

ベリタスウェビナー

Veritas Vibe

クラウド上のクラスタリングに対する パフォーマンスの不安はベリタスが解消

Veritas InfoScale on Clouds

2020年 8月 21日（金）

11:00



本日のプレゼンター

星野 隆義

クラウド推進&SDS担当部長

テクノロジーセールス本部

ベリタステクノロジーズ合同会社

VERITAS™

ベリタスウェビナー: Veritas Vibe へようこそ！



本日のセッションは
事前に収録



ご質問はチャット（当日のみ）
もしくはメールで可能



各種資料はサイト上の
リンクからアクセス可能



本日の内容

- 1 クラウド上でのクラスタリングに関する不安
- 2 切り替え時間の不安は心配無用
- 3 ベリタスによるパフォーマンステスト結果
- 4 パフォーマンス向上の為の方策



本日の内容

1 クラウド上でのクラスタリングに関する不安

2 切り替え時間の不安は心配無用

3 ベリタスによるパフォーマンステスト結果

4 パフォーマンス向上の為の方策

クラウド（AWS, Azure, GCP）上のクラスタリングは常識？

Google AWS Azure クラスタ

sios.jp > products > lkdk > ha-cluster > cloud ▾

クラウド環境での高可用性の仕組み | よくわかる！「HA ...

また、あらゆるクラウドに、完全自社開発したHA**クラスター**『LifeKeeper』を組み合わせること、クラウド上にHA**クラスター**環境を簡単に構築可能です。代表的なクラウド環境。Amazon EC2/**Microsoft Azure**/IBM Cloud など。その他の ...

bcblog.sios.jp > azure-multiaz-shogaitaisaku ▾

AzureでのマルチAZ構成による障害対策 | ビジネス継続とIT ...

多くのシステムが影響を受けた**AWS**の広域障害 に移動 - 8/23に発生した**AWS**の広域障害では、多くの影響が出ました。... さらに、障害の検知から切り替えまでを自動的に行うことのできるHA**クラスター**構成の活用が最も効果的である ...

qiita.com > AWS ▾

AWS/Azure/GCP における SQL Server の高可用性パターン ...

2020/08/05 - 各クラウドごとに以下のパターンを記載しています。マネージドサービスを利用する場合; VM で Always On Availability Groups を構成する場合; VM で Always On Failover Cluster Instances を構成する ...

qiita.com > AWS ▾

AWS/Azure/GCPサービス比較 2020.08 - Qiita

Hadoop**クラスター**の展開, Amazon EMR, HD Insight/**Azure** Databricks, CloudDataproc. Elasticsearch**クラスター**の展開, Amazon Elasticsearch Service, Elasticsearch Service on Elastic Cloud, Elastic Cloud on GCP. ストリーミング処理 ...

jpn.nec.com > clusterpro > clpx > sysrep_cloud ▾

クラウド環境: HAクラスタリングソフト CLUSTERPRO X ...

AWS; BIGLOBEクラウドホスティング; Enterprise Cloud; Google Cloud Platform; IaaS型クラウドサービス; IDCf ... 備考, **Microsoft Azure** において、異なる Fault Domain (※) に配置した Virtual Machine 同士でミラーディスク型**クラスター**を ...

Azure上でのマルチAZ構成による障害対策

皆さんこんにちは、 です。

本日は当社の取り組みとして、Azure上でのAvailability Zone（以下、AZ）をまたいだHAクラスター構成について一足早くお伝えいたします。Azureへ基幹系の移行を検討されている方はぜひご覧ください。



Contents [hide]

多くのシステムが影響を受けたAWSの広域障害

シングルAZとマルチAZの違いって？

Azure上でのAZをまたいだHAクラスター構成

1. の組み合わせ
2. の組み合わせ

ファイルサーバーをAZまたぎで冗長化してみた

関連情報

InfoScale on AWS ホワイトペーパー公開中

Qiita

ホーム

コミュニティ

🔍

📁 ストック一覧

📄 投稿する

0

VERITAS

VERITAS

@vxse_japan

2019年12月19日に更新

396 views

✎ 編集する

⚙️

AWS上の Windows で InfoScale を用いてクラスタリングしよう

AWSvcsクラスタVeritasInfoScale

はじめに

InfoScale は、AWS上のWindowsでのクラスタリングを保証しています。ただし、顧客要件によって実装パターンが複数存在します。そして、InfoScaleをAWS上のWindowsに構築する場合は、以下2つのポイントに留意する必要があります。

1. どのような要件を満たすために、どのような構成のクラスターを構築するか
2. オンプレとは異なるInfoScaleの前提条件

本記事では、上記2つのポイントを中心に、AWS上でクラスタリングが必要になった場合に、要件毎の最適なソリューションと、実装パターン毎の注意点を説明します。



分かりやすい説明 & 構築手順

■AWS上の Windows で InfoScale を用いてクラスタリングしよう

https://qiita.com/vxse_japan/items/bddaf20b07fb8bf302ef

【ホワイトペーパー】

https://www.veritas.com/content/support/en_US/doc/InfoScale7.4.1_Win_on_AWS_deploy_EBS_PrivateIP_JP

https://www.veritas.com/support/en_US/doc/InfoScale7.4.1_Win_on_AWS_deploy_VVR_OverlayIP

■AWS上の RHEL でInfoScaleを用いてクラスターを構築しよう

https://qiita.com/vxse_japan/items/49ddbdd381dca48ada9b

【ホワイトペーパー】

https://www.veritas.com/content/support/en_US/doc/InfoScale7.4.1_RHEL_on_AWS_deploy_EBS_PrivateIP_JP

https://www.veritas.com/content/support/en_US/doc/InfoScale7.4.1_RHEL_on_AWS_deploy_FSS_PrivateIP_JP

https://www.veritas.com/support/en_US/doc/InfoScale7.4.1_RHEL_on_AWS_deploy_FSS_OverlayIP

■AWS上のRHELでファイルシステムの拡張が必要になったらどうする？

https://qiita.com/vxse_japan/items/5e9beb7d937078f328a6

【ホワイトペーパー】

https://www.veritas.com/support/en_US/doc/InfoScale7.4.1_RHEL_on_AWS_FSS_storage_maintenance

■AWS上のWindowsでファイルシステムの拡張が必要になったらどうする？

https://qiita.com/vxse_japan/items/6374ded18d9b835e0030

【ホワイトペーパー】

https://www.veritas.com/support/en_US/doc/InfoScale7.4.1_Win_on_AWS_VVR_storage_maintenance

InfoScale on Azure ホワイトペーパー公開中



@vxse_japan 2020年04月22日に投稿 142 views

Azure上の Redhat でInfoScaleを用いてクラスターを構築しよう

Azure vcs クラスタ Veritas InfoScale

はじめに

InfoScale は、Azure上のRHELでのクラスタリングを保証しています。ただし、顧客要件によって実装パターンが複数存在します。そして、InfoScaleをAzure上のRHELに構築する場合は、以下2つのポイントに留意する必要があります。

1. どのような要件を満たすために、どのような構成のクラスターを構築するか
2. オンプレとは異なるInfoScaleの前提条件

本記事では、上記2つのポイントを中心に、Azure上でクラスタリングが必要になった場合に、要件毎の最適なソリューションと、実装上の注意点を説明します。



分かりやすい説明 & 構築手順

■ Azure上の Windows で InfoScale を用いてクラスタリングしよう

https://qiita.com/vxse_japan/items/180a8ee5e8d0da3b87a6

【ホワイトペーパー】

https://www.veritas.com/support/en_US/doc/InfoScale_7.4.2_Win_on_Azure_deploy_PrivateIP

https://www.veritas.com/support/en_US/doc/InfoScale_7.4.2_Win_on_Azure_deploy_VVR_OverlayIP

■ Azure上の RHEL でInfoScaleを用いてクラスターを構築しよう

https://qiita.com/vxse_japan/items/5d6b39d848ad79c7ffcc

【ホワイトペーパー】

https://www.veritas.com/support/en_US/doc/InfoScale_7.4.1_RHEL_on_Azure_deploy_PrivateIP

https://www.veritas.com/support/en_US/doc/InfoScale_7.4.1_RHEL_on_Azure_deploy_FSS_OverlayIP

https://www.veritas.com/support/en_US/doc/InfoScale_7.4.1_RHEL_on_Azure_deploy_VVR_OverlayIP

■ AWS上のRHELでファイルシステムの拡張が必要になったらどうする？

https://qiita.com/vxse_japan/items/445e9e68563a210e7a93

【ホワイトペーパー】

https://www.veritas.com/support/en_US/doc/InfoScale_7.4.1_RHEL_on_Azure_Storage_Maintenance

実際のお客様の声

～2020/6/26 セミナー他のアンケートより～



- ✓ 実装パターン毎の切り替え時間について言及が無かった
- ✓ 次回はパフォーマンスについて説明して欲しい
- ✓ AWS EBS の性能が、クラスタリング時にどの程度変わるのか示して欲しい
- ✓ レプリケーションと仮想ミラーのパフォーマンスの違いが知りたい
- ✓ 実装方法の議論は、性能面での現実性を示した後にすべきである

クラウド上のクラスタリング普及の為の2つハードル





本日の内容

1 クラウド上でのクラスタリングに関する不安

2 切り替え時間の不安は心配無用

3 ベリタスによるパフォーマンステスト結果

4 パフォーマンス向上の為の方策

クラウド上のクラスタリング

～オンプレと違って切り替え時間が長くないか不安です～



AWS

代表的な利用パターンで
2分以内の切り替えを実現

Azure

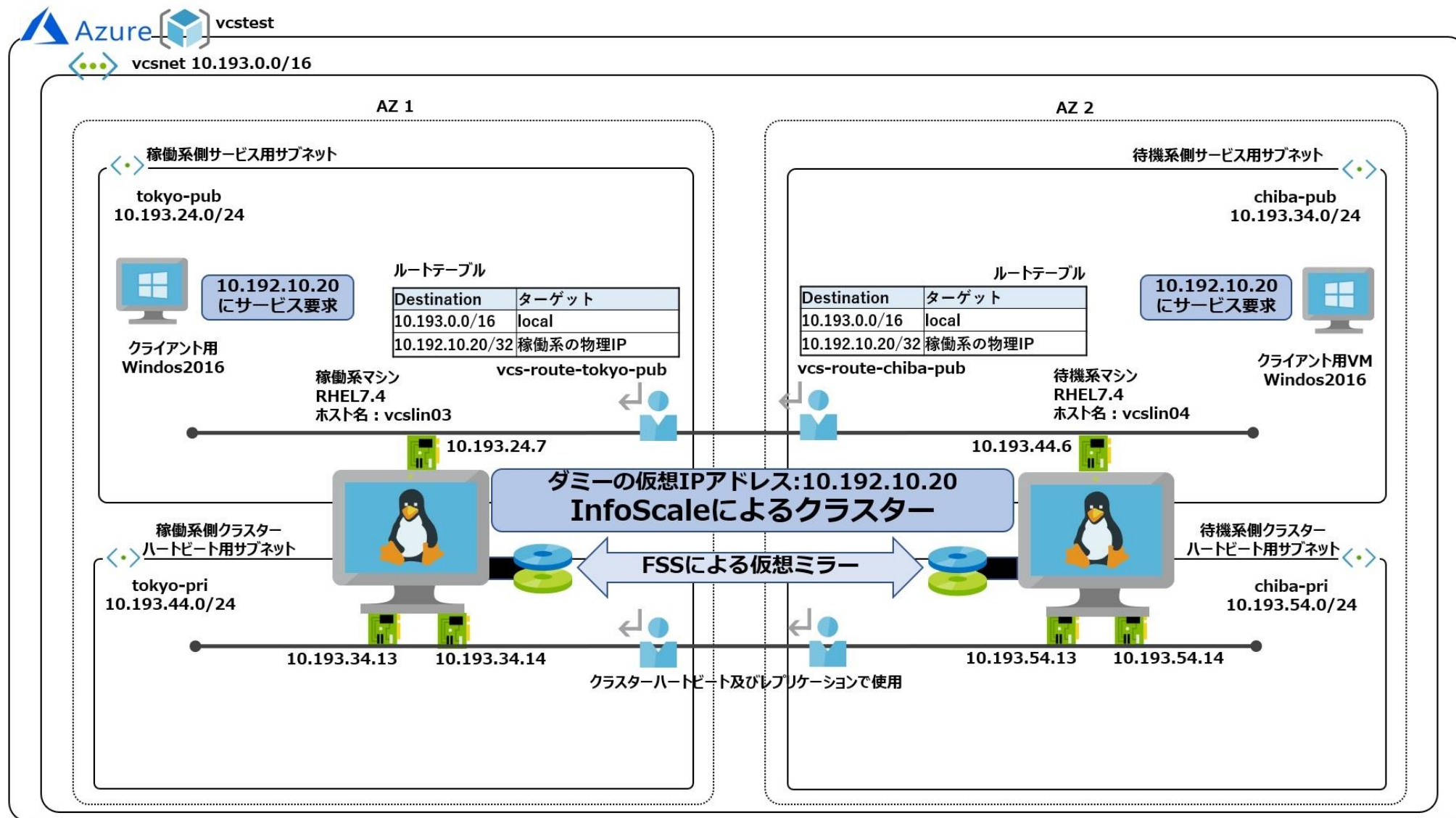
代表的な利用パターンで
2分以内の切り替えを実現

AWS上での切り替え時間検証結果

～RHEL上のapacheをクラスター監視下とし、httpdを強制終了～

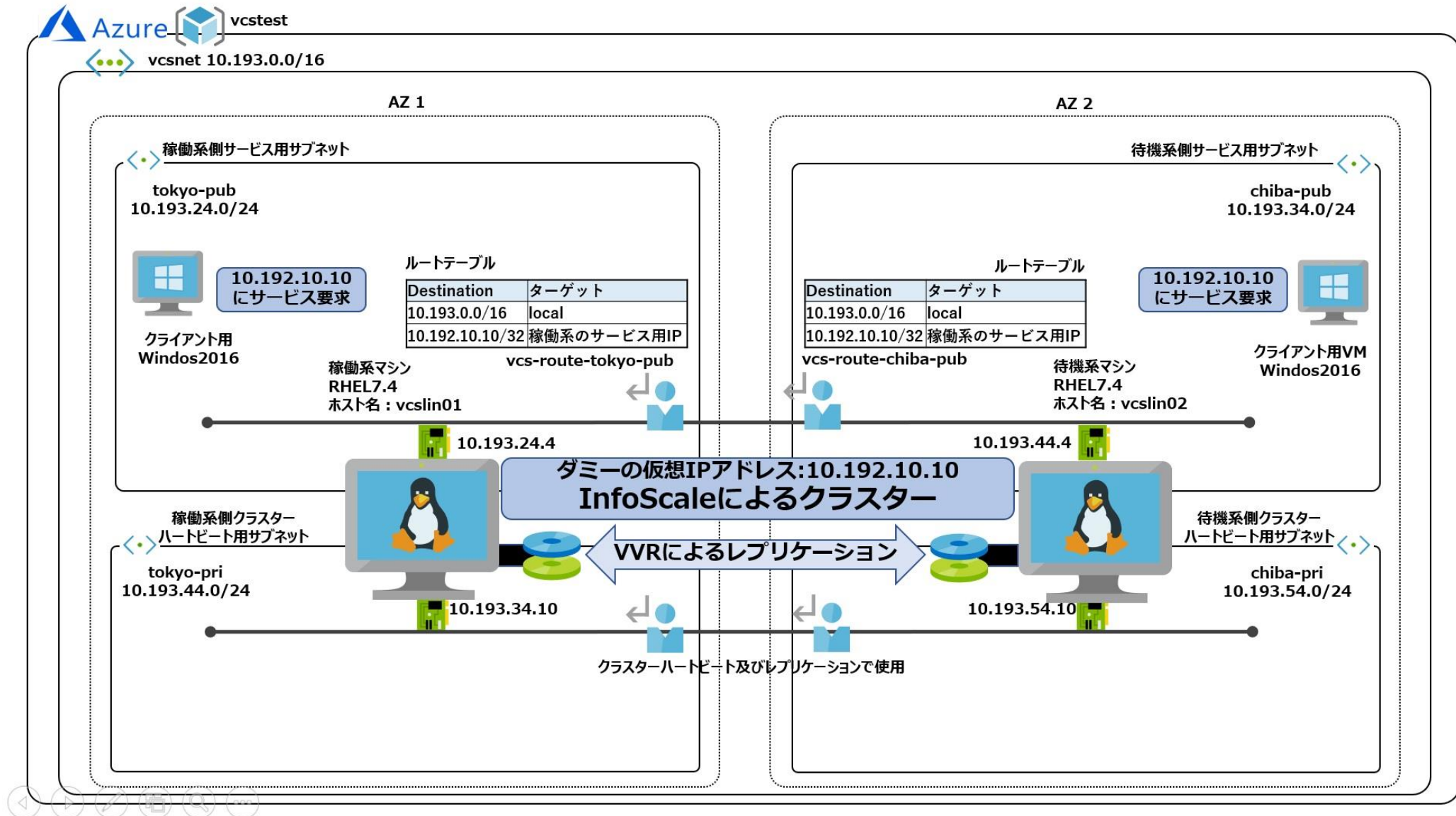
	PrivateIP切替型	ルートテーブル切替型	DNS切替型
共有ディスク型	AZ内クラスター: 1分12秒	AZ内クラスター: 1分16秒	AZ内クラスター: 1分38秒
仮想ミラー型	AZ内クラスター: 49秒	AZ内クラスター: 54秒	AZ内クラスター: 1分22秒
		AZ跨ぎクラスター: 56秒	AZ跨ぎクラスター: 1分28秒
レプリケーション型	AZ内クラスター: 1分4秒	AZ内クラスター: 1分11秒	AZ内クラスター: 1分29秒
		AZ跨ぎクラスター: 1分19秒	AZ跨ぎクラスター: 1分31秒

仮想ミラー+ルートテーブル切替によるAZ跨ぎクラスター構成



仮想ミラー＋ルートテーブル切替によるAZ跨ぎクラスター 切り替えデモ

レプリケーション+ルートテーブル切替によるAZ跨ぎクラスター構成



レプリケーション＋ルートテーブル切替によるAZ跨ぎクラスター 切り替えデモ

Azure上での切り替え時間検証結果

～RHEL上のapacheをクラスター監視下とし、httpdを強制終了～

	PrivateIP切替型	ルートテーブル切替型	DNS切替型
共有ディスク型	AZ内クラスター: 1分19秒	AZ内クラスター: 1分18秒	AZ内クラスター: 1分21秒
仮想ミラー型	AZ内クラスター: 58秒	AZ内クラスター: 54秒	AZ内クラスター: 1分18秒
		AZ跨ぎクラスター: 59秒	AZ跨ぎクラスター: 1分22秒
レプリケーション型	AZ内クラスター: 1分5秒	AZ内クラスター: 1分8秒	AZ内クラスター: 1分25秒
		AZ跨ぎクラスター: 1分15秒	AZ跨ぎクラスター: 1分28秒



本日の内容

1 クラウド上でのクラスタリングに関する不安

2 切り替え時間の不安は心配無用

3 ベリタスによるパフォーマンステスト結果

4 パフォーマンス向上の為の方策

クラウド上のクラスタリング

～クラスタリングによって、シングル構成に比べてパフォーマンスが落ちないか不安です～



AWS

共有ディスク構成なら
シングル構成と同等

Azure

代表的な構成パターンで
シングル構成と同等

AWS上でのパフォーマンステスト条件

1 マシン環境

- ✓ OS: RHEL7.7
- ✓ クラスタソフト : InfoScale7.4.2
- ✓ 仮想マシンは t2.large (2vCPU, 8Gbyteメモリ)
- ✓ データ用ディスクは汎用SSD (gp2) を用いて5Gbyteのファイルシステムを構築

2 構成パターン

以下の5つのパターンで比較

1. シングルノード構成(非クラスタリング)
2. AZ内の共有ディスククラスター
3. AZ内の仮想ミラークラスター (ハートビートはTCPを使用)
4. AZ内の仮想ミラークラスター (ハートビートはUDPを使用)
5. AZを跨いだ仮想ミラークラスター (ハートビートはUDPを使用)

3 パフォーマンス計測ツール

- ✓ フリーのツール : IOMETERを使用
https://ja.osdn.net/projects/sfnet_iometer/
- ✓ 様々なI/Oパターンで、それぞれ3分間のI/O負荷をかけ、IOPS、スループット、CPU使用率を計測

AZ: ap-northeast-1a

AZ: ap-northeast-1d

subnet-tokyo-pub
10.193.24.0/24



10.192.0.10
にサービス要求

クライアント兼
操作用EC2
Windos2016

nat-gateway-tokyo

ルートテーブル

Destination	ターゲット
10.193.0.0/16	local
0.0.0.0/0	vcs-gateway
10.192.0.10	クラスター稼働系NIC



nat-gateway-chiba

ルートテーブル

Destination	ターゲット
10.193.0.0/16	local
0.0.0.0/0	vcs-gateway
10.192.0.10	クラスター稼働系NIC

subnet-chiba-pub
10.193.34.0/24



10.192.0.10
にサービス要求

クライアント用EC2
Windos2016

subnet-tokyo-pri
10.193.44.0/24

稼働系マシン
RHEL7.7
ホスト名: vcslinlab09
物理IP: 10.193.44.109

ルートテーブル

Destination	ターゲット
10.193.0.0/16	local
0.0.0.0/0	nat-gateway-tokyo
10.192.0.10	クラスター稼働系NIC

待機系マシン
RHEL7.4
ホスト名: vcslinlab10
物理IP: 10.193.54.110

subnet-chiba-pri
10.193.54.0/24

ルートテーブル

Destination	ターゲット
10.193.0.0/16	local
0.0.0.0/0	nat-gateway-chiba
10.192.0.10	クラスター稼働系NIC

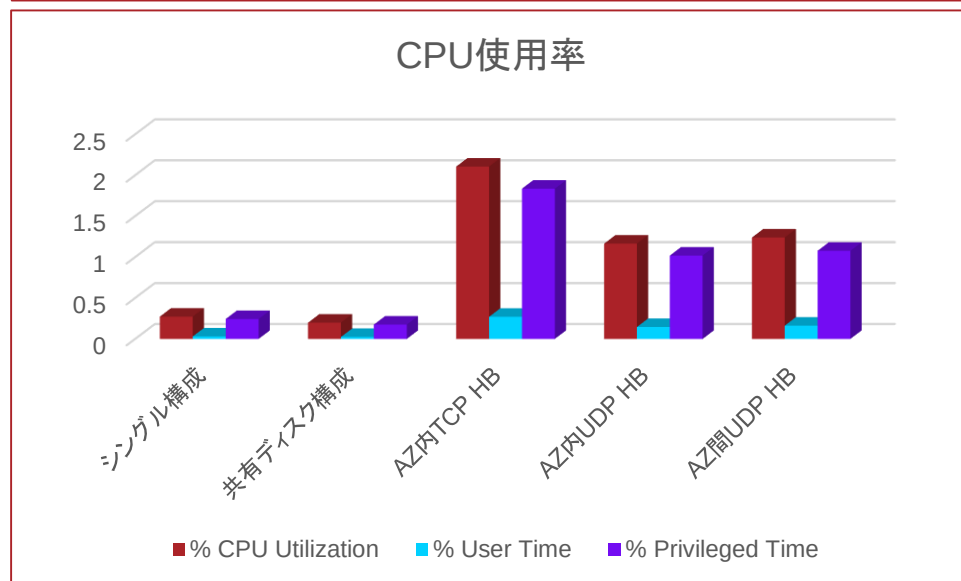
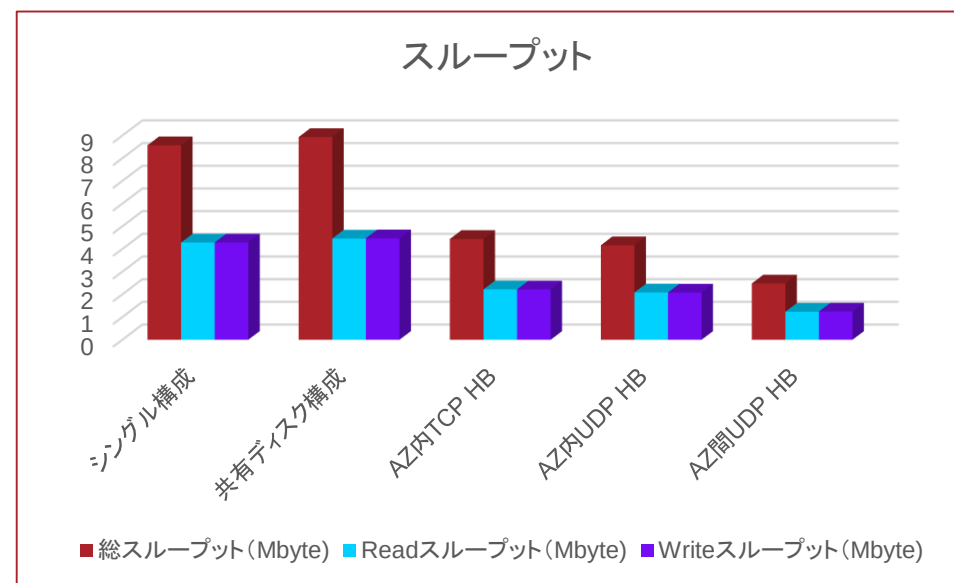
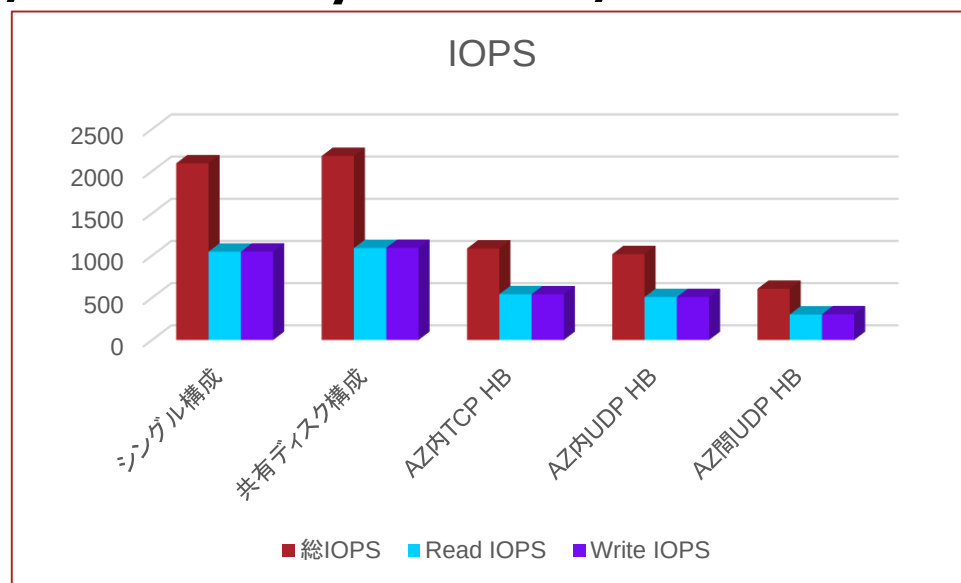
ダミーの仮想IPアドレス: 10.192.0.10
InfoScaleによるクラスター

FSSによる仮想ミラー

仮想ミラー＋ルートテーブル切替によるAZ跨ぎクラスター パフォーマンス計測デモ

AWS上でのパフォーマンステスト結果（チューニング無し）

～I/Oサイズ4Kbyte、Read/Write比率5対5、ランダムアクセスでテスト～



- ✓ 共有ディスク構成の場合、シングル（非クラスタリング構成）とパフォーマンスは変わらない
- ✓ 仮想ミラーを行った場合、チューニングしなければ3割程度にパフォーマンスが落ちる
- ✓ CPU使用率は2%程度上昇するのみ

Azure上でのパフォーマンステスト条件

1 マシン環境

- ✓ OS: RHEL7.7
- ✓ クラスタソフト : InfoScale7.4.2
- ✓ 仮想マシンは Standard A4_v2 (4Core, 8Gbyteメモリ)
- ✓ データ用ディスクはStandard SSDを用いて5Gbyteのファイルシステムを構築

2 構成パターン

以下の5つのパターンで比較

1. シングルノード構成(非クラスタリング)
2. AZ内の共有ディスククラスター
3. AZ内を跨いだレプリケーションクラスター
4. AZを跨いだ仮想ミラークラスター (ハートビートはUDPを使用、チューニング無し)
5. AZを跨いだ仮想ミラークラスター (ハートビートはUDPを使用、チューニングあり)

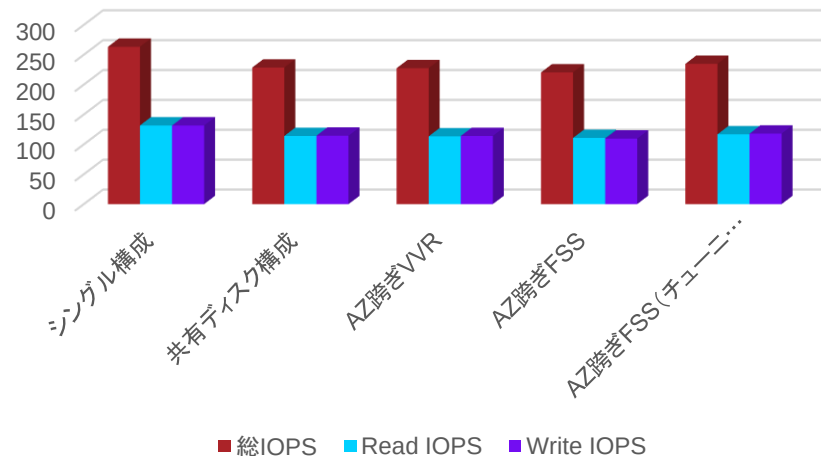
3 パフォーマンス計測ツール

- ✓ フリーのツール : IOMETERを使用
https://ja.osdn.net/projects/sfnet_iometer/
- ✓ 様々なI/Oパターンで、それぞれ3分間のI/O負荷をかけ、IOPS、スループット、CPU使用率を計測

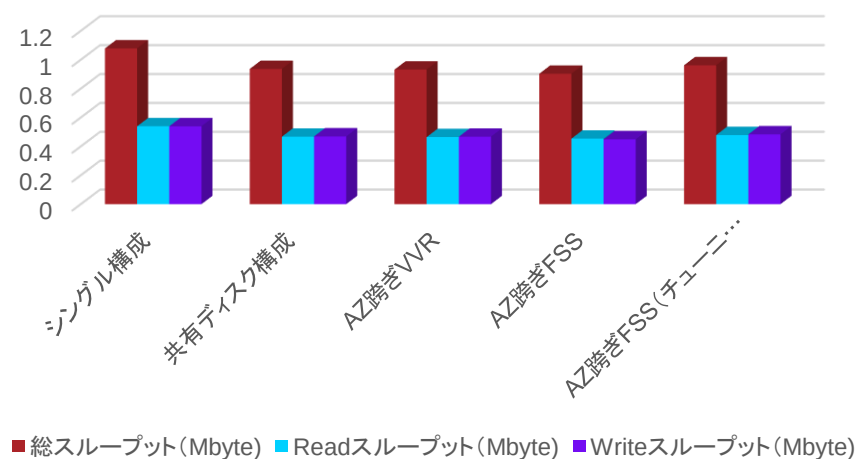
Azure上でのパフォーマンステスト結果

～I/Oサイズ4Kbyte、Read/Write比率5対5、ランダムアクセスでテスト～

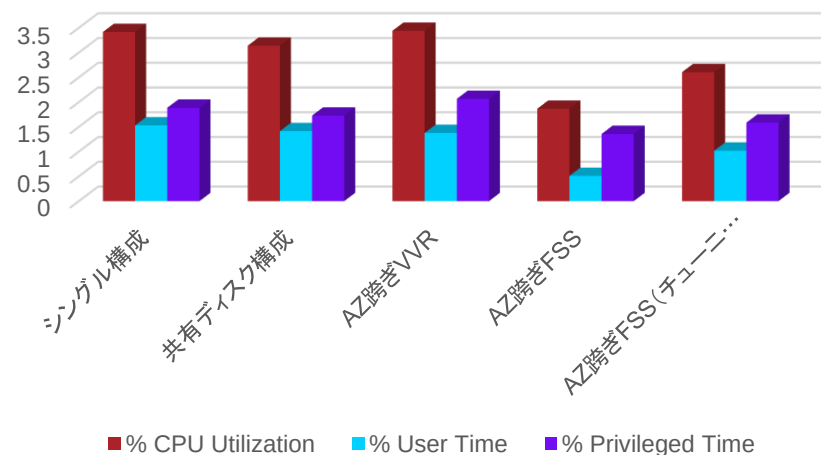
IOPS



スループット



CPU使用率



- ✓ Azureの場合、構成が小規模だとシングル構成でもパフォーマンスが出にくい
- ✓ 共有ディスク、仮想ミラー、レプリケーションの全てにおいて、シングル構成と同等のパフォーマンス
- ✓ CPU使用率も殆ど変わらない

AWS上の仮想ミラー構成ではチューニングの必要あり





本日の内容

1

クラウド上でのクラスタリングに関する不安

2

切り替え時間の不安は心配無用

3

ベリタスによるパフォーマンステスト結果

4

パフォーマンス向上の為の方策

クラスターのパフォーマンスを向上させる方策

構成：仮想ミラー

手法：
ハートビートネットワーク関連の
チューニングにより向上

- ✓ MTUサイズ拡張
- ✓ 通信バッファサイズ拡張
- ✓ LLTフロー制御閾値拡張

構成：レプリケーション

手法：
レプリケーション用カーネルバッ
ファのチューニングにより向上



<https://sort.veritas.com/DocPortal/pdf/79604030-141543652-1> の13章

MTUサイズ拡張による仮想ミラーパフォーマンス向上

具体的手法

ifconfig コマンドで、MTUサイズを変更する

```
# ifconfig eth1 mtu 9000
# ifconfig -a | grep mtu
eth1 . . . . . mtu 9000
```

向上する理由

データ転送の単位が大きくなるので、転送効率が良くなる。ただし、データの書き込みサイズが小さい場合は、逆に転送効率が悪くなるので注意

適したシーン

- バッチ系の処理
- シーケンシャルアクセス
- 書込みが多い
- 書き込みサイズが大きい、例えばファイルシステムのブロックサイズが8Kである

通信バッファサイズ拡張による仮想ミラーパフォーマンス向上

具体的手法

/etc/sysctl.conf の
XXXXXX 部分のパラメーター
を変更

```
net.core.rmem_default=XXXXXX
net.core.wmem_default=XXXXXX
net.core.rmem_max=XXXXXX
net.core.wmem_max=XXXXXX
net.core.optmem_max=XXXXXX
net.ipv4.tcp_rmem="4096 87380 XXXXXX"
net.ipv4.tcp_wmem="4096 65536 XXXXXX"
```

※ 今回は、メモリサイズと同じ8G
(8388608) に設定

向上する理由

通信バッファサイズを増やす
事で、転送効率があがる。た
だし、万能ではなく、搭載し
ているメモリとの兼ね合いも
あるので、注意

適したシーン

- バッチ系の処理
- クラスターのハートビート
にUDPを選択
- 書込みが多い
- 書き込みサイズが大きい、
例えばファイルシステムの
ブロックサイズが8Kである

LLTフロー制御閾値拡張による仮想ミラーパフォーマンス向上

具体的手法

lltconfig コマンドで、フロー制御の閾値を変更する

```
# lltconfig -F highwater:2000
# lltconfig -F rpothighwater:2000
# lltconfig -F query | grep highwater
highwater = 2000
rpothighwater = 2000
```

向上する理由

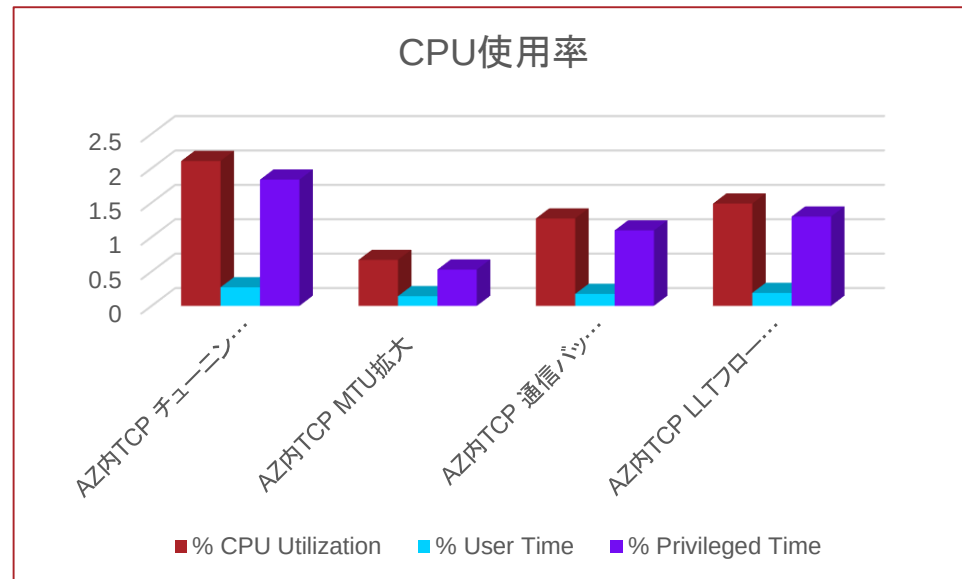
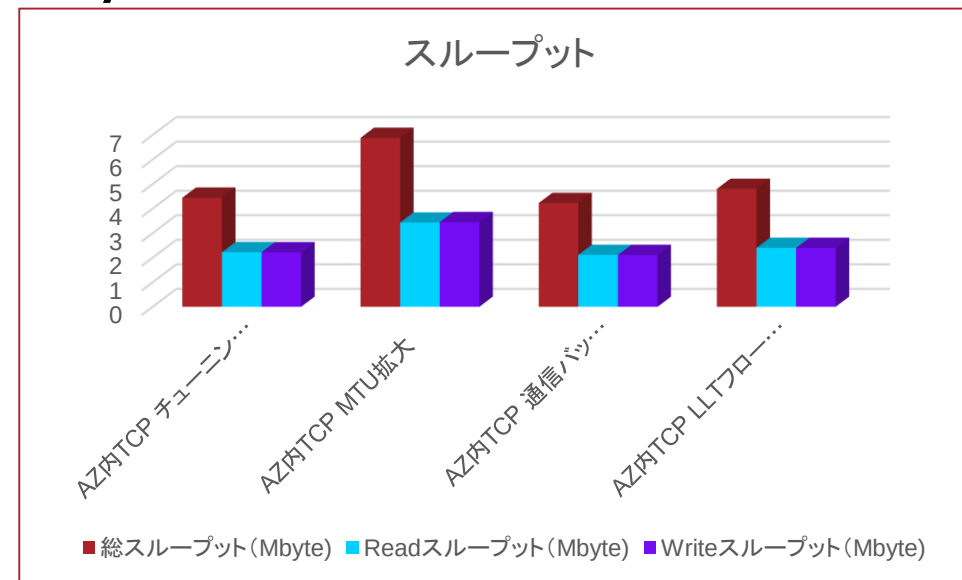
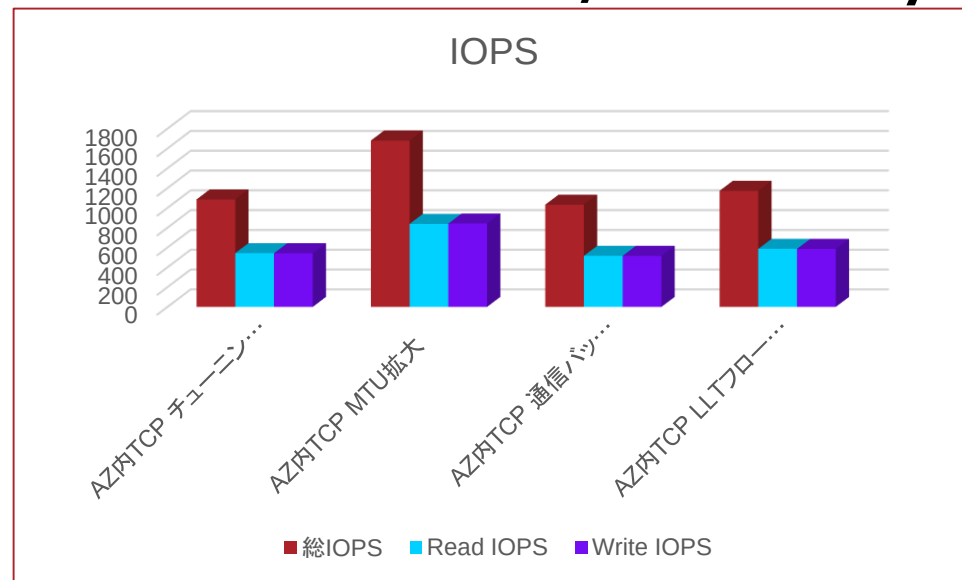
キューに溜められるデータ量が増えるので、転送効率があがる。ただし、万能ではなく、搭載しているメモリとの兼ね合いもあるので、注意

適したシーン

- AZ跨ぎなど、ネットワークのLatencyが大きい
- 書込みが多い

AWS上でのMTUサイズ拡張が有効に機能した例

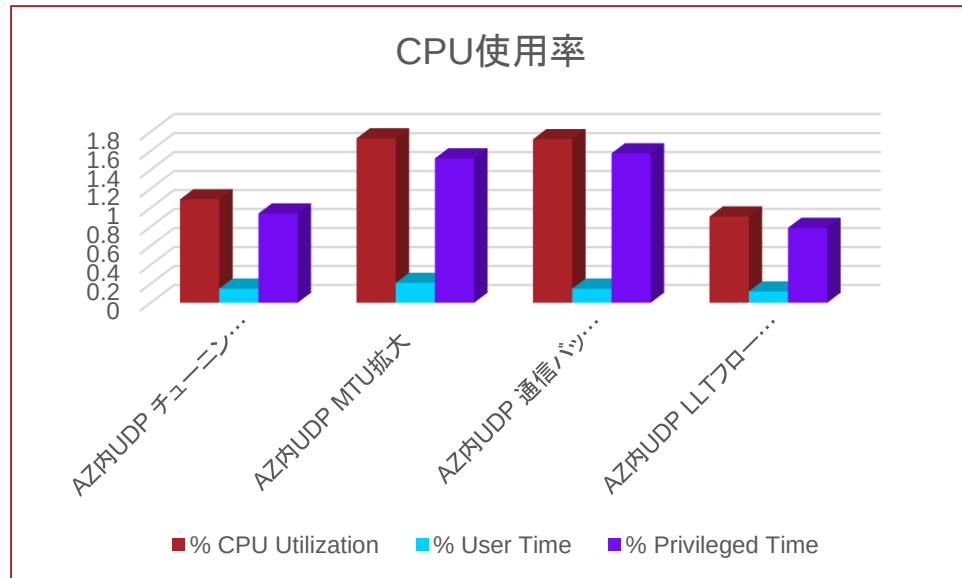
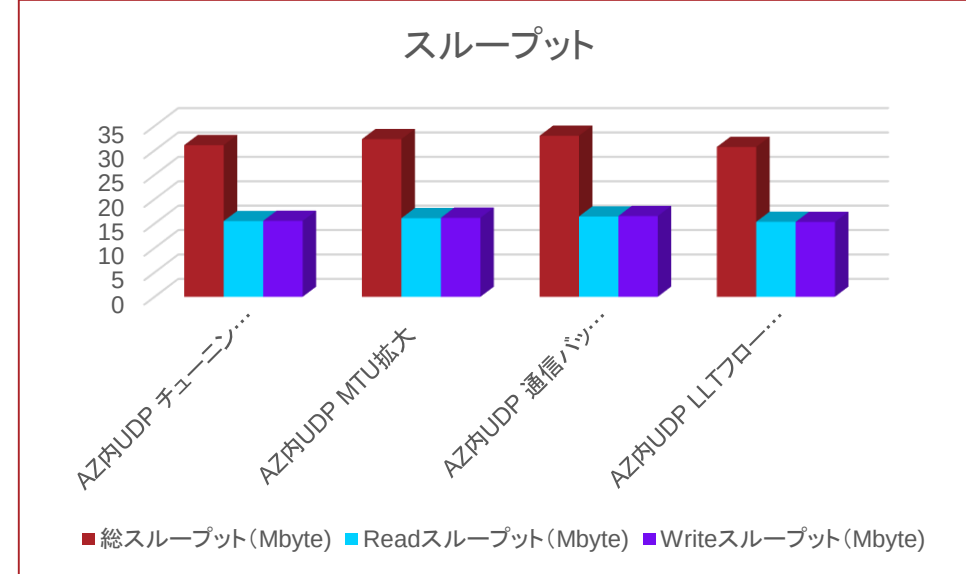
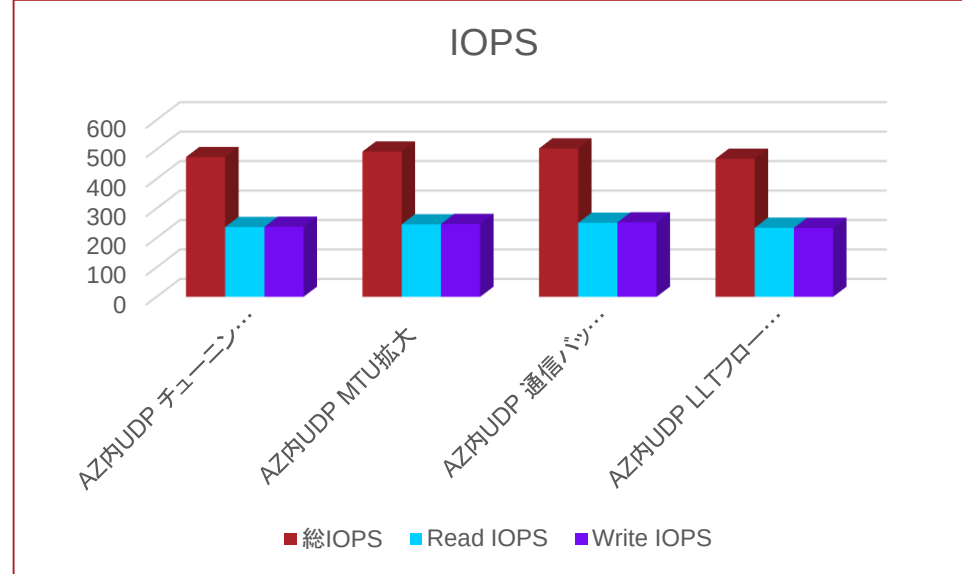
～AZ内TCPクラスタリング、I/Oサイズ4Kbyte、Read/Write比率5対5、シーケンシャルアクセス～



- ✓ チューニング無しに比べ、約1.8倍のパフォーマンス向上
- ✓ CPU使用率も低下

AWS上での通信バッファサイズ拡張が有効に機能した例

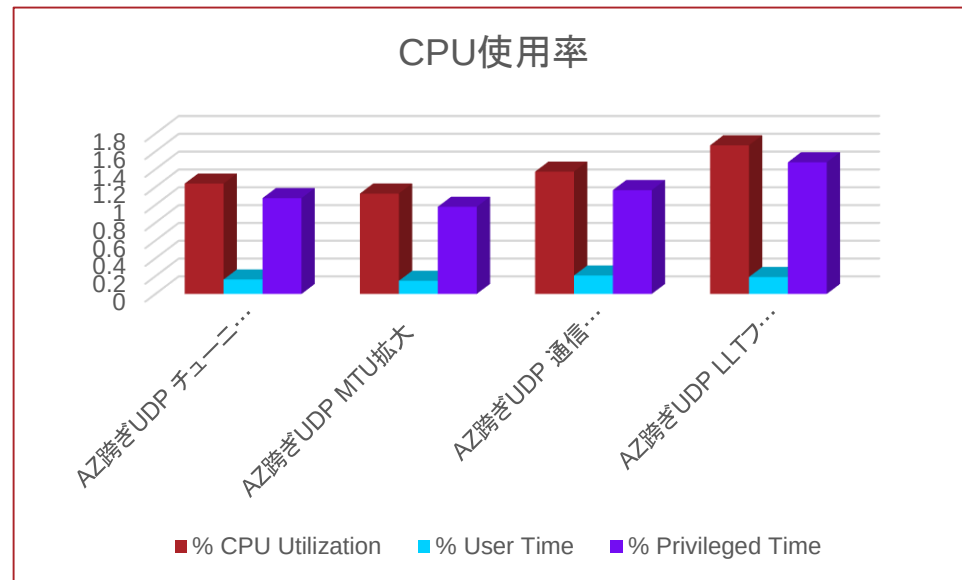
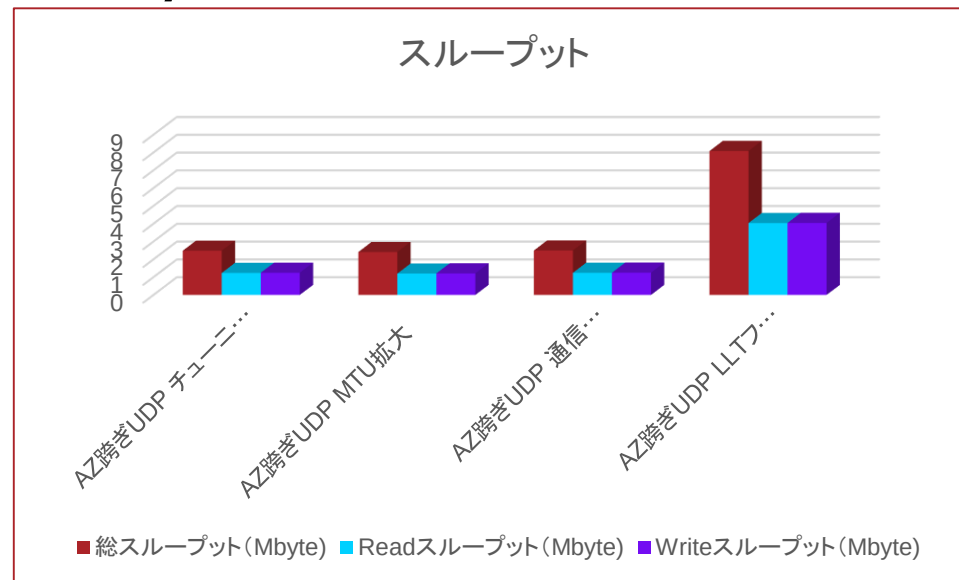
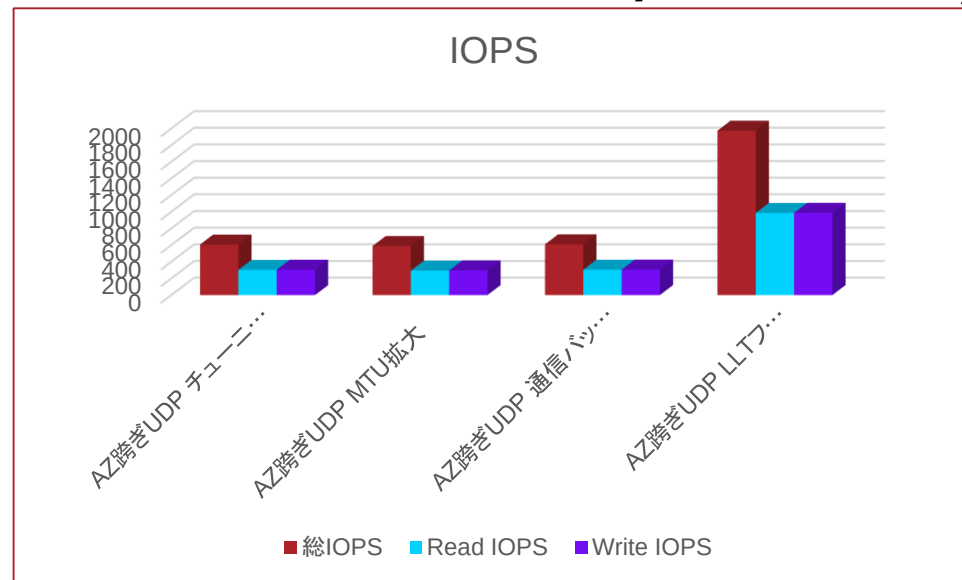
～AZ内UDPクラスタリング、I/Oサイズ64Kbyte、Read/Write比率5対5、シーケンシャルアクセス～



- ✓ チューニング無しに比べ、約1.2倍のパフォーマンス向上
- ✓ CPU使用率は上がったが1%未満

AWS上でのLLTフロー制御閾値拡張が有効に機能した例

～AZ跨ぎUDPクラスタリング、I/Oサイズ4Kbyte、Read/Write比率5対5、シーケンシャルアクセス～



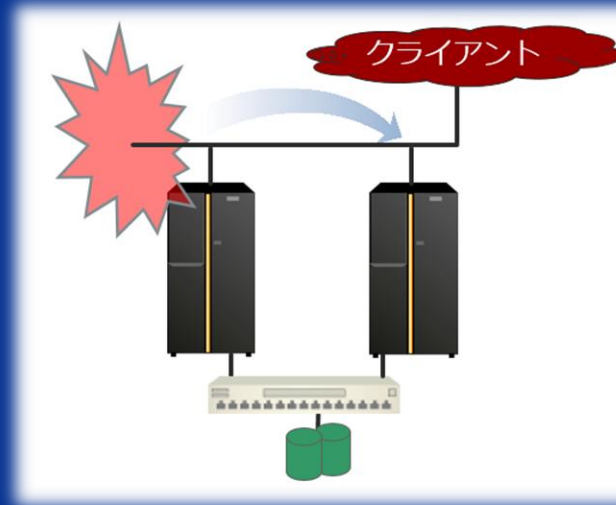
- ✓ チューニング無しに比べ、約3倍のパフォーマンス向上
- ✓ CPU使用率は上がったが1%未満

まとめ



基幹業務クラウド移行における課題

- 切り替え時間
- パフォーマンス

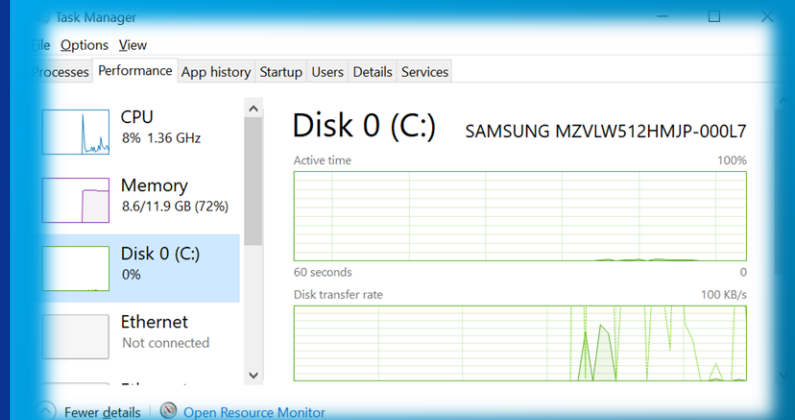


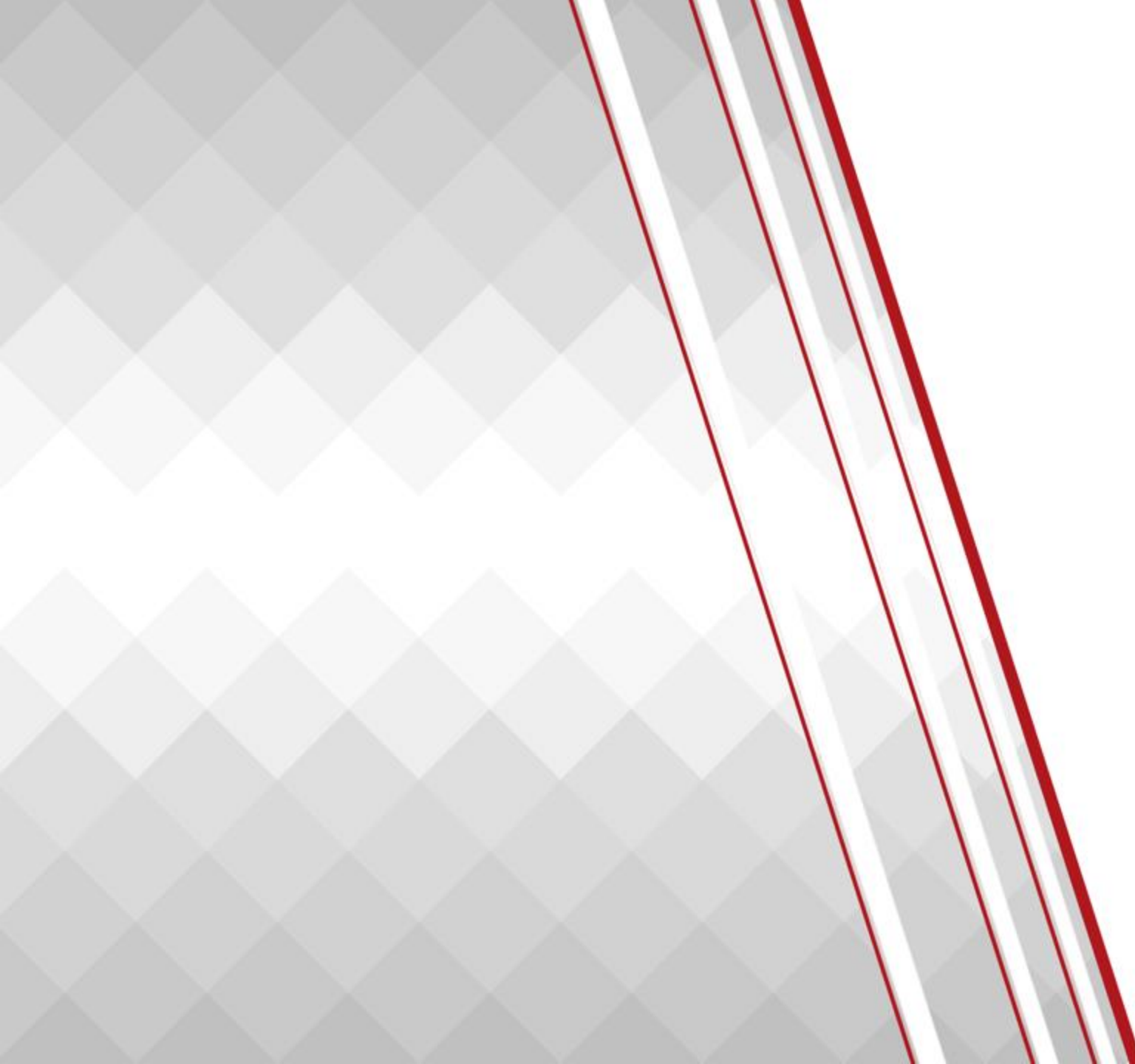
クラウド上のクラスタリング 切り替え時間は2分程度

- AWS、Azure、構成パターンに依存せず
- オンプレと変わらないSLAを実現可能

パフォーマンス

- Azureでは非クラスター構成と同等
- AWSの場合、チューニングが必要なパターンあり
- AWSのチューニングは、3つを使い分ける





Q & A

VERITAS™

ありがとうございました!

Copyright © 2020 Veritas Technologies, LLC. All rights reserved. Veritas and the Veritas Logo are trademarks or registered trademarks of Veritas Technologies or its affiliates in the U.S. and other countries. Other names may be trademarks of their respective owners.

This document is provided for informational purposes only and is not intended as advertising. All warranties relating to the information in this document, either express or implied, are disclaimed to the maximum extent allowed by law. The information in this document is subject to change without notice.