IIJ Internet Initiative Japan

IIJ Technical DAY 2018
クラウド基盤技術の変遷とオーケストレーションの進化

株式会社インターネットイニシアティブ
技術戦略室 田口 景介

Ongoing Innovation

このセッションでは

本セッションでは、この10年でパブリッククラウドに起きた基盤技術の進化と変遷についてお話しします。

クラウドの歴史とは、ストレージとネットワークの制御技術開発の歴史ですから、前半はそんな話が中心になります。

それから、タイトルにオーケストレーションの進化とありますが、実は進化が始まるのはこれからです。後半では、完成の域に近づいたIaaSが次にどこへ向かうのか、Serverless, Containerといった技術からお話しします。*

自己紹介

技術戦略室（通称CTO室）

Keisuke Taguchi （keisuk-t@iij.ad.jp）

アカウント名は最大8文字が当たり前だった時代の名残で間違えやすいです。

オンラインストレージサービス、メールゲートウェイサービス、クラウドサービス
などサービス開発を続けてきましたが、IaaSのプロジェクトを立ち上げてからは
半分インフラエンジニアになりつつあります。

C/C++, Javaと来ましたが、最近はgolang一辺倒。
IKEと名付けたコンテナプラットフォームを実装中。

サービスプロバイダ視点では、IaaSの10年はこのように見えています。

クラウド黎明期（2006～2008年頃）	**IaaSが何者なのか、まだわからなかった時代**
クラウド成長期（2008～2013年頃）	**IaaSに理解を求めた時代**
クラウド成熟期（2013～2018年頃）	**IaaSが当たり前になっていった時代**
クラウド完熟期（これから）	**ハイブリッドクラウドの時代**
	より高いレイヤーでのオーケストレーションへ

黎明期（2006～2008年ごろ）

- IaaSがニッチな割り切りサービスに見えていた
- AWSのEC2に永続的ストレージが存在しないことに驚いた*
- IIJがクラウドサービスの開発に着手した頃

成長期（2008～2013年ごろ）

- IaaSが急速に高度化
- PaaSに期待が集まっていた時代
- IIJがクラウドサービスGIOを2010年にリリース

IaaSの成長期

開発者を中心にPaaSが注目を集める

- IaaSよりも高いレイヤーでマネージドサービスを提供
- 運用をアウトソースして開発に専念できる環境を提供*

背景にあるのはIaaSへの懸念、不満

- 漠然としたセキュリティへの懸念
- ストレージやネットワークの性能やスケーラビリティの不足*

国内のIaaS市場はエンタメ系企業がけん引

- IaaSのアジリティとスケーラビリティが市場にマッチ
- エンタメ市場では切実にIaaSがもたらすメリットを必要としていた*

IIJ GIOをリリース

2010年11月にIIJ GIOをリリース

- サービス基盤用オーケストレータを拡張して開発

管理対象をベアメタルサーバからVMへ変更して、大規模な改修を加えたもの

- 仮想マシンはXenで管理
- ストレージはiSCSI接続
- ネットワークはVLANでテナントを分離

GIOの使われ方

VMは最小グレードと最大グレードが最も利用された

- ワークロードに見合った適切なグレードを選択するのが一般的
- 大規模ユーザーは最大グレードのVMを多数並べて、上手に利用*

ネットワークはサーバあたり1Gbps。これを数台のVMでシェア

ストレージはHDDが中心。数千IOPSを数台のVMでシェア

- Fusion-io社のioMemoryが大活躍。「エンジニアをダメにするデバイス」

エフェメラルストレージを提供しなかったことは、今でも後悔*

成熟期（2013～2018年ごろ）

- IaaSの成長に伴い、IaaSがクラウドの主役とみなされるように
- エンタープライズ市場においてもIaaSの利用が活発に
- 周辺サービスでクラウドベンダを選択する時代
- メガクラウドベンダがそろい踏み*

大幅な性能向上を果たしたIaaS

- ストレージはSSDが主流に
- ネットワークは仮想化され、帯域も10～40Gbpsへ向上
- I/O性能の不足に苦しんできたIaaSだが、逆にデバイスの能力を使い切れない状況が生まれる

続くストレージの技術開発

ストレージの性能向上

- HDD（数百IOPS）
- SSD（数万IOPS）
- NVMe SSD（数十万IOPS）

SSDの性能を生かし切れていない

- 仮想化レイヤーのオーバーヘッドが許容できないレベルに。IOPSが頭打ち
- 旧来のSANではストレージのトラフィックをまかなえない。スループットが頭打ち

その頃のIIJ GIO

第2世代IaaSであるIIJ P2を2015年にリリース

- ネットワークを仮想化、10Gbps化
- ストレージをSSDへ移行
- ハイパーバイザをkvmへ変更

P2＝Public and Private

- パブリッククラウドとプライベートクラウドのシームレスな連携を目指した
- 最初期はIP2（IIJ Public and Private）と呼ばれたが、早々に却下されP2に

スケーラビリティを大きく向上させ、ネットワーク構成の柔軟性を向上

- 運用負荷を大きく改善し、お客様のみならず弊社スタッフの満足度を向上
- 以前はサーバリソースに余裕があっても、VLANIDを先に使い切ってしまうこともあった*
- ゾーンの大規模化により在庫コントロールが容易に

P2のストレージ

PCI接続SSD, NVMe SSDの性能を引き出すには

- ベアメタルサーバが最適だが、IaaSから利用できる利便性も捨てがたい
- 現時点でリモートストレージは時期尚早
- ローカル接続、PCI Passthroughで仮想化層のオーバーヘッドを回避

ストレージの管理システムを独自に実装

- シンプルなハードウェア上にソフトウェアで付加価値を提供
- バックアップ・リストア、暗号化、クローニング、etc

P2のネットワーク

ネットワークをテナントごとに分離するためのアーキテクチャ

- 仮想L2ネットワーク
- PBR（policy based routing）
- BGP

オーバーレイかルーティングか

- 運用はシンプルだが、スイッチがボトルネックになりうるオーバーレイ方式
- パフォーマンスの懸念は少ないが、運用が複雑になりうるルーティング方式

最終的にVXLANベースの仮想L2ネットワークを選択

- 仮想スイッチがボトルネックになるのを避けるため二段階のスイッチを採用
- 目論見通り大半のトラフィックは同一ゾーンに閉じて、安定的にパフォーマンスを発揮
- 大規模な突発トラフィックには注意が必要

P2のハイパーバイザ

Xenからkvmへ

- ゲストOSにWindowsを利用する際のパフォーマンス向上
- 脆弱性対応の負荷軽減

完熟期（2018年～）

- セキュリティや性能に対する懸念は大幅に後退
- コンプライアンスや「自分で責任を持つ範囲」がクラウド利用可否の判断基準に

IaaSの次へ

- 開発や運用の効率化
- ハイブリッドクラウド、マルチクラウドの活用

オーケストレーション

IaaSを利用してもインフラ以外の運用は大きく変わらない

- 仮想化の有無はあれど、管理対象はサーバ、ストレージ、ネットワーク
- 根本的なところは変化がなく、ワークフローも同じ

オーケストレーションのレイヤーを上げて効率化

- Container
- Serverless

システム設計や運用をインフラから切り離す

- サイジングをシステムリソースではなく、ワークロードの規模で
- サーバ単位のサイジングから、コンテナ単位のサイジングへ

オーケストレーションのレイヤー

どのレイヤーでオーケストレーションするか

- Virtual Machine (IaaS)
- Container (CaaS, docker/Kubernetes)
- Function (FaaS, Serverless)

オーケストレーションのレイヤーが高いほど効率化は期待できるが、用途は限定される

- 高： Function > Container > VirtualMachine ：低

それぞれに適した領域があるので、目的を明確にする

- 開発、運用の効率化をしたいのか
- IaaSの特性を生かしてコストダウンを図りたいのか
- 必要なのは汎用的なプラットフォームか、特定用途の最適化か

ハイブリッドクラウド

クラウドファーストが浸透しても、100%クラウドを選択されるケースは極めてまれ

- 必然定期にハイブリッドクラウドへ向かう

ハイブリッドクラウドの形態は目的により様々

- IaaSとオンプレのリソースを連携させるのか
- オペレーションの統一が必要なのか
- ワークロードをIaaSとオンプレの間でマイグレートさせたいのか
- 複数のIaaSを使い分けたいのか

KubernetesとServerless

IaaSとオンプレを密接に連携させるのであれば、Kubernetesが有効

- コンテナ化がもたらすポータビリティ
- Kubernetesがインフラを抽象化し、オペレーションを統一

内製アプリケーションのランタイム環境ならばServerlessが有望

- PaaSの再来
- Serverlessのインターフェイスはプロダクト依存のためポータビリティには難あり
- イベントフォーマットに標準化の動きあり

IIJ Internet Initiative Japan

ご清聴ありがとうございました

Ongoing Innovation