

발표:



Google Cloud

Accelerate State of DevOps Report 2023



프리미어 스폰서

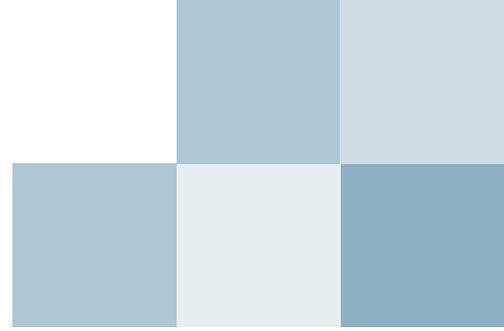


Deloitte.

Qorik

v. 2023-12

목차



서론

핵심 요약	03
개념 및 측정	06

1장

비교 방법	10
-------------	----

2장

사용자 중심 전략	17
-----------------	----

3장

기술 역량으로 실적 개선 기대	20
------------------------	----

4장

모든 업무의 기초가 되는 문서	27
------------------------	----

5장

신뢰성으로 실적 개선	31
-------------------	----

6장

유연한 인프라는 성공의 핵심	38
-----------------------	----

7장

문화에 대한 투자가 없다면 무용지물	45
---------------------------	----

8장

자신을 파악하는 것이 중요한 이유	51
--------------------------	----

후기

최종 의견	57
-------------	----

감사의 말	58
-------------	----

저자	59
----------	----

방법론	62
-----------	----

인구통계 및 기업통계	72
-------------------	----

모델	81
----------	----

추가 자료	91
-------------	----

부록	92
----------	----

모든 인용은 2023년 9월 27일에 수집됨



서론

핵심 요약

지난 10년 가까이 DORA 연구 프로그램은 실적이 우수한 기술 기반 조직의 역량과 측정값을 조사해 왔습니다. 규모와 업종을 막론한 수많은 조직에서 36,000명이 넘는 전문가의 다양한 의견을 들었습니다. 유용한 정보를 공유해 주신 모든 분께 감사드립니다.

DORA는 일하는 방식(즉, 역량)과 결과(즉, 전체 조직 및 여기에 속한 모든 사람과 관련 있는 유의미한 성과) 간의 상관관계를 이해하고자 했습니다. 이 연구는 엄격한 통계적 평가와 플랫폼에 제약을 받지 않는 접근 방식을 사용합니다 ([방법론](#) 참조).

DORA는 리더와 실무자가 이러한 유용한 정보를 통해 자신이

영향을 미칠 수 있는 부분에 대한 감각을 키울 수 있기를 바랍니다.

올해 연구에서는 3가지 주요 성과와 성과를 달성하는 데 도움을 준 역량에 대해 조사했습니다.

- **조직 실적** - 조직은 수익 뿐만 아니라 고객과 더 넓은 차원의 커뮤니티를 위한 가치도 창출해야 합니다.
- **팀 실적** - 애플리케이션 또는 서비스팀이 가치를 창출하고 혁신하고 공동작업할 수 있는 능력입니다.
- **직원 웰빙** - 조직이나 팀이 채택하는 전략은 번아웃을 줄이고, 직무 만족도를 높이고, 가치 있는 결과를 창출하기 위한 능력(즉, 생산성)을 향상시키는 등 직원에게 이점을 제공해야 합니다.

또한 이번 연구에서는 그 자체를 목적으로도 방법 또는 실적 측정값을 살펴봤습니다.

- **소프트웨어 배포 실적** - 팀이 안전하고 빠르고 효율적으로 기술 시스템을 바꿀 수 있는 능력입니다.
- **운영 실적** - 서비스가 사용자에게 신뢰할 수 있는 경험을 제공하는 능력입니다.



주요 결과



건강한 문화 조성

문화는 기술적 역량을 구축하고, 기술 실적을 증대하고, 조직의 실적 목표를 달성하며, 직원의 성공을 지원하기 위한 토대입니다. 생성적 문화를 갖춘 팀은 조직 실적이 30% 더 높습니다.



사용자를 염두에 두고 구축

사용자에게 초점을 맞추면 이 연구에서 살펴보는 모든 기술, 프로세스, 문화 관련 역량 전반에 걸쳐 도움을 받고 개선을 이룰 수 있습니다. 아무리 빠르고 성공적으로 배포하더라도 사용자를 염두에 두지 않으면 무의미합니다. 사용자에게 중점을 두는 팀은 조직 실적이 40% 더 높습니다.



보다 빠른 코드 검토로 소프트웨어 배포 실적 개선

코드 검토의 가속화는 소프트웨어 배포 실적을 개선하는 가장 효율적인 방법입니다. 코드 검토를 보다 빠르게 수행할 수 있는 팀은 소프트웨어 배포 실적이 50% 더 높습니다.



양질의 문서로 기술 역량 강화

고품질 문서는 기술 역량이 조직 실적에 미치는 영향을 강화합니다. 예를 들어 트렁크 기반 개발의 경우 고품질 문서는 저품질 문서보다 조직 실적에 미치는 영향이 12.8배 더 높은 것으로 추정됩니다.



클라우드 인프라 유연성 향상

클라우드 컴퓨팅이 유익한 이유는 유연한 인프라를 만들기 때문입니다. 예를 들어 퍼블릭 클라우드를 사용하면 클라우드를 사용하지 않을 때보다 인프라 유연성이 22% 향상됩니다. 이러한 유연성은 결과적으로 유연성이 결여된 인프라에 비해 30% 더 높은 조직 실적으로 이어집니다. 클라우드의 가치를 최대한 얻기 위해서는 클라우드가 제공하는 차별화된 특성과 역량, 즉 인프라 유연성을 활용하는 것이 핵심입니다.



배포 속도, 운영 실적, 사용자 중심 전략 간의 균형

최대한의 조직 실적을 달성하려면 강력한 소프트웨어 배포 실적과 운영 실적이 모두 필요합니다. 사용자에게 우선순위를 두고 이러한 두 실적 간에 균형을 유지하면 최상의 조직 성과를 얻는 동시에 직원 웰빙을 개선할 수 있습니다.



공정한 업무 분배

소수 집단에 속하는 것으로 식별되는 직원과 여성 또는 자신의 성별을 스스로 규정하기로 선택한 직원은 더 높은 수준의 번아웃을 경험합니다. 이러한 결과를 초래하는 여러 가지 체계적이고 환경적인 요인이 있습니다. 더 반복적인 작업을 수행하는 응답자가 더 높은 수준의 번아웃을 경험할 가능성이 높으며, 소수 집단의 구성원이 더 반복적인 작업을 수행할 가능성이 높다는 연구 결과는 놀랍지 않습니다. 소수 집단의 응답자는 그렇지 않은 집단의 응답자보다 번아웃을 24% 더 많이 경험합니다. 소수 집단의 응답자는 그렇지 않은 집단의 응답자보다 반복적인 작업을 29% 더 많이 수행합니다. 여성 또는 자신의 성별을 스스로 규정한 직원은 남성보다 반복적인 작업을 40% 더 많이 수행합니다.

DORA의 유용한 정보를 자신의 상황에 적용하기

지속적 개선의 사고방식과 관행을 채택하는 팀은 가장 많은 이점을 누릴 가능성이 높습니다.¹ DORA는 여러분의 개선 이니셔티브에 좋은 방향으로 영향을 미칠 수 있습니다.

이 연구를 최대한 활용하려면 본인의 팀과 사용자의 상황에 대입해 보세요. 예를 들어 앞서 코드 검토를 보다 빠르게 수행할 수 있는 팀은 소프트웨어 배포 실적이 50% 더 높다고 언급했습니다. 하지만 코드 검토는 빠르게 수행되지만 그 속도가 시스템상의 다른 곳에서 제한을 받는 경우 소프트웨어 배포 실적이 개선되지 않을 수 있습니다. 연구를 상황에 맞게 대입하는 것은 실무자들이 매일 업무 상황에 대한 대화를 나눌 때 비로소 가능합니다. 이러한 대화는 서로 공감하고 공동작업하고 각 참여자의 동기를 이해하는 데 도움이 될 수 있습니다.

개선 작업에는 끝이 없습니다. 시스템에서 병목 현상을 찾아내고 문제를 해결한 후에 과정을 반복해야 합니다. 가장 중요한 비교는 동일한 애플리케이션을 시간 흐름에 따라 살피는 것이지 다른 애플리케이션, 조직 또는 산업을 살피는 것이 아닙니다.

측정항목과 측정

측정항목과 대시보드는 팀이 진행 상황과 올바른 방향을 모니터링할 수 있게 해줍니다.

실무자와 리더는 조직 실적과 팀 실적, 직원 웰빙을 위해 노력하고 있습니다. 소프트웨어를 배포하는 것이 목표가 아닌 것처럼 측정 역시 목표가 아닙니다.

실적 측정항목을 고착화하면 비효율적인 행동으로 이어질 수 있습니다. 역량과 학습에 투자하는 것이 성공을 위한 더 나은 방법입니다. 가장 많이 학습하는 팀이 가장 많은 개선을 이룰 수 있습니다.

혼자만으로 이룰 수 있는 개선은 없습니다

우리 모두는 서로의 경험을 통해 배울 수 있습니다.

DORA Community 사이트(<https://dora.community>)에서 개선 이니셔티브에 관한 의견을 나누고 배워보시기 바랍니다.

¹2022 Accelerate State of DevOps Report. <https://dora.dev/research/2022/dora-report/2022-dora-accelerate-state-of-devops-report.pdf#page=7>



개념 및 측정

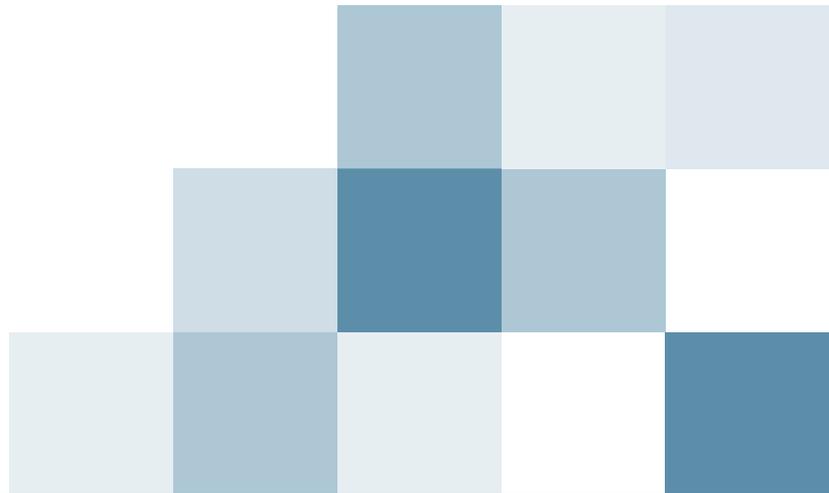
이 섹션에서는 DORA가 측정하고자 하는 개념을 설명합니다.¹ 개념은 보고서와 DORA 모델의 기초가 됩니다. 이러한 섹션에는 보고서를 작성하기 위해 사용한 자료가 포함되어 있습니다. 따라서 DORA는 보고서의 저자로서 이러한 개념이 무엇인지 명확히 알고 이를 이야기하는 방식에서 일관성을 유지하는 것이 중요합니다. 다음 표는 공유된 용어에 대해 독자와 저자 모두에게 명확성과 근거를 제공하기 위해 작성되었습니다.

이 보고서의 개념 중 다수는 다면적이기 때문에 개념을 파악하기 위해 여러 가지 지표를 사용하는 경우가 많습니다. 이러한 개념을 성공적으로 파악했는지 평가하는 방법 중 하나는 탐색적 요인 분석과 확인적 요인 분석을 사용하는 것입니다. 프로세스에 관한 자세한 내용은 [방법론](#)에서 확인할 수 있습니다. 측정 방법을 평가한 후에는 0~10 점의 척도를 설정했습니다. 0점은 개념의 존재가 완전히 결여되었다는 의미이고 10점은 개념의 존재가 최대치라는 의미입니다. 이를 통해 개념이 어떻게 작동하는지 표현하는 방법을 표준화하고 연도별 데이터를 쉽게 비교할 수 있습니다.

여기서 논의하는 각 개념에는 다음 정보가 수반됩니다.

- 이 장을 참고 자료로 사용할 때 의미를 전달하고 쉽게 찾을 수 있도록 돕는 아이콘
- 샘플에서 특정 개념에 대한 평균 점수(평균값)
- 사분위수 범위의 경계(IQR). 데이터의 중위 50%가 속하는 두 개의 수치(25번째 및 75번째 백분위수)를 제공하여 이러한 경계가 응답 분포를 전달하도록 합니다.
- 데이터 세트의 중간값(중앙값). 평균값과의 차이가 클 경우 데이터가 왜곡되었음을 나타냅니다.
- 개념 및 개념을 측정하는 방법의 설명

¹분석에 사용된 설문조사 질문은 <https://dora.dev>에 게시되어 있습니다.



주요 성과

주요 결과는 개인, 팀 또는 조직 차원에서 달성(예: 조직 실적) 또는 회피(예: 번아웃)하기 위해 노력하는 목표입니다. 따라서 DORA는 측정이 자신과 자신의 팀, 조직을 평가하는 중요한 방법이라고 생각합니다.

조직 실적			팀 실적			소프트웨어 배포 실적		
평균값	IQR	중앙값	평균값	IQR	중앙값	평균값	IQR	중앙값
6.3	5-8	6.3	7.6	6.6~9	8	6.3	5.1~7.8	6.4
<p>실적이 우수한 조직은 기본 제품이나 서비스에 대해 더 많은 고객과 더 높은 수익을 보유하고 있으며 상대적 시장점유율도 더 높습니다.</p>			<p>실적이 우수한 팀은 변화에 적응하고 서로를 지원하며, 효율적으로 일하고 혁신하고 협업합니다.</p>			<p>다음 4개 측정항목은 소프트웨어 배포의 속도와 안정성을 측정합니다.</p> <ul style="list-style-type: none"> • 배포 빈도 • 변경 리드 타임 • 변경 실패율 • 배포 실패 복구 시간 		
운영 실적			신뢰성 목표					
평균값	IQR	중앙값	평균값	IQR	중앙값			
6.2	5-7.5	6.3	7	5-7.5	7.5			
<p>가용성 및 실적 등의 측정치를 포함하여 서비스가 사용자의 기대를 충족할 수 있는 정도를 나타냅니다.</p>			<p>가용성, 실적, 정확성 등의 측정치에 대한 규정된 목표를 서비스가 충족하는 정도를 나타냅니다.</p>					

웰빙 수준은 번아웃, 생산성, 직무 만족도가 복합적으로 작용하여 결정됩니다.

번아웃			생산성			직무 만족도		
평균값	IQR	중앙값	평균값	IQR	중앙값	평균값	IQR	중앙값
4.1	2~6	4	7.5	6.7~8.8	7.9	6.08	5.7~7.1	7.1
<p>업무로 인한 심리적, 신체적 대가뿐만 아니라 업무의 가치와 의미를 어떻게 평가하는지를 나타냅니다. 번아웃은 냉소주의를 초래합니다.²</p>			<p>생산성이 높은 직원은 자신의 능력에 맞고 가치를 창출하며 효율적으로 업무를 수행할 수 있도록 하는 일을 합니다.</p>			<p>응답자에게 모든 사항을 고려하여 자신의 직무에 대해 전반적으로 느끼는 바를 평가하도록 하는 단일 항목 질문입니다.³</p>		

²마슬라흐 C, 레이터 MP. Understanding the burnout experience: recent research and its implications for psychiatry(번아웃 경험의 이해: 최근 연구 및 번아웃이 정신의학에 미치는 영향). World Psychiatry. 2016 Jun;15(2):103-11. doi: 10.1002/wps.20311. PMID: 27265691; PMCID: PMC4911781.

³워, P., 쿡, J., 윌, T. 'Scales for the measurement of some work attitudes and aspects of psychological well-being(업무 태도와 심리적 웰빙의 일부 측면을 측정하기 위한 척도).' Journal of Occupational Psychology, 52(2), 1979. 129~148. <https://doi.org/10.1111/j.2044-8325.1979.tb00448.x>

프로세스 및 기술 역량

팀 또는 조직에서 발생할 수 있는 활동, 관행 또는 상태를 말합니다. 달리 말하면, 팀이 수행하는 업무 또는 팀이 존재하는 방식입니다.

<div data-bbox="162 426 232 495"></div> <div data-bbox="250 441 407 476">인공지능 기여</div> <table border="1"> <thead> <tr> <th>평균값</th> <th>IQR</th> <th>중앙값</th> </tr> </thead> <tbody> <tr> <td>3.3</td> <td>0.3~6.3</td> <td>2.4</td> </tr> </tbody> </table> <p>다양한 기술적 과제에 기여하는 데 있어 인공지능이 수행하는 역할의 중요성을 평가합니다.</p>	평균값	IQR	중앙값	3.3	0.3~6.3	2.4	<div data-bbox="621 426 691 495"></div> <div data-bbox="709 426 893 489">느슨하게 결합된 아키텍처</div> <table border="1"> <thead> <tr> <th>평균값</th> <th>IQR</th> <th>중앙값</th> </tr> </thead> <tbody> <tr> <td>6.4</td> <td>4.7~8.3</td> <td>6.7</td> </tr> </tbody> </table> <p>독립적으로 작성, 테스트, 배포될 수 있는 소프트웨어인지 평가합니다.</p>	평균값	IQR	중앙값	6.4	4.7~8.3	6.7	<div data-bbox="1083 426 1153 495"></div> <div data-bbox="1170 441 1308 476">지속적 통합</div> <table border="1"> <thead> <tr> <th>평균값</th> <th>IQR</th> <th>중앙값</th> </tr> </thead> <tbody> <tr> <td>6.9</td> <td>5~8.9</td> <td>7.8</td> </tr> </tbody> </table> <p>소프트웨어 변경사항을 자동으로 빌드하고 테스트하는 방식을 평가합니다.</p>	평균값	IQR	중앙값	6.9	5~8.9	7.8
평균값	IQR	중앙값																		
3.3	0.3~6.3	2.4																		
평균값	IQR	중앙값																		
6.4	4.7~8.3	6.7																		
평균값	IQR	중앙값																		
6.9	5~8.9	7.8																		
<div data-bbox="162 850 232 919"></div> <div data-bbox="250 865 383 900">지속적 배포</div> <table border="1"> <thead> <tr> <th>평균값</th> <th>IQR</th> <th>중앙값</th> </tr> </thead> <tbody> <tr> <td>7.0</td> <td>5.7~8.7</td> <td>7.3</td> </tr> </tbody> </table> <p>새로운 기능, 구성 변경사항, 버그 수정, 실험 등 모든 유형의 변경사항을 안전하고 빠르게 지속 가능한 방식으로 프로덕션에 적용하거나 사용자에게 제공하는 능력을 평가합니다.⁴</p>	평균값	IQR	중앙값	7.0	5.7~8.7	7.3	<div data-bbox="621 850 691 919"></div> <div data-bbox="709 865 873 900">코드 검토 속도</div> <table border="1"> <thead> <tr> <th>평균값</th> <th>IQR</th> <th>중앙값</th> </tr> </thead> <tbody> <tr> <td>6.5</td> <td>6~8</td> <td>6</td> </tr> </tbody> </table> <p>pull 요청부터 코드 변경 검토까지 걸리는 시간을 평가하는 단일 항목입니다.</p>	평균값	IQR	중앙값	6.5	6~8	6	<div data-bbox="1083 850 1153 919"></div> <div data-bbox="1170 865 1227 900">문서</div> <table border="1"> <thead> <tr> <th>평균값</th> <th>IQR</th> <th>중앙값</th> </tr> </thead> <tbody> <tr> <td>5.8</td> <td>3.8~7.9</td> <td>6.25</td> </tr> </tbody> </table> <p>조직 내 직원이 일상 업무에서 작성하여 사용하는 콘텐츠의 품질을 평가합니다.</p>	평균값	IQR	중앙값	5.8	3.8~7.9	6.25
평균값	IQR	중앙값																		
7.0	5.7~8.7	7.3																		
평균값	IQR	중앙값																		
6.5	6~8	6																		
평균값	IQR	중앙값																		
5.8	3.8~7.9	6.25																		
<div data-bbox="162 1339 232 1409"></div> <div data-bbox="250 1354 383 1390">신뢰성 관행</div> <table border="1"> <thead> <tr> <th>평균값</th> <th>IQR</th> <th>중앙값</th> </tr> </thead> <tbody> <tr> <td>5.9</td> <td>3.9~8.3</td> <td>6.1</td> </tr> </tbody> </table> <p>팀이 서비스의 운영 실적을 개선하기 위해 사용하는 활동 및 방식을 평가합니다.</p>	평균값	IQR	중앙값	5.9	3.9~8.3	6.1	<div data-bbox="621 1339 691 1409"></div> <div data-bbox="709 1354 899 1390">트렁크 기반 개발</div> <table border="1"> <thead> <tr> <th>평균값</th> <th>IQR</th> <th>중앙값</th> </tr> </thead> <tbody> <tr> <td>5.6</td> <td>3.9~7.8</td> <td>5.6</td> </tr> </tbody> </table> <p>버전 제어 시스템의 기본 코드 브랜치로 정기적으로 병합되는 작고 빈번한 변경을 수행하는 방식을 평가합니다.</p>	평균값	IQR	중앙값	5.6	3.9~7.8	5.6	<div data-bbox="1083 1339 1153 1409"></div> <div data-bbox="1170 1354 1334 1390">유연한 인프라</div> <table border="1"> <thead> <tr> <th>평균값</th> <th>IQR</th> <th>중앙값</th> </tr> </thead> <tbody> <tr> <td>6.6</td> <td>5~8.3</td> <td>7.3</td> </tr> </tbody> </table> <p>탄력적이고 액세스 가능하며 측정할 수 있는 확장 가능한 인프라인지 평가합니다.⁵</p>	평균값	IQR	중앙값	6.6	5~8.3	7.3
평균값	IQR	중앙값																		
5.9	3.9~8.3	6.1																		
평균값	IQR	중앙값																		
5.6	3.9~7.8	5.6																		
평균값	IQR	중앙값																		
6.6	5~8.3	7.3																		

⁴What is Continuous Delivery(지속적 배포란 무엇인가요?) <https://continuousdelivery.com/>

⁵National Institute of Standards and Technology (2018) NIST The NIST Definition of Cloud Computing(2018 미국 국립표준기술연구소(NIST)의 클라우드 컴퓨팅에 대한 NIST 정의). <https://csrc.nist.gov/pubs/sp/800/145/final>에서 이용 가능

문화적 측면

문화를 규정하기란 쉽지 않지만 우세한 규범(예: 유연성), 일반적인 방향(예: 사용자 중심 전략), 직장 분위기(예: 조직 안정성)라고 말할 수도 있습니다.

<div style="background-color: #4a86e8; color: white; padding: 5px; display: flex; align-items: center;"> 업무 분배 </div> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="padding: 5px;">평균값</th> <th style="padding: 5px;">IQR</th> <th style="padding: 5px;">중앙값</th> </tr> </thead> <tbody> <tr> <td style="padding: 5px;">5.8</td> <td style="padding: 5px;">3.8~7.9</td> <td style="padding: 5px;">5.8</td> </tr> </tbody> </table> <p style="padding: 10px 10px 0 10px;">직원이 팀 내에서 업무를 공정하게 분배할 수 있도록 도와주는 공식 프로세스를 평가합니다.</p>	평균값	IQR	중앙값	5.8	3.8~7.9	5.8	<div style="background-color: #4a86e8; color: white; padding: 5px; display: flex; align-items: center;"> 유연성 </div> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="padding: 5px;">평균값</th> <th style="padding: 5px;">IQR</th> <th style="padding: 5px;">중앙값</th> </tr> </thead> <tbody> <tr> <td style="padding: 5px;">7.7</td> <td style="padding: 5px;">6.6~8.9</td> <td style="padding: 5px;">8.3</td> </tr> </tbody> </table> <p style="padding: 10px 10px 0 10px;">직원이 업무를 수행하는 방법, 장소, 시간이 얼마나 자유로운지를 평가합니다.⁶</p>	평균값	IQR	중앙값	7.7	6.6~8.9	8.3	<div style="background-color: #4a86e8; color: white; padding: 5px; display: flex; align-items: center;"> 고용 보장 </div> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="padding: 5px;">평균값</th> <th style="padding: 5px;">IQR</th> <th style="padding: 5px;">중앙값</th> </tr> </thead> <tbody> <tr> <td style="padding: 5px;">5.9</td> <td style="padding: 5px;">3.3~8.3</td> <td style="padding: 5px;">6.7</td> </tr> </tbody> </table> <p style="padding: 10px 10px 0 10px;">직원들에게 고용 보장에 대해 얼마나 자주 걱정하는지 물어보는 단일 항목 측정입니다. 점수가 높을수록 걱정이 적음을 나타냅니다.</p>	평균값	IQR	중앙값	5.9	3.3~8.3	6.7
평균값	IQR	중앙값																		
5.8	3.8~7.9	5.8																		
평균값	IQR	중앙값																		
7.7	6.6~8.9	8.3																		
평균값	IQR	중앙값																		
5.9	3.3~8.3	6.7																		
<div style="background-color: #4a86e8; color: white; padding: 5px; display: flex; align-items: center;"> 조직 안정성 </div> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="padding: 5px;">평균값</th> <th style="padding: 5px;">IQR</th> <th style="padding: 5px;">중앙값</th> </tr> </thead> <tbody> <tr> <td style="padding: 5px;">7.2</td> <td style="padding: 5px;">6.7~8.3</td> <td style="padding: 5px;">8.3</td> </tr> </tbody> </table> <p style="padding: 10px 10px 0 10px;">직원을 위한 업무 환경이 얼마나 안정적인지 또는 불안정한지 물어보는 단일 항목 측정입니다.</p>	평균값	IQR	중앙값	7.2	6.7~8.3	8.3	<div style="background-color: #4a86e8; color: white; padding: 5px; display: flex; align-items: center;"> 지식 공유 </div> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="padding: 5px;">평균값</th> <th style="padding: 5px;">IQR</th> <th style="padding: 5px;">중앙값</th> </tr> </thead> <tbody> <tr> <td style="padding: 5px;">6.4</td> <td style="padding: 5px;">5.0~8.3</td> <td style="padding: 5px;">6.7</td> </tr> </tbody> </table> <p style="padding: 10px 10px 0 10px;">아이디어와 정보가 조직 전체에 확산되는 방법을 평가합니다. 팀 구성원이 질문에 답을 한 다음 정보를 다른 팀원에게 제공합니다. 답변을 기다릴 필요는 없습니다.⁷</p>	평균값	IQR	중앙값	6.4	5.0~8.3	6.7	<div style="background-color: #4a86e8; color: white; padding: 5px; display: flex; align-items: center;"> 사용자 중심주의 </div> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="padding: 5px;">평균값</th> <th style="padding: 5px;">IQR</th> <th style="padding: 5px;">중앙값</th> </tr> </thead> <tbody> <tr> <td style="padding: 5px;">7.8</td> <td style="padding: 5px;">5.6~8.3</td> <td style="padding: 5px;">7.8</td> </tr> </tbody> </table> <p style="padding: 10px 10px 0 10px;">제품 및 서비스를 더 좋게 만들기 위한 사용자 요구와 목표를 이해하고 통합하는 방법을 평가합니다.⁸</p>	평균값	IQR	중앙값	7.8	5.6~8.3	7.8
평균값	IQR	중앙값																		
7.2	6.7~8.3	8.3																		
평균값	IQR	중앙값																		
6.4	5.0~8.3	6.7																		
평균값	IQR	중앙값																		
7.8	5.6~8.3	7.8																		
<div style="background-color: #4a86e8; color: white; padding: 5px; display: flex; align-items: center;"> 웨스트럼 조직 문화 </div> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="padding: 5px;">평균값</th> <th style="padding: 5px;">IQR</th> <th style="padding: 5px;">중앙값</th> </tr> </thead> <tbody> <tr> <td style="padding: 5px;">7.3</td> <td style="padding: 5px;">6.1~8.6</td> <td style="padding: 5px;">7.8</td> </tr> </tbody> </table> <p style="padding: 10px 10px 0 10px;">조직이 문제와 기회에 대응하는 방법을 평가합니다. 문화에는 생성적 문화, 관료적 문화, 병리적 문화 등 3가지 유형이 있습니다.⁹</p>	평균값	IQR	중앙값	7.3	6.1~8.6	7.8	<p>⁶시프린, 니콜 V., 제시 S. 미셸. 'Flexible work arrangements and employee health: A meta-analytic review(탄력 근무제 및 직원건강: 메타분석 검토).' <i>Work & Stress</i> 36, no. 1, 2022. 60~85</p> <p>⁷'2022 Developer Survey(2022년 개발자 설문조사)' https://survey.stackoverflow.co/2022#overview</p> <p>⁸커스틴 마이크. Project to Product: How to survive and thrive in the age of digital disruption with the flow framework(프로젝트에서 제품까지: 흐름 프레임워크를 사용하여 디지털 혁신 시대에서 살아남고 성공하는 방법) (IT Revolution, 2018), 54. https://itrevolution.com/product/project-to-product/</p> <p>⁹웨스트럼 R. 'A typology of organisational cultures(조직 문화 유형 분류 체계).' <i>BMJ Quality & Safety</i>, 2004. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1765804/</p>													
평균값	IQR	중앙값																		
7.3	6.1~8.6	7.8																		

비교 방법

굿하트의 법칙: 측정값이
목표가 되면 더 이상 좋은
측정값이 아니다.¹

핵심사항

실적을 개선하기 위한 첫 번째 단계는 애플리케이션의 현재 소프트웨어 배포 실적, 운영 실적, 사용자 중심 전략을 위한 기준을 세우는 것입니다. 이러한 측정값은 팀이 현재 성과를 평가하고 장기적으로 상황이 어떻게 변할지 알 수 있는 좋은 신호가 됩니다.

하지만 이러한 측정값은 팀이 개선할 수 있는 수단은 아닙니다. 이 기준을 가지고 다양한 인력, 프로세스, 기술 역량 전반에서 팀의 강점을 평가하여 앞으로 나아가는 데 방해가 될 수 있는 요소를 식별하는 것이 중요합니다.² 그런 다음 팀은 조정, 실험, 재평가를 위한 시간과 공간이 필요합니다. 이 프로세스를 반복 수행하면 팀이 지속적 개선에 대한 사고방식과 관행을 채택할 수 있습니다.

¹ 스트레던 매릴린(1997). 'Improving ratings': audit in the British University system('평가 개선': 영국 대학 시스템 감사). European Review. John Wiley & Sons. 5 (3): 305-321. doi:10.1002/(SICI)1234-981X(199707)5:3<305::AID-EURO184>3.0.CO;2-4. S2CID 145644958.

² Capability catalog(역량 카탈로그)(<https://dora.dev/devops-capabilities/>)에 나열된 보고서와 리소스가 도움이 될 수 있습니다.

³ 폴스그렌, N., 스토리, M-A.외. 'The SPACE of Developer Productivity: There's more to it than you think(개발자 생산성의 SPACE: 생각보다 많은 고려사항).', 2021. <https://queue.acm.org/detail.cfm?id=3454124>

이러한 비교를 수행할 때는 다음을 비롯한 여러 함정에 주의해야 합니다.

- **상이한 대상 비교.** 이러한 클러스터만을 기준으로 애플리케이션을 비교하는 것은 그다지 유용하지 않을 수 있습니다. 이렇게 하면 개선 목표에 악영향을 미치는 방식으로 각 애플리케이션의 상황을 무시하게 됩니다.
- **측정항목을 목표로 설정.** 굿하트의 법칙을 무시하고 "모든 애플리케이션은 연말까지 '뛰어난' 실적을 내야 합니다."라고 포괄적인 지시를 내리면 팀들이 측정항목을 달성하기 위해 경쟁을 하게 될 가능성이 커집니다.
- **모두를 통제하는 하나의 측정항목.** '중요한 하나의 측정항목'으로 복잡한 시스템을 측정하려는 시도를 말합니다. 여러 측정항목을 조합해서 사용해야 더 깊이 이해할 수 있습니다.³
- **좁은 범위의 측정항목.** 팀이나 조직은 가장 의미 있는 측정항목보다 가장 측정하기 쉬운 항목을 선택하는 경향이 있습니다.
- **산업을 개선의 방패막이로 사용.** 예를 들어 규제가 심한 산업의 일부 팀은 현상 유지를 방해하지 않기 위한 구실로 규제를 사용할 수 있습니다.

지속적 개선에 대한 사고방식과 관행을 촉진하기 위한 연구 결과와 조언을 자세히 확인하려면 'How to transform(혁신 방법)'(dora.dev/devops-capabilities/cultural/devops-culture-transform/)을 참조하세요.

소개

DORA에서는 매년 하나 이상의 클러스터 분석을 수행하여 애플리케이션 전반의 공통된 추세를 파악합니다. 이러한 분석을 통해 비교 방법을 이해하는 것이 좋지만 비교를 고착화할 필요는 없습니다. 가장 좋은 것은 저마다 상황이 다른 애플리케이션 여러 개를 비교하는 것이 아니라 동일한 애플리케이션을 시간 흐름에 따라 비교하는 것입니다.

팀은 사용자를 위한 소프트웨어를 빌드해야 하며, 사용자는 서비스의 안정성과 유용성을 최종적으로 판단하는 존재입니다. 사용자의 니즈에 집중하는 팀은 올바른 서비스를 구축할 수 있는 역량이 더 높습니다. 사용자 중심 전략과 소프트웨어 배포 실적 및 운영 실적을 결합하면 올바른 결정을 내릴 수 있는 역량도 더욱 높아집니다.

사용자의 니즈에 집중하는 팀은 올바른 서비스를 구축하고 올바른 결정을 내릴 수 있습니다.

결과

소프트웨어 배포 실적

DORA에서는 소프트웨어 배포 실적을 평가하기 위해 다음과 같은 측정항목을 사용합니다.

- **변경 리드 타임** - 변경사항이 커밋에서 배포로 이동하는 데 걸리는 시간입니다.
- **배포 빈도** - 변경사항이 프로덕션으로 푸시되는 빈도입니다.
- **변경 실패율** - 배포가 즉각적인 개입을 필요로 하는 실패를 유발하는 빈도입니다.
- **배포 실패 복구 시간** - 배포 실패에서 복구되는 데 걸리는 시간입니다.

4가지 측정항목을 모두 개선하기 위한 일반적인 접근 방법은 애플리케이션의 변경 배치 크기를 줄이는 것입니다.⁴ 변경사항이 작을수록 더 쉽게 추론하고 배포 프로세스를 진행할 수 있습니다. 변경사항이 작으면 실패 시 복구하기도 더 쉽습니다. 팀은 배포 프로세스를 빠르고 안정적으로 만들기 위해 변경사항을 최대한 줄여야 합니다. 이러한 방식으로 작업을 수행하면 변경 속도와 안정성을 모두 개선할 수 있습니다.

⁴기능은 독립적으로 배포되는 여러 변경사항으로 분류될 수 있습니다. 소프트웨어 배포 실적의 측정은 애플리케이션 또는 서비스에 대한 변경사항을 평가합니다.

올해에는 소프트웨어 배포 실적에 대한 측정항목을 조정했습니다. 이러한 변경사항에 대한 자세한 내용은 [부록](#)의 '소프트웨어 배포 실적의 측정 방법 조정'에서 읽어보세요.

다음은 올해 설문조사 응답자들의 소프트웨어 배포 실적을 나타냅니다.

실적 수준	배포 빈도	변경 리드 타임	변경 실패율	배포 실패 복구 시간	응답자 비율
엘리트 성과자	주문형	하루 미만	5%	1시간 미만	18%
높음	하루 한 번에서 일주일에 한 번 사이	하루에서 일주일 사이	10%	하루 미만	31%
중간 성과자	일주일에 한 번에서 한 달에 한 번 사이	일주일에서 한 달 사이	15%	하루에서 일주일 사이	33%
낮음	일주일에 한 번에서 한 달에 한 번 사이	일주일에서 한 달 사이	64%	1개월에서 6개월 사이	17%



운영 실적

DORA에서는 응답자에게 소속 회사의 서비스가 다음 상황에 얼마나 자주 노출되는지 물어봄으로써 운영 실적을 평가했습니다.

- 시스템 신뢰성에 불만족스러움을 느끼는 최종 사용자의 보고서 접수
- 이용 불가, 예상보다 느리게 작동 또는 올바르지 않은 작동

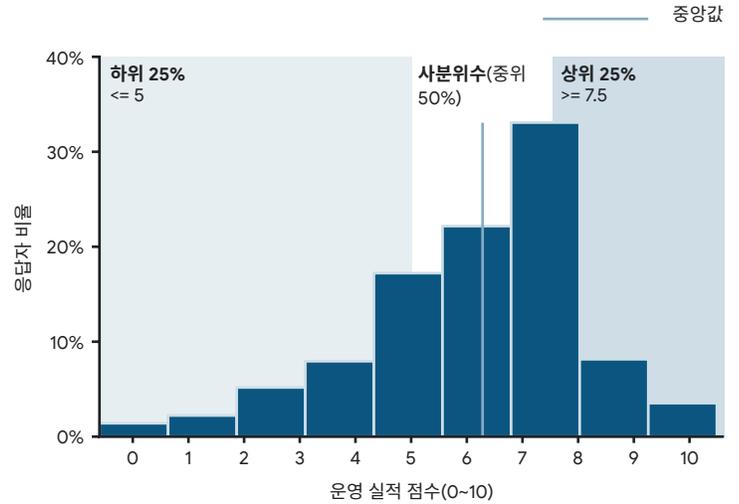
운영 실적으로 조직 실적을 예측하는 방법을 알아보려면 [5장 - 신뢰성으로 실적 개선](#)의 내용을 참조하세요.

사용자 중심 전략

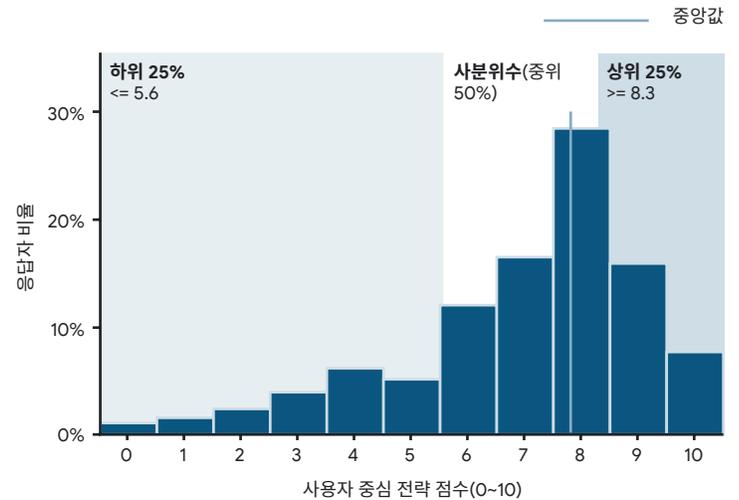
사용자 중심적인 애플리케이션 또는 서비스는 최종 사용자를 염두에 두고 빌드됩니다. 이와 같은 제품을 빌드하려면 사용자가 원하는 것을 포착하여 제품 로드맵에 통합할 수 있는 감각이 필요합니다. DORA는 응답자에게 다음 문장에 얼마나 충실한지 물어봄으로써 사용자 중심 전략을 평가했습니다.

- 팀이 사용자가 달성하고자 하는 바를 명확히 이해합니다.
- 팀의 성공이 조직과 애플리케이션 사용자에게 제공되는 가치에 따라 평가됩니다.
- 사양(예: 요건 계획)이 사용자 신호에 따라 지속적으로 재검토되고 우선순위가 재지정됩니다.

다음은 올해 설문조사 응답자들의 운영 실적을 나타냅니다.



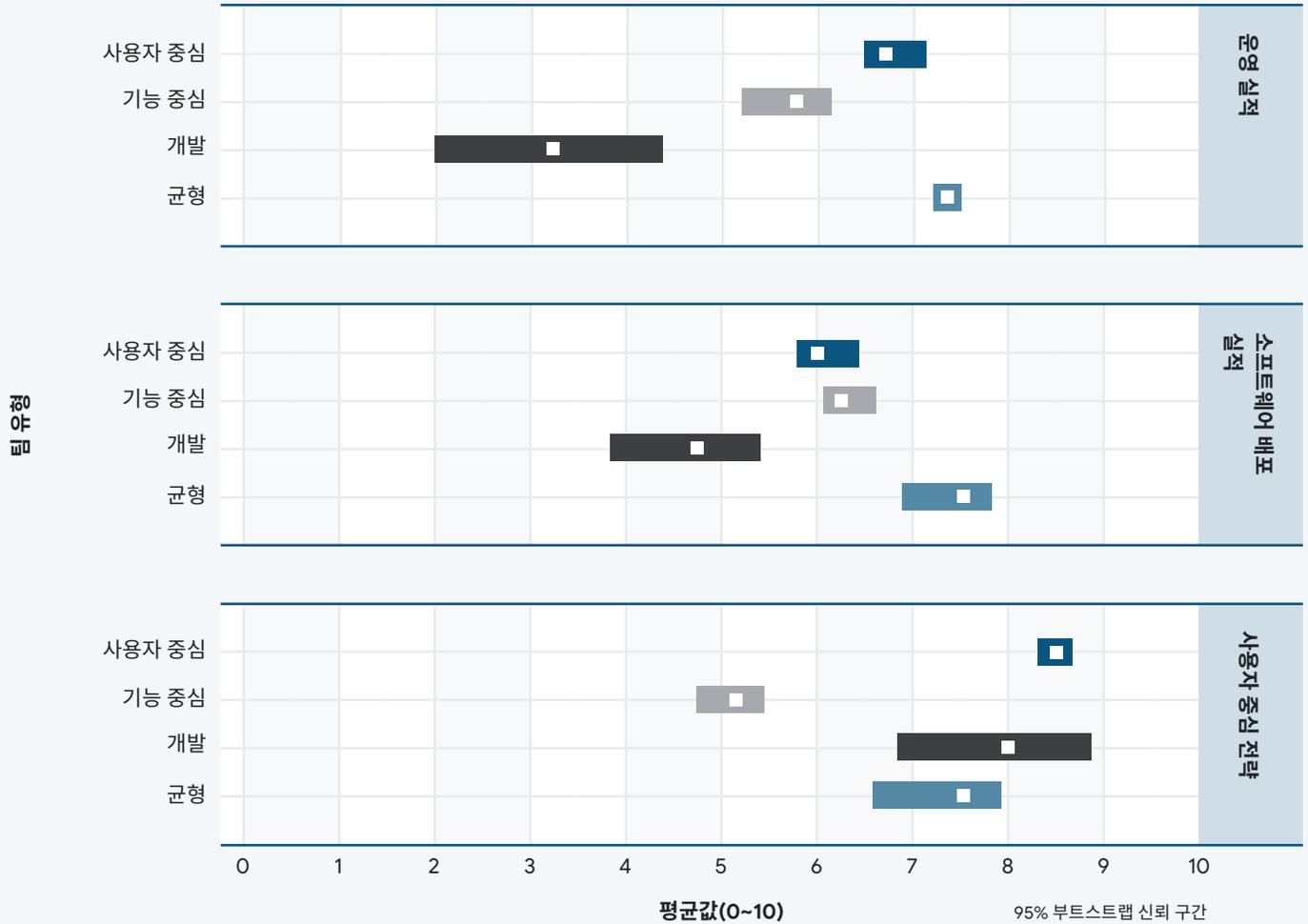
다음은 올해 설문조사 응답자들의 사용자 중심 전략을 나타냅니다.



팀 유형 분류

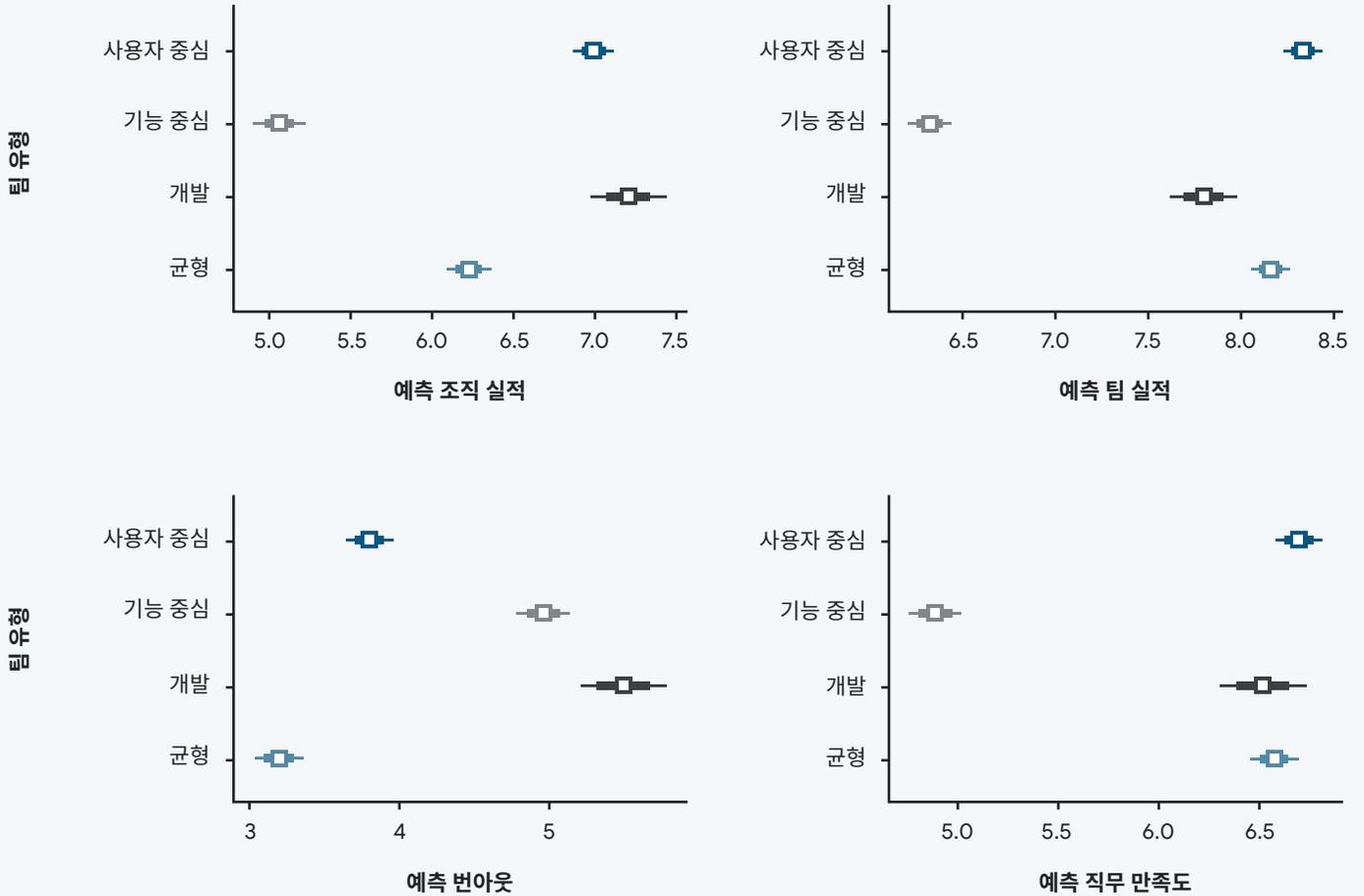
소프트웨어 배포 실적, 운영 실적, 사용자 중심 전략을 하나의 단위로 비교하면 4가지 유형의 팀으로 분류할 수 있습니다. 각 유형의 팀은 이들을 구성하는 모든 측정값과 마찬가지로 애플리케이션 또는 서비스의 수준을 나타냅니다.

각 팀에는 사용자 중심, 기능 중심, 개발, 균형이라는 이름을 붙였습니다.



앞서 살펴본 실적 측정항목을 조직이나 팀이 조직 실적, 팀 실적, 팀 내 개개인의 웰빙을 변화시키기 위한 조절 장치라고 생각해 보세요.

아래 그래프에는 팀 유형별로 예측한 실적 결과가 나와 있습니다.



*점은 팀 유형 평균의 점 추정치를 나타냅니다. 두꺼운 간격은 시뮬레이션 결과의 66%에 속한다는 의미입니다. 얇은 간격은 시뮬레이션 결과의 89%에 속한다는 의미입니다.

각 팀 유형은 고유한 특성이 있습니다. 또한 응답자의 상당 부분을 차지하며 서로 다른 결과를 산출합니다. 팀이 반드시 특정 유형에 맞는 것은 아니며, 시간이 지나도 해당 팀 유형을 일관성 있게 유지할 것이라고 기대하지 않습니다.



결과가 나타내는 의미

» 사용자 중심 팀 유형

이 팀은 사용자의 니즈에 가장 집중하는 유형입니다. 이러한 집중 전략은 강력한 소프트웨어 배포 실적 및 운영 실적과 결합하여 높은 수준의 조직 실적을 거둘 것으로 예상합니다. 하지만 균형 팀보다는 번아웃 가능성이 약간 더 높습니다. 사용자 중심 팀 유형에는 소프트웨어 배포 실적 또는 운영 실적을 개선하는 것이 번아웃을 줄일 수 있는 가장 좋은 방법일 수 있습니다.

» 기능 중심 팀 유형

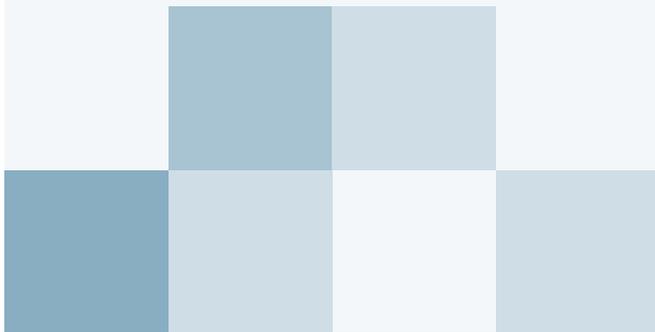
이 팀 유형은 기능 출시를 우선시합니다. 출시에만 맹목적으로 집중하면 팀이 사용자 니즈를 충족시키는 데 소홀하게 될 수 있습니다. 이는 사용자 중심 전략 및 운영 실적 부문의 낮은 수치로 알 수 있습니다. 기능 중심 팀 유형은 높은 수준의 번아웃, 낮은 수준의 직무 만족도, 팀 실적, 조직 실적을 보입니다. 직원들은 기능뿐만 아니라 가치 제공을 중요시합니다. 기능 중심 팀은 사용자의 니즈를 반영함으로써 출시되는 기능에서 더 많은 가치를 얻을 수 있습니다.

» 개발 팀 유형

이 팀 유형은 애플리케이션 사용자의 니즈에 집중하여 우수한 조직 실적을 달성합니다. 하지만 여전히 제품과 시장 간의 적합성이나 기술 역량을 개발할 수 있는 여지가 있습니다. 개발 팀은 소규모 조직에서 더 많이 발견됩니다. 이들은 소프트웨어 배포 실적과 운영 실적이 낮으며, 이러한 애플리케이션을 개발하는 팀은 균형 팀 또는 사용자 중심 팀 유형보다 더 높은 수준의 번아웃을 보고합니다. 이러한 팀은 과중한 프로세스나 번거로운 작업을 자동화하여 소프트웨어 배포 실적 및 운영 실적을 개선할 수 있습니다.

» 균형 팀 유형

이 팀 유형은 균형 있고 지속 가능한 접근 방식을 취합니다. 이들은 우수한 조직 실적, 팀 실적, 직무 만족도를 달성하기 위해 지속 가능한 방식으로 기술을 사용합니다. 이러한 팀은 또한 낮은 수준의 번아웃을 보고합니다. 균형 팀은 3가지 측정값 모두에서 좋은 실적을 낼 수 있도록 역량을 조정했습니다. 사용자 중심 전략을 개선하면 더 나은 조직 실적을 달성할 수 있습니다.



사용자에 집중하여 조직 실적 개선

핵심사항

조직은 개발자 생산성 및 조직 실적의 개선을 추구하는 과정에서 최신 기술 및 관리 경향에 지나치게 집착하게 될 수 있습니다. DORA의 연구에 따르면 애플리케이션 및 서비스 빌드를 위한 사용자 중심 접근 방식이 전반적인 조직 실적의 가장 강력한 예측 변수 중 하나인 것으로 나타났습니다. 실적을 개선하기 위해서는 사용자에게 깊이 이해하고 사용자의 피드백을 반복적으로 조정 및 통합해야 합니다.

소개

DevOps 접근 방식은 더 나은 사용자 가치를 제공하고자 개발팀과 운영팀간 더 효율적인 공동작업을 장려하기 위한 방법으로 시작되었습니다. 두 팀의 연합은 초기 성공을 거두었고 DevOps의 아이디어와 역량은 두 부서를 넘어 더 광범위하게 확장되었습니다. 오늘날 고성능의 기술 기반 조직은 조직 목표를 달성하기 위해 모든 팀을 아우르는 연합의 중요성을 인지하고 있습니다.

DORA에서는 사용자가 중심이 되는 3가지 중요 특성을 조사했습니다.

- 팀이 사용자의 니즈를 얼마나 잘 이해하는가
- 사용자 니즈를 충족하기 위해 팀이 얼마나 잘 협력하는가
- 업무 우선순위를 정할 때 사용자 피드백이 반영되는가

사용자 중심적인 팀은

40%

조직 실적이 더 높습니다

결과

소프트웨어 개발에서 사용자 중심적인 접근 방식을 추구하면 유의미한 실적 개선으로 이어진다는 결과가 나타났습니다. 조직은 사용자를 우선시할 때 막대한 이점을 얻을 수 있습니다. 사용자 피드백은 팀이 프로젝트의 우선순위를 정하고 사용자 니즈를 충족하는 제품 및 서비스를 만드는 데 도움을 줍니다. 이러한 접근 방식은 더 나은 사용자 경험과 사용자 만족도, 그리고 더 높은 수익으로 이어집니다.

사용자 중심주의의 효과		
조직 실적		크게 증가
팀 실적		크게 증가
소프트웨어 배포 실적		약간 증가
운영 실적		크게 증가
트렁크 기반 개발		크게 증가
신뢰성 관행		크게 증가
지속적 통합		크게 증가
지속적 배포		크게 증가
느슨하게 결합된 아키텍처		크게 증가
번아웃		약간 감소*
직무 만족도		크게 증가
생산성		크게 증가

*번아웃 감소는 긍정적인 징후입니다.

결과가 나타내는 의미

사용자 니즈에 집중하는 전략은 전반적인 조직 실적의 강력한 예측 변수입니다. 사용자에게 집중하는 전략을 강화하려면 적절한 인센티브, 협력, 업무 방식이 필요합니다. 사용자에게 초점을 맞추면 이 연구에서 살펴보는 모든 기술, 프로세스, 문화 관련 역량 전반에 걸쳐 도움을 받고 개선을 이룰 수 있습니다.

이러한 결과가 조직 전반의 다양한 팀에 어떤 영향을 미칠 수 있는지 살펴보겠습니다.

>> 제품 개발 및 배포팀

사용자에게 초점을 맞추면 제품 개발 및 배포팀이 지속 가능한 방식으로 사용자에게 적합한 제품을 만들 수 있습니다. 균형 유형의 팀은 이것이 가능합니다. 균형 팀은 사용자 니즈에 중점을 두고 있어 배포, 운영, 조직 실적 전반에서 강점을 보여줍니다. 이러한 팀의 구성원은 사용자 니즈를 명확하게 이해하고 사용자 피드백을 기반으로 계획을 조정할 수 있는 능력에서 이점을 얻습니다.

결과에 따르면 기능 중심 팀은 최고의 조직 실적을 달성하지 못합니다. 이러한 팀은 조직 실적과 조직 내 직원 웰빙보다는 배포 실적을 지나치게 우선시하는 것으로 나타납니다.

>> 운영팀

운영 실적에 집중하는 팀은 CPU 사용과 같은 시스템 관련 측정항목을 최적화하기 위해 많은 노력을 기울입니다. 하지만 사용자가 서비스에 기대하는 것을 파악하지 못하면 성능이 느리다는 사용자 보고서를 자주 받게 될 수 있습니다. 사용자가 중요시하는 서비스 수준 지표를 식별하고 일반 사용자의 만족도를 유지하는 데 집중하는 서비스 수준 목표를 설정하는 등의 사이트 안정성 엔지니어링(SRE) 방식은 운영팀이 보다 사용자 중심적인 사고방식을 갖는 데 도움이 될 수 있습니다.



플랫폼 엔지니어링팀

플랫폼 엔지니어링팀은 플랫폼을 빌드할 때 '만들기만 하면 사용자가 알아서 찾아올 것'이라는 생각을 가질 수 있습니다. 좀 더 성공적인 접근 방식은 개발자를 플랫폼 사용자로 인식하는 것입니다. 이렇게 중점을 두는 대상을 바꾸려면 플랫폼 엔지니어링팀은 개발자가 문제가 있는 부분을 식별하고 제거하기 위해 어떤 방식으로 일하는지 파악해야 합니다. 팀은 플랫폼 빌드를 위한 노력이 팀이 더 나은 성과를 달성하는 데 도움을 주는지 모니터링하기 위한 신호로 소프트웨어 배포 및 운영 실적 측정값을 사용할 수 있습니다.



리더

리더는 사용자에게 가치를 제공하는 팀에 보상을 주는 인센티브 구조를 만들어 사용자에게 집중하는 환경을 조성하도록 지원할 수 있습니다. 이러한 구조가 없다면 팀은 제공되는 기능의 수를 측정하거나 서비스 중단을 줄이는 역할만 한다고 느낄 수 있습니다. DORA는 혁신적인 리더십의 역할을 조사했으며¹ 이 역량을 개선할 준비가 된 리더에게 도움이 될 만한 조언을 준비했습니다. 자세한 내용은 'DevOps 역량: 혁신적인 리더십'[\(<https://dora.dev/devops-capabilities/cultural/transformational-leadership>\)](https://dora.dev/devops-capabilities/cultural/transformational-leadership)의 내용을 참조하세요.

사용자에게 집중하면

20%

더 높은 직무 만족도로 이어집니다

시작하는 데 도움이 되는 리소스

보다 사용자 중심적인 역량을 개발하는 것은 성공의 중요한 동인입니다. 2023년의 연구 결과는 제품 관리 역량을 통해 소프트웨어 배포 실적과 조직 실적을 개선할 수 있다는 2018년의 연구 결과²를 강화합니다.

고객 피드백,³ 가치 흐름에서의 업무 가시성,⁴ 소규모 배치 작업,⁵ 팀 실험⁶과 같은 사용자 집중 역량을 높여 팀 실적을 개선하세요.

¹2017 State of DevOps Report. <https://dora.dev/publications/pdf/state-of-devops-2017.pdf>, 12~19

²2018 Accelerate: State of DevOps Report: Strategies for a New Economy(새로운 경제 전략). <https://dora.dev/publications/pdf/state-of-devops-2018.pdf>, 49~51

³Customer feedback(고객 피드백). <https://dora.dev/devops-capabilities/process/customer-feedback/>

⁴Visibility of work in the value stream(가치 흐름에서의 업무 가시성). <https://dora.dev/devops-capabilities/process/work-visibility-in-value-stream/>

⁵Working in small batches(소규모 배치 작업). <https://dora.dev/devops-capabilities/process/working-in-small-batches/>

⁶Team experimentation(팀 실험). <https://dora.dev/devops-capabilities/process/team-experimentation/>

기술 역량으로 실적 개선 기대

핵심사항

리소스와 노력을 지속적 통합, 느슨하게 결합된 아키텍처, 코드 검토 속도 개선에 투자하면 조직 실적 향상, 팀 실적 향상, 소프트웨어 배포 실적 향상, 운영 실적 향상과 같은 여러 가지 유익한 결과로 이어질 수 있습니다. 이는 애플리케이션이나 서비스 개발에 참여하는 직원의 웰빙에 아무런 악영향을 끼치지 않고 일부 이점을 제공하는 경우도 있습니다.

소개

핵심 요약에서는 DORA에서 연구한 기술 역량에 대해 설명하고 이러한 기술 역량이 여러 실적과 웰빙 측정값에 어떤 영향을 미치는지 알아봤습니다. DORA는 항상 다양한 프로세스와 기술 역량으로 실적 개선이 기대되는 정도를 조사하고 정량화하는 일을 주요 작업으로 삼아왔습니다.

올해에는 다음과 기술 역량으로 실적이 얼마나 개선되는지 조사했습니다.

- 인공지능
- 트렁크 기반 개발
- 느슨하게 결합된 아키텍처
- 지속적 통합
- 신속한 코드 검토

이러한 기술 역량으로 다음과 같은 실적 측정항목을 예측한 방법도 살펴봤습니다.

- 팀 실적
- 조직 실적
- 소프트웨어 배포 실적
- 운영 실적

또한 이러한 역량과 다양한 지표 간의 연관성을 테스트하여 작업을 수행하는 사람들에게 어떤 영향을 미쳤는지 확인했습니다.

- 번아웃
- 생산성
- 직무 만족도

결과

DORA에서 연구한 기술 역량과 프로세스는 다양하지만 주요 실적 측정항목에는 전반적으로 긍정적인 영향을 줍니다.

기술 역량 및 프로세스	팀 실적에 미치는 영향	조직 실적에 미치는 영향	소프트웨어 배포 실적에 미치는 영향	운영 실적에 미치는 영향
AI	영향 없음	약간 증가	약간 감소	크게 감소
지속적 통합	약간 증가	약간 증가	약간 증가	영향 없음
코드 검토 속도	약간 증가	영향 없음	크게 증가	크게 증가
느슨하게 결합된 아키텍처	크게 증가	크게 증가	약간 증가	크게 증가
트렁크 기반 개발	약간 증가	약간 증가	약간 증가	약간 감소

느슨하게 결합된 팀 또는 느슨하게 결합된 아키텍처를 보유한 팀은 다른 팀과 협조하지 않고도 시스템에 막대한 변화를 줄 수 있습니다. 이를 통해 팀이 더 빠르게 움직일 수 있습니다. 주제 전문가가 팀에 가까워지면 변화의 영향을 더 깊이 이해할 수 있기 때문에 코드 검토를 더 빠르게 수행할 수 있습니다. 느슨하게 결합된 설계를 통해 팀은 다른 팀의 병목 현상 없이 테스트, 빌드, 배포를 실행할 수 있습니다.

느슨하게 결합된 아키텍처에서 우리의 변화에 미치는 영향이 작더라도 팀 내 다른 개발자와 충돌을 일으키지 않아야 합니다. 소규모 배치로 일하는 팀은 충돌 위험이 줄어들기 때문에 각 커밋에서 소프트웨어가 빌드되고 자동화된 테스트가 트리거되어 개발자에게 빠른 피드백이 제공됩니다.

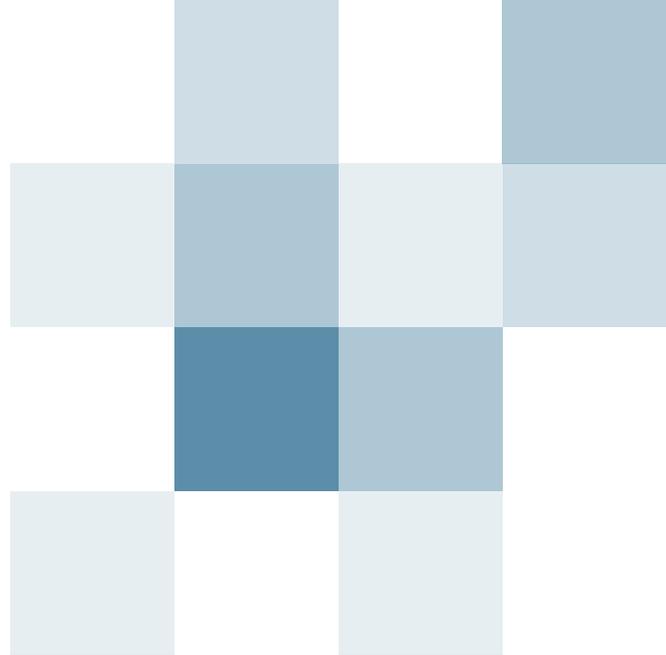
코드 검토 시간이 짧은 팀은 소프트웨어 배포 실적이 50% 더 높습니다. 효율적인 코드 검토 프로세스는 코드 개선, 지식 이전, 공유 코드 소유권, 팀 소유권, 투명성으로 이어집니다.

코드 검토가 병목 현상을 유발하나요? 코드 검토 프로세스와 코드 검토 프로세스가 리드 타임 변경에 미치는 영향을 평가하면 개선 기회를 포착할 수 있습니다. 다음 질문을 고려하세요.

- 프로세스에 피어 코드 검토가 포함되어 있는가?
- 코드 완료와 코드 검토 사이의 기간은 얼마나 되는가?
- 코드 검토의 평균 배치 크기는 어떻게 되는가?
- 검토에 얼마나 많은 팀이 참여하고 있는가?
- 검토에 얼마나 많은 지리적 위치가 참여하고 있는가?
- 팀이 코드 검토 제안을 기반으로 코드 품질 자동화를 개선하고 있는가?

코드 완료와 검토 사이의 기간이 길수록 개발자의 효율성과 배포된 소프트웨어의 품질에 부정적인 영향을 미치는 것으로 나타났습니다. 여러 지리적 위치에서 다수의 팀이 개입하면 시간이 길어지고 프로세스 참여도가 낮아지며 비용이 늘어나게 됩니다.¹

2022년 연구에서는 기술적 역량이 서로 간에 영향을 미치며 발전한다는 사실을 발견했습니다. 코드 검토 속도를 개선하면 코드 유지관리성, 학습 문화(지식 이전), 생성적 문화 형성을 비롯한 여러 기술 역량을 개선하는 데 도움이 될 수 있습니다.²



빠른 코드 검토 속도는 느슨하게 결합된 팀의 이점 중 하나로서 소프트웨어 배포 실적 및 운영 실적의 상당한 개선으로 이어집니다. 코드 검토의 효율성을 개선하기 위한 몇 가지 방법이 있습니다. 검토 중인 코드가 팀 아키텍처의 범위에만 영향을 미칠 때 검토자는 코드가 시스템에 미치는 영향을 더 깊이 이해할 수 있습니다. 코드 검토 규모가 적을수록 검토자가 변화의 결과를 더 쉽게 이해할 수 있습니다. 소규모 배치로 작업하면 팀의 피드백 주기, 효율성, 팀 집중도를 개선할 수 있습니다.³ 페어 프로그래밍은 현재 아키텍처 및 통합 관행에 관계없이 코드 검토 시간을 줄일 수 있는 방법입니다.⁴

또한 이러한 역량 및 프로세스는 작업을 수행하는 개인의 웰빙에 부정적인 영향을 미치지 않습니다. 실제로 이러한 역량 및 프로세스 대부분은 개인의 웰빙을 개선할 것으로 기대됩니다.

¹ Investigating the effectiveness of peer code review in distributed software development based on objective and subjective data(객관적, 주관적 데이터를 기반으로 분산 소프트웨어 개발 작업에서 피어 코드 검토의 효율성 조사). <https://jserd.springeropen.com/articles/10.1186/s40411-018-0058-0>

² Expectations, Outcomes, and Challenges of Modern Code Review(최신 코드 검토에 대한 기대, 결과, 과제). <https://dl.acm.org/doi/10.5555/2486788.2486882>

³ Working in small batches(소규모 배치 작업). <https://dora.dev/devops-capabilities/process/working-in-small-batches/>

⁴ On Pair Programming(페어 프로그래밍). <https://martinfowler.com/articles/on-pair-programming.html>

기술 역량 및 프로세스	번아웃에 미치는 영향*	직무 만족도에 미치는 영향	생산성에 미치는 영향
AI	↓ 약간 감소	↑ 약간 증가	↑ 약간 증가
지속적 통합	⊖ 효과 없음	↑ 약간 증가	⊖ 효과 없음
코드 검토 속도	↕ 크게 감소	↑ 약간 증가	↑ 약간 증가
느슨하게 결합된 아키텍처	↕ 크게 감소	↗ 크게 증가	↗ 크게 증가
트렁크 기반 개발	↗ 크게 증가	⊖ 효과 없음	⊖ 효과 없음

*번아웃의 경우 색 구성표가 반전되는 것을 확인할 수 있습니다. 번아웃 감소는 긍정적 효과이기 때문입니다.

느슨하게 결합된 아키텍처, 지속적 통합, 효율적인 코드 검토를 사용하면 팀이 조직 성과를 개선하는 동시에 웰빙 수준을 유지하고 때로는 개선한다는 결과가 나타났습니다.

사용자에게 가치를 제공하는 신뢰할 수 있는 시스템을 유지하고 개선하도록 자율성을 보장하면 직무 만족도, 팀 실적, 소프트웨어 배포 실적이 개선되는 효과를 경험할 수 있습니다. 아키텍처는 팀이 사용자에게 집중하고 소프트웨어 배포를 개선할 수 있는 중요한 역할을 합니다. 소규모로 시작하고 사용자에게 집중하는 팀은 트렁크 기반 개발, 느슨하게 결합된 아키텍처, 지속적 통합, 지속적 배포, SRE 전반에서 상당한 개선 효과를 거두었습니다. 기술 역량을 개선하려면 팀 실험과 지속적 개선을 위한 기회를 제공해야 합니다.⁵



⁵ Team experimentation(팀 실험): <https://dora.dev/devops-capabilities/process/team-experimentation>

지속적 배포의 이점

작성자: 데이브 팔리

지속적 배포(CD)의 기본 원리는 소프트웨어가 항상 출시 가능한 상태에 있도록 하는 것입니다. 이렇게 하기 위해서는 고품질 제품을 만들어야 합니다. 그래야 문제가 발생했을 때 쉽게 수정하여 다시 출시 가능한 상태로 빠르고 간단하게 복구할 수 있습니다.

소프트웨어를 출시 가능한 상태로 유지하려면 신속하게 피드백을 적용하고 실패에서 빠르게 복구해야 합니다.

이 연례 보고서의 독자로서 저는 이러한 아이디어가 익숙하게 들립니다. **안정성**(변경 실패율 및 배포 실패 복구 시간) 측정항목을 위해서는 품질이 중요하고 **처리량**(변경 리드 타임 및 배포 빈도) 측정항목을 위해서는 피드백과 문제 감지의 용이성이 중요합니다.

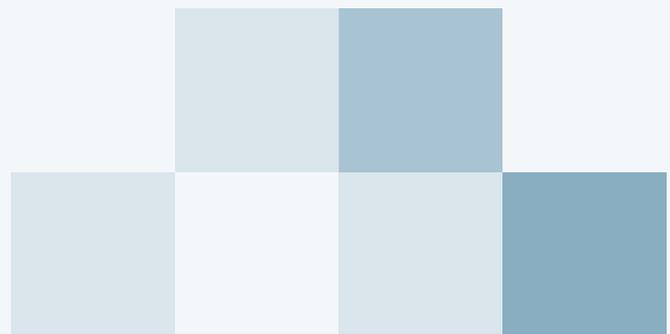
CD를 실행하면 **안정성 및 처리량** 부문에서 높은 점수를 받을 수 있습니다 CD를 실행하지 않았다면 **안정성 및 처리량** 부문에서 높은 점수를 받는다고 상상하기 어려울 것입니다.

올해 분석에는 각 역량의 매개변수를 찾아 역량이 실적을 향상하는 방법을 살펴보는 것이 포함되었습니다. CD는 모든 종류의 주문형 변경사항을 빠르고 안전하고 지속 가능하게 출시할 수 있는 기능으로 여러 기술 역량의 중요 매개변수입니다. 다시 말하면 이러한 역량은 CD를 가능하게 만드는 환경을 조성하기 때문에 효과를 발휘합니다. 결과적으로 CD의 실행은 이러한 역량으로 더 강력한 소프트웨어 배포 실적을 예측할 수 있는 메커니즘을 제공합니다.

출시 가능 상태는 소프트웨어 개발을 위해 일반적으로 충족해야 하는 주요 표준이므로 CD에서 이를 강조하는 것입니다. 출시 가능 상태는 주관적이지만 분명하며 상황에 따라 적용 가능한 품질 표준이기 때문에 중요합니다. 출시 가능 상태를 정의하는 엄격함의 수준은 안전이 중요한 시스템을 개발할 때와 케이크 매장을 위한 소프트웨어를 개발할 때가 다를 수 있습니다. 하지만 두 경우 모두 출시 가능 상태는 코드가 사용자에게 출시할 준비가 되었고 적합하며 안전하다고 말하기 위해 필요한 모든 사항을 마쳤다는 것으로 정의됩니다.

따라서 변경사항을 출시 가능하도록 유지하는 것 역시 시스템을 위해 허용 가능한 최소 품질의 상황별 정의에 맞게 최적화하는 것입니다.

높은 수준의 빠른 피드백을 받고 이를 적용하는 일을 우선시하는 팀은 소프트웨어 배포 실적이 더 높습니다.



지속적 통합(CI) 및 트렁크 기반 개발이 소프트웨어 배포 실적에 큰 영향을 미치지 못했다는 점은 다소 놀라운 결과였습니다. 특히 CI는 제 개인적으로는 꽤나 기본적인 거라고 생각했기 때문에 이런 결과는 받아들이기가 다소 힘듭니다. 하지만 이들을 따로 떼서 생각할 수는 없습니다. 예를 들어 코드가 통합되지 않았다면 어떻게 처리량 부문에서 높은 점수를 얻을 수 있을까요? 또 신뢰성을 확인하지도 않고 어떻게 안정성이 높다고 확신할 수 있을까요? CI는 우리가 이러한 사실을 파악하는 방법이며, 따라서 소프트웨어 배포 실적의 주요 매개변수라 할 수 있습니다. 이게 해석의 문제일까요, 아니면 더 심오한 무언가가 있는 걸까요? 참 궁금하네요.

기술 역량 및 프로세스	소프트웨어 배포 실적에 미치는 영향	지속적 배포를 통한 조절효과 여부*
AI	↓ 약간 감소	⊗ 없음
지속적 통합	↑ 약간 증가	☑ 예, 완전한 매개효과
코드 검토 속도	⬆ 크게 증가	☑ 예, 부분적 매개효과
느슨하게 결합된 아키텍처	↑ 약간 증가	☑ 예, 부분적 매개효과
트렁크 기반 개발	↑ 약간 증가	☑ 예, 완전한 매개효과

*매개효과는 기본적인 영향을 주는 메커니즘이나 경로를 평가하는 테스트입니다. 예를 들어 '데이터는 소프트웨어 배포 실적에 대한 트렁크 기반 개발의 영향이 지속적 배포(매개변수)를 통해 발생한다는 가설을 뒷받침합니다'라고 말할 수 있습니다. 완전한 매개효과는 전체 효과가 매개변수를 통해 설명되는 것처럼 보이는 경우 발생합니다. 부분적 매개효과는 일부 효과만이 매개변수를 통해 설명되는 경우 발생합니다.

조직 프로세스 및 역량 최적화

우리는 모두 문화가 성공을 이끈다는 것을 알고 있습니다. 그렇다면 문화를 이끄는 것은 무엇일까요? 각자가 다른 대답을 내놓을 만큼 아주 흥미로운 질문이죠. 그건 상황에 따라 다릅니다.

실무자의 관점에서는 일상 업무 방식을 개선하면 위험 분담, 공동작업 증가, 심리적 안전 확립 등 문화적 요소에 긍정적인 영향을 미칩니다. 예를 들어 버전 제어 시스템의 기본 브랜치에 정기적으로 변경사항을 통합하면 지식 공유 및 공동작업이 증가합니다. 보안팀이 개발자와 함께 일하며 코드형 정책 작업에 협조하면 팀 간에 믿음이 쌓이고 배포되는 변경사항의 신뢰도가 높아집니다.

리더십 관점에서 문화는 문화의 중요성에 대한 인식과 교육으로부터 시작합니다. 혁신적인 리더십⁶은 실험과 학습을 장려하고 실무자에게 믿음과 발언권을 주는 책임감 있는 환경을 조성하는 데 도움을 줄 수 있습니다. 엔지니어는 작업 요청에 응하는 것이 아니라 복잡한 문제를 해결하기 위해 존재하는 것입니다. 이를 위해서는 비즈니스에 대한 가시성과 조치를 취할 수 있는 자율성이 필요합니다. 궁극적으로 문화란 리더십에서부터 발생하는 것입니다.

이상적으로 최상의 결과는 문화를 상향식 관점과 하향식 관점에서 보는 것으로부터 나옵니다.

⁶ 'Transformational leadership(혁신적인 리더십)'. <https://dora.dev/devops-capabilities/cultural/transformational-leadership>

인공지능(AI)

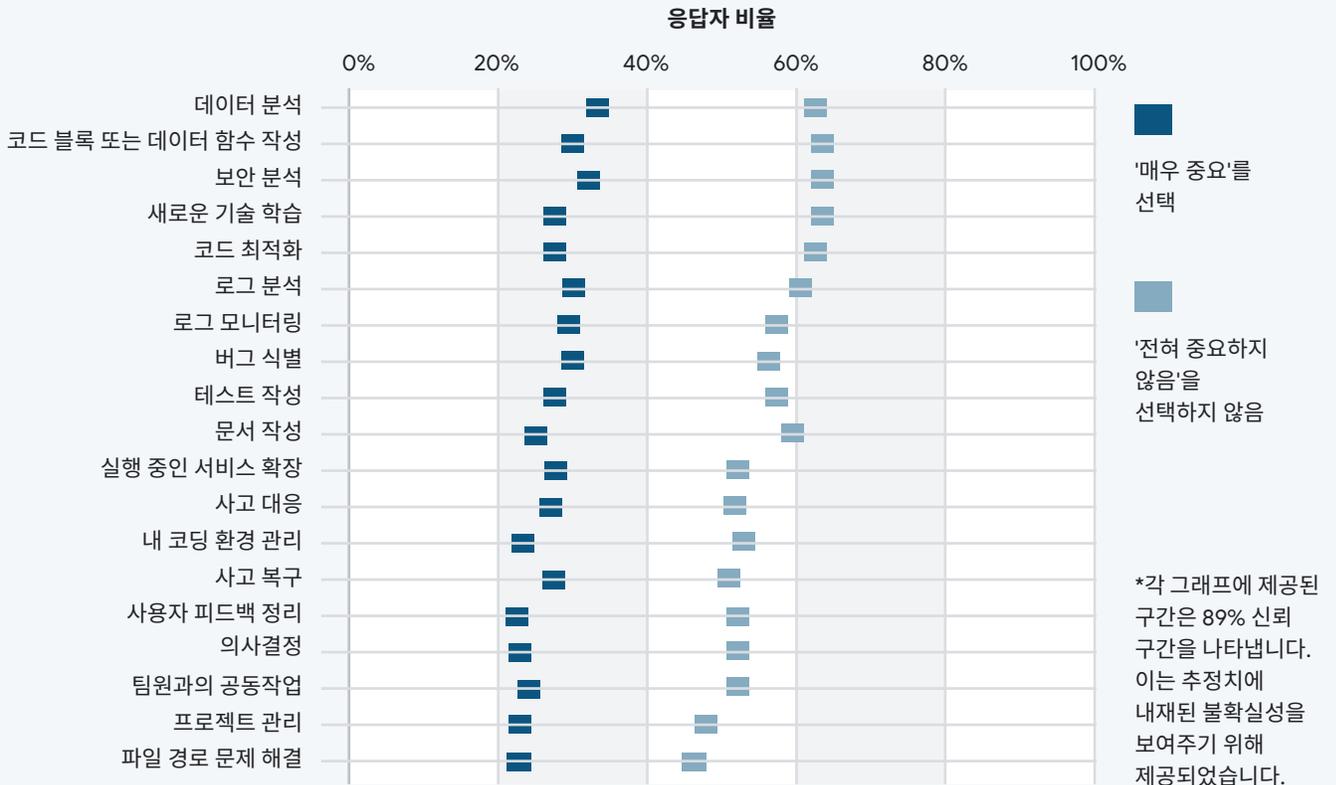
일부 분석가와 기술 전문가들은 AI가 직업 웰빙에 악영향을 끼치지 않고 소프트웨어팀의 성과를 더 높일 것이라고 예상합니다. 지금까지 DORA의 설문조사 증거는 이를 뒷받침하지 못했습니다. DORA의 연구에 따르면 AI는 개인의 웰빙 측정항목(예: 번아웃 및 직무 만족도)을 약간 개선하지만 그룹 수준의 결과(팀 실적 및 소프트웨어 배포 실적)에는 중립적이거나 부정적인 영향을 미친다는 것이 증명되었습니다.

여러 기업 사이에서 이루어지고 있는 AI 도구 도입의 초기 단계가 이러한 혼합된 증거를 설명하는 데 도움이 될 수 있을 것입니다. 일부 대기업은 다양한 AI 기반 도구를 광범위하게 사용할지에 대한 결정을 내리기 전에 시험적으로 테스트하고 있습니다. 대다수의 사람이 일부 AI를 우리가 요청한 작업에 통합하는 것에서 알 수 있듯이 AI 개발 도구의 잠재력에 대한 기대감은 엄청납니다. 이 내용은 아래 그래프에 나와 있습니다. 하지만 AI 기반 도구가 널리 확산되고 업계에서 조정되는 데에는 다소 시간이 걸릴 것으로 예상됩니다.

AI의 중요성

기술 작업에 대한 AI의 영향

작업 중인 주요 애플리케이션 또는 서비스의 경우 인공지능(AI)의 역할이 다음 작업에 영향을 미치는 데 얼마나 중요한가요?



시간이 지남에 따라 도입률이 증가하고 그러한 성장세가 실적 측정값 및 조직에서 중요시하는 결과에 미치는 영향을 지켜보는 것은 아주 흥미롭습니다.

모든 업무의 기초가 되는 문서



핵심사항

고품질의 문서는 기본입니다. 이는 기술 역량의 성공적인 구현을 유도하고 그러한 역량이 조직 실적에 미치는 영향을 강화합니다. 문서는 팀 실적, 생산성, 직무 만족도와 같은 결과에도 긍정적인 영향을 미칩니다. 하지만 문서 품질을 개선하는 것만으로는 모두에게 더 나은 웰빙을 가져다 주지 않습니다. 문서 품질이 개선되면서 번아웃 정도가 심해졌다고 보고하는 응답자도 있었습니다.

소개

올해에는 조직 내 직원이 일상적으로 사용하는 서면 지식과 같은 내부 문서를 더 깊이 살펴봤습니다. 그리고 문서가 기술 역량과 주요 결과에 미치는 영향을 조사했습니다.

문서 품질을 측정하기 위해 문서의 신뢰성, 검색 가능성, 업데이트 여부, 관련성 등을 측정했습니다. 그런 다음 전체 문서 경험에 대해 하나의 점수를 계산했습니다. 문서를 페이지별로 평가하는 게 아니라 전체적으로 평가하는 것입니다.

결과

문서는 모든 업무의 기초로서 기술 역량을 높입니다

2021년¹과 2022년²의 연구에서 밝혀낸 것처럼 문서 품질은 DORA에서 연구하는 모든 기술 역량의 성공적인 구현을 지속적으로 이끕니다.

다음 표에 나와 있는 것처럼 문서 품질은 2022년의 연구 결과³와 비슷하게 각각의 기술 역량이 조직 실적에 미치는 영향을 강화합니다.

기술 역량	조직 실적에 대한 영향의 증가율*
지속적 통합	2.4배*
지속적 배포	2.7배*
트렁크 기반 개발	12.8배*
느슨하게 결합된 아키텍처	1.2배*
신뢰성 관행	1.4배*
인공지능 기여	1.5배*

*계산 방법: 고품질 문서를 사용했을 때 기술 역량에 미치는 영향
저품질 문서를 사용했을 때 기술 역량에 미치는 영향

¹ Accelerate State of DevOps 2021.

<https://dora.dev/publications/pdf/state-of-devops-2021.pdf>

² 2022 Accelerate State of DevOps Report.

<https://dora.dev/research/2022/dora-report/2022-dora-accelerate-state-of-devops-report.pdf>

³ 2022 State of DevOps Report data deep dive: Documentation is like sunshine(데이터 심층 탐구: 햇살과 같은 문서). <https://cloud.google.com/blog/products/devops-sre/deep-dive-into-2022-state-of-devops-report-on-documentation>

문서는 생산성과 직원 및 조직 만족도를 높일 것입니다

고품질 문서는 기술 역량을 개선할 뿐만 아니라 개인의 웰빙에도 긍정적인 영향을 미쳐 번아웃을 줄이고, 직무 만족도와 생산성을 높이는 것으로 나타났습니다. 이러한 영향을 미치는 이유 중 일부는 고품질 문서가 지식 공유를 증가시키기 때문이라는 사실을 알아냈습니다.

이러한 영향은 그리 놀랍지 않습니다. 방법을 알면 작업이 쉬워지고 지식을 공유하면 불편이 줄어듭니다.

웰빙의 요소	고품질 문서의 영향
번아웃	 크게 감소
직무 만족도	 크게 증가
생산성	 크게 증가

고품질 문서는 팀 실적, 조직 실적, 운영 실적과 같은 핵심 결과물에도 영향을 미칩니다.

주요 성과	고품질 문서의 영향
팀 실적	 크게 증가
조직 실적	 크게 증가
소프트웨어 배포 실적	 효과 없음*
운영 실적	 크게 증가

* 고품질 문서가 소프트웨어 배포 실적에는 영향을 미치지 않는다는 사실은 놀라운 결과였습니다. 이러한 결과가 나타난 건 올해가 두 번째입니다.

3가지 핵심 결과물에 긍정적 영향을 미친다는 사실에는 어떤 배경이 숨겨져 있을까요? 독자의 입장에서는 명확한 문서를 사용하는 것이 도움이 됩니다. 작성 프로세스도 하나의 요인이 될 수 있습니다. 고품질 문서를 작성하기 위해 팀이 가장 먼저 해야 할 일은 프로세스를 결정하는 것입니다. 문서는 조직 전반의 팀들이 무슨 일을 어떻게 해야 하는지에 대해 명시적으로 논의하고 조율할 수 있도록 합니다.

고품질 문서는 사람들이 오가는 상황에서 팀 지식을 위한 저장소 같은 역할도 합니다. 조직 전반에서 긴 시간에 걸쳐 지식이 확장되도록 도와주기 때문입니다.

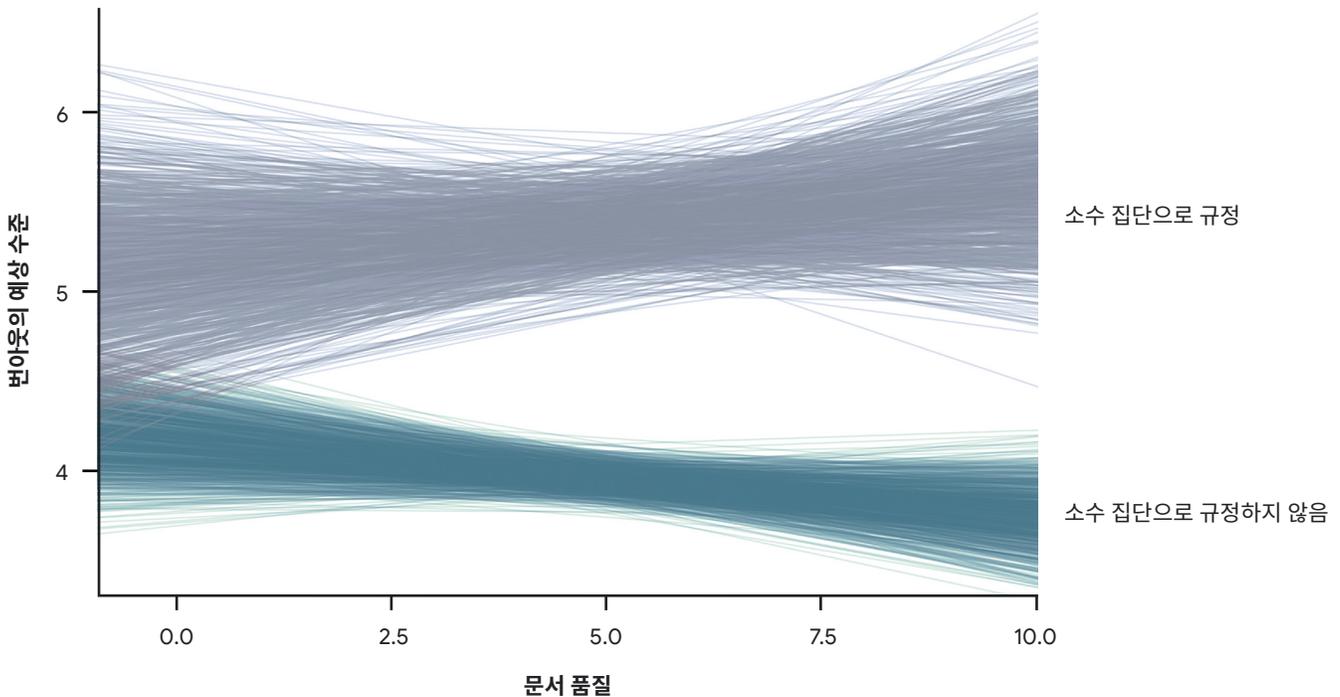
문서가 웰빙을 저하시키는 경우도 있을까요?

소수 집단의 응답자를 살펴봤을 때 예상치 못한 트렌드가 발견되었습니다. 이 집단의 경우 문서의 품질이 번아웃을 증가시키는 것으로 나타났습니다.

성별에 관계없이 이러한 결과가 나타난다는 사실도 놀라웠습니다. 자신을 남성 또는 여성으로 인정하거나 성별을 스스로 규정한 응답자 모두 고품질 문서를 통해 번아웃이 감소한다는 결과가 나타났습니다. 하지만 스스로 소수 집단으로 규정한 직원은 성 정체성에 관계없이 고품질 문서가 있을 때 번아웃 비율이 높아졌습니다.

다음 그래프는 데이터를 기반으로 한 시뮬레이션된 예측을 보여줍니다. 아래쪽에 나온 것처럼 다수 집단의 응답자는 문서 품질이 높아지면 번아웃이 감소하는 것으로 나타났습니다. 하지만 위쪽에 나온 것처럼 스스로 소수 집단으로 규정한 직원의 경우 번아웃이 크게 증가하는 결과가 나타났습니다.

그래프는 각 집단에 대한 1,000개의 시뮬레이션된 선을 보여줍니다. 좀 더 뾰뾰한 곳에 위치한 선의 기울기가 이번 연구의 데이터를 반영할 가능성이 큼니다.



이러한 결과는 문서 품질, 생성적 문화, 팀 안정성에서 모두 비슷하게 나타났습니다. 소수 집단으로 규정한 경우 이러한 속성이 증가하면 번아웃도 증가합니다. 문서의 경우 어떤 이유에서 그런 걸까요?

고품질 문서를 만들고 유지하는 데는 품이 많이 듭니다. 이는 기술적 업무이기 때문에 기술 역량, 팀 생산성, 조직 실적에 상당한 영향을 미칩니다. 또한 중요성과 영향력은 인정받지 못하는 경향이 큼니다. 스스로 소수 집단으로 규정한 직원에게 이러한 작업이 불균형적으로 많이 배정되는 것입니다. 이게 번아웃의 영향을 설명하는 데 도움이 될 것입니다.

문서 사용에 의존하는 게 문제가 될까요? 문서 품질을 높이면 일부 응답자의 경우 지식 공유가 늘어나지 않을까요? 늘어난다고 한다면 이 집단에서 번아웃을 유발하는 다른 요인에 대응하기에 충분하지 않은 걸까요?

고품질 문서를 유도하지만 소수 집단으로 규정한 응답자의 번아웃을 유발하는 데 역할을 하는 완전히 다른 요인이 있을 가능성이 있습니다. 추가적인 연구가 필요합니다.

여러분이 팀에서 어떤 역할을 하는지가 중요합니다. 고품질 문서와 같은 업무 환경적 요소는 팀과 전체 조직에 상당한 이점을 줍니다. 하지만 일부 사람들에게는 부정적인 결과와 연결될 수도 있습니다. 여기에 대해서는 [8장 - 자신을 파악하는 것이 중요한 이유](#)에서 자세히 알아봅니다.

⁴ Accelerate State of DevOps 2021, 22. <https://dora.dev/publications/pdf/state-of-devops-2021.pdf#page=22>

시작하는 데 도움이 되는 리소스

고품질 문서를 촉진하는 관행은 2021년 보고서에서 확인할 수 있습니다.⁴ 올해에는 문서 작업을 배분하는 공식 프로세스를 포함한 업무 분배가 문서 품질을 크게 높인다는 결과가 나타났습니다.

기술적 작성을 위한 수많은 리소스와 학습 프로그램이 마련되어 있습니다. 다음 리소스에서 자세히 알아볼 수 있습니다.

- 기술 커뮤니케이션을 위한 사회(stc.org)
- 엔지니어를 위한 기술적 작성 과정(developers.google.com/tech-writing)
- 문서 작성(writethedocs.org)

신뢰성으로 실적 개선

핵심사항

강력한 신뢰성 관행은 운영 실적, 팀 실적, 조직 실적을 개선할 것으로 예측됩니다. 데이터에 따르면 이러한 관행을 개선함으로써 얻는 효과는 비선형적인 것으로 나타났습니다. 즉, 조직이 더 강력한 역량을 키울수록 실적 개선이 지연되는 것처럼 보일 수 있다는 것입니다. 하지만 시간이 흐를수록 이러한 관행에 집중하면 좋은 결과로 이어질 것입니다.

소개

신뢰성은 IT 운영 부문에서 널리 사용되는 용어입니다. DORA에서는 신뢰성을 가용성, 실적, 정확성 등의 측정치에 대한 규정된 목표를 서비스가 충족하는 정도로 정의합니다. 높은 신뢰성 성과를 달성하기 위한 일반적인 접근 방식은 Google에서 시작되어(<https://sre.google>) 현재 많은 조직에서 자리 잡은 SRE입니다. SRE는 경험적 학습, 여러 직종의 팀 간 협업, 광범위한 자동화 활용, 서비스 수준 목표(SLO)를 포함한 측정 기술 사용을 우선시합니다.

많은 조직에서 신뢰성 관행을 따르지만 SRE라는 용어는 사용하지 않습니다. 대신 프로덕션 엔지니어링, 플랫폼팀, 인프라팀, TechOps 등의 용어를 사용합니다. 이러한 관행의 범위를 최대한 객관적으로 평가하기 위해 DORA의 설문조사 텍스트에서는 중립적이고 설명적인 언어를 사용합니다.

또한 팀이 신뢰성 목표를 달성할 수 있는 정도인 안정성 엔지니어링의 결과에 대한 데이터를 수집합니다. 신뢰성 관행과 신뢰성 성과(운영 실적이라고 부름)는 다른 역량과 함께 DORA의 예측 모델에 반영됩니다.

신뢰성 관행

이번 연구에서는 응답자에게 운영의 3가지 필수 측면을 통해 신뢰성에 대해 생각해 보라고 요청했습니다. 첫째, 종속성에 대한 완화 계획이 있나요? 둘째, 중단 시뮬레이션, 실습형 장애 조치 또는 모의 훈련을 통해 재해 복구 계획을 정기적으로 테스트하나요? 마지막으로 신뢰성 목표를 달성하지 못했을 때 개선 작업을 수행하나요, 아니면 작업 우선순위를 재조정하나요?

DORA는 이러한 측정값이 '위험 수용' 및 '사용자 만족도 측정'과 같은 SRE의 확립된 원칙을 따르는 팀의 정신을 반영한다고 생각합니다. 이러한 팀은 사용자 만족도에 부합하는 합리적인 목표를 세웁니다. 그런 다음 해당 목표를 달성할 수 있는지 확인하는 테스트를 수행하되 문제가 발생하면 계획을 변경합니다. DORA에서는 팀의 평가를 특정 SRE 구현과 연계하지 않고 성공적으로 'SRE를 수행'하는 팀에 대한 대리 평가로 이를 사용합니다.

결과

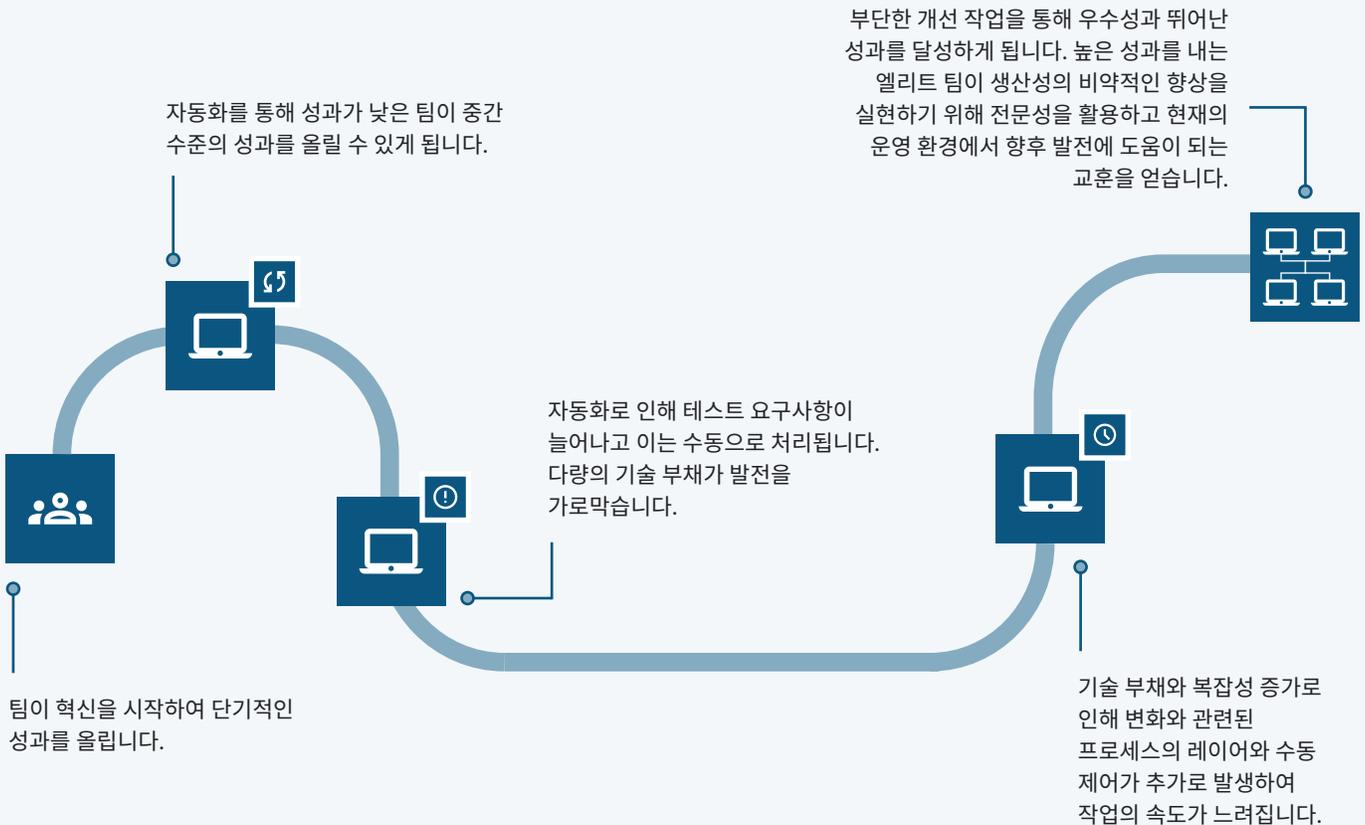
신뢰성 관행의 J 곡선 확인

2018년부터 DORA는 운영 실적 및 자동화와 같은 관행 사이에 비선형적 관계(그림 1)가 있다는 사실을

이론화했습니다. 신뢰성 관행에 대한 조사를 심화하면서 설문조사 데이터에서 이러한 패턴의 증거를 발견했습니다.

2022년에는 이러한 데이터를 직접 측정했습니다. 여러 팀을 대상으로 설문조사를 시행한 결과 신뢰성 관행과 신뢰성 성과 간의 관계가 실제로 이러한 유형의 비선형성 곡선(그림 2)을 따르는 것으로 나타났습니다. 이는 팀들이 수많은 신뢰성 관행을 도입한 후에야 상당한 신뢰성 개선 효과를 얻을 수 있음을 시사합니다.¹ 하지만 이런 방식으로 데이터를 살펴보니 전체 그림은 보지 못하는 것처럼 느껴졌습니다. 2022년 곡선은 SRE가 전문가만을 위한 것이거나 투자할 가치가 없는 것처럼 느껴지게 만들었고 이는 많은 SRE팀의 경험과 상충되는 것이었습니다. 그래서 더 많은 데이터가 필요했습니다.

그림 1: 2018년의 가설 J 곡선



¹ 2022 Accelerate State of DevOps Report. 27~28. <https://dora.dev/research/2022/dora-report/2022-dora-accelerate-state-of-devops-report.pdf#page=27>

그림 2: 2022년 곡선

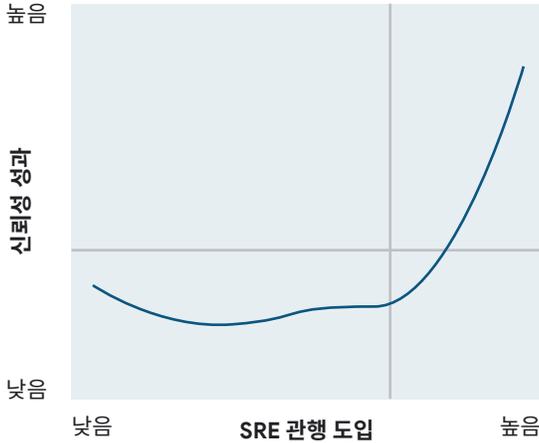
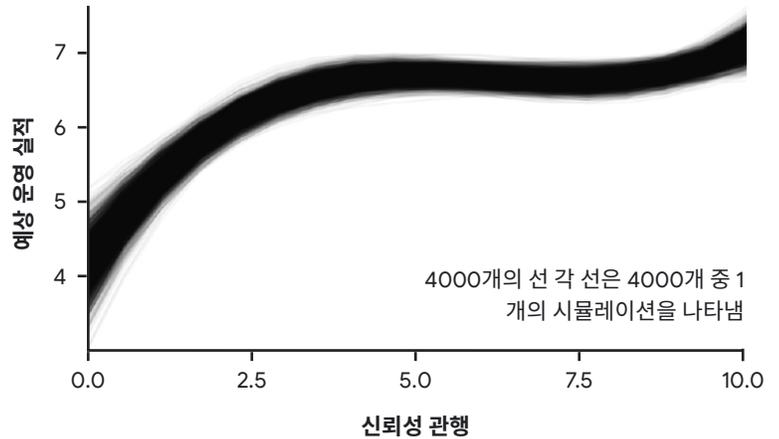


그림 3: 2023년 곡선



2023년에는 더 많은 질문을 할 수 있었습니다. 그리고 이를 통해 우리의 생생한 경험과 더욱 밀접하게 일치하는 곡선을 더 정확히 묘사할 수 있었습니다. 새로운 곡선은 2018년 보고서에 묘사된 가설 J 곡선에 더 가깝습니다(분석을 수행한 방법에 대한 자세한 내용은 [방법론](#) 참조). 이는 신뢰성 관행을 도입한 초기에는 효과가 있었으며 복잡성으로 인해 새로운 문제가 발생하며 소강 상태를 보였고, 마지막으로 운영 실적이 다시 약간 상승했음을 보여줍니다. 이러한 결과는 DORA가 많은 팀에서 발견한 결과를 뒷받침합니다.

이 곡선은 다음과 같은 몇 가지 이유로 인해 중요합니다.

- 기업이 극단적인 수준의 신뢰성을 바라지 않거나 SRE에 상당한 투자를 하도록 기대하지 않더라도 초기 SRE의 도입을 합리화하고 자금을 도달하는 데 도움을 줍니다. 적은 수준의 신뢰성 관행을 도입하더라도 운영 실적이 개선되어 팀 실적과 조직 실적에 추가적인 이점을 제공할 수 있습니다.

- 신뢰성에 많은 투자를 하고자 하는 기업이 소강 상태를 무사히 벗어날 수 있도록 도와줍니다. SRE에 대한 장기적 투자에서 선형적인 긍정적 결과를 기대할 수는 있지만 데이터는 다르게 말합니다. 팀이 이 곡선의 비선형성에 대해 미리 알고 있으며 투자를 해야 할지에 대한 결정을 내리고, 모든 이점을 실현시키기 전에 투자를 포기하지 않도록 계획할 수 있습니다.
- 이와 같은 변화에는 문화 혁신²이 필요할 수 있습니다. DORA는 상향식 변화와 하향식 변화의 조합에서 성공적인 결과가 있었다는 사실을 알아냈습니다. 팀은 신뢰성 관행을 도입하고 즉각적인 보상을 얻을 수 있습니다. 그러면 리더십팀에서 이러한 이점을 다른 팀에 소개하고 강화하며 인센티브를 제공할 수 있습니다. 이러한 인센티브와 구조화된 프로그램은 J 곡선을 염두에 두고 설계할 수 있습니다.

²How to transform(혁신 방법) - <https://dora.dev/devops-capabilities/cultural/devops-culture-transform/>

신뢰성 관행과 웰빙

기존의 운영 관행은 사후 대응적이며 사용자의 만족도보다는 기술 시스템의 상태에 중점을 두는 경우가 많습니다. 사용자의 경험에 영향을 미치지 않는 대기 알림, 반복적인 수동 업무, 실수에 대한 두려움 등은 팀 내 개인의 번아웃과 열악한 웰빙으로 이어집니다.

신뢰성 관행을 활용하는 팀에서는 반대 상황에 펼쳐집니다. 이러한 팀은 신뢰성 관행을 사용하지 않는 팀보다 생산성과 직무 만족도가 높고 번아웃 수준이 낮습니다. 일부 SRE 관행이 이러한 웰빙의 개선을 주도하는 것으로 예상됩니다.

- 반복 업무 감소³
- 비난 없는 사후 분석⁴
- 팀 자율성⁵
- 팀의 저선형 확장⁶

³ 베이어, 벤티 외. 사이트 안정성 엔지니어링: Google의 프로덕션 시스템 운영 방법 (O'Reilly, 2016), 49~54. <https://sre.google/sre-book/eliminating-toil/>

⁴ Ibid, <https://sre.google/sre-book/postmortem-culture/>

⁵ 베이어, 벤티 외. 사이트 안정성 워크북(O'Reilly, 2018), <https://sre.google/workbook/team-lifecycles/>

⁶ 제임스 브룩뱅크 및 스티브 맥기. SRE 엔터프라이즈 로드맵(O'Reilly, 2022), 11. <https://sre.google/resources/practices-and-processes/enterprise-roadmap-to-sre/>

운영 실적

응답자에게 해당 서비스의 운영 실적을 묘사해 달라고 요청하기도 했습니다. 첫 번째로 서비스 신뢰성 불만족에 대해 사용자로부터 직접 듣는 빈도를 물었습니다. 다음으로 서비스가 이용 불가 상태가 되거나 느려지거나 잘못 작동하는 빈도를 물었습니다.

▶ 운영 실적을 통해 팀 실적과 조직 실적을 강화하는 신뢰성 관행

신뢰성 관행을 도입하는 팀은 운영 실적을 개선할 수 있습니다. 조직이 생산 시설을 효율적으로 가동할 수 있다면 이는 다른 결과 수준을 *높인다*는 사실을 발견했습니다. 결과 수치가 높으면 신뢰성 관행이 이를 더 높게 만들 것입니다. 결과 수치가 낮으면 신뢰성 관행은 도움이 되지 않고 수치가 그대로 유지될 것입니다.

신뢰성 시스템을 위해서는 여전히 고객을 위한 올바른 소프트웨어 역량을 갖추고 효율적으로 전달하는 것이 필요합니다. SRE가 외부와 단절된 상태에서 작동하도록 만든 것이 아니기 때문입니다. 신뢰성 목표를 달성하는 것은 SRE팀의 성공을 측정하는 주요 지표이며, 이는 운영 실적에 반영됩니다. 신뢰성 관행을 사용하면 다른 이점도 있을 수 있지만, 데이터에 따르면 가장 중요한 것은 운영 실적에 대한 영향입니다. 또한 운영 실적이 개선되면 서비스 상태 외에 다른 이점도 줍니다. 실제로 신뢰성 관행을 사용하면 실무자의 웰빙이 개선된다는 사실이 증명되었습니다.

▶ 웰빙에 영향을 미치는 운영 실적

일반적인 업계 인식은 신뢰성이 높은 서비스는 근무 시간 외의 긴급 활동이나 응급 유지보수 작업 등으로 인해 서비스 운영자의 웰빙에 부정적인 영향을 미친다는 것입니다. 하지만 운영 실적이 높으면 번아웃이 줄어들고,

생산성이 개선되며, 직무 만족도가 높아진다는 결과가 나타났습니다. 이는 반복 업무를 줄인다는 SRE 원칙과 일맥상통합니다.⁷ 운영의 수동적인 부분을 자동화하면 개인의 만족도를 높일 뿐만 아니라 팀의 지속적인 부담을 줄인다는 결과가 나타납니다.



운영 실적을 강화하는 조직 실적과 팀 실적

운영 실적은 팀 실적 및 조직 실적 모두에 상당히 긍정적 영향을 주는 것으로 나타났습니다. 이는 DevOps 방식을 따르는 사람들에게는 그리 놀랍지 않을 것입니다. 머신을 효율적으로 작동할 수 있으면 팀이 더 많은 성과를 이루고 조직이 성공할 수 있습니다.



소프트웨어 배포 실적을 강화하는 운영 실적

소프트웨어 배포 실적은 팀 실적과 조직 실적을 모두 개선할 수 있습니다. 또한 이 두 가지 실적은 운영 실적에 의해 상당한 개선 효과를 얻습니다. 또한 실적이 우수한 소프트웨어 배포팀은 높은 운영 실적을 달성하지 않고는 매우 높은 팀 실적 및 조직 실적을 달성할 수 없습니다. 이 두 가지 요건을 모두 충족해야 한다는 뜻이죠. 실제로 적당한 수준의 운영 실적을 달성하지 않고 소프트웨어 배포 실적만 개선하는 팀은 조직 성과의 악화로 이어지게 됩니다. 따라서 뛰어난 소프트웨어를 아무리 빨리 만들더라도 잠재고객의 기대치를 충족하는 방식으로 프로덕션을 운영하지 못하면 시장에서 보상을 얻지 못하게 됩니다.

놓치고 있는 것과 다음 단계

이러한 상호작용을 이해하는 데 도움이 될 수 있는 더 많은 측정값이 있을 것입니다. 예를 들어 올해의 공통 질문은 비용 관리가 이러한 역량 및 결과에 어떤 역할을 했느냐입니다. 다른 조직보다 비용에 더 민감한 조직도 있습니다. 그리고 비용은 조직이 계획을 세우고 결정을 내리는 방식에 영향을 미칩니다. 마찬가지로 DORA는 명시적으로 추가하거나 계획하지 않았더라도 신뢰성 관행이 공동작업이 활발히 이루어지는 문화에서 발생할 수 있다는 사실을 이론화합니다.⁸ DORA는 팀이 기존의 IT 운영 관행을 어떻게 진화시키며 그러한 진화가 시스템 신뢰성, 팀 실적, 웰빙에 어떤 영향을 미치는지 더 자세히 알아보려고 합니다.

여러분의 의견을 듣고 싶습니다. [DORA.community](https://dora.community)⁹에서 DORA 담당자 및 다른 실무자들과 의견을 나눠보세요. SRE는 여전히 새로운 분야입니다. 그 영향은 신뢰성 관행을 도입하는 모든 조직 또는 지금까지 SRE를 시행하고 있었던 조직에서 다르게 나타납니다. 이러한 변화는 느립니다. 따라서 DORA는 시간 흐름에 따른 진척도를 보여주기 위해 일관성 있게 측정을 수행하고자 합니다. DORA는 하나의 커뮤니티로서 연구 내용을 공유하여 모두가 함께 성장할 수 있도록 지원할 것입니다.

⁷ 베이어, 벤티 외. 사이트 안정성 엔지니어링: Google의 프로덕션 시스템 운영 방법 (O'Reilly, 2016), 49~54. <https://sre.google/sre-book/eliminating-toil/>

⁸ 제임스 브룩뱅크 및 스티브 맥기. SRE 엔터프라이즈 로드맵(O'Reilly, 2022), 5. <https://sre.google/resources/practices-and-processes/enterprise-roadmap-to-sre/>

⁹ DORA Community, <https://dora.community/>

Google이 SRE를 수행하는 방법

Google에서 SRE는 지속적인 성장을 이루고 자금 지원을 활발히 받는 조직으로 20년 혁신의 역사를 보유하고 있습니다. SRE는 큰 투자 없이도 Google 검색과 광고의 초고속 성장을 가능하게 하는 필수 요건이었습니다. 초기에는 이러한 제품의 실시간 특성으로 높은 신뢰성이 필요했습니다. 검색 또는 광고에서 발생하는 일시적인 오류는 즉각적인 고객 손실로 이어져 다시 시도할 기회조차 주어지지 않았습니다. 동적 광고는 밀리초 단위로 계산되어야 합니다. 느린 검색 속도는 Google 브랜드의 전반적인 가능성을 저하합니다.

동시에 SRE가 새로운 유형의 회사, 즉 구매보다 빌드에 중점을 둔 상향식의 엔지니어링 주도 회사에서 개발되고 있었습니다. Google은 분산형 시스템과 컴파일러 디자인과 같은 학술적 컴퓨터 공학 분야에 강점을 가진 사이트 안정성 엔지니어링(SRE) 직원들을 활용했습니다. 이러한 문화의 DNA는 SRE가 출현하고 발전할 수 있는 풍부한 환경을 제공했습니다. SRE팀은 프로덕션에서 핵심적인 역할을 맡을 뿐만 아니라 시스템을 확장할 수 있는 새로운 방법을 찾는 역할도 맡았습니다.

그런데 SRE는 시간이 흐르며 어떻게 확장했을까요? 한마디로 저선형 방식이었습니다. Google이 2배로 성장할 때마다 SRE 직원도 2배로 늘릴 수는 없었습니다. Gmail, Google 지도, Android, YouTube, Google Cloud와 같은 새로운 제품이 출시되는 상황에서 초기의 Google 제품이 글로벌 수요를 충족하기 위해 확장하는 속도를 고려할 때 고객 증가 속도에 맞춰 SRE를 확장하는 것은 불가능했습니다. 이는 SRE가 아직 시장이 나온지 얼마 되지 않아 리소스가 부족했기 때문이기도 했습니다.

이런 의견도 있었습니다.

- SRE는 사용자 수가 늘어남에 따라 선형적으로 확장할 수 없습니다.
- SRE는 서버 수가 늘어남에 따라 선형적으로 확장할 수 없습니다.
- SRE는 클러스터 수가 늘어남에 따라 선형적으로 확장할 수 없습니다.
- SRE는 서비스 수가 늘어남에 따라 선형적으로 확장할 수 없습니다.

이와 같은 제한된 성장 모델이 유지될 수 밖에 없는 관리 구조입니다. SRE는 협력했던 제품 개발팀과 보조를 맞추는 동시에 서로 협조하고 의견을 맞추었습니다. SRE는 자체 관리 체인을 통해 보고했으며 벤 트레이너-슬로스¹⁰가 최상위에 있었습니다. 이 팀은 제품 개발('dev') 제품 영역 (PA)과 정확히 일치하는 PA로 팀을 분류했습니다. SRE팀은

¹⁰ 벤자민 트레이너 슬로스, 엔지니어링 부문 부사장. 사이트 안정성 엔지니어링: Google의 프로덕션 시스템 운영 방법(O'Reilly, 2016). <https://sre.google/sre-book/part-i-introduction/>, <https://sre.google/sre-book/introduction/#id-2opuzSjFr>

개발팀과 협조하여 SRE를 최대한 활용할 수 있는 방법을 결정했습니다. 개발팀은 새로운 SRE에 직접 자금을 조달할 수 있었습니다. 모든 개발팀이 사용자 대상 제품을 만드는 것은 아니었습니다. 많은 제품이 Bigtable¹¹(구조화된 데이터 스토리지), Borg¹²(컴퓨팅 스케줄링), Colossus¹³(분산형 스토리지)와 같은 인프라팀과 공유되었습니다. 이러한 공유 인프라 서비스를 통해 고객 대상팀은 자체 SRE팀을 보유하지 않고도 확장할 수 있었습니다.

이러한 팀을 자체 조직 내에 둬으로써 팀은 일관성 있는 고용 및 승진 프로세스를 유지할 수 있었습니다. SRE팀은 개발팀 규모의 10분의 1 미만이므로 SRE팀이 자율성을 갖고 SRE 원칙에 어긋나는 방향으로 끌려가지 않도록 하는 것이 중요했습니다.

SRE는 내부 제품 관리자와 함께 내부 제품을 개발했습니다. SRE 제품의 고객은 프로덕션을 실행하는 더 나은 방법을 찾고 있는 다른 팀이었습니다. SRE팀은 사고 대응, 모니터링 및 관측 가능성, 출시 관리, 용량 계획, 문제 해결과 관련된 제품을 개발했습니다.

SRE팀이 성장하는 과정에는 항상 팀 상태와 지속 가능성이 고려되었습니다. 예를 들어 새로운 긴급 대기 SRE팀은 최소 12명으로 구성되어 두 곳에 6명씩 배치되었습니다. 이를 통해 여러 시간대 범위를 커버하고 한 팀에 충분한 인원 수를 확보하여 번아웃 없이 일과 생활의 균형을 유지할 수 있었습니다.

SRE는 지금도 계속해서 적응하고 있습니다. 모든 팀이 위에서 설명한 모델을 정확하게 따르는 것은 아니며 일부 팀은 다른 길을 선택했습니다. Google의 SRE와 같은 대규모 조직 역시 다른 대규모 그룹과 마찬가지로 시장 상황의 영향을 받습니다. 따라서 유연성이 중요합니다. 그러나 무엇보다도 SRE팀은 위험을 수용하고, 서비스 수준을 측정하고, 반복 업무를 줄이고, 자동화를 수용하고, 단순성을 추구하는 등의 원칙을 준수해야 합니다.



¹¹ 창, 파이 외. Bigtable: 구조화된 데이터를 위한 분산형 스토리지 시스템, 운영체제 설계 및 구현(OSDI)에 관한 제7회 USENIX 심포지엄, {USENIX} (2006), pp. 205-218, <https://research.google/pubs/pub27898/>

¹² 베르마, 아비섹 외. Borg를 사용한 Google의 대규모 클러스터 관리, Proceedings of the European Conference on Computer Systems(EuroSys), ACM, 프랑스 보르도 (2015), <https://research.google/pubs/pub43438/>

¹³ 힐데브랜트, 딘 외. Colossus 심층 분석: Google의 확장 가능한 스토리지 시스템 간단히 알아보기. 2021년 4월 19일 - <https://cloud.google.com/blog/products/storage-data-transfer/a-peek-behind-colossus-googles-file-system>

유연한 인프라는 성공의 핵심

핵심사항

유연한 인프라는 팀 실적, 조직 실적, 운영 실적, 소프트웨어 배포 실적의 예측 변수입니다. 클라우드 컴퓨팅은 유연한 인프라를 가능하게 하는 핵심 요소이지만 이러한 이점이 자동으로 실현되는 것은 아닙니다. DORA의 데이터에 따르면 클라우드를 *어떻게* 사용하는지가 관건입니다.

소개

DORA는 수많은 연구를 통해 미국 국립표준기술연구소 (NIST)에서 정의한 다음과 같은 클라우드 컴퓨팅의 기본적인 특징에 중점을 두고 실무자들에게 해당 인프라에 관해 질문했습니다.¹

- 주문형 셀프서비스
- 광범위한 네트워크 액세스
- 리소스 풀링
- 빠른 탄력성
- 측정되는 서비스

DORA의 연구에서는 이러한 5가지 특성이 조직 실적과 소프트웨어 배포 실적을 개선한다는 결과가 지속적으로 도출되었습니다. 올해에는 클라우드 컴퓨팅을 사용하면 좀 더 유연한 인프라를 예상할 수 있는지 확인하고자 했습니다.



¹NIST 특별 간행물 800-145: 'The NIST Definition of Cloud Computing(클라우드 컴퓨팅에 대한 NIST 정의).'

	비율
멀티 클라우드	19.6%
퍼블릭 클라우드	51.5%
하이브리드 클라우드	33.6%
온프레미스	19.8%
로컬	3.2%
기타	2.5%

* 응답자는 복수 선택 가능했습니다.

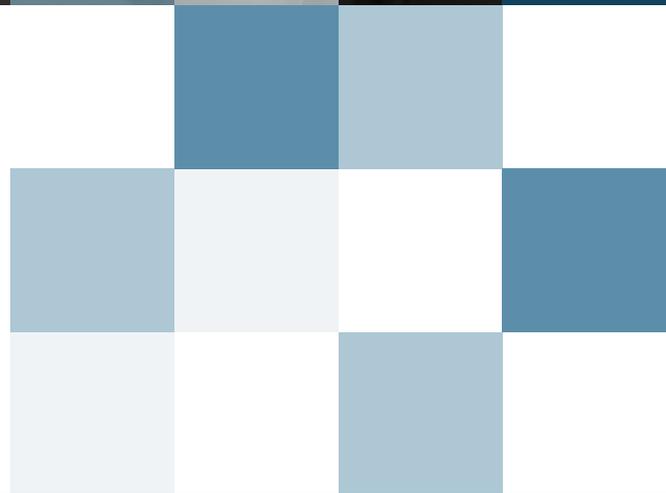
결과

앞서 언급한 연구 결과를 다시 설명해 보겠습니다. 클라우드를 사용한다는 *사실보다 어떻게 사용하는지가* 실적의 더 강력한 예측 변수입니다. 클라우드가 강력한 원동력이 될 수는 있지만 클라우드를 사용한다고 해서 자동으로 이점이 발생하는 것은 아닙니다. 실제로 팀이 유연한 인프라를 사용하지 *않으면* 퍼블릭 클라우드가 소프트웨어 및 운영 실적 *저하*로 이어진다는 강력한 지표를 확인했습니다. 이러한 연구 결과는 단순한 '리프트 앤 시프트' 접근 방식(워크로드를 데이터 센터에서 클라우드로 전환하는 방식)은 유용하지 않으며 오히려 불리할 수 있다는 생각을 더욱 확고하게 합니다.

클라우드 컴퓨팅을 사용하면 번아웃이 크게 감소하고 직무 만족도와 생산성이 크게 증가하는 것으로 나타났습니다.

컴퓨팅 환경

위 표는 응답자가 기본 애플리케이션 또는 서비스를 실행하고 있는 위치를 나타냅니다.



클라우드 유형	조직 실적	팀 실적	소프트웨어 배포 실적	운영 실적
프라이빗	영향 없음	클라우드 컴퓨팅 사용과 결합하여 크게 증가	영향 없음	클라우드 컴퓨팅 사용과 결합하여 크게 증가
퍼블릭	클라우드 컴퓨팅 사용과 결합하여 매우 크게 증가		클라우드 컴퓨팅 사용과 결합하여 크게 감소	클라우드 컴퓨팅 사용과 결합하여 크게 감소
하이브리드			영향 없음	영향 없음
멀티	영향 없음		클라우드 컴퓨팅 사용과 결합하여 크게 감소	클라우드 컴퓨팅 사용과 결합하여 크게 감소

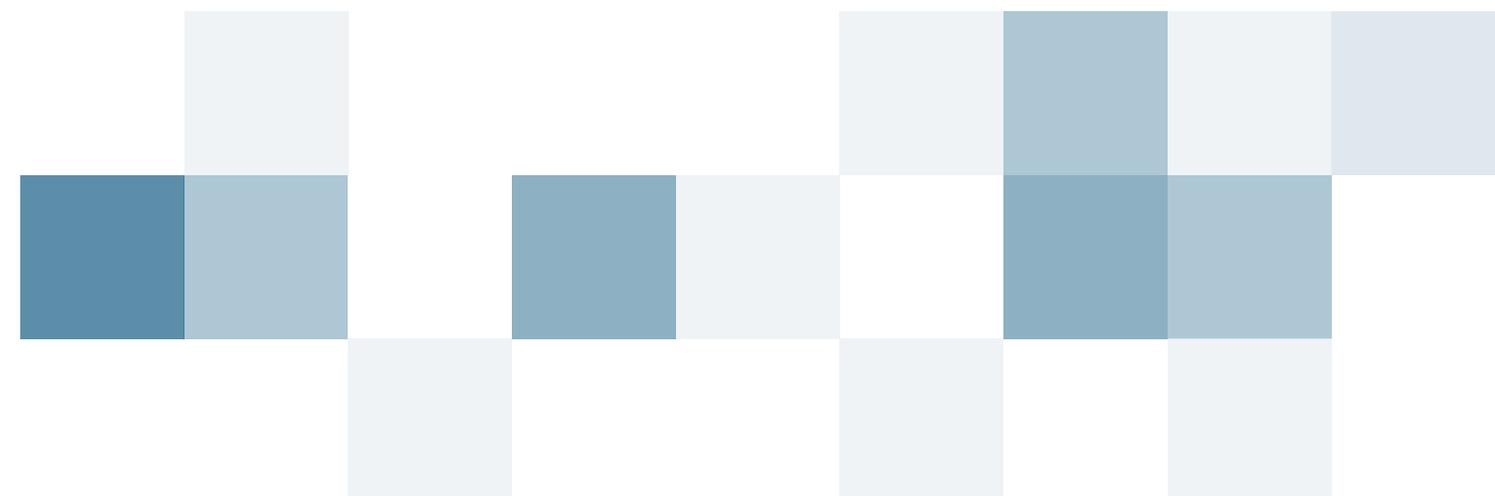
'클라우드를 사용'하기만 해서는 혼합된 결과가 나타납니다

결과 표에 나와 있듯이 '클라우드를 사용'하기만 해서는 소프트웨어 배포 및 운영 실적에 중립적이거나 부정적인 영향을 미칩니다. 이러한 중립적/부정적 영향은 클라우드 여정을 이제 막 시작한 실무자들이 새로운 환경에서 새로운 도구를 사용해 *뭔가* 색다른 방식으로 일해야 하는 상황을 맞닥트렸기 때문일 수 있습니다. 기업들은 종종 새로운 환경의 복잡성과 인지 부담이 가중된 상황에서 자체 데이터 센터에서 했던 것과 동일한 방식으로 클라우드를 사용하곤 합니다.

이러한 새로운 환경에 적응하지 못하면 소프트웨어 배포 또는 운영 실적을 개선하기는 커녕 저하시킬 수 있습니다.

이러한 연구 결과의 한 가지 예외는 프라이빗 클라우드를 사용할 때의 운영 실적입니다.

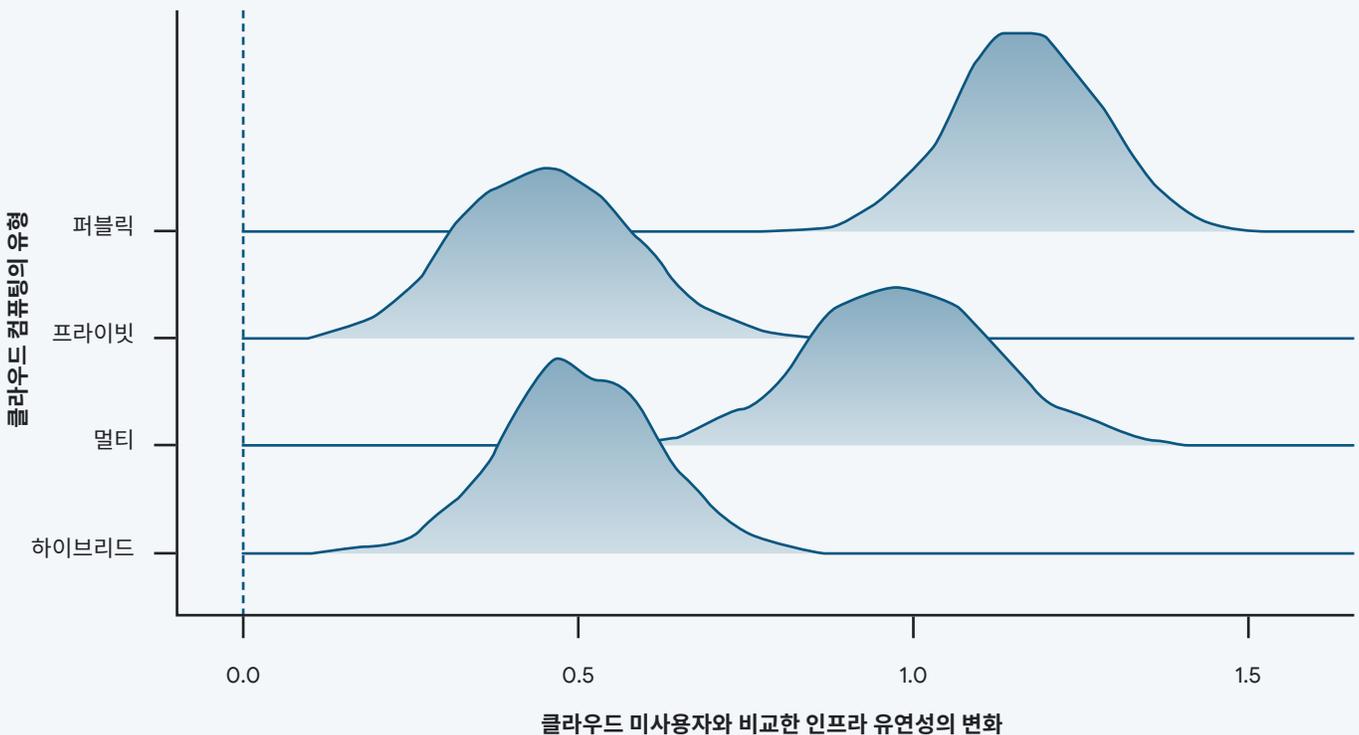
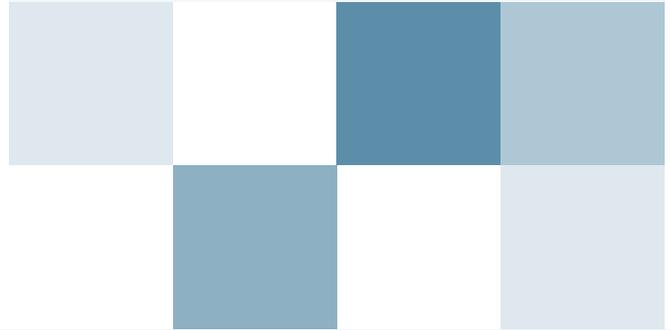
소프트웨어 배포 실적과 운영 실적을 *개선*하는 것은 바로 **유연한 인프라**입니다. 이에 대해서는 곧 살펴보겠습니다.



유연성을 촉진하는 클라우드 인프라

퍼블릭 클라우드를 사용하면 클라우드를 사용하지 않을 때와 비교하여 인프라 유연성이 22% 증가합니다. 멀티 클라우드를 *사용해도* 퍼블릭 클라우드만 사용할 때보다 적지만 증가 효과가 있습니다. 왜 그럴까요? DORA의 데이터에 따르면 클라우드 컴퓨팅을 통해 촉진되는 유연한 인프라는 클라우드 플랫폼을 사용할 때보다 더 큰 영향을 미칩니다. 대부분의 경우 클라우드는 새로운 작업 방식을 의미하며, 적응하는 데 시간이 걸리는 것으로 알려져 있습니다. 각각의 클라우드 플랫폼은 다르기 때문에 클라우드 플랫폼 수를 늘리면 각 플랫폼을 운영하는 데 필요한 인지 부담 역시 늘어난다는 의미입니다.

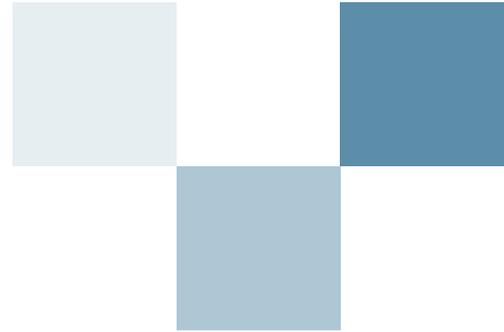
퍼블릭 클라우드를 사용하면 클라우드를 사용하지 않을 때보다 인프라 유연성이 높아집니다.



유연한 인프라가 주요 부문에서 실적을 높일 것으로 기대됩니다

역량	조직 실적	팀 실적	소프트웨어 배포 실적	운영 실적
유연한 인프라	 더 유연한 인프라와 결합하여 크게 증가	 더 유연한 인프라와 결합하여 크게 증가	 더 유연한 인프라와 결합하여 크게 증가	 더 유연한 인프라와 결합하여 증가

유연한 인프라가 조직 실적, 팀 실적, 소프트웨어 배포 실적, 운영 실적의 성공을 이끈다는 사실을 인식하는 것이 중요합니다. 많은 조직이 인프라를 클라우드로 리프트 앤 시프트하는 방식을 선택하고 있습니다. 이는 훌륭한 첫 걸음이 될 수 있지만 여정의 시작에 불과합니다. 워크로드의 일부를 리프트 앤 시프트하기로 결정했다면 다음 단계는 유연한 인프라를 활용할 수 있도록 리팩터링하여 워크로드를 현대화하는 것입니다.



유연한 인프라를 통해 주요 결과에 긍정적 영향을 미치는 클라우드 컴퓨팅

유연한 인프라가 결합된 클라우드 유형	조직 실적	팀 실적	소프트웨어 배포 실적	운영 실적
프라이빗	 유연한 인프라를 통해 부분적 영향을 받음	 유연한 인프라를 통해 부분적 영향을 받음	 유연한 인프라를 통해 부분적 영향을 받음	 유연한 인프라를 통해 부분적 영향을 받음
퍼블릭	 유연한 인프라를 통해 완전한 영향을 받음	 유연한 인프라를 통해 완전한 영향을 받음	 유연한 인프라를 통해 부분적 영향을 받음	 유연한 인프라를 통해 부분적 영향을 받음
하이브리드	 유연한 인프라를 통해 부분적 영향을 받음	 유연한 인프라를 통해 완전한 영향을 받음	 유연한 인프라를 통해 부분적 영향을 받음	 유연한 인프라를 통해 완전한 영향을 받음
멀티	 유연한 인프라를 통해 부분적 영향을 받음	 유연한 인프라를 통해 부분적 영향을 받음	 유연한 인프라를 통해 부분적 영향을 받음	 유연한 인프라를 통해 부분적 영향을 받음



클라우드 컴퓨팅 플랫폼을 유연한 인프라의 특성을 극대화하는 방식으로 사용하면 소프트웨어 배포 및 운영 실적에 긍정적인 영향을 줄 것으로 예상됩니다. 이러한 영향의 차이는 실무자와 리더십팀이 이미 알고 있는 것에 대해 말해줍니다. 워크로드를 단순히 데이터 센터에서 클라우드로 전환하는 것만으로는 성공을 거두지 못한다는 것입니다. 핵심은 클라우드가 지원하는 유연한 인프라의 이점을 활용하는 것입니다.

이점을 극대화하려면 애플리케이션을 빌드, 테스트, 배포,

인프라 유형	성과
유연한 인프라와 결합된 클라우드	
유연성이 결여된 클라우드	

모니터링하는 방법을 다시 생각해봐야 합니다. 주문형 셀프서비스, 광범위한 네트워크 액세스, 리소스 풀링, 빠른 탄력성, 측정 서비스 등 클라우드 컴퓨팅의 5가지 특성을 활용하는 방식을 중점적으로 다시 생각해 보시기 바랍니다.



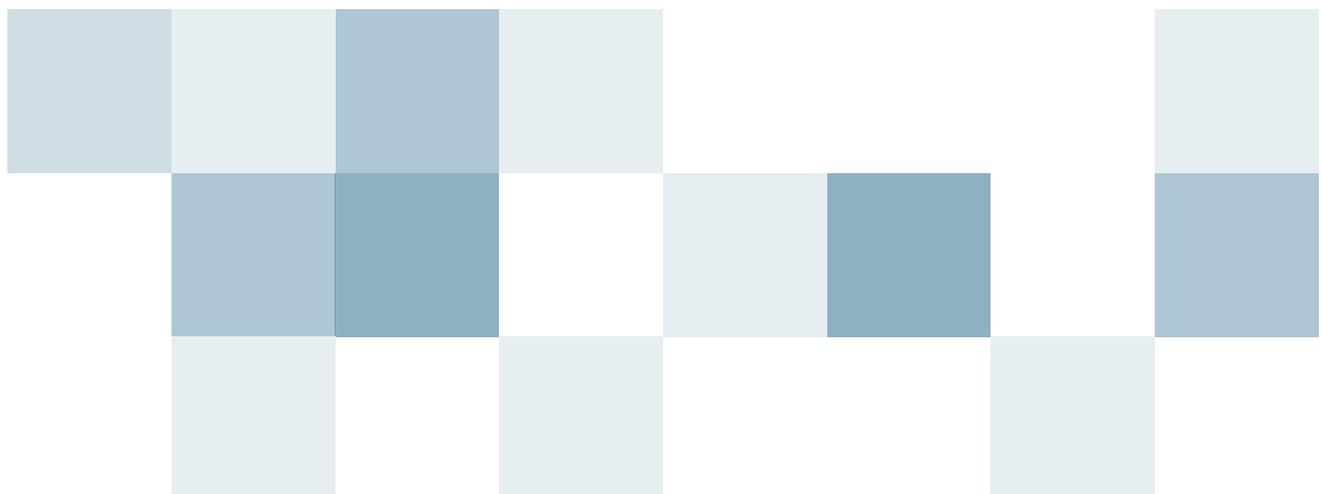
웰빙을 개선하는 클라우드 컴퓨팅

클라우드 유형	번아웃*	직무 만족도	생산성
프라이빗	영향 없음		
퍼블릭	클라우드 컴퓨팅 사용과 결합하여 매우 크게 감소	클라우드 컴퓨팅 사용과 결합하여 크게 증가	클라우드 컴퓨팅 사용과 결합하여 크게 증가
하이브리드	영향 없음		
멀티	영향 없음		

*번아웃의 경우 색 구성표가 반전되는 것을 확인할 수 있습니다. 번아웃 감소는 긍정적 효과이기 때문입니다.

데이터에 따르면 클라우드 컴퓨팅은 직원 웰빙에 큰 도움이 됩니다. 직무 만족도와 생산성을 크게 증가시키며 번아웃에는 중립적이거나 긍정적인 영향을 주는 것으로 나타났습니다. 달리 말하면, 클라우드 컴퓨팅이 추가적인 인지 부담을 주고 새로운 도구와 작업 방식에 적응해야 하는 상황을 가져오지만 클라우드는 웰빙에 부정적인 영향을 미치지 않습니다.

DORA는 실무자로서 이런 현상이 일어나는 몇 가지 이유를 제시할 수 있습니다. 엔지니어는 학습하고 문제를 해결하는 것을 좋아하며, 유연한 컴퓨팅 특성을 갖춘 환경에서 일하는 것을 즐깁니다. 새로운 기술을 배우는 것은 재미있는 일일뿐만 아니라 경력 개발에도 도움이 되는 것입니다. 엔지니어는 조직이 성공했을 때 더 큰 행복감을 느낍니다.



7장

문화에 대한 투자가 없다면 무용지물

핵심사항

문화는 직원 웰빙과 조직 실적의 핵심 동인입니다. 건강한 문화는 번아웃을 줄이고 생산성과 직무 만족도를 높이는 데 도움이 될 수 있습니다. 이는 또한 조직 실적, 소프트웨어 배포 실적, 운영 실적, 팀 실적의 유의미한 개선으로 이어집니다. 건강한 조직 문화는 팀이 성과 개선으로 이어지는 기술 역량을 성공적으로 구현할 수 있도록 도와줍니다.



소개

문화를 한 마디로 정의하기란 어렵습니다. DORA는 직원의 업무 경험에 대해 말해줄 수 있는 지표에 초점을 맞춥니다. 그리고 조직 문화에 대한 웨스트럼의 유형 체계를 사용합니다. 이것이 실적에 대한 강력한 예측 변수로 사용되어 왔기 때문입니다. 또한 올해는 웨스트럼의 정의에 따른 생성적 문화를 갖춘 조직이 계속해서 좋은 실적을 낼 것으로 예상됩니다.

다음 표에는 팀과 조직 문화에 기여할 것으로 생각되는 요소의 목록이 나와 있습니다.

측면	정의
웨스트럼의 조직 문화	조직이 문제와 기회에 대응하는 방법을 평가합니다. 문화에는 생성적 문화, 관료적 문화, 병리적 문화 등 3가지 유형이 있습니다.
조직 안정성	직원을 위한 업무 환경이 얼마나 안정적인지 또는 불안정한지 평가합니다.
고용 보장	직원이 고용 보장에 대해 얼마나 자주 걱정하는지 평가합니다.
유연성	언제 어디서나 어떻게든 사용할 수 있는지 평가합니다.
지식 공유	아이디어와 정보가 조직 전체에 확산되는 방법을 평가합니다. 팀 구성원이 질문에 답을 한 다음 정보를 다른 팀원에게 제공합니다. 답변을 기다릴 필요는 없습니다.
사용자 중심주의	소프트웨어를 개발할 때 최종 사용자에게 중점을 두는 정도와 사용자의 니즈 및 목표에 대한 이해도를 평가합니다. 제품 및 서비스를 더 좋게 만들기 위해 사용자 신호가 사용됩니다.
업무 분배	팀이 힘든 업무를 모든 구성원에게 고르게 분배할 수 있도록 하는 공식 프로세스를 평가합니다.

문화적 요소, 프로세스 역량, 기술 역량 사이의 경계가 항상 명확한 것만은 아닙니다. 문화는 관행으로부터 나오고, 관행은 문화로부터 나온다고 믿습니다. 이에 대해서는 나중에 연구 결과를 논의하며 다시 살펴보겠습니다.

¹<http://bmi.co/1BRGh5q>

연구 결과와 그 의미

건강한 문화는 주요 결과를 개선합니다

전반적으로 건강한 문화는 모든 주요 결과에 긍정적 영향을 미칩니다. 생성적 문화가 조직 실적, 소프트웨어 배포 실적, 운영 실적을 개선한다는 과거 연구 결과가 재현되었습니다. 올해에는 또 다른 실적 지표를 개선한다는 결과가 나타났습니다. 바로 팀 실적입니다.

소프트웨어 개발에서 사용자 중심적인 접근 방식을 추구하면 유의미한 실적 개선으로 이어진다는 결과가 나타났습니다. 주목할 만한 결과입니다. 조직은 사용자를 우선시할 때 막대한 이점을 얻을 수 있습니다. 사용자 피드백은 팀이 프로젝트의 우선순위를 정하고 사용자 니즈를 충족하는 제품 및 서비스를 만드는 데 도움을 줍니다. 이는 더 나은 사용자 경험과 사용자 만족도, 그리고 더 높은 수익으로 이어집니다.

팀간 업무 분배를 측정하여 조직 문화의 건전성에 대해서도 평가했습니다. 여기에서는 공평한 업무 분배가 팀 실적 및 조직 실적에 이점을 제공한다는 결과가 나타났습니다. 하지만 공평한 업무 분배가 낮은 소프트웨어 배포 실적과 연결된다는 결과도 나타났습니다. 이는 업무 분배와 관련된 공식 프로세스가 소프트웨어 배포 파이프라인의 일부인 힘든 업무의 완수를 늦추기 때문일 수 있습니다. 또한 공식 프로세스가 주어진 업무를 맡아야 할 팀 구성원을 선정하는데 영향을 미치기 때문일 수도 있습니다.

조화롭지 않아 보이는 또 다른 결과로는 조직 안정성이 소프트웨어 배포 실적을 미미하지만 유의미하게 감소시킨다는 것입니다. 이는 보다 확실히 자리 잡고 (더

생성적 문화를 보유한 팀은
그렇지 않은 팀보다

30%

더 높은 조직 실적을
달성합니다

큰) 조직이 확실히 자리 잡지 못하고 (더 작은) 신생 조직보다 빠르게 움직여야 한다는 부담이 적기 때문일 수 있습니다. 보다 확실히 자리 잡은 조직에는 소프트웨어 배포 속도와 관련해 유연성을 제공하는 확실히 자리 잡은 제품이 있을 수 있습니다.

정보가 쉽게 전파되면 일도 쉽게 풀리는 법입니다. 더 높은 수준의 정보 공유가 이루어지면 소프트웨어 배포 실적 및 운영 실적이 개선된다는 결과가 나타났습니다. 정보에 쉽게 접근할 수 있고 지식 사일로가 적을 때, 사람들은 업무를 수행하기 위해 필요한 정보를 쫓는 대신 중요한 업무에 시간을 쓸 수 있습니다.

마지막으로 직원들에게 일하는 시간, 위치, 방법을 결정할 수 있는 유연한 업무 환경을 제공하면 모든 실적 지표 전반에 유익한 영향을 미칩니다. 이는 특히 소프트웨어 배포 실적의 경우 더욱 두드러집니다. 조직이 더 엄격한 원격 근무 정책을 펼치더라도 직원들에게 약간의 유연성을 제공하면 이점이 될 수 있습니다.

문화적 측면	팀 실적에 미치는 영향	조직 실적에 미치는 영향	소프트웨어 배포 실적에 미치는 영향	운영 실적에 미치는 영향
웨스트럼의 조직 문화	↑↑ 크게 증가	↑↑ 크게 증가	↑↑ 크게 증가	↑↑ 크게 증가
조직 안정성	↑ 약간 증가	↑↑ 크게 증가	↓ 약간 감소	⊖ 효과 없음
고용 보장	↑ 약간 증가	⊖ 효과 없음	↑ 약간 증가	↑ 약간 증가
유연성	↑ 약간 증가	↑ 약간 증가	↑↑ 크게 증가	↑ 약간 증가
지식 공유	↑ 약간 증가	↓ 약간 감소	↑↑ 크게 증가	↑↑ 크게 증가
사용자 중심주의	↑↑ 크게 증가	↑↑ 크게 증가	↑ 약간 증가	↑↑ 크게 증가
업무 분배	↑↑ 크게 증가	↑↑ 크게 증가	↓↓ 크게 감소	⊖ 효과 없음



건강한 문화는 기술 역량을 개선합니다

DORA의 연구 결과에 따르면 좋은 문화는 기술 역량을 개선하는 데 도움이 됩니다. 문화와 기술 역량은 상호적인 관계입니다. 문화는 관행으로부터 나오고, 관행은 문화로부터 나오기 때문입니다.

문화는 광범위하고 한 마디로 정의하기가 어렵습니다. 기술 역량이 범위가 좁고 잘 정의된 것과는 반대죠. 이는 조직 내에서 개인이 변화를 주도하는 방식에 영향을 미칩니다.

예를 들어 리더는 생성적 문화를 조성하는 인센티브 구조를 만들 수 있습니다. 리더와 개별 기여자는 모두 소프트웨어 개발에 대한 사용자 중심적인 접근 방식을 강조합니다.

개별 기여자는 트렁크 기반 개발, 지속적 통합, 신뢰성 관행, 느슨하게 결합된 아키텍처와 같은 실적을 개선하는 기술 역량의 구현을 주도할 수 있습니다. 이러한 기술 역량을 구현하는 것은 쉽지 않습니다. 또한 이것이 성공하기 위해서는 열린 마음을 가지고 서로 협력하고 의지하며 배울 수 있어야 합니다. 이러한 것이 건강한 문화의 모든 구성요소입니다. 이러한 팀은 조직 내에서 자신의 역량을 활용하면 변화를 주도할 수 있는 권한이 더 강해진다고 느끼는 사람들의 본보기가 될 수 있습니다.

조직 문화에 대한 장기적이고 유의미한 변화는 변화를 일으키기 위한 상향식 노력과 하향식 노력이 동시에 작용하여 발생합니다.

문화적 측면	트렁크 기반 개발에 미치는 영향	신뢰성 관행에 미치는 영향	지속적 통합에 미치는 영향	지속적 배포에 미치는 영향	느슨하게 결합된 아키텍처에 미치는 영향
웨스트럼의 조직 문화	↑↑ 크게 증가	↑↑ 크게 증가	↑↑ 크게 증가	↑↑ 크게 증가	↑↑ 크게 증가
조직 안정성	↑ 약간 증가	↑↑ 크게 증가	⊖ 효과 없음	⊖ 효과 없음	⊖ 효과 없음
고용 보장	↓ 약간 감소	↓ 약간 감소	⊖ 효과 없음	⊖ 효과 없음	⊖ 효과 없음
유연성	⊖ 효과 없음	↓ 약간 감소	↑↑ 크게 증가	↑ 약간 증가	↑↑ 크게 증가
지식 공유	⊖ 효과 없음	⊖ 효과 없음	⊖ 효과 없음	↑ 약간 증가	↑ 약간 증가
사용자 중심주의	↑↑ 크게 증가	↑↑ 크게 증가	↑↑ 크게 증가	↑↑ 크게 증가	↑↑ 크게 증가
업무 분배	↑↑ 크게 증가	↑↑ 크게 증가	↑↑ 크게 증가	↑↑ 크게 증가	↑↑ 크게 증가

건강한 문화는 직원 웰빙을 개선합니다

건강한 문화는 번아웃을 줄이고 직무 만족도와 생산성을 높이는 방식으로 직원 웰빙 수준을 높입니다. 직원 웰빙은 선택이 아닙니다. 조직 전반의 건강과 성공을 위한 토대입니다.

조직이 더 나은 문화를 위해 투자하지 않으면 어떻게 될까요? 번아웃 가능성이 높아지고 직무 만족도가 낮아지게 됩니다.

직원들은 냉소주의에 빠지고 생산성이 저하됩니다. 신체적 건강과 정신적 건강에도 부정적인 영향을 미칩니다.^{2,3} 번아웃이 지속되어 휴식을 취한 후에도 회복하지 못하게 됩니다. 번아웃은 이직률도 높여 직원들이 더 건강한 업무 환경을 찾아 떠나게 됩니다.⁴ 따라서 번아웃을 완화하기 위해서는 그 원인을 해결하는 조직 차원의 변화가 필요합니다.

문화적 측면	번아웃에 미치는 영향*	직무 만족도에 미치는 영향	생산성에 미치는 영향
웨스트럼의 조직 문화	 크게 감소	 크게 증가	 크게 증가
조직 안정성	 크게 감소	 크게 증가	 약간 증가
고용 보장	 크게 감소	 약간 증가	 약간 증가
유연성	 약간 감소	 약간 증가	 약간 증가
지식 공유	 크게 감소	 약간 증가	 약간 증가
사용자 중심주의	 약간 감소	 크게 증가	 크게 증가
업무 분배	 효과 없음	 약간 증가	 약간 증가

*번아웃의 경우 색 구성표가 반전되는 것을 확인할 수 있습니다. 번아웃 감소는 긍정적 효과이기 때문입니다.

² 아담 베이즈, 가브리엘라 타벨라, 고든 파커(2021) 'The biology of burnout: Causes and consequences(번아웃의 생물학: 원인과 결과)', The World Journal of Biological Psychiatry, 22:9, 686~698. DOI: 10.1080/15622975.2021.1907713. <https://doi.org/10.1080/15622975.2021.1907713>

³ 마슬라흐 C, 레이터 MP. 'Understanding the burnout experience: recent research and its implications for psychiatry(번아웃 경험의 이해: 최근 연구 및 번아웃이 정신의학에 미치는 영향)'. World Psychiatry, 2016년 6월, 15(2), 103~11. DOI: 10.1002/wps.20311. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4911781/> PMID: 27265691; PMCID: PMC4911781.

⁴ L.A. 켈리 외. 'Impact of nurse burnout on organizational and position turnover(간호사의 번아웃이 조직 및 직무 이직에 미치는 영향)'. Nursing Outlook, 2021년 1월, 96~102. 2021년 1월~2월, 96~102. DOI: doi.org/10.1016/j.outlook.2020.06.008

자신을 파악하는 것이 중요한 이유

핵심사항

자신을 파악하는 것이 중요합니다. 특정 집단의 응답자들에게서는 번아웃 가능성이 높아진다거나 생산성이 줄어드는 등 다른 응답자들과 다른 결과가 나타났습니다. 또한 이러한 일부 부정적인 결과를 완화하기 위해 실행할 수 있는 구체적인 방안도 확인했습니다.

소개

2022년의 분석에서는 작업 설정 방식이 전부가 아닌 일부 직원의 웰빙에 도움이 될 수 있다는 일반적인 현상이 주로 나타났습니다.

2022년 DORA의 연구에서는 소수 집단의 직원이 더 높은 수준의 번아웃을 보고한다는 결과가 나타났습니다.¹ 이번 장에서는 이러한 연구 결과가 반복되고 있다는 점을 살펴보고 소수 집단이 번아웃을 경험할 가능성이 높은 이유와 이를 방지할 수 있는 요인을 알아봅니다.

또한 여러 산업에 팽배한 불안정성은 신규 채용에 관한 의구심으로 이어졌습니다. 조직들은 신입 직원을 생산성 있는 인재로 키우는 데 많은 시간이 걸린다는 점을 우려하고 있습니다. 이들은 신입 직원이 더 빠르게 발전할 수 있는 방법을 찾고 있습니다. DORA 역시 그 방법을 찾아볼 것입니다.

연구 결과와 그 의미

번아웃에 빠지기 더 쉬운 직원

작년 DORA의 연구에서는 자신을 여성으로 식별하거나 자신의 성별을 스스로 규정한 응답자와 자신을 소수 집단에 속한다고 여기는 응답자가 자신을 남성으로 식별하거나 소수 집단에 속하지 않는다고 여기는 응답자보다 번아웃을 더 많이 경험하는 것으로 나타났습니다. 이러한 결과는 소수 집단에 속한 직원이 대표성을 띤 동료보다 더 많은 번아웃²을 경험하고 업무 관련 스트레스³를 받기 쉽다고 시사한 과거 연구의 주요 결과와 일치합니다.

이러한 이유로 올해의 연구 데이터에서도 번아웃과 관련해서 차이를 보이는지 확인하고자 했고 실제로 차이가 있었습니다. 자신을 여성으로 식별하거나 자신의 성별을 스스로 규정한 응답자는 자신을 남성으로 식별한 응답자보다 번아웃을 6% 더 많이 경험하는 것으로 나타났습니다. 자신을 소수 집단에 속한다고 여기는 응답자는 자신을 소수 집단에 속하지 않는다고 여기는 응답자보다 번아웃을 24% 더 많이 경험하는 것으로 나타났습니다.

¹ 2022 Accelerate State of DevOps Report. <https://dora.dev/research/2022/dora-report/>

² 시갈리트 로넨, 아얄라 말라치 파인스, 'Gender Differences in Engineers' Burnout(엔지니어 간 업무 번아웃의 성별 차이), Equal Opportunities International, 2008년 11월 7일, <https://www.emerald.com/insight/content/doi/10.1108/02610150810916749/full/html>

³ 달레산드로 C, 로벨 A, 테너 OC, 'Race, Marginalization, and Perceptions of Stress Among Workers Worldwide Post-2020(2020년 이후 전 세계 근로자의 인종, 소외, 스트레스에 대한 인식)', Sociological Inquiry, 2023년 8월 3일, <https://onlinelibrary.wiley.com/doi/pdf/10.1111/soin.12505>

더 많은 번아웃을 유발하는 것으로 나타난 업무 유형

고품질 문서나 안정적인 팀 같이 중립적이거나 긍정적으로 보일 수 있는 직장 요소가 모든 개인의 번아웃을 줄이지는 않는다는 결과가 나타났습니다(4장 - 모든 업무의 기초가 되는 문서 참조). 이러한 결과의 이유가 조직에는 이점을 주지만 일부 개인에게는 번아웃을 유발할 수 있는 작업 때문인지 궁금했습니다.⁴

응답자의 번아웃 경험을 이해하기 위해 이들이 수행하는 업무에 대해 질문을 했습니다. 그리고 다음과 같은 두 가지 방법으로 측정했습니다.

- 코딩, 회의 또는 팀원 지원과 같은 특정 업무
- 계획되지 않은 업무, 업무에 대한 가시성 또는 대량 반복 업무와 같은 작업 특성

작업의 특성은 중요합니다. 같은 업무라도 직원마다 또는 시간에 따라 다르게 느낄 수 있기 때문입니다. 예를 들어 코드 검토 작업 중에서도 일부는 계획되지 않은 반복 업무일 수 있고, 다른 일부는 팀 내에 미리 공유되어 리더십 및 기술 전문성을 보여줄 수 있는 업무일 수 있습니다.⁵

소수 집단에 속하는 것으로 식별되는 응답자는 그렇지 않은 응답자보다 24% 더 많은 반복 업무를 수행하는 것으로 나타났습니다. 자신을 여성으로 식별하거나 자신의 성별을 스스로 규정한 응답자는 자신을 남성으로 식별한 응답자보다 40% 더 많은 반복 업무를 수행하는 것으로 나타났습니다. 이러한 두 집단의 응답자는 또한 계획되지 않은 업무를 더 많이 수행하는 것으로 나타났으며, 이러한 업무는 동료들에게 공유되지 않았거나 자신의 전문성에 맞지 않는 것이었습니다. 이 연구 결과는 이러한 집단에서 보고한 번아웃을 부분적으로 설명합니다.

⁴ 린다 밥콕, 브렌다 페이스, 리제 베스터룬드, 로리 바인가르트. *The No Club*(뉴욕: 사이먼 및 슈스터, 2022), 17.

⁵ 머피-힐, E. 외. 'Systemic Gender Inequities in Who Reviews Code(코드 검토자 사이의 체계적 성 불평등)', *Computer Supported Cooperative Work* (2023) (예정), <https://research.google/pubs/pub52204>

⁶ 밥콕 외, *The No Club*, 17.

승진 없는 잡무

밥콕 등은 저서 *The No Club*에서 다음과 같은 특정 유형의 업무를 설명했습니다. '승진 없는 잡무는 조직에는 중요하지만 개인의 경력 발전에는 도움이 되지 않습니다.'⁶

여기서 *발전*이란 단어는 광범위하게 사용되는데요, 다른 업무에서는 증가된 보상 또는 개선된 시장성을 의미할 수 있습니다.

주로 여성들이 이러한 유형의 업무를 더 많이 처리한다는 증거가 있습니다. 밥콕을 비롯한 저자는 이러한 불평등한 업무 분배의 이유를 설명합니다. 예를 들어 여성들은 이런 유형의 업무를 하도록 요구받을 가능성이 더 크고 거절할 때의 사회적 비용 때문에 승낙할 가능성이 크다는 것입니다.

또한 이러한 불평등한 업무 분배의 결과로 여성들이 다음과 같은 상황을 겪을 수 있다고 설명합니다.

- 경력 또는 수입에 부정적인 영향을 미칩니다.
- 자신의 경력과 관련된 적절한 양의 업무를 처리하는데 더 많은 시간을 소모합니다.

업무 분배를 위한 공식 프로세스는 일부 응답자의 번아웃을 줄여줍니다

응답자에게 업무를 고르게 분배하기 위한 공식 프로세스가 있는지 물었습니다. DORA는 이를 **업무 분배**라고 부르며, 이 프로세스가 일부 응답자가 경험한 번아웃을 완화시킬 것이라고 기대합니다.

업무 분배가 자신을 남성으로 식별한 응답자와 자신을 여성으로 식별하거나 자신의 성별을 스스로 규정한 응답자의 번아웃을 줄여준다는 결과를 확인했습니다. 업무 분배 수준이 높은 경우 성별 간 번아웃 차이가 사라졌습니다.

하지만 소수 집단에 속하는 것으로 식별되는 응답자의 번아웃 수준에 업무 분배가 영향을 미치지 못한다는 사실은 놀라웠습니다. 이 연구 결과는 또 다른 의문을 가져왔습니다. 업무를 고르게 분배하기 위한 공식 프로세스가 여전히 불평등한 업무 분배를 초래하는가? '동등한 작업'라는 것은 중단 또는 가시성과 같은 업무의 특성을 고려하는가? 작업과 별개로 번아웃을 초래하는 다른 요소를 완화하려면 어떻게 해야 하는가? 이 집단에는 그러한 요소가 더 중요한 것인가?



소속 문화 조성

작성자: 제프리 와이너 박사,

주치의 심리학자, 보스턴 아동 병원 조교수, 하버드 의학전문대학원

주요 연구 결과와 배경

이 보고서의 주요 연구 결과에 따르면 소수 집단에 속하는 것으로 식별되는 직원은 동료들보다 번아웃을 훨씬 더 많이 경험합니다. 보고서에서는 이에 대한 몇 가지 가능성 있는 이유를 살펴봤습니다. 이번 섹션에서는 이러한 연구 결과를 소속감 및 연대 의식이 강한 조직 전략에 대한 광범위한 연구와 연결하여 살펴봅니다.

자신을 소수 집단이라고 인식하는 것은 확립된 심리학 이론인 '불확실한 소속감'⁷에 대한 취약성이 드러나는 것입니다 (제프리 코헨 박사의 'Understanding and Overcoming Belonging Uncertainty(소속 불확실성의 이해와 극복)' 참조).⁸ 이러한 불확실성(예: “내가 여기에 소속된 것이 맞나?”, “나 같은 사람도 여기서 성공할 수 있을까?”)은 그러한 기대에 대한 사람들의 지속적인 경험과 해석을 통해 강화되거나 재정의됩니다. 불확실한 소속감에 대한 이러한 확립된 프로세스는 소수 집단에 속하는 것으로 식별되는 사람이 더 많은 번아웃을 경험한다는 이 보고서의 연구 결과를 맥락에 맞게 이해하는 데 도움이 될 수 있습니다.

그렇다면 조직은 무엇을 할 수 있을까요?

다양성, 포용성, 형평성, 소속감은 모두 다른 것을 의미하며 이를 형성하기 위해서는 서로 다르지만 상호 연결되어 있고 지속적인 전략이 필요합니다. 소속감을 형성하기 위해서는 진실하고 지속적인 노력이 필요합니다.

구성원이 조직 내에서 어려움을 겪고 있는 경우 이런 첫 질문은 곤란합니다. “이 사람에게 무슨 문제가 있는 것일까?” 첫 질문은 이래야 합니다. “이런 기분을 느끼는 이유가 무엇이며 우리 조직의 어떤 구조적 요인 때문에 이런 기분이 들게 하는 걸까?(어떤 요인 때문에 이런 기분이 들게 하고, 심지어 그 기분을 더 악화시키는 걸까?)”

문제를 파악했다면 조직 차원에서 변화를 주도하고 개인별로 지원을 제공하는 두 가지 접근 방식을 취해야 합니다. 개인이 조직을 관리하는 시스템에 영향을 주도록 지원하면 변화가 시스템에 녹아들어 개별 행위자보다 오래 지속될 것입니다. 이러한 시스템과 지속 가능성 사고방식을 갖추면 변화가 기관에 녹아들어 개별 행위자보다 오래 지속될 것입니다. 이렇게 생성적 문화의 질을 높이는 것이 조직에서 소속감을 형성하기 위해 할 수 있는 일입니다. 여기서는 *지속적인* 노력이 핵심입니다. 소속감은 지속적인 경험과 행동을 통해 형성되며 여기에는 끝이 없습니다. 그렇기 때문에 소속감은 직장의 건강한 문화와 생산성을 위한 기본 요소인 것입니다.

이 부문에서 조직을 지원하기 위한 수많은 도구가 있습니다. 예를 들어 외로움을 주제로 한 2023 Surgeon General의 보고서에서는 사회적 연결과 소속감이 외로움과 번아웃을 해소할 수 있는 핵심 수단이라는 점을 확인했습니다.⁹

⁷ 월튼 GM, 코헨 GL, 'A Brief Social-Belonging Intervention Improves Academic and Health Outcomes of Minority Students(간단한 사회적 소속 개입으로 소외계층 학생들의 학업 및 건강 개선)', Science(뉴욕주 뉴욕시), 2023년 9월 20일, <https://pubmed.ncbi.nlm.nih.gov/21415354/>

⁸ <https://behavioralscientist.org/understanding-and-overcoming-belonging-uncertainty/>

⁹ 미국 보건부 차관실(OASH), 'New Surgeon General Advisory Raises Alarm about the Devastating Impact of the Epidemic of Loneliness and Isolation in the United States(신임 보건총감 자문, 미국을 뒤흔치는 외로움과 소외감이 미칠 파괴적인 영향에 대해 경고)', HHS.gov, 2023년 5월 3일, <https://www.hhs.gov/about/news/2023/05/03/new-surgeon-general-advisory-raises-alarm-about-devastating-impact-epidemic-loneliness-isolation-united-states.html>

생산성 관련 문제를 겪는 신입 사원

팀에 합류한지 1년 미만의 신입 사원은 1년 이상의 경력이 있는 동료보다 생산성이 8% 낮은 것으로 나타났습니다. 이는 충분히 예상 가능한 결과입니다. 새 팀에 적응하기란 쉽지 않은 일이고 해당 직무에 경험이 있다 할지라도 순조로운 출발을 위해 필요한 대량의 팀별 지식을 습득하는 것도 어려울 수 있습니다. 또한 팀에 소속된다는 것은 단순히 기술과 지식의 문제가 아닙니다. 사회적 요소 역시 생산성을 위해 필수적입니다. 소속감, 팀에 기여한다는 생각, 심리적 안전 같은 요소는 발전하는 데 시간이 걸립니다.

그렇다면 신입 사원이 빠르게 적응하도록 도울 수 있는 방법이 있을까요?

DORA는 조직이 신입 사원을 3가지 방법으로 도울 수 있다는 가설을 세웠습니다.

- 고품질 문서 제공
- 다른 연구에서 경험 많은 직원보다 경험이 부족한 직원에게 더 유용한 것으로 나타난 인공지능을 워크플로에 통합
- 온보딩 단계에서 특히 유용한 것으로 나타난 대면 공동작업



올해에는 고품질 문서가 생산성에 상당한 영향을 미치며 [\(4장 - 모든 업무의 기초가 되는 문서\)](#) 시는 개인 생산성에 미미한 이점을 제공한다는 [\(3장 - 기술 역량으로 실적 개선 기대\)](#) 결과가 나타났습니다. 이것은 신입 사원에게까지 영향을 줄 가능성이 있습니다. 데이터를 살펴보면 이러한 관행이 신입 사원에게 도움이 되기는 하지만, 다른 직원에 비해 신입 사원에게 더 많은 도움을 주거나 덜 주는 것은 아니라는 사실을 알 수 있습니다. 다시 말해 신입 사원이라고 해서 이러한 관행으로부터 특별한 이점을 얻는 것은 아닙니다.

신입 사원을 비롯한 모든 직원을 돕고 싶다면 고품질 문서는 생산성에 미치는 명확하고 상당한 영향을 고려할 때 가장 먼저 준비해야 할 사항입니다. 잘 쓰인 문서를 갖춘 팀의 신입 사원(표준 편차가 평균보다 1 높음)은 잘 쓰이지 못한 문서를 갖춘 팀의 신입 사원(표준 편차가 평균보다 1 낮음)보다 생산성이 130% 높다는 점을 주목할 필요가 있습니다.

논쟁을 격화시키고 싶지는 않으니 사무실 복귀에 관한 내용은 간단히 다루겠습니다. [7장 - 문화에 대한 투자가 없다면 무용지물](#)에서 유연성의 중요성에 대해 논의했습니다. 또한 DORA의 데이터는 실험적이지 않으며 결과가 편향될 수 있는 요인을 통제하려고 노력했지만 업무 방식의 이점은 복잡하고 사회학적으로 많은 논의가 필요한 문제인 만큼, (주제에 대한 연구나 분석을 읽을 때 염두에 둘 만한 가치가 있는) 명확한 결론을 내리기가 어렵습니다. DORA의 데이터에서 명확하게 나타난 것은 유연성이 생산성에 긍정적인 영향을 미친다는 점입니다. 명확하지 않은 것은 일하는 장소에 따른 영향의 변화입니다.

신입 사원의 경우에도 같습니다. 대면 공동작업이 신입 사원에게 특별히 유익하다는 증거는 없습니다. 생산성에만 최적화하려 한다면 일하는 방법, 장소, 시간의 관점에서 신입 사원에게 유연성을 제공하는 것이 사무실에 강제로 있게 하는 것보다 더 확실한 방법인 것 같습니다. 물론 조직은 생산성에만 최적화하지 않으며, 그렇게 해서도 안 됩니다. 몇 줄의 코드를 비롯한 결과물이 아닌 가치로 이어지며 번아웃이나 반복 업무를 초래하지 않는 작업의 측면에서 생산성을 고려해야 합니다.



최종 의견

올해의 연구에 참여하고 이 보고서를 읽어주셔서 감사합니다. DORA는 항상 팀이 일하는 방식과 달성할 수 있는 결과 사이의 연관성을 밝히기 위한 더 나은 방법을 찾고 있습니다.

다년간의 연구 프로그램을 통해 얻은 가장 중요한 내용은 지속적인 개선에 대한 사고방식과 관행을 도입하는 팀이 최상의 결과를 얻을 수 있다는 점입니다.

지금까지 탐구한 역량은 원하는 결과를 달성하기 위한 조절 장치로 사용할 수 있습니다. 그중에는 개인이 접근할 수 있는 조절 장치도 있지만 전체 조직 단위에서 조직화된 노력을 통해서만 접근할 수 있는 조절 장치도 있습니다. 따라서 조직을 위해 어떤 조절 장치를 조정해야 하는지 파악한 다음 그러한 조정 조치에 투자해야 합니다.

개선 작업에는 끝이 없지만 개인, 팀, 조직의 장기적인 성공을 도울 수는 있습니다. 리더와 실무자가 이러한 개선 작업을 주도하기 위한 책임을 공유해야 하는 이유입니다.

연구 결과를 실무에 적용하는 방법

조직, 팀, 고객에게 제공하는 서비스의 맥락에서 연구 결과를 살펴보세요.

DORA Community(<https://dora.community>)에 참여하여 지속적인 개선의 여정에서 다른 여행자와 지식과 경험, 영감을 공유해 보세요.

감사의 말

이 보고서는 매년 전 세계 열렬한 기여자들의 도움을 받고 있습니다. 설문조사 질문 설계부터 현지화, 분석, 내용 작성, 편집, 조판까지 보고서 제작 과정 전반에는 이러한 수많은 노력을 실현하는 데 도움을 준 동료들의 손길이 닿아 있습니다. 의견과 조언을 제공하고 동지애를 보여주신 모든 분들께 감사 말씀을 드립니다.

기여자

핵심 팀

제임스 브룩뱅크
킴 카스티요
데렉 드벨리스
네이슨 하비
미셸 어바인
아만다 루이스
에릭 맥스웰
스티브 맥기
데이브 스탠크
케빈 스토어러
다니엘라 비알바
브레나 워싱턴

편집자

맨디 그로버
제이 하우저
스탠 맥켄지
안나 임스 미카워
마이크 포프
타비사 스미스
올린다 터너

설문조사 현지화

다니엘 아마데이
쿠마 아라카와
윌리엄 바틀릿
안토니오 구즈만
쇼고 하마다
유키 이와나리

빈센트 조바르드

구스타보 라파
마우리시오 멜렌데즈
제레미 파토니어
미구엘 레예스
페드로 소사
로랑 타르디프
키미 우
비니시우스 사비에르
요시 야마구치

업계 전문가 및 전문가

자레드 바티
리사 크리스핀
롭 에드워즈
데이브 팔리

스티브 펜턴

니콜 폴스그렌 박사
애런 길리스
데날리 롬마
에머슨 머피-힐
하리니 샴패스
로빈 사비나르
더스틴 스미스
제스 츠메리스
로리에 웨인가르트 방사
베차엘레 (서울) 윌리엄슨
제프리 와이너 박사

스폰서



저자



데렉 드벨리스

데렉은 Google의 정량적 사용자 경험 연구원이자 DORA의 리드 조사 담당자입니다. 데렉은 설문조사 연구, 로그 분석 업무에 주력하고 있으며, 제품 또는 기능을 보여주는 개념의 측정 방법을 파악하여 가변 자본 가치를 제공하는 일을 주로 합니다. 인간 대 AI 상호작용, 코로나19 발병이 금연에 미치는 영향, NLP 오류에 대한 설계, 개인 정보 보호 논의에서 UX의 역할, 팀 문화, AI와 직원 웰빙 및 생산성의 관계를 발표했습니다. 현재 과외 연구를 통해 신념과 권한의 전파를 시뮬레이션하기 위한 방법을 탐구하고 있습니다.



아만다 루이스

아만다 루이스는 DORA.community의 개발 리드이며 Google Cloud의 DORA Advocacy팀에서는 개발자 관계 엔지니어를 맡고 있습니다. 아만다는 개발자, 운영자, 제품 관리자, 프로젝트 관리, 리더십 전반에서 소통하는 역할을 하며 경력을 쌓았습니다. 또한 전자상거래 플랫폼, 콘텐츠 관리 시스템, 관측 가능성 도구를 개발하고 개발자를 지원하는 팀에서 일한 경험이 있습니다. 이러한 소통 및 대화 능력은 고객 만족과 더 나은 비즈니스 성과로 이어졌습니다. 아만다는 자신의 경험과 공감 능력을 업무에 활용해 팀이 소프트웨어 배포 및 신뢰성 관행을 이해하고 구현할 수 있도록 돕고 있습니다.



다니엘라 비알바

다니엘라 비알바는 Google의 사용자 경험 연구원입니다. 설문조사 연구를 통해 개발자의 만족도와 생산성을 높일 수 있는 요인을 파악합니다. Google에 합류하기 전에는 명상 훈련의 이점, 대학생의 경험에 영향을 미치는 심리 사회적 요인을 연구했습니다. 다니엘라는 플로리다 인터내셔널 대학교에서 실험 심리학 박사 학위를 받았습니다.



데이브 팔리

데이브 팔리는 Continuous Delivery Ltd.의 상무이사 겸 설립자이자 *Modern Software Engineering*의 저자, 베스트셀러인 *Continuous Delivery*의 공동 저자입니다. *Reactive Manifesto*의 저자 중 한 명이며 오픈소스 LMAX Disruptor 프로젝트의 Duke Award 수상자이기도 합니다. 데이브는 지속적 배포의 선구자이고, CD, DevOps, 테스트 기반 개발(TDD), 소프트웨어 설계 분야의 사고 리더이자 전문가입니다. 성과가 우수한 팀을 만들고, 성공을 위한 조직을 개편하고, 뛰어난 소프트웨어를 만드는 데 많은 경험을 가지고 있습니다. 데이브는 전 세계의 소프트웨어 개발자들에게 자신의 경험과 기술을 공유하며 이들이 소프트웨어 디자인, 품질, 신뢰성을 개선하도록 돕고 있습니다. 또한 자신의 컨설팅¹, YouTube 채널², 교육 과정을 통해 전문성을 공유합니다.



에릭 맥스웰

에릭 맥스웰은 Google에서 DevOps 혁신 관행을 이끌고 있으며 세계 최고 기업을 대상으로 가치를 더 빠르게 제공하여 개선하는 방법에 대해 조언합니다. 에릭은 엔지니어로서 최전선에서 모든 일을 자동화하고 다른 실무자들에 대한 공감대를 형성하는 데 경력의 전반부를 보냈습니다. 또한 Google 클라우드 애플리케이션 현대화 프로그램(CAMP)의 공동 작성자이며 DORA팀의 멤버입니다. Google에서 근무하기 전에는 Chef Software에서 즐거운 동료들과 멋진 소프트웨어를 개발했습니다.



제임스 브룩뱅크

제임스 브룩뱅크는 Google의 클라우드 솔루션 설계자입니다. 솔루션 설계자는 복잡한 기술 문제를 해결하고 아키텍처와 관련된 전문적 조언을 제공하여 Google Cloud 고객을 돕습니다. Google에 합류하기 전에는 IT 인프라 및 금융 서비스에 중점을 두고 여러 대기업에서 근무했습니다.



제프리 와이너 박사

제프리 P. 와이너 박사는 보스턴 아동 병원 외상 및 커뮤니티 회복 센터(TCRC)의 주치의 심리학자, 행동 건강 시스템 컨설턴트, 심리사회적 치료 개발자이자 하버드 의학전문대학원의 조교수입니다. 제프리 박사는 TCRC의 동료들과 함께 난민 및 이민자 배경의 청소년과 가족을 대상으로 문화적으로 대응하고 외상에 근거한 심리사회적 개입을 구축, 테스트, 보급, 실시하는 연구에 주력하고 있습니다. *Mental Health Practice with Immigrant and Refugee Youth: A Socioecological Framework*(이민자 및 난민 청소년을 위한 정신 건강 관행: 심리사회적 프레임워크)³의 공동 저자이며 미국과 캐나다에서 프로그램 관련 컨설팅을 진행했습니다. 제프리 박사가 개발하거나 개작한 심리사회학 예방 및 개입 도구는 전 세계적으로 사용되고 있습니다. 자세한 내용은 <http://www.drjeffwiner.com>을 참조하시기 바랍니다.

¹ <https://continuous-delivery.co.uk/engineering-for-software>

² <https://www.youtube.com/@ContinuousDelivery>

³ <https://www.apa.org/pubs/books/4317536>



케빈 스토어러

케빈 M. 스토어러는 Google의 사용자 경험 연구원으로, 소프트웨어 개발팀이 DevOps 도구를 통해 상호작용하는 방법을 이해하는 연구를 주도하고 있습니다. Google에 합류하기 전에는 어바인의 캘리포니아 대학교에서 정보 과학 분야 박사 학위를 취득했습니다. 케빈은 공공 부문과 민간 부문을 넘나들며 인간 중심 프로그래밍, 개발자 경험, 정보 행동, 접근성, 유비쿼터스 컴퓨팅을 주제로 한 영향력 있는 출판물을 저술했습니다.



킴 카스티요

킴 카스티요는 Google의 사용자 경험 프로그램 관리자입니다. 킴은 DORA의 연구 운영을 감독하는 것부터 이 보고서의 출판까지 DORA의 다양한 활동을 주도하고 있습니다. 또한 Google Cloud에서 Duet AI의 UX 연구도 진행하고 있습니다. Google에 합류하기 전에는 소프트웨어 배포 부문에서 경력을 쌓고, 기술 프로그램 관리 및 애자일 코칭 부문에서 업무를 수행했습니다. 킴은 모국인 필리핀에서 일어나는 정부당국의 법적 절차 없는 살인, 도시 빈곤 지역 개발, 지역사회 트라우마, 회복력에 초점을 둔 심리사회 연구를 주로 진행해 왔습니다. DORA는 킴이 보유한 다방면의 열정을 소프트웨어 배포 관행, 심리적 안전, 팀조직에 결합합니다.



미셸 어바인

미셸 어바인은 Google의 테크니컬 라이터로, 기술 문서의 영향과 제작에 관한 연구를 주도하고 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.



네이슨 하비

네이슨 하비는 Google Cloud의 개발자 관계 엔지니어링 관리자로 DORA Advocacy팀을 이끌고 있습니다. 네이슨은 최고의 팀 및 오픈소스 커뮤니티와 협력하여 DevOps 및 SRE의 원칙과 관행을 적용하는 데 도움을 주었습니다. 또한 지난 3년간 *Accelerate State of DevOps* 보고서의 공동 저자로 활동해 왔습니다. 네이슨은 *97 Things Every Cloud Engineer Should Know*를 공동 편집하고 기고하기도 했습니다.



스티브 맥기

스티브 맥기는 팀이 세계 최고 수준의 신뢰할 수 있는 서비스를 구축하고 운영할 수 있는 최고의 방법을 이해하도록 지원하는 신뢰성 지지자입니다. 그 전에는 Google에서 사이트 안정성 엔지니어로 10년 넘게 근무하며 검색, YouTube, Android, Google Cloud에서 글로벌 시스템을 확장하는 방법을 배웠습니다. 또한 캘리포니아, 일본, 영국에서 여러 엔지니어링팀을 관리했습니다. 스티브는 캘리포니아 기반의 기업을 도와 클라우드로 전환하는 업무를 맡기도 했습니다.

방법론

이 장에서는 초기 아이디어를 좀 더 완성도 높은 아이디어로 발전시켜 어떻게 지금 여러분 앞에 있는 보고서로 탄생했는지 간략히 설명합니다. 이 보고서 작성 과정에 대한 수많은 질문의 답이 되길 바라며 여러분이 자체적으로 수행하는 연구를 도울 수 있는 청사진을 제공할 것입니다.

1단계: 실적이 우수한 기술 기반 조직에 중요하다고 생각되는 일련의 성과 생성

매우 중요한 단계입니다. DORA의 프로그램은 직원들이 중요하게 생각하는 곳으로 안내하는 것을 주 목적으로 합니다. 만약 직원, 조직 또는 팀이 가려는 곳을 모른다면 좋지 못한 시작일 될 것입니다. 이것을 어떻게 알 수 있을까요? DORA는 질적 조사(직원에게 자신과 자신의 팀 또는 조직이 달성하고자 하는 목표를 묻는 것), 설문조사, 다양한 커뮤니티와의 소통, 여러 차례의 워크숍을 활용합니다. 그리고 일관되게 다음과 같은 성과가 중요하다는 결론이 도출되었습니다.

- **조직 실적.** 조직은 수익 뿐만 아니라 고객과 더 넓은 차원의 커뮤니티를 위한 가치도 창출해야 합니다.
- **팀 실적.** 애플리케이션 또는 서비스팀이 가치를 창출하고 혁신하고 공동작업할 수 있는 능력입니다.
- **직원 웰빙.** 조직이나 팀이 채택하는 전략은 번아웃을 줄이고, 직무 만족도를 높이고, 가치 있는 결과를 창출하기 위한 능력(즉, 생산성)을 향상시키는 등 직원에게 이점을 제공해야 합니다.

또한 다음과 같은 목표를 중요시한다는 의견도 들었습니다.

- **소프트웨어 배포 실적.** 팀이 소프트웨어를 빠르고 성공적으로 배포할 수 있는 능력입니다.
- **운영 실적.** 출시되는 소프트웨어에서 사용자에게 안정적인 경험을 제공하는 능력입니다.



2단계: 이러한 성과가 달성되는 방법, 시기, 요인에 관한 가설 설정

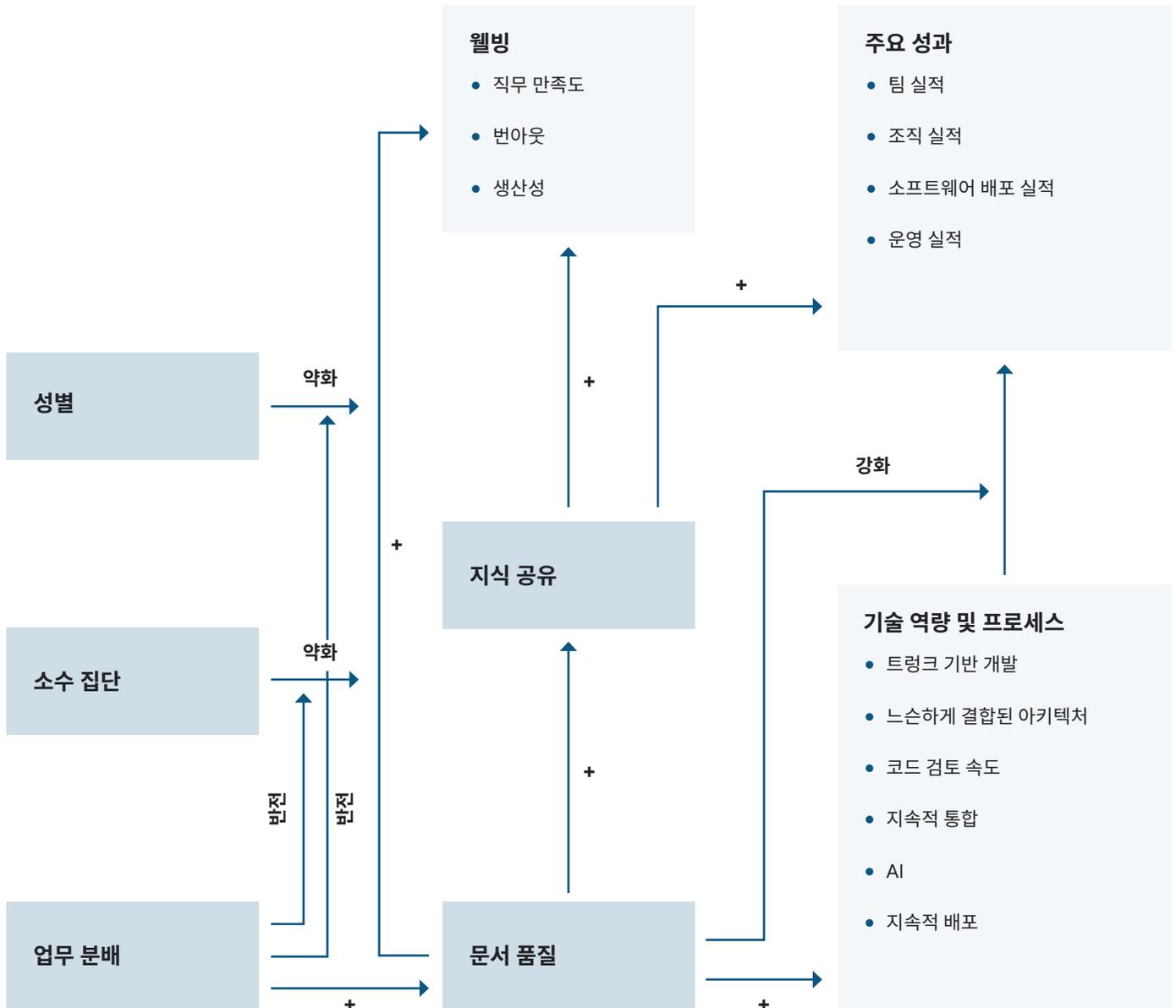
1단계에서 살펴본 성과를 염두에 두고 팀이 나아갈 수 있는 방향에 대한 가설을 설정해야 합니다. 여기에는 성과에 신뢰할 수 있는 영향을 미치는 것으로 보이는 요인을 살펴보는 것이 포함됩니다. 예를 들면 이런 식의 가설입니다. '모든 조건을 똑같이 유지했을 때 x가 y에 미치는 영향은?' 이 정보는 실무자가 어떤 변화를 시도할지에 대해 데이터에 근거한 의사결정을 내릴 수 있도록 도와줍니다.

DORA는 어떤 경로가 영향을 미칠지 파악할 뿐만 아니라, 그러한 경로가 어떤 조건하에서 더 큰 영향을 줄지, 아니면 더 작은 영향을 줄지 알아보려고 했습니다. 이는 '언제', 그리고 '누구에게'라는 질문을 하는 것과 같습니다. 예를 들어 문서 품질은 평균적으로 번아웃을 크게 감소시키는 것으로 드러났습니다. 소수 집단에 속하는 것으로 식별되는 응답자를 살펴보면 반대 결과가 나타납니다. 문서 품질은 번아웃을 증가시킵니다. 이러한 영향에 대한 조건을 이해하는 것이 중요합니다. 왜냐하면 평균적인 사람이나 팀은 드물기 때문입니다.

DORA는 *왜* 또는 *어떻게*를 설명하는 *메커니즘*에 대한 가설도 설정합니다. 올해에는 작년에 나타난 결과와 주제에 대한 다양한 문헌을 토대로 소수 집단에 속하는 것으로 식별되는 직원이 더 많은 번아웃을 경험했다는 가설을 세웠습니다. 직원들이 가장 먼저 하는 질문은 "왜 이런 현상이 일어나는 거지?"입니다. 이 질문에 대한 답을 하기 위해 테스트할 수 있는 잠재적 메커니즘에 대한 가설을 세웠습니다. 예를 들어 소수 집단에 속하는 것으로 식별되는 사람들에게는 좀 더 반복적인 업무가 배정되기 *때문에* 번아웃을 더 많이 경험한다는 것입니다.



그런 다음 이러한 가설을 기반으로 설문조사를 설계하고 분석을 실행합니다. 다음은 모든 영향, 조건, 메커니즘이 포함된 문서 장애 대한 가설 모델의 예시입니다.

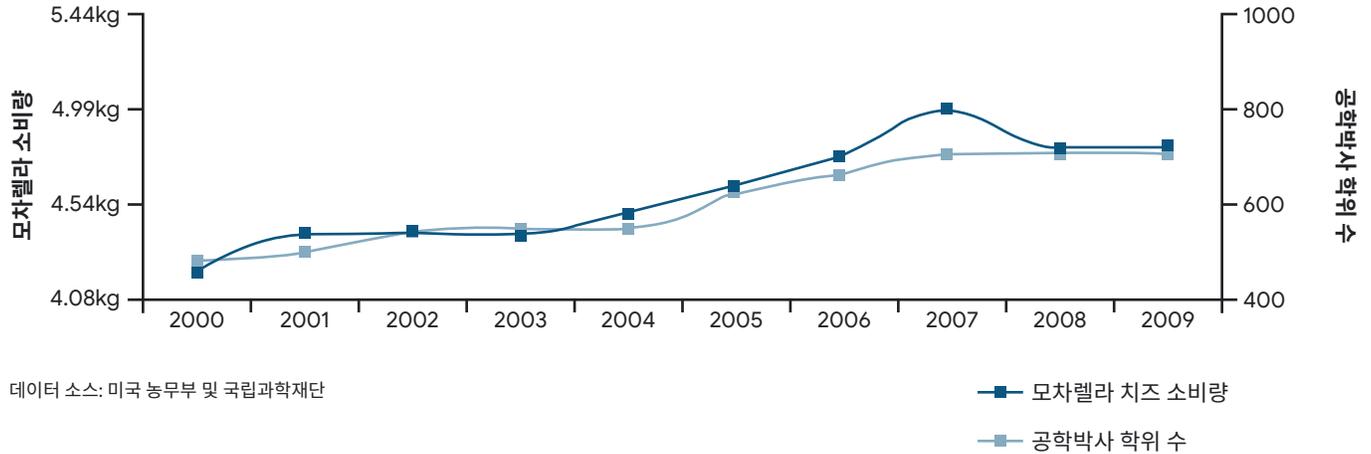


3단계: 잠재적 혼란 변수에 관한 가설 설정

데이터에 대해 논의한 적이 있다면 비논리적인 상관관계를 발견하는 경우가 있었을 것입니다. 다음과 같이 비논리적인 상관관계¹를 표시하는 웹사이트도 많이 보셨을 것입니다.

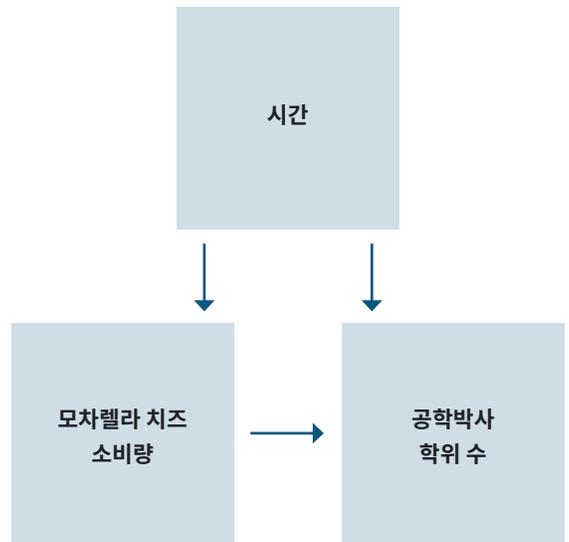
모차렐라 치즈의 1인당 소비량과 수여된 토목공학 분야 박사 학위 개수 간의 상관관계

상관관계: 95.86%($r=0.958648$)



공학박사 학위 수와 모차렐라 치즈 소비량 간에 인과관계가 있을 가능성은 거의 없습니다. 그럼에도 불구하고 이 관계의 이면에는 혼란 요소가 숨어 있습니다. 바로 시간입니다. 모차렐라 치즈 소비량과 공학박사 학위 수가 모두 동일한 기간에 증가 추세를 보인다면 이 둘 간에는 양의 상관관계가 있을 가능성이 있는 것입니다.

만약 모델에 시간을 추가하거나 데이터의 추세를 제거하면 이러한 상관관계가 사라질 수 있습니다. 이 모델은 다음과 같이 그릴 수 있습니다.



¹ <https://www.tylervigen.com/spurious-correlations>

시간(세 번째 변수)을 고려하지 않으면 데이터가 모차렐라 치즈 소비량과 공학박사 학위 수 간 비논리적인 상관관계를 나타낼 수 있습니다.

이와 관련해서 연구원에게 도움을 줄 수 있는 도구가 있습니다. 대표적으로 Dagitty(<https://dagitty.net/dags.html>)가 있습니다. 이 도구는 인과 모델을 정하는 데 도움을 줍니다. Y에 대한 X의 영향을 적절하게 추정할 수 있도록 모델의 의미, 고려해야 할 사항, 고려하지 *말아야* 할 사항 등을 알려줍니다. Dagitty 등의 도구는 상관관계가 인과관계를 의미하지는 않아도 누군가 인과관계에 대해 *생각*하는 방식을 의미한다는 결론을 도출할 수 있습니다.

연구원들의 추정에 편향 효과를 줄 수 있는 모든 요소를 포착하는 것은 불가능합니다. 50시간씩 걸리는 설문조사를 진행할 수는 없을뿐더러 이 세상 모든 걸 다 알고 있는 사람은 없으니 말입니다. 그럼에도 DORA는 직원이 중요시하는 성과에 대한 다양한 활동, 기술, 구조의 정확한 영향 추정치를 제공하기 위해 편향된 경로를 고려하려고 노력합니다. 결국 실무자들은 이러한 주요 성과에 영향을 미칠 수 있는 요인을 알고 싶어합니다. 편향 효과를 고려하지 못한 모델은 실무자에게 제대로 된 가이드를 전달하지 못할 것입니다. 모차렐라 치즈가 소프트웨어 배포 실적을 높인다고 말하고 싶지 않습니다. 다만 이러한 실수는 여러분이 생각하는 것보다 더 쉽게 발생합니다.



4단계: 설문조사 개발

설문조사 개발에는 운영, 경험, 현지화의 3가지 요소가 있습니다.

운영

DORA는 관심 있는 개념을 적절하게 포착하고 이를 신뢰할 수 있는 방식으로 측정하려고 합니다. 추상적인 개념을 측정 가능한 것으로 변환하는 것이 운영의 묘미입니다. 이러한 측정은 모든 분석의 기초가 되는 요소입니다. 만약 측정을 통해 우리가 명확한 신호를 얻지 못한다면 분석의 나머지 부분을 어떻게 신뢰할 수 있을까요? 생산성과 같은 규정하기 어려운 개념을 어떻게 측정할 수 있을까요? 번아웃이나 운영 실적은 또 어떨까요?

먼저 기존의 성공적인 측정 사례가 있었는지 자료를 찾아봅니다. DORA에서 진행할 설문조사에 이전에 검증된 측정값을 사용할 수 있다면 설문조사와 해당 질문에 관해 축적된 모든 자료를 연결하는 다리가 만들어지는 것입니다. 이전에 검증된 측정값을 재사용하는 사례로 웨스트럼이 개발한 *조직 문화의 유형*을 계속해서 사용하는 것이 있습니다.

하지만 DORA가 연구하는 분야에서는 이전에 검증된 개념이 많지 않습니다. 이 경우 직원들이 개념을 이해하는 방식을 해결하기 위해 질적 조사를 수행하며 개념의 복잡성에 대한 좀 더 철학적인 자료를 살펴봅니다.

설문조사 경험

DORA는 이해하기 쉽고, 명료하며, 필요 이상으로 길지 않고, 광범위하게 접근할 수 있는 설문조사를 작성하고자 합니다. 묻고 싶은 질문의 양과 질문에 답을 하기 위해 필요한 기술적 이해도, 특정 관행에 대한 명칭이 다르다는 점을 고려할 때 이는 어려운 목표입니다. DORA는 설문조사가 특정 기준점을 넘어서 수행되도록 관리자가 개입하지 않는 원격 평가를 실시합니다. 이를 위해서는 여러 차례의 반복 수행이 필요합니다.

현지화

매년 전 세계 많은 사람들이 DORA의 설문조사에 응답했습니다. 올해에는 설문조사를 English, Español, Français, Português, 日本語로 현지화하여 더 넓은 범위의 독자가 접근할 수 있도록 했습니다. 여기에는 DORA 커뮤니티의 훌륭한 구성원들이 주도한 풀뿌리 차원의 노력이 있었습니다. 전 세계 여러 Google 직원과 업계 파트너도 이러한 노력에 동참했습니다. 그중 프랑스어 현지화에 도움을 준 Zenika(<https://www.zenika.com>)에 감사의 인사를 전합니다. 앞으로 이러한 노력을 확대하고 진정으로 다문화적인 설문조사를 만들 수 있기를 바랍니다.

5단계: 설문조사 응답 수집

응답자 모집에는 다중 채널을 사용합니다. 이러한 채널은 크게 두 가지 카테고리로 나눌 수 있습니다. 바로 *자연 방식*과 *패널 방식*입니다.

*자연 방식*은 원하는 소셜 수단을 모두 사용해서 사람들에게 설문조사에 참여해 달라고 알리는 것입니다. 블로그 게시물을 작성하고 이메일 캠페인을 사용합니다. 그리고 소셜 미디어에 게시물을 작성하고 커뮤니티의 사람들에게도 똑같이 해달라고 요청합니다. 스노볼 효과를 노리기 위해서죠.

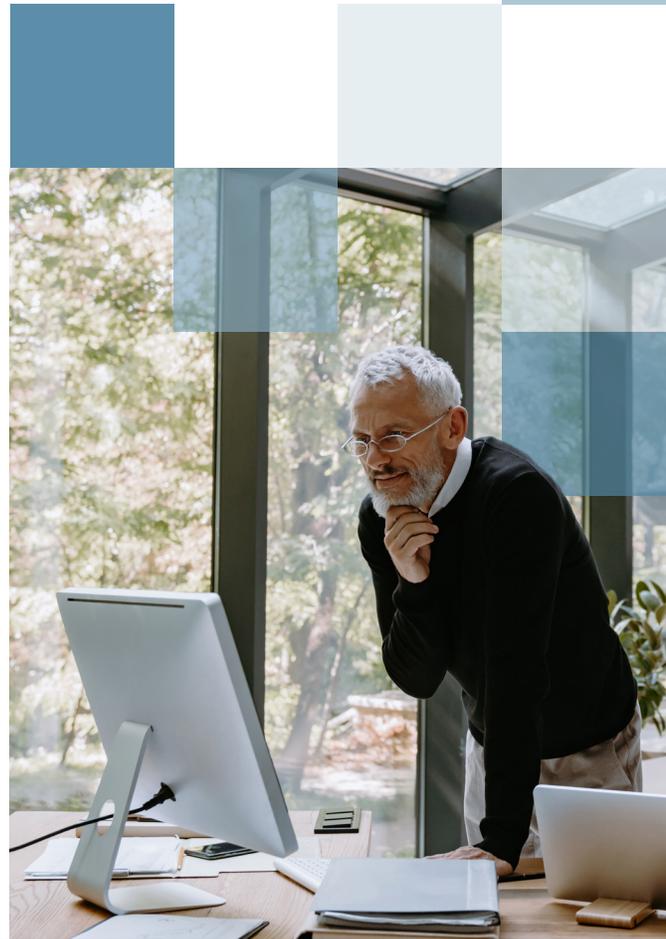
*패널 방식*은 자연 채널을 보완하기 위해 사용합니다. 여기서는 더 광범위한 기술 커뮤니티에서 전통적으로 소수 집단에 속하는 사람들을 모집하려고 합니다. 그리고 특정 산업과 조직에서 충분한 응답을 얻고자 합니다. 간단히 말하면 이 방식을 통해 모집 인원을 통제할 수 있습니다. 자연 방식으로는 불가능한 부분을 통제하는 것입니다. 패널 방식을 통해 충분한 응답자를 얻을 수도 있습니다. 자연 방식으로는 분석을 수행하기 위해 충분한 응답을 얻을 수 있을지 모르기 때문입니다.

6단계: 데이터 분석하기

분석에는 데이터 정리, 측정 검증, 모델 평가의 3가지 핵심 단계가 있습니다.

데이터 정리

데이터 정리의 목표는 신호 대 잡음비를 높이는 것입니다. 일부 응답에는 다양한 이유로 노이즈가 섞여 있습니다. 몇몇 응답자의 경우 설문조사 수행 시 산만한 상태이거나 제대로 읽지 않고 대충 훑어보거나 충실히 답변하지 않는 패턴이 나타납니다. 이런 패턴은 DORA의 데이터에서도 매우 드물지만 존재합니다. 실제 경험에 반하여 허위로 답변한 징후가 보이는 응답은 노이즈이기 때문에 분석 데이터에서 제외됩니다. 노이즈를 제거할 때 주의해야 할 점은 편향된 방식으로 또는 가설을 검증하는 방식으로 신호를 제거하지 않도록 하는 것입니다. 예를 들어 DORA에서 특정 값이 높게 나타나고 다른 값은 낮게 나타나는 것이 논리적이지 않다고 결론을 내리는 경우 해당 응답자를 제외시킬 수 있습니다. 우리의 신념과 더 일치하는 데이터를 만들고 가설이 실현될 가능성을 높이기 위해서입니다.



측정 검증

이 보고서의 시작 부분에서 측정하려는 개념에 대해 언급했었습니다. 우리가 참여할 수 있는 언어 게임에는 여러 가지가 있지만 한 가지 공통적인 관점은 이러한 개념의 측정을 변수라고 부른다는 점입니다.² 이러한 변수는 DORA의 연구에 포함된 모델의 재료입니다. 이러한 측정의 유효성을 분석하기 위한 방법으로는 내부적인 방법과 외부적인 방법이 있습니다.

측정의 내부적 유효성을 이해하기 위해 개념의 존재에 대한 지표가 되는 대상을 살펴봅니다. 예를 들어 고품질 문서는 문제를 해결하기 위해 문서를 활용하는 사람의 지표가 될 수 있습니다.

대부분의 변수는 다수의 지표로 구성됩니다. DORA에서 관심을 가지고 지켜보는 구성 개념은 다면적 특성을 가지고 있기 때문입니다. 변수의 다면적 특성을 이해하기 위해서는 DORA에서 사용하는 항목이 해당 구성 개념과 얼마나 잘 맞는지 테스트합니다. 잘 맞는 경우(즉, 높은 수준의 공통 변수를 공유하는 경우), 무언가가 해당 관심 개념의 기저를 이룬다고 가정합니다.

행복을 예로 들어보겠습니다. 행복은 다면적입니다. 우리는 누군가가 행복감을 느낄 때 특정한 방식으로 느끼고, 특정한 방식으로 행동하고, 특정한 방식으로 생각하길 기대합니다. DORA에서는 행복이 특정한 패턴의 감정, 생각, 행동에 기반한다고 가정합니다. 따라서 행복이 찾아오면 특정 유형의 감정, 생각, 행동이 동시에 나타날 거라고 기대합니다. 그런

다음 그러한 감정, 생각, 행동에 대한 질문을 합니다. 이들이 실제로도 동시에 나타나는지 테스트하기 위해서는 확인적 요인 분석을 수행합니다.

올해에는 이 분석을 수행하기 위해 Lavaan R 패키지를 활용했습니다.³ Lavaan은 구성 개념이 실제로 사람들이 질문에 답변하는 방식을 대표하는지 파악할 수 있도록 도와주는 다양한 적합도 통계치를 반환합니다. 개념의 지표가 맞지 않으면 개념을 측정할 신뢰할 수 있는 방법을 찾지 못한 것이기 때문에 개념을 수정 또는 삭제해야 할 수 있습니다.

구성 개념의 외부적 유효성은 구성 개념이 실제 세계와 얼마나 일치하는지 살펴보는 것으로 확인할 수 있습니다. 하나의 구성 개념은 다른 구성 개념과 특정한 관계를 가질 것으로 기대할 수 있습니다. 행복과 슬픔이라는 두 가지 구성 개념은 음의 상관관계에 있다고 예상할 수 있습니다. 만약 슬픔과 양의 상관관계에 있는 행복 측정값이 도출된다면 측정 결과나 이론에 의문을 품을 것입니다. 마찬가지로 두 가지 구성 개념이 양의 상관관계에 있다고 예상할 수 있으나 관계성이 강하지는 않을 것입니다. 생산성과 직무 만족도는 양의 상관관계가 있을 가능성이 있지만 동일한 개념은 아닙니다. 상관관계가 너무 높으면 동일한 것을 측정하는 것처럼 보일 수 있습니다. 이는 측정값이 두 개념 간의 차이를 식별할 수 있을 만큼 충분히 보정되지 않았거나 가설을 세운 차이가 실제로는 존재하지 않는다는 의미입니다.

² 무어, 윌 H., 데이비드A. 시겔. A mathematics course for political and social research(정치 및 사회 연구를 위해 고안된 수학 과정). Princeton University Press, 2013.

³ 로셀, Y. 'lavaan: An R Package for Structural Equation Modeling(lavaan: 구조 방정식 모델링을 위한 R 패키지)', Journal of Statistical Software, 48(2), 2012. 1-36. <https://doi.org/10.18637/jss.v048.i02>

모델 평가

2~3단계에서는 가설 모델을 빌드했습니다. 6단계에서 데이터를 정리한 이후에는 모델이 데이터에 얼마나 잘 맞는지 확인했습니다. 올해는 데이터가 귀무 가설(즉, 효과 없음)로 결론 내려질 가능성 대신 데이터에 대한 다양한 가설이 얼마나 타당한지 파악하기 위해 베이지 접근 방식을 채택했습니다. DORA에서 R에 사용하는 주요 도구는 blavaan⁴ 및 rstanarm⁵입니다. 영향이 미미하지 않고 상당하거나 극적일 가능성을 테스트할 수 있습니다. 모델을 평가할 때는 절약의 법칙을 사용합니다. 이는 매우 단순한 모델에서 시작하여 복잡성이 더 이상 정당화되지 않을 때까지 복잡성을 추가하는 방식입니다. 예를 들어 조직 실적이 소프트웨어 배포 실적과 운영 실적 간 상호작용의 산물이라고 예측합니다.

이번 연구에 사용한 단순 모델에는 상호작용이 포함되지 않습니다.

조직 실적 ~ 소프트웨어 배포 실적 + 운영 실적

두 번째 모델은 상호작용을 추가합니다.

조직 실적 ~ 소프트웨어 배포 실적 + 운영 실적 +
소프트웨어 배포 실적 x 운영 실적

'Regression and other stories(회귀분석 및 기타 예시)⁶ 및 'Statistical Rethinking(통계적 재검토)⁷에서 권장하는 내용에 따라 추가적인 복잡성이 필요한지 확인하기 위해서는 LOOCV(Leave-One-Out Cross-Validation) 및 광범위하게 적용 가능한 Watanabe–Akaike 정보 기준⁸을 사용합니다.

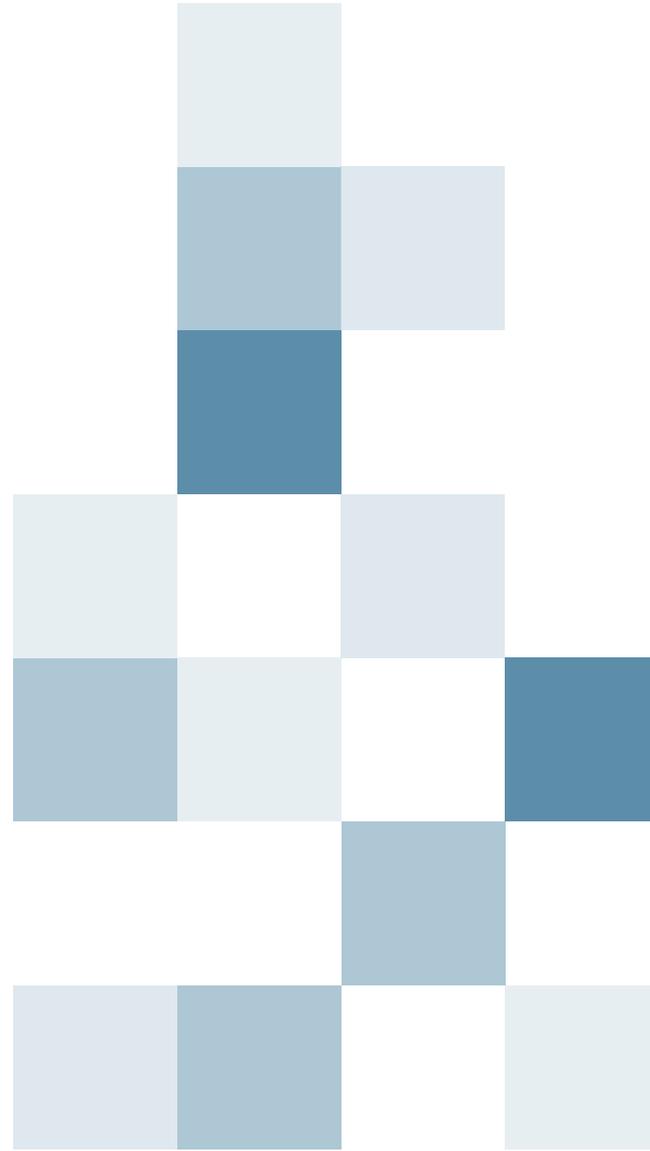
⁴ 머클, 에드가 C., 이브 로셀. 'blavaan: Bayesian structural equation models via parameter expansion(blavaan: 매개변수 확장을 통한 베이지안 구조 방정식 모델)', arXiv preprint, 2015. <https://doi.org/10.48550/arXiv.1511.05604>

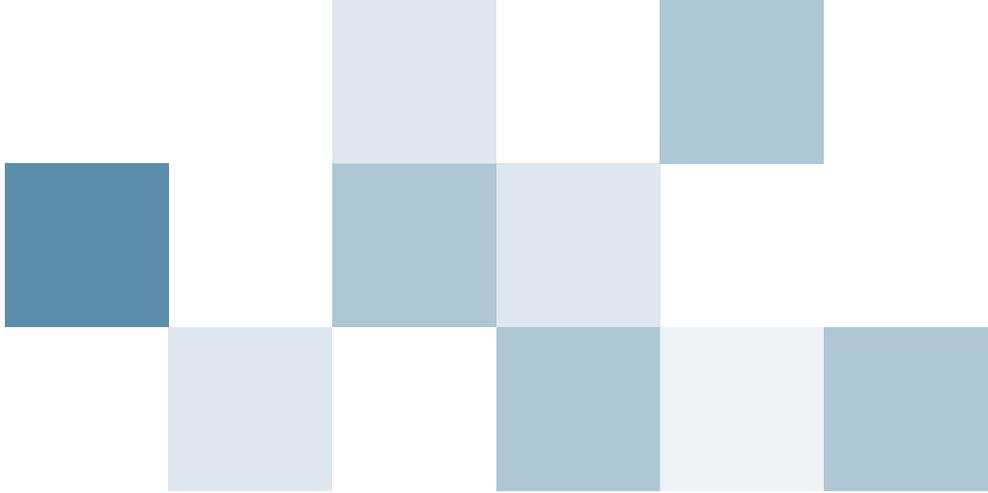
⁵ 굿리치, 벤, 조나 가브리, 이마드 알리, 샘 브릴만. 'rstanarm: Bayesian applied regression modeling via Stan(rstanarm: Stan을 통해 적용된 베이지안 회귀 모델링)'. R package version 2, no. 1 (2020).

⁶ 켈만, 앤드류, 제니퍼 힐, 아키 베흐타리. Regression and Other Stories (회귀분석 및 기타 예시) (Cambridge University Press, 2020).

⁷ 매셀리스, 리차드. Statistical rethinking: A Bayesian course with examples in R and Stan(통계적 재검토: R 및 Stan의 예를 사용한 베이지안 방법 과정). Chapman and Hall/CRC, 2018.

⁸ 베흐타리, 아키, 앤드류 켈만, 조나 가브리 'Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC(Leave-One-Out 교차 검증 및 WAIC를 사용한 실용적인 베이지안 모델 평가)'. Statistics and Computing, 27, 2017. 1,413–1,432. <https://doi.org/10.1007/s11222-016-9696-4>





7단계: 보고서 결과

그런 다음 팀 단위로 이러한 결과를 검토했습니다. 올해에는 콜로라도주 볼더에서 며칠간 함께 지내며 주제 전문가의 경험에 데이터를 결합했습니다. 보고서의 모든 장에서 각 가설별로 이 작업을 수행했습니다. 데이터 해석에는 항상 편견, 추측, 개인적 견해, 비약 등의 위험이 존재합니다. 이러한 위험은 다양한 배경을 가진 여러 명의 사람들을 모아 질문, 의견 차이, 독특한 관점, 호기심을 장려하는 회의를 진행하며 완화할 수 있었습니다.⁹

보고서 저자들은 회의 결과를 가지고 각자의 세상으로 돌아가 보고서를 작성했습니다. 작성 프로세스 전반에서 편집자와 주제 전문가의 자문을 받았습니다. 이러한 관점은 아이디어를 전달하는 데 중요했습니다. 이 데이터의 분석을 책임지는 사람은 회의에서 나온 어떤 발언도 데이터가 말하는 것을 벗어나지 않도록 했습니다.

DORA의 유능한 디자인 파트너인 BrightCarbon¹⁰은 보고서에 포함된 각 장을 조화롭게 엮었습니다.

8단계: 연구 결과를 커뮤니티에 결합

DORA는 커뮤니티 참여를 통해 이러한 연구 결과를 활용하고 해석할 수 있는 방법을 찾습니다. 권장 사항을 제시하려고 노력하지만 결국에는 DORA의 연구를 통해 밝혀진 결과를 기반으로 팀이 다양한 시도를 해볼 수 있습니다. 예를 들어 느슨하게 결합된 아키텍처는 DORA의 측정 결과를 기반으로 유용한 관행이 것으로 나타났습니다. 하지만 느슨하게 결합된 아키텍처를 구축할 수 있는 방법이 유일하지 않다는 점은 확실합니다. 커뮤니티로서 접근 방식을 생성하고 공유하는 것이 지속적으로 개선할 수 있는 유일한 방법입니다. DORA가 그리는 세상은 여러분과 여러분의 팀, 조직이 운영하는 영역과 상황에 대해 해석하고 추상화하는 것입니다.

DORA의 글로벌 커뮤니티에 참여하려면 DORA Community 사이트(<https://dora.community>)를 방문하세요.

⁹ 스타서, G., 티터스, W. (1985). 'Pooling of unshared information in group decision making: Biased information sampling during discussion(그룹 의사결정을 위해 공유되지 않은 정보 수집: 토론 중 편향된 정보 샘플링)'. Journal of Personality and Social Psychology, 48(6), 1985. 1,467-1,478. <https://doi.org/10.1037/0022-3514.48.6.1467>

¹⁰ <https://www.brightcarbon.com/>

인구통계 및 기업통계

설문조사 대상

지난 10년 가까이 DORA 연구 프로그램은 실적이 우수한 기술 기반 조직의 역량, 관행, 측정값을 조사해 왔습니다. 규모와 업종을 막론한 수많은 조직에서 일하는 36,000명이 넘는 전문가의 다양한 의견을 들었습니다. 유용한 정보를 공유해 주신 모든 분께 감사드립니다. 올해는 전 세계의 다양한 업계에서 일하는 약 3,000명의 전문가들이 경험을 공유하여 실적이 우수한 기술 기반 조직이 성공하는 요인에 대한 이해를 높이는 데 도움을 주었습니다.

올해의 인구통계 및 기업통계 질문은 Stack Overflow에서 완료한 연구를 활용했습니다. 2022년 Stack Overflow 개발자 설문조사에는 70,000여 명의 응답자가 참여했습니다.¹ 해당 설문조사는 여러 가지 이유로 모든 기술 실무자에게 닿은 것은 아니지만 개발자 업계의 모든 인구층에 최대한 근접하고자 했습니다. 여기서 얻은 인구에 대한 감각을 통해 데이터에서 응답 편향을 찾아 연구 결과를 어느

정도까지 일반화하고자 하는지 파악했습니다. 해당 데이터와 Stack Overflow 개발자 설문조사에서 제시된 인구통계 및 기업통계 질문은 우수한 품질로 작성되었으며 차용할 가치가 충분합니다. Stack Overflow 개발자 설문조사에 비해 DORA의 샘플에는 여성, 장애인 참가자, 대기업 소속 참가자가 더 높은 비율로 포함되었습니다. DORA의 샘플은 인종 및 민족의 관점에서 Stack Overflow와 유사합니다.

올해 설문조사에서는 자연 방식 응답자 수가 2022년에 비해 3.6배 늘었습니다.

올해 설문조사에서는
자연 방식 응답자
수가 2022년에 비해

3.6

늘었습니다

¹ <https://survey.stackoverflow.co/2022#overview>

인구통계

성별

2022년에 비해 올해의 샘플에는 여성 응답자가 더 적은 비율로 포함되었습니다(12% 대 18%).

성별	응답자 비율
밝히고 싶지 않음	3%
스스로 규정	2%
여성	12%
남성	81%

장애

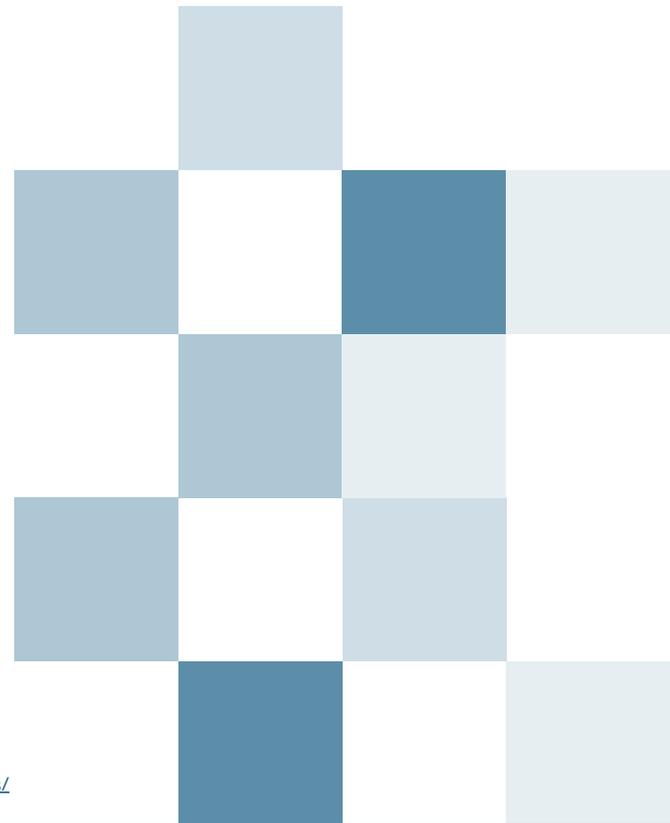
장애는 Washington Group Short Set²의 안내를 따르는 6 가지 기준으로 식별했습니다. 소수 집단에 대한 질문을 한 것은 올해가 5년째입니다. 장애인 비율은 2022년 11%에서 2023년 6%로 줄었습니다.

장애	응답자 비율
장애 없음	87%
예	6%
밝히고 싶지 않음/응답하지 않음	7%

소수 집단

소수 집단의 구성원으로 식별하는 것은 인종, 성별 또는 다른 특성을 나타낼 수 있습니다. 소수 집단에 대한 질문을 한 것은 올해가 6년째입니다. 소수 집단에 속하는 것으로 식별되는 응답자의 비율은 2022년 19%에서 2023년 15%로 약간 감소했습니다.

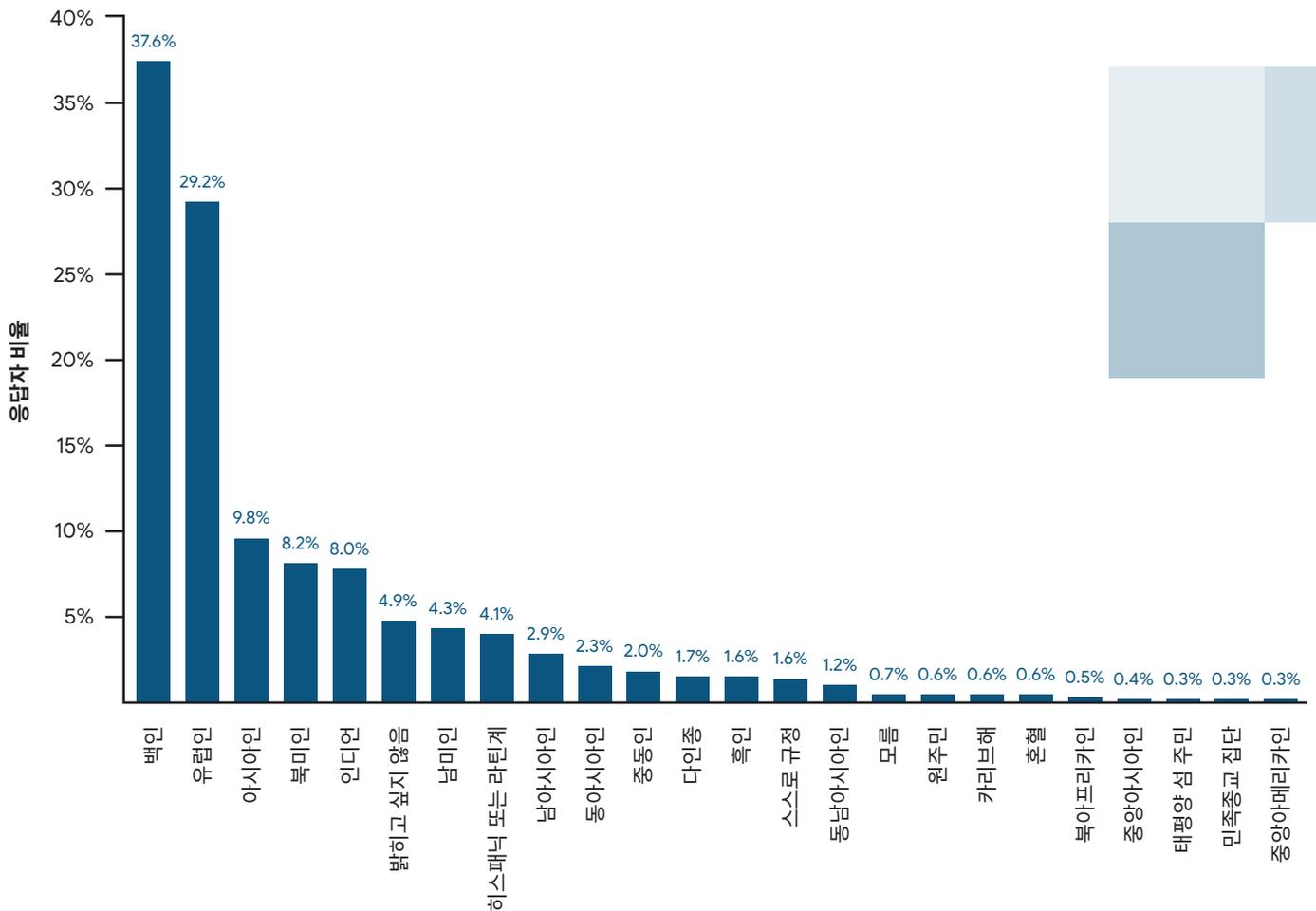
소수 집단	응답자 비율
없음	77%
예	15%
응답하고 싶지 않음	7%



² <https://www.washingtongroup-disability.com/question-sets/wg-short-set-on-functioning-wg-ss/>

인종 및 민족

2022년 Stack Overflow 개발자 설문조사의 질문을 채택했습니다.³ 앞서 언급했듯이 샘플은 한 가지 편차에 있어 유사합니다. 유럽인 비율이 낮다는 것입니다.

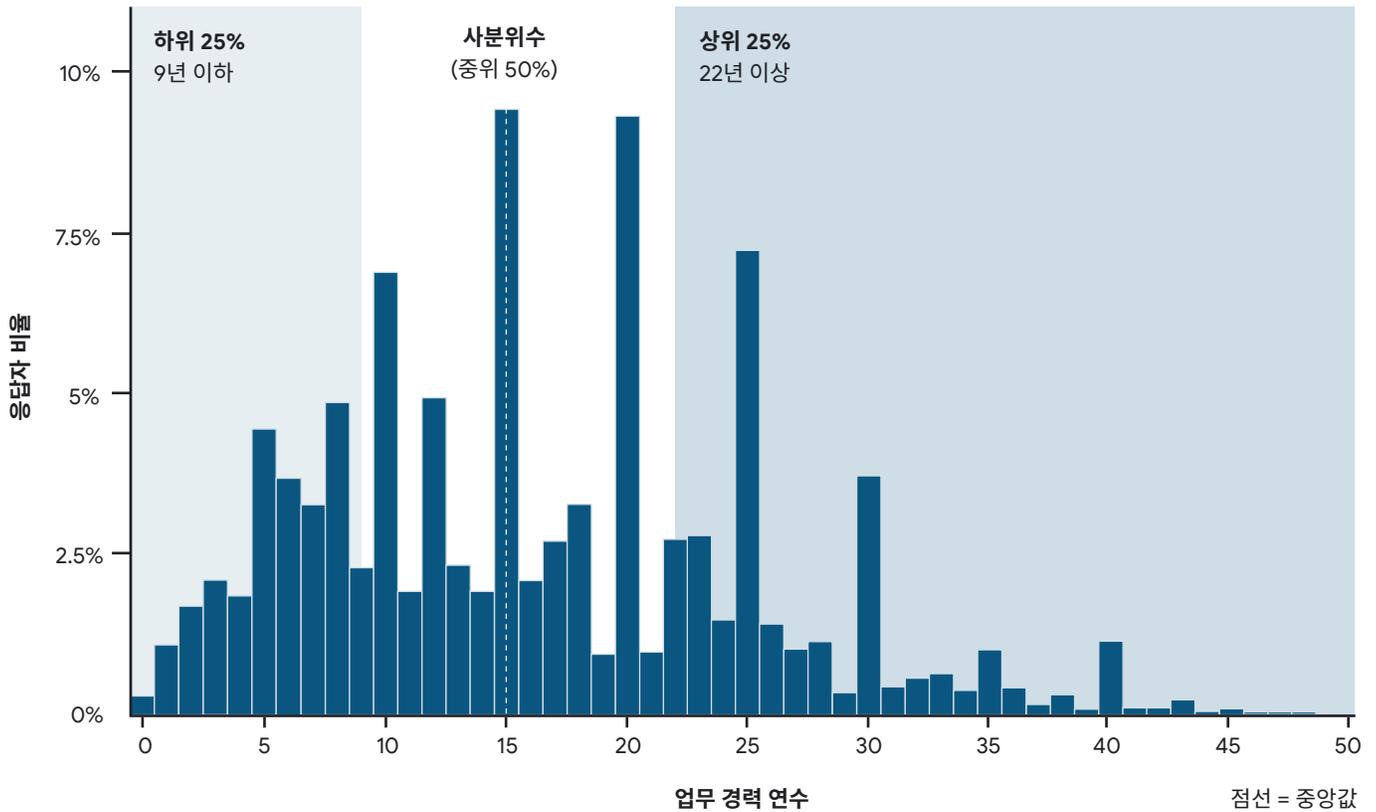
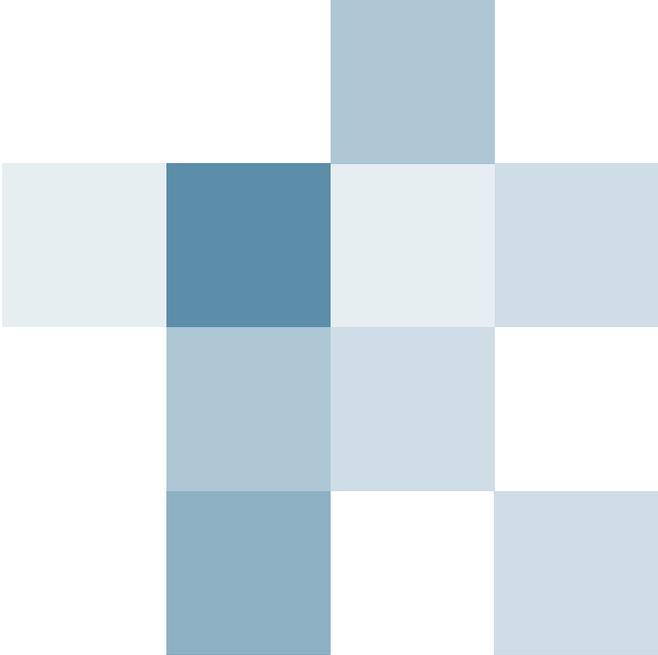


³ <https://survey.stackoverflow.co/2022#overview>

경력 기간

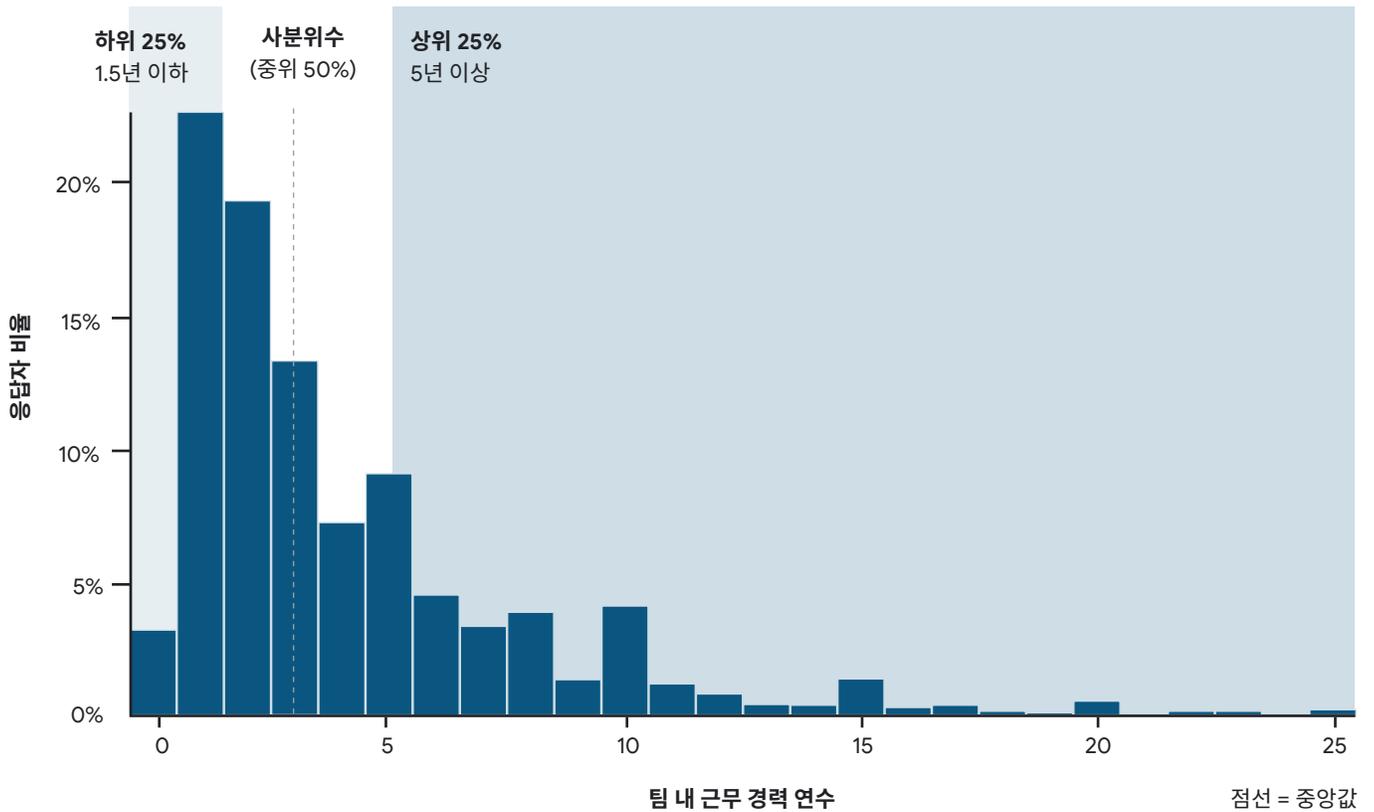
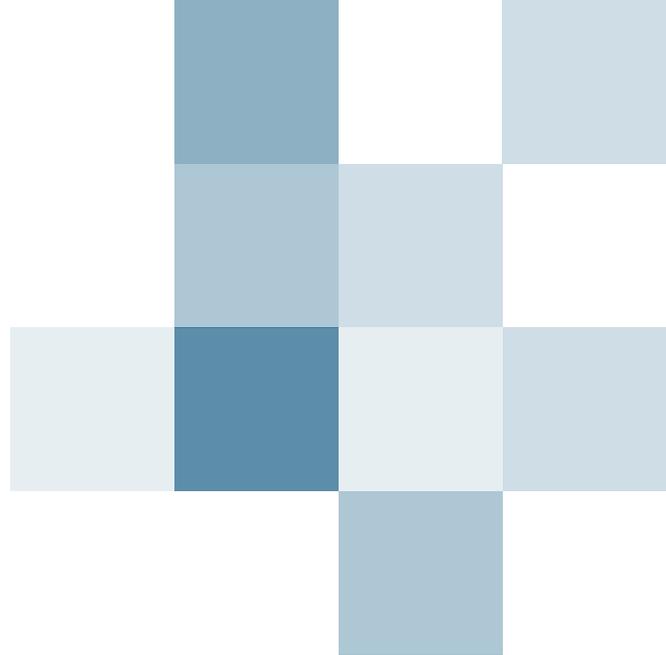
업무 경력

직원들의 업무 경력 기간을 알아보려고 했습니다. 여기서는 “업무 경력이 몇 년인가요?”라고 질문했습니다. 요약하면 DORA는 상당한 경력을 가진 실무자 그룹의 데이터를 수집합니다. 응답자의 50%가 15년 이상의 경력을 가지고 있었습니다. 22년 이상의 경력자는 25%, 9년 이하의 경력자는 25%였습니다. 지나고 나서 보니 '업무'라는 개념의 정의가 명확하지 않았습니다.



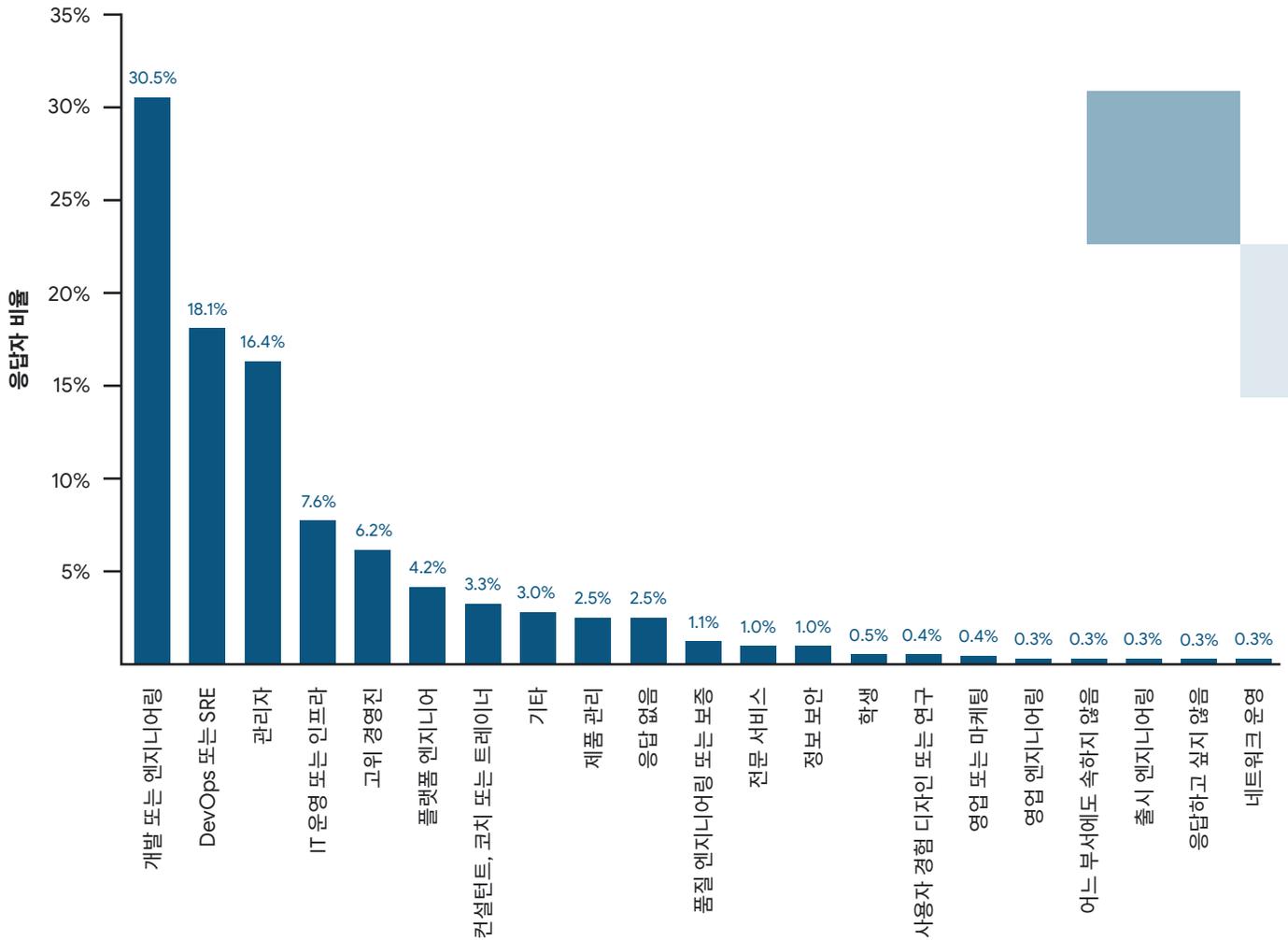
팀 내 경력

새 팀에서 일하는 것은 처음부터 다시 시작하는 것처럼 느껴질 수 있습니다. 관련 데이터를 얻기 위해 “현재 소속된 팀에서 일한 지 몇 년이나 되셨나요?”라고 물었습니다. 다수의 응답자가 업무 경력은 오래됐을지라도 소속 팀에서는 새로운 존재였습니다. 50%의 응답자가 팀에 합류한 지 3년이 채 되지 않았습니다. 25%는 팀에 합류한 지 1.5년을 넘지 않았습니다. 응답자의 25%만이 현재 팀에서의 근무 경력이 5년 이상이었습니다. 이것이 지속적인 개선에 대한 응답자들의 심리를 반영하는 것일까요? 아니면 경제의 변동성과 불안정성을 반영하는 것일까요?



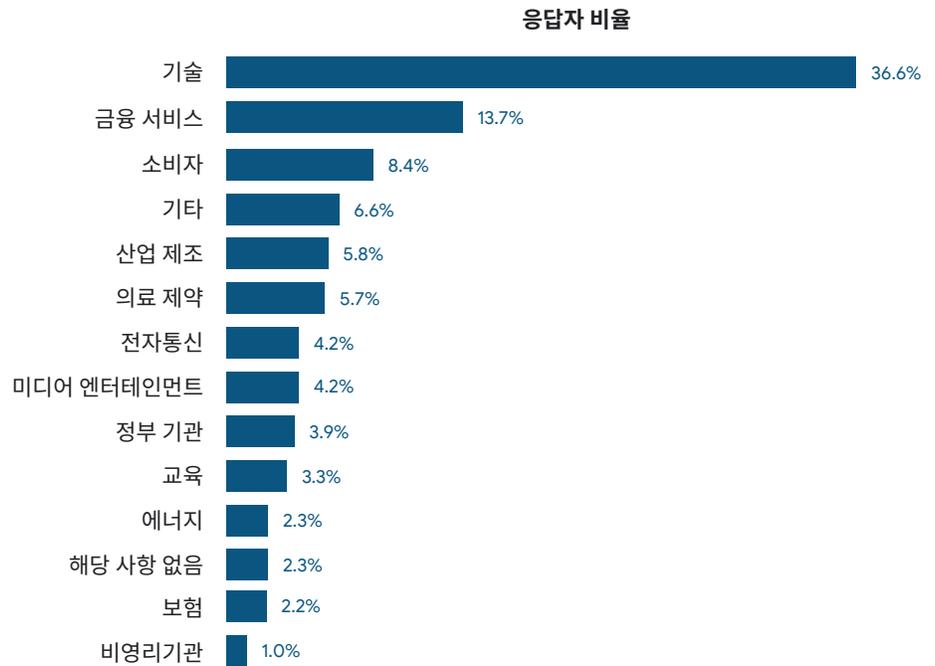
역할

응답자의 72%는 개발 또는 엔지니어링팀(30%), DevOps 또는 SRE팀(18%), IT 운영 또는 인프라팀(8%)에서 근무하는 개인이거나 관리자(16%)로 구성됩니다. 2022년에는 해당 직무에서 근무하는 개인이 응답자의 85%를 차지했습니다. 이러한 4개 직무의 응답자가 줄어들었다는 것은 다양한 직무의 더 많은 개인에게 닿을 수 있었음을 시사합니다. IT 운영 또는 인프라팀(8%)의 비율은 2022년 상승 전환(19%)한 이후 다시 2021년 수준(9%)으로 돌아왔습니다.



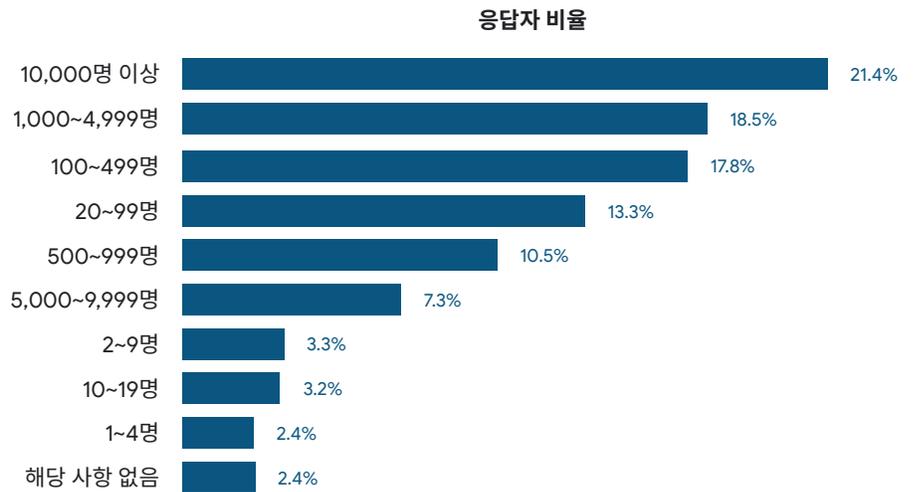
기업통계

업종



직원 수

조직 내 직원은 몇 명인가요?



국가

전 세계 많은 분이 설문조사에 참여해 주시고 계십니다.

감사합니다.



국가					
미국	덴마크	리투아니아	튀니지	방글라데시	과테말라
영국	스위스	태국	우루과이	도미니카 공화국	온두라스
인도	오스트리아	헝가리	아프가니스탄	가나	라트비아
캐나다	케냐	이스라엘	알제리	홍콩 특별행정구	레바논
독일	남아프리카 공화국	베트남	이집트	카자흐스탄	룩셈부르크
오스트레일리아	아르헨티나	아랍에미리트	에스토니아	미얀마	몰디브
브라질	체코	불가리아	아이슬란드	사우디아라비아	몰타
해당 없음	벨기에	크로아티아	이란	소말리아	모리셔스
네덜란드	콜롬비아	에콰도르	나이지리아	수단	몽골
일본	핀란드	인도네시아	페루	우간다	모로코
프랑스	아일랜드	필리핀	슬로바키아	알바니아	네팔
스페인	중국	아르메니아	슬로베니아	바하마	카타르
스웨덴	루마니아	조지아	대한민국	벨라루스	구 유고슬라비아 마케도니아 공화국
이탈리아	싱가포르	그리스	스리랑카	볼리비아	
뉴질랜드	멕시코	말레이시아	안도라	캄보디아	트리니다드 토바고
폴란드	튀르키예	파키스탄	양골라	코스타리카	
노르웨이	우크라이나	러시아 연방	앤티가 바부다	지부티	탄자니아
포르투갈	칠레	세르비아	바레인	엘살바도르	짐바브웨

업무 방식

고용 상태

응답자의 88%가 정규직 직원입니다. 응답자의 10%는 계약직 직원입니다. 일부 계약직 직원은 정규직 직원과 사뭇 다른 경험을 하는 것으로 나타납니다.

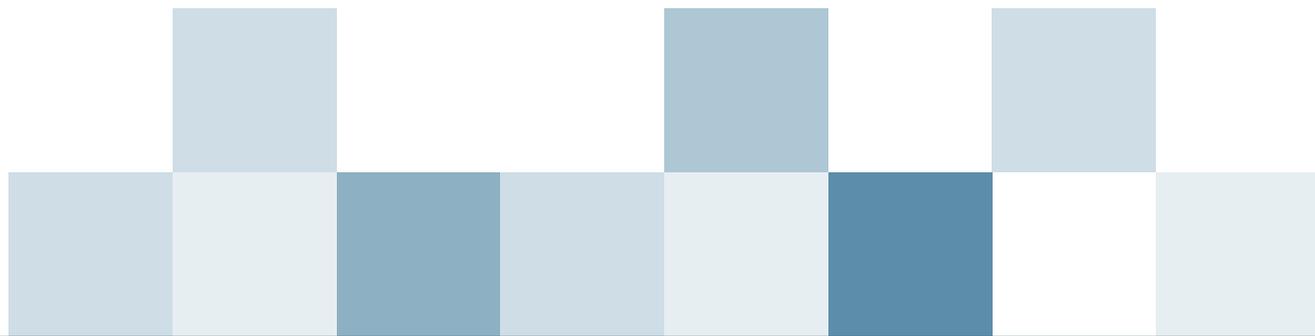
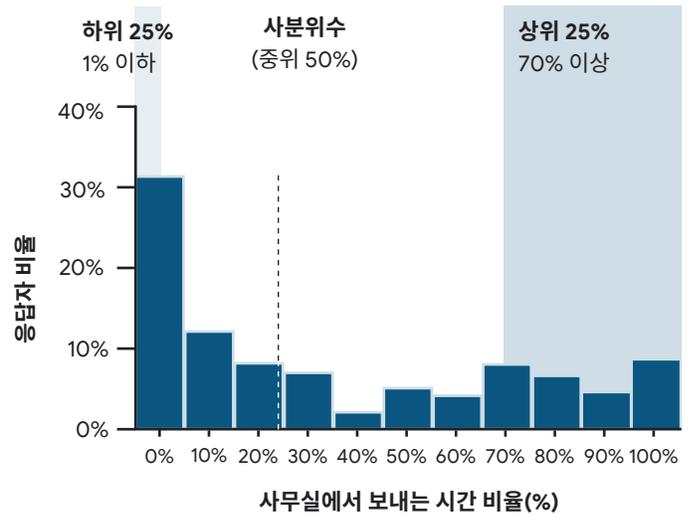
계약	응답자 비율
정규직 직원	88%
상근 계약직	8%
시간제 정규직	2%
시간제 계약직	2%

*작업 중인 주요 애플리케이션 또는 서비스를 소유한 조직에서 본인의 고용 상태를 가장 잘 설명하는 것은 무엇인가요?

경험이 다른 것은 팀에 얼마나 잘 어우러지는지에 기인할 수 있습니다. 몇몇 계약직 직원은 함께 일하는 팀에 잘 섞인다고 보고합니다. 이는 매일 팀원들과 긴밀히 협조하며 자신과 정규직 직원 간 차이가 무시해도 될 정도로 작다고 생각한다는 것을 의미합니다. 계약직 직원 70%는 자신이 팀에 잘 섞인다는 문장에 매우 동의 또는 동의한다고 응답했습니다.

위치

올해에는 사무실 복귀에 대한 압력을 받는 상황에서도 여전히 재택근무가 많은 근로자에게 현실이라는 응답 패턴이 나타났습니다. 33%에 가까운 응답자가 거의 집에서만 근무합니다(사무실 근무 시간이 5% 미만). 응답자의 63%는 사무실보다 집에서 일하는 시간이 더 많습니다. 나머지 응답자의 경우 하이브리드 업무가 가장 일반적인 방식이었습니다. 이는 응답자의 75%가 사무실에서 근무하는 시간이 70% 미만이라는 사실을 시사합니다. 사무실에 대한 애착이 강한 사람은 많지 않았습니다. 겨우 9%의 응답자만이 95% 이상의 시간을 사무실에서 보냅니다.



모델

소개

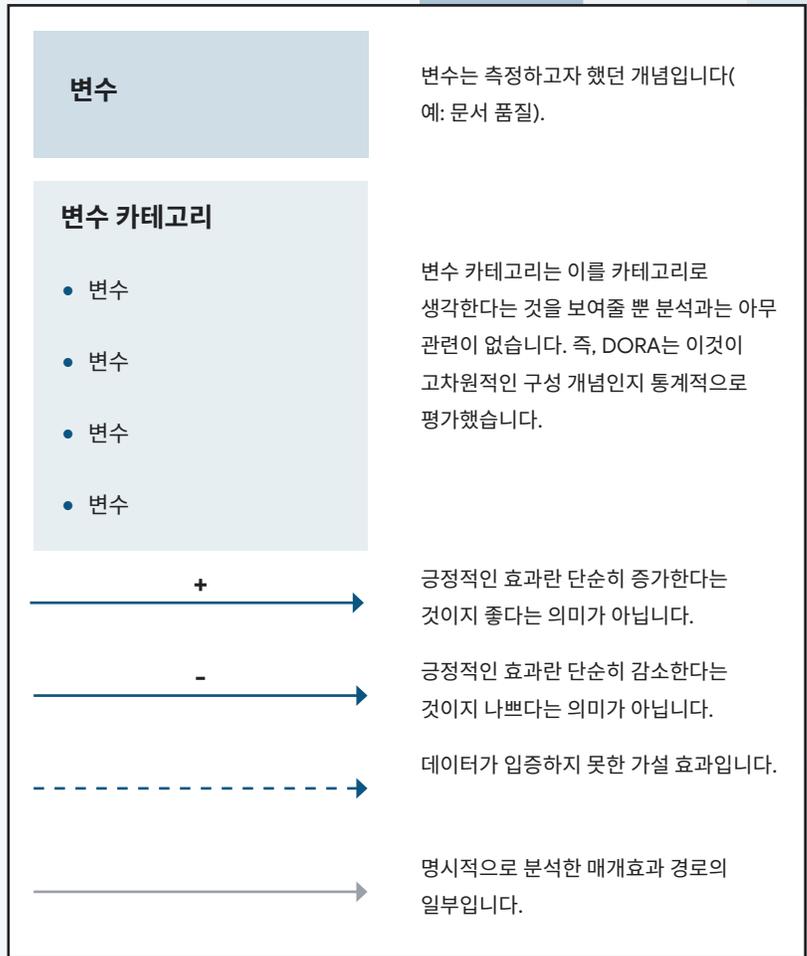
과거에는 하나의 거대한 모델을 만들었습니다. 올해는 다양한 이유로 그 거대한 모델을 여러 개의 모델로 나누기로 결정했습니다.

- 거대한 모델은 금방 통제할 수 없게 될 수 있습니다. 추가되는 모든 변수는 모델이 기능하는 방식을 변화시킵니다. 이는 부정확한 추정으로 이어질 수 있으며 변화의 이유를 찾기 어렵게 만듭니다.
- 올해에는 섹션별로 가설을 세웠습니다. 따라서 각 섹션에 대한 모델을 만드는 것이 타당합니다.
- X가 Y에 미치는 영향을 추정하는 데 있어 거대 모델의 이점은 명확하지 않습니다. X가 Y에 미치는 영향을 이해하기 위해 방향성 비순환 그래프를 사용하여 모델에 포함해야 할 공변량과 포함하지 말아야 할 공변량을 파악했습니다.
- 올해 DORA에서 다른 가설의 수는 독자가 거대 모델을 이해하기 매우 어렵게 만들 것입니다. 아래의 모든 시각화를 하나로 결합한다고 생각해 보세요.



이러한 다이어그램을 읽는 방법

이 다이어그램을 읽는 방법을 배우면 수많은 정보를 전달하는 효율적인 도구로 활용할 수 있을 것입니다.



경고: 모델은 일반 합산입니다.

일부 변수는 읽기 쉽도록 분류했습니다. 이와 같은 분류 전략은 변수 카테고리로 이동하거나 변수 카테고리에서 이동하는 화살표를 일반적인 결과 패턴으로 만들 수 있지만 카테고리의 모든 변수에 해당하지는 않습니다. 예를 들어 지식 공유는 대부분의 주요 성과에 긍정적인 영향을 미칩니다. 따라서 더하기 기호(+)가 있는 화살표를 지식 공유에서 주요 성과 변수 카테고리로 이어지게 그림니다. 그러나 지식 공유는 소프트웨어 배포 실적에는 긍정적인 영향을 미치지 않습니다. 세부정보를 확인하려면 관련 장을 참조하세요.

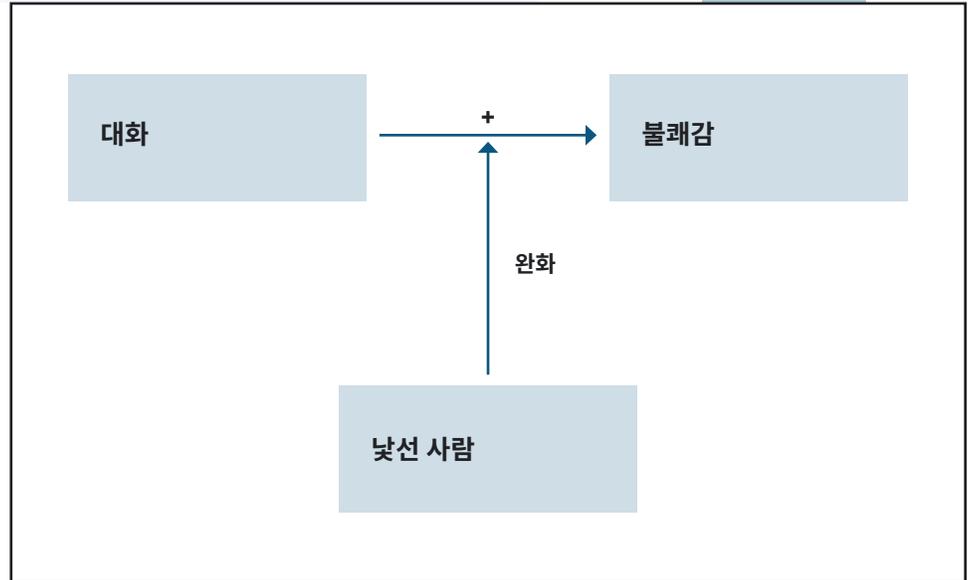
조절효과 예시

조절효과는 통계 분야에서는 쉽게 이해하기 어려운 개념이지만 현실에서는 '상황에 따라 다르다'라는 의미입니다. 이 보고서의 맥락에서 조절효과의 개념을 명확히 하기 위해 간단히 예를 들어보겠습니다.

커브 유어 엔수지애즘 시즌 3에서 래리 데이비드는 이렇게 말합니다 “난 내가 아는 사람과 이야기하는 것이 싫어. 하지만 낯선 사람과는 문제없지.” 이 말을 간단한 다이어그램으로 만들어 논의해 보겠습니다.

“난 내가 아는 사람과 이야기하는 것이 싫어, 하지만 낯선 사람과는 문제 없지.”

래리 데이비드



이 다이어그램에서 대화는 래리의 불쾌감에 긍정적인 영향을 미칩니다. 여기서 긍정적이라는 것은 증가한다는 의미일 뿐 반드시 좋다는 건 아닙니다. 이는 대화와 불쾌감 사이의 검은색 실선과 불쾌감을 가리키는 화살표를 표현됩니다. 이 화살표는 인과관계의 흐름이 대화에서 불쾌감으로 이동한다는 것을 보여줍니다. 여기서 알 수 있는 것은 대화가 래리의 불쾌감을 유발하기 쉽다는 것입니다.

두 번째로 주목해야 할 것은 낯선 사람(여기에서는 불리언 낯선 사람 예/아니요로 표현)이 다른 변수를 가리키지 않는다는 점입니다. 대신 효과(화살표)를 가리킵니다. 이는

낯선 사람이 변수를 수정하지는 않지만 효과는 수정한다는 것을 의미합니다. 그러한 이유로 낯선 사람에게서 출발한 화살표가 다른 변수가 아닌 다른 화살표를 향하도록 그린 것입니다. 래리가 낯선 사람과 이야기하는지 여부가 대화가 불쾌감에 미치는 효과에 영향을 준다는 것입니다. 다시 말하면 대화가 불쾌감에 미치는 효과는 래리가 대화하고 있는 사람이 낯선 사람인지 여부에 달려 있다는 뜻입니다. 그 사람이 낯선 사람이라면 대화의 효과는 래리에게 '문제가 없는' 것입니다. 래리가 대화할 때 낯선 사람의 존재가 불쾌감을 완화하는 것이라고 말할 수 있습니다.

무언가가 다른 무언가를 모더레이션할 수 있는 방법에는 여러 가지가 있습니다.

- **강화** - 긍정적인 효과를 더 긍정적으로 만들고 부정적인 효과를 더 부정적으로 만듭니다.
- **약화** - 효과를 약하게 만듭니다.
- **완화** - 긍정적인 효과를 덜 긍정적으로 만들고 부정적인 효과를 덜 부정적으로 만듭니다.
- **반전** - 긍정적인 효과를 부정적으로 만들고 부정적인 효과를 긍정적으로 만듭니다.
- **수정** - 효과가 변화하는 것으로 패턴을 한 마디로 정의하기 어렵습니다. 이와 같은 현상은 원인으로서는 카테고리형 범주가 있을 때 발생합니다. 예를 들어 특정 산업에서는 경제적 변동성과 같은 여러 조건에 따라 다른 상황이 발생할 수 있습니다.

매개효과 예시

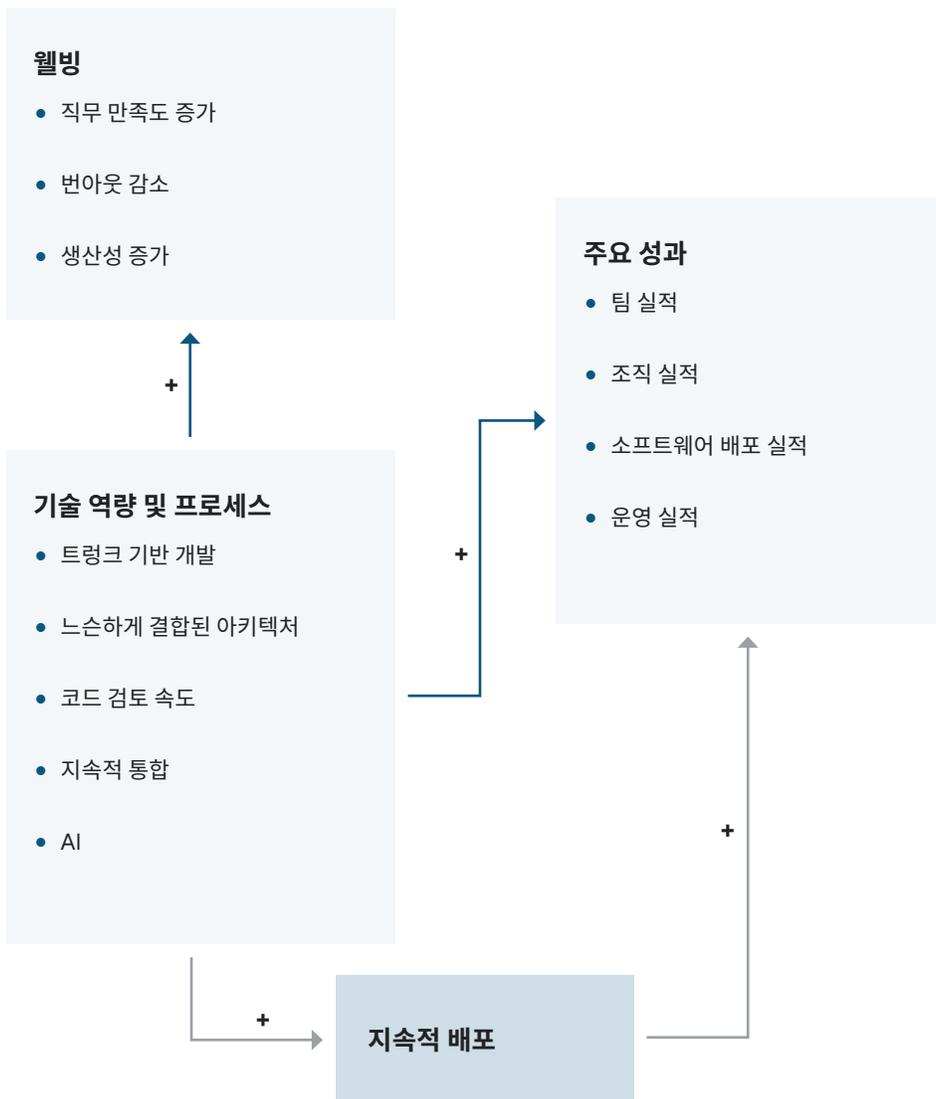
조절효과와 마찬가지로 매개효과도 크게 고려합니다. 매개효과와 근간은 *왜(이유)* 또는 *어떻게(방법)*입니다. 태양이 식물을 더 크게 만드는 이유는 무엇일까? 딸기를 먹으면 단맛을 느끼는 이유는 무엇일까? 가속기 페달은 어떻게 차를 움직이게 만들까? 진통제는 어떻게 불편함을 줄여줄까? 통계에서 매개효과에 대해 테스트할 수 있습니다.¹이것을 말로 풀어내면 'X가 Y에 미치는 영향은 M으로 설명되거나 부분적으로 설명될 수 있습니다.'라는 의미입니다. '태양이 식물의 키에 미치는 영향은 광합성으로 설명될 수 있습니다.'라는 문장을 예로 들 수 있습니다.



¹<https://lavaan.ugent.be/tutorial/mediation.html>

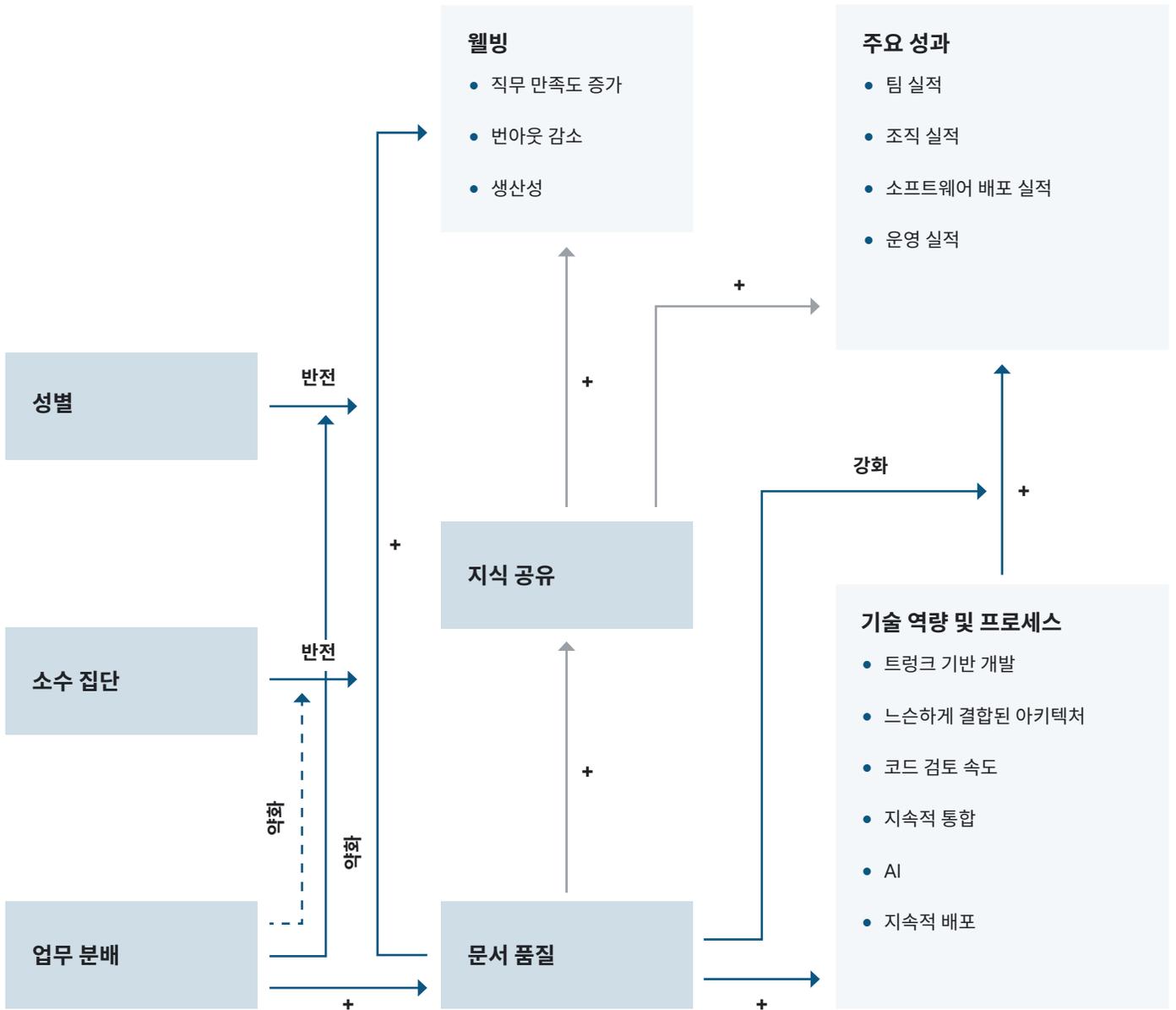
3장의 모델 기술 역량으로 실적 개선 기대

기술 역량과 프로세스는 웰빙과 주요 성과에 긍정적인 효과를 줍니다. 주요 성과에 대한 기술 역량의 효과는 지속적 배포에 따라 부분적으로 매개 역할을 합니다. 그리고 이러한 기술 역량은 주요 성과에 다운스트림 영향을 미치는 지속적 배포 환경을 만듭니다.



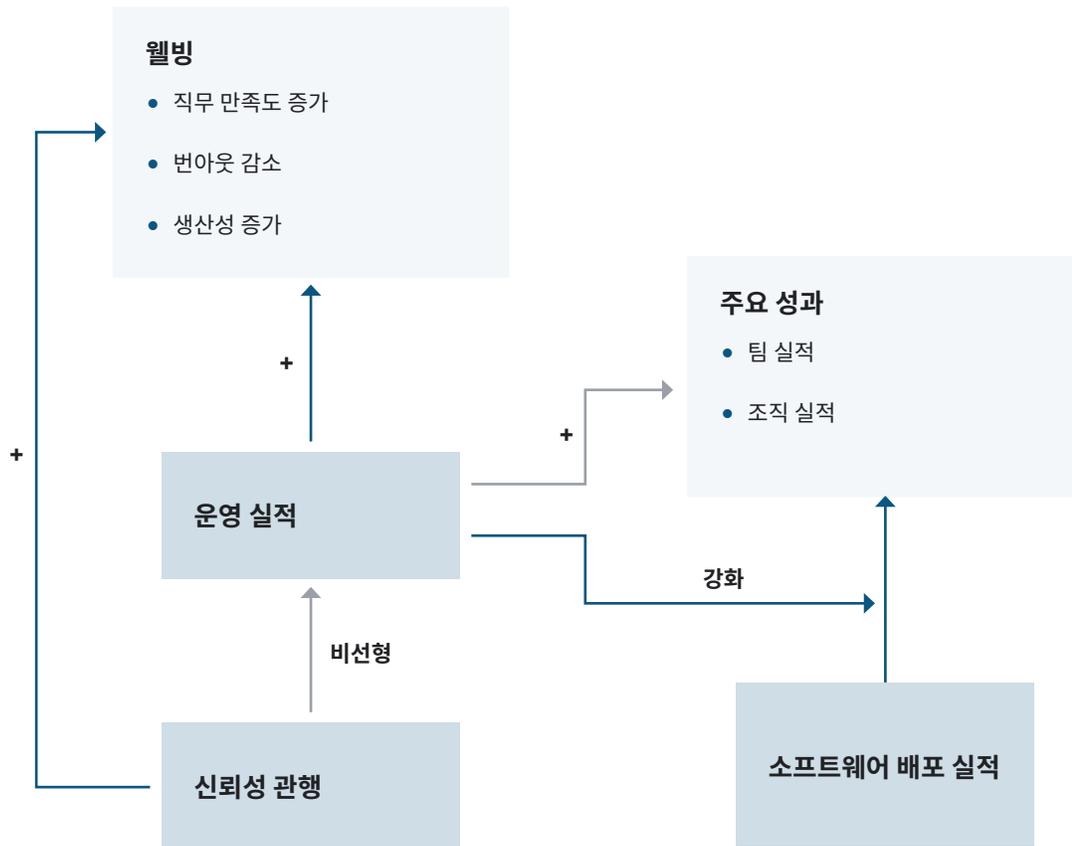
4장의 모델 모든 업무의 기초가 되는 문서

간단히 말해 고품질 문서는 웰빙과 주요 성과에 긍정적인 영향을 미칩니다. 고품질 문서는 지식 공유가 가능한 환경을 만들기 때문이기도 합니다. 고품질 문서는 팀이 기술 역량을 키우고 프로세스를 수립하는 데에도 도움을 줍니다. 또한 기술 역량과 프로세스가 주요 성과에 더 큰 영향을 미치도록 도와줍니다. 마지막으로 문서 품질만으로는 모두에게 더 나은 웰빙을 가져다 주지 않습니다. 이러한 복잡한 연구 결과에 대해 분석하려면 해당 섹션을 읽어보는 것이 좋습니다.



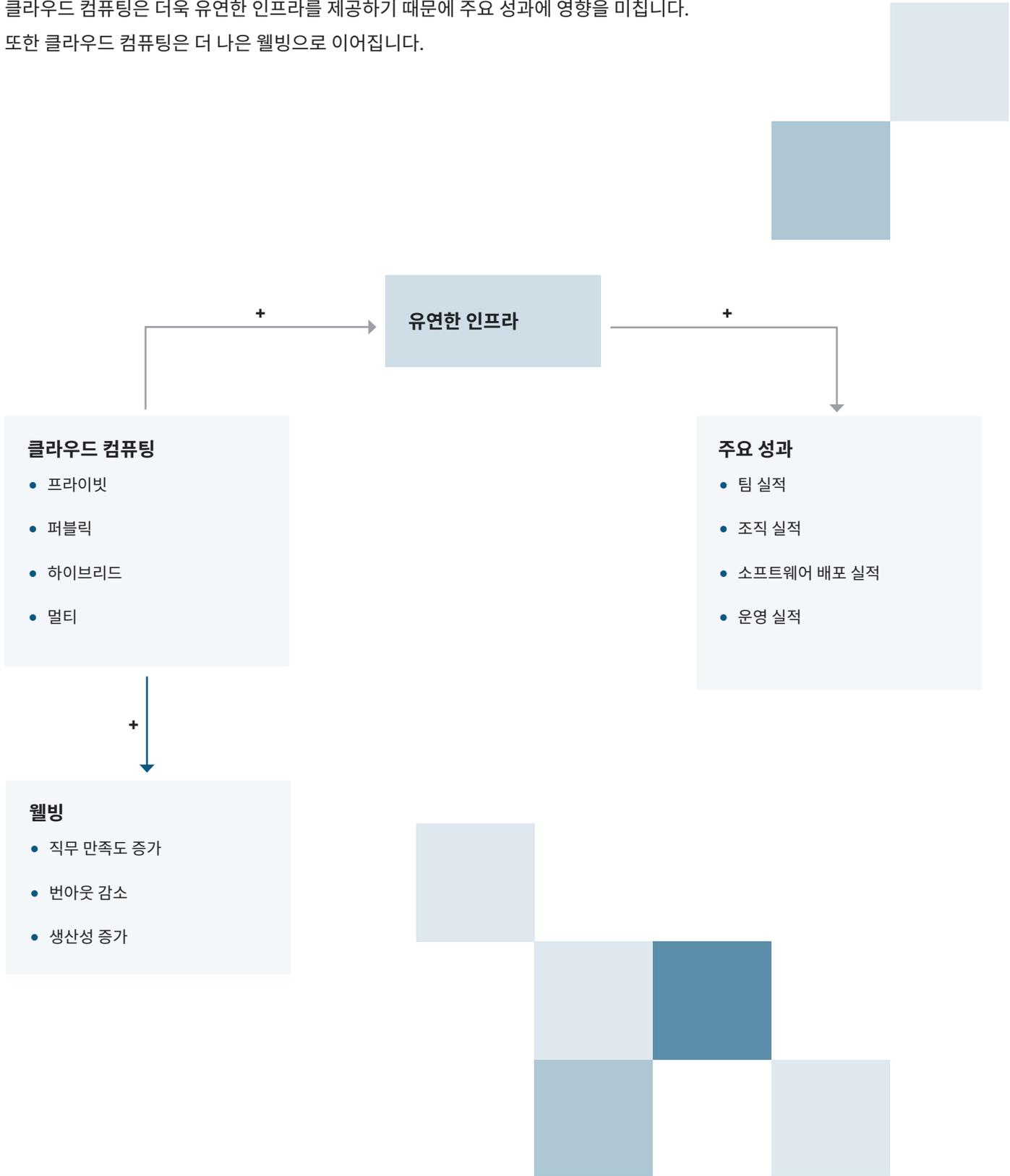
5장의 모델 신뢰성으로 실적 개선

여기에서는 운영 실적의 중심 역할을 알아봅니다. 운영 실적은 웰빙과 주요 성과를 개선할 뿐만 아니라 소프트웨어 배포 실적의 효과를 강화하기도 합니다. 신뢰성 관행은 운영 실적과 비선형적인 관계에 있습니다. 이와 관련된 자세한 내용을 알아보려면 이 장을 참고하시기 바랍니다.



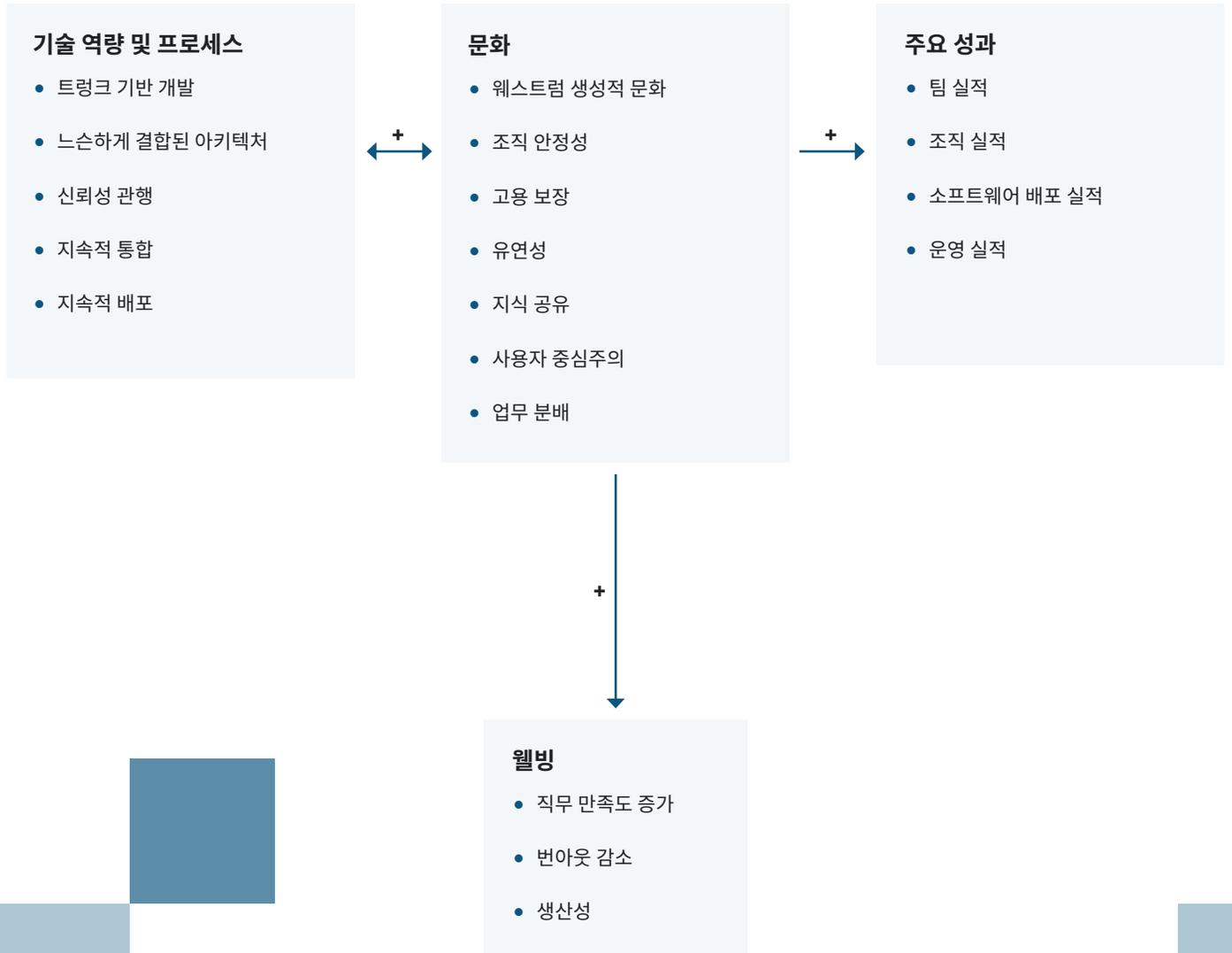
6장의 모델 유연한 인프라는 성공의 핵심

클라우드 컴퓨팅은 더욱 유연한 인프라를 제공하기 때문에 주요 성과에 영향을 미칩니다.
또한 클라우드 컴퓨팅은 더 나은 웰빙으로 이어집니다.



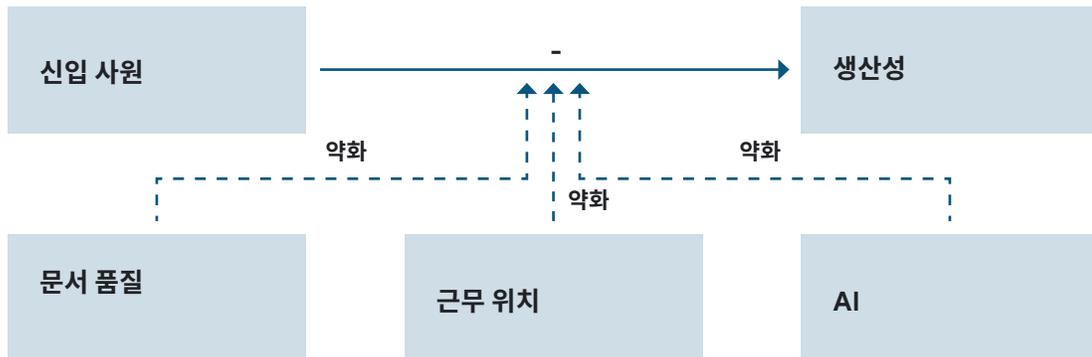
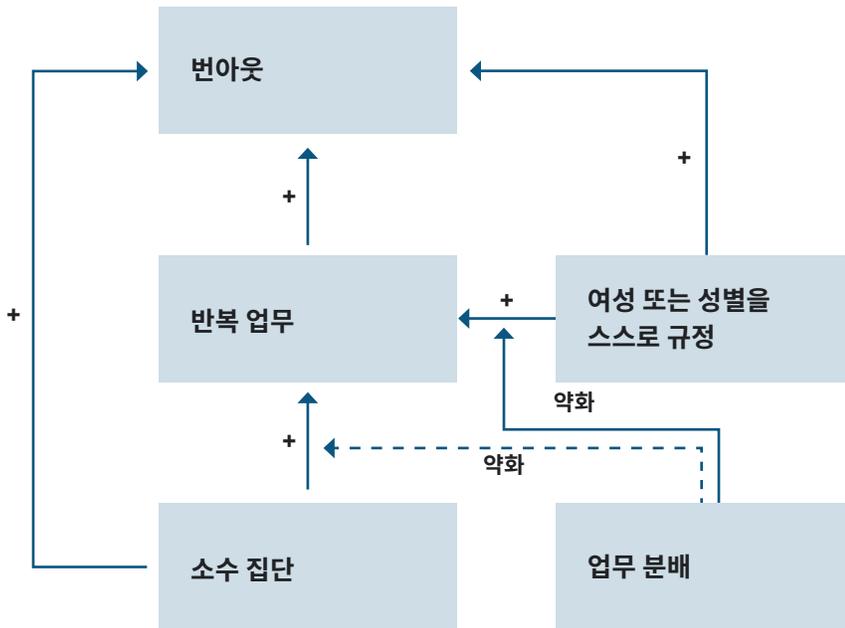
7장의 모델 문화에 대한 투자가 없다면 무용지물

이 다이어그램에서 문화는 많은 것의 중심이라는 사실을 확인할 수 있습니다. 문화는 기술 역량, 주요 성과, 웰빙과 양의 관계에 있습니다.



8장의 모델 자신을 파악하는 것이 중요한 이유

이 섹션에는 2가지 모델이 있습니다. 첫 번째 모델에서는 소수 집단에 속하는 것으로 식별되는 사람과 남성으로 식별되는 사람이 더 많은 번아웃을 경험하는 이유와 상황을 탐구합니다. 다른 모델은 문서 품질, 근무 위치 또는 AI가 신입 사원의 생산성을 높이는 데 도움을 주는지 여부를 탐구합니다.



추가 자료

DORA Community에 참여하여 소프트웨어 배포 및 운영 성과 개선을 주제로 논의하고 배우며 공동작업해 보세요.

[DORA.community](https://dora.community)

DORA DevOps 빠른 점검 수행하기 <https://dora.dev/quickcheck>

소프트웨어 배포 및 조직 실적을 향상시키는 기술, 프로세스, 문화적 역량을 살펴보세요. <https://dora.dev/devops-capabilities/>

SRE에 대한 리소스 찾기

<https://sre.google>

<https://goo.gle/enterprise-roadmap-sre>

도서 읽어보기: *Accelerate: The science behind devops: Building and scaling high performing technology organizations*. IT Revolution. <https://itrevolution.com/product/accelerate/>

SPACE 프레임워크를 사용해 팀을 위한 적절한 측정항목 집합을 찾아보세요. 'The SPACE of Developer Productivity: There's more to it than you think(개발자 생산성의 SPACE: 생각보다 많은 고려사항)'. <https://queue.acm.org/detail.cfm?id=3454124>

최신 코드 검토에 대해 여러 연구가 수행되었습니다. 살펴볼 만한 몇 가지 보고서를 소개합니다.

- 'Expectations, Outcomes, and Challenges of Modern Code Review(최신 코드 검토에 대한 기대, 결과, 과제)'. <https://dl.acm.org/doi/10.5555/2486788.2486882>

- 'Code Reviews - From bottlenecks to Superpowers (코드 검토: 병목 현상 극복부터 팀의 초능력 발휘까지)'. <https://learning.acm.org/techtalks/codereviews>
- 'Modern Code Review- A Case Study at Google (최신 코드 검토 - Google 우수사례)'. <https://dl.acm.org/doi/10.1145/3183519.3183525>
- 'The Choice of Code Review Process: A Survey on the State of the Practice(코드 검토 프로세스 선택: 실무 현황 조사)'. https://link.springer.com/chapter/10.1007/978-3-319-69926-4_9
- 'Investigating the effectiveness of peer code review in distributed software development based on objective and subjective data(객관적, 주관적 데이터를 기반으로 분산 소프트웨어 개발 작업에서 피어 코드 검토의 효율성 조사)'. <https://jserd.springeropen.com/articles/10.1186/s40411-018-0058-0>

도서 읽어보기: *The No Club: Putting a Stop to Women's Dead-End Work*. 사이먼, 슈스터. <https://www.simonandschuster.com/books/The-No-Club/Linda-Babcock/9781982152338>

이전 Accelerate State of DevOps 보고서를 포함한 DORA 연구 프로그램의 간행물. <https://dora.dev/publications/>

연구 및 보고서에 관한 자주 묻는 질문. <http://dora.dev/faq>

정오표 - 이 보고서에 대한 변경사항, 수정사항, 설명을 읽고 제출하세요. <https://dora.dev/publications/errata>

부록

소프트웨어 배포 실적의 측정 방법 조정

올해에는 변경 실패 및 실패에서 복구하는 과정을 평가하는 방식에 변화를 주었습니다.

응답자가 **변경 실패율**을 보고하는 방식을 바꿨습니다. 예년에는 응답자에게 6가지 옵션(0~15%, 16~30% 등)이 제시되었습니다. 올해에는 응답자에게 0%부터 100%까지의 값을 자유롭게 선택할 수 있는 슬라이더를 제시했습니다.

이러한 변경에는 2가지 이유가 있습니다.

- 변경 실패율은 항상 소프트웨어 배포 실적의 다른 3가지 측정값과 약간 다르게 움직였습니다.¹ 이는 부분적으로 버킷 크기 때문일 수 있다는 이론을 세웠습니다. 보다 정확하게 답변할 수 있도록 하면 더 나은 통계 결과가 산출될 수 있습니다. DORA는 이 이론을 검증할 수 있었습니다.
- DORA는 많은 팀이 약 10년 전 이 연구를 시작했을 때보다 현재 변경 실패율을 더 잘 이해하고 있다는 가설을 세웠습니다. 이 가설을 검증할 방법은 없습니다.

올해에는 2가지 방법으로 오류에서 복구하는 과정에 대해 질문했습니다.

- 이전에는 다음과 같이 질문했습니다. "작업 중인 주요 애플리케이션 또는 서비스의 경우 사용자에게 영향을 미치는 서비스 이슈 또는 결함(예: 예기치 않은 서비스 중단, 서비스 손상)이 발생할 때 서비스를 복원하는 데 시간이 보통 얼마나 걸리나요?"
- 올해에는 다음과 같은 수식 어구를 질문에 추가했습니다(여기서는 굵게 처리했지만 설문조사에서는 굵게 처리하지 않음). "작업 중인 주요 애플리케이션 또는 서비스의 경우 **프로덕션 또는 사용자에 대한 출시에 변경사항을 적용한 이후 서비스 저하(예: 서비스 장애 또는 서비스 중단으로 이어짐)가 발생했다면 서비스를 복원하고 이후에 필요한 조치(예: 핫픽스, 롤백, 향후 수정, 패치 적용)를 수행하는 데 시간이 보통 얼마나 걸리나요?"**

¹ 폴스그렌, N., 험블, J., 킴, G. *Accelerate: The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations*(가속화: 린 소프트웨어 및 DevOps의 과학: 실적이 우수한 기술 조직 구축 및 확장)(IT Revolution Press, 2018), 37~38.

복구 시간을 묻는 이전 방식의 질문을 통해서도 소프트웨어 변경으로 시작된 실패와 지진으로 인한 데이터 센터 서비스 중단으로 시작된 실패를 구분하기 어려웠습니다. 더욱 정확한 언어를 사용하면 서로 유사한 실패 유형을 비교할 수 있고 해당 언어가 소프트웨어 배포 실적의 다른 3가지 측정값과 통계적으로 더 잘 맞을 것이라는 가설을 세웠습니다.

이제 '배포 실패 복구 시간'이라는 용어를 사용해 과거에 사용했던 '복원 시간(MTTR)'이라는 좀 더 일반적인 용어와 측정값을 구분하고 있습니다. MTTR에서 'M'이 평균값을 의미하는지 중앙값을 의미하는지 불분명하여 커뮤니티에 혼란을 가져왔었습니다. 또한 실패에서 교훈을 찾으려는 복원력 엔지니어링 분야 등의 실무자는 MTTR을 학습 및 개선을 위한 신뢰할 수 있는 측정값으로 보지 않습니다.²

새롭게 추가된 질문과 배포 실패 복구 시간이라는 새로운 측정항목이 소프트웨어 배포 실적을 측정하는 데 더 잘 어울립니다.

² 'Moving Past Simple Incident Metrics: Courtney Nash on the VOID'
<https://www.infoq.com/articles/incident-metrics-void/>



비교에 사용된 수학

이 보고서 전반에는 특정 변수에 대한 측정값이 높으면 다른 항목의 점수가 N배 높아진다는 통계가 있습니다. 이것은 무엇인가요? 이것은 무엇과 관련이 있을까요? 다음 이러한 수식을 만들기 위한 방법입니다.

1. 잠재적인 편향 경로를 고려하는 회귀 기법을 사용해 평가된 모델에서 수식을 만듭니다.

$$\text{행복} \sim 5.64 + 0.19 * \text{햇빛} + 0.14 * \text{온도}$$

2. 햇빛과 같은 관심 변수에는 비교를 위한 높은 값과 낮은 값을 찾습니다.

$$\text{평균값} = 6.3, \text{표준편차} = 1.4$$

$$\begin{aligned} \text{높은 햇빛} &= \text{평균값보다 표준편차 1 높음} = \\ &\text{평균값} + \text{표준편차} = 6.3 + 1.4 = 7.7 \end{aligned}$$

$$\begin{aligned} \text{낮은 햇빛} &= \text{평균값보다 표준편차 1 낮음} = \\ &\text{평균값} - \text{표준편차} = 6.3 - 1.4 = 4.9 \end{aligned}$$

3. 공변량을 위한 평균값을 계산합니다. 이 예시에서는 온도이며 값은 0.24(표준화된 온도)입니다.
4. 높은 햇빛과 낮은 햇빛에는 1단계의 수식을 채웁니다. 수식에서 하나의 숫자만 다르다는 걸 확인할 수 있습니다. 이것이 한 가지 관심 변수만 다르게 하게 나머지는 모두 동일하게 유지하는 방법입니다.

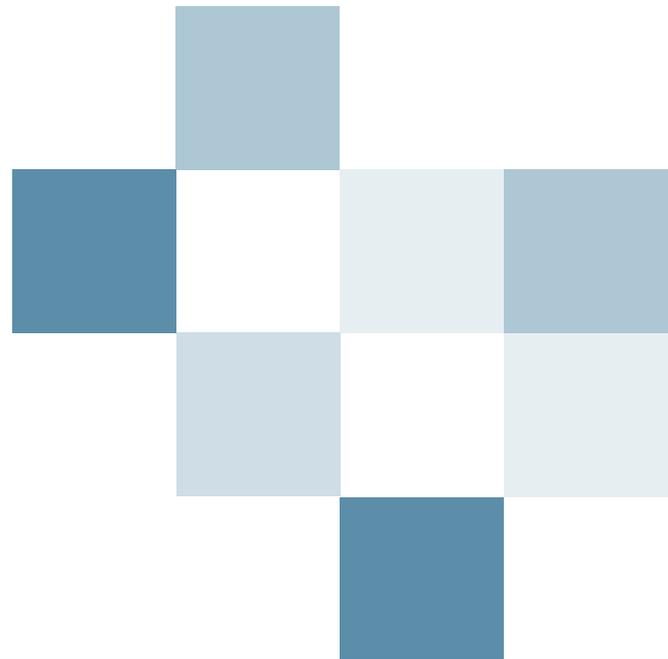
$$\begin{aligned} \text{예상되는 높은 햇빛 행복감} &= 5.64 + 0.19 * 7.7 + \\ &0.14 * 0.24 = 7.1 \end{aligned}$$

$$\begin{aligned} \text{예상되는 낮은 햇빛 행복감} &= 5.64 + 0.19 * 4.9 + \\ &0.14 * 0.24 = 6.6 \end{aligned}$$

5. 비율을 계산합니다.

$$\begin{array}{r} \text{예상되는 높은 햇빛} \\ \text{행복감} \\ \hline \text{예상되는 낮은 햇빛} \\ \text{행복감} \end{array} = \frac{7.1}{6.6} = 1.1\text{배}$$

6. 이 비율은 높은 수준의 햇빛이 낮은 수준의 햇빛보다 10% 높은 행복감을 준다는 결과를 시사합니다.



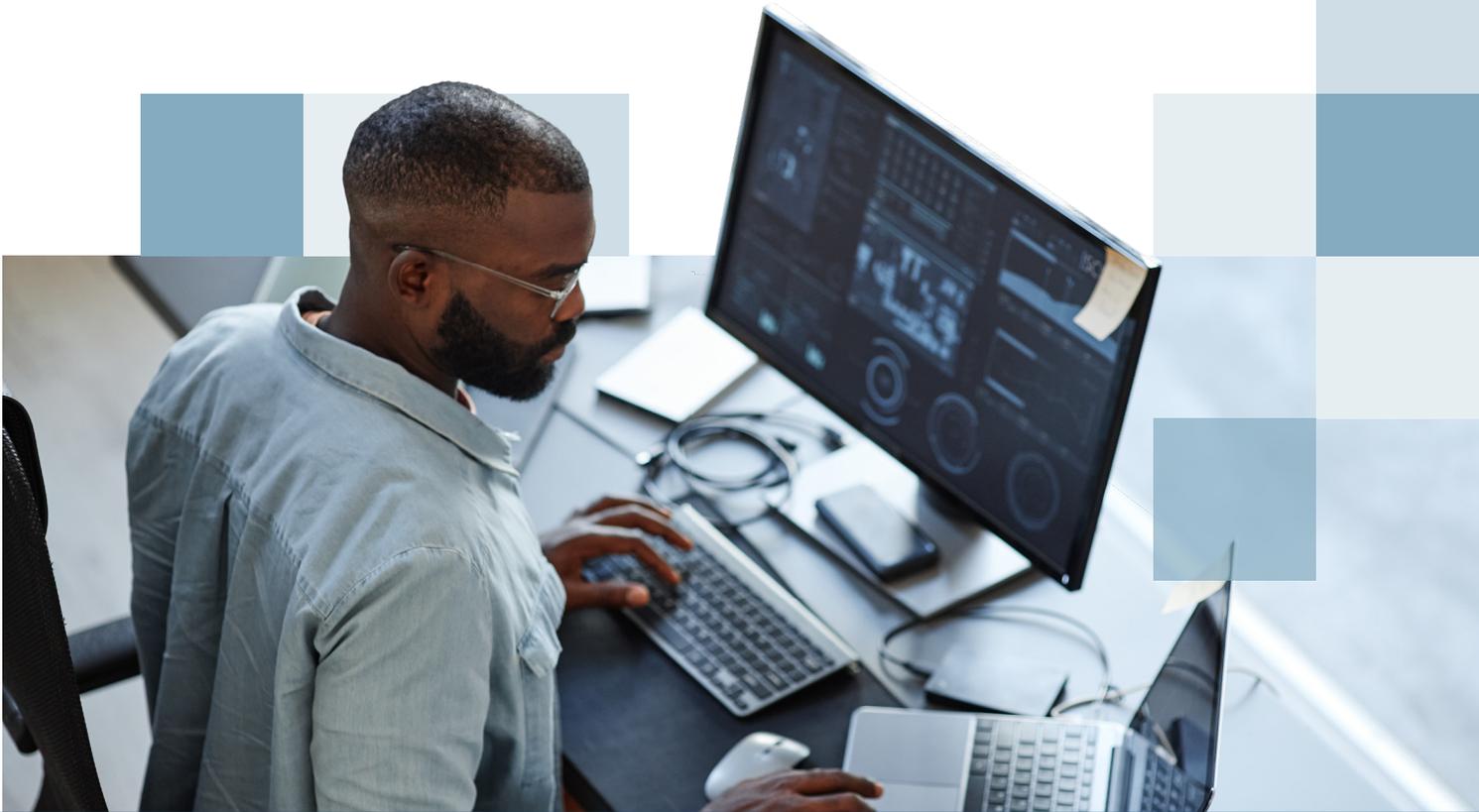
'시뮬레이션'이란?

데이터는 DORA에서 직접 만드는 것이 아닙니다. DORA는 베이저안 통계를 사용하여 '다른 매개변수 값이 나타날 예상 빈도'를 포착하는 **사후 확률**을 계산합니다.³ '시뮬레이션'은 데이터가 주어진 매개변수(평균값, 베타 가중치, 시그마, 절편 등)에 대해 가장 신뢰할 수 있는 값을 찾기 위해 이 사후 확률에서 1,000회 이상 데이터를 추출하는 과정입니다.⁴ 사후 확률을 0.1, 0.7, 0.5, 1과 같은 매개변수 값으로 가득 찬 버킷이라고 가정해 보겠습니다. 버킷 안에서 각각의 값은 사후 확률에 비례하여 존재합니다. 따라서 최댓값 가까이에 있는 값이 최솟값 가까이에 있는 값보다 훨씬 더 일반적입니다.⁴ 이 모든 과정은 시뮬레이션을 사용해 데이터의 가능한 해석을 탐구하고 불확실성 수준을 파악하는 것과 같습니다. 각각의

시뮬레이션은 데이터 외에는 아무 것도 모르는 AI와 정보에 입각한 추측으로 빈칸(매개변수)을 채우려는 몇 가지 규칙으로 생각해 볼 수 있습니다. 이 과정을 4,000번 수행하면 4,000개의 AI가 주어진 매개변수에 대해 추측한 값을 얻게 되는 것입니다. 이러한 추측으로 많은 것을 알 수 있습니다. 평균적인 추측값이 무엇인지, 이러한 추측의 89%가 어떤 값에 속하는지, 얼마나 많은 추측값이 특정 수준 이상인지, 이러한 추측값에 얼마나 많은 변수가 있는지 등을 알게 될 수 있습니다. 여러 모델에 걸친 결합 추측(시뮬레이션)과 같은 흥미로운 작업도 수행할 수 있습니다. 수많은 선 또는 잠재적 값의 분포가 포함된 그래프는 데이터를 고려할 때 가장 타당한 것이 무엇이며 얼마나 많은 불확실성이 존재하는지 표현하는 것입니다.

³ 메셀리스, 리차드. Statistical rethinking: A Bayesian course with examples in R and Stan(통계적 재검토: R 및 Stan의 예를 사용한 베이저안 방법 과정). Chapman and Hall/CRC, 2018, 페이지 50

⁴ 메셀리스, 리차드. Statistical rethinking: A Bayesian course with examples in R and Stan(통계적 재검토: R 및 Stan의 예를 사용한 베이저안 방법 과정). Chapman and Hall/CRC, 2018, 페이지 52



Google LLC의 'Accelerate State of DevOps 2023'은 CC BY-NC-SA 4.0에 따라 라이선스가 부여됩니다.