

엔터프라이즈 쿠버네티스

기업용 쿠버네티스 플랫폼

쿠버네티스 소개

간단한 역사 및 기술 소개

쿠버네티스

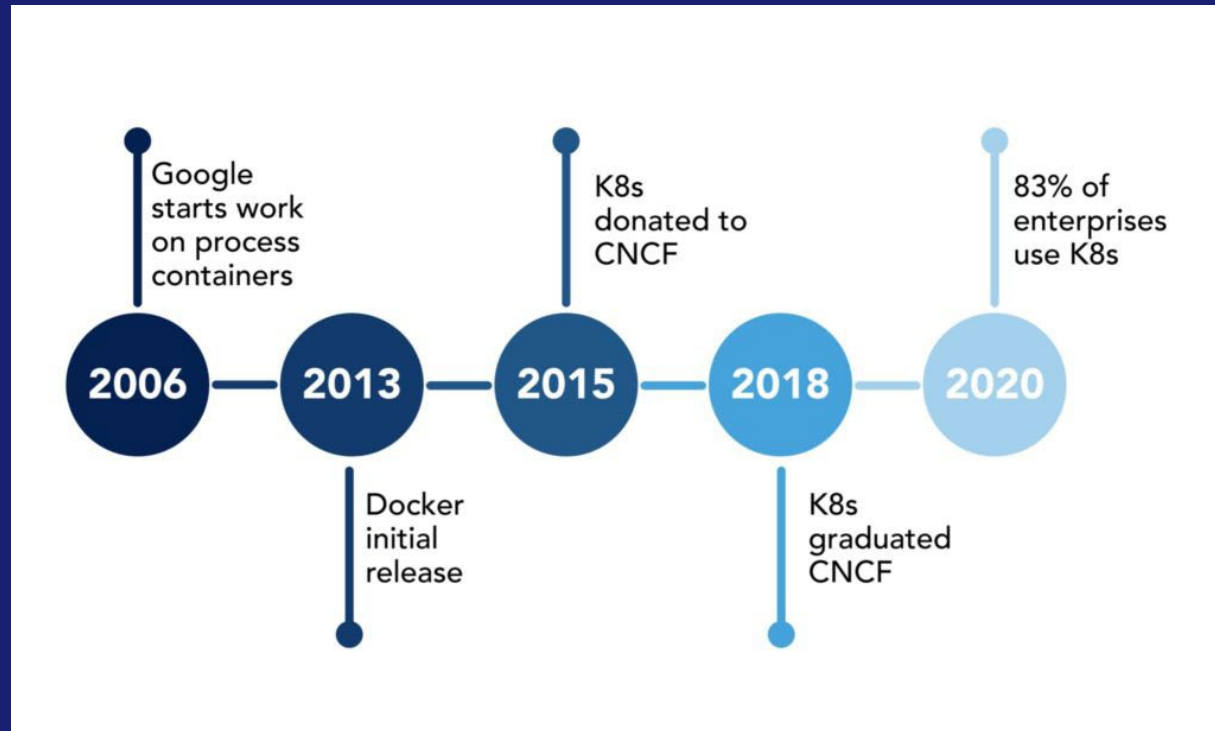
쿠버네티스 본래 구글에서 내부 서비스 관리 용도로 사용함. 구글에서 Borg라는 이름으로 2014년도에 릴리즈 하였음. 구글에서는 대략 2006년도에서 내부적으로 서비스 및 컨테이너 관리 방법에 대해서 논의가 됨.

구글은 2014년도에 깃-허브에 처음 코드를 릴리즈 하였으며, 현재 기준 11년 동안 꾸준히 릴리즈가 됨. 현재 CNCF에서는 Serverless 및 Knative를 구현하기 위해서 쿠버네티스를 사용하고 있으며, 현재는 가상머신도 같이 컨테이너와 동일하게 POD기반으로 관리한다.

오픈스택은 쿠버네티스를 API기반으로 지원 및 자원 연동을 할 수 있도록 지원하고 있으며, 현재는 STARINGX를 OpenInfra를 통해서 지원하고 있다.

쿠버네티스

쿠버네티스는 구글에서 Borg라는 이름으로, 내부적으로 클러스터 관리 및 배포 용도로 사용하였다. 초기 Borg는 C++로 작성이 되었으나, 쿠버네티스로 넘어 오면서 Go언어 기반으로 작성이 되었다. 초기 릴리즈는 2015년도, 발표는 2014년도에 이루어 졌다.



가상화/컨테이너

쿠버네티스는 자체적으로 런타임을 지원하지 않는다. 여기서 말하는 런타임(runtime) 컨테이너 이미지에 있는 내용을 메모리에 불러와서 프로세서를 격리 및 추적하여, 호스트와 분리 운영이 가능하도록 환경을 구성을 한다.

이러한 이유 때문에 가상머신과 많은 혼동이 있는데, 둘은 엄연히 큰 차이가 있다.

가상머신은 링 구조(ring structure)를 생성 및 재구성하기 때문에, 더 복잡하고 독립적인 자원을 가지고 동작한다. 가상머신은 런타임에서 동작하는 구조가 아닌, 일반적으로 하이퍼바이저(hypervisor)나 혹은 가속기(kernel module level)에서 동작한다.

컨테이너는 이와 반대로 가상화 같은 하드웨어 수준의 기술이 필요하지 않으며, 커널 수준에서 격리 및 추적 기술 기반으로 프로세스를 관리한다.

이러한 이유로, 가상머신에서 사용하던 기능을 전부 컨테이너로 마이그레이션이 가능하지 않기 때문에, 서비스 아키텍트 및 구성에 따라서 플랫폼을 선택 및 구성하면 된다.

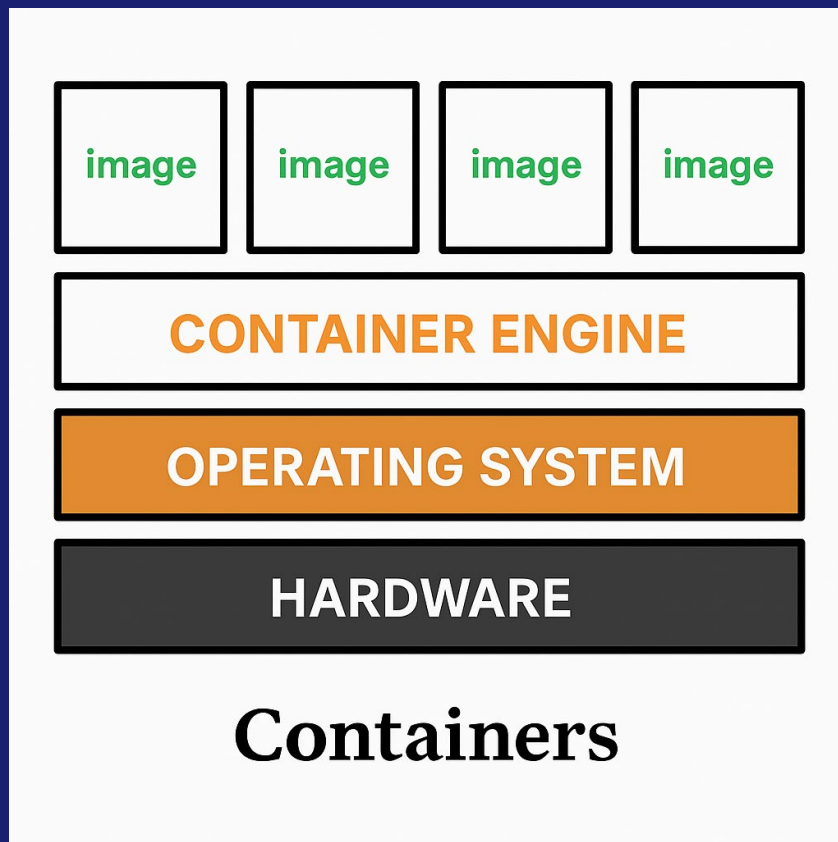
가상화/컨테이너

시점	가상화	컨테이너
격리	각각 시스템에 게스트 운영체제가 설치 및 동작이 되어야 된다	호스트 커널의 커널을 공유해서 사용한다
리소스 자원	각 가상머신의 자원을 별도로 가지고 있음	호스트의 자원을 cgroup, namespace를 통해서 공유 받음
성능	인스턴스(VM)를 통해서 많은 오버헤드 링/채널이 발생함.	호스트 커널과 공유하기 때문에 오버헤드는 거의 발생하지 않음
이동성	가상머신은 이동성(하이퍼바이저)은 매우 낮음	컨테이너 시스템은 다른 시스템에 이동이 가능

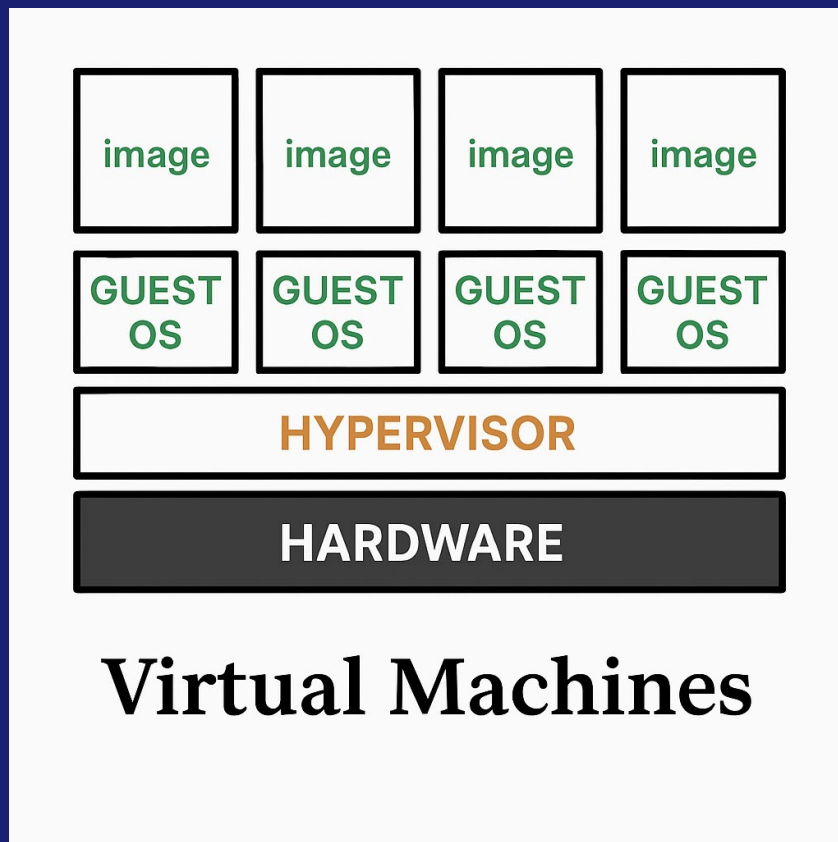
가상화/컨테이너

시점	가상화	컨테이너
배포 속도	배포속도는 느림. OS부팅도 필요함	배포 속도는 빠름. 부팅이 필요 없으며, 하이퍼바이저 관리도 필요 없음
자원 관리	인스턴스 개수에 따라서 더 많은 자원이 필요함.	기존 호스트의 자원을 최대한 효율적으로 사용
시스템 환경	많은 도구와 하이퍼바이저 및 관리도구 필요. 커널 수정도 필요	도구는 매우 가볍고 간단함 쿠버네티스 및 포드만과 같은 도구
케이스	다수 애플리케이션을 하나의 컴퓨트 노드에서 실행	기존 애플리케이션 및 마이크로서비스 형태로 배포 및 구성이 가능

컨테이너



가상화



자원 구성

쿠버네티스가 사용하는 물리적 자원은 다음과 같다.

1. Controller(AKA Master)
2. Compute(AKA minion/worker)
3. Infra Node(Optional)

위의 두 개 용어, "master", "minion", "worker"는 오랫동안 IT계열에서 많이 사용하였지만, 차별적인 단어로 현재는 취급이 되고 있어서 통상적으로 "controller", "compute"와 같은 단어로 대체가 되어가고 있다.

자원 구성

쿠버네티스는 오래전 버전은 모든 서비스가 hosted 형태로 구성이 되어 있었지만, 현재 쿠버네티스는 kubelet제외한 나머지 서비스는 containerized가 되었다. 이 말은 kubelet은 Proxy/Bootstrap/Monitor를 hosted상태에서 담당하고, 나머지는 서비스는 Static-POD가 되었다.

Static-POD는 우리가 알고 있는 POD와 동일하지만, 쿠버네티스에서 구성 시 기본적으로 생성하는 POD서비스 이다. Static-POD는 다음과 같다.

- kube-scheduler
- kube-apiserver
- kube-ControllerManager
- CoreDNS
- ETCD

릴리즈 형태

The Kubernetes project maintains release branches for the most recent three minor releases (1.32, 1.31, 1.30). Kubernetes 1.19 and newer receive [approximately 1 year of patch support](#). Kubernetes 1.18 and older received approximately 9 months of patch support.

Kubernetes versions are expressed as **x.y.z**, where **x** is the major version, **y** is the minor version, and **z** is the patch version, following [Semantic Versioning](#) terminology.

자세한 내용은 "<https://kubernetes.io/releases/>"에서 확인이 가능하다. 이를 통해서 업그레이드 방법을 고려해야 한다. 쿠버네티스에서 업그레이드 가이드 문서가 있지만, 구체적인 내용에 대해서는 사용자가 스스로 해야 한다.

기업용 쿠버네티스

쿠버네티스

쿠버네티스(OCP/SUSE Rancher/Nutanix)

엔터프라이즈 쿠버네티스 플랫폼에서 많이 사용하는 제품은 다음과 같다.

1. OpenShift Platform
2. SUSE Rancher/Harvester
3. Nutanix NKP

위의 모든 제품은 쿠버네티스 기반으로 컨테이너/가상화 그리고 인프라까지 통합적으로 제공하는 제품이다. 가상머신 기반으로는 북미에서는 Nutanix가 VMware가 선두를 다투고 있으며, Redhat, SuSE 경우에는 가상화 시장에서 추격자로 올라오고 있다.

북미 시장

VMware는 여전히 가상화 소프트웨어 시장에서 약 43%의 점유율을 유지하며 선두를 지키고 있다. 이는 오랜 기간 동안의 기술적 안정성과 광범위한 고객 기반으로 가능.

하지만, Nutanix는 HCI 시장에서 약 25%의 글로벌 점유율을 보유하고 있으며, 북미에서도 강력한 입지를 확보하고 있다. 특히, VMware의 가격 인상과 복잡성에 대한 우려로 인해 Nutanix로의 전환을 많은 북미 기업들이 고려하고 있다.

특히, 가상머신 시장에서는 Nutanix HCI를 매우 고려하고 있다. 최근에 Nutanix는 컨테이너 기술도 인수를 통해서 자사에 흡수, NKP라는 이름으로 쿠버네티스 플랫폼도 HCI기반으로 통합을 진행하고 있다.

시장 변화 요인

VMware는 다음과 같은 이유로 시장에서 변화가 발생하고 있다. 뛰어난 안전성은 제공하지만, 회사가 지속된 독립 및 인수를 통해서 운영 안정화가 약해지기 시작했다. 이러한 이유로, CFO/CTO 그리고, 전략 사업팀에서는 플랫폼 전환에 대해서 고려를 하기 시작 하였다.

1. Broadcom의 VMware 인수 이후, 일부 고객들은 라이선스 정책 변경과 가격 인상에 대한 우려로 대안모색
2. Nutanix는 AWS, Cisco, Dell 등과의 파트너십을 통해 시장 점유율을 확대하고 있으며, 유연한 라이선스 정책과 비용 효율성으로 주목
3. Redhat은 비용 문제로 고민이 많은 고객을 위해서 Openshift Virtualization을 출시. 하지만, VMware vSphere와 비교 하였을 때 관리자 도구의 부재가 큼
4. SuSE는 Harvester기반으로 가상화 플랫폼을 지원 및 제공. Rancher와 손쉬운 통합을 제공하고 있으나, 컴퓨터 노드의 통합은 아직 완벽하지 않음.

쿠버네티스(REDHAT/SUSE/NKP)

이름	설명	프로덕트 레디
Vanila Kubernetes (AKA Kubernetes)	순수 쿠버네티스 버전. 기본적인 기능만 있으며, Federation API를 제공하기 때문에, 클러스터 연합 도구를 통해서 다중 클러스터 운영 기능	아니요
Redhat OpenShift	레드햇 오픈 시프트는 순수 쿠버네티스 기반으로 운영에 필요한 필수 기능을 통합 및 운영이 가능하도록 되어 있다. 제일 큰 차이점은 오퍼레이터를 통해서 모든 자원이 구성 이 된다. 기존에 사용하던 쿠버네티스 표준 자원도 그대로 활용 및 적용이 가능하다. 다만, 다시 쿠버네티스로 마이그레이션 은 불가능하다.	네
Redhat Openshift Virtualization	OpenShift Virtualization은 OpenShift 클러스터 내에서 전통적인 가상머신(VM)을 컨테이너 네이티브 방식으로 함께 운영 할 수 있게 해주는 기능이다. 기반은 KubeVirt이며, OpenShift의 네이티브 리소스로 VM을 생성, 배포, 운영할 수 있다. 하지만, VMware와 비교 시, 기능이 많이 부족한다.	네

쿠버네티스(REDHAT/SUSE/NKP)

이름	설명	프로덕트 레디
SuSE Rancher	수세 랜처는 바닐라 쿠버네티스 및 통합된 쿠버네티스 클러스터도 같이 제공한다. 최대 장점은 여러 클러스터를 다중 구성 및 운영이 손쉽고, 기존에 사용하던 쿠버네티스 클러스터를 그대로 Rancher에 통합하여 운영 및 사용이 가능하다.	네
SuSE Harvester	Harvester는 기존의 OpenStack이나 vSphere보다 훨씬 가벼우며, YAML 기반의 배포 자동화와 Rancher 연동을 통한 DevOps 친화성이 높은 것이 장점이다. 그러나 2023~2024년까지는 일부 엔터프라이즈 기능(HA 구성, 백업, SR-IOV 등)이 제한적이었으며, 최근 들어 점진적으로 보완되고 있다.	네
Nutanix Kubernetes Platform	NKP는 Nutanix 인프라에 최적화된 쿠버네티스 플랫폼으로, 하이퍼컨버지드 인프라(HCI)와 통합되어 간편한 배포 및 운영이 가능하다. Life Cycle 관리, 인증 통합, RBAC, 모니터링 등 엔터프라이즈에 필요한 기능을 내장하고 있으며, 여러 클러스터를 통합 관리할 수 있다. VM 기반 환경에서의 컨테이너 도입에 적합하다.	네

제품 평가

많이 사용하는 쿠버네티스 제품에 대해서 비교하면 보통 다음과 같이 평가가 된다.

플랫폼	설명
SuSE Rancher	쿠버네티스 기반으로 구성된 보안 중심 컨테이너 오케스트레이션 도구
SuSE Harvester	쿠버네티스 가상화 기반으로 기업수준 가상화 환경 제공
OpenShift Container	쿠버네티스 기반으로 구성된 레드햇 컨테이너 오케스트레이션 도구
OpenShift Virtualization	쿠버네티스 가상화 기반으로 기업수준 가상화 환경 제공. 다만, OpenShift Container를 제외하고 사용하는 경우, UI기반의 관리도구가 많이 부족하다.
Kubernetes (Upstream K8s)	표준 오픈소스 컨테이너 오케스트레이션 도구
Nutanix Kubernetes Platform	Nutanix의 하이퍼컨버지드 인프라에 최적화된 쿠버네티스 플랫폼으로, 간편한 배포와 통합 관리 기능을 제공하는 엔터프라이즈 오케스트레이션 도구

제품 평가

컨테이너 기반으로 평가 시 다음과 같다.

사용 목적	추천 플랫폼
보안 중심 Kubernetes 운영	RKE2(기본 CIS 하드닝), NKP(기본 보안 정책 및 Prism 기반 접근 제어)
엔터프라이즈 환경	RKE2/OpenShift/NKP
순수 Kubernetes 운영	Kubernetes(Upstream)
멀티클러스터 및 하이브리드 클라우드	RKE2/OpenShift/NKP(Prism Central 기반 클러스터 관리)
빠른 배포 및 경량 운영	RKE2/NKP(간편한 설치 자동화)
강력한 CI/CD 및 개발자 환경 제공	RKE2/OpenShift/NKP (DevOps 친화적인 Prism UI 연동)

제품 기능 비교(컨테이너)

설치 및 운영 편의성은 다음과 같다.

기능	RKE2	OpenShift Container	Kubernetes	NKP
설치 편의성	쉬움(스크립트 기반)	어려움 (앤서블/테라폼 혼용)	중간 (kubeadm/YAML)	Prism UI 기반 설치 및 자동화 지원
보안	기본적으로 CIS 하드닝 적용	SELinux, SCC(보안 컨텍스트) 적용	운영체제에 따라서 다름	기본 CIS 적용, Prism 기반 인증 통합
네트워크	CNI 플러그인 선택 가능 (Flannel, Calico 등)	OpenShift SDN (OVN, Multus)	표준 CNI 플러그인 전부 지원	Calico 기반 기본 제공, CNI 커스터마이징 가능
서비스 메쉬	Istio, Linkerd 사용 가능	기본 Istio 내장 (Service Mesh Operator)	표준 CNCF 사용 가능	Istio 사용 가능 (수동 설치)
스토리지	CSI 지원			Nutanix CSI 지원

제품 기능 비교(컨테이너)

기능	RKE2	OpenShift Container	Kubernetes	NKP
운영 관리	Rancher UI와 통합	OpenShift 콘솔 제공 (기본 UI, GitOps 지원)	kubectl 및 대시보드 (추가 설치)	Prism Central을 통한 중앙 통합 관리
업데이트 관리	Rancher UI를 통한 자동화	레드햇 가이드 기반으로 업데이트	수동 업데이트 필요	Prism UI 기반 자동 업데이트 기능 제공
멀티클러스터	Rancher 기반 멀티클러스터 지원	OpenShift Cluster Manager 제공	Kubernetes Federation V1/V2지원	Prism Central 및 Karbon 플랫폼 기반 지원

제품 기능 비교(컨테이너)

항목	RKE2	OpenShift	Kubernetes	NKP
오버헤드	낮음 (경량화)	높음	보통	낮음 (Prism 기반 최적화)
컨테이너 실행 속도	빠름	빠름 (CRI-O 사용)	빠름	빠름
네트워크 성능	CNI에 따라 다름	OpenShift SDN, 느림	CNI에 따라 다름	CNI에 따라 다름 (기본 Calico 성능 양호)
노드 확장성	고성능	엔터프라이즈 환경에서 강력	확장 가능하지만 구성 필요	Prism Central 기반 확장, 고성능
업데이트 성능	빠른 롤링 업데이트	안정적이지만 복잡함	노드 단위 업데이트	Prism UI 기반 자동화된 롤링 업데이트

가상화

쿠버네티스 가상화

소개

오픈소스 기반 회사에서 제공하는 쿠버네티스와 연동 및 호환이 되는 플랫폼은 다음과 같다.

1. SuSE Rancher/Harvester
2. OpenShift Container/Virtualization
3. Nutanix NKP

각 회사별로 장단점이 있으며, 회사 환경에 맞게 적절한 제품을 선택해야 한다. 하지만, 대다수 상용회사에서는 자신 제품에 대해서 장점만 내세우기 때문에, 매뉴얼 기준으로 아래와 같이 정리 하였다.

Harvester

Harvester는 SUSE가 개발한 오픈소스 기반의 경량 HCI(하이퍼컨버지드 인프라) 플랫폼이다. KubeVirt 기반으로 가상 머신을 관리하고, Rancher와 통합되어 DevOps 환경과 친화적인 가상화 인프라를 구성할 수 있다.

UI는 Rancher Dashboard에 통합되어 있어 비교적 직관적이고, Longhorn을 통해 기본 스토리지를 제공한다.

■ 장점

- 오픈소스 기반으로 라이선스 비용 부담이 없고, Rancher와 연동이 쉬워 DevOps와 잘 어울린다.
- 설치가 간편하며, 리소스가 적은 환경에도 적합하다.

■ 단점

- 고급 기능(예: 라이브 마이그레이션, 정교한 네트워크 정책 등)이 제한되거나 실험적이다.
- 대규모 프로덕션 환경에서의 안정성은 아직 제한적이다.

■ Kubernetes 연동

- Rancher를 통해 Kubernetes 클러스터를 직접 배포하거나 관리할 수 있으며, VM 기반 K8s 노드를 만들 수도 있다.
- 하지만 컨테이너/VM 혼합 스케줄링은 지원하지 않고, 병렬 운영 구조다.

OpenShift Virtualization

OpenShift Virtualization은 OpenShift 위에서 KubeVirt를 기반으로 가상머신을 Kubernetes 리소스로 실행할 수 있게 해주는 기술이다. VM과 컨테이너가 동일한 네트워크, 보안 정책, 스토리지 정책 내에서 함께 동작할 수 있으며, GitOps 및 DevSecOps 환경에 자연스럽게 통합된다.

■ 장점

- Kubernetes 리소스 관리 방식으로 VM을 컨테이너처럼 다룰 수 있는 통합된 모델을 제공한다.
- 보안 정책(Security Context, SCC), SELinux, 네트워크 멀티플러그(Multus) 등 엔터프라이즈 기능이 풍부하다.

■ 단점

- 초기 구축이 복잡하고, OpenShift 구독이 필수이므로 비용 부담이 크다.
- VM 관리 관점에서 보면 기능이 다소 단순하며, 운영자 입장에서 vSphere나 Prism만큼 편하진 않다.
- 다만, 별도로 OV만 구입하는 경우, OpenShift Container와 통합이 되지 않고, 독립적으로 동작한다.

■ Kubernetes 연동

- OpenShift 자체가 Kubernetes 기반이므로, 완전한 통합 환경이다.
- VM을 Kubernetes 리소스로 선언하고, Deployment/Service 등과 동일하게 처리 가능하다.

Nutanix NKP (Karbon + AHV)

Nutanix Kubernetes Platform(NKP)은 Nutanix의 AHV 기반 HCI 위에 Kubernetes 클러스터를 쉽게 구축하고 운영할 수 있도록 지원하는 플랫폼이다. Karbon 또는 Karbon Platform Services를 통해 Kubernetes 클러스터를 자동 배포하고, Prism을 통해 가상머신과 스토리지, 네트워크를 통합 관리할 수 있다.

■ 장점

- Prism UI를 통해 VM, 컨테이너, 스토리지를 단일 창에서 관리할 수 있어 운영이 매우 단순하다.
- AHV 하이퍼바이저는 라이선스 비용이 없으며, Nutanix AOS는 자동 복구, 스냅샷, 복제 기능이 내장되어 있다.
- 엔터프라이즈 환경에서 VM 중심 워크로드를 Kubernetes와 **부드럽게 병행 운영**할 수 있다.

■ 단점

- Nutanix 인프라에 종속되며, 타사 HCI나 클라우드에서는 활용할 수 없다.
- DevOps 친화성이나 오픈소스 확장성 측면에서는 상대적으로 제한된다.

■ Kubernetes 연동

- Karbon이 생성한 클러스터는 CSI/CNI가 통합되어 있으며, **스토리지와 네트워크 리소스를 자동 연계**할 수 있다.
- VM과 K8s는 분리 운영되지만, Prism UI에서 모두 관찰하고 제어할 수 있다.

VMware vSphere

VMware vSphere는 가장 성숙하고 널리 사용되는 가상화 플랫폼으로, ESXi 하이퍼바이저와 vCenter를 중심으로 가상머신 인프라를 구성한다.

수년간의 기술 성숙도와 광범위한 에코시스템으로 인해 기업 표준 인프라로 자리 잡았다.

■ 장점

- 가상머신 고급 기능(실시간 마이그레이션, DRS, 고가용성, 백업 등)이 매우 성숙되어 있다.
- 수많은 서드파티 도구, 모니터링, 자동화 플랫폼과의 호환성이 높다.
- AD 연동 및 정교한 권한 제어, 엔터프라이즈 보안 설정이 가능하다.

■ 단점

- 라이선스 비용이 매우 높고, 최근 Broadcom 인수 이후 정책 불확실성이 커졌다.
- Kubernetes와의 연동은 vSphere with Tanzu를 통해 가능하나, 복잡하고 제약이 많다.

■ Kubernetes 연동

- vSphere with Tanzu를 통해 Kubernetes를 통합할 수 있지만, 별도 구성과 라이선스가 필요하다.
- 가상머신 기반 클러스터 노드로 K8s를 구성하는 경우가 더 일반적이다.

정리

위의 내용을 정리하면 다음과 같다.

- Harvester는 DevOps와 오픈소스 친화적이지만, 기능은 아직 성장 중이다.
- OpenShift Virtualization은 보안과 통합은 뛰어나지만, 진입 장벽이 높다.
- NKP는 운영 편의성과 안정성이 뛰어나며, VM 중심 조직엔 최적의 선택지다.
- VMware는 가장 성숙한 가상화 플랫폼이지만, 비용과 정책 측면에서 부담이 크다.

HCI 비교

Nutanix Kubernetes Platform

SUSE Rancher

Redhat Openshift

VMWare

Nutanix Kubernetes Platform (NKP) – HCI 기반

Nutanix Kubernetes Platform(NKP)은 Nutanix의 AHV 하이퍼바이저 기반 HCI(하이퍼 컨버지드 인프라) 위에서 컨테이너 오케스트레이션 환경을 손쉽게 구축할 수 있도록 지원하는 플랫폼이다. NKP는 Nutanix의 **Karbon** 또는 **Karbon Platform Services**를 기반으로 하며, Nutanix Prism 콘솔을 통해 가상머신, 스토리지, 네트워크, 컨테이너를 하나의 통합 인터페이스에서 관리할 수 있는 점이 가장 큰 특징이다.

기존의 컨테이너 플랫폼(OpenShift, Rancher 등)은 별도의 하이퍼바이저 환경이나 독립된 VM 위에 설치되는 형태가 일반적인 반면, NKP는 Nutanix의 AHV 하이퍼바이저와 **직접적으로 통합**되어 있어 운영 복잡도가 매우 낮고, 초기 도입도 간편하다. AHV는 VMware에 비해 라이선스 비용이 없기 때문에 전체적인 TCO(총소유비용) 절감에도 매우 유리하다.

개요 및 아키텍처 특징

1. NKP는 Nutanix AHV 기반의 HCI 인프라 위에서 Kubernetes 클러스터를 직접 구축할 수 있는 플랫폼이다.
2. Nutanix Prism을 통해 VM, 컨테이너, 스토리지, 네트워크 자원을 단일 인터페이스에서 통합 관리할 수 있다.
3. AHV 하이퍼바이저는 라이선스 비용이 없고 고성능, Kubernetes와의 연동이 자연스럽다.
4. 기존 VM 기반 인프라를 유지하면서 컨테이너 환경을 점진적으로 도입하기에 적합하다.
5. Nutanix AOS + AHV 위에 Karbon (또는 Karbon Platform Services) 를 설치하여 K8s 클러스터 구성
6. Nutanix Volumes 및 Files는 CSI 기반의 영구 볼륨 제공, 컨테이너 환경에 최적화된 스토리지 통합
7. 백업, DR, 보안, 리소스 자동화 등 Nutanix 기반 엔터프라이즈 기능과 완전 연동

NKP의 HCI 기반 장점 및 도입 효과

핵심 장점

- 운영 단순화: VM과 K8s 클러스터를 단일 Prism UI에서 통합 관리
- 비용 절감: AHV는 라이선스 무료, VM + K8s를 하나의 노드에서 운용 가능
- 설치 자동화: Karbon을 통해 클릭 몇 번으로 K8s 클러스터 설치 및 업그레이드 가능
- 스토리지 통합: Files/Volumes 연계로 고성능 영구 볼륨 자동 제공
- 기존 인프라 활용: 별도 하드웨어/소프트웨어 없이 즉시 Kubernetes 환경 구성 가능

NKP의 HCI 기반 장점 및 도입 효과

도입 효과

- 초기 진입장벽이 낮고, 운영 리소스 최소화 가능
- 기존 VM 기반 환경을 유지하면서도 컨테이너 도입을 무리 없이 병행 운영
- OpenShift/Rancher 대비 비용과 복잡성에서 유리한 구조

세부 비교 설명

HCI 인프라 개요

HCI(Hyper-Converged Infrastructure)는 서버, 스토리지, 네트워크를 하나의 통합된 노드 단위로 구성하여, 전통적인 인프라 대비 더 단순하고 확장 가능한 구조를 제공하는 차세대 데이터센터 아키텍처이다.

기존의 3티어 인프라는 컴퓨트, 스토리지, SAN 스위치가 분리되어 있어 구성과 관리가 복잡하고 장애 지점이 많았던 반면, HCI는 이러한 리소스를 소프트웨어 정의 방식(SDS, SDN 등)으로 통합하여 **노드 단위의 블록 확장성과 중앙 집중형 운영 자동화**를 가능하게 한다.

HCI는 전통적으로 가상머신 기반으로 구성되며, VM 워크로드를 안정적으로 실행하기 위한 기반으로 설계되었다. 최근에는 컨테이너 기술과의 융합도 활발히 이루어지고 있으나, 컨테이너는 아직까지 VM에 비해 부가적인 컴퓨팅 역할로 활용되는 경우가 많다.

따라서, 많은 개수의 VM이 생성되거나 컴퓨팅 자원이 자주 증감하는 환경에서는 HCI 기반 인프라를 통해 **자원을 유연하고 효율적으로** 관리할 수 있다.

제품 비교 설명

1. Nutanix NKP (Karbon/NKP)

- **강점:** AHV 기반 HCI에 최적화된 구조. Nutanix Prism으로 VM과 Kubernetes를 함께 운영 가능.
- **제약:** VM과 POD를 완전히 통합된 단일 인터페이스로 관리하기는 어려움.
- **활용:** 기존 Nutanix 환경에서 VM과 컨테이너를 모두 사용하는 기업에 적합.

2. Rancher + Harvester

- **강점:** 오픈소스 기반 HCI(Harvester) + Rancher 조합으로 VM과 Kubernetes 통합 가능.
- **구성 유연성:** 다양한 인프라 위에 설치 가능하며, 클라우드와의 통합도 쉬움.
- **활용:** 비용을 절감하면서도, VM과 컨테이너를 오픈소스 기반으로 구성하고 싶은 조직에 적합.

제품 비교 설명

3. OpenShift + OpenShift Virtualization

- **강점:** VM과 POD의 완전 통합, GitOps/DevSecOps 내장, 보안과 정책 제어 강화.
- **HCI 최적화:** Red Hat HCI 혹은 Ceph 기반 환경에서 높은 최적화 달성.
- **활용:** 정부, 금융, 대기업에서 VM에서 컨테이너로 천천히 이전하고 싶은 경우에 적합.

HCI 환경 기준 컨테이너 및 가상화 기능 비교

항목	Nutanix NKP	Rancher	OpenShift
기반 인프라	Nutanix AHV 기반 HCI (AOS)	다양한 인프라에 유연하게 설치 가능 (on-prem, HCI, 클라우드)	Red Hat 기반 인프라 요구 (특히 OpenShift Virtualization은 HCI에 적합)
하이퍼바이저 통합	AHV (자체 하이퍼바이저) 완전 통합, Prism 기반 관리	직접적인 하이퍼바이저 제공 안함. Harvester와 결합 시 HCI + VM 제공 가능	KubeVirt 기반 VM 제공 (OpenShift Virtualization), 고급 네트워크와 스토리지 통합 필요
컨테이너 오케스트레이션	Kubernetes (Nutanix Karbon/NKP) 기반, CSI/CSI 연동 지원	CNCF 순정 Kubernetes, 다양한 클러스터 배포와 멀티 클러스터 관리 지원	OpenShift Kubernetes (엄격한 정책과 보안), 강력한 CI/CD 통합 지원

HCI 환경 기준 컨테이너 및 가상화 기능 비교

항목	Nutanix NKP	Rancher	OpenShift
가상화 기능	AHV 기반 VM + CSI로 Kubernetes 연동 (VM ↔ POD 혼합 운용은 어려움)	Harvester 사용 시 KubeVirt 기반으로 VM/POD 공존 가능	KubeVirt 기반으로 컨테이너와 VM을 통합 운영 가능 (OpenShift Virtualization)
스토리지 통합	Nutanix Files/Volumes 완전 통합, CSI 기반 볼륨 제공	외부 스토리지 플러그인 /Longhorn/Rook 등으로 유연하게 구성	OpenShift Data Foundation (ODF), Ceph 기반 강력한 스토리지 통합
HCI 최적화 수준	가장 최적화, VM과 컨테이너 통합 운영은 제한적	HCI에 설치 가능하지만, HCI 최적화는 낮음. Harvester 병용 시 오픈소스 HCI 구현 가능	Red Hat HCI (RHVP 등) 기반 연동 가능, HCI 환경에 맞춰 VM/컨테이너 융합 가능

HCI 환경 기준 컨테이너 및 가상화 기능 비교

항목	Nutanix NKP	Rancher	OpenShift
라이선스/비용	상용, Nutanix Prism 관리 포함, AHV 무료	오픈소스, 기업용 서브스크립션 선택 가능	상용, Red Hat 구독 필수 (OpenShift Virtualization 포함 시 높은 비용)
운영 난이도	Prism 중심 쉬운 관리, 초기 세팅은 Nutanix 중심 지식 필요	중급 이하 수준에서도 운영 가능. 구성 유연	높은 학습 곡선. 전문적인 지식 및 Red Hat 생태계 이해 필요
대표 활용 사례	HCI기반 데이터센터, 엔터프라이즈 VM → K8s 전환 용	중소기업/유연한 환경에 적합. 멀티 클러스터 운영 및 DevOps 환경에 강점	금융/공공/대기업 중심, 복잡한 워크로드 통합 및 보안 중심 운영에 적합

HCI 결론

HCI 플랫폼의 결론: 왜 HCI인가?

HCI 설명

HCI(Hyper-Converged Infrastructure)는 컴퓨트, 스토리지, 네트워크 기능을 소프트웨어 정의 방식으로 통합하여, 하나의 서버 노드 단위에서 복수의 리소스를 자동으로 제공하고 확장할 수 있는 인프라 구조이다.

기존의 전통적인 3티어 인프라(서버, 스토리지, SAN 스위치)가 구성과 관리가 복잡하고 비용이 높은 구조였다면, HCI는 모든 리소스를 단일 하드웨어 노드에 통합하여 운영 단순화, TCO 절감, 확장 유연성을 제공한다.

HCI의 핵심은 분산 스토리지(Software-Defined Storage)와 하이퍼바이저 통합 운영, 그리고 자동화된 관리 플랫폼(UI/API)에 있으며, 이러한 요소가 통합된 아키텍처를 통해 VM 기반 환경뿐만 아니라 컨테이너, DevOps 워크로드까지 유연하게 수용할 수 있다.

최근에는 HCI가 단순한 가상화 인프라를 넘어, Kubernetes 클러스터의 기반 인프라로도 채택되며, 멀티 클라우드 및 엣지 컴퓨팅 환경까지 확장되는 추세다.

출처: IDC, "HCI in the Multi-Cloud Era", 2024 / Gartner Magic Quadrant for HCI Software, 2024

HCI 지원 여부

각 제품들은 HCI를 지원한다. 하지만, 수준의 차이가 조금씩 있다.

제품명	HCI 제공 여부	HCI 구성 요소 포함 여부	비고 설명
Nutanix NKP	예 (완전 제공)	AHV(하이퍼바이저) + AOS(분산 스토리지) + 네트워크 + K8s	진정한 의미의 풀스택 HCI. 가장 완성도 높음
Harvester	부분 제공	KubeVirt 기반 VM + Longhorn(CSI) + 기본 네트워크	경량형 오픈소스 HCI로 볼 수 있음. 소규모에 적합
VMware vSphere	부분 제공	vSphere + vSAN + NSX(선택적) 조합으로 HCI 구성 가능	HCI 구성은 가능하지만 별도 제품 결합 필요
OpenShift Virtualization	HCI 아님	KubeVirt 기반 VM만 제공. 스토리지/네트워크는 외부 솔루션 의존	인프라 자체를 제공하진 않음. K8s 위에서 VM만 실행

HCI의 필요성과 가치

HCI의 필요성과 가치

- **HCI(Hyper-Converged Infrastructure)**는 서버, 스토리지, 네트워크를 노드 단위로 통합하여, 전통적 3티어 인프라 대비 구성과 운영을 더 단순하고 효율적으로 만드는 아키텍처이다
- 스토리지 복제, 고가용성, 백업, 자원 자동화 등 복잡한 기능을 기본 내장하고 있어, 기업은 별도 스토리지나 백업 장비 없이도 고성능의 VM 환경을 운영할 수 있다
- 특히 노드 단위로 확장되기 때문에, **컴퓨팅/스토리지 자원**이 자주 변하는 환경에 이상적이다
- 최근에는 컨테이너까지 포함하여, **VM + K8S**의 통합 인프라 플랫폼으로 발전하고 있다

HCI의 필요성과 가치

HCI의 핵심 장점

- 운영 자동화, 자가 복구, 단일 관리 인터페이스
- 빠른 구축, 낮은 TCO, 높은 확장성과 유연성
- VM 중심 환경에 가장 실용적이고 안정적인 선택지

어떤 HCI 플랫폼이 가장 실용적인가?

HCI 플랫폼별 요약 평가(인프라 관점)

플랫폼	장점	한계점
Nutanix (NKP)	가장 성숙한 HCI 인프라, VM+K8s 완전 통합, Prism UI 단일화	Nutanix HCI 기반 필요 (전용 하드웨어 or OEM)
VMware vSAN	안정성과 호환성, 기존 환경과 연계 우수	고비용, vCenter + vSAN 구성 복잡
OpenShift	K8s + VM 완전 통합(OCP Virt), 보안 및 정책 중심 인프라	고비용, 높은 기술 장벽, RHEL 생태계 의존
Rancher + Harvester	오픈소스 기반 유연성, 저비용, K8s 관리에 적합	기능 성숙도 및 대규모 환경에 대한 검증이 아직 없음

HCI 조합 및 권장

시나리오 별로 구성하면 다음과 같이 구성을 권장한다.

시나리오	권장 HCI 플랫폼	비고 설명
운영 단순화 & VM 중심 환경	Nutanix (NKP + AOS/AHV)	Prism UI, 자동화, 성숙도 가장 높음
오픈소스 기반 DevOps 환경	SUSE Harvester + Rancher	YAML/Helm 기반 배포에 적합, 중/대규모 환경 적합
보안 규제 중심 기업 (공공/금융)	OpenShift Virtualization	K8s + VM 보안 정책 통합(SCC, SELinux 등). 단, 관리기능 부족
기존 VMware 중심 조직	vSphere + vSAN + NSX 조합	높은 성숙도, 그러나 라이선스 비용과 운영 복잡성 존재
파일럿/경량 K8s + VM 환경	Harvester (단독)	빠른 구축, 단일 노드 가능. OpenShift Virtualization보다 많은 기능 제공. 하지만, 여전히 부족한 관리 기능.