

클라우드 네이티브 개발로 전환

기존 애플리케이션을 현대화하여 클라우드 네이티브를 향한 경로 구축

컨테이너에서 기존 애플리케이션을 실행할 때의 이점

이식성: 인프라에서 애플리케이션을 분리하고, 컨테이너를 지원하는 모든 플랫폼에서 애플리케이션을 실행합니다.

확장성: 수요에 대응하기 위해 필요에 따라 유연한 확장 또는 축소가 가능하며, 리소스 활용도가 높습니다.

유연성: 필요하지 않을 때 리소스를 묶어둘 필요 없이, 더욱 손쉽게 컨테이너를 배포하여 필요 시 테스트 환경을 조성할 수 있습니다.

언어 및 기술에 종속되지

장점: 컨테이너는 다양한 언어, 데이터베이스, 프레임워크, 툴링을 선택할 수 있도록 지원합니다. 오래된 코드 및 신규 작성 코드 등 모든 코드를 컨테이너 내부에서 종속성을 분리하고 캡슐화하며, 컨테이너에 어떤 요소가 포함되었는지 상관없이 동일한 방식으로 관리합니다. 그 결과, 컨테이너는 보다 현대적인 기술과 함께 레거시 기술을 구동할 수 있습니다.

소개

소프트웨어 개발은 비즈니스에 필수적인 요소입니다. 애플리케이션은 비즈니스를 차별화하며 새로운 서비스와 수익 창출, 시장 확장을 향한 길을 열어줍니다. 더 빠른 제공을 위해서는 개발자 생산성을 높여야 합니다. 개발 속도가 제한되는 원인은 오래된 코드뿐만 아니라 구식 인프라, 개발 사례, 절차까지 포함하는 경우가 많습니다. 릴리스는 드물게 이루어지고, 상당한 수동 작업이 필요하며, 테스트가 적합하지 않은 경우도 있습니다.

클라우드 네이티브 개발은 조직이 혁신적인 애플리케이션을 더 빨리 제공할 수 있도록 지원합니다. 그러나 클라우드 네이티브 개발은 단순한 기술 이상을 필요로 하며, 인력과 프로세스를 모두 포함해야 합니다.

클라우드 네이티브 개발 사례

클라우드 네이티브 애플리케이션은 배포 위험을 낮추면서 속도와 유연성, 품질을 높이기 위해 클라우드 컴퓨팅 모델을 활용하도록 설계되었습니다. 클라우드 네이티브 방식은 이름에도 불구하고 이러한 방식은 애플리케이션이 배포되는 위치가 아닌, 애플리케이션이 구축, 배포 및 관리되는 방식에 중점을 두고 있습니다.

클라우드 네이티브 개발은 마이크로서비스와 같은 모듈식 아키텍처 접근 방식, 현대적인 컨테이너 기반 및 애플리케이션 프로그래밍 인터페이스(API) 기반 인프라, DevOps 사례를 결합하여 지속적인 애플리케이션 제공을 지원하는 애플리케이션 개발 및 배포 방식입니다. 클라우드 네이티브 애플리케이션과 마이크로서비스는 종종 함께 언급되지만, 동일한 원칙으로 더 나은 모듈식 모놀리스 애플리케이션을 구축하는 데 집중하여 클라우드 네이티브 접근 방식의 이점을 누리는 조직이 많습니다.

- 탄력적으로 결합된 서비스 기반 아키텍처
- 경량화되고 기술에 구애받지 않는 구성 요소 간 API 커뮤니케이션
- 환경 전반에서 이식성을 제공하면서 애플리케이션과 서비스를 빠르게 배포하고 확장하여 용량 수요를 충족할 수 있는 컨테이너 기반 인프라
- 협업을 통해 애플리케이션을 구축하고 제공하는 데 중점을 둔 지속적 통합/지속적 제공(CI/CD) 및 DevOps 사례를 활용한 애자일 개발 방식

변화를 위한 포지셔닝: 기존 애플리케이션을 컨테이너로 이동

오래된 인프라에 배포되어 있는 기존 애플리케이션으로 인해 클라우드 네이티브 개발로 이동하기가 복잡해질 수 있습니다. 또한, 기존 인프라에 있는 고정되고 제한적인 리소스가 성장을 억제하는 요소로 작용할 수 있습니다. 지원되는 최신 미들웨어 및 기타 소프트웨어 버전은 오래된 인프라에서 구동되지 않을 수도 있습니다.

컨테이너를 사용해 기존 애플리케이션을 재배포하면 변화를 가로막는 장애 요소를 없애고 진화를 위한 유연성을 제공할 수 있습니다. 기존 애플리케이션을 컨테이너로 이동하면 많은 이점을 얻을 수 있습니다.

- 컨테이너는 기술에 구애를 받지 않기 때문에 컨테이너 내부에서 사용되는 기술이 얼마나 오래 되었는지, 그 유형은 무엇인지를 고려할 필요가 적어집니다. 개발자는 자유롭게 적합한 태스크에 대해 올바른 툴로 기술을 활용할 수 있습니다. 또한 기존 및 신규 기술 모두 개별 컨테이너를 사용해 동일한 플랫폼에 배포할 수 있습니다.



표준 컨테이너 오케스트레이션 툴로 사실상 자리잡은 쿠버네티스는 스케일에 따라 대량의 컨테이너를 구동하는 Google의 경험들 기반으로 컨테이너의 배포와 관리를 자동화하는 오픈소스 솔루션입니다. 쿠버네티스는 자동으로 컨테이너를 재시작하고, 각기 다른 호스트에서 컨테이너를 다시 스케줄링하고, 자동 스케일링과 같은 활용 사례에 대해 컨테이너를 복제하는 등 자동화된 자가 복구(Self-healing) 메커니즘을 사용합니다. 쿠버네티스는 기본적으로 Linux® 컨테이너와 연동됩니다.

수년간 Red Hat은 Google 다음으로 쿠버네티스 프로젝트에 가장 크게 기여하는 기업으로 거론되어 왔습니다.¹ 2015년부터 쿠버네티스 프로젝트는 Cloud Native Computing Foundation에서 관리되고 있습니다. 쿠버네티스는 개방성 덕분에 업계에서 널리 채택되고 신속한 혁신을 촉진했으며, 쿠버네티스에 기반하는 오픈소스 프로젝트를 생성하게 되었습니다.

- 최신 미들웨어, 애플리케이션 서비스, 그리고 인공지능/머신 러닝(AI/ML)과 같은 신규 기술을 컨테이너 플랫폼에 배포하여 현대화와 통합을 위한 기회를 창출합니다.
- 기존 애플리케이션이 인프라에서 분리되면 애플리케이션 이식성이 증가하여 간편하게 애플리케이션을 이동하거나 리소스를 추가할 수 있습니다.
- 현대적인 개발 방법론은 품질과 신뢰성을 개선하기 위해 자동화된 테스트에 의존하는 경향이 있습니다. 모든 구성 요소가 컨테이너에서 구동되는 경우, 새로운 소프트웨어의 개발을 지원하기 위한 테스트 환경을 훨씬 쉽게 조성할 수 있습니다. 컨테이너 플랫폼이 셀프 서비스를 지원하는 경우, 개발자는 다른 사람이 환경을 프로비저닝할 때까지 기다리지 않고도 필요할 때 자체적으로 테스트 환경을 조성하거나 해제할 수 있습니다.

스트랭글러(Strangler) 패턴은 기존 모놀리식 애플리케이션을 성공적으로 현대화하는 데 사용되어 온 접근 방식입니다. 이 방법은 기능의 작은 부분을 식별하여 클라우드 네이티브 형태의 마이크로서비스, 애플리케이션 서비스 또는 서버리스 기능으로 대체합니다. 모든 요소가 컨테이너 플랫폼에서 구동되고 있는 경우 현대화할 기존 애플리케이션과 함께 새로운 소프트웨어 구성 요소를 간편하게 배포할 수 있습니다.

Red Hat OpenShift: 클라우드 네이티브 애플리케이션을 구축하기 위한 기반

쿠버네티스는 컨테이너화된 애플리케이션을 구동하기 위한 실질적인 표준으로 자리 잡았습니다. Red Hat® OpenShift®는 퍼블릭 클라우드의 간소화 및 자동화를 제공하는 엔터프라이즈급 쿠버네티스 컨테이너 플랫폼입니다. 하지만 쿠버네티스는 주로 개발자가 아닌 IT 운영팀의 필요에 따라 구축됩니다. Red Hat OpenShift는 컨테이너 구축 및 구동을 위한 컨테이너 플랫폼으로, 개발자와 운영자가 모두 사용할 수 있도록 설계되었습니다. Red Hat OpenShift는 쿠버네티스를 확장하여 개발자의 작업 방식과 개발 라이프사이클에 맞게 조정합니다. 여기에는 소스 코드에서 컨테이너를 자동 구축하는 단계부터 새로 구축된 애플리케이션을 컨테이너에 배포하는 단계까지 포함됩니다.

개발자는 Red Hat OpenShift를 컨테이너 구축 및 구동을 위한 셀프 서비스 플랫폼으로 사용하여 필요에 따라 개발 및 테스트 환경을 조성하고 해제합니다. 운영자는 Red Hat OpenShift를 사용해 프로덕션 환경을 모니터링하고 관리합니다. 하나의 시스템에 대한 개발자와 운영자의 관점이 다르다는 점을 고려하여 Red Hat OpenShift는 그래픽 사용자 인터페이스(GUI)에 개발자와 관리자의 관점을 모두 포함합니다. 즉, 관리자 커맨드라인 인터페이스(CLI)와 개발자 중심 CLI도 제공됩니다. 이렇듯 개발자와 관리자 모두에게 동일한 기본 시스템 및 메트릭에 대한 액세스를 부여하여 DevOps 사례를 촉진할 수 있습니다.

Red Hat OpenShift를 통해 개발자는 쿠버네티스를 학습하지 않아도 쿠버네티스에 애플리케이션을 배포할 수 있습니다. 예를 들어, 쿠버네티스는 여러 YAML 파일을 사용해 배포 및 설정 세부 정보를 저장하기 때문에 배우기 어렵습니다. 반면, Red Hat OpenShift를 활용하면 GUI와 배포 템플릿이 제공되어 YAML 파일을 직접 편집할 필요가 없습니다. 배포 템플릿은 애플리케이션의 소스 코드와 함께 버전 제어 시스템에 저장되어 애플리케이션의 배포를 간소화하고 코드형 인프라(IaC)를 지원합니다. Red Hat OpenShift를 사용하면 정교한 CI/CD 파이프라인을 손쉽게 생성해 배포 라이프사이클을 자동화할 수 있습니다.

¹ Stackalytics. "쿠버네티스," 2020년 8월 액세스됨.

Red Hat OpenShift는 퍼블릭 클라우드 서비스와 유사한 기능을 제공하는 동시에, 온프레미스와 퍼블릭 클라우드 모두에서 개발하고 구동할 수 있는 유연성을 지원합니다. 개발자는 퍼블릭 클라우드에서 서비스로서 구동되는 Red Hat OpenShift를 조직의 자체 인프라 또는 자신의 노트북에서도 사용할 수 있습니다. 독립적인 개발 환경을 선호하는 개발자들은 Red Hat CodeReady Containers를 다운로드하여 최소한으로 사전 구성된 Red Hat OpenShift 환경을 로컬에서 실행할 수 있습니다.

하이브리드 클라우드의 기존 애플리케이션 및 클라우드 네이티브 애플리케이션을 위한 플랫폼

기존 애플리케이션을 컨테이너로 이전할 때의 과제는 해당 애플리케이션이 클라우드 네이티브가 아니며, 데이터, 로그, 그리고 경우에 따라 설정에 대한 퍼시스턴트 스토리지가 필요하다는 점입니다. 그러나 컨테이너는 스테이트풀 애플리케이션용으로 설계되지 않았습니니다. 영구적인 스토리지가 필요한 기존 애플리케이션을 수용하기 위해 Red Hat OpenShift는 실행 시 자동으로 애플리케이션의 컨테이너에 연결되는 유연한 퍼시스턴트 스토리지 볼륨을 제공합니다. 블록, 파일 또는 오브젝트 액세스 방법으로 사용 가능한 소프트웨어 정의 퍼시스턴트 스토리지를 제작하는 데 [Red Hat OpenShift Container Storage](#)를 사용할 수 있습니다.

또한 규제, 데이터 지역성, 보안, 성능 등의 이유로 온프레미스에 남아 있어야 하는 애플리케이션은 여전히 Red Hat OpenShift를 애플리케이션 플랫폼으로 활용할 수 있습니다. 조직에서는 자체 데이터센터에 Red Hat OpenShift를 배포할 수 있습니다. 컨테이너화하기 어려운 애플리케이션을 위해 Red Hat OpenShift는 단일 인터페이스에서 가상 머신과 컨테이너화된 애플리케이션을 관리하는 방법도 도입했습니다.

퍼블릭 클라우드에서 Red Hat OpenShift를 사용하면 기본 클라우드 또는 컨테이너 플랫폼의 세부 정보를 추상화하여 조직이 하이브리드 및 멀티클라우드 배포로 전환할 때 개발자와 운영자에게 도움을 줍니다. 개발자는 컨테이너 기반 애플리케이션을 구축하고 배포하기 위해 하나의 환경만 학습하면 됩니다.

Red Hat OpenShift는 기존 및 신규 애플리케이션 모두에 대해 공통의 일관된 애플리케이션 플랫폼을 제공합니다. 이러한 애플리케이션 이식성과 배포 유연성 덕분에 조직은 가장 적합한 환경에서 애플리케이션을 실행할 수 있습니다.

클라우드 네이티브 개발을 위한 툴

Red Hat은 개발자가 컨테이너에서 구동할 애플리케이션을 구축할 수 있도록 지원하는 다양한 툴을 제공합니다. 이러한 툴은 Red Hat OpenShift에서 제공하는 기능을 기반으로 통합되어 있습니다.

Red Hat CodeReady Workspaces는 클라우드 네이티브 개발 환경을 위해 Red Hat OpenShift에서 실행되는 쿠버네티스 네이티브 개발자 워크스페이스 서버로서, 개발자가 로컬 머신에 소프트웨어를 설치하거나 코드를 복사하지 않아도 되는 브라우저 기반 통합 개발 환경(IDE)을 제공합니다.

Red Hat CodeReady Studio는 컨테이너와 멀티플 프로그래밍 모델을 위한 광범위한 툴링 세트를 갖춘 기존 데스크톱 IDE입니다. 이 가운데 개발자는 원하는 툴을 선택하여 사용할 수 있습니다. Red Hat은 OpenShift Connector를 비롯해 Microsoft Visual Studio Code를 위한 확장 기능 및 Java™, YAML, XML용 언어 확장을 제공합니다.

애플리케이션 구동을 위한 컨테이너 구축에는 언어, 런타임, 프레임워크, 애플리케이션 서버가 필요합니다. 기본 이미지에 대한 소스는 여러 개가 있지만, 알려진 소스에서 이를 획득하는 것은 쉽지 않습니다. Red Hat은 알려진 취약성이 없는 더욱 안전한 최신 상태의 기본 이미지를 제공하는 신뢰할 수 있는 소스입니다.

Red Hat Runtimes은 Red Hat OpenShift 통합 런타임 컬렉션으로, 클라우드 네이티브 개발을 간소화합니다. 런타임은 Quarkus, Spring Boot, Vert.x, Node.js, Thorntail을 지원하므로 개발자는 적합한 태스크에 대해 올바른 툴을 선택할 수 있습니다. Quarkus는 빠르고 경량화된 마이크로서비스와 서버리스 애플리케이션을 구축하기 위한 쿠버네티스 네이티브 Java 스택입니다. Red Hat Container Catalog는 개발자가 기본 이미지로 사용할 수 있도록 검증된 컨테이너 이미지의 추가 라이브러리를 제공합니다.

애플리케이션 제공을 간소화하기 위해 Red Hat Middleware는 컨테이너로 패키징되며 Red Hat OpenShift에서 구동할 준비가 되어 있습니다. **Red Hat Middleware 포트폴리오**에는 스트리밍, 브로커 기반 메시지, 통합 서비스, SSO(Single Sign-On), API 관리, 프로세스 자동화, 의사 결정 관리가 포함되어 있습니다.

Red Hat OpenShift로 자동화 구축 및 배포

마이크로서비스 기반 애플리케이션과 함께, 수십 또는 수백 개의 서비스를 개별적으로 릴리스하고 배포할 수 있습니다. 이 모든 코드를 더욱 자주 안정적으로 배포하려면 구축, 테스트, 배포를 자동화하는 것이 필수적입니다. 개발자는 Red Hat OpenShift를 사용해 CI/CD를 위한 자동화된 빌드를 설정할 수 있습니다. 새 코드가 소스 코드 버전 제어 시스템에 체크인될 때마다 빌드가 자동으로 시작되며, 빌드가 성공적으로 완료되면 자동으로 배포되어 이전 버전을 대체합니다. 개발자는 복잡하게 처음부터 빌드 환경을 구축하지 않고도 Jenkins와 같은 익숙한 툴을 사용할 수 있습니다. 또한 Red Hat OpenShift 콘솔을 사용해 강력한 빌드 자동화 파이프라인을 만들 수 있습니다. 빌드 태스크는 Red Hat OpenShift에서 관리하는 컨테이너에서 쿠버네티스 작업으로 실행됩니다.

Red Hat OpenShift는 플랫폼에서 실행되는 애플리케이션 서비스에 대해 라우팅 및 로드 밸런싱 기능을 제공합니다. 이 기능을 통해 제어된 사용자 하위 집합에서 새 릴리스를 테스트할 수 있으므로 소프트웨어 릴리스와 관련한 리스크를 낮추는 고급 배포 패턴을 사용할 수 있습니다. 롤링, 블루 그린(Blue-Green), 카나리(Canary) 배포 등의 패턴이 가능합니다.

클라우드 네이티브 개발로 전환

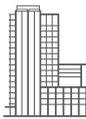
클라우드 네이티브 개발을 시작하려면 다음 사항들을 진행해야 합니다.

- 기존 애플리케이션을 클라우드 혹은 온프레미스로 구동 중인 Red Hat OpenShift에서 실행되는 컨테이너로 이전합니다. 컨테이너에서 배포된 현대적인 런타임과 미들웨어로 애플리케이션을 업데이트합니다.
- 빌드 파이프라인과 롤링 배포를 사용하는 CI/CD를 구현해 제공 시간을 단축합니다.
- 새로운 마이크로서비스로 분할하거나, 애플리케이션 서비스 또는 서버리스 기능으로 대체할 수 있는 기존 애플리케이션의 부분을 식별합니다. 클라우드 서비스는 물론 각기 다른 프레임워크와 아키텍처 접근 방식과 같이 적합한 태스크에 대한 올바른 툴 사용을 통해 신규 배포 속도를 높입니다.
- DevOps 사례를 구현하고 메트릭을 사용해 학습하고 개선합니다. 서비스 메시와 같은 Red Hat OpenShift의 네트워킹 기능을 사용해 애플리케이션 구성 요소 간 커뮤니케이션을 관리하고, 모니터링하고, 추적합니다.

자세히 알아보기

클라우드 네이티브 애플리케이션 개발에 대한 Red Hat의 접근 방식은 기존 애플리케이션을 현대화하고 기존 및 신규 애플리케이션과 데이터를 통합해 애플리케이션 개발과 배포를 가속화할 수 있도록 지원합니다. Red Hat의 [클라우드 네이티브 개발 솔루션](#)에 대해 자세히 알아보세요.

한국레드햇 홈페이지 <https://www.redhat.com/ko>



RED HAT 정보

Red Hat은 세계적인 엔터프라이즈 오픈소스 솔루션 공급업체로서 커뮤니티 기반 접근 방식을 통해 신뢰도 높은 고성능 Linux, 하이브리드 클라우드, 컨테이너, 쿠버네티스 기술을 제공합니다. 또한 고객으로 하여금 신규 및 기존 IT 애플리케이션을 통합하고, 클라우드 네이티브 애플리케이션을 개발하며, 업계를 선도하는 Red Hat의 운영 체제를 기반으로 표준화하는 동시에 복잡한 환경의 자동화, 보안 및 관리를 실현할 수 있도록 지원합니다. Red Hat은 전세계 고객에게 높은 수준의 지원과 교육 및 컨설팅 서비스를 제공하여 권위있는 어워드를 다수 수상한 바 있으며, Fortune 선정 500대 기업의 신뢰를 받는 어드바이저로 인정받고 있습니다. 또한 기업, 파트너, 오픈소스 커뮤니티의 전략적인 파트너로서 고객들이 디지털 미래에 대비할 수 있도록 지원하고 있습니다.

 www.facebook.com/redhatkorea
구매문의 080 708 0880
buy-kr@redhat.com