔루션, 콘텐츠, 컨설팅, 교육 등을 통합적으로 제공하고 관리를 대행해주는 플랫폼 서비스이다. 지원하는 업종에 따라 기업의 협업 및 정보 공유를 위한 비즈메카 엔터프라이즈, 중소 의료기관과 의료 민원의 전자적 처리를 위한 비즈메카 호스피탈, 신용카드 및 후불제 결재를 위한 비즈메카 파이낸스 등으로 나뉘며, 현재 40여 개의 솔루션을 제공하고 있다. 이 중 미용실을 위한 비즈메카 헤어짱과 카센터를 겨냥한 비즈메카 하이웨이가 높은 시장점유율을 보이고 있다.

비즈메카 플랫폼(그림 8) 참조)은 초기에 서비스 관리, 고객 관리, 비즈니스 관리 등 3개의 영역별 플랫폼으로 구축되었으며 통합 인증, 빌링, 통합 운용 관리, 서비스 연결 프레임워크 등의 핵심 기능을 포함



Bizmeka 비즈니스 모델

Bizmeka 플랫폼 모델

<자료>: SaaS피아 개발 1차 워크숍, 2009.[19]

(그림 8) 비즈메카 비즈니스 모델과 플랫폼 모델

하도록 확장되었다. 비즈메카 플랫폼은 새로운 서비스가 웹 서비스 기술을 통해 손쉽게 추가되고 연계되어 개발되도록 지원하기 때문에 고객이 필요로 하는 맞춤형 서비스를 제공할 수 있다는 데 의의가 있다.

나. 공영 DBM의 SaaS 기술 개발 동향

공영 DBM은 1994년, 데이터베이스 마케팅 IT 솔루션 개발을 시작으로 CRM/DBM 컨설팅 서비스에서 솔루션 구축에 이르기까지 토탈 CRM 서비스를 제공해온 CRM 전문 회사로서, 외국인 솔루션으로는 구현할 수 없는 국내 기업 환경을 최대한 반영한 CRM 솔루션 개발 및 서비스 제공을 통하여 국내 CRM 업계를 선도하고 있으며, 특히 CRM을 구성하는 모듈들을 웹서비스 컴포넌트화 하여 개별 컴포넌트에 온라인 접근 가능한 서비스와 더불어 CRM 전체 솔루션에 대하여 온라인 서비스를 제공하고 있다.

다. AnyTNS의 SaaS 기술 개발 동향

AnyTNS는 국내 중소기업에 SaaS 솔루션을 저렴한 비용으로 CRM 외에도 다양한 서비스를 제공하는 업체이다. AnyTNS는 외산 플랫폼이나 획일적인 플랫폼만으로는 다양한 커스터마이징을 원하는

까다로운 국내 사용자들을 충족시키기는 어렵다고 판단하였다. 그래서 다른 국내 SaaS 업체와 다르게 자체적으로 멀티테넌시를 지원하는 SaaS 플랫폼(AnyPlatform)을 앞서 개발하였다.

AnyTNS에서 제공하는 서비스는 AnyPlatform을 기반으로 제공되며, 기업의 개별 요구사항을 충족시키기 위해서 SOA 기반의 웹 2.0, 리포팅 및 RIA 기술을 기반으로 다양한 커스터마이징을 통해 제공되고 있다.

AnyTNS는 SaaS의 핵심인 CRM 솔루션뿐만 아니라 그룹웨어, PMS, PRM 등 다양한 솔루션을 보유하고 있어, 사용자에게 토탈 서비스를 제공할 수 있다.

라. MANinSOFT의 SaaS 기술 개발 동향

MANinSOFT는 업무프로세스관리(BPM)를 기반으로 하는 중소기업용 업무처리 시스템을 SaaS 방식으로 공급하는 '스마트웍스닷넷' 서비스를 제공하고 있다[20]. 스마트웍스닷넷은 업무 공동 참여 기능을 통해 공동 의사 결정에 의한 업무 협업 기능과 업무 처리를 실시간으로 모니터링하는 프로세스 기반 업무 처리 방식을 지원한다. 스마트웍스닷넷은

업무처리 시스템 및 하드웨어, 그리고 구축 인력 등에 대한 임대 서비스를 제공하며, 고객이 원하는 업무 콘텐츠를 글로벌 등록 저장소에서 다운받아 사용할 수 있도록 하였다. 스마트웍스닷컴의 세부 기술은 크게 SaaS 플랫폼, SaaS 플랫폼 활성화 도구, 서버 플랫폼 등 세 가지로 나누어 볼 수 있다. MAN-inSOFT는 현재 출판, IT, 제약 등 다양한 분야의 중소 기업들을 대상으로 ‘스마트웍스닷컴’ 서비스를 제공하고 있다[21].

마. 다우기술의 Twincamp

2007년부터 세일즈포스닷컴과 전략적 제휴를 맺고 SaaS 시장에 뛰어들었던 다우기술은 2008년에 독자적으로 SaaS 기반의 팀 업무관리 솔루션인 트윈캠프를 개발하였다. 트윈캠프는 회사에서 업무를 총괄하는 팀장이 팀의 업무 진행상황을 파악하고 효율적으로 관리하도록 하며 팀원간 업무 일정 및 진행 경과를 공유하도록 지원하는 솔루션으로 별도의 시스템 관리 인력을 운용할 필요가 없이 인터넷을 통해 언제 어디서나 사용이 가능하도록 SaaS 방식으로 개발되었다. 또한, 서비스 이용료를 지불하는 방식이기 때문에 도입비용의 부담이 적은 것이 장점이다.

트윈캠프는 프로젝트 관리, 팀 업무 관리, 캘린더, 메일링 게시판, 웹 자료실, 맞춤형 보고서, CRM형 주소록, 근태관리 등을 주요 기능으로 제공하며, 현재까지 출시된 많은 업무관리 솔루션들이 기업 전체 또는 개인의 일정관리를 위해 제공되었던 것과 달리 팀 단위의 업무관리를 위한 기능들로 구성되어 있다.

트윈캠프의 가장 큰 특징은 정기, 비정기적 보고서의 생성과 관리가 자동으로 된다는 것으로 웹 자료실과 연동되어 보고서의 버전 관리가 되기 때문에 수정, 변경 이력이 기록되고 수정된 신규 데이터뿐만 아니라 기존 데이터까지도 보관, 열람이 가능하다.

다우기술은 2009년 SaaS 코리아 컨퍼런스에 참가하여 트윈캠프를 전시하고 사이버외교 사절단인 반크의 도입사례를 발표하는 등 트윈캠프의 고객유치에 앞장서고 있다.

바. ACS의 MES 및 u-Manufacturing

(주)에이시스는 군소 제조업체를 대상으로 선진 IT화 기반의 자율재구성형 u-MES 플랫폼을 통하여 생산자원을 디지털화함으로써 정보화, 지식화 및 지능화하여 생산현장을 지식집약적인 현장으로 바꾸어 생산성을 향상시키고 급변하는 시장에 대한 민첩성을 극대화하는 고도기술을 저가로 사용하게 하는 시스템을 개발하기 위하여 u-MES 관련 솔루션을 SaaS 방식으로 제공하는 시스템 개발을 계획하고 있다.

3. ETRI의 SaaS 기술개발 현황

가. SaaSSpia™

한국전자통신연구원은 다양한 애플리케이션 소프트웨어를 중소기업에 저비용으로 온라인 서비스화하기 위한 SaaS 플랫폼 개발과제를 2009년부터 수행하고 있다. SaaS 플랫폼에 대한 민간기업들의 관심과 기술수요가 큰 만큼 민간분야의 SaaS 관련 단체인 SaaS 코리아포럼과의 협조를 통해 개발과정에 민간기업의 의견이 충분히 반영되도록 하고 있다. SaaSSpia™는 SaaS와 Utopia를 결합한 것으로 개발될 SaaS 플랫폼의 명칭이다. 현재, 관련 기술과 개발 사례를 연구하고 민간기업의 요구사항을 취합하여 사용자 요구사항을 도출했고 이를 통해 개발 시스템의 아키텍처를 완성하고 있으며 조만간 시스템요구사항을 완성하여 상위설계를 진행할 예정이다.

SaaSSpia™에서 목표로 하는 중요 개발기술로는 크게 세 가지가 있는데, 고성능 메타데이터 처리 기술, 고성능 멀티테넌트 애플리케이션 실행엔진 기술, 그리고 테넌트별 데이터의 무결성을 보장하는 보안기술이다.

나. 개발환경

SaaSSpia™의 개발환경은 현재 구축 방안을 고려하고 있다. 개발환경에 대한 요구사항은 전통적인 웹 기반 애플리케이션의 개발환경과 큰 차이점은 없으나 라이선스 비용이 없는 공개 소프트웨어를 기반

으로 구축하는 것으로 추진하고 있다. 최근에 전자정부 프레임워크 표준화가 이루어졌는데 여기서의 개발환경은 라이선스 비용이 없는 공개소프트웨어들로 구축되었다[22]. 현재, 과제 일차년도의 빠른 프로토타입 구현을 위해서 전자정부 표준 프레임워크의 개발환경 도입을 고려하고 있다.

IV. 결론

본 고에서는 최근 국내 소프트웨어 시장에서 주요 트렌드로 부각되고 있는 SaaS 기술을 정의하고 관련한 개념들을 설명했다. 그 다음으로 SaaS 플랫폼을 구현하는 데 있어서 중요 특징인 멀티테넌시 구현을 위해 참조할 수 있도록 Gartner의 참조모델을 소개했고 클라우드 컴퓨팅 서비스에서 SaaS의 위치를 설명했다. 또한, 해외 및 국내의 기술 개발 동향을 소개했다.

SaaS 플랫폼에서의 멀티테넌시는 하이퍼바이저를 기반으로 한 가상머신 기술을 이용해서 하드웨어를 공유하는 방식과 데이터 플랫폼과 애플리케이션 플랫폼을 모두 공유(플랫폼의 전체 스택을 공유)하는 방식으로 구현될 수 있으나 진정한 클라우드 컴퓨팅의 이념을 실행하는 후자의 방식이 주류를 이룰 것으로 예상된다[10]. SaaSSpia™ 역시 후자의 방향으로 개발하되 고성능 메타데이터 처리 기술, 고성능 멀티테넌트 애플리케이션 실행엔진 기술, 그리고 테넌트별 데이터의 무결성을 보장하는 보안기술을 핵심으로 하여 개발될 것이다. SaaSSpia™는 온라인 소프트웨어 서비스 환경을 구축할 수 있도록 하며 국내 소프트웨어 산업의 플랫폼 기술 자립의 근간이 될 것으로 기대된다.

약어 정리

ASP	Application Service Provider
ISV	Independent Software Vendor
JVM	Java Virtual Machine

MES	Manufacturing Engineering System
PaaS	Platform as a Service
PMS	Project Management System
PRM	Partner Relationship Management
RIA	Rich Internet Application
SaaS	Software as a Service
SEAP	SaaS/cloud-enabled Application Platform
SOA	Software Oriented Architecture

참고 문헌

- [1] 신현석, "MS 클라우드 컴퓨팅과 애저 서비스 플랫폼 이해," ZDNet Korea, 2008. 12. 12.
- [2] 김영만, "웹기반 SaaS 플랫폼," 2008. 12. 17., <http://www.software.or.kr/ICSfile/afieldfile/2008/12/18/5.pdf>
- [3] 디지털타임스, "SaaS와 소프트웨어의 미래," 2006. 5. 4.
- [4] http://en.wikipedia.org/wiki/Software_as_a_service
- [5] What is SaaS?, <http://www.salesforce.com/naas/>
- [6] 전자신문, "[집중진단-SaaS, Web 2.0시대 연다] (상) SaaS 시대 개막," 2007. 2. 12.
- [7] <http://ko.wikipedia.org/wiki/SaaS>
- [8] Frederick Chong and Gianpaolo Carraro, "Architecture Strategies for Catching the Long Tail," Microsoft, Apr. 2006. (<http://msdn.microsoft.com/en-us/library/aa479069.aspx>)
- [9] <http://en.wikipedia.org/wiki/Multitenancy>
- [10] Yefim V. Natis, "Reference Architecture for Multitenancy: Enterprise Computing," in the Cloud, Gartner, Dec. 3, 2008, pp.4-8.
- [11] 디지털데일리, "클라우드 컴퓨팅과 비즈니스의 진화," 2009. 5. 21.
- [12] <http://www.salesforce.com/platform/>
- [13] "Force.com: Create and Run any Application, On Demand," Salesforce.com
- [14] SaaSGrid Quickstart, apprenda, 2008. 4.
- [15] Developer Guide, apprenda, 2008. 4.
- [16] Software-plus-Services, Microsoft, 2009.
- [17] Introducing the Azure Services Platform, David Chappell, Chappell & Associates, 2008.
- [18] Mary-Jo Foley, "Microsoft Takes Off Its xRM

- Platform-as-a-Service Gloves,” <http://blogs.zdnet.com/microsoft/?p=3053>, June 12, 2009.
- [19] 조강래, “bizmeke 플랫폼 & 서비스의 진화와 발전 방향,” SaaSpia 개발 1차 워크숍, 2009. 6. 4.
- [20] 스마트웍스닷컴 소개, MANinSOFT, 2009. 5.
- [21] BPM Conference 참여 기업 소개, 전자신문, 2008. 5.
- [22] 전자정부 표준 프레임워크 포털, <http://www.egovframe.go.kr/>