



코끼리에게 춤추는 방법 가르치기

핵심 요약

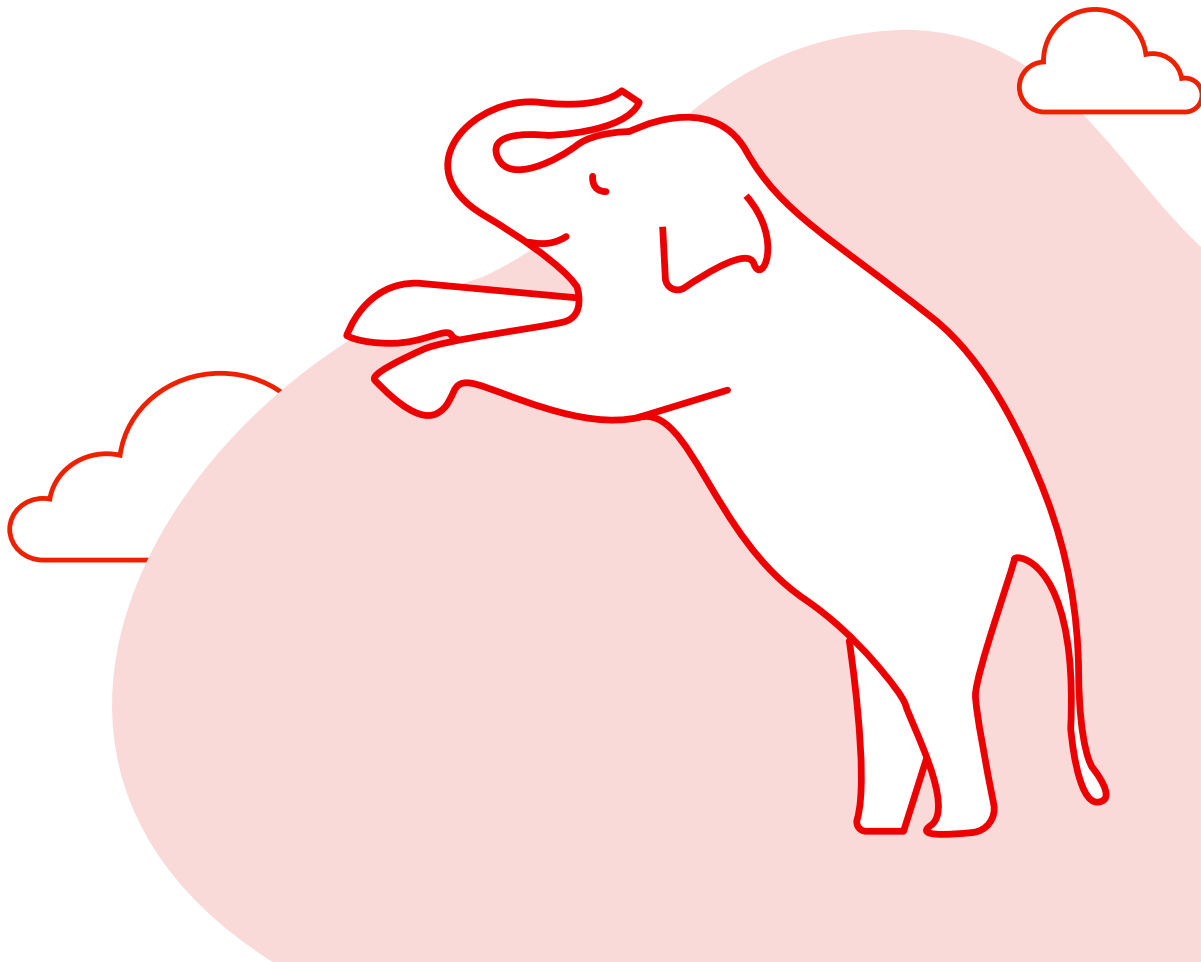
팀, 프로세스, 애플리케이션 전반의 지향적
진화(Intentional evolution)

Burr Sutter, Red Hat 개발자 경험 담당 디렉터

Deon Ballard, Red Hat 포트폴리오 제품 마케팅 매니저

Marty Wesley, Red Hat 포트폴리오 제품 마케팅 디렉터

Mike Randall, Red Hat 글로벌 오퍼레이션 공유 서비스 디렉터



소개

방 안에 있는 코끼리

디지털 트랜스포메이션은 비즈니스의 전략적인 변화를 의미합니다. 이는 핵심 서비스를 신속하게 변경하여 새로운 요구 사항을 충족하는 데 초점을 두지만, 조직에서 디지털 트랜스포메이션을 수행하기 위해 반드시 수반되어야 하는 공통적인 변경 사항은 없습니다. 또한, 디지털 트랜스포메이션은 컨테이너와 마이크로서비스 같은 새로운 아키텍처나 DevOps와 같은 새로운 프로세스의 도입을 의미하기도 합니다. 그렇지만, 손에 넣을 수 있는 특정한 대상이 아니라 각 조직에서 고유한 목적에 맞게 정의해야 하는 개념이 바로 디지털 트랜스포메이션입니다.

방 안에 있는 코끼리는 조직의 현재 기술 환경을 뜻합니다. 모든 조직이 마이크로서비스와 DevOps에 기반하는 현대적인 기술 환경에서 작업하는 것을 선호하지만, 대부분의 조직은 재구축이 필요한 여러 인프라 레이어 및 계층 구조로 이루어져 있습니다. 이러한 기업에서는 현재의 코끼리가 날렵한 발레리나처럼 춤을 출 수 있도록 가르쳐야 하는 어려운 과제를 해결해야 합니다.

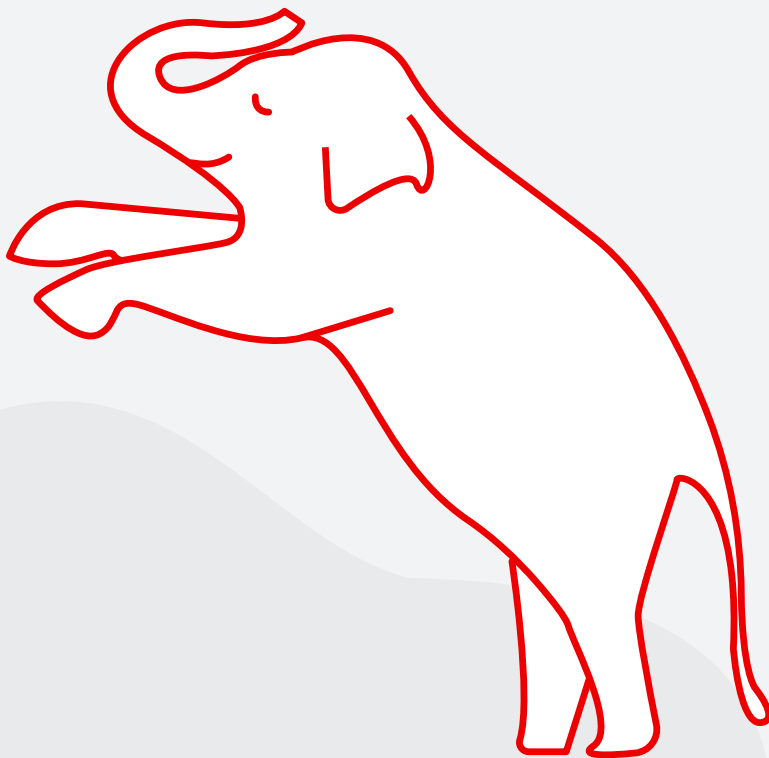
조직에 대한 디지털 트랜스포메이션을 구현할 수 있는 단일 아키텍처 패턴이나 기술 플랫폼은 없습니다. 디지털 트랜스포메이션을 성공적으로 실현한 기업들은 자사의 목표를 가장 명확하게 이해하고 이러한 목표를 달성하는 데 필요한 변경 사항을 식별해 구현한 기업들입니다.

예를 들면, 다음과 같습니다.

- [아르헨티나 보건부](#)는 17개 지역에서 200만 명이 넘는 환자의 의료 데이터를 통합함으로써, 스케일은 물론 대응 능력을 확보하여 트랜잭션 용량이 1,200% 향상했습니다.
- [Ford Motor Company](#)는 지속적 통합/지속적 제공(CI/CD) 워크플로우를 통해 DevOps 프로세스로 전환하여 표준화된 개발 환경 및 셀프 서비스 프로비저닝으로 생산성을 높였습니다.
- [Via Varejo](#)에서는 개방형 사례를 활용하고 숙련된 팀을 구축해 Sale Simulator 애플리케이션의 유지 관리에 드는 비용과 수고를 줄이는 동시에, 협업을 지원하며 혁신적인 개발 접근 방식을 채택하여 시의적절하게 유용한 솔루션을 제공했습니다.
- [Alliance Bank](#)는 민첩한 협업과 신속한 혁신 문화를 구축하여 팀에서 중복 프로세스를 재설정하고 지점의 물리적 공간을 평가할 수 있도록 지원합니다. 이 결과로, 계좌 개설 시간이 70% 단축되었고, 직원의 고객 지원 능력이 75% 향상되었으며, 지점에 새로운 커뮤니티 공간을 도입한 동시에 운영 비용을 절감했습니다.

이 기업들은 서로 다른 팀 구조, 프로세스, 애플리케이션, 아키텍처를 사용했습니다. 이 기업들이 성공할 수 있었던 이유는 내부 조직, 팀, 현재 기술적 채무 및 비즈니스 전략에 대한 평가로 시작한 다음, 계획에 따라 필요한 방향으로 나아갔기 때문입니다.

이것이 바로 코끼리를 춤출 수 있게 가르치는 과정입니다.

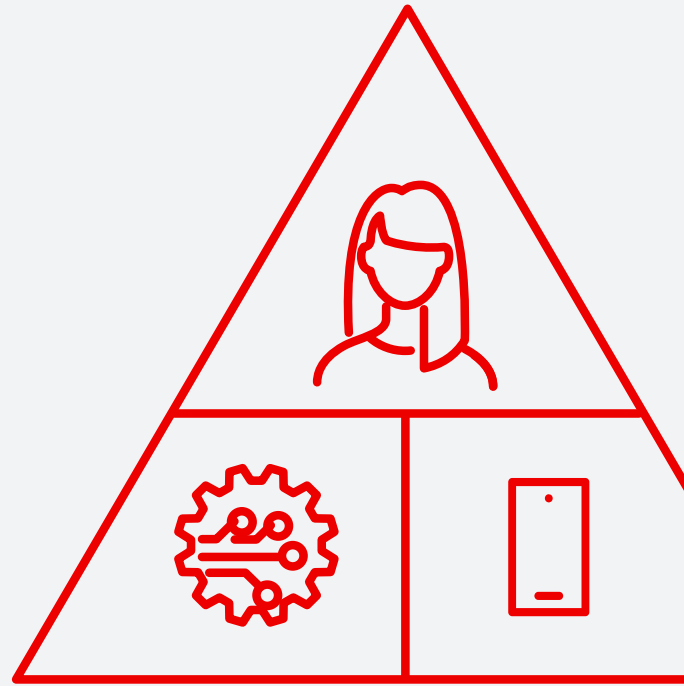


디지털 트랜스포메이션에 대한 고유한 접근 방식

디지털 트랜스포메이션을 단일 이니셔티브로 보는 경향이 있습니다. 이 접근 방식은 조직이 예전의 습관으로 돌아가는 것을 방지하려면 디지털 트랜스포메이션에 대한 변경 사항을 지속적으로 적용해야 한다는 점을 간과합니다. 또한, 디지털 트랜스포메이션의 최종 상태는 이미 정해져 있는 것이 아니라, 진화의 다음 단계를 실현하기 위해 각기 다른 단계를 진행하는 연속적인 과정이라고 볼 수 있습니다.

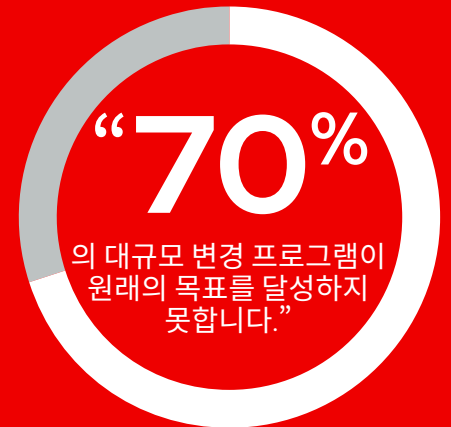
문화 우선

디지털 트랜스포메이션은 단순한 기술적 변화가 아니라, 사람과 프로세스의 변화도 수반합니다. 사실상 사람과 문화의 변화가 가장 중요한 요소라고 할 수 있습니다.



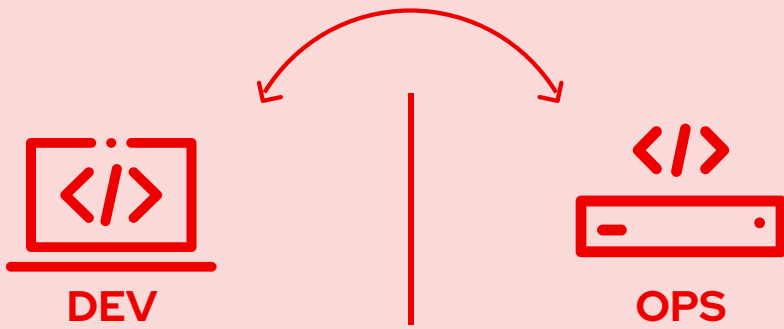
“...안타까운 현실은 대부분의 트랜스포메이션이 실패한다는 점입니다. 연구 결과에 따르면 복잡하고 규모가 큰 변화를 위한 프로그램 중 70%가 원래의 목표를 달성하지 못한다고 합니다. 이러한 실패의 일반적인 원인은 직원 참여 부족, 경영진의 지원 부족, 부서 간 협업의 부족 또는 부재, 책임 의식 부족 등입니다. 또한, 트랜스포메이션의 영향을 지속하기 위해서 관점이나 행동에 근본적인 변화가 있어야 합니다. 하지만 이를 실현하는 방법을 아는 리더는 많지 않습니다... 혁신을 추진하는 데 있어서 가장 어려운 부분은 무엇을 해야 하는지가 아니라 어떻게 실현할 것인지에 대한 결정입니다.”¹

¹ Michael Bucy, Adrian Finlayson, Greg Kelly, Chris Moyer, “트랜스포메이션 방법(The ‘how’ of transformation).” McKinsey & Company, 2016년 5월 9일



성공적인 디지털 트랜스포메이션으로 이어지는 문화란 하나로 정해져 있는 것이 아니지만, 보다 성공적인 트랜스포메이션을 실현할 수 있는 조직적 특성은 존재합니다. Harvard Business Review의 설문조사 결과, 보다 높은 수준의 협업, 적응력, 투명성, 포용성을 갖춘 기업이 더욱 성공적으로 디지털 트랜스포메이션을 실현한 것으로 나타났습니다.² 이러한 특성을 조직에 적용하면 디지털 트랜스포메이션에 반드시 성공한다고 보장하지는 못하지만 성공 확률은 높아질 것입니다.

² Harvard Business Review, “디지털 트랜스포메이션에 대한 재고(Rethinking digital transformation).” 후원: Red Hat, 2020년



주요 프로세스 변경 사항

문화적 변경을 수행한 후, 다음으로 진행해야 할 중요한 과정은 바로 여러 팀 간 커뮤니케이션 및 협업의 문화 속에서 운영되는 역동적인 소규모 팀을 통해 자동화된 애자일 프로세스로 전환하는 것입니다.

DevOps

DevOps는 디지털 트랜스포메이션 프로세스 변화의 기반입니다. DevOps, 또는 이와 유사한 애자일 프로세스는 개발 논의에 더 많은 이해관계자가 참여하도록 하고 고객의 서비스 이용 방식에 대해 보다 넓은 인사이트를 제공합니다. 이는 개방적인 커뮤니케이션 방식이 필요한 팀 간에 더욱 긴밀한 피드백 루프를 생성합니다. 이러한 신속한 프로세스 주기와 개방형 커뮤니케이션 사례는 디지털 트랜스포메이션 진화를 위한 기반입니다.

첫 번째로 적용할 수 있는 프로세스 변경 사항은 바로 기술 그룹이 일관된 환경을 신속하게 구축할 수 있도록 지원하는 셀프 서비스 기능을 도입하는 것입니다.

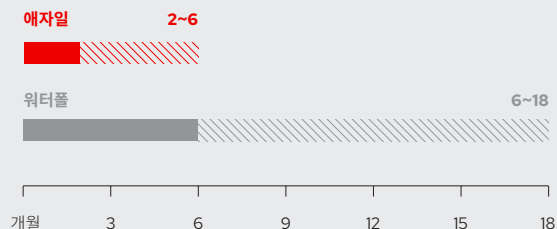
CI/CD 파이프라인

또 다른 중요한 프로세스 변경 사항은 CI/CD의 첫 번째 부분인 지속적 통합입니다. 지속적 통합을 통해 모든 코드 체크인은 자동화된 테스트 제품군과 결합되어 안정성과 기능을 검증합니다. 이 프로세스에서는 코드 품질 문제뿐만 아니라 비즈니스 요구 사항에 대한 애플리케이션의 적합성 관련 문제도 더 빨리 식별합니다.

지속적 통합 프로세스를 실행하는 조직에서는 지속적 배포를 구현할 수 있습니다. 즉, 변경 사항을 자동화된 방식으로 프로덕션 환경으로 푸시할 수 있습니다. 덕분에 고객은 새로운 서비스를 더 빨리 경험할 수 있으며, 개발 및 운영 팀에서는 더 빠르게 기능 및 보안 수정 사항을 롤아웃하여 더욱 안전하고 강력한 시스템을 제공할 수 있습니다.

오픈 컨테이너 플랫폼으로 빠르게 혁신한 Alliance Bank

Alliance Bank는 Red Hat® Consulting과 협력하여 전용 하드웨어에 대해서 마이크로서비스 기반 아키텍처로 전환했습니다. 그리고 애자일 전략을 지원하기 위해 팀 간 협업이 가능한 개방형 컨테이너 플랫폼을 선택했습니다. 이러한 플랫폼을 통해 Alliance Bank는 모든 지점의 상품과 서비스를 태블릿에 설치하여 직원과 고객이 함께 사용할 수 있습니다. 이제 Alliance Bank는 디지털 혁신을 이루었으며, 이를 신속하게 확대하여 고객의 요구 사항을 충족할 수 있습니다. 기존의 워터폴(Waterfall) 방법으로는 6~18개월이 걸리던 프로젝트 완료 기간이 애자일 방식을 통해 2~6개월로 단축되었습니다.



디지털 트랜스포메이션을 추진하는 여러 조직에서 이러한 프로세스 변경 사항을 채택했지만, 조직 고유의 상황에 맞게 프로세스를 변경해야 한다는 점을 기억해야 합니다. 조직 전반의 민첩성을 높이면 새로운 아이디어와 변화하는 시장 상황에 빠르게 대응할 수 있습니다.

아키텍처에 대한 재고

디지털 트랜스포메이션의 궁극적인 목표는 고객에게 유용하며 조직과 관련된 애플리케이션을 구축하는 것입니다.

유익한 애플리케이션은 다음과 같은 일련의 특성을 지니고 있습니다.

- 사용자에 대한 빠른 대응 능력
- 핵심 비즈니스 기능 또는 목적 반영
- 환경의 동적인 변화에 대한 능동적 또는 반응적 대응
- 환경 전반 연결
- 기능을 빠르게 추가 또는 수정할 수 있는 경량화 및 유연성

모놀리식 애플리케이션과 현대적인 마이크로서비스 애플리케이션 모두에서, 기술 아키텍처의 목표는 의미 있는 애플리케이션 서비스를 손쉽게 제공해야 한다는 점입니다.

코끼리에게 춤추는 방법 가르치기

현재의 단계 선택

디지털 트랜스포메이션 전략 계획을 시작할 때, 다음과 같은 사항을 스스로 질문해 보시기 바랍니다.

- 현재 귀하의 팀 또는 사업부는 무엇인가요?
- 이들 그룹 간의 커뮤니케이션 패턴은 어떻게 되나요?
- 계획 주기에 현재 누가 참여하고 있나요?
- 기능 측면에서, 기존 애플리케이션 아키텍처는 희망하는 애플리케이션 아키텍처에 얼마나 근접해 있나요?
- 주요 보안 결함에 얼마나 빠르게 대응할 수 있나요? 몇 주, 며칠 또는 몇 분이 걸리나요?
- 조직 내 리스크 또는 내결함성 수준은 어느 정도인가요?
- 귀사의 자료 및 정보 흐름을 얼마나 정확하게 파악하고 있나요?
- 고객 또는 운영 팀의 요구에 부응하기 위해 얼마나 자주 업데이트를 릴리스할 수 있어야 하나요?
- 비즈니스 목표 또는 개발 요건에 따라 필요한 새로운 기능은 무엇인가요?

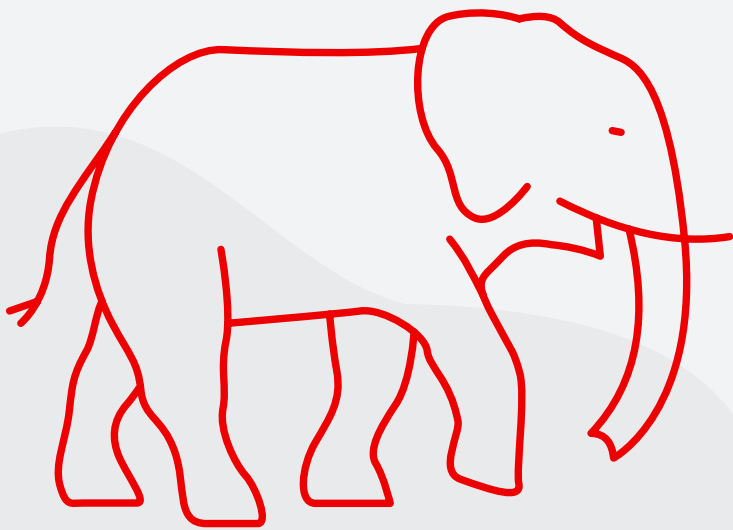
운영 원칙 정의

경영진이 지지하고 팀 전반의 지원을 받는 일련의 핵심 원칙을 수립하면 디지털 트랜스포메이션 이니셔티브를 강화하고 팀을 단결시킬 수 있습니다. 이러한 핵심 원칙을 예를 들면 다음과 같습니다.

- 조직과 사람이 우선입니다.
- 실험은 필수적이며 적절하게 이루어집니다.
- 실패가 발생할 수 있습니다. 실패로부터 배우고 개선하는 데 집중합니다.
- 항상 신뢰할 수 있습니다.
- 투명성을 지킵니다.

미래의 민첩성을 위한 아키텍처 설계

프로세스의 간소화 또는 의미 있는 애플리케이션 생성 등 모든 분야에서, 아키텍처의 기반은 변화에 따라 적응할 수 있도록 구축되어야 합니다. 명확하고 간소한 개발 및 배포 프로세스를 구축하면 민첩성과 유연성을 갖출 수 있으며, 요구 사항의 변화에 따라 애플리케이션도 진화할 수 있습니다.



결론

시간이 지나면서 엔터프라이즈 애플리케이션은 이해하기 힘들고 업데이트가 번거로우며 변화에 적응하는 속도가 느린 전형적인 코끼리로 진화하는 경향이 있습니다. 하지만 이 엔터프라이즈 애플리케이션들은 조직에 중요한 핵심 업무와 매출이 창출되는 운영을 실행합니다. 마치 방 안에 있는 코끼리처럼 말입니다.

최종 상태에 대한 명확한 비전과 이를 지원하는 사람, 프로세스, 기술의 변화만 있다면 이 코끼리들을 민첩하고 혁신적으로 변화하도록 훈련할 수 있습니다. 이 전략은 바로 진화 과정으로서 디지털 트랜스포메이션입니다. 이상적인 결과가 단 하나로 국한되지는 않습니다. 각 혁신 경로는 해당 기업의 고유한 목적과 특성을 반영합니다.

귀사의 비즈니스 요구에 가장 일치하는 진화 수준을 기반으로 하는 디지털 트랜스포메이션 전략을 수립하십시오.

자체 문화를 확립하는 데 초점을 맞추고, 기술 변화와 이와 관련한 프로세스 변화 사이의 균형을 유지함으로써 팀이 기술을 완벽하게 지원하도록 해야 합니다.

애플리케이션 및 아키텍처를 평가하고, 독립적인 서비스를 격리하거나 개발하여 비즈니스 우선 순위가 변경되거나 추가되면 이에 맞춰 조정될 수 있는 애자일 아키텍처를 구축합니다.

마지막으로, 혁신 역량을 강화합니다. 이는 리스크와 장애를 어느 정도 허용하는 것을 뜻하며, 여기에는 시간, 비용 및 인프라 등의 리소스를 확보하기 위한 규정이 필요합니다. 실험은 혁신을 위한 근간이며 디지털 트랜스포메이션의 성공을 위한 더 나은 기회를 제공합니다. 또한 많은 개발자들과 운영 인력들이 처음에 입사할 때의 희망, 즉 무엇인가를 창조하고 그 성장 과정을 볼 수 있는 역량을 회복할 수 있도록 합니다.

디지털 트랜스포메이션을 시작해 보세요.

