

JAPAN | 2024

aws SUMMIT



ゼンリンデータコム様の AMD インスタンス導入による効率化事例

中村 正澄

日本AMD株式会社

コマース営業本部
ソリューションアーキテクト

水尾 千寿

株式会社ゼンリンデータコム

技術本部 技術統括部
シニアエンジニア

ゼンリンデータコム様のAMDインスタンス 導入による効率化事例

AMDデータセンター向け最新テクノロジーのご紹介

第7世代 AMAZON EC2 インスタンス POWERED BY 第4世代 AMD EPYC™ “GENOA” クラウドプロセッサ

“M7a インスタンスは Amazon EC2
汎用シリーズの中で **最高の x86 パ
フォーマンスとコストパフォー
マンスを実現** するように設計されていま
す”¹

- AWS による発表 (2023年6月)

M7a は前世代のM6aと比較し
て約 **50% 性能向上** を提供
します。 (平均)¹

低コストで 生産性を向上

第4世代AMD EPYC 搭載
Amazon EC2 インスタンスで実現



クラウドワークロードの
加速とコストの削減を両立

より小規模かつ低コストで
ジョブを高速に実行

新たな投資をせずに
演算能力を強化

より多くのクラウドリソースを
一律料金で入手

クラウドアプリケーションとデータ
を保護

シリコンレベルでEPYC独自の
コンフィデンシャル
コンピューティングを実現

CPU の選択が重要

AMD EPYC "Genoa" 搭載プラットフォームのメリット

業界をリードする コスト パフォーマンス



コスト最適化オプションを提供

- より低コストで実行時間を短縮
- 同じコストで処理能力を向上
- 同じ性能でフットプリントを削減

クラス最高水準の電力効率



サステナビリティに最適化 されたホスティング

- 運用コストの削減
- GHG 排出量の削減
- 2.6 倍のワットあたり
パフォーマンス**

セキュリティの強化



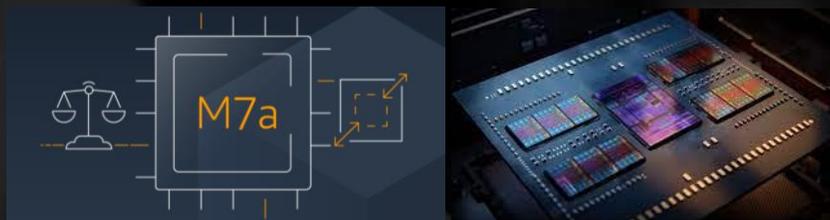
シリコン レベルで実現する コンフィデンシャル コンピューティング

- 使用中のデータを保護
- 独自 API は不使用
- プッシュボタン移行

最適なAMD EPYC搭載インスタンスは？

パフォーマンスと実行時間を短縮したいワークロード

第4世代 AMD EPYC “Genoa” CPU



パフォーマンス必須、ミッションクリティカルなアプリケーション

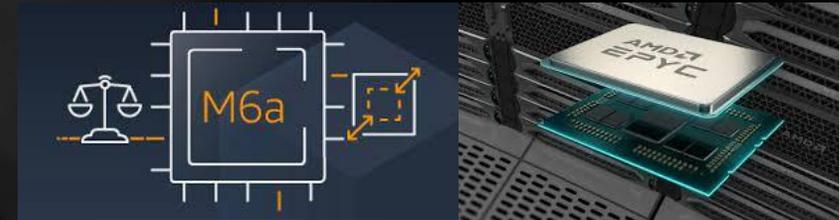
最大 **2.2x** 高性能

最大 **48%** EC2 コスト削減

vs. M7i instances powered by 4th Gen Intel Xeon CPUs
Redis / Bench set + get, 2xl + 4xl

常時稼働だがCPUをあまり必要としないワークロード

第3世代 AMD EPYC “Milan” CPU



For cost-sensitive, business-critical applications

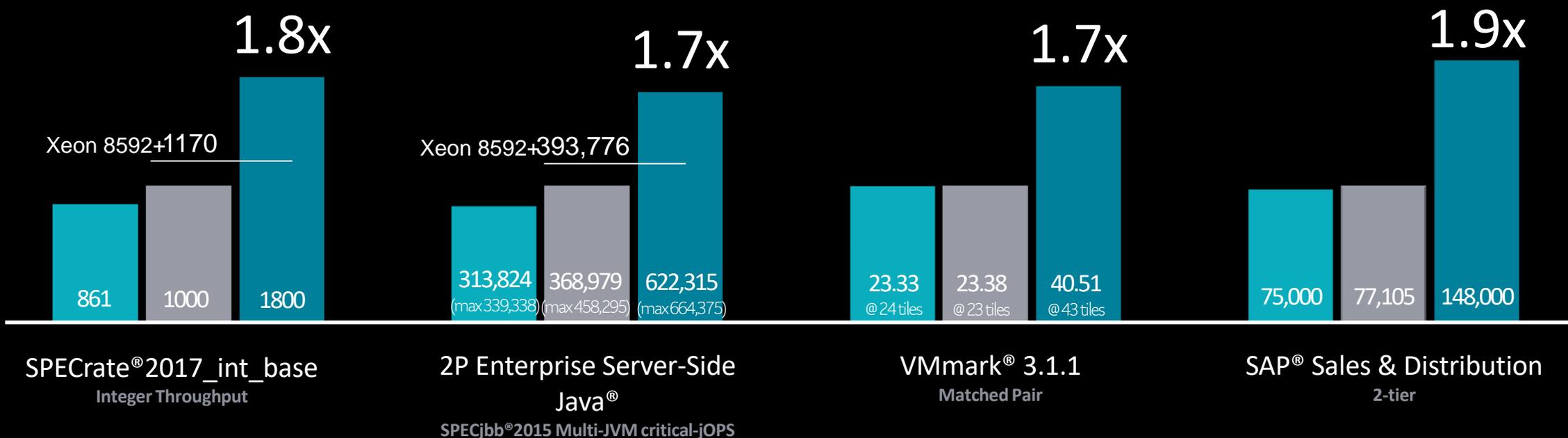
5% 高性能

14% EC2 コスト削減

vs. M6i instances powered by 3rd Gen Intel Xeon CPUs
SPECrate® 2017 Integer

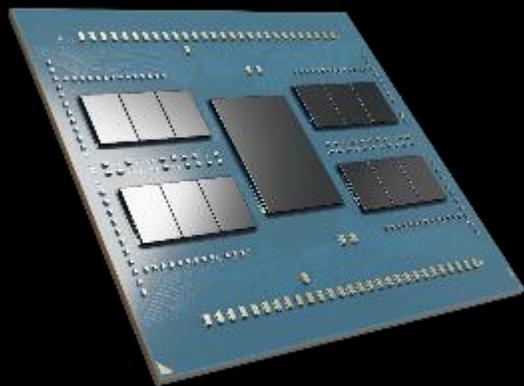
AMD EPYC

第4世代 EPYC™ CPU - “Genoa” パフォーマンスリーダーシップ



As of 6/13/2023, see [SP5-104A](#), [SP5-049C](#), [SP5-056B](#)

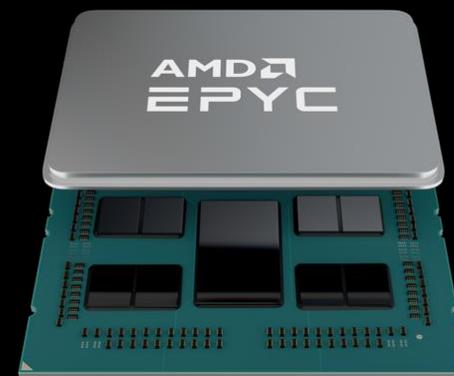
第4世代 AMD EPYC™ プロセッサ



Zen4
マイクロアーキテクチャー
5nm プロセステクノロジー
最大 96コア
12ch DDR5-4800MT/s

HPC、クラウド、エンタープライズの
ワークロードで、競合製品と
比較して優れた性能を実現

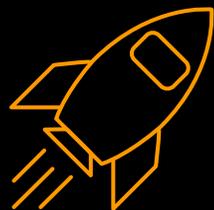
第3世代 AMD EPYC™ プロセッサ



Zen3
マイクロアーキテクチャー
7nm プロセステクノロジー
最大 64コア/128スレッド
8ch DDR4-3200MT/s

パワフルで
効率性に優れた
Zen 3コアを搭載

各種のワークロードに対応する AMD EPYC™ CPU 搭載 EC2 インスタンス



結果が出るまでの時間を短縮

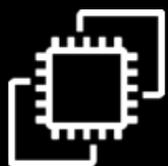


運用コストを抑制



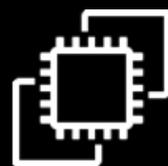
シームレスな移行

第 7 世代の EC2 インスタンス
AMD EPYC "Genoa" CPU を搭載



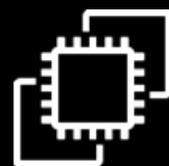
M7A

GENERAL
PURPOSE



R7A

MEMORY
OPTIMIZED



C7A

COMPUTE
OPTIMIZED



HPC7A

HPC
OPTIMIZED

最適なコストと最適なパフォーマンスを選択するには:

AMD EPYCを搭載したインスタンスへ移行することでジョブコストや時間当たりのコストを削減

小規模なデプロイメント

AMD搭載

第7世代 EC2 インスタンス
を選択

M7a / C7a / R7a / Hpc7a

高性能

小規模なデプロイメント

ジョブコスト削減 /
高スループットの実現

より早い実行時間

AMD搭載

第7世代 EC2 インスタンス
を選択

M7a / C7a / R7a / Hpc7a

高性能

高速な(短い)ランタイム

ジョブコスト削減 /
高スループットの実現

時間当たりコストの削減

AMD搭載

第6世代 EC2 インスタンス
を選択

M6a / C6a / R6a / Hpc6a

同等、またはそれ以上の性能

10% - 45% 低い時間当たりのコスト

M7a JAVA アプリケーション サーバー 性能 (4xlarge)

EPYCの卓越した性能により、同じワークロードでM7iよりも低コストで小規模な導入が可能

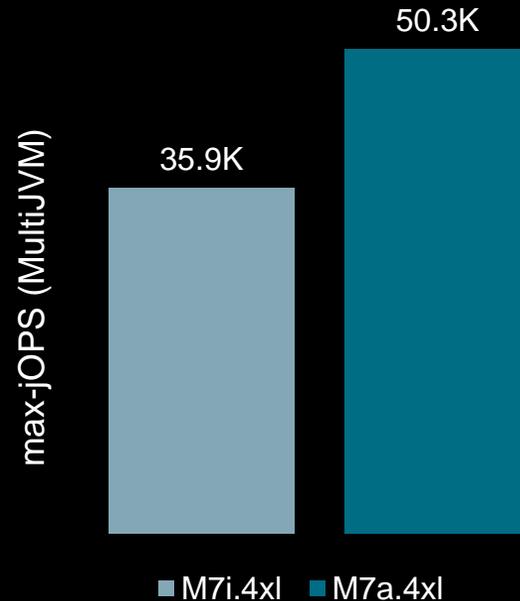
エンタープライズ
アプリケーション

Java

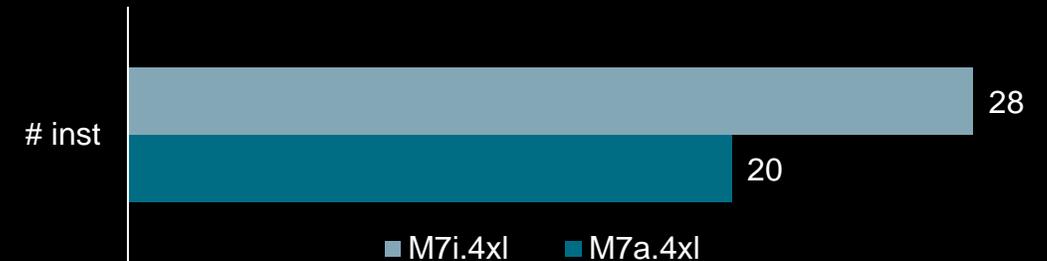
29%
少ないインスタンス
M7a vs M7i¹

\$1M の予算を1
\$180K削減

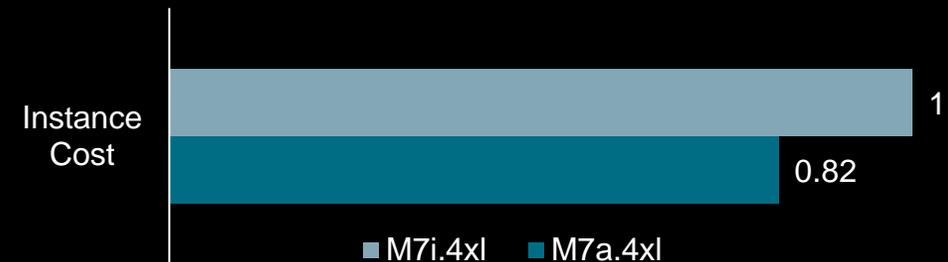
Java 2015 パフォーマンス



1M max Java opsを実現するために必要な インスタンス数



1M max Java opsを実現するために必要な コスト



NGINX Web サーバー M7a

EPYCの優れた性能によりM7aインスタンスがM7iよりも小規模で低コストで同じジョブを実行可能です

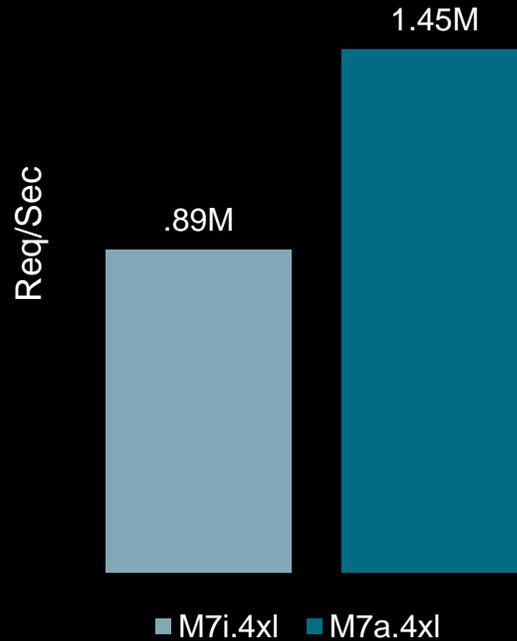
Web サービス

NGINX

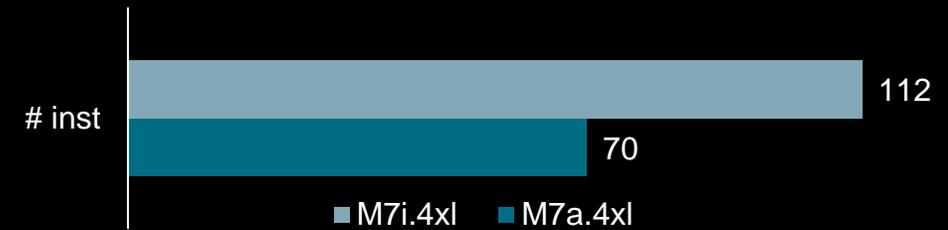
38%
少インスタンス
M7a vs M7i

Save \$280K
per \$1M spend

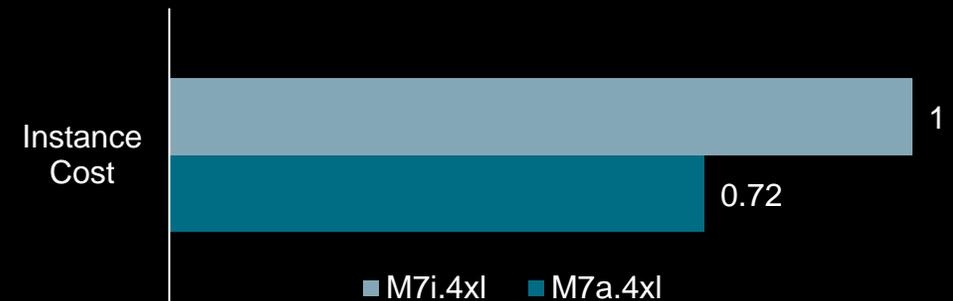
NGINX パフォーマンス



100M Req/Sec を提供するために必要な インスタンス数



100M Req/Sec を提供するために必要なコスト

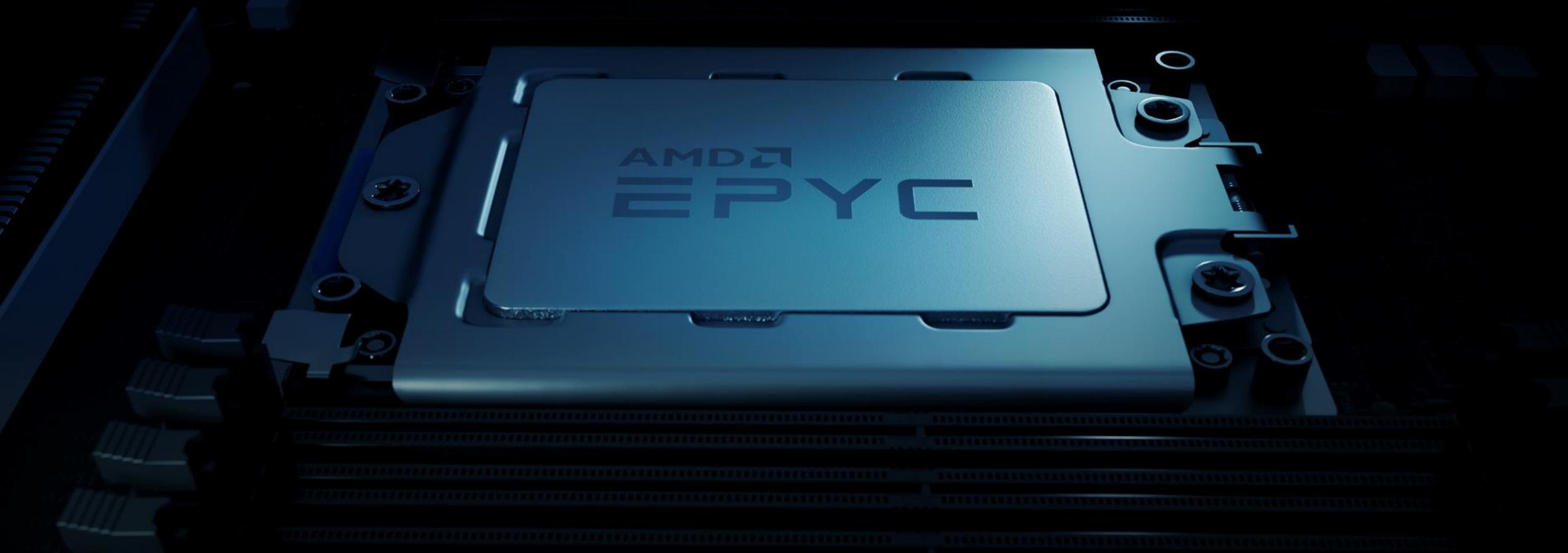


EPYCの性能でクラウドのフットプリントとコストを削減

EPYCの高い性能によりよりM7iよりも小さなデプロイが可能に、これによりコストとCO2削減に貢献します

Web/App			データベース		
エンタープライズアプリケーション	Web サービス	ビデオプロセッシング	SQL データベース (Transactional)	SQL データベース (Analytics)	NoSQL データベース
Java	NGINX	FFMPEG	MySQL	MS SQL Server	Redis
29% インスタンス数削減 M7a vs M7i ¹	38% インスタンス数削減 M7a vs M7i ²	47% インスタンス数削減 M7a vs M7i ³	29% インスタンス数削減 M7a vs M7i ⁴	23% インスタンス数削減 on M7a vs M7i ⁵	55% インスタンス数削減 M7a vs M7i ⁶
Save 18% M7a vs M7i ¹	Save 28% M7a vs M7i ²	Save 38% M7a vs M7i ³	Save 19% M7a vs M7i ⁴	Save 12% M7a vs M7i ⁵	Save 48% M7a vs M7i ⁶

第4世代Intel Xeon搭載のM7i インスタントと比較してM7aでは
平均 **37%** 小さなフットプリントで **27%** インスタンスコストを削減 しジョブを実行可能です。 ¹⁻⁶



THANK YOU

AMDインスタンス導入による効率化のご紹介

2024.6.21

株式会社ゼンリンデータコム

技術本部 技術統括部

水尾 千寿

目次

- 自己紹介 / 会社概要 …02
- ZDCにおけるAWS活用術 …06
 - ✓ ZDCのインフラ沿革
 - ✓ AWS費用を抑制するための取り組み
- AMDインスタンスを導入する際の考慮点 …10
 - ① 手軽さ：簡単に導入できるか
 - ② 金額面：利用料金は妥当か
 - ③ 性能面：処理能力は問題ないか
- まとめ …23

自己紹介

- 名前：水尾 千寿（みずお ちとし）
- 所属：株式会社 ゼンリンデータコム
技術本部 技術統括部 シニアエンジニア
- AWS利用歴：
東京リージョンが利用可能になった
2011～2012年頃から利用中
- 好きなAWSサービス：Amazon Athena



会社概要

会社名	株式会社ゼンリンデータコム (ZENRIN DataCom CO., LTD.)
設立	2000年4月13日
本社	東京都港区芝浦三丁目1番1号 msb Tamachi 田町ステーションタワー N22階
代表取締役社長	清水 辰彦
資本金	2,283,010千円
従業員数	417名 (2024年4月1日現在)
事業所	本社オフィス、名古屋オフィス 大阪オフィス、福岡オフィス
株主	株式会社ゼンリン
関連会社	株式会社リースシステム企画 株式会社アクトキューブ 株式会社コミュニケーション・プロジェクト

<https://www.zenrin-datacom.net/>

展開する事業

コンシューマ事業

コンシューマー向けに
さまざまなモバイル地図サービスを提供



パートナー事業

各パートナー企業との協業を通じて
各分野にあったソリューションを提供

IoT事業

IoT分野をはじめとした多種にわたる業界の法人向けに
ビジネスで役立つ地図ソリューションを提供

オートモーティブ事業

人と道路と自動車サービス向けに高い技術力で
ナビゲーションサービス、テレマティクスサービスを提供

ビジネス概要 [1/2]

世界でも類を見ない独自の調査と徹底した情報管理で生成される高品質なゼンリン地図データを活用

ZENRIN

調査スタッフと専用車両による現地調査



データ入力と整備された地図データベース

高品質地図データ



住宅地図／過去地図

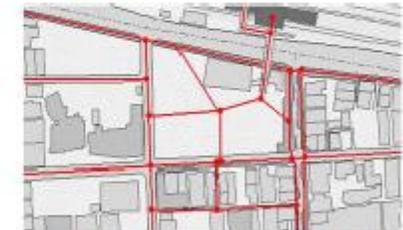


3D地図

高精度ネットワークデータ



車ネットワークデータ



歩行者ネットワークデータ

豊富な検索データ

住所検索データ

約3,400万件

郵便番号検索データ

約14万件

施設情報 (POI)

約1,000万件

ビジネス概要 [2/2]

ゼンリンが保有する地図・ナビゲーション関連データと各種コンテンツデータを目的に応じて独自性の高いテクノロジーで構築し魅力あるソリューションやサービスを提供



ロジスティクス

物流のデジタルトランスフォーメーションに貢献



位置情報ビッグデータ解析

人の流れや動向を把握して社会の好循環に貢献



ビジネス拠点管理

BIツールとしてお客様のビジネスを多角的に支援



API/SDK

お客様のサービスやシステム開発を支援するAPI/SDKを提供

ZENRIN DataCom



インドアロケーション

物流倉庫/商業施設等の屋内業務でスマート化と生産性向上を支援



ナビゲーション

通信環境やOSに依存せずニーズに応じた高機能アプリ



ドローン

豊富な実績でドローンビジネスの可能性を広げる



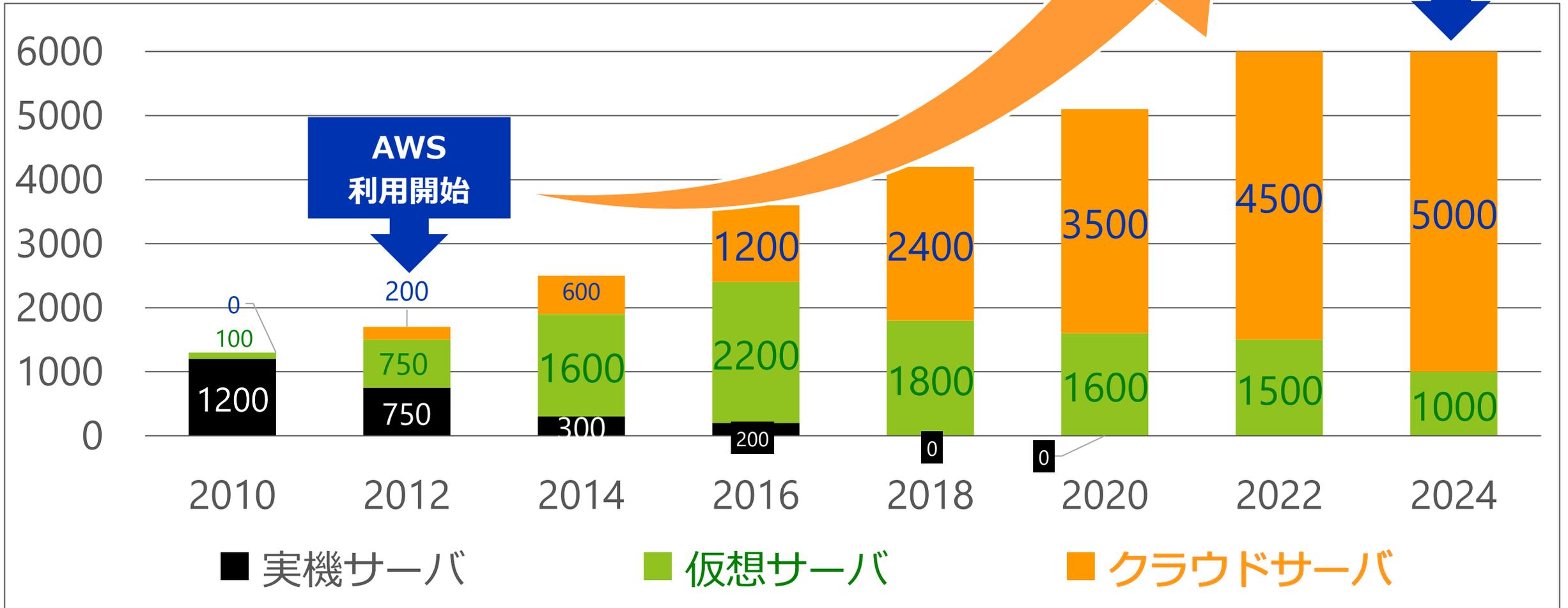
運用監視

サービス品質の維持/安定化と運用コスト最適化

ZDCにおける AWS活用術

ZDCのインフラ沿革

● ZDCのサーバ台数変遷



AWS費用を抑制するための取り組み

STEP1

済

利用しない→停止する

- ✓主に開発環境/検証環境
- ✓AWS Instance Schedulerを利用
夜間土日の自動停止を実現

50% DOWN

STEP2

済

長期契約による割引

- ✓主に本番環境 (24/365稼働)
- ✓SavingsPlans 活用 (高Coverage率の維持)

1～3年の長期利用をする場合に
大幅な割引が受けられる仕組み

50% DOWN

STEP3

サーバリソース最適化

✓以下の施策を実施中

- ①ダウンスライジング (AIレコメンド機能活用)
- ②アーキテクチャ改善 (コンテナ/サーバレス)

③AMDインスタンスの導入



AMDインスタンスを 使ってみよう

AMDインスタンスを導入する際の考慮点



- 以下の3点になります 

1

手軽さ：簡単に導入できるか

2

金額面：利用料金は妥当か

3

性能面：処理能力は問題ないか

AMDインスタンスを導入する際の考慮点①

1 手軽さ：簡単に導入できるか

- ▶ AMDインスタンスはx86互換



例：m系Large の x86系インスタンス

	<第5世代>	<第6世代>	<第7世代>
AMD	m5a.large	m6a.large	m7a.large
他社	m5.large	m6i.large	m7i.large

インスタンスタイプを変更するだけで簡単に試してみることが出来る



x86互換のインスタンスであればプログラムの改修作業は必要ない

AMDインスタンスを導入する際の考慮点②

2 金額面：利用料金が妥当か



EC2インスタンス料金（例：c系Large の x86系インスタンス）

料金の値下げが頻繁にあるため
最新情報は必ず確認して下さい

インスタンスタイプ		オンデマンド			SavingPlans		
		第5世代	第6世代	第7世代	第5世代	第6世代	第7世代
c系Large	AMD	\$0.0960	\$0.0963	\$0.12920	\$0.0520	\$0.0449	\$0.06103
	他社	\$0.1070	\$0.1070	\$0.11235	\$0.0570	\$0.05055	\$0.05307

※Compute Savings Plans(3年全額前払/東京リージョン)

x86互換のEC2インスタンスでは…

SavingPlansを活用した場合

- ➔ 利用料金の **最安値**：第6世代のAMDインスタンス
- ➔ 利用料金の **最高額**：第7世代のAMDインスタンス

ユースケースに応じて
使い分けが必要になる

AMDインスタンスを導入する際の考慮点③

3 性能面：処理能力に問題がないか



▶ 以下のテストを行い性能比較を実施しました

i. ベンチマークテスト

ii. 負荷テスト（限界値）

パターン	対象サーバ	ミドルウェア	比較内容
①：限界値計測	Engineサーバ	Apache等	第6世代： AMD × 他社 第7世代： AMD × 他社

iii. 性能テスト（パフォーマンス）

パターン	対象サーバ	ミドルウェア	比較内容
②：レスポンスタイム計測	APIサーバ	Nginx等	AMD ： 第6世代 × 第7世代
③：バッチ処理時間計測	Batchサーバ	PostgreSQL等	

i . ベンチマークテスト : 試験内容

● 前提条件

- ▶ ベンチマークツール : UnixBench 🔍
 - ✓ 「並列度 2」の 各テスト項目の 総合スコア を算出
 - ✓ 評価に用いたベンチマークのテスト項目は下記を参照

テスト項目	内容
dhry2reg	整数演算処理のテスト
e-double	浮動小数点処理のテスト
execl	システムコールを繰り返すテスト
fstime	ファイルコピーのテスト(2MBのファイルを1024Bごとに処理)
fsbuffer	ファイルコピーのテスト(500KBのファイルを256Bごとに処理)
fsdisk	ファイルコピーのテスト(7.9MBのファイルを4096Bごとに処理)
pipe1	パイプ処理のテスト(512Bのデータをパイプ処理で繰り返し実行)
pipe2	パイプ処理のテスト(プロセスのコンテキストスイッチを実行)
spawn	プロセスのフォークを繰り返すテスト
shell1	テキスト処理を繰り返すテスト
shell8	shell1と同じ内容を8並列で繰り返すテスト
syscall	システムコールを繰り返すテスト

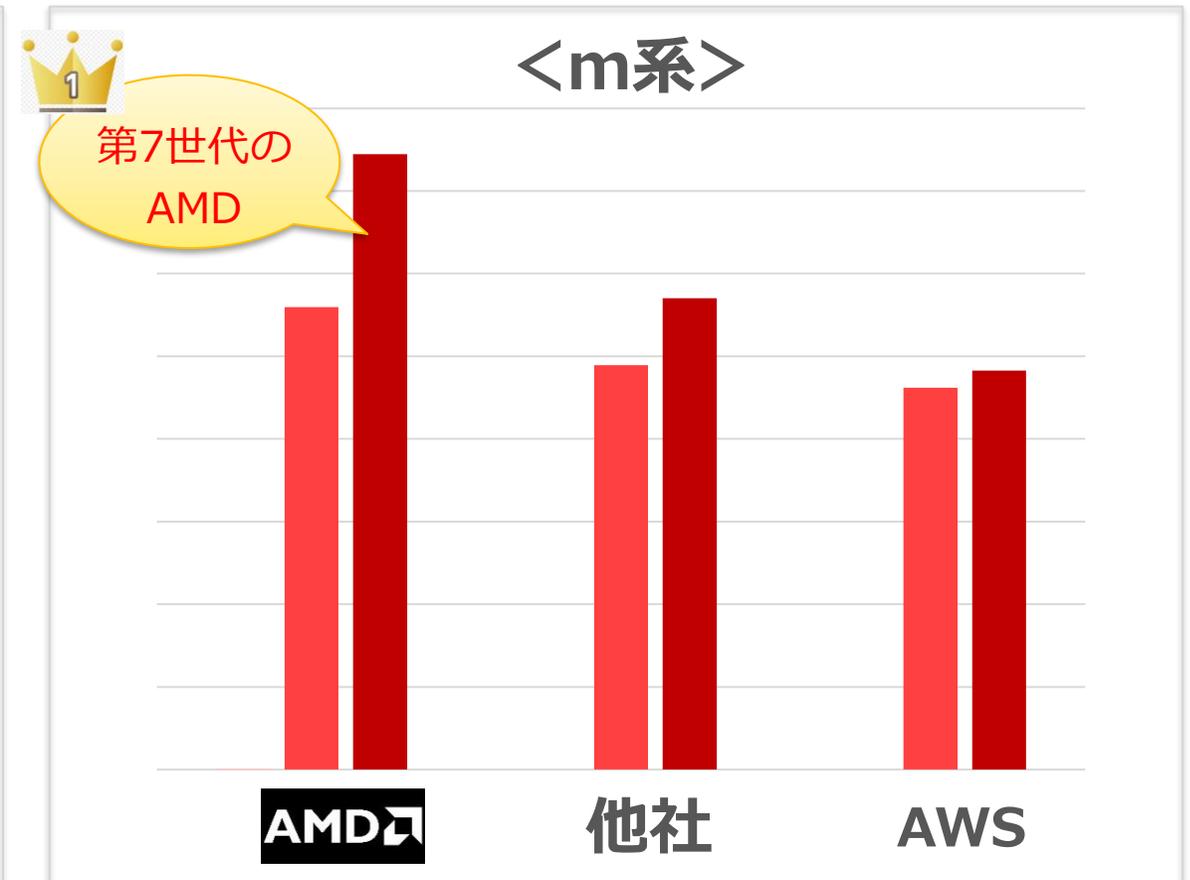
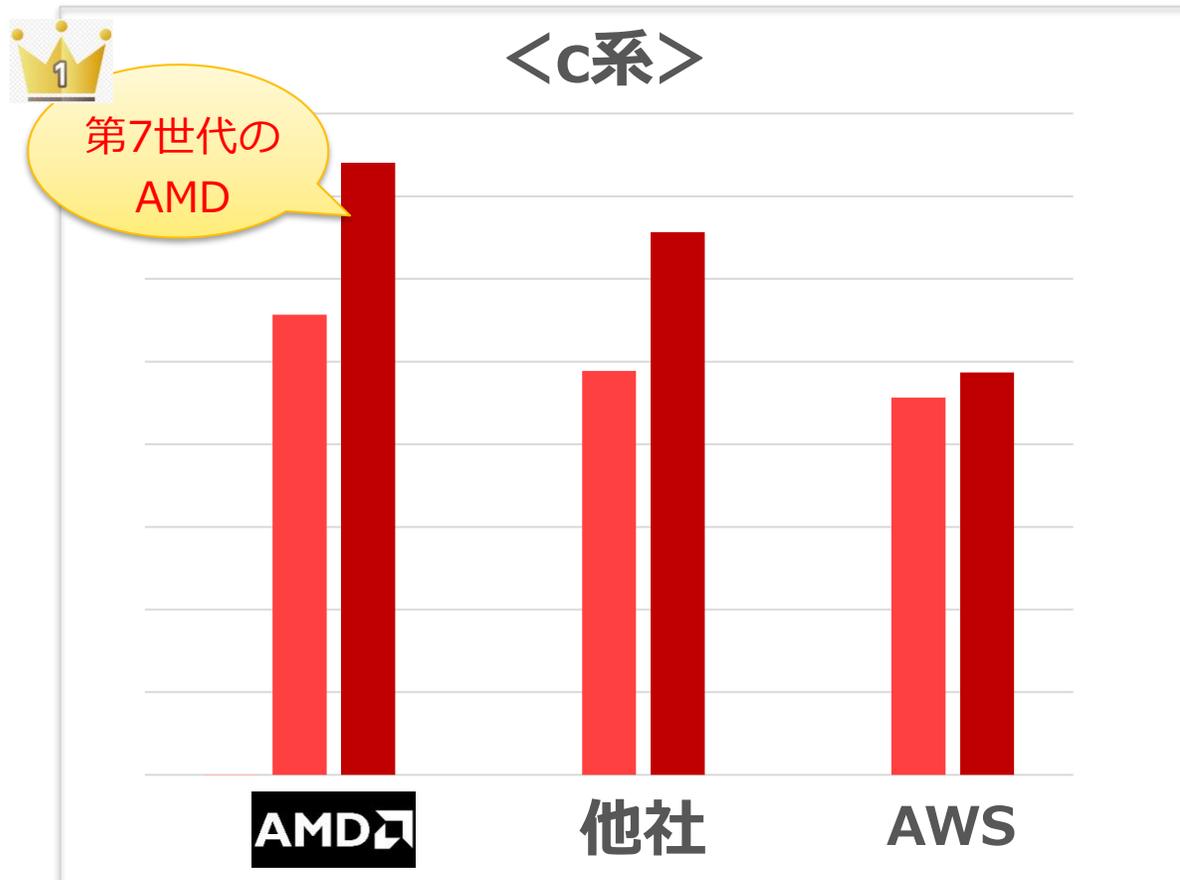
- ▶ インスタンスタイプ : c系 と m系 の **第6世代 と 第7世代**
 - ✓ 選定した比較対象のタイプは下記の表を参照
- ▶ OS : Amazon Linux 2023

c系	タイプ	vCPU	mem	arch	備考
AMD	c6a.large	2	4	x86	AMD EPYC 7R13 Processor
	c7a.large	2	4	x86	AMD EPYC 9R14 Processor
他社	c6i.large	2	4	x86	Intel(R) Xeon(R) Platinum 8375C
	c7i.large	2	4	x86	Intel(R) Xeon(R) Platinum 8488C
AWS	c6g.large	2	4	Arm	Graviton 2
	c7g.large	2	4	Arm	Graviton 3

m系	タイプ	vCPU	mem	arch	備考
AMD	m6a.large	2	8	x86	AMD EPYC 7R13 Processor
	m7a.large	2	8	x86	AMD EPYC 9R14 Processor
他社	m6i.large	2	8	x86	Intel(R) Xeon(R) Platinum 8375C
	m7i.large	2	8	x86	Intel(R) Xeon(R) Platinum 8488C
AWS	m6g.large	2	8	Arm	Graviton 2
	m7g.large	2	8	Arm	Graviton 3

i. ベンチマークテスト：結果

- 結果：各世代で **AMDインスタンスのスコアが一番高い** 結果となった
全てのインスタンスで新しい世代の方がスコアが向上した

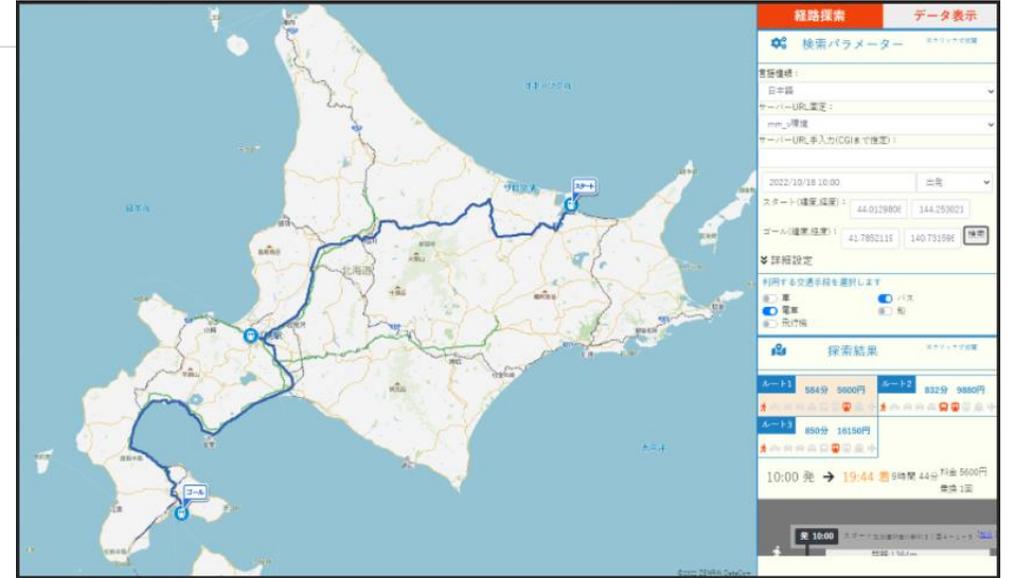


■ : 第6世代 ■ : 第7世代

ii . 負荷テスト (①Engineサーバ) : 試験内容

● 前提条件

- ▶ リージョン : 東京リージョン
- ▶ OS : Amazon Linux 2
- ▶ ミドルウェア : Apache
- ▶ 処理 : 複数の交通手段で指定した目的地の経路検索を行う
 - ✓ 主に経路計算を行うプログラムでCPUリソースを消費する
- ▶ 測定ツール : gatling



- ▶ 測定ツール : gatling
 - ✓ 徐々に同時接続数を増やしていき限界性能を調査
- ▶ インスタンスタイプ(パターン1) : **第6世代** 同士の比較

	Vcpu	mem	arch	
c6a.large	2	4	x86	AMD EPYC 7R13 Processor
c6i.large	2	4	x86	Intel(R) Xeon(R) Platinum 8375C

AMD
第6世代

× 他社
第6世代

- ▶ インスタンスタイプ(パターン2) : **第7世代** 同士の比較

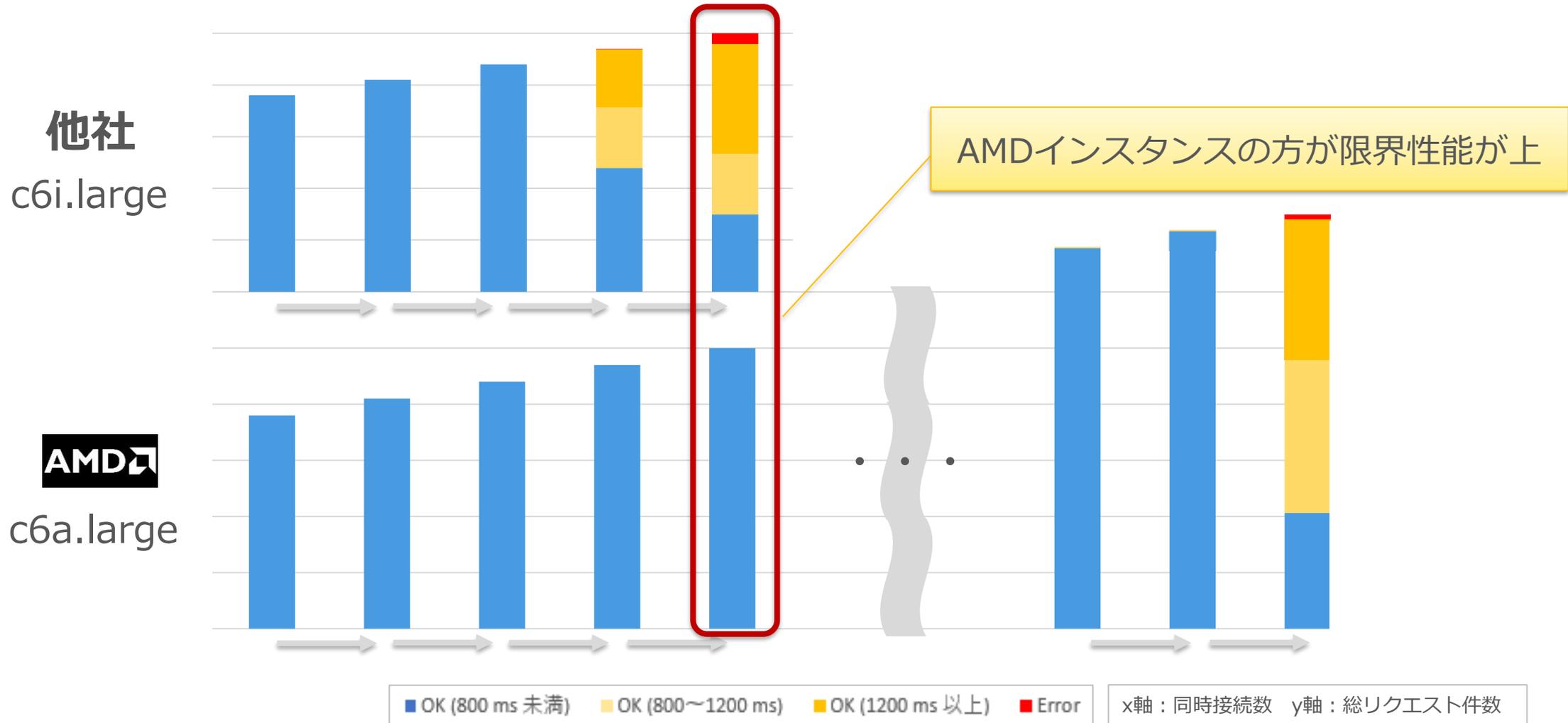
	vcpu	mem	arch	
c7a.large	2	4	x86	AMD EPYC 9R14 Processor
c7i.large	2	4	x86	Intel(R) Xeon(R) Platinum 8488C

AMD
第7世代

× 他社
第7世代

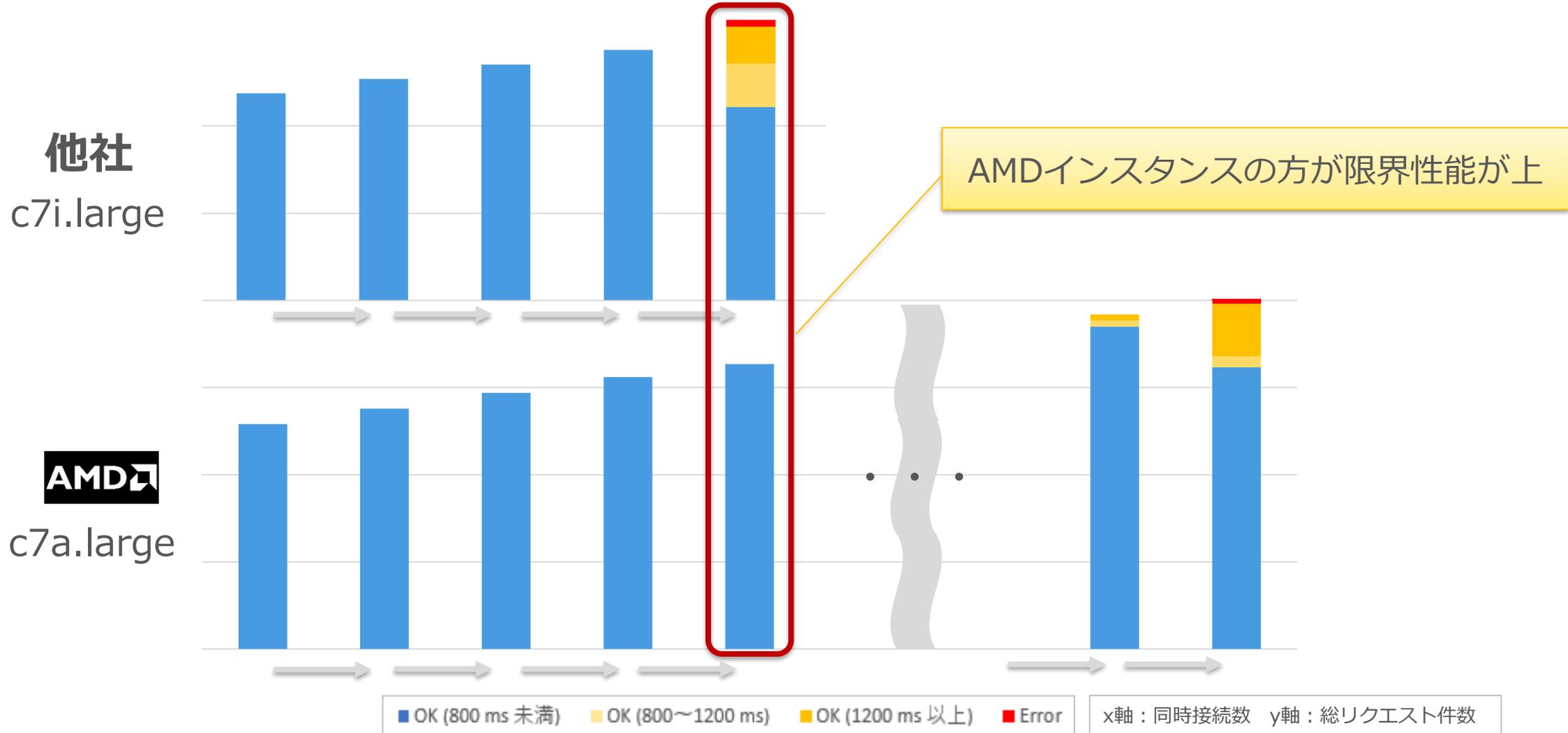
ii. 負荷テスト (①Engineサーバ) : 結果 (第6世代)

- 結果 : **第6世代**の比較では **AMDインスタンス**が、**約8%** 性能が向上



ii. 負荷テスト (①Engineサーバ) : 結果 (第7世代)

- 結果 : **第7世代**の比較では **AMDインスタンス**が、**約12%** 性能が向上



iii. 性能テスト (②APIサーバ) : 試験内容

● 前提条件

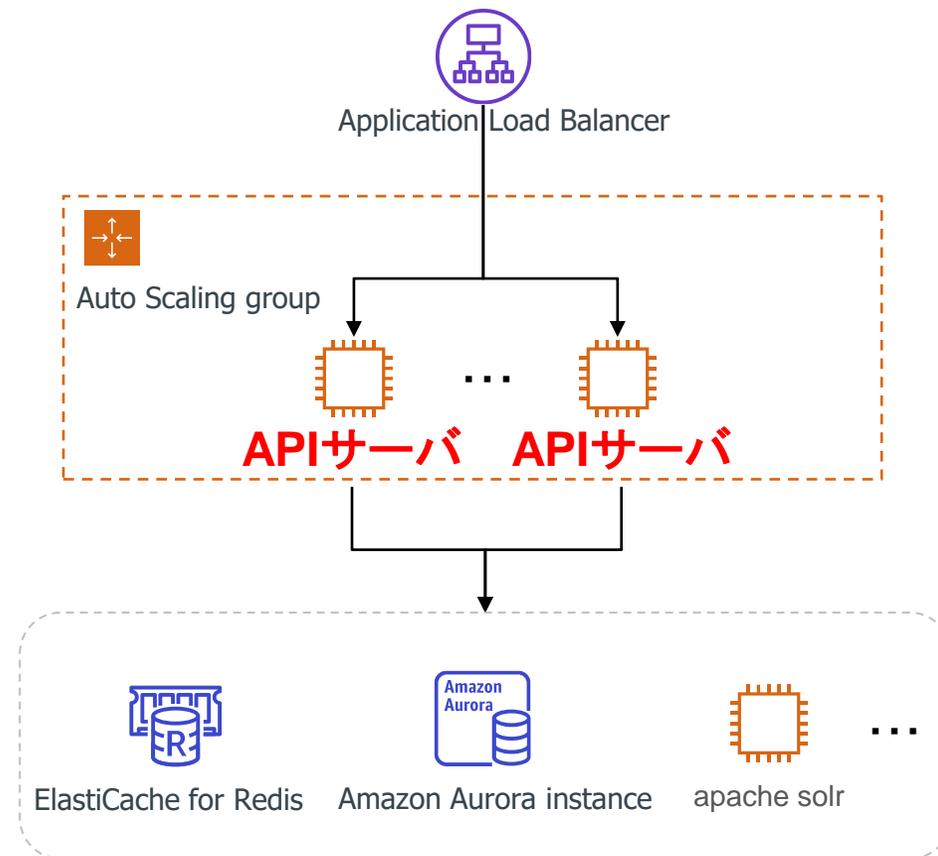
- ▶ リージョン : 東京リージョン
- ▶ インスタンスタイプ : **AMD の 第6世代×第7世代** を比較

	vcpu	mem	arch	
m6a.2xlarge	8	32	x86	AMD EPYC 7R13 Processor
m7a.2xlarge	8	32	x86	AMD EPYC 9R14 Processor



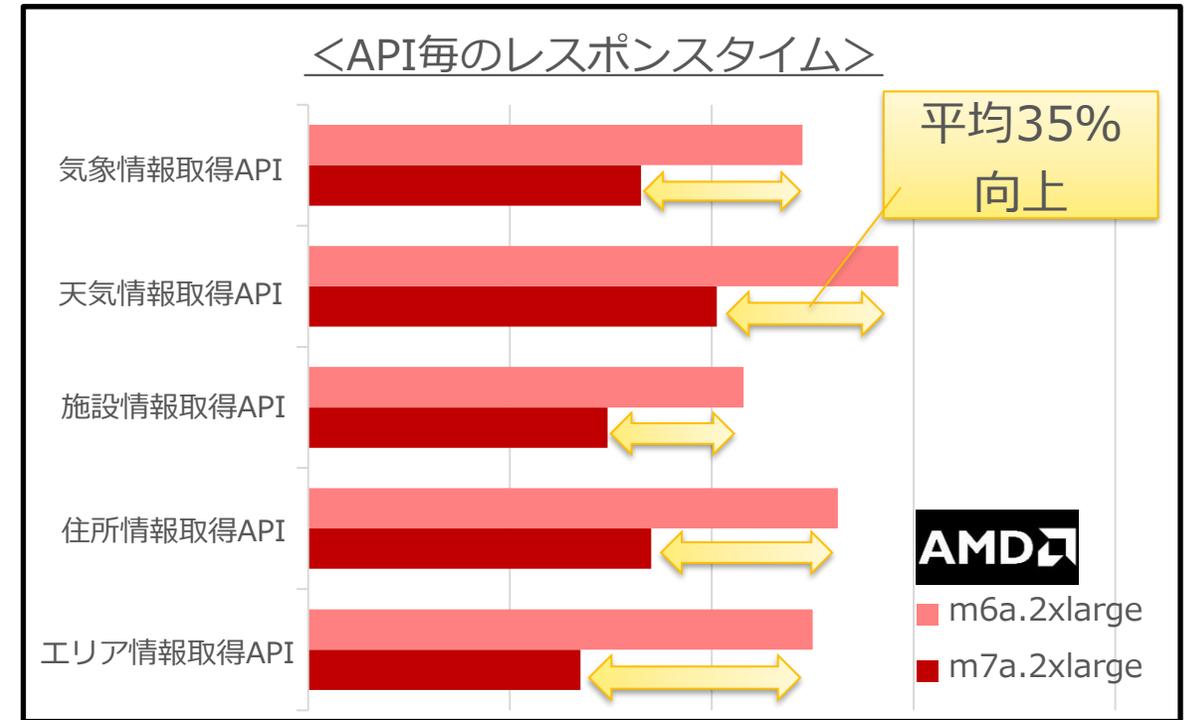
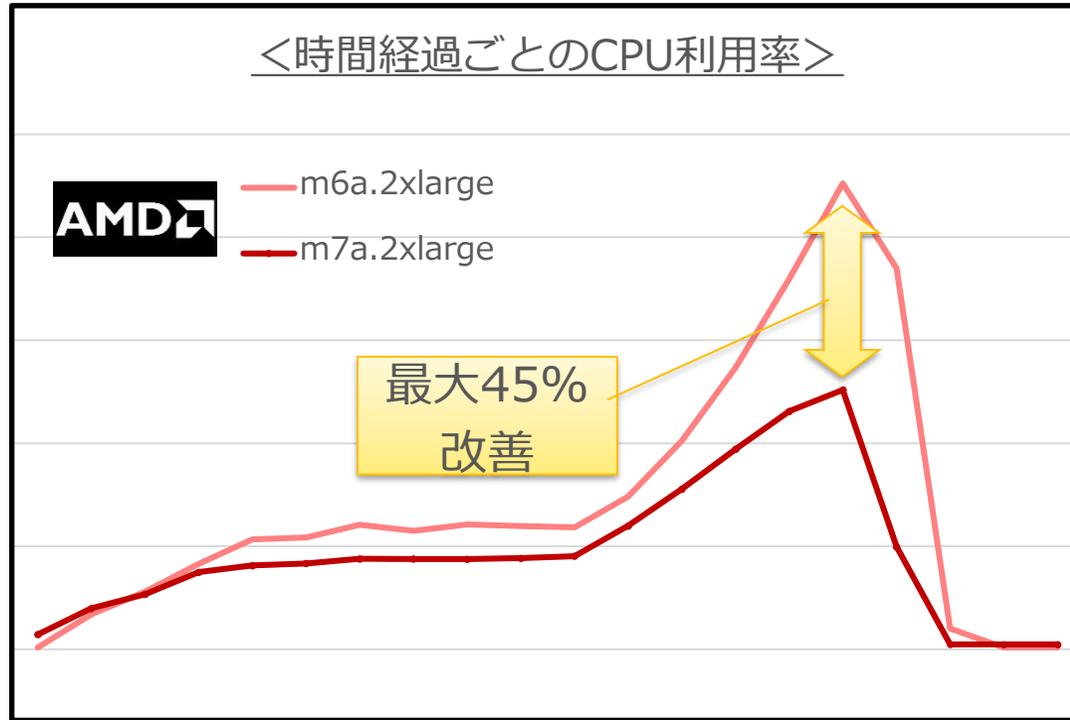
第6世代 → 第7世代

- ▶ OS : Amazon Linux 2
- ▶ ミドルウェア : Nginx
- ▶ 処理 : 様々なコンテンツデータにアクセスして
必要な情報を取得してリクエスト元に返却する
- ▶ 測定ツール : JMeter
 - ✓ 本番同等の負荷掛けを行い
「CPU利用率」と「レスポンスタイム」を調査



iii. 性能テスト (②APIサーバ) : 結果 (第6世代×第7世代)

- 結果 : AMDインスタンスの **第7世代** にすることで
CPU利用率が **最大45%** 改善、レスポンスタイムが **平均35%** 向上



第6世代から第7世代に変更したことで **約40%の台数削減** を実現
性能向上に伴う台数削減により、**第7世代の方が費用対効果が良いケースがある**

iii. 性能テスト (③Batchサーバ) : 試験内容

● 前提条件

- ▶ リージョン : 東京リージョン
- ▶ インスタンスタイプ : **AMD の 第6世代×第7世代** を比較

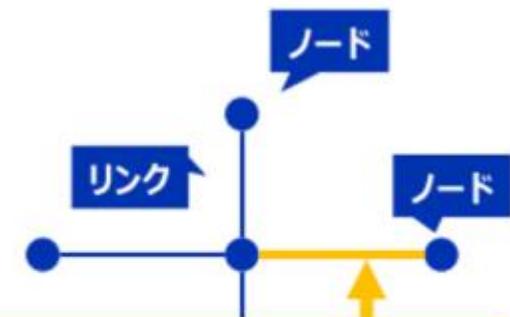
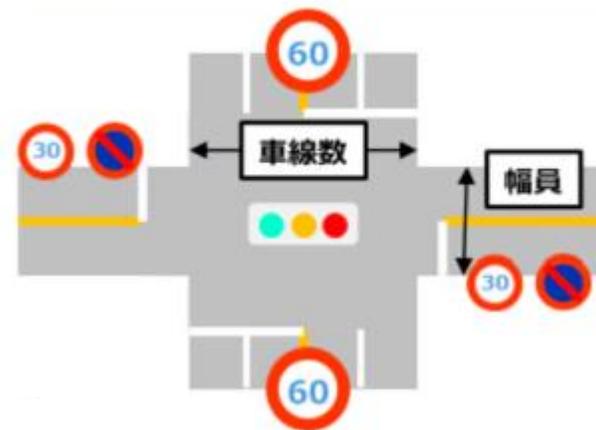
	vcpu	mem	arch	
m6a.4xlarge	16	64	x86	AMD EPYC 7R13 Processor
m7a.4xlarge	16	64	x86	AMD EPYC 9R14 Processor



第6世代 → 第7世代

- ▶ OS : Amazon Linux 2
- ▶ ミドルウェア : PostgreSQL
- ▶ 処理 : 道路情報をコンピュータ(API)で扱える形式に変換する
 - ✓ 道路情報としては 車線数/幅員/制限速度 などが挙げられる
- ▶ 測定方法 : データ変換のバッチ処理にかかる時間を比較する

道路ネットワークデータのイメージ

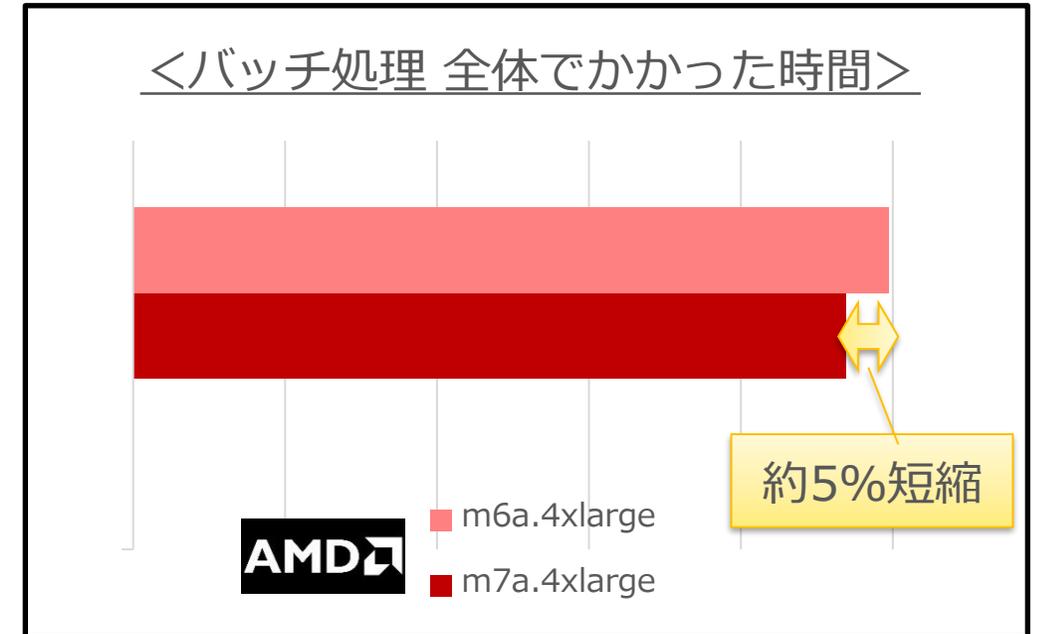
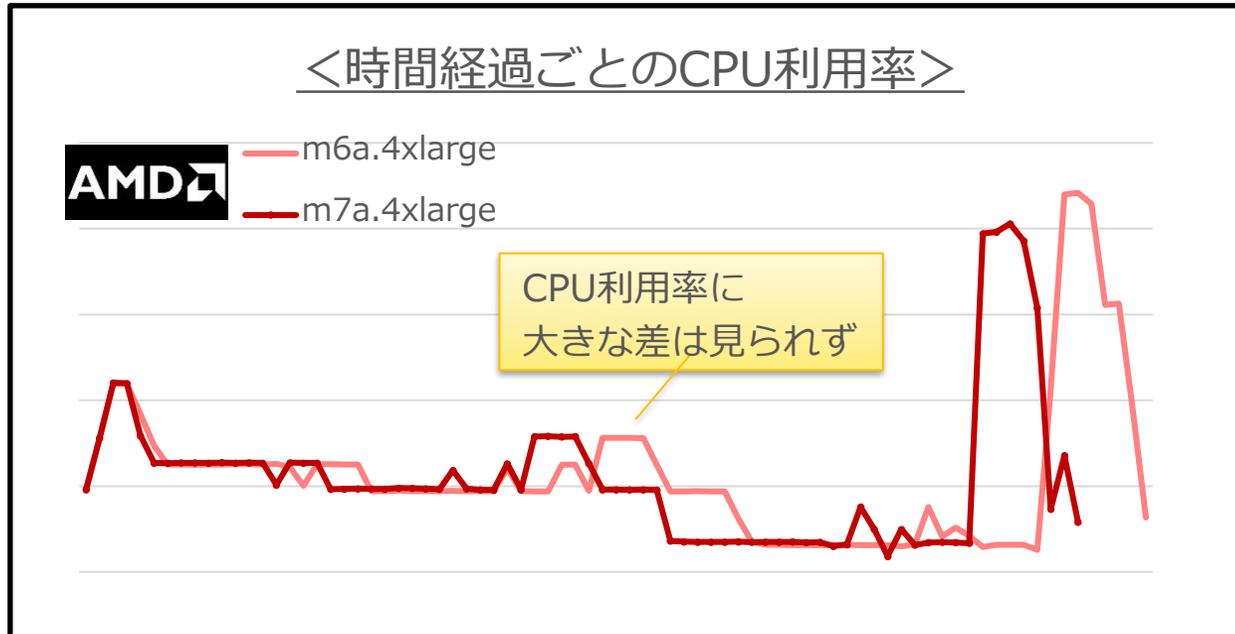


道路ネットワークデータ例

リンクID : link001
ノード座標 : 34.23,139.11、34.25...
幅員 : 12m、車線数 : 2
制限速度 : 30km/h、規制 : 駐車禁止

iii. 性能テスト (③Batchサーバ) : 結果 (第6世代×第7世代)

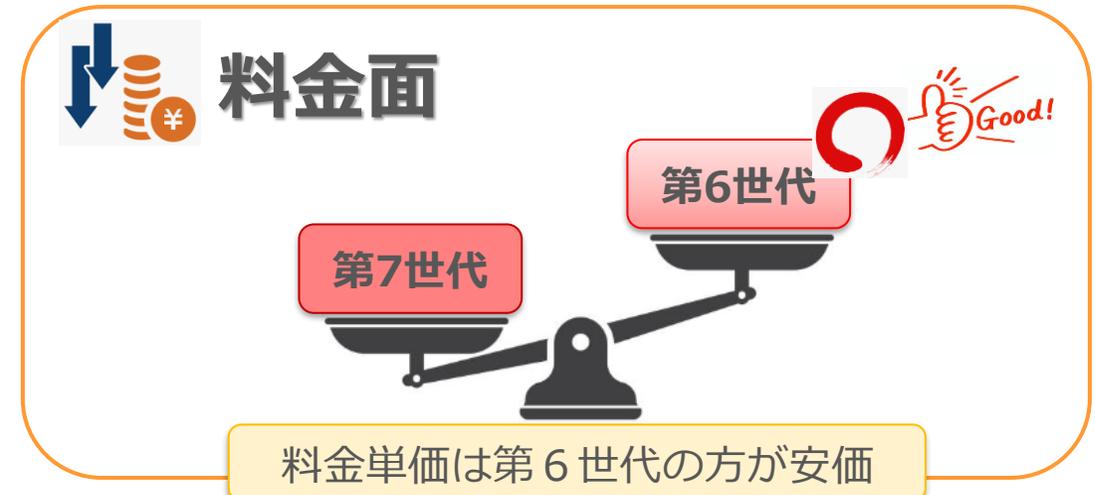
- 結果 : AMDインスタンスの **第7世代** にすることで、CPU利用率は **同程度**、バッチ処理時間は **約5%** の短縮 となった



CPUが高負荷になるような計算処理が少ないと、想定よりも時間短縮せず
結果として、**第6世代の方が 費用対効果** を見込めるケースもある

まとめ

✓ AMDインスタンスは簡単に試せて、お手軽に削減が見込めます

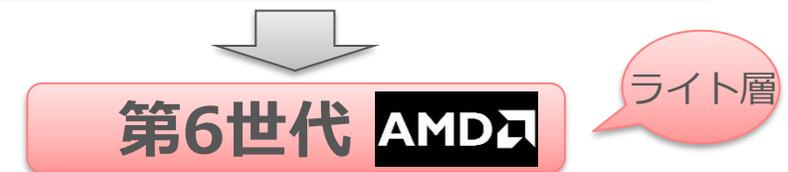


✓ ユースケースに応じて使い分けましょう

- ・サーバ台数が多い（規模:大）
- ・CPUの演算処理がボトルネック
- ・レスポンスタイムの向上が課題



- ・サーバ台数が少ない（規模:小）
- ・少しでも料金を安くしたいがサイジングに時間がとれない



さいごに

地図に関するお困りごとがございましたら
お気軽にお問い合わせ下さい

<https://www.zenrin-datacom.net/contact/business/index>

Area Marker!



いつもNAVI

GODOOR

ゼンリン
地図ナビ

Appendix

M7a Runs Same Job in a Smaller Footprint and Lower Cost

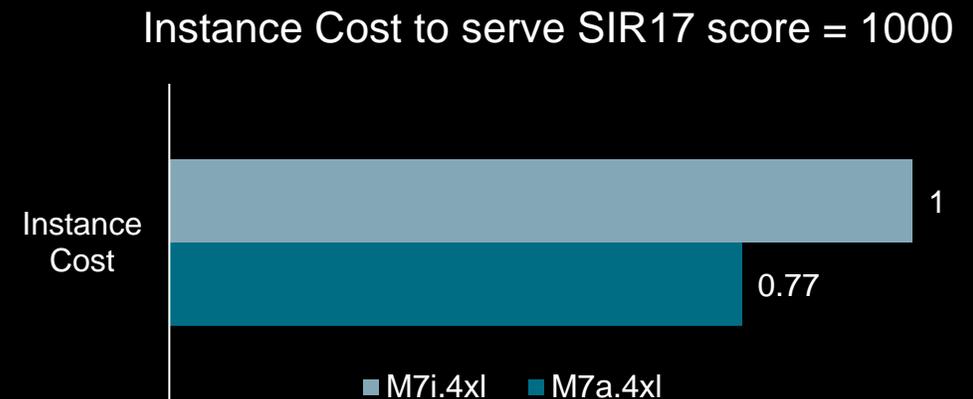
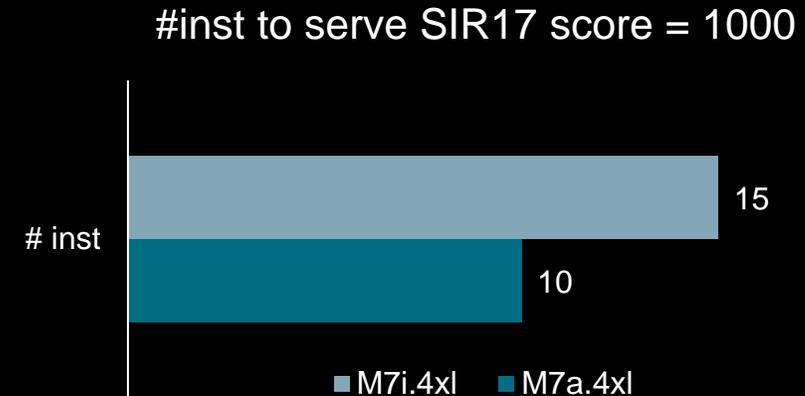
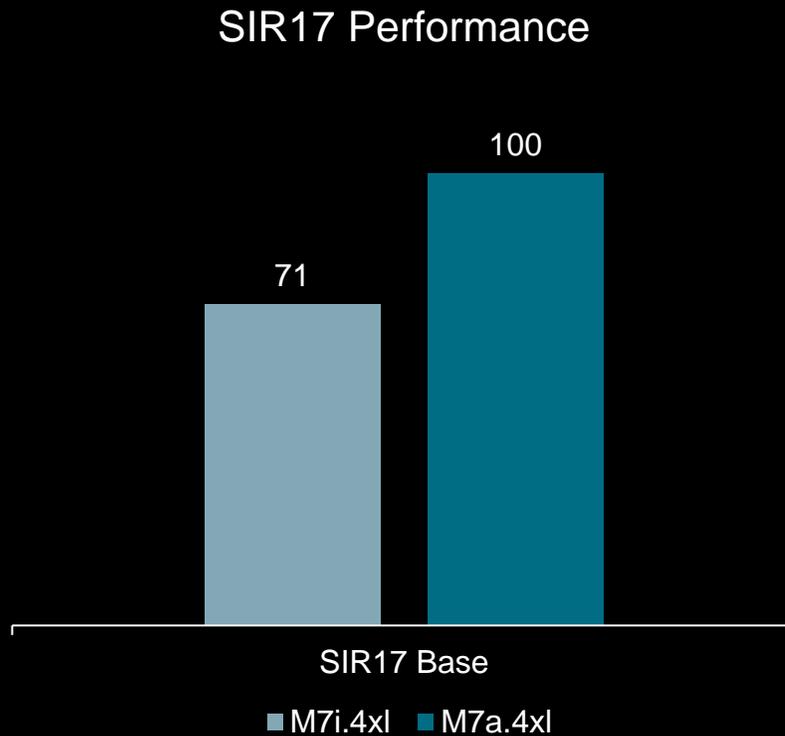
EPYC's leading performance means your M7a instances can run the same job smaller and cheaper vs M7i

GENERAL COMPUTE

SIR17

33% fewer instances on M7a vs M7i

Save \$230K per \$1M spend



Relational (SQL) Databases on M7a - Transactional

EPYC's leading performance means your M7a instances can run the same job smaller and cheaper vs M7i

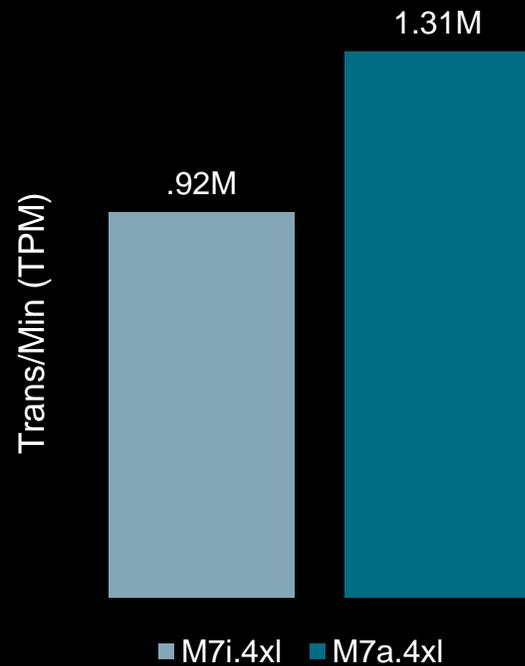
SQL Databases (Transactional)

MySQL

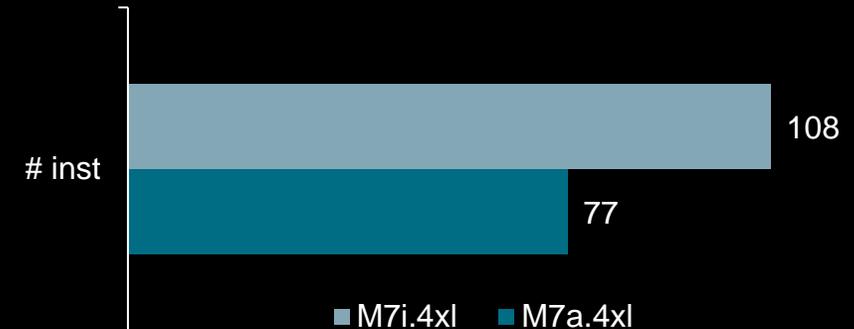
29% fewer instances on M7a vs M7i

Save \$180K per \$1M spend

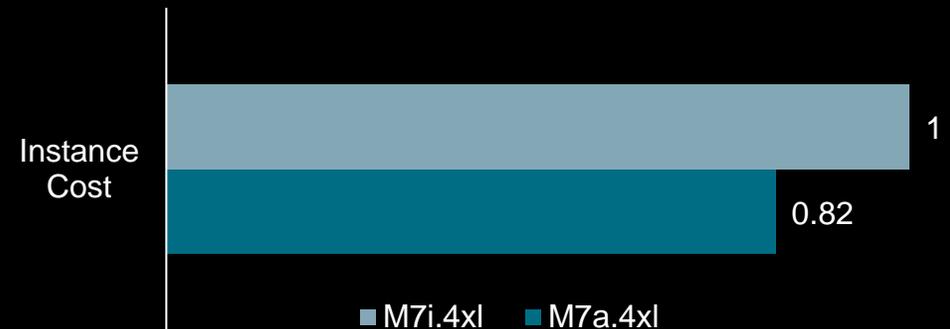
MySQL (TPROC-C) Performance



instances to process 100M SQL TPM



Instance Cost to process 100M SQL TPM



Relational (SQL) Databases on M7a - Analytics

EPYC's leading performance means your M7a instances can run the same job smaller and cheaper vs M7i

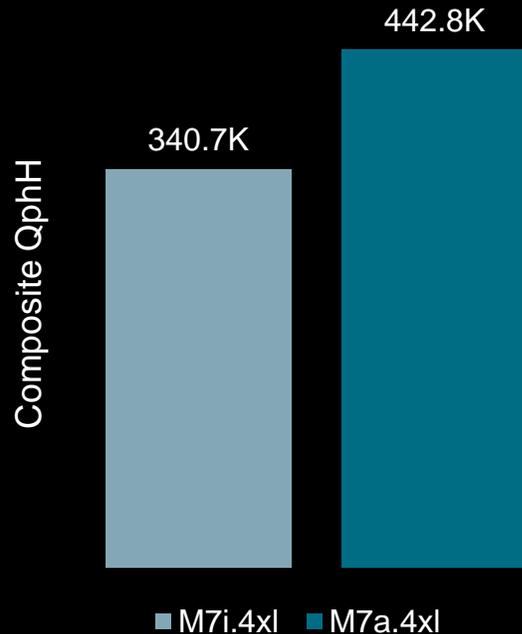
SQL Databases (Analytics)

MS SQL Server

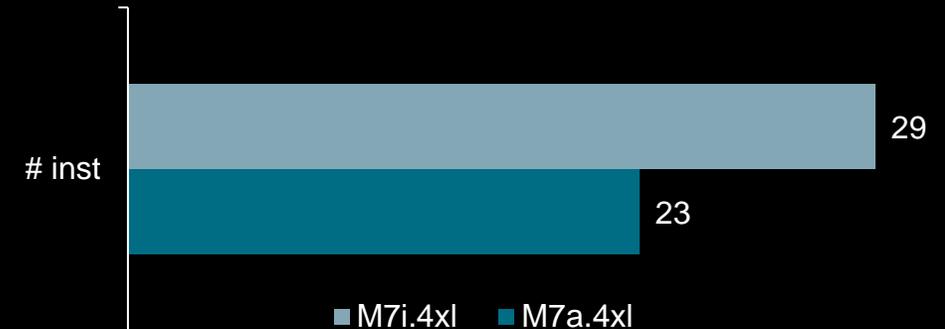
21% fewer instances on M7a vs M7i

Save \$90K per \$1M spend

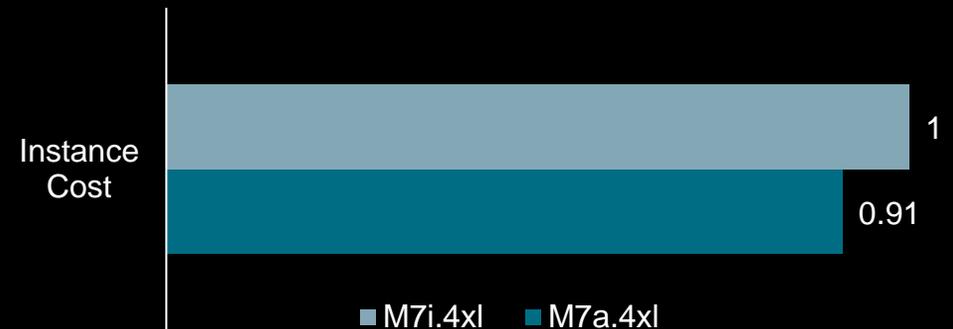
MS SQL Server (TPROC-H) Performance



instances to process 10M QphH



Instance Cost to process 10M QphH



In-memory Non-relational Databases on M7a

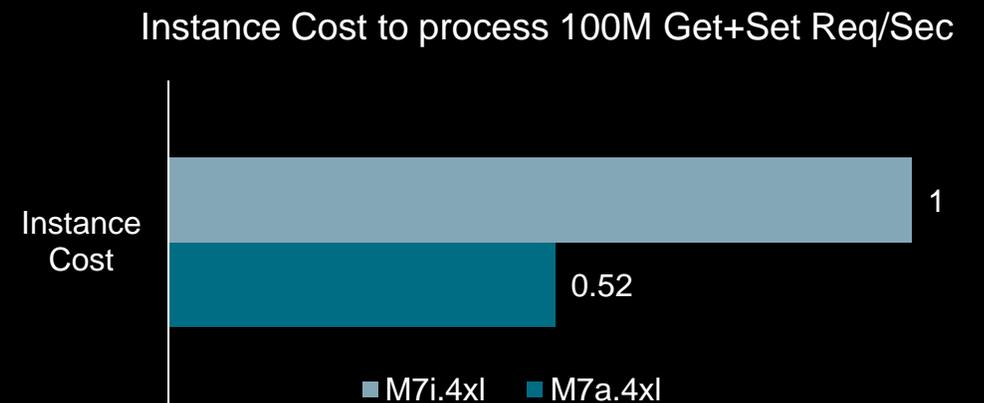
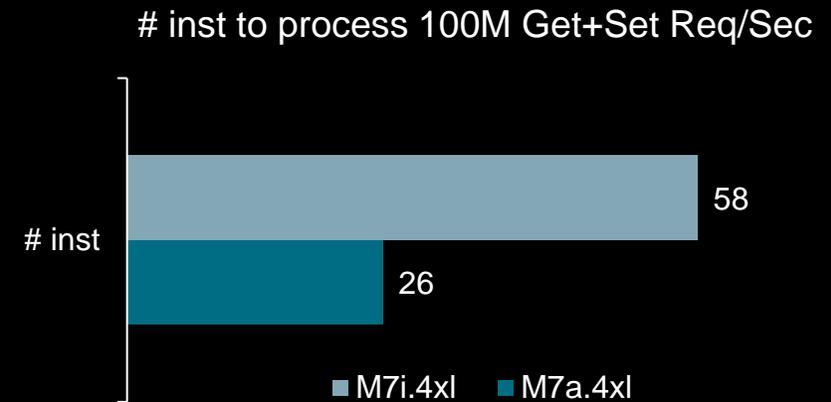
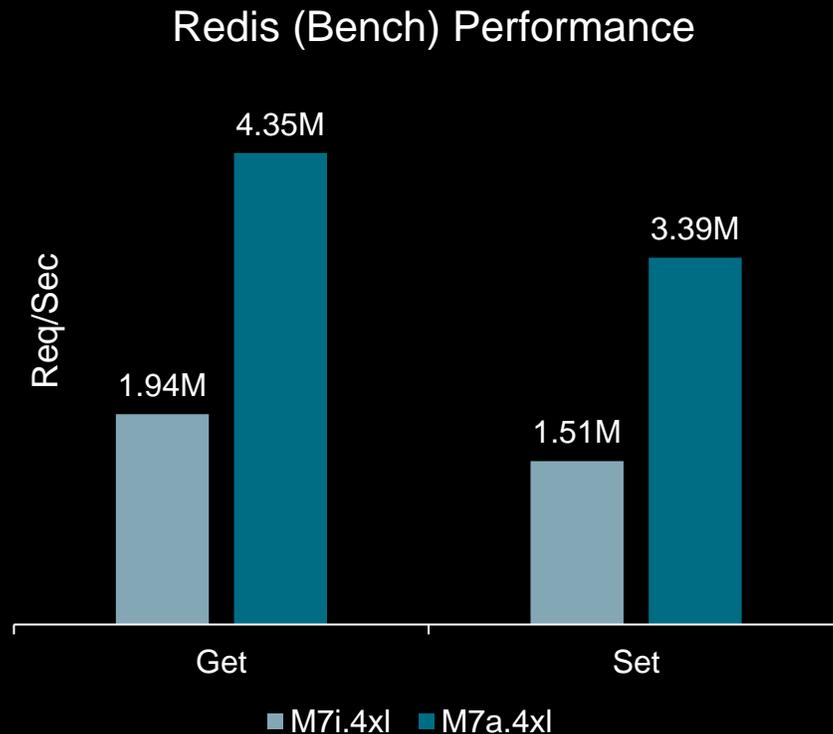
EPYC's leading performance means your M7a instances can run the same job smaller and cheaper vs M7i

In-Memory DB

REDIS

55%
fewer instances
on M7a vs M7i

Save \$480K
per \$1M spend
on M7a vs. M7i



Video Encoding & Transcoding on M7a

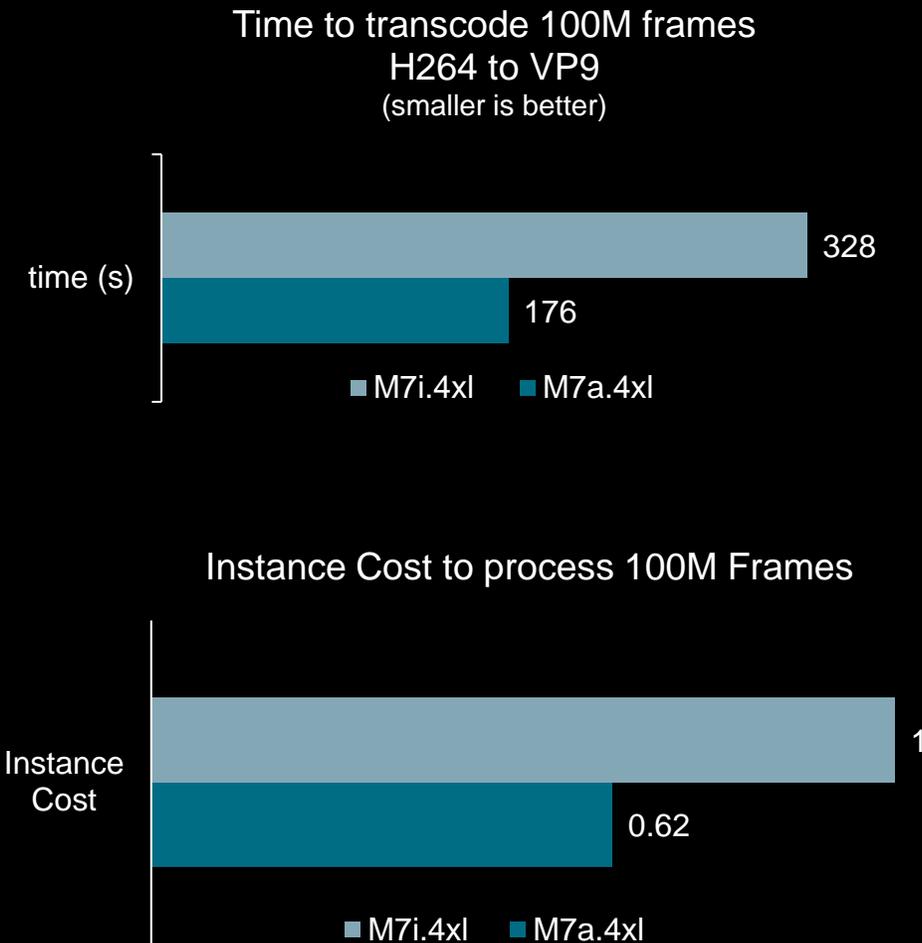
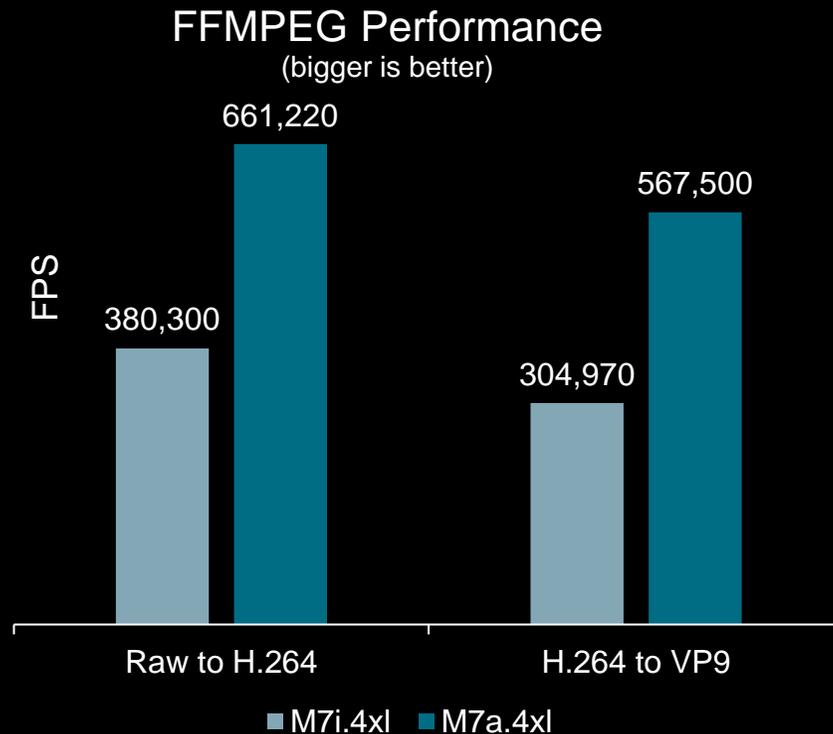
EPYC's leading performance means your M7a instances can run the same job faster and cheaper vs M7i

In-Memory DB

FFMPEG

46% fewer instances on M7a vs M7i

Save \$380K per \$1M spend on M7a vs. M7i



M6a Runs Same Job in a Smaller Footprint and Lower Cost

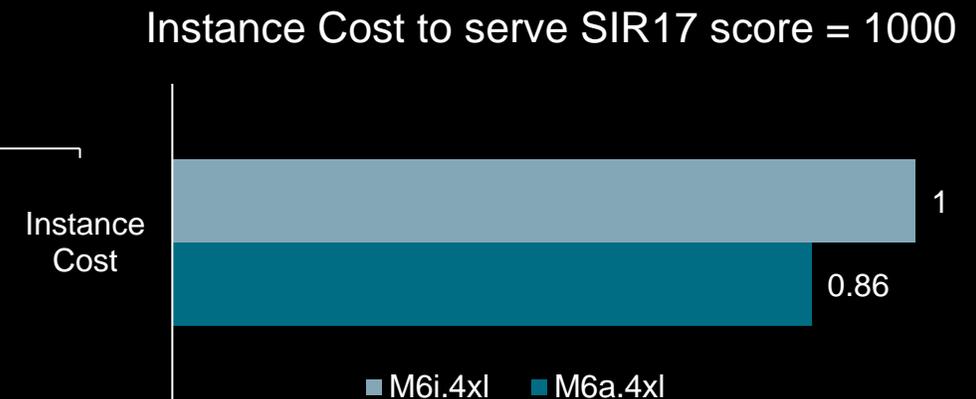
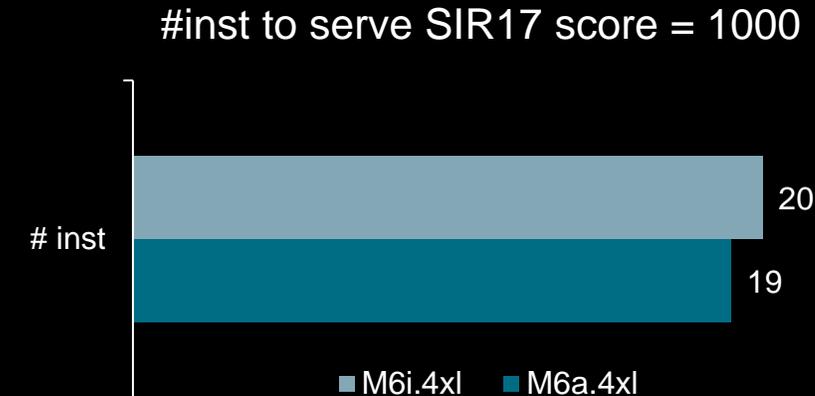
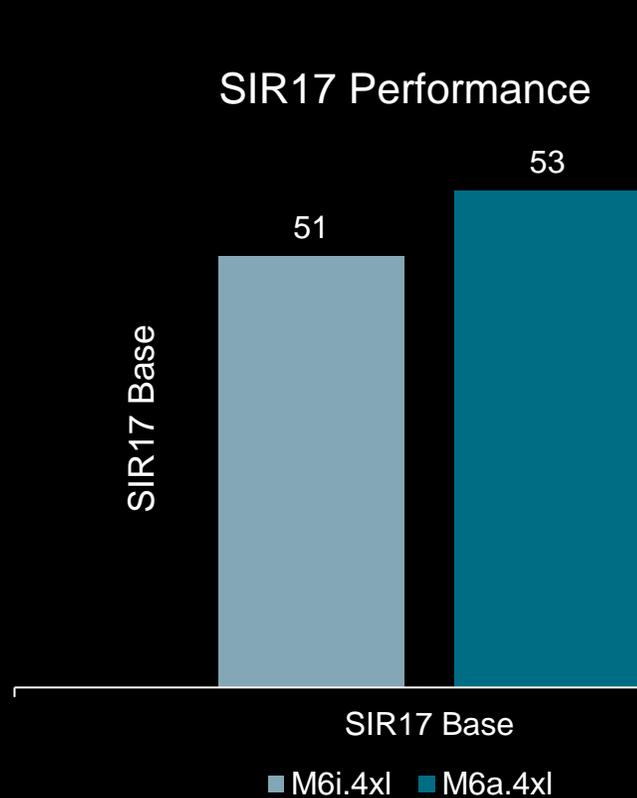
EPYC's leading performance means your M6a instances can run the same job smaller and cheaper vs M6i

GENERAL COMPUTE

SIR17*

5% fewer instances on M6a vs M6i

Save \$140K per \$1M spend



* SPEC CPU®2017 Integer Rate

Thank you!

