SaaS Technical Skills Guide

SaaS 아키텍처의 정의

> SaaS 아키텍처는 단지 하나의 기술 또는 디자인 패턴 이라기 보다 비즈니스의 성장과 수익 극대화를 위한 마인드셋과 접근법 입니다. 따라서 SaaS 아키텍트는 이 마인드셋과 접근법을 바탕으로 신속하게 테넌트를 온보딩 하고 서비스를 업데이트 하고 규모의 경제를 달성하며 확장 할 수 있는 아키텍처를 만드는데 집중 합니다.

SaaS 아키텍처 설계 과정은 단지 하나의 기술 또는 디자인 패턴을 다루는 과정 이라기 보다, SaaS 비즈니스의 성장과 수익 화라는 목표 달성을 위해 필요한 SaaS 아키텍처 핵심 요소를 구현하는데 적절한 전략과 기술을 선택하고 적용 하는 과정 입니다.

SaaS 아키텍처 주요 개념

SaaS 아키텍처 설계 시 고려되어야 하는 핵심 요소가 있습니다. 이것들을 이해하고 설계 한다면 성공 적인 SaaS 비즈니스를 위해 필요한 모범 사례 SaaS 아키텍처를 구현할 수 있습니다.

Copyright © Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Data Partitioning(데이터 파티셔닝)

멀티 테넌트 환경에서는 각 테넌트별 데이터를 어떻게 나누고 어떤 저장소에 저장 할지 고려해야 합니다. 예를 들어 데이터를 각각 다른 데이터베이스에 저장할 것인지 아니면 공유 구조를 사용할지 결정해야 합니다. 이러한 멀티 테넌트 스토리지 메커니즘 및 패턴을 일반적으로 데이터 파티셔닝이라고 합니다.

Onboarding(온보딩)

SaaS 서비스에 신규 테넌트를 생성하는 데 필요한 모든 요소를 성공적으로 생성 하고 구성하기위해 여러 설계 구성 요소의 조율이 필요할 수 있습니다. SaaS 아키텍처에서 이 프로세스를 테넌트 온보딩이라고 합니다. 테넌트 온보딩은 테넌트가 직접 시작하거나, 공급자가 관리하는 프로세스의 일환으로 시작될 수 있습니다.

Tenant isolation(테넌트 격리)

테넌트 격리는 멀티 테넌트 환경에서 각 테넌트가 다른 테넌트의 리소스에 액세스하는 것을 방지 하는 것을 말합니다.
테넌트 격리를 달성하기 위한 전략과 접근 방식은 다양 합니다. 따라서 테넌트를 격리 설계 시 테넌트 요구 사항, 비즈니스 도메인, 규정 준수, 배포 모델, AWS 서비스 선택을 고려하여 효과적인 구조를 선택해야합니다.

Tenant Tiering(테넌트 계층)

SaaS 애플리케이션은 종종 다양한 시장, 다양한 고객 프로파일 에 각기 다른 요금 및 경험을 제공합니다. 이러한 프로파일을 Tier(계층) 이라고 합니다. 서로 다른 계층의 다양한 요구를 지원하기 위해서는 각 계층의 경험을 형성할 수 있는 아키텍처 구조를 도입해야 합니다. 이러한 계층화 모델은 SaaS 솔루션의 비용, 운영, 관리 및 신뢰성에 영향을 미칠 수 있습니다.

SaaS Identity(SaaS 자격증명)

SaaS identity(자격 증명)는 사용자를 테넌트로 치환하고 이 테넌트에 대해 티어링 정보, 그리고 SaaS 서비스 자원에 대한 접근 권한을 부여한 일종의 테넌트 신원 정보를 말합니다. 이것을 바탕으로 각 테넌트는 SaaS 서비스 사용 시 다른 테넌트 데이터 접근이 제한 되고, 과금 되고, 모니터링 됩니다.

Metering and billing(측정 및 청구)

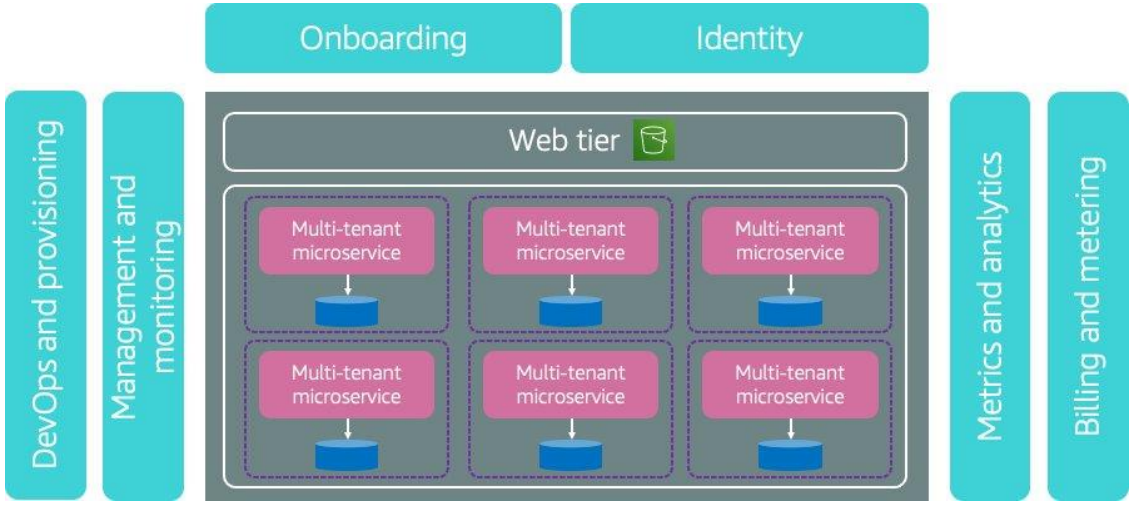
SaaS 제품은 고객의 서비스 사용량을 기준으로 비용이 결정되고 판매될 수 있습니다. 이를 위해서는 SaaS 서비스는 테넌트의 소비량을 측정하는 메커니즘 정의하고 도입해야 합니다. 이렇게 측정된 데이터는 빌링 시스템을 거쳐 테넌트 에게 청구 됩니다.

Copyright © Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

SaaS 아키텍처 구성 요소

SaaS 아키텍처 설계 시 기술적 고민은 일반적으로 두 가지 영역으로 시작합니다. 첫번째는 SaaS 서비스의 핵심 기능이 담겨 있는 SaaS 애플리케이션 영역 입니다. 이 영역은 테넌트 별로 독립된 자원을 제공 할지(Silo), 공동으로 사용 하는 형태의 자원을 제공 할지(Pool), 아니면 이 둘을 혼합 할지(Silo+Pool)의 결정에 따라 디자인 형태가 달라 집니다. 두 번째는 SaaS 서비스가 다수의 테넌트 에게 서비스 하면서 효율성과 민첩성 을 확보 하도록 지원 하는 기능들로 구성된 SaaS 공통 영역 입니다. 이 두 영역은 SaaS 아키텍처 모범 사례를 따르는 것이 좋습니다.

Copyright © Amazon Web Services, Inc. and/or its affiliates. All rights reserved.



SaaS 애플리케이션 영역과 SaaS 공통 영역 이 합쳐진 일반적인 SaaS 아키텍처의 구성 모습은 위 그림과 같습니다. 가운데 영역에 SaaS 애플리케이션 영역과 이를 둘러싼 SaaS 공통 영역이 위치 하고 있습니다. 앞에서 설명한 SaaS 아키텍처 주요 개념의 각 요소들을 SaaS 아키텍처 모범 사례를 준수하여 만든 다면 성공적인 SaaS 비즈니스 성장을 이끄는 SaaS 아키텍처가 될 수 있습니다. (각 요소별 모범사례는 아래 가이드에서 다루게 됩니다.)

SaaS Technical Skills Guide

SaaS Technical Skills Guide 는 SaaS 아키텍처 정의, SaaS 아키텍처 주요 개념 및 용어, 그리고 SaaS 아키텍처 구성 요소에 대한 상세한 내용을 담고 있는 AWS 가 제공 하는 SaaS 학습 자료 모음 입니다. 이 가이드는 SaaS 아키텍처 주요 구성 요소들에 대한 학습을 위해 선정된 자료로 각 구성 요소 별로 그 목적 의미, 구현 패턴, 그리고 실습을 포함합니다.

이 학습 자료들은 지속 적으로 업데이트, 추가 될 수 있으며, SaaS Factory Insight Hub 를 통해 보다 다양하고 최신의 SaaS 학습 자료를 찾아 볼 수 있습니다.

Copyright © Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Identity and Onboarding

Topic	Type
AWS 멀티 테넌트 SaaS 환경의 보안 규범	Blog
SaaS 와 OpenID Connect	Video
AWS SaaS Factory Architecture Track: SaaS Identity and Onboarding	Training
Amazon Cognito 를 활용한 SaaS Identity 관리	Blog
동적 생성 IAM 정책을 통한 SaaS 테넌트 격리	Blog
SaaS Identity	AWS Document
ABAC 과 IAM 을 활용한 SaaS 테넌트 격리 구현	Blog

Multi-tenancy

Topic	Type
SaaS 테넌트 격리 전략	White paper
SaaS 테넌트 격리 패턴	Video
AWS SaaS Factory Architecture Track: Tenant Isolation	Training
Deconstructing SaaS: Building Multi-Tenant Solutions on AWS	Video
SaaS 스토리지 전략	White paper
Architecting multi-tenant SaaS on EKS	Video
AWS SaaS Factory Architecture Track: Multi-Tenant Data Partitioning	Training

Metrics and analytics

Topic	Type
SaaS Metrics: The Ultimate View of Tenant Consumption	Video
AWS 기반의 SaaS 서비스의 멀티 테넌트 metric 수집과 시각화	Blog
Use AWS Application Cost Profiler to capture SaaS cost-per-tenant metrics	Video
AWS X-Ray 와 AWS Application Cost Profiler 를 활용한 테넌트 리소스 사용 비용 추적	Blog

Copyright © Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Billing and metering

Topic	Type
3rd party SaaS 미터링/미터링 솔루션을 활용한 구현	Blog
SaaS 환경에서의 테넌트 비용 계산	Blog
AWS Marketplace 의 SaaS 구독/계약 모델 활용한 출시	Blog

DevOps and Provisioning

Topic	Type
AWS Ramp-Up Guide: Developer(DevOps 포함)	White paper
AWS 프로토타이핑 엔지니어가 들려주는 AWS CDK 로 시작하는 DevOps	Video
Amazon EKS 를 활용한 멀티테넌트 PaaS 아키텍팅	Video

Management and monitoring

Topic	Type
AWS 멀티테넌트 SaaS 서비스 성능 효율화	Blog
SaaS 테넌트 모니터링	AWS Document
개별 테넌트의 자원 소비 현황 측정 방법	AWS Document
fairness in multitenant systems	Blog
Amazon EKS 를 활용한 테넌트 SaaS 애플리케이션을 위한 보안 규범	White paper

Hands-on(Reference architecture)

Topic	Type
Building SaaS Solutions on AWS	Hands-on
AWS Serverless SaaS Workshop	Hands-on
SaaS Factory EKS SaaS Workshop	Hands-on
AWS SaaS Factory Monolith to Serverless SaaS Workshop	Hands-on
AWS SaaS Boost	Reference architecture

Copyright © Amazon Web Services, Inc. and/or its affiliates. All rights reserved.