



스팟인스턴스를 통해 EKS 요금 절약하기

GloZ 데브옵스 엔지니어 양승만

오늘 발표 난이도

100 ~ 200

저는요

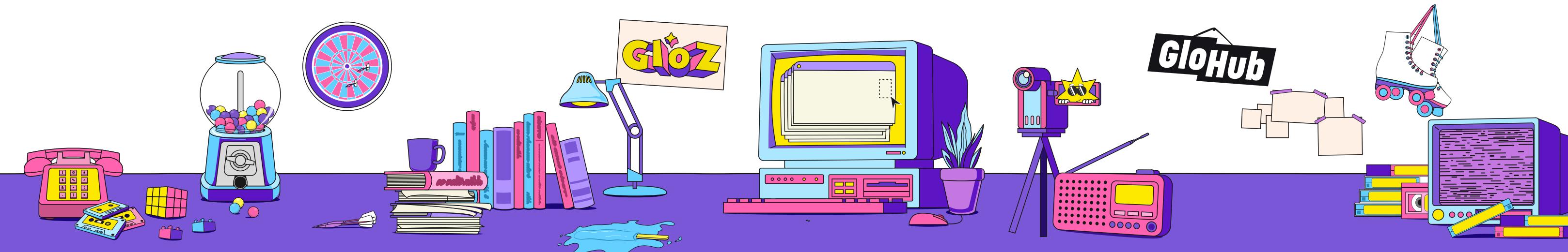
- 유닉스(AIX) / 리눅스 HA 엔지니어 (2018.01 ~ 2021.11)
- 온프레미스 환경의 인프라 관리와 운영 (2018.01 ~ 2021.11)
- GloZ 데브옵스 엔지니어(2021.12 ~ 현재)



글로지 비전

80억 인구가 GloZ로 소통하는 세상!

세계 누구나 언어의 장벽없이 콘텐츠를 즐길 수 있도록 소통하는 세상을 만드는 것입니다.



글로지 서비스

글로지는 콘텐츠들을 고객과 함께 현지화하고 있습니다.



GloZ Enterprise

최상의 번역작가님들과 협업하여 최상의 퀄리티의 품질을 제공하는 글로벌 현지화 서비스



GloHub Creator

크리에이터와 시청자가 많이 유입되는 뉴미디어 시장에 맞춘 가성비 높은 글로벌 현지화 서비스



GloHub Pro

커뮤니티 형성, 작업 툴제공, 커리어 성장까지 번역작가들을 위한 원스톱 플랫폼 서비스

글로지 서비스

GloZ Enterprise

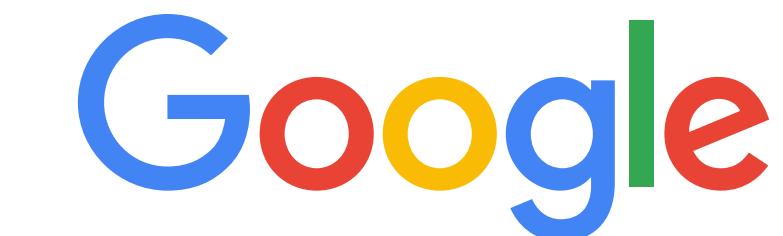
세계 곳곳의 최상의 번역가들을 발굴하고 관리하여 최상의 퀄리티를 제공합니다.

61개 이상의 언어로 현지화 서비스가 가능하며

글로지만의 KPI 시스템으로 최고의 번역가들과 QCer들의 퀄리티 높은 글로벌 현지화 서비스입니다.

고객과의 의사소통과 빠른 피드백을 수용하기 위해 전문 전담 팀이 배정되며

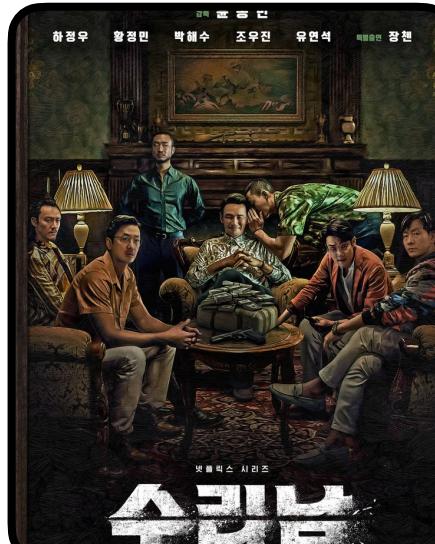
넷플릭스, 아마존, 구글, NHN 등 풍부한 경험을 바탕으로 최상의 서비스를 제공합니다.



글로지 대표작

콘텐츠 현지화 산업에서 다양한 콘텐츠 분야와 글로벌 기업들과 협업하며, 성장하고 있습니다.

OTT.영화



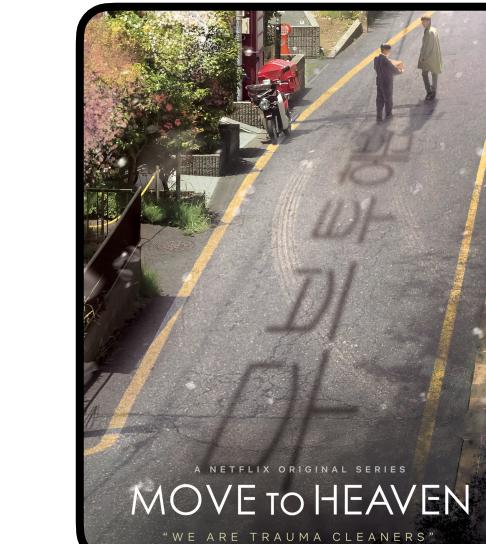
수리남



서울대작전



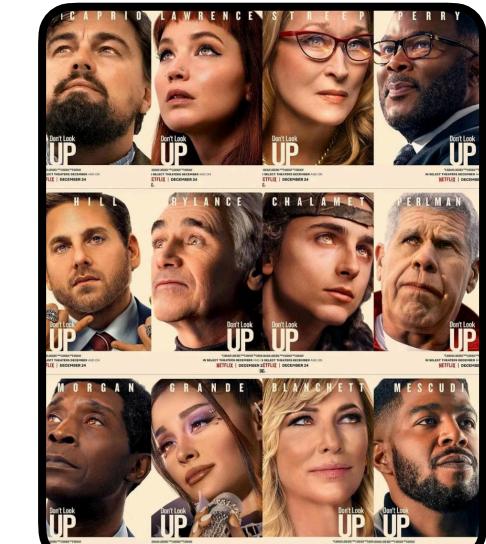
응답하라 1988



무브 투 헤븐



기묘한 이야기



돈 룩업

웹툰, 웹소설



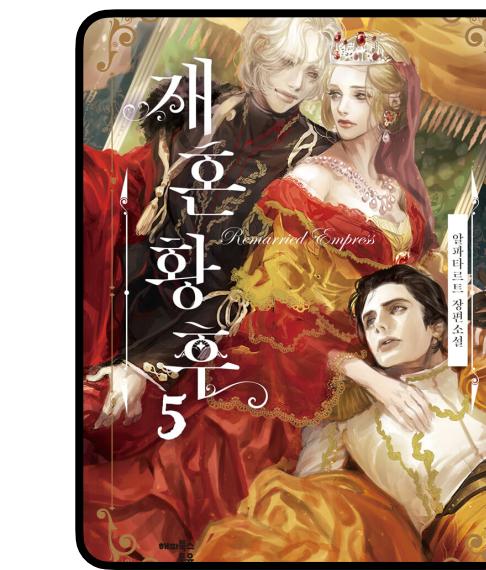
세븐 페이츠 : 차코



다크 문 : 달의 제단



별을 쫓는 소년들



재혼황후

유튜브 영상 번역



선미



뱀뱀



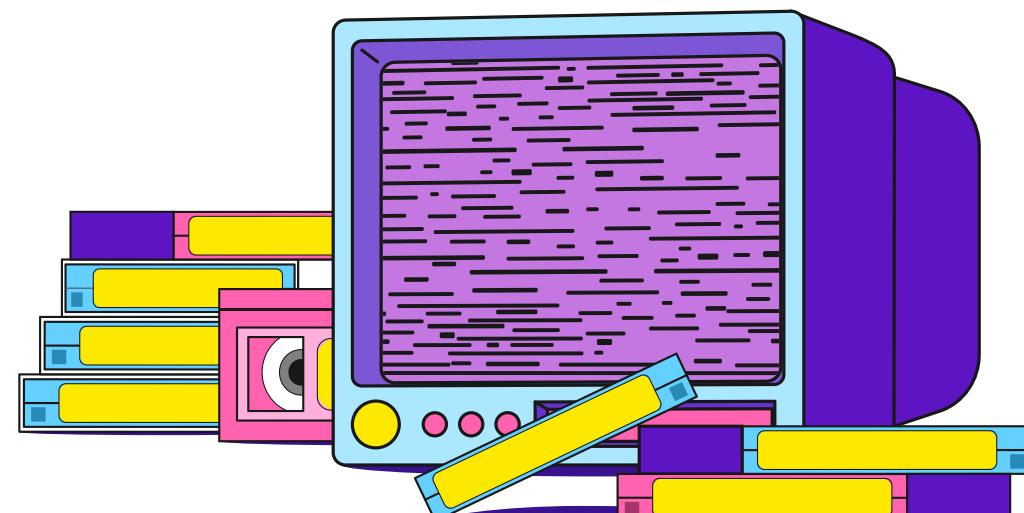
샌드박스 게임즈



담원기아



JTBC
KCON VOYAGE





- Amazon Elastic Kubernetes Service(EKS)를 사용해서 딥러닝 시스템을 AWS 위에서 실행 할 수 있도록 환경 구축.
- 환경을 구성할 때, worker node를 p3.2xlarge 인스턴스로 생성해서 사용함.
- 여러개의 worker node 생성 후 일주일 방치.



- Amazon Elastic Kubernetes Service(EKS)를 사용해서 딥러닝 시스템을 AWS 위에서 실행 할 수 있도록 환경 구축.
- 환경을 구성할 때, worker node를 p3.2xlarge 인스턴스로 생성해서 사용함.
- 여러개의 worker node 생성 후 일주일 방치.



- Amazon Elastic Kubernetes Service(EKS)를 사용해서 딥러닝 시스템을 AWS 위에서 실행 할 수 있도록 환경 구축.
- 환경을 구성할 때, worker node를 p3.2xlarge 인스턴스로 생성해서 사용함.
- 여러개의 worker node 생성 후 일주일 방치.



여기가 맞는 게시판인지 모르겠네요;; 저는 개발자 이구요.. AWS 계정하나 터서 이걸로 공부하고 있거든요. 근데 3월 말에 갑자기 제 계정을 닫았다고 메일이 오더라구요. 이게 뭐지 하면서 로그인 해보니까 계정은 닫혀있고 썬한 기분에 청구서 페이지 [\[↗\]](#)를 확인 해보니까...아.. 26만 달러가 청구가 되어 있었습니다..

ㅠㅠ

네이버 환율 돌리는데 3억 2천 나오는데 진짜 잘못본줄 알고 몇번을 확인했습니다..
첨엔 현실성이 없어서 구글링부터 했는데 저 같은 상황이 많더라고요. 해커들이 해킹해서 주로 코인 채굴하나봐요.....

1,437.0300

↑20.17%

+241.2300 1년

10월 11일, 오전 4시 54분 25초 UTC · 면책조항

1일

5일

1개월

6개월

YTD

1년

5년

최대

1,450

1,400

1,350

1,300

1,250

1,200

1,150

2021년 11월

2022년 1월

2022년 3월

2022년 5월

2022년 7월

2022년 9월

1,437.0300
2022. 10. 11.



1,437.0300

↑20.17%

+241.2300 1년

10월 11일, 오전 4시 54분 25초 UTC · 면책조항

1일

5일

1개월

6개월

YTD

1년

5년

최대

1,450

1,400

1,350

1,300

1,250

1,200

1,150

2021년 11월

2022년 1월

2022년 3월

2022년 5월

2022년 7월

2022년 9월

1,437.0300
2022. 10. 11.



지금 인프라 그냥 놔둬도 20%



왜 이렇게 많이 나올까?

좀 더 깎을 수는 없을까?

한사람 연봉을 더 내야하네?

지금 인프라 그냥 놔둬도 20%



비용절감 KPI에 포함?

이것보다 적게 낼 수 있다던데...

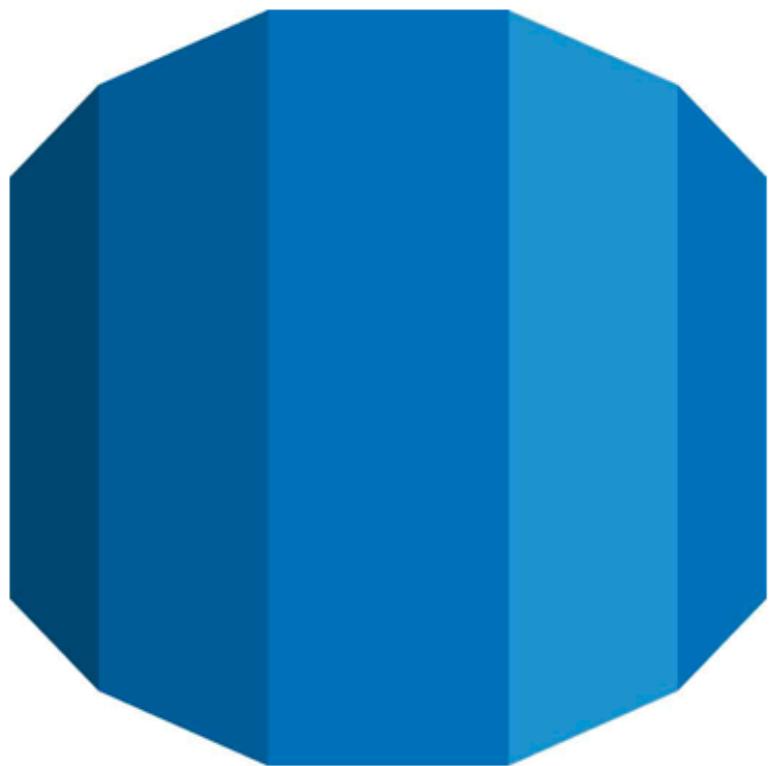
FinOps

“FinOps is an evolving cloud financial management discipline and cultural practice that enables organizations to get maximum business value by helping engineering, finance, technology and business teams to collaborate on data-driven spending decisions.”

* What is FinOps?

<https://www.finops.org/introduction/what-is-finops/>

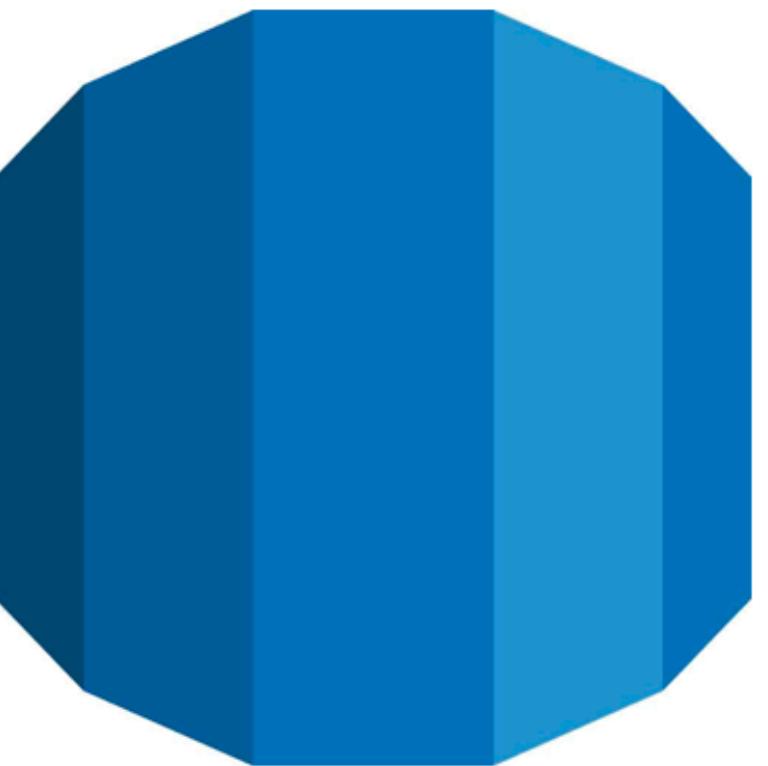
DevFinOps



Amazon RDS



amazon
DynamoDB



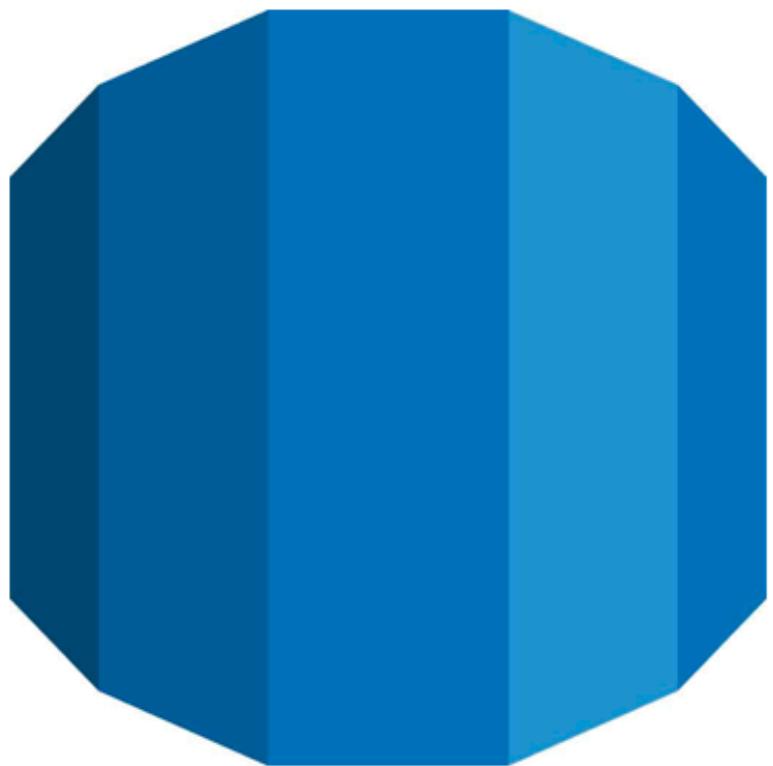
Amazon RDS



amazon
DynamoDB



Amazon EKS



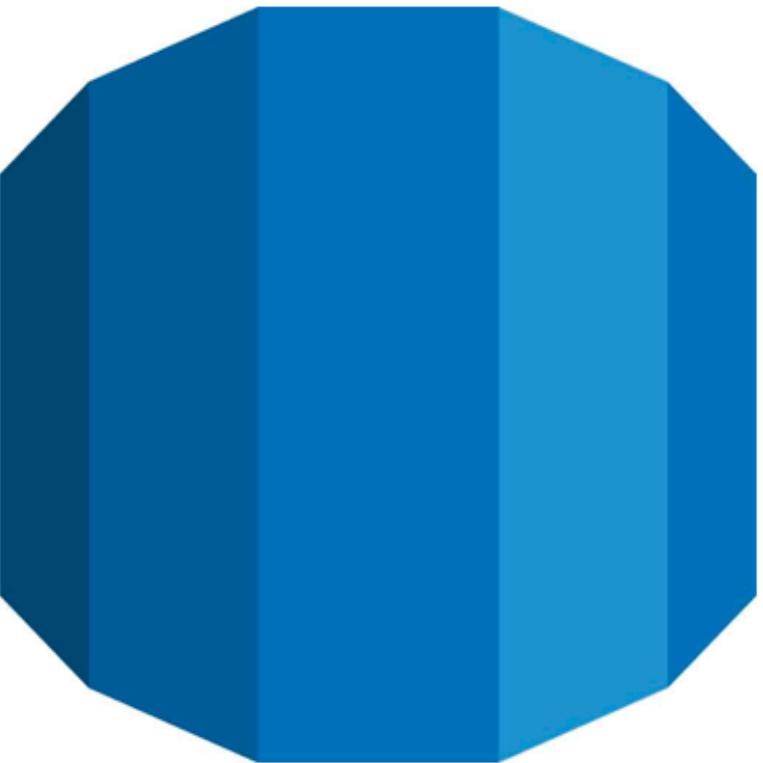
Amazon **RDS**



amazon
DynamoDB



Amazon EKS

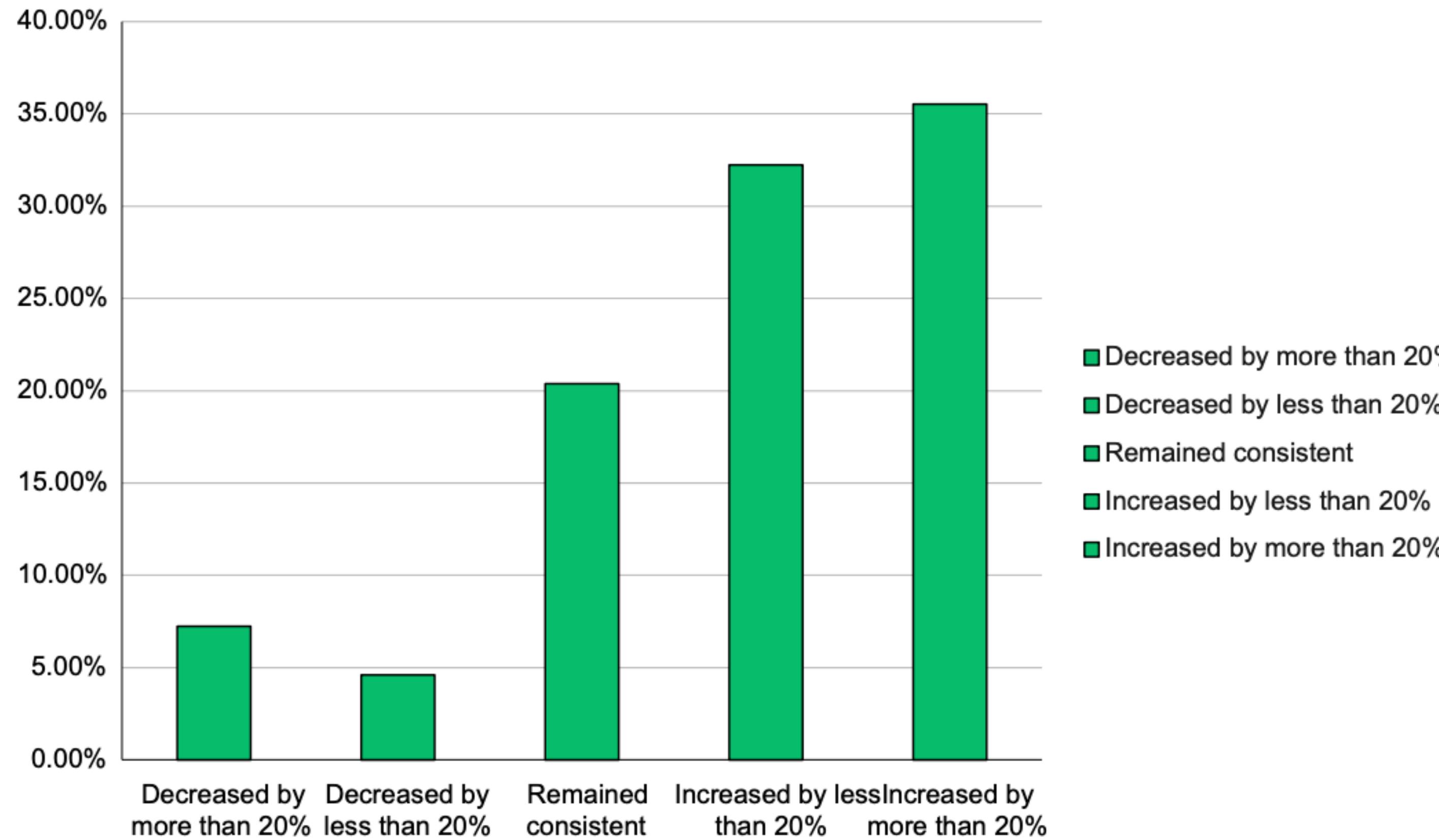


Amazon RDS



Amazon EC2

How much has your Kubernetes-related spend grown in the last 12 months?



* 2021 May CNCF FinOps Report

https://www.cncf.io/wp-content/uploads/2021/06/FINOPS_Kubernetes_Report.pdf

돈먹는 하마 EKS 편

FinOps 를 위한 툴은 없을까?

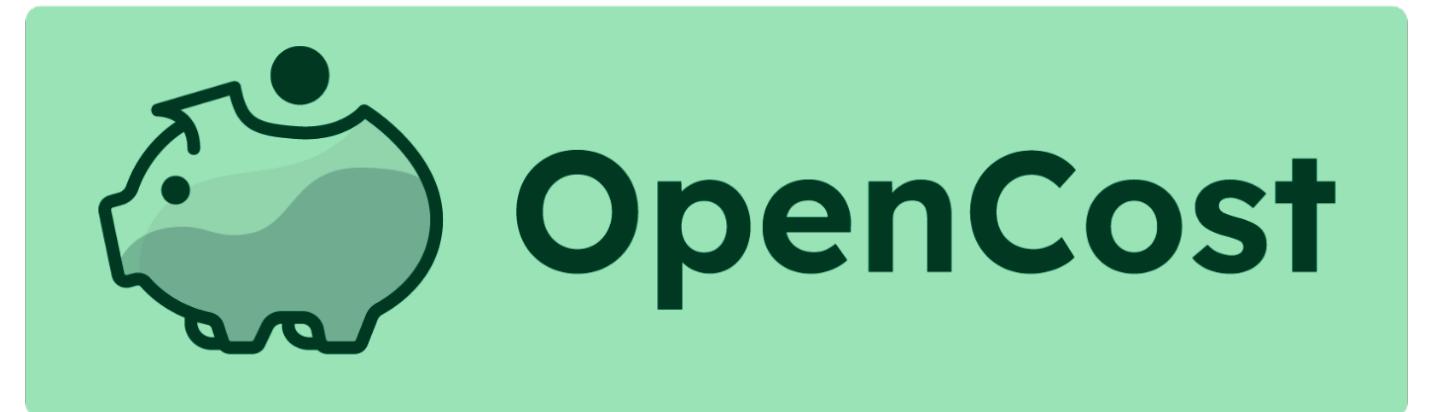
K8S를 위한 FinOps tool



K8S를 위한 FinOps tool



얘도 있어요 '공짜' →



그래서, 쓸만한가요?

- 헬름차트 이용한 간단한 설치
- 가시성 - 실시간 비용정보 제공
 - 월별 클러스터 비용
 - 네임스페이스, 디플로이먼트, 서비스 비용
 - CPU, Memory, Network, PV 사용률 별 비용 모니터링
- 비용 최적화 인사이트 제공

***30%**

Cost Allocation

Assets

Savings

Health

Reports

Alerts

US\$119.23

Monthly savings identified

[VIEW SAVINGS >](#)**US\$185.67**

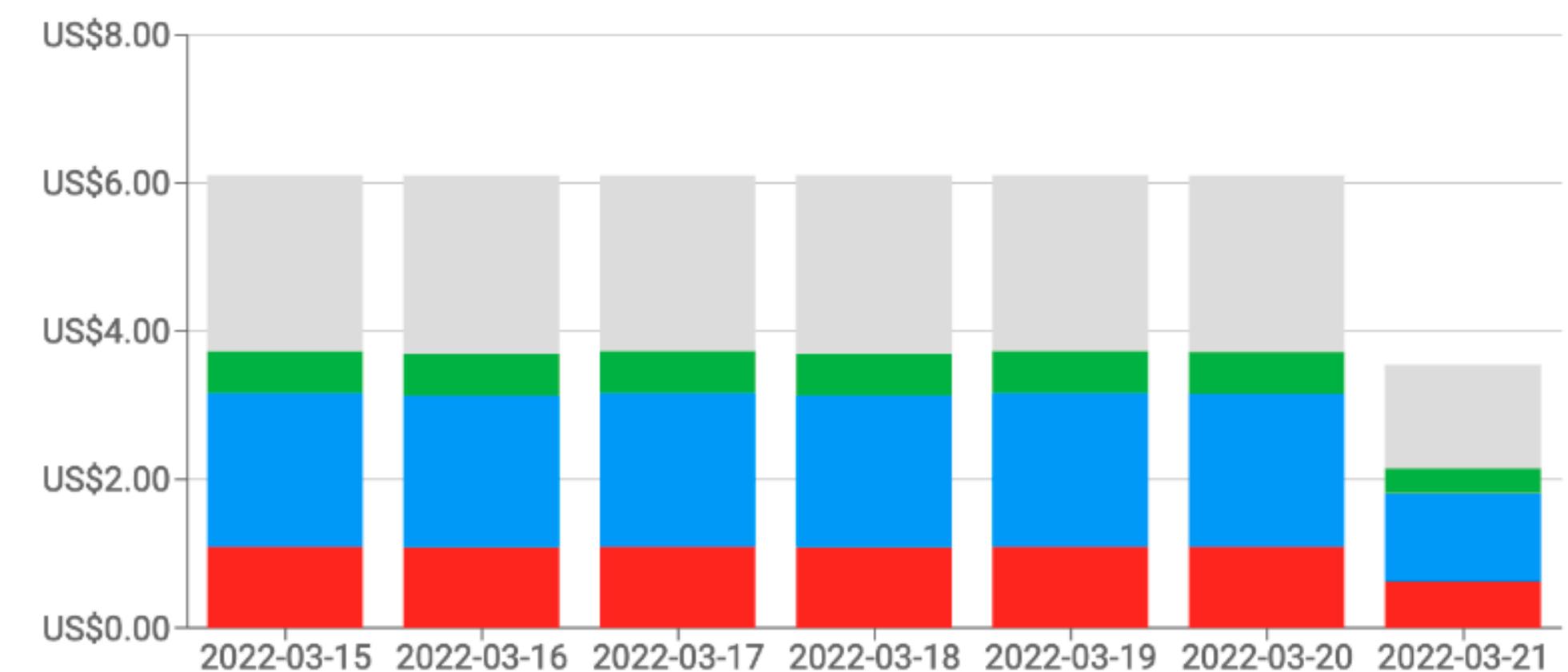
Monthly Kubernetes costs

[MONTHLY COSTS >](#)**9.2%**

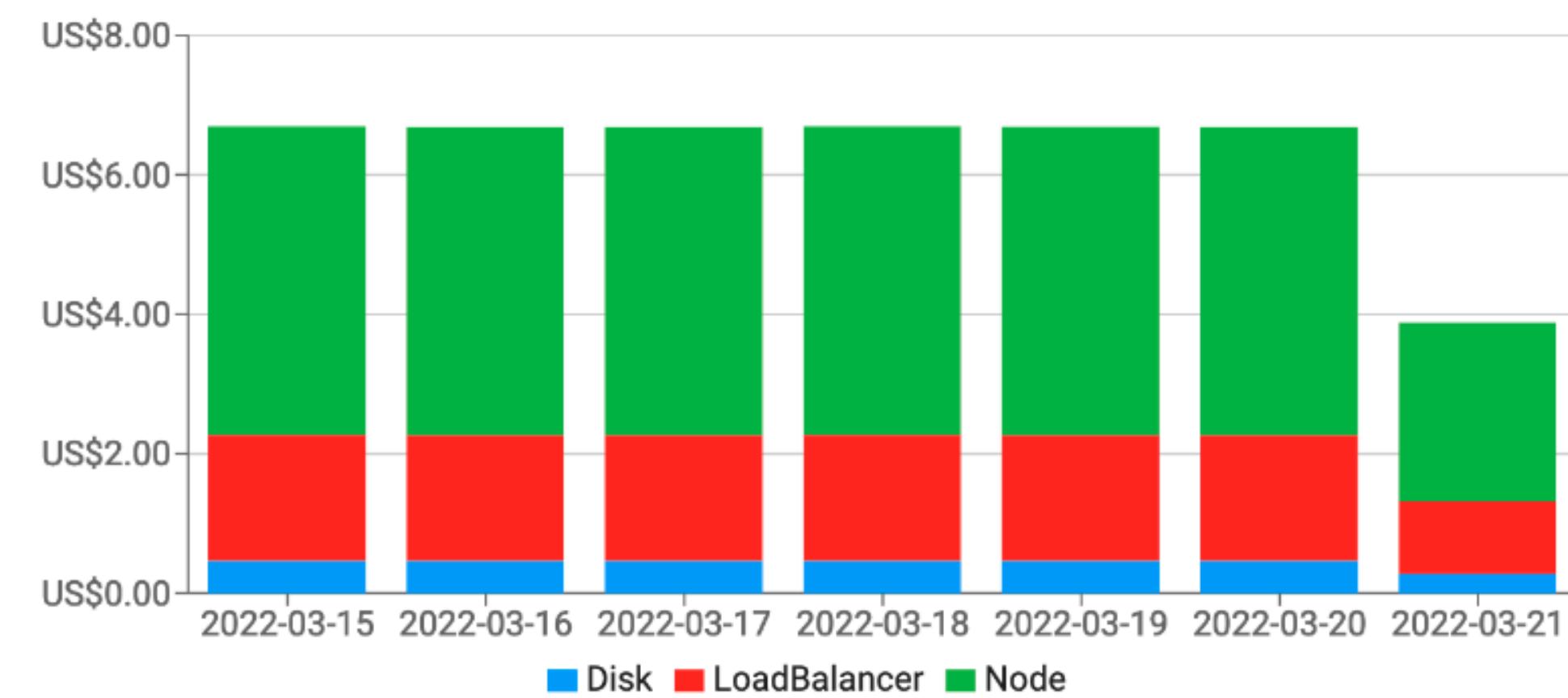
Cost efficiency

[COST EFFICIENCY >](#)**Cost Allocation**

Cumulative costs per namespace over the last 7 days

[COST ALLOCATION >](#)**Kubernetes Asset Costs**

Cumulative costs per Kubernetes asset type over the last 7 days

[ASSETS >](#)**Cluster Efficiency**

Daily usage, allocation, and idle resources over the last 7 days

US\$8.00

Switch cluster
Azure Cluster #1

Settings

Allocation

\$ Savings

❗ Health

🔔 Notifications

Total measured cost

Top results measured over selected time window



Namespace	Mem Cost	CPU Cost	Network Cost	PV Cost	GPU Cost	External Cost	Total Cost
kube-system	\$0.00	\$0.04	Add network	\$0.00	\$0.00	\$0.00	\$0.04
gloo-system	\$0.00	\$0.02	Add network	\$0.00	\$0.00	\$0.00	\$0.02
kubecost	\$0.00	\$0.01	Add network	\$0.00	\$0.00	\$0.00	\$0.01
default	\$0.00	\$0.00	Add network	\$0.00	\$0.00	\$0.00	\$0.00
gremlin	\$0.00	\$0.00	Add network	\$0.00	\$0.00	\$0.00	\$0.00

비용 최적화 인사이트

\$149.63
Estimated Savings ⓘ

💡 Introducing the new Kubecost Spot readiness guide

Pods with over-provisioned requests	\$106.44	>
Manage underutilized nodes	\$70.18	>
Make reserved instance commitments	\$50.11	>
Potential abandoned workloads identified	\$3.47	>
Manage unclaimed volumes	\$0.00	>
Local disks with low utilization found	\$0.00	>
Cluster nodes can be right-sized	\$0.00	>

비용 최적화 인사이트 - 노드 조정

Development 

Show advanced metrics

Requirements for development clusters include a minimum of one node and a target resource utilization of 85% at 85th-percentile allocation over the last 2 days.

Recommendation: Complex 

\$59.65/mo.

Compared to now	\$29.59/mo. (33.2%) savings
Node count	3
CPU	2.7 VCPUs (83.1% utilized)
RAM	4.6 GB (69.1% utilized)

1 n2-highcpu-2 

1 f1-micro 

1 e2-small 

Recommendation: Simple 

\$72.22/mo.

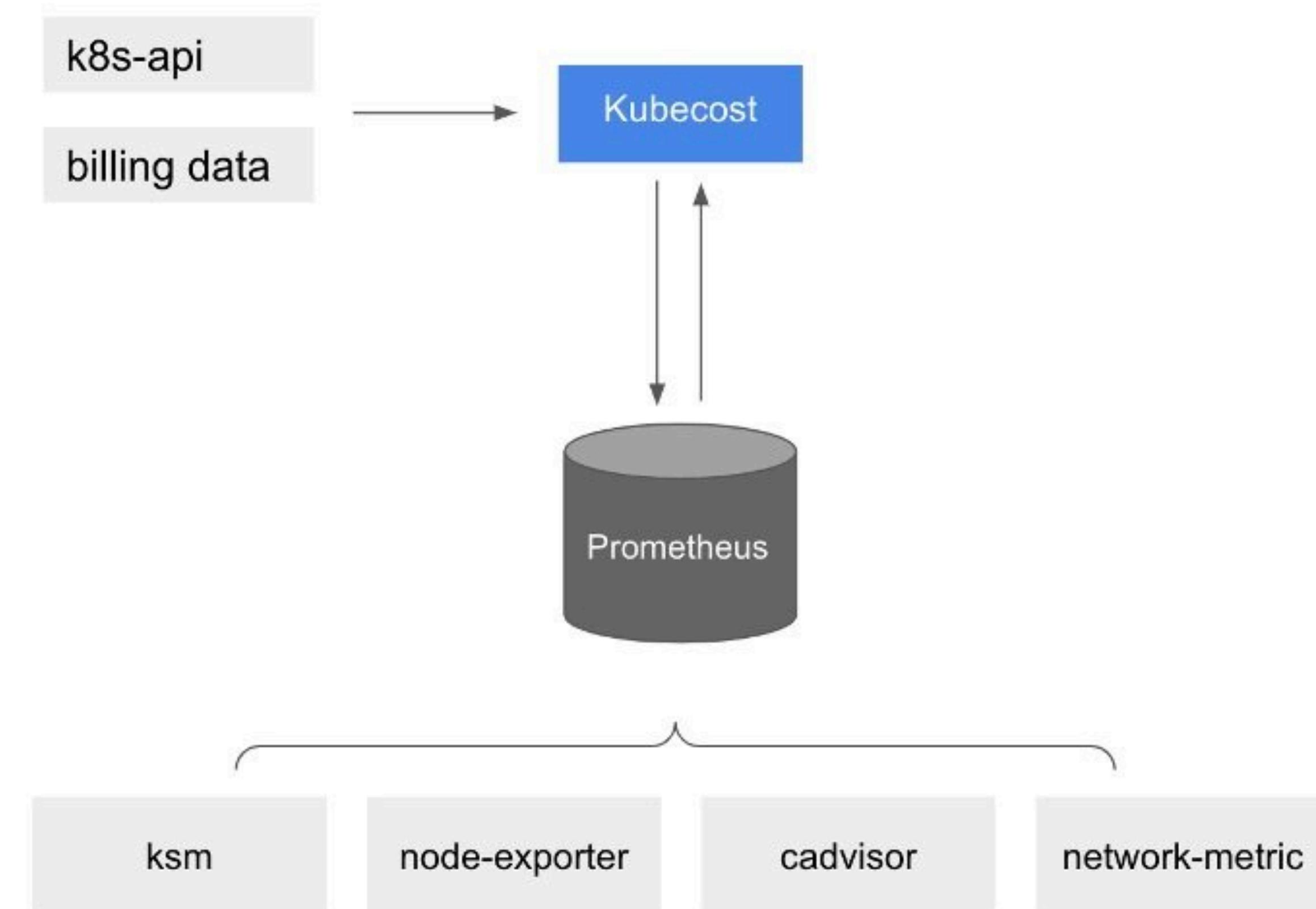
Compared to now	\$17.01/mo. (19.1%) savings
Node count	1
CPU	4 VCPUs (51% utilized)
RAM	4 GB (63.9% utilized)

1 e2-highcpu-4 

비용 최적화 인사이트 - 파드별

Summary							\$118.98/mo	
Resource	Requested	Usage	Under-provisioning ⓘ			Over-provisioning ⓘ		Savings
CPU	5.41	856.7m	365.3m			4.25		\$106.15/mo
RAM	11.7 GiB	9.8 GiB	7.3 GiB			3.9 GiB		\$12.83/mo
Container	Cluster	CPU usage	CPU request	CPU recomm'd	RAM usage	RAM request	RAM recomm'd	Efficiency ↓ Savings
istio-system/istiod:discovery	kubernetes-homelab/cluster-one	6.8m	500m	11m	193.8 MiB	2 GiB	298.1 MiB	2.8% \$16.52/mo
monitoring/grafana:grafana	kubernetes-homelab/cluster-one	8.5m	500m	14m	74.8 MiB	1 GiB	115.1 MiB	2.2% \$13.92/mo
metallb-system/speaker:speaker	kubernetes-homelab/cluster-one	8m	99.7m	13m	39.9 MiB	99.7 MiB	61.3 MiB	7.1% \$12.71/mo
wordpress/wordpress:wordpress	kubernetes-homelab/cluster-one	7.2m	300m	12m	169.3 MiB	512 MiB	260.4 MiB	5.8% \$7.38/mo
monitoring/x509-certificate-exporter-nodes:istio-proxy	kubernetes-homelab/cluster-one	3.6m	100m	10m	77.5 MiB	128 MiB	119.2 MiB	8.2% \$6.29/mo

Kubecost 아키텍처



* kubecost architecture

<https://guide.kubecost.com/hc/en-us/articles/4407595922711-Kubecost-Core-Architecture-Overview>

Kubecost 추천 비용절감

파드의 사이즈를 작게

Cpu, mem require 사용률에 근거하여 수정하기

- 사용률을 적당히, 넉넉히 잡는다면 kubecost 추천값으로 파드 리소스 조정 권장
- HPA를 사용하여 필요 시 추가 파드를 사용할 수 있게 컨피그하여 비용 절감

노드 아키텍처 전환

x86에서 arm으로

- 퍼포먼스도 좋은데 가격까지 약 20% 저렴!
- 오히려 좋은데 안 쓸 이유가 있을까?

노드 아키텍처 전환

x86에서 arm으로

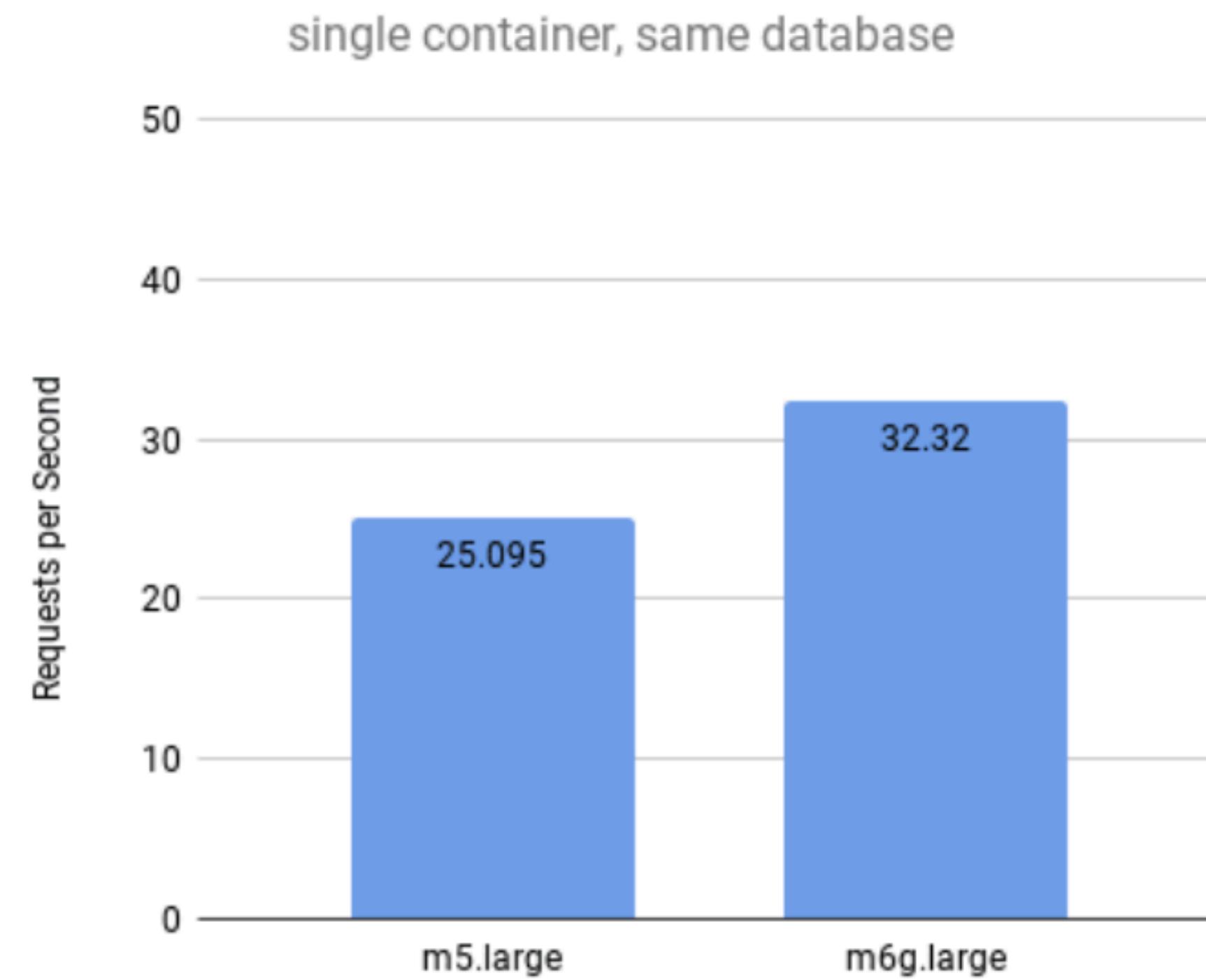
	Intel	ARM
Instance Count	1	1
Instance Type	m5.large	m6g.large
CPU Tested	Intel(R) Xeon(R) Platinum 8259CL CPU @ 2.50GHz	Custom built AWS Graviton2 Processor with 64-bit Arm Neoverse cores
CPU Count	2	2
Clock (MHz)	3397.426	N/A
Region/AZ	ap-southeast-2c	ap-southeast-2c
Price (Sydney)	\$0.120 per Hour	\$0.096 per Hour

* ec2 architecture price battlecard

<https://dnx.solutions/arm-vs-intel-a-real-world-comparison-using-ec2/>

노드 아키텍처 전환

x86에서 arm으로



Graviton2 is almost 29% faster when shooting requests to a single container and the database behind is exactly the same.

노드 아키텍처 전환 - 고려할 점

x86에서 arm으로

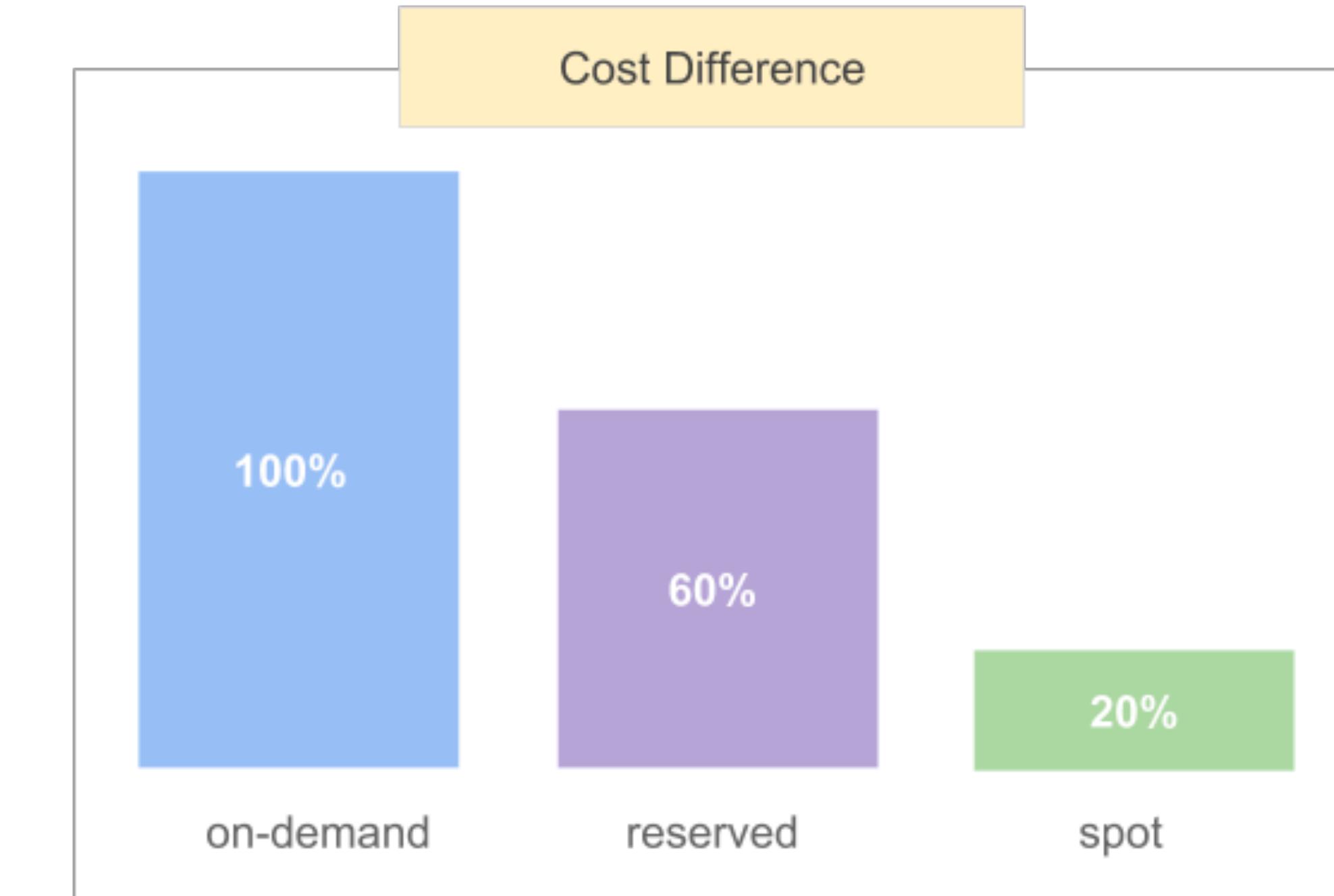
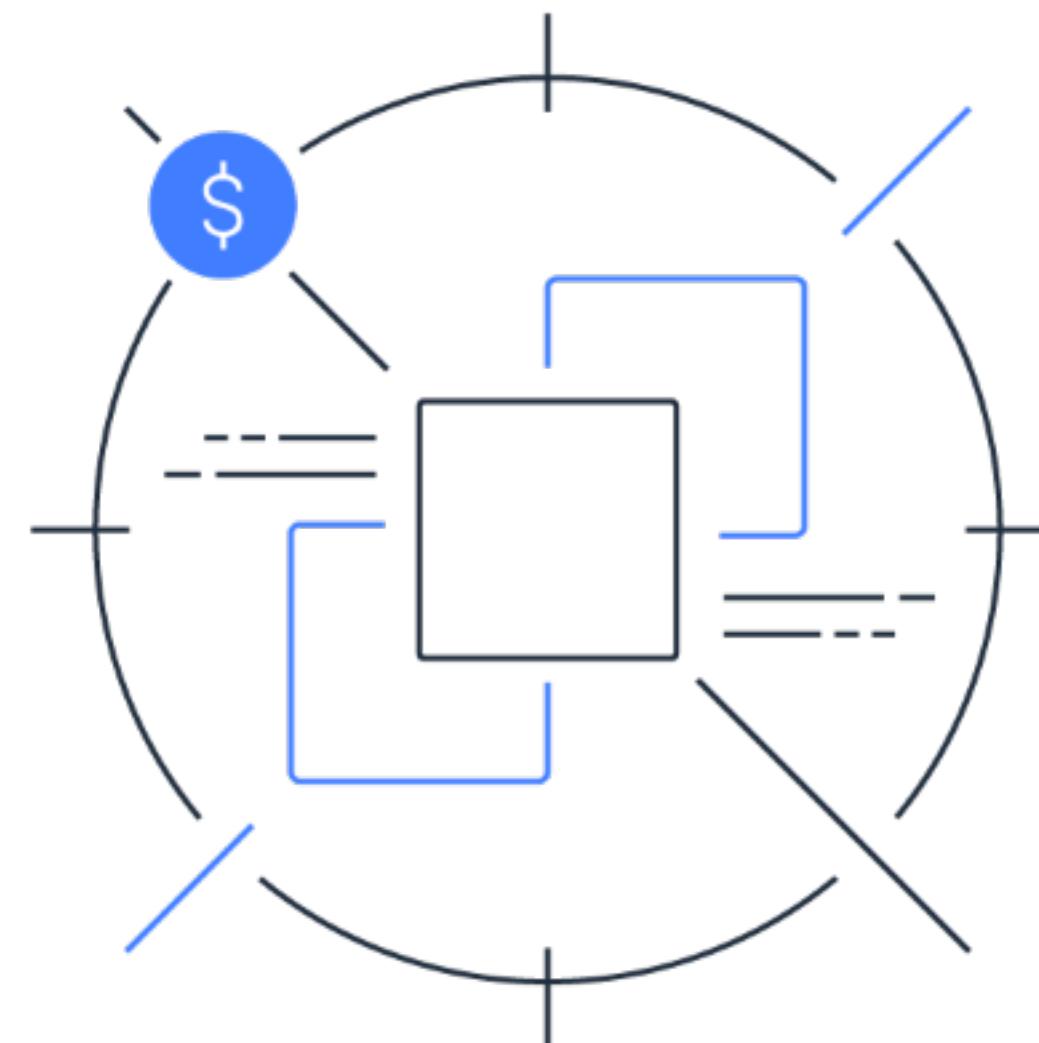
- 회사마다 꿈 같은 이야기일 수도
- 어플리케이션이 arm에서 빌드와 실행에 문제가 없는지?
- 특정 라이브러리에 아키텍처 종속성이 있어서 문제 발생 소지가 있음
- 원활한 빌드와 실행이 되는 서비스부터 충분한 검증 후 마이그레이션

온디맨드 노드는 RI로 전환

계약기간별로 요금할인은 상이

- 계약 기간동안 할인된 비용에 인스턴스 사용 가능
- 온디맨드에 비해 40% 가량 저렴한 가격

Spot Instances on EKS



Spot Instances on EKS

C4	AZ1	AZ2	AZ3	On-Demand
8XL	\$0.28	\$0.27	\$0.29	\$1.76
4XL	\$0.21	\$0.19	\$0.16	\$0.88
2XL	\$0.08	\$0.07	\$0.08	\$0.44
XL	\$0.04	\$0.05	\$0.04	\$0.22
L	\$0.01	\$0.01	\$0.02	\$0.11

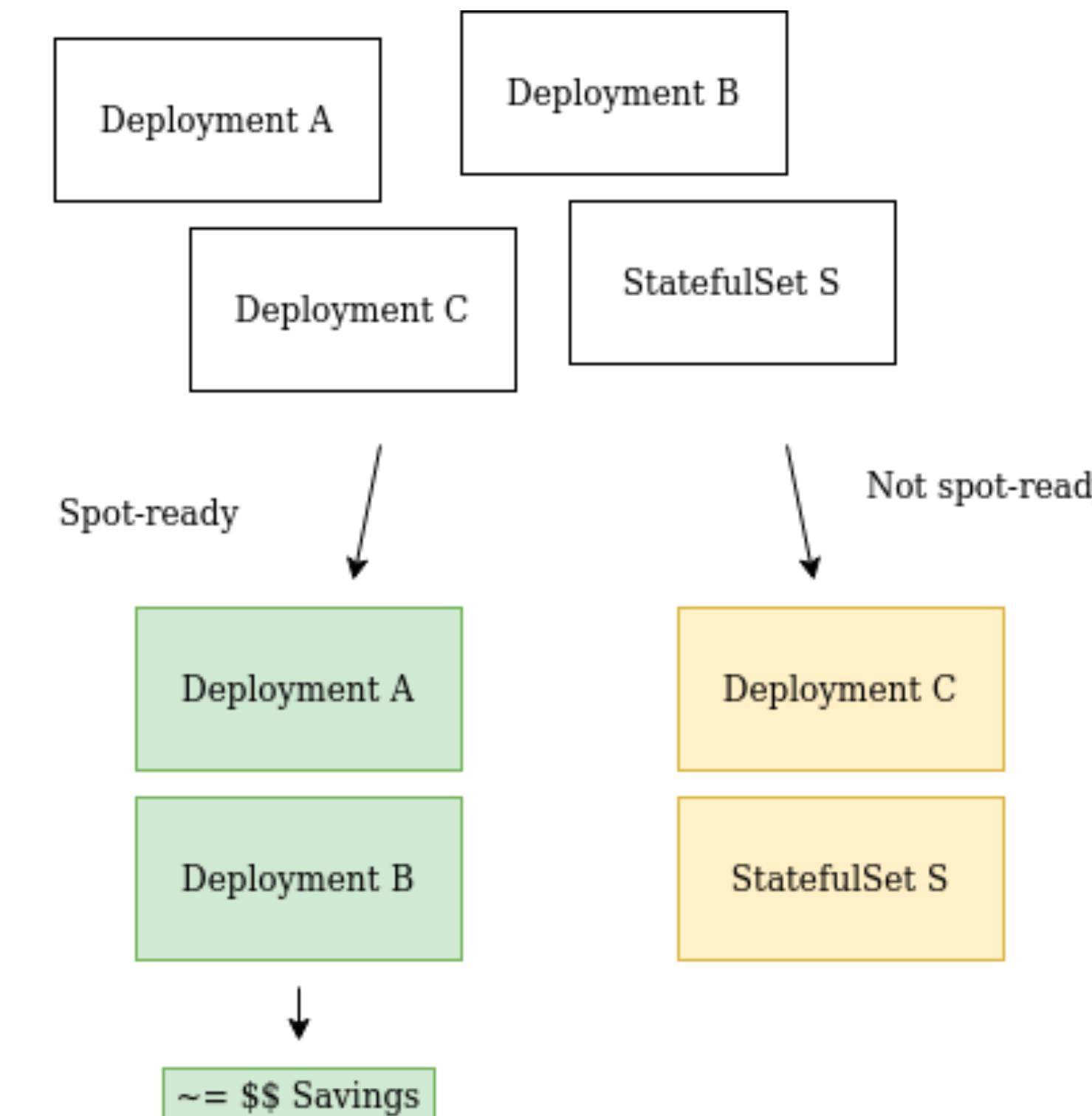
EKS에서 스팟인스턴스 좋긴한데 불안해!

- 예고없이 죽는 스팟인스턴스, EKS 노드로 사용해도 될까?
- 파드가 갑자기 죽어도 괜찮은지 모르겠네?

체크리스트

내 서비스가 스팟인스턴스로 가도 될까?

Spot-readiness checklist by kubecost



* kubecost spot checklist

<https://guide.kubecost.com/hc/en-us/articles/4407595980311-Spot-Checklist>

내 서비스가 스팟인스턴스로 가도 될까?

Spot-readiness checklist by kubecost

Overall Readiness	Controller	Controller Type	Replicas	Local Storage	Controller PDB	Deployment Rolling Update	Namespace Annotation Override	Controller Annotation Override	Monthly Rate	Monthly Spot Rate	Monthly Savings	Savings Pct
✓	guestbook-with-pdb: frontend	✓	✓ 6	✓	✓ 3/6	✓ 5/6	✗	✗	11.72	3.12	8.60	73.37%
?	guestbook: frontend	✓	✓ 3	✓	✗	✗ 3/3	✗	✗	5.86	1.56	4.30	73.37%
?	guestbook: redis-follower	✓	✓ 2	✓	✗	✗ 2/2	✗	✗	2.54	0.68	1.86	73.37%
?	guestbook-with-pdb: redis-follower	✓	✓ 2	✓	✗	✗ 2/2	✗	✗	2.28	0.61	1.67	73.37%
?	kube-system: kube-dns	✓	✓ 2	✓	✗	✗ 2/2	✗	✗	9.52	2.53	6.98	73.37%
!	kube-system: stackdriver-metadata-agent-cluster-level	✓	✗ 1	✓	✗	✓ 0/1	✗	✗	2.15	0.57	1.58	73.37%

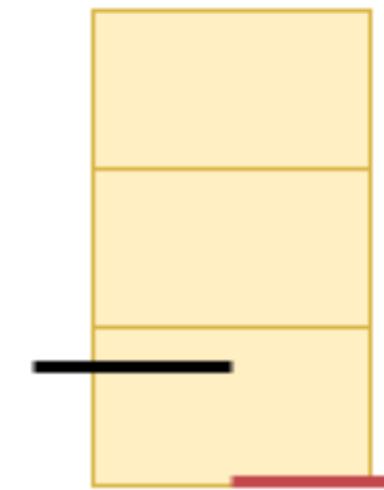
내 서비스가 스팟인스턴스로 가도 될까?

Spot-readiness checklist by kubecost

- Controller-type; Deployment 권장(stateless)
- Replica Count; 2개 이상 권장
- Local Storage; emptyDir이 없어야 함

+ Rolling update strategy(available ratio)

3 Replicas
25% max unavailable

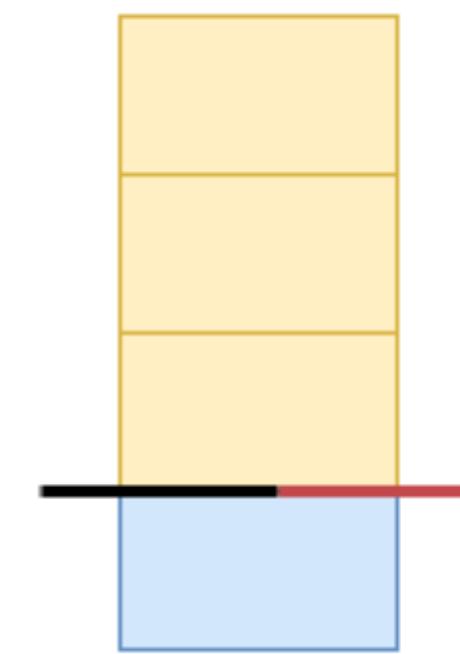


Equivalent to min
available = $3 / 3 = 1.0$



Does not meet
threshold of min
available ratio ≤ 0.9

4 Replicas
25% max unavailable

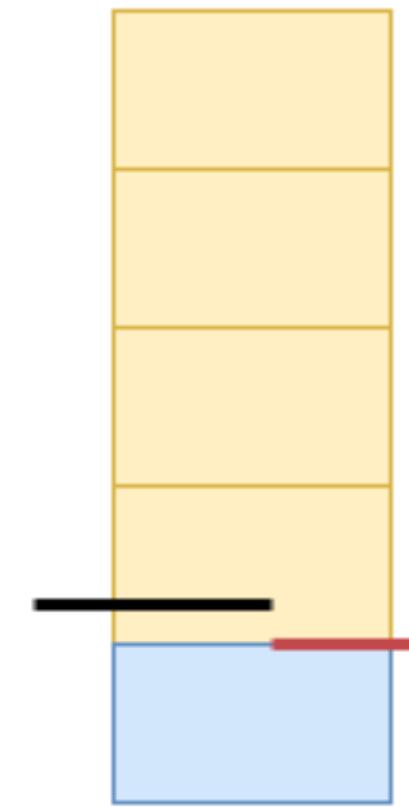


Equivalent to min
available = $3 / 4 = 0.75$



Meets threshold of min
available ratio ≤ 0.9

5 Replicas
25% max unavailable



Equivalent to min
available = $4 / 5 = 0.8$



Meets threshold of min
available ratio ≤ 0.9

Allowed to be
temporarily down

Cannot go down

Kubernetes % max
unavailable



Integer-rounded
max unavailable



AWS Node Termination Handler

Gracefully handle EC2 instance shutdown within Kubernetes

- 오픈소스 스팟 노드의 graceful shutdown, cordon, drain 자동화 툴
- 인스턴스의 메타데이터 혹은 ASG Lifecycle Event를 모니터링
- 인스턴스 종료 2분전 Spot Instance Interruption 발생
- NTH가 인터럽션 발생 노드를 cordon 하여 스케줄링 방지, 새로운 스팟노드 호출
- 호출이 완료되면 인터럽션 노드의 파드 재배치

AWS Node Termination Handler

유의점

- Rebalance Recommendation에 대응하지만...
- 그렇지 않은 경우 2분 안에 파드 재배치까지 이뤄져야 함
 - 스팟인스턴스 호출 + 부팅 + 파드 적재까지의 시간 고려
 - 파드가 재기동 되는데 걸리는 시간이 가능한 짧아야 할 것
- 라이브라면 재배치가 실패할 경우를 대비
 - 하나의 서비스에서 일반과 스팟 인스턴스에 나눈 pod 배치 고려
 - 오버프로비저닝도 답이 될 수 있겠음

AWS Node Termination Handler

Gracefully handle EC2 instance shutdown within Kubernetes

Which one should I use?

Feature	IMDS Processor	Queue Processor
Spot Instance Termination Notifications (ITN)	✓	✓
Scheduled Events	✓	✓
Instance Rebalance Recommendation	✓	✓
ASG Termination Lifecycle Hooks	✗	✓
Instance State Change Events	✗	✓

AWS Node Termination Handler

유의점

- 메타데이터를 이용하는 IMDS 가 설치가 간단하여 테스트에 용이
- 인스턴트 상태 알람, 디플로이먼트를 한 개만 사용하는 Queue는 비용면에서 이점
 - AWS 리소스를 사용하기 위한 권한설정이 추가로 필요
 - 스팟 인스턴스 호출 상태도 확인 가능
- 운영이 편한것으로 하세요

결론

- FinOps가 핫하다. 클라우드 운영에서 비용관리의 중요성이 점점 증가
- Kubecost를 이용하여 EKS의 비용가시성 확보 + 최적화 인사이트 얻으세요~
- x86 ➔ ARM 아키텍처 이전 = 40% 비용절감
- 온디맨드 대비 80% 절약이 가능한 스팟인스턴스 노드
 - 서비스의 스팟 리드니스 체크리스트 확인
 - 노드 가용성 확보를 위한 Node Termination Handler