Available online www.ejaet.com

European Journal of Advances in Engineering and Technology, 2022,9(7):51-58

Research Article ISSN: 2394-658X

Compute Cost Optimization: Unleashing Cost Savings and Efficiency Wins with Current-Gen AWS EC2 Instances/ Virtual Machines (VM)

Venkata Sasidhar (Sasi) Kanumuri

Email id - sasinrt@gmail.com

ABSTRACT

The ever-growing cloud landscape demands strategic cost management practices, and non-containerized workloads often pose a significant cost challenge. This article explores how new-generation EC2 instances (C, R, M families) offer a transformative approach to optimizing and empowering these workloads. We delve into the limitations of traditional methods, highlighting the performance, pricing, and feature advancements offered by new-gen instances. Concrete comparisons and quantifiable results showcased through real-world case studies demonstrate substantial cost savings, performance gains, and enhanced efficiency. Finally, the article provides a roadmap for a successful migration, empowering businesses to unlock the potential of new-gen EC2 instances for their non-containerized deployments and achieve a more sustainable and cost-effective cloud strategy.

Key words: Cloud computing, cost optimization, non-containerized workloads, EC2 instances, performance, efficiency, migration, AWS compute instances, current-gen instances, m family, r family, t family and c family instance types.

INTRODUCTION

The explosive growth in cloud computing has made powerful computing resources more accessible to a broader range of sectors, encouraging creativity and adaptability. However, as cloud computing has advanced, cost management has become more scrutinized since businesses look for ways to reduce costs without sacrificing service quality or performance. Even while containerization has become the de facto standard for effective microservices design, many cloud workloads still need to be part of this containerized environment. This article highlights these 'non-containerized' workloads, highlighting their potential as a critical frontier for cost optimization and efficiency gains within the AWS cloud ecosystem.

Recent studies indicate a rising trend of cost concerns in cloud computing, with organizations needing help reconciling their cloud adoption ambitions with budgetary constraints. A prominent report by Flexera revealed that 30% of cloud spend is wasted due to underutilized or idle resources, emphasizing the need for strategic cost management practices. Traditional approaches often focus on infrastructure optimization within containerized environments, neglecting the substantial cost footprint of non-containerized workloads.

This article posits that organizations can unlock significant cost savings and enhanced efficiency by strategically addressing non-containerized deployments on Amazon Elastic Compute Cloud (EC2) instances and optimizing overall cloud utilization. This paper mainly focuses on AWS EC2 instances and examines how the newest generation of instances (C5, R5, and M5 families) provides an excellent foundation for optimizing workloads that are not containerized. Compared to prior generations, these next-generation instances have the potential to yield significant cost reductions and increased resource usage. The paper explores the performance, price, and feature advancements that these instances offer. The subsequent sections will further elaborate on these benefits, showcasing concrete comparisons and quantifiable results to guide businesses in navigating this innovative approach to cloud cost optimization.

Through this analysis, we aim to equip cloud architects and engineers with valuable insights into leveraging non-containerized deployments on new-gen EC2 instances. This article underscores this approach's potential to optimize cost and boost resource efficiency, ultimately contributing to a more sustainable and cost-effective cloud strategy within the AWS environment.

OLD GEN VS. CURRENT GEN EC2 INSTANCES: A PERFORMANCE EVOLUTION

A. UNDERSTANDING OLD GEN INSTANCES

Cloud computing was made possible by the Old Gen Amazon EC2 instances, although the New Gen versions have since superseded them. They performed slower than current models since they were powered by older Intel Xeon Nehalem processors, which had less CPU and memory capacity. They might also not support the newest features and optimizations found in instance, more recent families. Examples include M1, M2, M3, and M4 (general-purpose), C1, C3, and C4 (compute-intensive), R3 and R4 (memory-intensive), and T1 and T2 (low-cost). With their limited capabilities and deprecation by AWS, Old Gen instances are generally more expensive and less efficient than their New Gen successors. As technology evolves, so does the demand for more powerful and cost-effective solutions, making New Gen instances the clear choice for modern cloud deployments.

B. INTRODUCING CURRENT AND NEW GEN INSTANCES

The cloud landscape is a living, breathing entity that constantly evolves and demands innovation. Amazon's ever-evolving EC2 instances exemplify this, with the latest generation—"c5," "r5," "m5," and their "a" variants—marking a quantum leap in performance, features, and cost-effectiveness compared to their predecessors.

These instances cater to a diverse range of workloads with specialized options. For computationally demanding tasks like data analytics and scientific simulations, the c5 instances stand ready, boasting a high vCPU-to-memory ratio for maximum processing power. Memory-hungry applications like large in-memory databases or real-time analytics can leverage the memory-prioritizing r5 instances. General-purpose workloads like web servers and small databases find their sweet spot with the balanced blend of vCPUs and memory offered by m5 instances. Even bursty workloads with fluctuating CPU demands can optimize costs with t3 instances, providing a baseline performance with the flexibility to burst when needed.

But these instances go beyond raw power. They embrace innovation, offering enhanced networking capabilities and the latest Nitro hypervisor for superior performance and security for your applications. Scalability allows you to tailor vCPU, memory, and storage to your specific needs with a diverse range of instance sizes. Specific instances are optimized for Amazon's Elastic Block Store, ensuring efficient storage utilization for your data.

Moreover, introducing AMD EPYC processors as an alternative to Intel Xeon options further expands your choice. While both deliver comparable performance, EPYC processors can offer cost savings for specific workloads due to their high core counts, large memory capacities, and cutting-edge features.

The latest generation of EC2 instances represents a paradigm shift. With their diverse offerings, superior performance, advanced features, and cost-effectiveness, they empower organizations to unlock new possibilities and optimize their cloud infrastructure for the future. This generation is not just a powerhouse; it's a versatile toolkit for building and scaling your cloud applications in a dynamic and cost-conscious manner. So, embrace the evolution and empower your cloud with the latest generation of EC2 instances.

The dynamic nature of cloud computing requires using flexible and adaptive application infrastructure and integrating robust autoscaling techniques. Here's where Kubernetes enters the scene. It is a popular platform for container orchestration that makes managing containerized apps easier. However, to maximize resource use, efficient autoscaling strategies are required. Kubernetes Autoscaling offers a dynamic platform that can adapt to real-time changing needs, improving scalability, effectiveness, and affordability.

C. WHY NEW GEN EC2 INSTANCES REIGN SUPREME

AWS's release of EC2 Instances presents a convincing improvement over the earlier instances regarding functionality, performance, price, security, and sustainability.

[1]. **Unleashing Performance:** At the core lies a performance leap. New generation instances leverage cutting-edge CPU architectures, faster clock speeds, and enhanced memory bandwidth, translating to significant boosts for compute-intensive workloads.

[2]. **Cost Optimization:** Cost efficiency is paramount in the cloud. New-generation instances deliver exceptional value, often offering superior price-to-performance ratios. This translates to obtaining more computational power for your investment, allowing you to optimize your cloud budget.

[3]. **Feature Innovation:** These instances go beyond raw power, incorporating advanced features unavailable in previous generations. This includes additional instruction sets, hardware acceleration for specific workloads (like machine learning), and improved networking capabilities for enhanced data throughput and latency reduction.

[4]. **Security Fortified:** Security remains a top priority. New generation instances benefit from the latest hardware and security upgrades, including updated CPUs with built-in security features like Intel SGX or ARM TrustZone. This strengthens the security posture of your applications and data, fostering a more secure cloud environment.

[5]. **Sustainability:** AWS prioritizes sustainability, and new-generation instances reflect this commitment. They are designed with energy efficiency in mind, consuming less power while delivering equal or even superior performance. This translates to a reduced carbon footprint and a contribution to a greener cloud landscape.

[6]. **Diverse Choices for Diverse Needs:** AWS expands its EC2 instance-type offerings with each generation, providing a broader spectrum of options to suit various workload demands. New types offer increased CPU, memory, storage, or networking capabilities, empowering your applications with greater flexibility and scalability.

[7]. **Reservations for Enhanced Savings:** By strategically utilizing Reserved Instances (RIs) with new instance types, organizations can maximize resource savings and efficiency compared to on-demand pricing in older generations.

New generation EC2 instances represent a significant leap forward, offering superior performance, cost-effectiveness, features, security, energy efficiency, and instance-type diversity. However, carefully evaluating your workload requirements and available instance types remains crucial for selecting the optimal EC2 generation to meet your needs. By embracing this evolution, you can unlock new possibilities and optimize your cloud infrastructure for the future.

The following figures, 1.1, 1.2, 1.3, and 1.4, highlight the key differences between the variants of the c-family, m-family, r-family, and t-family instances.

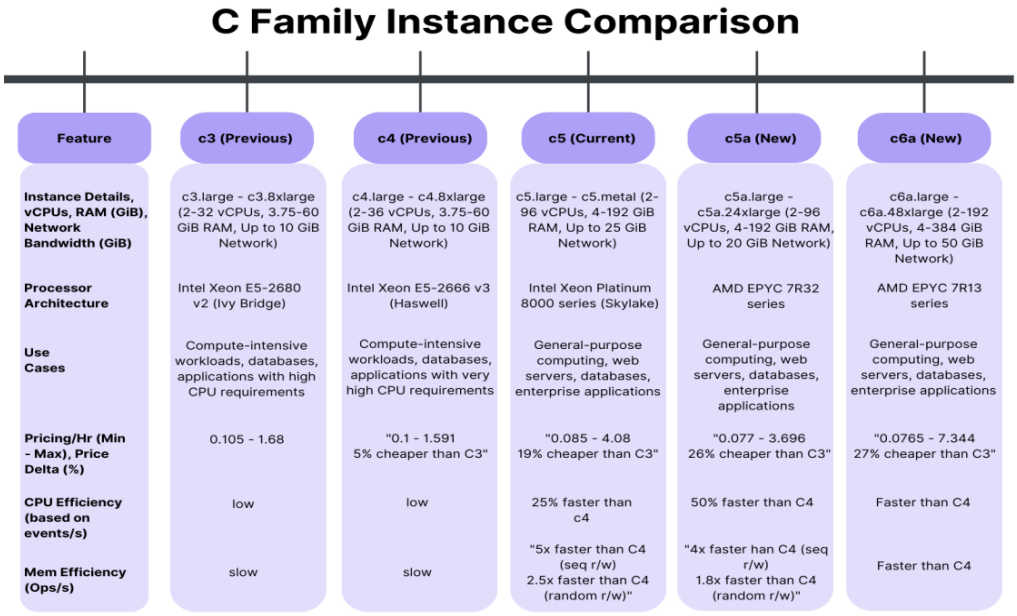

Figure 1: C-family Instance Comparison

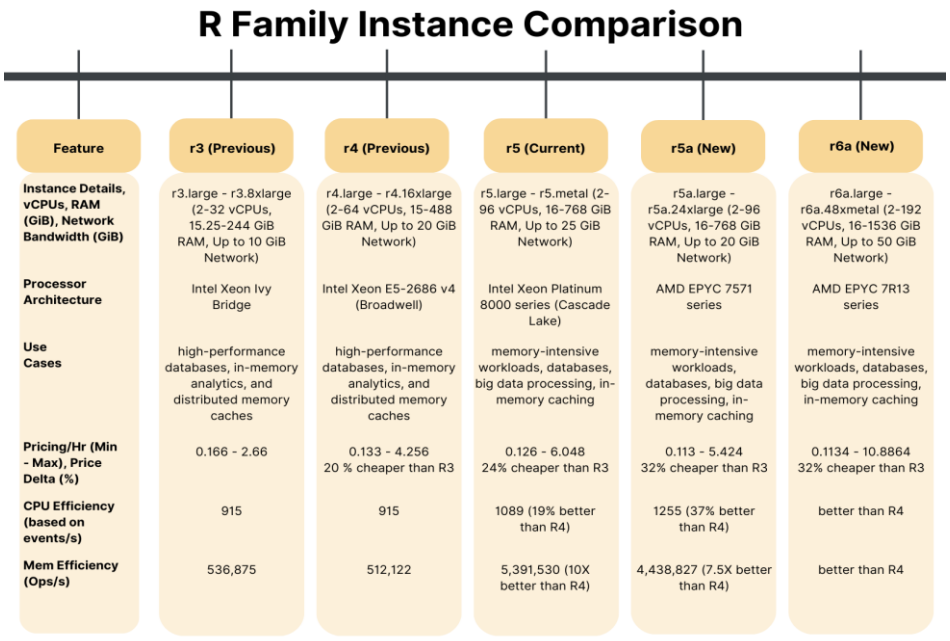

Figure 2: R-family Instance Comparison

M Family Instance Comparison

Feature	m3 (Previous)	m4 (Previous)	m5 (Current)	m5a (New)	m6a (New)
Instance Details, vCPUs, RAM (GiB), Network Bandwidth (GiB)	m3.medium - m3.2xlarge (1-8 vCPUs, 3.75-30 GiB RAM, High GiB Network)	m4.large - m4.16xlarge (2-64 vCPUs, 8-256 GiB RAM, Up to 20 GiB Network)	m5.large - m5.metal (2-96 vCPUs, 8-384 GiB RAM, Up to 25 GiB Network)	m5a.large - m5a.24xlarge (2-96 vCPUs, 8-384 GiB RAM, Up to 20 GiB Network)	m6a.large - m6a.metal (2-192 vCPUs, 8-768 GiB RAM, Up to 50 GiB Network)
Processor Architecture	Intel Xeon E5-2670 v2 (Ivy Bridge/Sandy Bridge)	Intel Xeon E5-2676 v3 (Haswell)	Intel Xeon Platinum 8000 series (Cascade Lake)	AMD EPYC 7571 series	AMD EPYC 7R13 series
Use Cases	general-purpose workloads, databases, applications with moderate compute and memory requirements	general-purpose workloads, databases, applications with moderate compute and memory requirements	general-purpose workloads, web servers, app servers, gaming servers	general-purpose workloads, web servers, app servers, gaming servers	general-purpose workloads, web servers, app servers, gaming servers
Pricing/Hr (Min - Max), Price Delta (%)	0.067 - 0.532	0.1 - 3.2	0.096- 4.608	0.101 - 4.848	0.1008 - 9.6768
CPU Efficiency (based on events/s)	low	low	20% faster than M4	40% faster than M4	faster than m4 and comparable with M5a
Mem Efficiency (Ops/s)	slow	slow	10x faster than M4 (seq r/w) 2-3x faster than C4 (random r/w)	7.5x faster than M4 (seq r/w) 2.5x faster than C4 (random r/w)	faster than m4 and comparable with M5a

Figure 3: M-famiy Instance Comparison

T Family Instance Comparison

Feature	t2 (Previous)	t3 (Current)	t3a (New)
Instance Details, vCPUs, RAM (GiB), Network Bandwidth (GiB)	t2.nano - t2.2xlarge (1-8 vCPUs, 0.5-32 GiB RAM, Moderate GiB Network)	t3.nano - t3.2xlarge (2-8 vCPUs, 0.5-32 GiB RAM, Up to 5 GiB Network)	t3a.nano - t3a.2xlarge (2-8 vCPUs, 0.5-32 GiB RAM, Up to 5 GiB Network)
Processor Architecture	Intel Xeon Family	Intel (sky Lake)	AMD EPYC 7571 series
Use Cases	web applications, development/testing environments with moderate resource requirements, batch processing workloads	burstable workloads, small-scale applications, development environments	burstable workloads, small-scale applications, development environments
Pricing/Hr (Min - Max), Price Delta (%)	0.0058 - 0.3712	0.0052 - 0.3328 10 % cheaper than t2	0.0047 - 0.3008 19 % cheaper than t2
CPU Efficiency (based on events/s)	768.84	987.92 28 % better than t2	1205.66 57 % better than t2
Mem Efficiency (Ops/s)	526,954	4415399.92 7.4X better than t2	4438112.57 7.4X better than t2

Figure 4: t-family Instance Comparison

ACHIEVING CLOUD EFFICIENCY: A FRAMEWORK FOR SELECTING OPTIMAL EC2 INSTANCES

The growing cloud computing landscape necessitates strategically allocating resources for optimal performance and cost-effectiveness. As a result, it's crucial to select the right EC2 instance type within Amazon Web Services (AWS) for your workload.

A. KEY WORKLOAD REQUIREMENTS

[1]. **Computational Demands:** Analyze the CPU requirements of your workload. Compute-intensive tasks like batch processing or media transcoding necessitate instances with high-performance processors (C-Family). Conversely, workloads involving balanced resource utilization, such as web servers or code repositories, can leverage general-purpose instances (M-Family).

[2]. **Memory Requirements:** Memory-bound workloads processing large datasets benefit from instances with ample RAM (R-Family). Consider the expected data size and access patterns when choosing an instance type.

[3]. Storage Needs: Select instances based on storage requirements. High-speed sequential access demands necessitate Storage Optimized instances (I-Family), while random I/O-intensive tasks might benefit from General Purpose or Compute Optimized options.

[4]. Network Bandwidth: Analyze how much network bandwidth your workload requires. Workloads requiring a lot of data transfer or real-time communication are best suited for instances with improved network capabilities (like G-Family).

[5]. Performance Expectations: Clearly define the desired performance benchmarks for your workload. Instances within each family offer varying performance levels, allowing for targeted selection based on specific needs.

B. AVAILABLE INSTANCE FAMILIES

AWS offers a diverse range of EC2 instance families, each tailored to address distinct workload characteristics:

[1]. General Purpose (M-Family): Balanced mix of CPU, memory, and network resources for various tasks.

[2]. Compute Optimized (C-Family): High-performance processors for compute-intensive workloads.

[3]. Memory Optimized (R-Family): Fast performance for large in-memory datasets.

[4]. Accelerated Computing (P/G/F/T-Families): Hardware accelerators for specific graphics processing or machine learning functions.

[5]. Storage Optimized (I-Family): High-speed local storage for workloads with sequential access demands.

[6]. High-Performance Computing (HPC) Optimized (Hpc6-Family): Optimized price/performance for large-scale HPC workloads.

Cost optimization is a critical aspect of EC2 instance selection. Use resources such as AWS Cost Explorer and EC2 Instance Comparison to assess resource costs and find affordable solutions that meet your workload needs. Organizations can achieve optimal performance and cost efficiency in cloud deployments by systematically analyzing workload demands and leveraging the diverse capabilities of available EC2 instance families. Remember, ongoing monitoring and optimization using tools like Amazon CloudWatch and AWS Trusted Advisor are crucial for maintaining resource alignment with evolving workload requirements.

NAVIGATING THE SHIFT: BEST PRACTICES AND CONSIDERATIONS FOR MIGRATING NON-CONTAINERIZED WORKLOADS TO NEW-GEN EC2 INSTANCES

A strong possibility for cost reduction and performance improvement exists when non-containerized workloads are moved to new-generation EC2 instances. However, A clear strategy guarantees a seamless transfer and reduces potential hiccups. This section equips businesses to negotiate this strategic transformation by outlining essential processes, best practices, and considerations for a successful migration.

A. Migration Framework:

[1]. Assessment and Planning:

Workload Profiling: Evaluate every workload in detail, noting dependencies, resource requirements, and performance bottlenecks.

Instance Selection: Based on the assessment, identify suitable new-gen instances considering factors like CPU, memory, network, and cost profile. Utilize AWS tools like Instance Advisor and Cost Explorer to aid selection.

Migration Strategy: Choose a migration strategy based on application complexity and downtime tolerance, such as lift-and-shift, partial migration, or staggered rollouts.

[2]. Preparation and Testing:

Compatibility Testing: Ensure application compatibility with the new-gen instance environment, including the operating system, drivers, and software dependencies. Conduct thorough testing in

non-production environments to identify and address potential issues. Test in lower (non-prod) environments first, then load test to ensure performance criteria are met.

Security Configuration: Apply appropriate security configurations and access controls to the new-gen instances, adhering to organizational security policies and compliance requirements. Leverage AWS Security Hub and Amazon Inspector for automated security assessments.

[3]. Migration Execution:

Pilot Migration: Begin by migrating a small subset of workloads in a pilot program to confirm the selected strategy and find any unexpected obstacles.

Phased Rollout: To reduce downtime and risk, implement a phased migration strategy that progressively migrates workloads into smaller groups. Use programs like AWS Migration Hub and AWS Application Autoscaling for orchestration and automation.

Post-Migration Validation: Ensure migrated workloads are tested and validated thoroughly in the new environment to guarantee security compliance, performance, and functionality. Use monitoring tools for continuous optimization and troubleshooting, such as AWS Trusted Advisor and Amazon CloudWatch.

Through a methodical approach, the utilization of pertinent tools and resources, and cautious handling of possible obstacles, enterprises can effectively shift their non-containerized workloads to the latest EC2 instances, resulting in substantial financial benefits, enhanced cloud efficiency, and performance gains.

BEYOND PERFORMANCE GAINS: NAVIGATING THE CHALLENGES OF CHANGING INSTANCE TYPES

While the allure of improved performance, cost, and features in new-generation EC2 instances is undeniable, migrating workloads come with their own hurdles. This section dives into the key challenges and mitigation strategies to ensure a smooth and successful transition.

A. DOWNTIME DURING THE SWITCH:

[1]. Challenge: Stopping instances for type changes leads to temporary inaccessibility.

[2]. Mitigation:

Communicate downtime clearly to stakeholders to avoid disruptions.

Utilize Infrastructure as Code (IaC) rollouts to minimize human error.

B. NETWORK CONNECTIVITY ISSUES:

[1]. Challenge: Public IPv4 addresses change upon instance type switch, potentially impacting internet accessibility.

[2]. Mitigation: Allocate and assign an Elastic IP address for persistent reachability.

C. RESOURCE AVAILABILITY CONSTRAINTS:

[1]. Challenge: Potential limitations in launching new-generation instances due to regional capacity or instance type limits.

[2]. Mitigation:

Proactively submit support requests for instance type limit increases if anticipating large-scale migration.

Be aware of regional availability limitations for specific instance types.

D. DATA INTEGRITY CONCERNS:

[1]. Challenge: While unlikely, data corruption or loss during migration is possible.

[2]. Mitigation: Create backups (AMIs) and volume snapshots for critical data before migration.

E. LINUX BOOT ISSUES WITH NITRO INSTANCES:

[1]. Challenge: Specific configuration issues can prevent booting on Nitro-based instances.

[2]. Mitigation:

Enable the ENA enaSupport attribute.

Install and load necessary ENA and NVMe modules.

Mount file systems using UUID/Label instead of the device name.

Run the NitroInstanceChecks script to identify and address further issues.

F. ADDITIONAL CONSIDERATIONS:

[1]. Application Compatibility: Test applications for compatibility with new instances. Consider software updates or containerization if needed.

[2]. Resource Planning: Carefully select instances and timing to ensure availability within desired regions and types. Utilize Reserved Instances for cost optimization and guaranteed capacity.

[3]. Downtime Minimization: Implement rolling deployments, minimize data transfer times, and leverage pre-configured instances to reduce downtime. Explore AWS services like Elastic Load Balancing and Auto Scaling for seamless traffic management.

Organizations may smoothly move to next-generation EC2 instances and enjoy the advantages of improved performance, cost-effectiveness, and cutting-edge capabilities by recognizing and proactively addressing these difficulties. Careful planning and execution are key to ensuring a smooth and successful migration.

UNLEASHING VALUE: REAL-WORLD SUCCESS STORIES OF NON-CONTAINERIZED WORKLOADS ON NEW-GEN EC2 INSTANCES

The theoretical advantages of migrating non-containerized workloads to new-gen EC2 instances translate into tangible benefits for organizations across diverse industries. This section showcases real-world case studies and testimonials, quantifying the cost savings and performance improvements achieved to empower businesses with practical insights.

A. SAMPLE CASE STUDY 1: COMPANY A TRANSFORMS LEGACY ANALYTICS PIPELINE WITH R5 INSTANCES:

[1]. **Company Profile:** Company A, a prominent e-commerce platform, needed help keeping up with the exponential expansion of client data in its on-premise data analytics pipeline. The relational database-based monolithic application encountered severe scalability issues and performance impediments that affected its ability to provide business insight and make decisions.

[2]. **Challenge:** The old infrastructure was unable to keep up with the expansion of data, resulting in sluggish reporting, delayed insights, and the formation of bottlenecks during busy shopping seasons. The on-premise solution's expensive maintenance and scalability costs made the difficulties worse.

[3]. **Solution:** Acknowledging the shortcomings of their current configuration, Company A strategically migrated to AWS. Attracted by the improved CPU performance, higher memory bandwidth, and faster networking capabilities, they chose R5 instances for their database and analytics pipeline. Leveraging Amazon RDS and Amazon Redshift, they achieved a fully managed and scalable solution.Results: Following the migration, Company A witnessed a remarkable transformation. Query execution times decreased by 40%, significantly improving report generation and data analysis workflows. Additionally, the elastic nature of R5 instances enabled seamless scaling during peak periods, eliminating performance bottlenecks. Most importantly, the migration resulted in a 25% reduction in overall infrastructure costs due to optimized instance type and size and improved resource utilization.

B. SAMPLE CASE STUDY 2: COMPANY B OPTIMIZES HPC WORKLOADS WITH M5A INSTANCES AND AMD EPYC PROCESSORS:

[1]. **Company Profile:** Company B, a distinguished research institute, values high-performance computing (HPC) for its sophisticated scientific modeling and simulations. However, its legacy HPC cluster, based on older-generation Intel-based processors, could have improved its research progress due to inherent limitations concerning processing power, memory capacity, and overall cost-effectiveness.

[2]. **Challenge:** Longer execution times and lower throughput resulted from the previous HPC cluster's inability to manage simulations, which became more complicated. A substantial financial strain was also created by the high cost of scaling and maintaining the infrastructure.

[3]. **Solution:** Company B evaluated several options to address these challenges and migrated to AWS M5a instances powered by AMD EPYC processors. These instances offered superior core performance, larger memory configurations, and a significant price-performance advantage compared to Intel counterparts. Utilizing Amazon EC2 Auto Scaling, they ensured dynamic resource allocation based on workload demands.

[4]. **Results:** The migration to M5a instances delivered impressive results. Simulation execution times were reduced by an average of 30%, enabling researchers to conduct more simulations and accelerate scientific discovery—moreover, the improved memory capacity allowed for handling larger datasets and more complex models. Interestingly, switching to AMD EPYC processors allowed for required performance levels to be met or exceeded while reducing HPC infrastructure expenses by 20%.

These real-world scenarios accurately illustrate the concrete advantages of moving non-containerized workloads to new-generation EC2 instances. With enhanced scalability and performance and substantial cost reductions, these solutions enable businesses to meet their cloud optimization objectives and open up fresh avenues for development and innovation.

CONCLUSION

Innovative solutions are required to address the shortcomings of conventional methods for managing non-containerized workloads in the cloud. This article has explored the transformative potential of leveraging new-

generation EC2 instances (C5, R5, M5 families) for these workloads, unveiling a compelling path toward cost optimization and enhanced efficiency.

Key Takeaways:

[1]. The rising prevalence of non-containerized workloads presents a significant cost management challenge within cloud deployments.
[2]. Traditional management approaches often need to be revised. They limit resource utilization and fail to address the unique requirements of non-containerized systems.
[3]. New-gen EC2 instances offer a game-changing paradigm, boasting advancements in performance, pricing, and features specifically suited for these workloads.
[4]. New-generation instances are an appealing option for optimization due to their quantifiable gains in CPU power, memory bandwidth, network performance, and affordability.
[5]. Effective real-world case studies show the advantages of increased scalability, cost reductions, and performance gains.

Embracing new-generation EC2 instances offers a strategic opportunity to optimize non-containerized applications as cloud architects and engineers negotiate an ever-changing market. Organizations may realize significant cost savings, enhance resource usage, and eventually develop a more sustainable and economical cloud strategy by closely evaluating workload needs, utilizing tools and resources that are readily available, and implementing a well-defined migration strategy.

We encourage businesses to explore the potential of new-gen EC2 instances for their non-containerized deployments. By harnessing the advancements offered by these innovative solutions, organizations can empower themselves to optimize costs and unlock new levels of performance, scalability, and agility within the AWS cloud ecosystem.

REFERENCES

[1]. AWS Documentation. "Getting Started with EC2 Auto Scaling." [Online]. Available: https://docs.aws.amazon.com/autoscaling/ec2/userguide/get-started-with-ec2-auto-scaling.html, Accessed June, 2022.
[2]. AWS Documentation. "Amazon EC2 Instance-Types." [Online]. Available: https://www.amazonaws.cn/en/ec2/instance-types/, Accessed June 2022.
[3]. AWS Documentation., "AWS and AMD." [Online]. Available: https://aws.amazon.com/ec2/amd/, Accessed June, 2022.
[4]. AWS Documentation. "Change the launch configuration for an Auto Scaling group." [Online]. Available: https://docs.aws.amazon.com/autoscaling/ec2/userguide/change-launch-config.html, Accessed June 2022.
[5]. AWS Documentation. "Previous Generation Instances." [Online]. Available: https://aws.amazon.com/ec2/previous-generation/, Accessed June 2022.
[6]. RightScale. (2019). "State of the Cloud Report." Flexera. [Online]. Available: https://resources.flexera.com/web/media/documents/rightscale-2019-state-of-the-cloud-report-from-flexera.pdf
[7]. Tan, T. (2022, May 20). "AWS EC2 Instances Comparison using Sysbench — t1 vs t2 vs t3 vs t3a vs t4g." Medium Blog. [Online]. Available: https://faun.pub/aws-ec2-instances-comparison-using-sysbench-t1-vs-t2-vs-t3-vs-t3a-vs-t4g-9540a221d563.
[8]. Tan, T. (2022, May 2). "AWS EC2 (Memory Optimized) Instances Comparison — R3 vs R4 vs R5x vs R6x." Medium Blog. [Online]. Available: https://faun.pub/aws-ec2-memory-optimized-instances-comparison-r3-vs-r4-vs-r5x-vs-r6x-1f763d1eb329