

Accelerate

State of DevOps 2021



Sponsored by



1장 핵심 요약

주요 결과 06

2장 비교 방법

소프트웨어 배포 및 운영 성과 09

3장 개선 방법

클라우드 15

SRE 및 DevOps 19

문서 21

보안 24

기술 DevOps 기능 26

코로나19 29

문화 31



4장

설문조사 대상

인구통계 및 기업통계

34

5장

최종 의견

6장

감사의 말씀

7장

저자

8장

방법론

9장

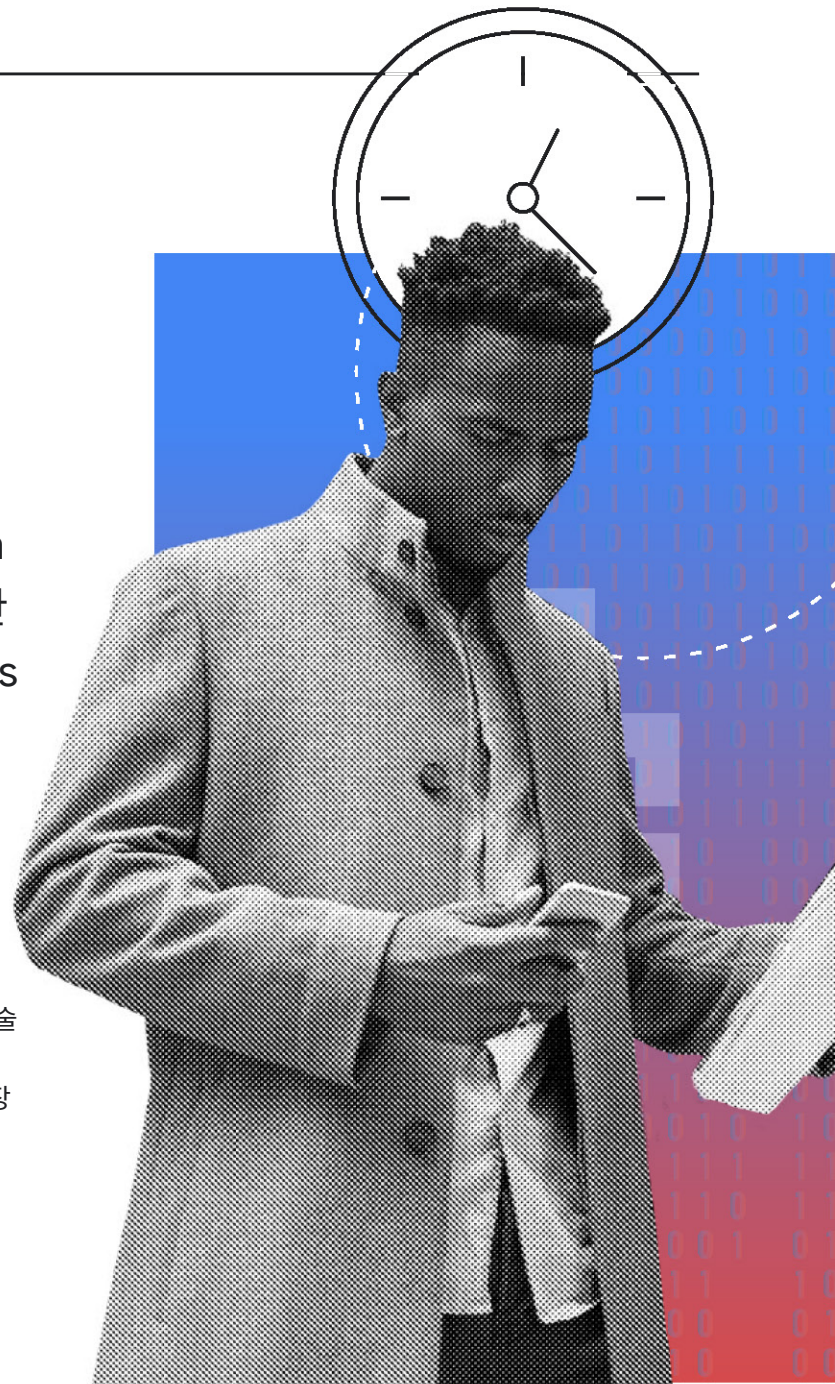
추가 자료

1장

핵심 요약

Google Cloud의 DevOps Research and Assessment(DORA)팀이 작성한 2021년 Accelerate State of DevOps Report는 전 세계 32,000여 명의 전문가를 대상으로 7년간 수집한 연구 및 데이터를 담고 있습니다.

본 연구에서는 소프트웨어 배포, 운영, 조직 성과를 주도하는 역량 및 관행을 조사합니다. 엄격한 통계 기술을 활용하여 기술 제공의 우수성과 강력한 비즈니스 성과로 이어지는 관행을 이해하고자 합니다. 이를 위해 기술을 개발하고 배포하는 가장 효과적이고 효율적인 방법에 대한 데이터 기반 통계를 제시합니다.



이번 연구 결과에서도 소프트웨어 배포 및 운영 성과의 우수성이 조직에서 기술 혁신 성과를 내는데 중요한 역할을 하는 것으로 확인되었습니다. 팀이 업계 대비 자체 평가를 할 수 있도록 클러스터 분석을 사용하여 의미 있는 성과 카테고리(예: 성과가 저조한 팀, 중간 팀, 높은 팀, 엘리트 팀)를 만들었습니다. 팀이 현재 성과를 업계와 비교한 후에는 예측 분석 결과를 사용하여 주요 결과를 개선하고 궁극적으로 상대 위치를 개선하도록 관행과 역량에 대한 목표를 세울 수 있습니다. 올해 보고서에서는 안정성 목표 달성, 소프트웨어 공급망 전반의 보안 통합, 양질의 내부 문서 작성, 클라우드의 잠재력을 최대한 활용하는 것의 중요성을 강조합니다. 또한 긍정적인 팀 문화가 코로나19 팬데믹에 따른 원격 근무의 영향을 완화할 수 있는지 살펴봅니다.

의미 있는 발전을 이루기 위해 팀에서는 지속적 개선의 철학을 채택해야 합니다. 벤치마크를 사용하여 현재 상태를 평가하고, 연구에서 조사한 역량을 기반으로 제약 조건을 식별하고, 이러한 제약을 완화하기 위한 개선 사항을 실험하세요. 실험에는 승리와 패배가 혼재하지만 팀에서는 두 시나리오 모두에서 얻은 교훈을 바탕으로 의미 있는 조치를 취할 수 있습니다.

주요 결과

01 최고 성과자는 성장을 이어가고 있으며 계속해서 기준을 높입니다.

엘리트 성과자는 연구에 참여한 팀의 26%를 차지하며 프로덕션 변경의 리드 타임을 줄였습니다. 업계가 계속해서 혁신에 속도를 내면서 팀에서는 큰 혜택을 보고 있습니다.

02 SRE와 DevOps는 상호 보완되는 철학을 갖고 있습니다.

사이트 안정성 엔지니어링(SRE)팀에서 제시한 현대적 운영 관행을 활용하는 팀이 더 높은 운영 성과를 보고합니다. 배포 및 운영 우수성을 모두 우선시하는 팀이 가장 뛰어난 조직 성과를 보고합니다.

03 더 많은 팀이 클라우드를 활용하고 있으며 그에 따른 큰 이점을 누리고 있습니다.

팀은 계속해서 워크로드를 클라우드로 이전하고 있으며 클라우드의 5가지 기능을 모두 활용하는 팀에서는 소프트웨어 배포 및 운영(SDO) 성과와 조직 성과가 향상됩니다. 팀이 각 제공업체의 고유한 기능을 활용할 수 있도록 멀티 클라우드 전략을 채택하는 경우도 증가하고 있습니다.

04 안전한 소프트웨어 공급망은 필수적이며 성과를 높입니다.

최근 몇 년 동안 악성 공격이 크게 증가함에 따라 조직은 수동적 관행을 따르는 것에서 벗어나 능동적 진단 조치를 취하는 태세로 전환해야 합니다. 소프트웨어 공급망 전반에서 보안 관행을 통합하는 팀은 소프트웨어를 빠르고 안정적이며 안전하게 배포합니다.

05 우수한 문서화 관행은 DevOps 기능을 성공적으로 구현하기 위한 기초입니다.

이 보고서에서는 처음으로 내부 문서의 품질과 이러한 품질에 기여하는 관행을 평가했습니다. 품질이 우수한 문서를 보유한 팀은 기술 관행을 더 잘 이행하고 전체적으로 더 높은 성과를 낼 수 있습니다.

06 긍정적인 팀 문화는 어려운 상황에서 번아웃을 완화합니다.

팀 문화는 소프트웨어를 배포하고 조직 목표를 이루거나 초과 달성하는 팀의 역량에 큰 차이를 가져옵니다. 생성적^{1,2} 문화를 갖춘 포용적인 팀은 코로나19 팬데믹 동안 번아웃을 덜 경험했습니다.

1 웨스트럼의 유형 조직 문화에서 생성적 팀 문화란 긴밀하게 협력하고, 사일로를 무너뜨리고, 실패에서 질문을 도출하고, 의사 결정의 위험을 공유하는 문화를 의미합니다.

2 Westrum, R. (2004). "A typology of organizational cultures." *BMJ Quality & Safety*, 13(suppl 2), ii22-ii27.



2장

비교 방법

팀이 업계의 다른 팀과 비교되는 방식이 궁금하신가요? 이 섹션에는 DevOps 성과에 대한 최신 벤치마크 평가가 포함되어 있습니다.

연구에서는 팀이 소프트웨어 시스템을 어떻게 개발, 배포, 운영하는지 조사한 다음 응답자를 엘리트 성과자, 높은 성과자, 중간 성과자, 저조한 성과자라는 4가지 성과 클러스터로 분류합니다. 팀의 성과를 각 클러스터의 성과와 비교하면 이 보고서 전체에 설명된 결과의 맥락에서 팀의 현재 위치를 확인할 수 있습니다.



소프트웨어 배포 및 운영 성과

조직은 끊임없이 변화하는 업계의 요구를 충족하기 위해 소프트웨어를 빠르고 안정적으로 배포 및 운영해야 합니다. 팀에서 더 빠르게 소프트웨어를 변경할수록 더 빠르게 고객에게 가치를 제공하고 실험을 실행하고 값진 피드백을 받을 수 있습니다. 이 보고서에서는 7년간의 데이터 수집 및 연구를 토대로 소프트웨어 배포 성과를 측정하는 4가지 측정항목을 개발하고 검증했습니다. 2018년 이후에는 운영 능력을 파악하기 위한 5번째 측정항목을 포함했습니다.

5가지 척도 모두에서 뛰어난 결과를 낸 팀은 탁월한 조직 성과를 보입니다. 이러한 5가지 측정항목을 **소프트웨어 배포 및 운영(SDO) 성과**라고 합니다. 이러한 측정항목은 시스템 수준의 결과에 초점을 맞추므로 전체 결과를 희생하면서 함수를 서로 맞추거나 로컬 최적화를 수행하는 것과 같은 소프트웨어 측정항목의 일반적인 함정을 피하는 데 도움이 됩니다.

소프트웨어 배포 성과 측정항목	엘리트 성과자	높은 성과자	보통 성과자	저조한 성과자
🕒 배포 빈도 작업하는 주요 애플리케이션 또는 서비스의 경우 조직에서 얼마나 자주 코드를 프로덕션에 배포하거나 최종 사용자에게 릴리스하나요?	주문형(하루에 여러 번 배포)	일주일에 한 번에서 한 달에 한 번 사이	한 달에 한 번에서 6개월에 한 번 사이	6개월에 한 번 미만
🕒 변경 리드 타임 작업 중인 주요 애플리케이션 또는 서비스의 경우 변경 리드 타임(예: 코드 커밋부터 프로덕션 환경에서 코드 실행 성공까지 걸리는 시간)은 얼마인가요?	1시간 미만	하루에서 일주일 사이	1개월에서 6개월 사이	6개월 이상
🕒 서비스 복원 시간 작업 중인 주요 애플리케이션 또는 서비스의 경우 사용자에게 영향을 미치는 서비스 사고 또는 결함(예: 예기치 않은 서비스 중단 또는 서비스 손상)이 발생할 때 서비스를 복원하는 데 시간이 보통 얼마나 걸리나요?	1시간 미만	1일 미만	하루에서 일주일 사이	6개월 이상
⚠️ 변경 실패율 작업 중인 주요 애플리케이션 또는 서비스의 경우 프로덕션에 대한 변경 또는 사용자에게 릴리스된 변경 중 몇 퍼센트가 서비스 저하(예: 서비스 장애 또는 서비스 중단으로 이어짐)를 초래하고 이후에 조치(예: 핫픽스, 롤백, 향후 수정, 패치)가 필요한가요?	0~15%	16~30%	16~30%	16~30%

배포 성과의 4가지 측정항목

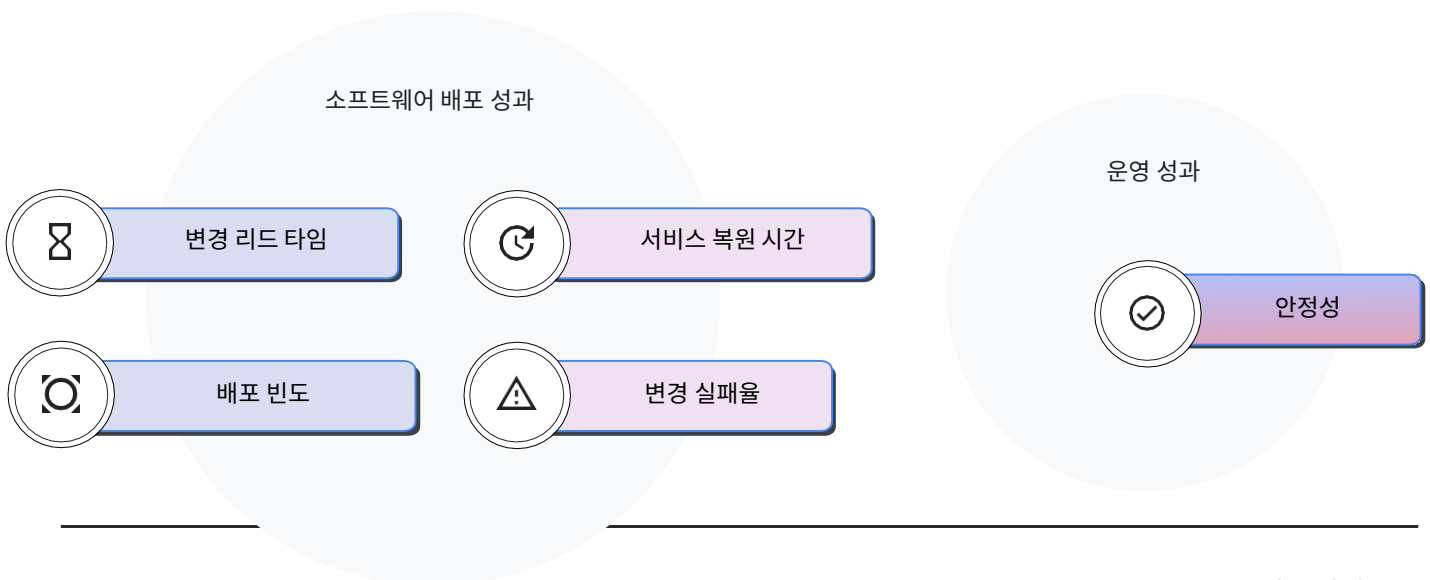
소프트웨어 배포 성과의 4가지 측정항목을 처리량과 안정성 측면에서 고려할 수 있습니다. 처리량은 **코드 변경의 리드 타임**(즉, 코드 커밋에서 프로덕션에 릴리스까지의 시간)과 **배포 빈도**를 사용하여 평가합니다. 안정성은 사고 발생 후 **서비스 복원 시간** 및 **변경 실패율**을 사용하여 평가합니다.

4가지 소프트웨어 배포 측정항목의 클러스터 분석에서도 4개의 성과 프로필(엘리트 성과자, 높은 성과자, 보통 성과자, 저조한 성과자)별로 처리량 및 안정성 기준에서 통계적으로 유의미한 차이가 다시 한번 확인됩니다. 이전과 마찬가지로 엘리트 성과자는 4가지 측정항목 모두에서 훨씬 더 나은 성과를 냈으며 저조한 성과자는 모든 영역에서 훨씬 더 나쁜 성과를 보였습니다.

5번째 측정항목: 가용성에서 안정성까지

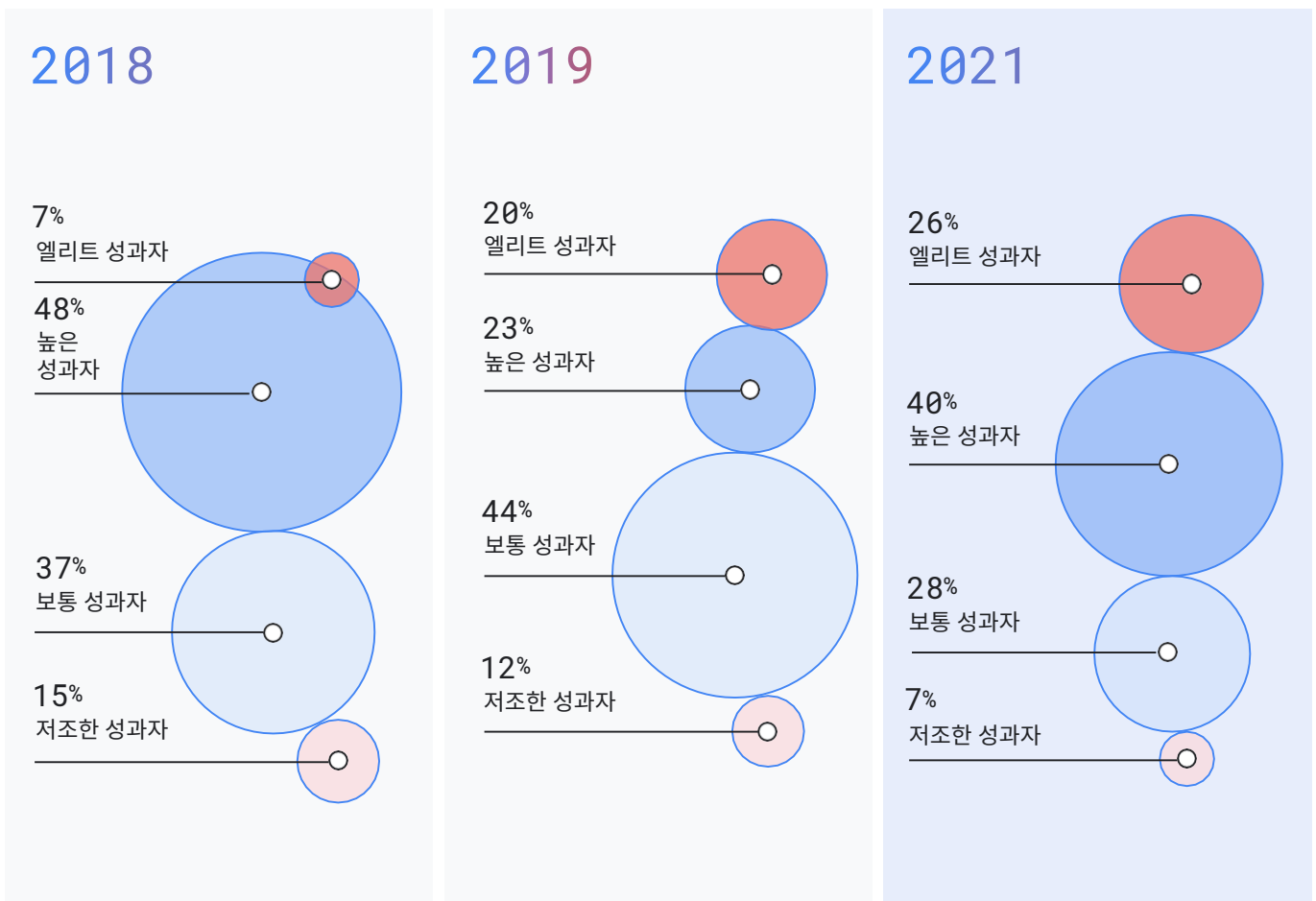
5번째 측정항목은 **운영 성과**를 나타내며 현대적 운영 관행의 척도입니다. 운영 성과의 주요 측정항목은 **안정성**으로, 팀이 운영하는 소프트웨어에 대한 약속과 주장을 지킬 수 있는 정도를 나타냅니다. 지금까지는 안정성보다 가용성을 평가했지만, 가용성은 안정성 엔지니어링에 초점을 두기 때문에 가용성, 지연 시간, 성능, 확장성이 보다 골고루 반영되도록 척도를 안정성으로 확장했습니다. 특히 응답자에게 안정성 목표를 달성하거나 초과하는 역량을 평가해 달라고 요청했습니다. 그 결과 배포 성과의 수준이 다양한 각 팀에서 운영 성과도 우선시할 때 성과가 향상된다는 점을 확인했습니다.

이전 보고서와 마찬가지로 특정 역량의 영향을 설명하기 위해 엘리트 성과자를 저조한 성과자와 비교했습니다. 다만 올해는 운영 성과의 영향을 설명하는 데 주안점을 두었습니다. 저조한 성과자부터 엘리트 성과자에 이르기까지 모든 배포 성과 카테고리에서 안정성 목표를 충족하거나 초과 달성하는 것을 우선시한 팀이 여러 결과에서 상당한 개선을 보였음을 확인했습니다.



발전을 위한 노력에 끊임없이 박차를 가하는 업계

매년 업계는 더 빠르고 안정적으로 소프트웨어를 배포하는 역량을 갖추기 위해 발전하려는 노력에 계속 박차를 가하고 있습니다. 이번 조사에서는 처음으로 엘리트 성과자와 높은 성과자가 응답자의 3분의 2를 차지했습니다. 또한 올해 엘리트 성과자들은 이전 평가와 비교했을 때 변경 리드 타임을 단축(예: 2019년 하루 미만에서 2021년 1시간 미만으로 개선)하여 기준을 또 다시 높였습니다. 엘리트 성과자만 전년대비 변경 실패율을 최소화한 것도 이번이 처음입니다. 이전에는 보통 성과자와 높은 성과자가 전년대비 변경 실패율을 최소화했습니다.



처리량

배포 빈도

엘리트 그룹은 이전과 마찬가지로 주문형 배포를 일상적으로 수행하고 하루에 여러 번 배포한다고 보고했습니다. 반면 저조한 성과자는 6개월에 1회 미만(연간 2회 미만) 배포한다고 보고했습니다. 2019년과 비교해서도 성과가 다시 퇴보한 것입니다. 연간 정규 배포 횟수는 엘리트 성과자의 경우 연 1,460회(하루 4회 x 365일)에 달하며 저조한 성과자는 연 1.5회(평균 1~2회)에 그치는 등 천차만별입니다. 이 분석에 따르면 엘리트 성과자의 코드 배포 빈도가 저조한 성과자의 973배에 이르는 것으로 추정됩니다.

변경 리드 타임

엘리트 성과자는 변경 리드 타임을 1시간 미만으로 보고하여 2019년보다 발전을 이루었습니다. 변경 리드 타임은 코드 커밋부터 프로덕션에 코드 배포 성공까지 걸린 시간으로 측정합니다. 엘리트 성과자는 2019년에 변경 리드 타임을 하루 미만으로 보고하여 성과가 향상된 것으로 나타났습니다. 엘리트 성과자와 달리 저조한 성과자는 리드 타임이 6개월 넘게 필요했습니다. 엘리트 성과자의 리드 타임은 1시간('1시간 미만'의 범위에서 보수적으로 상한값을 적용)이고 저조한 성과자의 리드 타임은 6,570시간(연간 8,760시간과 6개월간 4,380시간의 평균으로 추산)으로 엘리트 그룹의 변경 리드 타임이 저조한 성과자보다 6,570배 더 빠릅니다.

안정성

서비스 복원 시간

엘리트 그룹은 서비스 복원 시간을 1시간 미만으로 보고한 반면 저조한 성과자는 6개월이 넘는다고 보고했습니다. 이러한 계산에는 보수적인 범위를 선택했습니다. 높은 성과자의 경우 1시간, 저조한 성과자의 경우 1년(8,760시간)과 6개월(4,380시간)의 평균을 적용했습니다. 이를 토대로 계산하면 엘리트 성과자는 저조한 성과자보다 6,570배 빠르게 서비스를 복원합니다. 2019년과 비교한 서비스 복원 시간 성과는 엘리트 성과자의 경우 동일한 수준을 유지한 반면 저조한 성과자는 더 증가한 것으로 나타났습니다.

변경 실패율

엘리트 성과자는 0~15%의 변경 실패율을 보고한 반면 저조한 성과자는 16~30%의 변경 실패율을 보고했습니다. 두 범위의 변경 실패율 평균은 엘리트 성과자의 경우 7.5%이고 저조한 성과자의 경우 23%입니다. 저조한 성과자의 변경 실패율이 엘리트 성과자의 해당 비율보다 3배 더 높습니다. 올해 변경 실패율을 2019년과 비교할 때 엘리트 성과자는 동일한 수준을 유지하고 저조한 성과자는 개선되었으나 그 사이에 위치한 그룹에서는 악화되었습니다.

엘리트 성과자

엘리트 그룹을 저조한 성과자와 비교해보면 엘리트 성과자는...

973배 6,570배

더 빈번하게 코드 배포

더 빠른 리드 타임(커밋부터 배포까지)

예, 제대로 본
것이 맞습니다.
편집상의 오류가
아닙니다.

3배 6,570배

더 낮은 변경 실패율(변경 실패
가능성이 1/3 더 낮음)

더 빠른 사고 복구 시간

3장

개선 방법

SDO와 조직 성과를 어떻게 개선할 수 있을까요? 이 연구에서는 성과를 내는 역량을 강화하는 데 집중할 수 있도록 도와주는 증거 기반 지침을 제공합니다.

올해 보고서에서는 클라우드, SRE 관행, 보안, 기술 관행, 문화의 영향을 조사했습니다. 이 섹션 전체에서는 각 역량을 소개하고 다양한 결과에 미치는 영향을 중점적으로 살펴봅니다. DORA의 State of DevOps 연구 모델에 익숙한 분들을 위해 올해 모델과 이전의 모든 모델을 호스팅하는 온라인 리소스를 마련했습니다.³

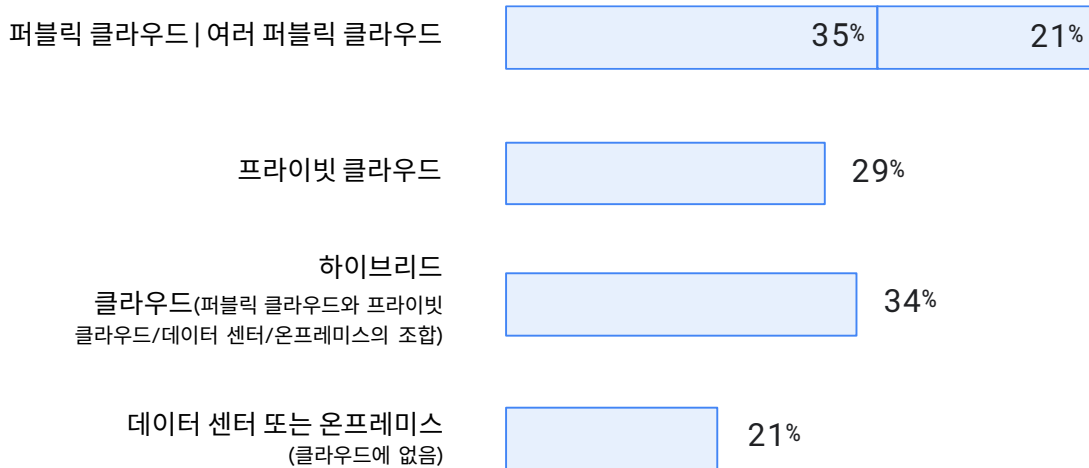


³ <https://devops-research.com/models.html>

클라우드

Accelerate State of DevOps 2019의 발견사항과 같이 점점 더 많은 조직에서 멀티 클라우드 및 하이브리드 클라우드 솔루션을 선택하고 있습니다. 설문조사에서는 응답자의 주요 서비스 또는 애플리케이션이 어디에 호스팅되는지를 물었으며 퍼블릭 클라우드의 사용이 증가하고 있는 것으로 나타났습니다. 응답자의 56%가 퍼블릭 클라우드를 사용한다고 답변했는데(여러 퍼블릭 클라우드 사용 포함) 이 수치는 2019년보다 5% 증가한 것입니다. 올해는 멀티 클라우드 사용에 대해서도 구체적으로 물었으며 응답자의 21%가 여러 퍼블릭 클라우드에 배포한다고 답했습니다. 응답자의 21%는 클라우드를 사용하지 않고 데이터 센터 또는 온프레미스 솔루션을 사용한다고 밝혔습니다. 마지막으로 응답자의 34%는 하이브리드 클라우드를 사용하고 29%는 프라이빗 클라우드를 사용한다고 답했습니다.

채택 현황



하이브리드 및 멀티 클라우드로 비즈니스 성과 가속화

올해 하이브리드 및 멀티 클라우드 사용이 증가하면서 기업에서 중시하는 성과에 상당한 영향을 미치고 있습니다. 하이브리드 또는 멀티 클라우드를 사용하는 응답자는 그렇지 않은 응답자보다 조직의 성과 목표를 초과 달성할 가능성이 1.6배 더 높았습니다. 또한 하이브리드 및 멀티 클라우드 사용자가 배포 빈도, 변경 리드 타임, 복구 시간, 변경 실패율, 안정성 측면에서 더 뛰어난 성과를 낼 가능성이 1.4배 높아 SDO에 대한 강력한 영향이 확인되었습니다.

멀티 클라우드를 채택하는 이유

2018년도 평가와 유사하게 응답자에게 여러 퍼블릭 클라우드 제공업체를 활용하는 이유를 물었습니다. 다만 올해에는 해당 항목을 모두 선택하는 방식을 이용하지 않고 응답자에게 여러 제공업체를 사용하는 주된 이유를 물었습니다. 4분의 1이 넘는(26%) 응답자가 각 클라우드 제공업체의 고유한 이점을 활용하기 위해서라고 답했습니다. 이는 응답자가 추가 제공업체를 선택할 때 현재 제공업체와 대안적인 제공업체 간의 차별점에 주목함을 나타냅니다. 멀티 클라우드로 전환하는 이유로 두 번째로 많이 나온 답변은 가용성(22%)이었습니다. 당연히 여러 클라우드 제공업체를 채택한 응답자는 안정성 목표를 충족하거나 초과 달성할 가능성이 1.5배 더 높았습니다.

여러 제공업체를 사용하는 주된 이유

각 제공업체의 고유한 이점 활용	26%
가용성	22%
재해 복구	17%
법적 규정 준수	13%
기타	08%
협상 전략 또는 조달 요구사항	08%
단일 제공업체에 대한 신뢰 부족	06%

벤치마크 변경사항

클라우드 인프라 구현 방법의 중요성

지금까지 연구 결과를 보면 모든 응답자가 동일한 방식으로 클라우드를 채택하는 것은 아닙니다. 클라우드를 얼마나 효과적으로 채택하는지에 따라 기업의 성과도 달라집니다. 이 연구에서는 미국 국립표준기술원(National Institute of Standards and Technology)에서 정의한 클라우드 컴퓨팅의 핵심 특성에 초점을 맞추고 이를 가이드로 사용하여 이러한 한계를 극복했습니다. 단순히 클라우드 채택이 SDO에 미치는 영향을 조사하기보다는 클라우드 컴퓨팅에 대한 NIST의 정의를 사용하여 핵심 관행이 SDO 성과에 미치는 영향을 조사했습니다.

팀이 클라우드 기술을 사용하는지 여부가 아니라 팀이 클라우드 서비스를 어떻게 구현하는지가 정말 중요하다는 점이 확인된 것은 이번이 세 번째입니다. 엘리트 성과자는 NIST가 정의한 클라우드 핵심 특성을 모두 충족할 가능성이 3.5배 더 높았습니다. 클라우드 인프라를 사용하고 있다고 답한 응답자의 32%만이 NIST에서 정의한 클라우드 컴퓨팅의 5가지 핵심 특성을 모두 충족한다는 데 동의하거나 강력히 동의했으며, 이는 2019년보다 3% 증가한 수치입니다. 전체적으로 NIST의 클라우드 컴퓨팅 특성 사용이 14~19% 증가했으며, 빠른 탄력성이 가장 크게 증가했습니다.

주문형 셀프서비스

소비자는 제공업체 측에 요구되는 인적 상호작용 없이 필요에 따라 자동으로 컴퓨팅 리소스를 프로비저닝할 수 있습니다.

광범위한 네트워크 액세스

기능을 널리 사용할 수 있으며 휴대폰, 태블릿, 노트북, 워크스테이션 등 여러 클라이언트를 통해 액세스할 수 있습니다.

73%

주문형 셀프서비스를
사용했다는 응답자 비율로,
2019년보다 16% 증가한 수치

74%

광범위한 네트워크 액세스를
사용했다는 응답자 비율로,
2019년보다 14% 증가한 수치

리소스 풀링

제공업체 리소스는 멀티 테넌트 모델에서 풀링되고, 물리적 리소스와 가상 리소스가 수요에 따라 동적으로 할당 및 재할당됩니다. 일반적으로 고객은 제공된 리소스의 정확한 위치를 직접 제어할 수 없지만 국가, 주 또는 데이터 센터와 같은 더 높은 수준의 추상화에서 위치를 지정할 수 있습니다.

빠른 탄력성

기능을 탄력적인 방식으로 프로비저닝 및 배포하여 수요에 따라 빠르게 수평으로 확장 또는 축소할 수 있습니다. 프로비저닝에 사용할 수 있는 소비자 기능은 무제한인 것으로 보이며 언제든지 수량에 관계없이 사용할 수 있습니다.

측정되는 서비스

클라우드 시스템은 스토리지, 처리, 대역폭, 활성 사용자 계정과 같은 서비스 유형에 적합한 추상화 수준에서 한도 측정 기능을 활용하여 리소스 사용량을 자동으로 제어하고 최적화합니다. 리소스 사용량을 모니터링, 제어, 보고하여 투명성을 확보할 수 있습니다.

73%

리소스 풀링을
사용했다는 응답자
비율로,
2019년보다 15% 증가한 수치

77%

빠른 탄력성을 사용했다는
응답자 비율로,
2019년보다 18% 증가한 수치

78%

측정되는 서비스를 사용했다는
응답자 비율로,
2019년보다 16% 증가한 수치

SRE 및 DevOps

공개 컨퍼런스와 대화에서 DevOps 커뮤니티가 대두되면서 Google 내부에서도 사이트 안정성 엔지니어링(SRE)이라는 유사한 개념의 접근 방식이 등장했습니다. SRE와 Facebook 프로덕션 엔지니어링 분야 같은 유사한 접근 방식은 DevOps에 동기를 부여하는 여러 동일한 목표와 기술을 수용합니다. 2016년 SRE는 사이트 안정성 엔지니어링에 대한 첫 번째 책⁴을 펴내면서 공식적으로 공개 담론에 합류했습니다. 그 이후로 이 접근 방식이 많이 채택되었으며 오늘날 전 세계의 SRE 실무자 커뮤니티는 기술 운영을 위한 관행에 대해 협력하고 있습니다.

이쯤에서 혼란이 올 수 있습니다. SRE와 DevOps의 차이점은 무엇일까요? 둘 중 하나를 선택해야 할까요? 어느 게 더 나을까요? 사실 이 둘은 서로 상충되지 않습니다. SRE와 DevOps는 고도로 상호 보완적이며, 연구 결과에 따르면 방향성이 일치합니다. SRE는 엘리트 DevOps팀에서 전형적으로 나타나는 성과 지향 생성적 문화의 핵심에 있는 가치와 동일한 부서 간 커뮤니케이션 및 심리적 안전을 우선시하는 학습 분야입니다. SRE는 핵심 원칙에서 나아가 서비스 수준 지표/서비스 수준 목표(SLI/SLO) 측정항목 프레임워크를 포함하는 실용적인 기술을 제공합니다. 린 제품 프레임워크가 이 연구에서 지원하는 신속한 고객 피드백 주기의 달성 방식을 규정하듯, SRE 프레임워크는 사용자와의 약속을 일관되게 지킬 수 있는 팀의 역량을 향상시킬 수 있는 관행 및 도구에 대한 정의를 제공합니다.

2021년 연구에서는 서비스 가용성 분석에서 나아가 보다 일반적인 안정성 카테고리를 아우르도록 운영에 대한 조사를 확대했습니다. 올해 설문조사에서는 SRE 관행에서 영감을 얻은 몇 가지 항목을 도입하여 팀이 다음을 수행하는 정도를 평가했습니다.

- 사용자 대면 행동 측면에서 안정성을 정의
- 오류 예산에 따라 작업의 우선순위를 지정하기 위해 SLI/SLO 측정항목 프레임워크를 채택
- 수동 작업 및 중단을 유발하는 알림을 줄이기 위해 자동화를 사용
- 이슈 대응을 위한 프로토콜 및 준비 훈련을 정의
- 소프트웨어 배포 수명 주기 전반에서 안정성 원칙을 포함('안정성에 대한 원점 회귀')

결과를 분석해보니 현대적 운영 관행에 탁월한 팀이 더 나은 SDO 성과를 보고할 가능성이 1.4배, 더 나은 비즈니스 성과를 보고할 가능성이 1.8배 더 높다는 증거가 확인되었습니다.

연구에서는 대다수 팀이 SRE 관행을 채택했습니다. 응답자의 52%가 이러한 관행을 어느 정도 사용한다고 답했지만 채택 정도는 팀마다 크게 다릅니다. 데이터는 관행의 활용 방법을 토대로 안정성과 전체 SDO 성과 향상을 예측할 수 있음을 보여줍니다. SRE는 DevOps 성공을 이끕니다.

⁴ Betsy Beyer et al., eds., Site Reliability Engineering(O'Reilly Media, 2016).

또한 개발자와 운영자가 공동으로 안정성에 기여할 수 있는 정도에 반영되는 공동 책임 운영 모델이 안정성 성과 향상으로 이어진다는 사실을 발견했습니다.

SRE는 객관적인 성과 평가를 개선하는 것 외에도 기술 실무자의 작업 경험을 향상시킵니다. 일반적으로 운영 작업 부하가 많은 개인은 번아웃되기 쉽지만 SRE는 긍정적인 효과가 있습니다. 팀에서 SRE 관행을 더 많이 채택할수록 팀원이 번아웃을 경험할 가능성이 적은 것으로 확인되었습니다. SRE는 리소스 최적화에도 도움이 될 수 있습니다. SRE 관행을 적용하여 안정성 목표를 달성하는 팀은 SRE를 실행하지 않는 팀보다 코드 작성에 더 많은 시간을 할애한다고 보고합니다.

연구에 따르면 저조한 성과자에서 엘리트 성과자에 이르기까지 어떤 수준의 SDO 성과를 기록하는 팀이라도 SRE 관행의 사용을 늘리면 이점을 누릴 가능성이 높습니다. 팀의 성과가 높을수록 현대적 운영 방식을 채택할 가능성이 커집니다. 엘리트 성과자는 저조한 성과자에 비해 SRE 관행을 사용한다고 보고할 가능성이 2.1배 더 높습니다. 그러나 최상의 수준으로 운영되는 팀이라도 성장할 여지가 있습니다. 엘리트 응답자의 10%만 연구에서 조사된 모든 SRE 관행을 팀에서 완벽하게 이행했다고 밝혔습니다. 여러 업계에서 SDO 성과 향상을 이어감에 따라 운영에 대한 각 팀의 접근 방식이 지속적인 DevOps 개선을 가능케 하는 주요 동인 역할을 하고 있습니다.

52%

SRE 관행을 사용한다고
보고한 응답자 비율

엘리트 성과자는 저조한 성과자에 비해 SRE 관행을 사용한다고 보고할 가능성이 2.1배 더 높습니다. 그러나 최상의 수준으로 운영되는 팀이라도 성장할 여지가 있습니다. 엘리트 응답자의 10%만 연구에서 조사된 모든 SRE 관행을 팀에서 완벽하게 이행했다고 밝혔습니다.

문서

올해 연구에서는 팀이 작업하는 서비스와 애플리케이션에 대한 매뉴얼, 리드미, 코드 주석과 같은 내부 문서의 품질을 살펴봤습니다. 문서 품질은 다음을 기준으로 평가되었습니다.

- 독자가 목표를 달성하는 데 도움이 되는 정도
- 정확하고 최신 상태이고 포괄적인지 여부
- 찾기 쉽고, 잘 정리되어 있고, 이해하기 쉬운 정도⁵

내부 시스템에 대한 정보를 기록하고 액세스하는 것은 팀의 기술 작업에서 중요한 부분입니다. 약 25%의 응답자가 양질의 문서를 보유하고 있는 것으로 나타났으며 이 문서 작업이 미치는 영향은 분명합니다. 우수한 품질의 문서를 보유한 팀은 소프트웨어 배포 및 운영(SDO)에서 더 나은 성과를 올릴 가능성이 2.4배 더 높습니다. 우수한 품질의 문서를 보유한 팀은 품질이 미흡한 문서를 보유한 팀보다 더 빠르고 안정적으로 소프트웨어를 배포합니다. 문서가 완벽할 필요는 없습니다. 연구에 따르면 문서 품질의 향상은 성과에 긍정적이고 직접적인 영향을 미칩니다.

오늘날의 기술 환경에서는 시스템이 점점 더 복잡해지는 한편 이러한 시스템의 다양한 측면을 담당하는 전문가와 특화된 역할이 마련되어 있습니다. 문서는 보안에서 테스트에 이르기까지 특화된 하위 팀과 폭넓은 팀 간에 전문 지식과 지침을 공유하는 핵심적인 방법입니다.

연구에서는 팀이 기술 관행을 성공적으로 이행하는지를 문서 품질을 통해 예측할 수 있음을 발견했습니다. 이러한 관행은 관측 가능성, 지속적 테스트, 배포 자동화와 같은 시스템의 기술 역량 향상으로 이어집니다. 연구에서는 양질의 문서를 보유한 팀에 대해 다음과 같은 사실을 발견했습니다.

- 보안 관행을 이행할 가능성이 3.8배 더 높음
- 안정성 목표를 충족하거나 초과 달성할 가능성이 2.4배 더 높음
- 사이트 안정성 엔지니어링(SRE) 관행을 이행할 가능성이 3.5배 더 높음
- 클라우드를 완전히 활용할 가능성이 2.5배 더 높음

⁵ 다음과 같은 기술 문서에 대한 기존 연구에서 제공하는 품질 측정항목:

— Aghajani, E. et al. (2019). Software Documentation Issues Unveiled. *Proceedings of the 2019 IEEE/ACM 41st International Conference on Software Engineering*, 1199-1210. <https://doi.org/10.1109/ICSE.2019.00122>

— Plösch, R., Dautovic, A., & Saft, M. (2014). The Value of Software Documentation Quality. *Proceedings of the International Conference on Quality Software*, 333-342. <https://doi.org/10.1109/QSIC.2014.22>

— Zhi, J. et al. (2015). Cost benefits and quality of software development documentation: A systematic mapping. *Journal of Systems and Software*, 99(C), 175-198. <https://doi.org/10.1016/j.jss.2014.09.042>

문서 품질을 개선하는 방법

기술 작업에는 정보의 발견과 사용이 수반되지만 양질의 문서는 콘텐츠를 작성하고 유지관리하는 인력에 달려 있습니다. 2019년의 연구에 따르면 내부 및 외부 정보 소스에 대한 액세스가 생산성을 좌우합니다. 올해의 연구는 이를 한 단계 더 발전시켜 사용되는 문서의 품질과 이 문서의 품질에 영향을 미치는 관행을 검토합니다.

연구 결과, 다음과 같은 관행이 문서 품질에 매우 긍정적인 영향을 미치는 것으로 나타났습니다.

제품 및 서비스의 중요한 사용 사례 문서화. 시스템에 대해 어떤 내용을 문서화하는지가 중요하며 독자는 사용 사례를 통해 정보를 얻고 관련 시스템을 작동할 수 있습니다.

기존 문서 업데이트 및 편집에 대한 명확한 가이드라인 작성. 문서 작업의 대부분은 기존 콘텐츠를 유지 관리하는 것입니다. 팀원이 업데이트를 수행하거나 부정확한 정보 또는 오래된 정보를 제거하는 방법을 알고 있다면 시간 경과에 따라 시스템이 변경되더라도 팀은 문서 품질을 유지할 수 있습니다.

소유자 정의. 양질의 문서를 보유한 팀에서는 문서에 대한 소유권을 명확하게 정의할 가능성이 더 큼니다. 소유권은 새 콘텐츠를 작성하고 기존 콘텐츠의 변경사항을 업데이트하거나 확인하는 명시적인 책임을 허용합니다. 양질의 문서를 보유한 팀은 작업 중인 애플리케이션의 모든 주요 기능에 대한 문서를 작성했다고 진술할 가능성이 더 높으며, 명확한 소유권은 이처럼 광범위한 적용 범위를 설정하는 데 도움이 됩니다.

양질의 문서를
보유한 팀

3.8배

보안 관행을 이행할
가능성이 더 높음

2.4배

안정성 목표를 충족하거나 초과
달성할 가능성이 더 높음

3.5배

사이트 안정성 엔지니어링(SRE)
관행을 이행할 가능성이 더 높음

2.5배

클라우드를 완전히 활용할
가능성이 더 높음

문서를 소프트웨어 개발 프로세스의 일부에 포함. 문서를 작성하고 시스템 변경에 따라 업데이트한 팀은 더 높은 품질의 문서를 보유하고 있습니다. 테스트와 마찬가지로 문서 작성 및 유지관리는 고성능 소프트웨어 개발 프로세스에서 필수적인 부분입니다.

성과 검토 및 프로모션 중 문서 작업 인정.

인정은 전반적인 문서 품질과 상관관계가 있습니다. 문서를 작성하고 유지관리하는 것은 소프트웨어 엔지니어링 작업의 핵심 부분이며 이를 통해 품질이 향상됩니다.

양질의 문서를 지원하는 것으로 확인된 기타 리소스는 다음과 같습니다.

- 문서 작성 및 유지관리 방법에 대한 교육
- 코드 샘플 또는 불완전한 문서에 대한 자동 테스트
- 문서 스타일 가이드 및 전 세계 잠재고객을 위한 작성 가이드와 같은 가이드라인

문서는 DevOps 기능을 성공적으로 구현하기 위한 기초입니다. 품질이 더 높은 문서는 보안, 안정성, 클라우드의 완전한 활용 등 개별 DevOps 기능에 대한 투자 성과를 배가시킵니다. 양질의 문서를 지원하는 관행을 이행하면 더 강력한 기술 역량과 더 높은 SDO 성과를 통해 이점을 누릴 수 있습니다.

보안

[원점 회귀] 및 전체 통합

기술팀이 빠른 속도로 발전을 이어감에 따라 보다 정교한 보안 위협을 더 많이 직면하게 됩니다. Tenable의 2020 Threat Landscape Retrospective Report에 따르면 2020년에 220억 개 이상의 기밀 개인 정보 또는 비즈니스 데이터 기록이 노출되었습니다.⁶ 보안은 사후 또는 배포 전 마지막 단계에 고려해서는 안 되며 소프트웨어 개발 프로세스 전반에 통합되어야 합니다.

소프트웨어를 안전하게 배포하려면 보안 관행을 악의적인 행위자가 사용하는 기술보다 더 빠르게 발전시켜야 합니다. 2020년 SolarWinds 및 Codecov 소프트웨어 공급망이 공격당했을 당시 해커들은 SolarWinds의 빌드 시스템과 Codecov의 bash 업로더 스크립트⁷를 손상시켜 두 회사의 고객 수천 명의 인프라에 은밀히 침투했습니다. 업계는 이러한 공격이 광범위한 영향을 미친다는 점을 감안하여 예방 접근 방식에서 진단 접근 방식으로 전환해야 합니다. 이때 소프트웨어팀은 시스템이 이미 손상되었다고 가정하고 공급망에 보안 기능을 구축해야 합니다.

이전 보고서와 마찬가지로 이번 연구에서도 엘리트 성과자가 보안 관행을 이행하는 데 탁월한 것으로 드러났습니다. 올해에는 안정성 목표를 달성하거나 초과 달성한 엘리트 성과자가 소프트웨어 개발 프로세스에 보안 관행을 통합한 가능성이 2배 높은 것으로 나타났습니다. 이는 안정성 표준을 유지하면서 배포를 가속화한 팀이 소프트웨어를 빠르고 안정적으로 배포하는 역량을 유지하면서도 보안 확인과 관행을 통합하는 방법을 찾았음을 시사합니다.

개발 프로세스 전반에 걸쳐 보안 관행을 통합하는 팀은 우수한 배포 및 운영 성과를 낼 뿐만 아니라 조직 목표를 달성하거나 초과 달성할 가능성이 1.6배 더 높습니다. 보안을 수용하는 개발팀은 비즈니스의 가치를 크게 끌어올립니다.

안정성 목표를 달성하거나 초과 달성한 엘리트 성과자는 소프트웨어 개발 프로세스에 보안 관행을 통합할 가능성이 2배 높았습니다.

⁶ <https://www.tenable.com/cyber-exposure/2020-threat-landscape-retrospective>

⁷ <https://www.cybersecuritydive.com/news/codecov-breach-solarwinds-software-supply-chain/598950/>

바로잡는 방법

보안의 중요성을 강조하고 팀이 보안에 우선순위를 두도록 제안하는 것은 쉽지만 이를 실천하려면 기존 정보 보안 방식에 몇 가지 변화가 필요합니다. 다음 관행을 활용하여 보안을 통합하고, 소프트웨어 배포 및 운영 성과를 개선하고, 조직 성과를 개선할 수 있습니다.

보안 테스트. 사전 승인된 코드를 사용해야 하는 영역을 포함하여 자동화된 테스트 프로세스의 일부로 보안 요구사항을 테스트합니다.

모든 단계에 보안 검토를 통합. 전체 소프트웨어 배포 수명 주기의 일상 업무에 정보 보안(InfoSec)을 통합합니다. 여기에는 애플리케이션 설계 및 아키텍처 단계에서 InfoSec팀이 의견을 제공하고 소프트웨어 데모에 참석하고 데모 중 의견을 제공하는 것이 포함됩니다.

보안 검토. 모든 주요 기능에 대한 보안 검토를 수행합니다.

사전 승인된 코드 빌드. InfoSec팀에서 개발자 및 IT 운영자가 업무에 사용할 수 있도록 사전 승인되고 사용하기 쉬운 라이브러리, 패키지, 도구 모음, 프로세스를 구축하도록 합니다.

조기에 빈번하게 InfoSec팀 초대. 계획 및 애플리케이션 개발의 모든 후속 단계에 InfoSec팀을 참여시켜 보안 관련 약점을 조기에 발견할 수 있도록 함으로써 팀에서 문제를 해결할 수 있는 충분한 시간을 줍니다.

앞서 언급했듯이 고품질 문서는 다양한 역량을 향상시키며 보안도 예외가 아닙니다. 고품질 문서를 보유한 팀이 개발 프로세스 전반에 보안 관행을 통합할 가능성이 3.8배 더 높은 것으로 드러났습니다. 조직의 모든 구성원이 암호화에 대한 전문 지식을 가지고 있는 것은 아닙니다. 이러한 전문 지식은 조직에서 문서화된 보안 관행을 통해 가장 효과적으로 공유됩니다.

보안 관행

보안 테스트	58%
모든 단계에 보안 검토 통합	54%
보안 검토	60%
사전 승인된 코드 빌드	49%
조기에 빈번하게 InfoSec팀 초대	63%

기술 DevOps 기능

연구에 따르면 지속적 배포를 채택하여 DevOps 변환을 경험하는 조직은 고품질, 저위험, 비용 효율적인 프로세스를 갖출 가능성이 더 높습니다.

이 연구에서는 특히 다음과 같은 기술적 관행을 평가했습니다.

- 느슨하게 결합된 아키텍처
- 트렁크 기반 개발
- 지속적 테스트
- 지속적 통합
- 오픈소스 기술 사용
- 모니터링 및 관측 가능성 관행
- 데이터베이스 변경 관리
- 배포 자동화

이 모든 관행이 지속적 배포를 개선하지만 느슨하게 결합된 아키텍처와 지속적 테스트가 가장 큰 영향을 미치는 것으로 확인되었습니다. 예를 들어 올해에는 안정성 목표를 달성하는 엘리트 성과자가 저조한 성과자에 비해 느슨하게 결합된 아키텍처를 사용할 가능성이 3배 더 높은 것으로 나타났습니다.

느슨하게 결합된 아키텍처

연구는 서비스와 팀 간의 세분화된 종속성을 줄이기 위해 노력함으로써 IT 성과를 높일 수 있음을 계속 보여줍니다. 사실 이는 성공적인 지속적 배포를 가장 강력하게 예측할 수 있는 요인 중 하나입니다. 느슨하게 결합된 아키텍처를 사용하는 팀은 서로 독립적으로 확장, 실패, 테스트, 배포를 수행할 수 있습니다. 팀은 각자의 속도로 작업을 진행하고, 더 작은 배치로 작업하고, 기술 부채를 덜 발생시키고, 실패로부터 더 빠르게 회복할 수 있습니다.

지속적 테스트 및 지속적 통합

이전 연구의 발견사항과 유사하게 지속적 테스트는 성공적인 지속적 배포를 가장 강력하게 예측할 수 있는 요인입니다. 안정성 목표를 달성하는 엘리트 성과자는 지속적 테스트를 활용할 가능성이 3.7배 더 높습니다. 팀은 배포 프로세스 전체에 빈번한 조기 테스트를 포함하여 처음부터 끝까지 개발자와 협력하는 테스터를 통해 제품, 서비스 또는 애플리케이션을 더 빠르게 반복하고 변경할 수 있습니다. 이 피드백 루프를 사용하여 고객에게 가치를 제공하는 한편 자동화된 테스트와 지속적 통합과 같은 관행을 손쉽게 포함할 수 있습니다.

또한 지속적 통합은 지속적 배포를 개선합니다. 안정성 목표를 달성하는 엘리트 성과자는 지속적 통합을 활용할 가능성이 5.8배 더 높습니다. 지속적 통합에서 각 커밋은 소프트웨어 빌드를 트리거하고 몇 분 안에 피드백을 제공하는 일련의 자동화된 테스트를 실행합니다.

지속적 통합을 통해 많은 경우 복잡한 수동 조정을 줄여서 성공적인 통합을 지원할 수 있습니다.

원래 켄트 벡과 Extreme Programming 커뮤니티가 정의한 지속적 통합에는 다음에 논의할 트렁크 기반 개발 관행도 포함됩니다.⁷

트렁크 기반 개발

이 연구에서는 성과가 우수한 조직에서 개발자가 소규모 배치로 작업하고 작업을 공유 트렁크에 빈번하게 병합하는 트렁크 기반 개발을 실행할 가능성이 더 높은 것으로 지속적으로 나타났습니다. 실제로, 안정성 목표를 달성하는 엘리트 성과자는 트렁크 기반 개발을 사용할 가능성이 2.3배 더 높습니다. 저조한 성과자는 수명이 긴 분기를 사용하고 병합을 지연할 가능성이 더 큼니다.

팀은 적어도 하루에 한 번(가능하면 하루에 여러 번) 작업을 병합해야 합니다. 트렁크 기반 개발은 지속적 통합과 밀접한 관련이 있습니다. 이 두 가지 기술 관행을 함께 사용할 때 영향력이 더 크므로 동시에 이행해야 합니다.

배포 자동화

이상적인 작업 환경에서는 컴퓨터가 반복적인 작업을 수행하고 작업자는 문제 해결에 집중합니다. 배포 자동화를 이행하면 팀이 이 목표에 다가가는 데 도움이 됩니다.

소프트웨어를 자동화된 방식으로 테스트에서 프로덕션으로 이동하면 더 빠르고 효율적인 배포를 지원하여 리드 타임을 줄일 수 있습니다. 또한 수동 배포에서 더 많이 발생하는 배포 오류의 가능성도 줄일 수 있습니다. 팀에서 배포 자동화를 사용하면 즉각적인 피드백을 받아 서비스나 제품을 훨씬 빠른 속도로 개선하는 데 도움이 됩니다. 지속적 테스트, 지속적 통합, 자동화된 배포를 동시에 이행할 필요는 없지만 이 세 가지 방법을 함께 사용하면 더 큰 개선 효과를 볼 수 있습니다.

데이터베이스 변경 관리

버전 제어를 통해 변경 사항을 추적하는 것은 코드 작성 및 유지관리와 데이터베이스 관리에서 중요한 부분입니다. 연구 결과, 안정성 목표를 달성하는 엘리트 성과자는 데이터베이스 변경 관리를 실행할 가능성이 저조한 성과자보다 3.4배 더 높습니다. 또한 성공적인 데이터베이스 변경 관리의 핵심은 모든 관련 팀의 공동작업, 커뮤니케이션, 투명성입니다. 특정 접근 방식 중에서 어떤 방식을 실행할지 선택할 수 있지만 데이터베이스를 변경해야 할 때마다 데이터베이스를 업데이트하기 전에 팀이 함께 모여 변경사항을 검토하는 것이 좋습니다.

⁷ Beck, K. (2000). Extreme programming explained: Embrace change. Addison-Wesley Professional.

모니터링 및 관측 가능성

이전의 연구와 마찬가지로 이번에도 모니터링 및 관측 가능성 관행이 지속적 배포를 뒷받침하는 것으로 나타났습니다. 안정성 목표를 성공적으로 달성하는 엘리트 성과자는 전체 시스템 상태에 관측 가능성을 포함하는 솔루션을 보유할 가능성이 4.1배 더 높습니다. 관측 가능성 관행을 통해 팀은 시스템을 보다 효과적으로 이해할 수 있으므로 문제를 식별하고 해결하는 데 걸리는 시간이 단축됩니다. 또한 연구에 따르면 관측 가능성이 높은 팀은 코딩에 더 많은 시간을 할애합니다. 그 이유 중 하나는 관측 가능성 관행을 이행하면 개발자가 문제의 원인을 찾는 데 시간을 들이지 않고 문제 해결과 코딩에 집중할 수 있기 때문입니다.

반면, 대부분의 오픈소스 기술에는 개발자가 지원에 사용할 수 있는 커뮤니티가 있습니다. 오픈소스 기술은 보다 광범위하게 액세스할 수 있고 상대적으로 저렴하며 맞춤설정이 가능합니다. 안정성 목표를 달성하는 엘리트 성과자는 오픈소스 기술을 활용할 가능성이 2.4배 더 높습니다.

DevOps 혁신을 이행할 때 오픈소스 소프트웨어를 더 많이 사용하도록 전환하는 것이 좋습니다.

오픈소스 기술

많은 개발자가 이미 오픈소스 기술을 활용하고 있으며 이러한 도구를 친숙하게 활용할 수 있는 경험은 조직에 큰 강점으로 작용합니다. 클로즈드 소스 기술의 가장 큰 약점은 조직 안팎으로 지식을 이전하는 역량이 제한된다는 것입니다. 예를 들어 조직의 도구에 이미 익숙한 사람을 고용할 수 없으며 개발자는 그간 축적한 지식을 다른 조직에 전수할 수 없습니다.

기술 DevOps 역량에 대한 자세한 내용은 DORA 역량(<https://cloud.google.com/devops/capabilities>)을 참조하세요.

코로나19

올해 연구에서는 코로나19 팬데믹 동안 팀의 성과에 영향을 미친 요인을 조사했습니다. 특히 코로나19 팬데믹이 소프트웨어 배포 및 운영(SDO) 성과에 부정적인 영향을 미쳤는지, 그 결과 팀이 번아웃을 더 많이 경험했는지, 마지막으로, 번아웃을 완화할 가능성이 있는 요인은 무엇인지 조사했습니다.

먼저 이번 연구에서는 팬데믹이 배포 및 운영 성과에 미친 영향을 파악하고자 했습니다. 많은 조직에서 극적인 시장 변화(예: 오프라인 구매에서 온라인 구매로 전환)를 수용하기 위해 현대화를 우선시했습니다. '비교 방법' 장에서는 소프트웨어 산업의 성과가 어떻게 급격히 가속화되었으며 그러한 추세를 이어가고 있는지 논의합니다. 이제 샘플에서 성과가 우수한 팀이 대다수를 차지하고 있으며 엘리트 성과자는 계속해서 기준을 높여 더 짧은 리드 타임, 더 빠른 복구 시간, 더 낮은 변경 실패율로 보다 빈번하게 배포하고 있습니다. 마찬가지로 GitHub의 연구에 따르면 2020년에는 푸시, pull 요청, 검토된 pull 요청, 사용자가 의견을 제기한 문제⁹ 등 개발자 활동이 증가했습니다. 분명 업계는 팬데믹 덕분에 이 시기보다 팬데믹에도 불구하고 발전을 이어갔습니다. 특히 이 어려운 시기에 SDO 성과가 하락하는 추세가 나타나지 않았다는 점은 주목할 만합니다.

팬데믹으로 인해 사람들의 근무 방식이 변화했으며 많은 사람들에게는 근무하는 장소도 바뀌었습니다. 이에 따라 팬데믹으로 인한 원격 근무의 영향을 살펴봤습니다. 응답자의 89%가 팬데믹으로 인해 재택근무를 하는 것으로 나타났습니다. 20%만이 팬데믹 이전에도 재택근무를 한 적이 있다고 답했습니다. 원격 작업 환경으로 전환은 소프트웨어 개발, 비즈니스 운영, 공동작업 방식에 큰 영향을 미쳤습니다. 많은 경우 재택근무로 인해 복도에서 마주친 김에 대화를 나누거나 얼굴을 맞대고 공동작업할 수 있는 기회가 사라졌습니다.

89%

팬데믹으로 인해
재택근무하는 응답자 비율

⁹ <https://octoverse.github.com/>

번아웃을 줄인 요인

그럼에도 불구하고 원격 근무에 따른 번아웃으로 팀이 어려움을 겪는지 여부에 큰 영향을 미친 한 가지 요인을 발견했습니다. 바로 문화입니다. 소속감을 느끼는 사람들로 구성된 생성적 팀 문화를 갖춘 팀은 팬데믹 기간 동안 번아웃을 경험할 가능성이 절반에 불과했습니다. 이는 팀과 문화를 우선시하는 것이 얼마나 중요한지를 잘 보여줍니다. 성과가 뛰어난 팀은 팀원들과 팀 모두를 압박하는 어려운 시기를 견뎌낼 준비가 잘 되어 있습니다.

소속감을 느끼는 사람들로 구성된 생성적 팀 문화를 갖춘 팀은 팬데믹 기간 동안 번아웃을 경험할 가능성이 절반에 불과했습니다.

문화

일반적으로 문화는 모든 조직에서 피할 수 없는 인간관계의 토대입니다. 문화는 직원들이 조직과 서로에 대해 생각하고 느끼고 행동하는 방식에 영향을 미치는 모든 요소입니다. 모든 조직에는 고유한 문화가 있으며 이번 조사에서는 문화가 조직 및 IT 성과의 주요 동인 중 하나인 것으로 확인되었습니다. 특히 분석 결과 **생성적 문화**가 소프트웨어 배포 및 운영(SDO) 성과 향상으로 이어지는 것으로 확인됩니다. 생성적 문화는 웨스트럼 조직 문화 유형과 조직 내에서 구성원들이 느끼는 소속감 및 포용성으로 측정됩니다. 예를 들어, 안정성 목표를 달성하는 엘리트 성과자는 생성적 팀 문화를 갖추고 있을 가능성이 저조한 성과자에 비해 2.9배 더 높습니다. 마찬가지로 생성적 문화는 더 우수한 조직 성과와 더 낮은 비율의 직원 번아웃으로 이어집니다. 다시 말해 문화가 정말 중요합니다. 다행히 문화는 항상 유동적이고 다면적이어서 변화할 수 있습니다.

DevOps를 성공적으로 실행하려면 조직에 공동작업하고 협업하는 팀이 있어야 합니다. 2018년에는 성과가 우수한 팀이 단일 교차 기능팀으로 소프트웨어를 개발하고 배포할 가능성이 2배인 것으로 나타났습니다. 이는 어떤 조직이든 성공을 거두는 데 공동작업과 협력이 결정적임을 보여줍니다. 한 가지 중요한 질문이 있습니다. 팀 간 공동작업을 장려하는 환경을 조성하는 데 기여하는 요소는 무엇일까요?

그동안 연구에서는 문화의 구성요소를 유형화하고 DevOps 커뮤니티에 문화가 조직 및 IT 성과에 미치는 영향에 대한 더 나은 이해를 돕고자 노력했습니다. 웨스트럼의 조직 문화 유형을 사용하여 실질적으로 문화를 정의하는 것으로 이 여정을 시작했습니다. 웨스트럼은 조직을 권력 지향, 규칙 지향, 성과 지향이라는 세 가지 유형으로 나눴습니다. 자체 연구에 이 프레임워크를 사용했으며 정보 흐름, 신뢰, 혁신, 위험 공유에 최적화된 성과 지향적인 조직 문화에서 우수한 SDO 성과를 낸다는 점을 발견했습니다.

문화와 DevOps에 대한 이해가 깊어짐에 따라 **심리적 안전**과 같은 다른 심리 사회적 요인을 포함하도록 문화에 대한 초기 정의를 확장하고자 노력했습니다. 성과가 뛰어난 조직은 직원들이 부정적인 결과에 대한 두려움 없이 면밀히 계산된 적당한 위험을 감수하도록 장려하는 문화를 가지고 있을 가능성이 더 큼니다.

문화는 항상 유동적이고
다면적이어서 변화할 수
있습니다.

소속감 및 포용성

문화가 성과에 지속적으로 큰 영향을 미친다는 점을 감안하여 올해 연구에서는 직원의 소속감과 포용성이 성과에 대한 문화의 유익한 영향에 기여하는지 알아볼 수 있도록 모델을 확장했습니다.

심리학 연구에 따르면 사람들에게는 타인과 끈끈하고 안정적인 관계를 맺고 유지하려는 동기가 내재되어 있습니다.¹⁰ 인간은 다른 사람들과 연결되어 있다고 느끼고 자신이 속한 다양한 그룹에서 인정받으려는 동기에 따라 움직입니다. 소속감은 신체 및 심리적으로 여러 긍정적인 결과를 낳습니다. 예를 들어, 연구에 따르면 소속감은 동기 부여에 긍정적인 영향을 미치고 학업 성취도 향상으로 이어집니다.¹¹

사람들이 직장에서 자신의 모습을 온전히 드러내는 데 편안함을 느껴야 하고 각자의 경험과 배경이 존중받고 인정받는다는 생각이 이러한 유대감을 형성합니다.¹² 조직에 대한 소속감을 주는 포용적 문화를 조성하기 위해 노력하면 번성하고 다양하며 의욕적인 인력을 구성하는 데 도움이 됩니다.

연구 결과에 따르면 소속감과 포용성을 중시하는 성과 지향적인 조직이 조직 문화가 덜 긍정적인 조직에 비해 직원의 번아웃 수준이 더 낮을 가능성이 있습니다.

심리 사회적 요인이 직원의 SDO 성과 및 번아웃 수준에 어떤 영향을 미치는지를 보여주는 증거를 고려하면 성공적인 DevOps 혁신을 추진하려는 경우 혁신하고자 하는 노력의 일환으로 문화 관련 문제를 해결하는 데 투자하는 것이 좋습니다.

10 Baumeister & Leary, 1995. The need to belong: Desire for interpersonal attachments as a fundamental human motivation. *Psychological Bulletin*, 117(3), 497–529.

<https://doi.org/10.1037/0033-2909.117.3.497>

11 Walton et al., 2012. Mere belonging: the power of social connections. *Journal of Personality and Social Psychology*, 102(3):513–32.

<https://doi.org/10.1037/a0025731>

12 Mor Barak & Daya, 2014; Managing diversity: Toward a globally inclusive workplace. Sage.

Shore, Cleveland, & Sanchez, 2018; Inclusive workplaces: A review and model, *Human Resources Review*.

<https://doi.org/10.1016/j.hrmmr.2017.07.003>

4장

설문조사 대상

업계 전문가를 대상으로 7년간 32,000개가 넘는 설문조사를 실시하여 작성된 Accelerate State of DevOps 2021은 팀과 조직을 가장 성공적으로 만드는 소프트웨어 개발 및 DevOps 관행을 설명합니다.

올해는 전 세계의 다양한 업계에서 일하는 1,200명의 전문가들이 경험을 공유하여 성과를 향상하는 요인에 대한 이해를 높이는 데 도움을 주었습니다. 요약하면, 인구통계 및 기업통계 전반에서 대표성이 매우 일관적으로 유지되었습니다.



인구통계 및 기업통계

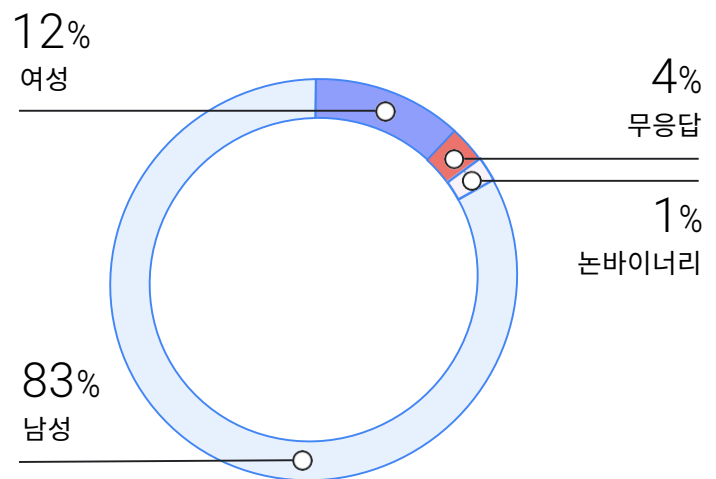
예년과 마찬가지로 각 설문조사 응답자로부터 인구통계 정보를 수집했습니다. 카테고리에는 성별, 장애인, 소수 집단이 포함됩니다.

올해는 회사 규모, 업종, 지역을 포함한 기업통계학적 카테고리 전반에 걸쳐 대표성이 이전 보고서와 일관되게 유지되었습니다. 이번에도 60%가 넘는 응답자가 엔지니어 또는 관리자에 해당하며 3분의 1은 기술 업계에서 근무합니다. 또한 응답자에는 금융 서비스, 소매, 산업/제조회사 종사자도 포함되었습니다.

인구통계

성별

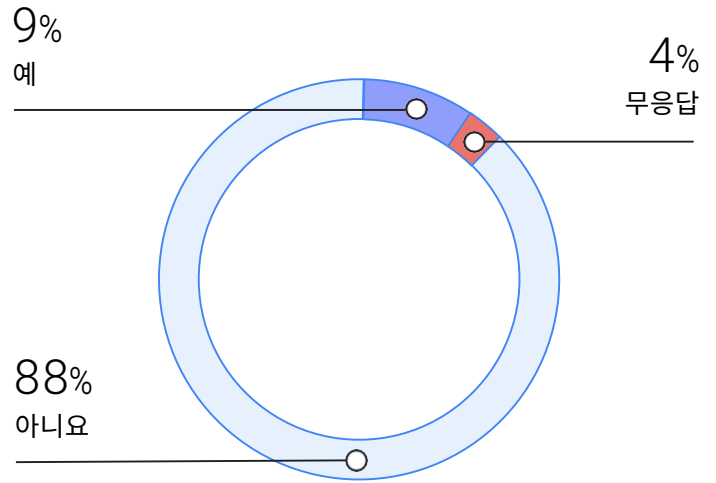
이전 설문조사와 마찬가지로 올해의 표본은 남성 83%, 여성 12%, 논바이너리 1%로 구성되어 있습니다. 응답자들은 팀원의 약 25%를 여성이 차지한다고 밝혔습니다. 이는 2019년(16%)보다 크게 증가한 수치이며 2018년(25%)과는 동일합니다.



올해 응답자는 팀의 25%가 여성(중앙값)이라고 밝혔으며 이는 2019년의 하락에서 회복한 것입니다.

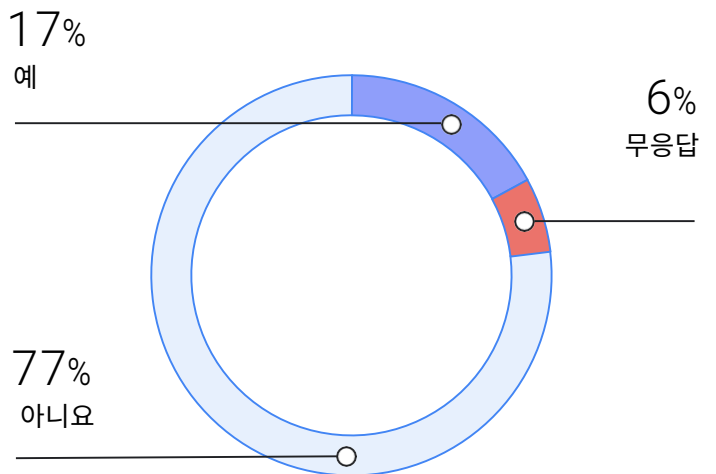
장애

장애는 Washington Group Short Set의 안내를 따르는 6가지 기준으로 식별됩니다.¹³ 장애에 대해 질문을 한 것은 올해가 3년째입니다. 장애인 비율은 2019년 보고서와 마찬가지로 9%였습니다.



소수 집단

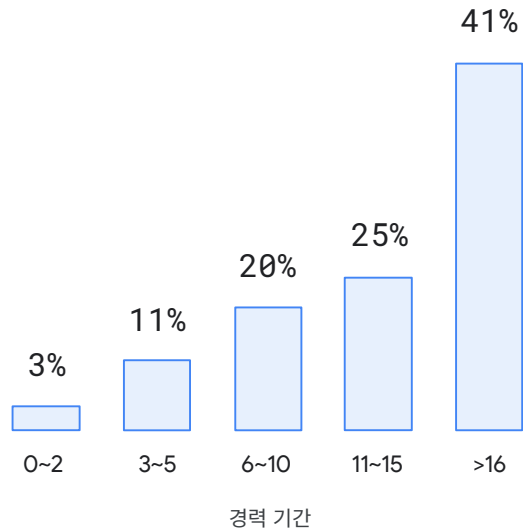
소수 집단의 구성원으로 식별하는 것은 인종, 성별 또는 다른 특성을 나타낼 수 있습니다. 소수 집단에 대한 질문을 한 것은 올해가 4년째입니다. 소수 집단에 속하는 것으로 식별되는 응답자의 비율은 2019년 13.7%에서 2021년 17%로 약간 증가했습니다.



¹³ <https://www.washingtongroup-disability.com/question-sets/wg-short-set-on-functioning-wg-ss/>

경력 기간

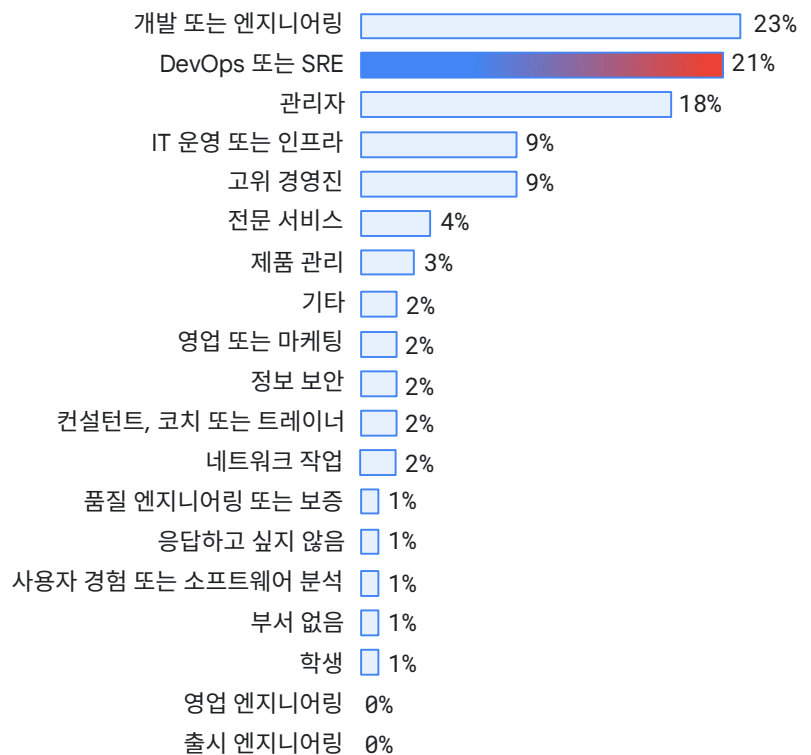
올해 설문조사의 응답자는 경험이 풍부합니다. 41%가 16년 이상의 경력을 가지고 있습니다. 응답자의 85% 이상이 6년 이상의 경력을 가지고 있습니다.



기업통계

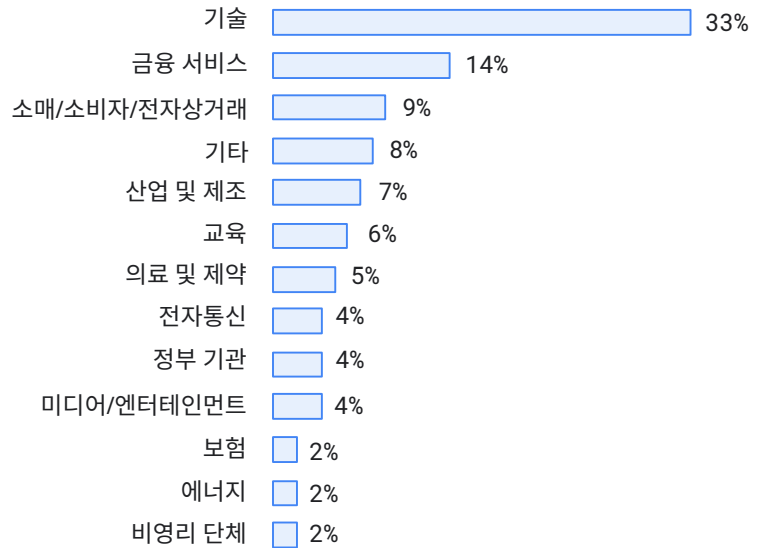
부서

응답자는 주로 개발 또는 엔지니어링팀(23%), DevOps 또는 SRE팀(21%), 관리자(18%), IT 운영 또는 인프라팀(9%)에서 일하는 개인으로 구성됩니다. 컨설턴트의 비중은 2019년 4%에서 2%로 줄어든 반면 고위 경영진의 비중은 2019년 4%에서 9%로 커졌습니다.



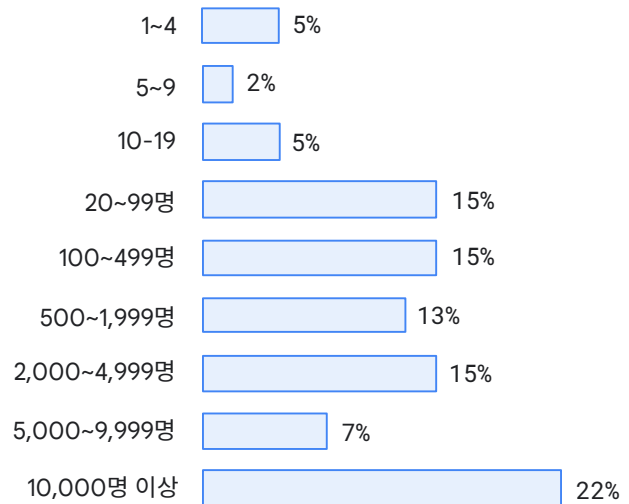
업종

이전의 Accelerate State of DevOps 보고서에서와 같이 대부분의 응답자가 기술 업계 종사자이며 금융 서비스업, 소매업, 기타 업종이 그 뒤를 잇고 있습니다.



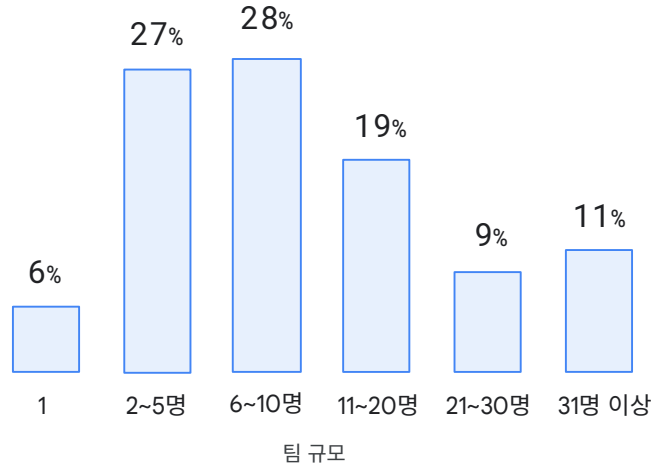
직원

이전의 Accelerate State of DevOps 보고서와 마찬가지로 응답자는 다양한 규모의 조직에 소속되어 있습니다. 응답자의 22%는 직원이 10,000명 이상인 회사 소속이며 7%는 직원이 5,000~9,999명인 회사 소속입니다. 응답자의 15%는 직원이 2,000~4,999명인 조직에 속해 있습니다. 또한 직원이 500~1,999명(13%), 100~499명(15%), 20~99명(15%)인 조직에 속한 응답자의 비율도 비슷했습니다.



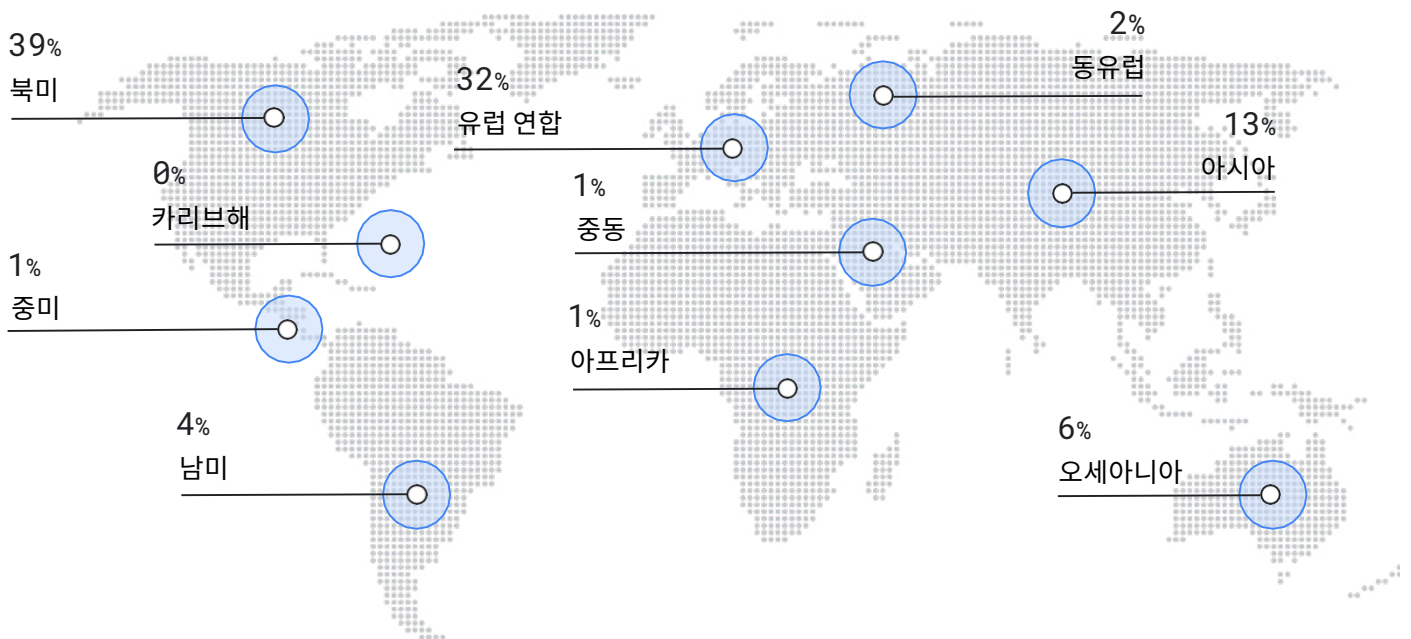
팀 규모

응답자의 절반 이상(62%)이 10명 이하로 구성된 팀에서 작업합니다(6~10명의 경우 28%, 2~5명의 경우 27%, 1인 팀의 경우 6%). 응답자의 19%는 11~20명으로 구성된 팀에 속해 있습니다.



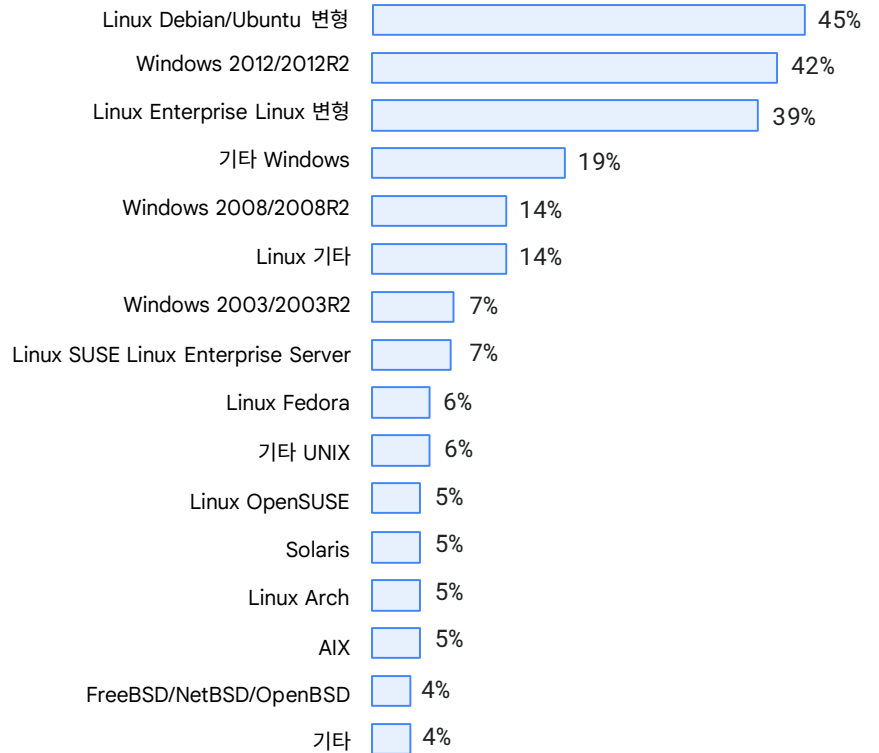
지역

올해 설문조사에서는 북미 지역의 응답이 줄었습니다(2019년 50%에서 2021년 39%). 대신 유럽(2019년 29%에서 2021년 32%), 아시아(2019년 9%에서 2021년 13%), 오세아니아(2019년 4%에서 2021년 6%), 남아메리카(2019년 2%에서 2021년 4%)의 응답이 증가했습니다.



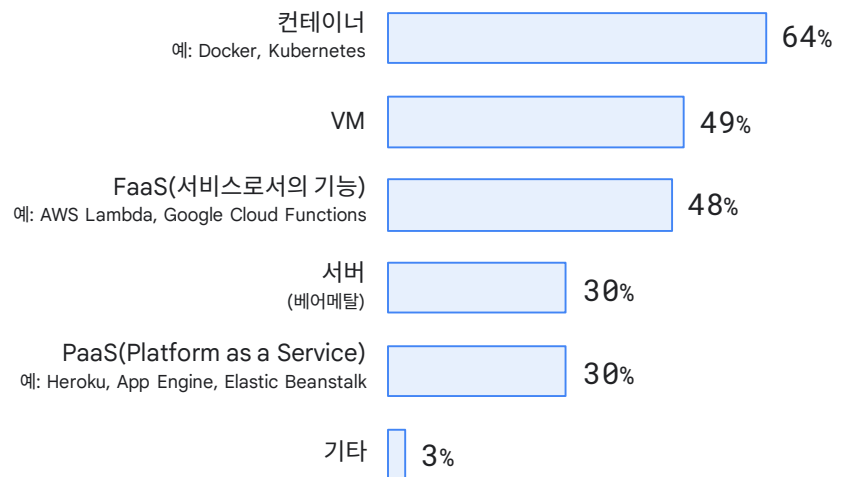
운영체제

운영체제의 분포는 이전 State of DevOps 보고서와 유사합니다. 또한 운영체제 목록에 업데이트를 사용할 수 있음을 강조하도록 도움을 준 응답자들에게 감사를 표합니다.



배포 대상

올해 응답자들이 작업하는 주요 서비스 또는 애플리케이션을 배포하는 위치를 살펴봤습니다. 놀랍게도 응답자의 대다수(64%)가 컨테이너를 사용하고 48%가 가상 머신(VM)을 사용합니다. 이는 업계가 최신 배포 대상 기술을 향해 변화하고 있음을 반영한 것입니다. 회사 규모별로 차이가 있는지 확인했지만 배포 대상 간에 큰 차이가 발견되지 않았습니다.



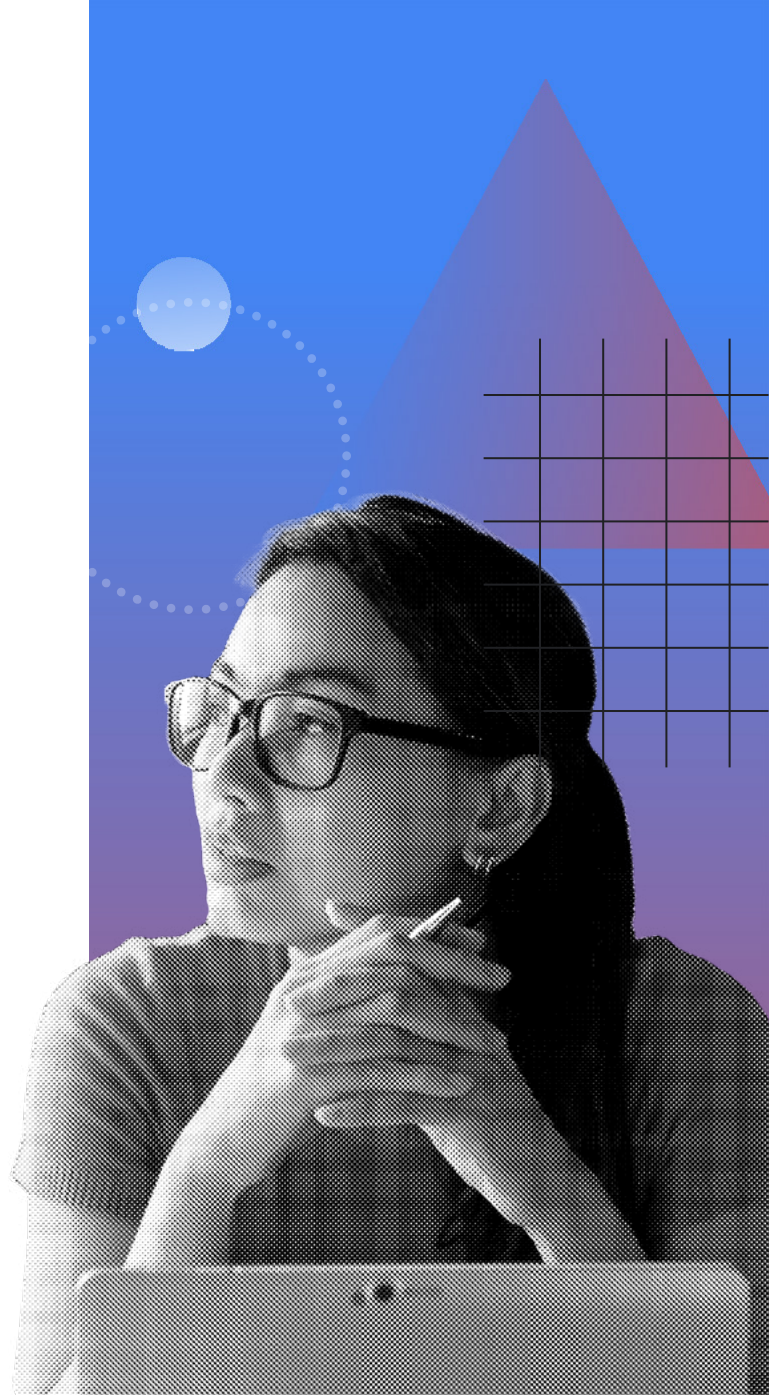
5장

최종 의견

7년간 연구를 이어오면서 DevOps가 조직에 제공하는 이점을 계속 확인하고 있습니다. 해를 거듭할수록 조직은 혁신을 가속화하고 발전하고 있습니다.

원칙과 기능을 받아들이는 팀은 소프트웨어를 빠르고 안정적으로 배포하는 한편 비즈니스에 직접적으로 가치를 선사합니다. 올해는 SRE 관행, 보안 소프트웨어 공급망, 양질의 문서가 미치는 효과를 조사하고 클라우드 활용을 다시 살펴보았습니다. 각 영역은 개인과 팀의 효율성을 높일 수 있습니다. 이 연구는 사람을 솔루션에 맞추는 것이 아니라 이러한 기능을 활용하는 사람에게 적합한 솔루션을 구성하는 것이 중요하다는 사실에 주목합니다.

올해 설문조사에 참여해 주신 모든 분들께 감사드리며, 이 연구가 설문 참여자와 해당 조직이 더 나은 팀과 소프트웨어를 구축하는 데 도움이 되면서도 일과 생활의 균형을 유지하는 데 보탬이 되기를 바랍니다.



6장

감사의 말씀

올해의 보고서는 열정적인 참여자분들이 있었기에 발간될 수 있었습니다. 설문조사 질문 설계, 분석, 작성, 편집, 보고서 디자인은 이 거대한 노력을 실현할 수 있도록 동료들이 도움을 준 많은 방법 중 일부일 뿐입니다. 저자들은 올해 보고서에 대한 정보와 안내를 제공해주신 모든 분들께 감사 말씀을 드립니다. 감사 인사를 전할 모든 분들의 이름은 알파벳순으로 나열됩니다.

팔리 밋

마리아 블레드소

제임스 브룩뱅크

얀 볼트만

롤리 체시

존 데이

라케쉬 두파

쉬본 도일

알렉스 엘데미르

니콜 폴스그렌

애런 길리스

켈시 하이타워

제즈 험블

데이비드 허

빅 이글레시아스

하리시 자야쿠마르

니킬 카울

리탈 레비

아만다 루이스

리오나 맥나마라

앤드류 맥빈

스티브 맥기

에린 맥킨

저신다 메인

에릭 맥스웰

라구 난단

클레어 피터스

가렛 플라스키

존 라이언

비나이 스리니바산

크리스티나 스톰

오렌 테이치

핀 토너

마르친 트레더

세스 바고

살림 버지

브레나 워싱턴

마이클 윈저

줄리아 야거 라이젠

7장 저자



더스틴 스미스

더스틴 스미스는 Google의 인적 요인 심리학자이자 직원 사용자 경험 연구 관리자이며 3년 동안 DORA 프로젝트에 참여했습니다. 지난 7년간 소프트웨어 엔지니어링, 부분 유료 구조의 게임, 의료, 군대와 같은 다양한 맥락에서 사람들이 주변 시스템과 환경에 어떻게 영향을 받는지 연구했습니다. Google에서 더스틴은 소프트웨어 개발자가 개발 중에 더 큰 만족감과 생산적인 느낌을 얻을 수 있는 영역을 식별하는 연구를 진행하며, 지난 2년 동안 DORA 프로젝트에 참여했습니다. 더스틴은 위치타 주립 대학교에서 인적 요인 심리학으로 박사 학위를 받았습니다.



다니엘라 비알바

다니엘라 비알바는 DORA 프로젝트 전담 사용자 경험 연구원입니다. 다니엘라는 개발자에게 만족감과 생산성을 선사하는 요소를 파악하는 연구에 주력합니다. Google에 합류하기 전에 다니엘라는 명상 훈련의 이점, 대학생의 경험에 영향을 미치는 심리 사회적 요인, 목격자 기억, 거짓 자백을 연구했습니다. 그녀는 플로리다 인터내셔널 대학교에서 실험 심리학 박사 학위를 받았습니다.



미셸 어바인

미셸 어바인은 Google의 테크니컬 라이터로, 개발자 도구와 이 도구의 사용자 사이의 격차를 해소하기 위해 노력하고 있습니다. Google에 입사하기 전에는 교육 출판 분야에서 일했으며 물리 시뮬레이션 소프트웨어의 테크니컬 라이터로도 활동했습니다. 미셸은 워털루 대학교에서 물리학 학사 학위와 수사학 및 커뮤니케이션 디자인 석사 학위를 받았습니다.



데이브 스탠크

데이브 스탠크는 Google의 개발자 관계 엔지니어로, 고객에게 DevOps 및 SRE 채택을 위한 관행에 대해 조언합니다. 경력 전반에 걸쳐 그는 스타트업 CTO, 제품 관리자, 고객 지원, 소프트웨어 개발자, 시스템 관리자, 그래픽 디자이너를 비롯한 다양한 역할을 수행했습니다. 그는 컬럼비아 대학교에서 기술 관리 석사 학위를 받았습니다.



네이슨 하비

네이슨 하비는 Google의 개발자 관계 엔지니어로, 팀이 비즈니스 성과에 맞게 기술을 조정하면서 잠재력을 실현하도록 지원하는 업무를 수행했습니다. 네이슨은 최고의 팀 및 오픈소스 커뮤니티와 협력하여 DevOps 및 SRE의 원칙과 관행을 적용하는 데 도움을 주었습니다. 네이슨은 2020년 O'Reilly의 '97 Things Every Cloud Engineer Should Know'를 공동 편집하고 기고했습니다.

8장

방법론

연구 설계

이 연구는 단면적 이론 기반 설계를 사용합니다. 이 이론 기반 설계는 추론적 예측으로 알려져 있으며 오늘날 비즈니스 및 기술 연구에서 수행되는 가장 일반적인 유형 중 하나입니다. 추론적 설계는 순수한 실험용 설계가 불가능하고 현장 실험이 선호되는 경우에 사용됩니다.

대상 모집단 및 샘플링

이 설문조사는 기술 및 혁신, 특히 DevOps에 익숙한 사람들과 관련되거나 긴밀하게 협력하는 실무자와 리더를 대상으로 했습니다. 이메일 목록, 온라인 프로모션, 온라인 패널, 소셜 미디어를 통해 설문조사를 홍보하고 사람들에게 설문조사를 네트워크와 공유하도록 요청했습니다(즉, 눈덩이 표집).

잠재 구성 생성

가능한 한 이전에 검증된 구성을 사용하여 가설과 구성을 공식화했습니다. 이론, 정의, 전문가 의견을 기반으로 새로운 구성을 개발했습니다. 그런 다음 설문조사에서 수집된 데이터가 신뢰할 수 있고 유효할 가능성이 높은지 확인하기 위해 의도를 명확히 하기 위한 추가 조치를 취했습니다.¹⁴

통계 분석 방법

클러스터 분석. 클러스터 분석을 사용하여 배포 빈도, 리드 타임, 서비스 복원 시간 및 변경 실패율을 기반으로 소프트웨어 배포 성과 프로필을 식별했습니다. 미리 정해진 수의 클러스터를 가질만한 산업적 또는 이론적 이유가 없기 때문에 잠재 클래스 분석¹⁵을 사용했으며 최적의 클러스터 수를 결정하기 위해 Bayesian 정보 기준¹⁶을 사용했습니다.

평가 모델. 분석을 수행하기 전에 varimax 회전을 사용하는 주요 구성 요소 분석과 탐색적 요인 분석을 사용하여 구성을 식별했습니다.¹⁷ 평균 분산 추출(AVE), 상관관계, 크론바흐 알파 계수¹⁸ 및 합성 신뢰도를 사용하여 수렴, 확산적 타당도 및 신뢰도에 대한 통계적 검정을 확인했습니다.

구조방정식 모델링. 상관관계 기반 SEM인 부분 최소 제곱(PLS) 분석을 사용하여 구조방정식 모델(SEM)을 테스트했습니다.¹⁹

14 Churchill Jr, G. A. "A paradigm for developing better measures of marketing constructs," *Journal of Marketing Research* 16:1, (1979), 64–73.

15 Hagenaars, J. A., & McCutcheon, A. L. (Eds.). (2002). *Applied latent class analysis*. Cambridge University Press.

16 Vrieze, S. I. (2012). Model selection and psychological theory: a discussion of the differences between the Akaike information criterion (AIC) and the Bayesian information criterion (BIC). *Psychological methods*, 17(2), 228.

17 Straub, D., Boudreau, M. C., & Gefen, D. (2004). Validation guidelines for IS positivist research. *Communications of the Association for Information systems*, 13(1), 24.

18 Nunnally, J.C. *Psychometric Theory*. New York: McGraw-Hill, 1978

19 Hair Jr, J. F., Hult, G. T. M., Ringle, C. M., & Sarstedt, M. (2021). "A primer on partial least squares structural equation modeling (PLS-SEM)." Sage publications.

9장

추가 자료

DevOps 기능에 대한 자세한 내용:

<https://cloud.google.com/devops/capabilities>

사이트 안정성 엔지니어링(SRE)에 대한 리소스:

<https://sre.google>

DevOps 빠른 점검 받기:

<https://www.devops-research.com/quickcheck.html>

DevOps 연구 프로그램 살펴보기:

<https://www.devops-research.com/research.html>

Google Cloud 애플리케이션 현대화 프로그램에 대해 알아보기:

<https://cloud.google.com/camp>

백서 'DevOps 변환의 ROI: 현대화 이니셔티브의 영향을 정량화하는 방법' 참조:

<https://cloud.google.com/resources/roi-of-devops-transformation-whitepaper>

이전 State of DevOps 보고서 보기:

State of DevOps 2014: <https://services.google.com/fh/files/misc/state-of-devops-2014.pdf>

State of DevOps 2015: <https://services.google.com/fh/files/misc/state-of-devops-2015.pdf>

State of DevOps 2016: <https://services.google.com/fh/files/misc/state-of-devops-2016.pdf>

State of DevOps 2017: <https://services.google.com/fh/files/misc/state-of-devops-2017.pdf>

Accelerate State of DevOps 2018: <https://services.google.com/fh/files/misc/state-of-devops-2018.pdf>

Accelerate State of DevOps 2019: <https://services.google.com/fh/files/misc/state-of-devops-2019.pdf>