데브옵스(DevOps) 팀이 분산된 환경에서 가시성을 확보하지 못하는 이유

툴의 무질서한 확산이 어떻게 팀의 업무 속도를 저하시키고 시스템의 상태를 악화시킬까요?

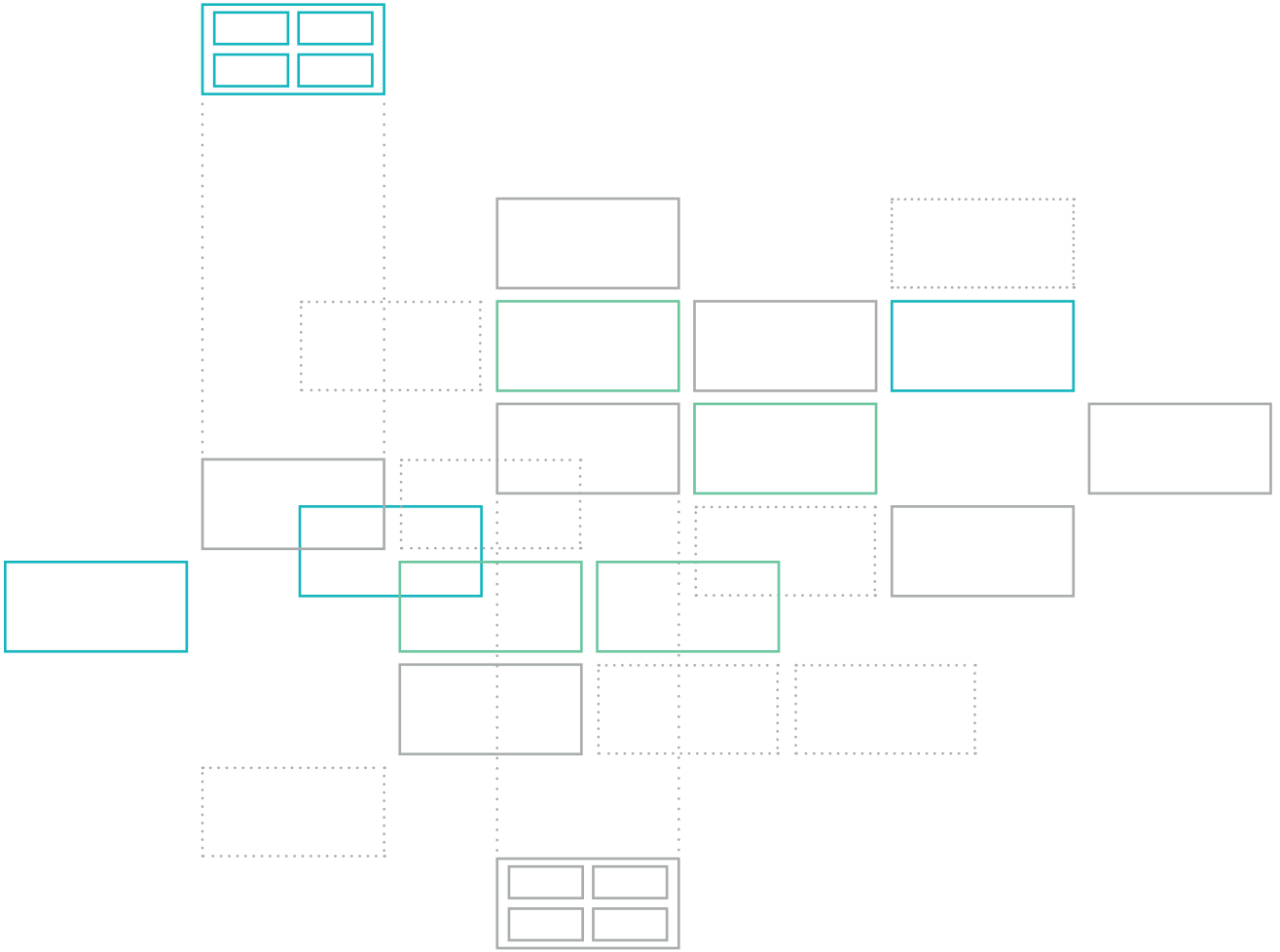
New Relic

변화하는 환경에서는 문맥이 핵심입니다.

팀이 소프트웨어를 배포하고 운영하는 방식은 지난 10년간 빠르게 발전해 왔습니다. 그 결과로 IT 운영자가 관리해야 하는 영역은 크기와 복잡성의 측면에서 급격히 증가하고 있습니다.

인프라의 세계에서, 변화는 한때 부담으로 여겨졌지만, 이젠 경쟁 우위를 위한 기반이 되었습니다.

기업은 인프라와 애플리케이션을 더 빠르고 자주 출시하기 위해 데브옵스 관행을 도입합니다. 속도, 확장성 및 성능을 향상시키기 위해 애플리케이션을 현대화합니다. 클라우드로 이동을 합니다. 마이크로서비스를 도입합니다. 쿠버네티스 같은 컨테이너 오케스트레이션 시스템을 실행합니다.



거기에다 이제 신속하고 끊임없는 변화가 인프라 관리 체계에 추가되었습니다.

소프트웨어 변경, 설정, 알림, 기타 모든 사항이 증가됩니다. 이와 동시에, 문제를 보다 빠르게 감지하고 해결하며 운영 시스템의 안정성과 신뢰성을 보장해야만 하는 부담도 커집니다.

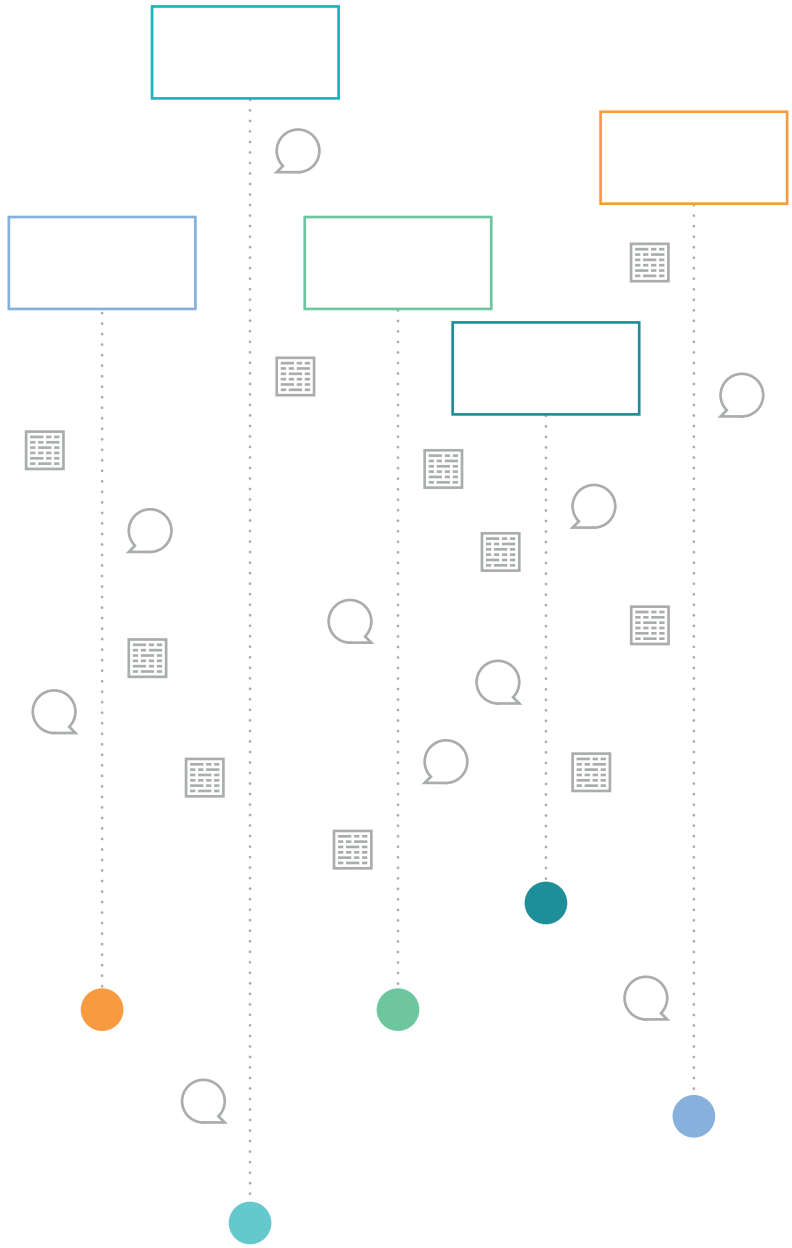
가속화와 확장으로 야기된 복잡성으로 인해 모니터링 전략을 수정해야 할 필요성이 생겼습니다. 대부분의 경우 시스템이 운영 단계에 돌입하기 전까지 어떻게 작동할지 알 수 없기 때문에, 운영 환경을 **관찰할 수 있는** 시스템이 필요합니다.

접근 방식을 바꾸면, 팀이 동적인 시스템들을 효과적으로 관리하는 데 도움이 됩니다.

문제는 개발자용 툴, IT 운영자용 툴, 비즈니스 관리자용 툴, 로그용 툴, 메트릭용 툴, 트레이스용 툴, 온프레미스용 툴, 클라우드용 툴 등, 여러 팀들이 다른 툴을 사용해 스택의 각 부분을 모니터링하고 있다는 것입니다.

물론 각 팀마다 필요에 가장 적합한 툴을 선택했을 것입니다.

하지만 이는 모든 팀들이 더 많은 알림과 텔레메트리 데이터를 처리하고 있으며, 무엇보다도 운영 데이터가 단편적이라는 것을 의미합니다.



무질서한 툴 확산의 문제: 단편적인 관찰성은 관찰성이 아닙니다.

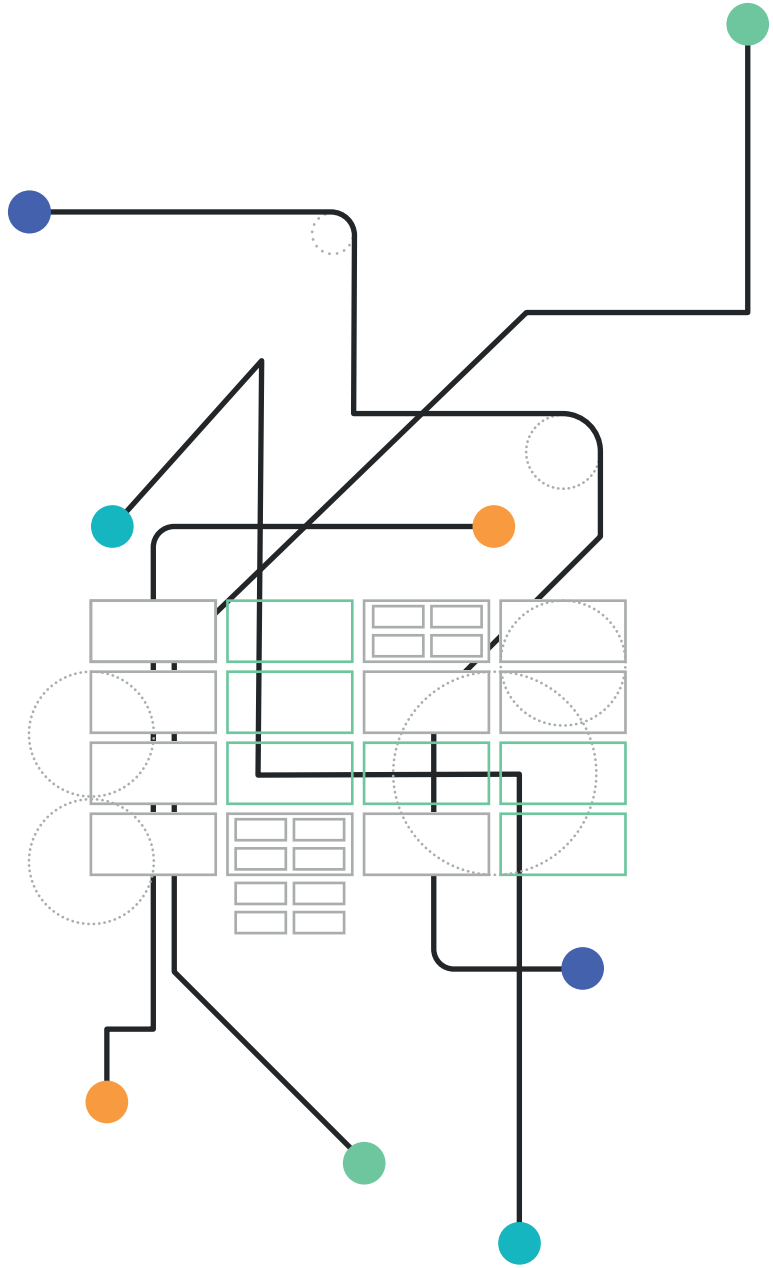
각 툴은 전체 그림의 일부만 보여줍니다. 그러나 그림은 동적이며, 스택의 부분들 간에 경계는 분명하지 않습니다. 애플리케이션이 충돌하면 피해가 확산되기 전에 코드나 인프라에서 무슨 일이 발생했는지 알아내야 하지만, 각 시스템의 구성 요소에 서로 다른 포인트 솔루션을 사용하고 있기 때문에 시간과 비용이 낭비됩니다.

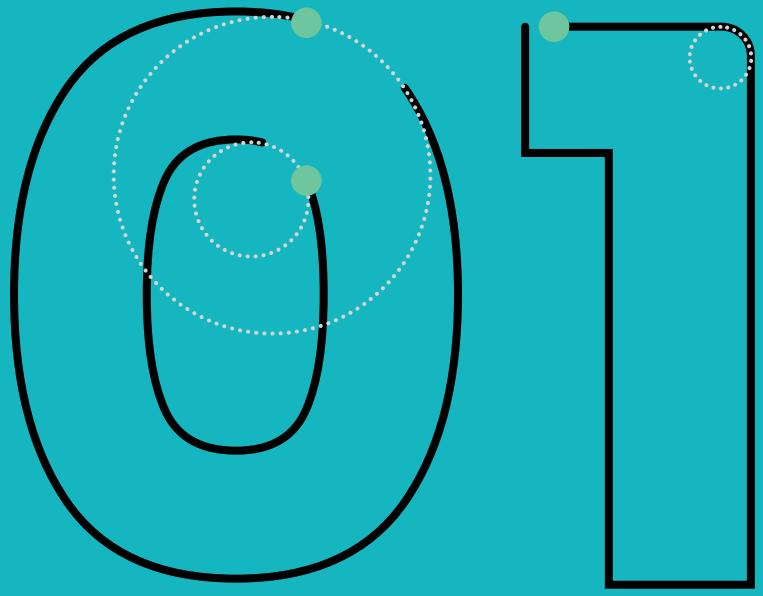
결국 데이터는 사일로에 갇히게 됩니다. 모든 툴이 다른 어휘를 사용해 다른 팀들은 이해할 수 없습니다. 그래서 평균진단시간(MTTD)과 평균장애복구시간(MTTR)이 길어집니다.

비용에는 단순히 금전적인 것만 포함되지는 않습니다. 비즈니스 전체에 폭포 효과를 일으킬 수 있습니다. IT 팀과 운영 팀은 문제 해결에 너무 많은 시간을 소비하다 보니 정작 혁신해 매진할 시간은 없습니다. 팀 간의 연계와 협업이 저하되고, 직원들의 사기가 떨어집니다.

비즈니스에 차질이 생깁니다.

이 가이드에서는 툴의 무질서한 확산이 비즈니스에 어떤 방식으로 피해를 주는지 자세히 알아보겠습니다. 또한 이를 극복하면 어떤 가능성이 열리는지에 대해서도 살펴보겠습니다.





무질서한 툴 확산이 얼마나 큰 문제일까요?

수치만으로는 그 사실을 알 수 없습니다. 그러나 **451 Research의 설문조사**에 따르면, 응답자의 39%가 애플리케이션, 인프라 및 클라우드 환경을 모니터링하기 위해 11~30개의 모니터링 툴을 사용하고 있었으며, 8%는 21~30개의 툴을 사용하고 있었습니다.

이러한 툴 중 대다수는 오픈소스('무료')일 확률이 높지만, 다운타임 비용을 차치하더라도 관련 비용은 빠르게 늘어날 수 있습니다.

이는 직원들의 속도를 저하시킵니다.

무질서한 툴 확산으로 인한 첫 번째 문제는, 하나의 툴에서 다른 툴로 문맥을 전환하는 데 엄청난 시간이 소모된다는 것입니다. 특정 상황에서 몇 초 또는 몇 분 정도로 보이는 문제들이 조직 전체에 분산되어 있다면 훨씬 더 큰 문제가 될 수 밖에 없습니다.

데이터 해상도가 떨어집니다.

IT 스택의 여러 부분을 모니터링하기 위해 다른 툴들을 사용하는 경우, 모든 구성 요소에서 애플리케이션 성능과 시스템 상태 간의 상관 관계를 파악할 수 없기 때문에 환경에 대한 가시성이 부족해 집니다.

관리자 부담이 늘어납니다.

툴 자체는 초기 비용이 없을 수도 있지만, 소프트웨어 라이선스, 내부 리소스, 애드온 모듈, 스토리지, 하드웨어, API 액세스와 관리를 하려면 툴을 설정하고 지속적으로 유지 관리해야 합니다. 단일한 팀 내에서도 많은 문제를 처리해야 하는데 이러한 문제들이 분산된 환경 전체로 확장된다면, 갑자기 비효율성이 뚜렷하게 드러납니다.

무질서한 툴 확산이 얼마나 큰 문제일까요?

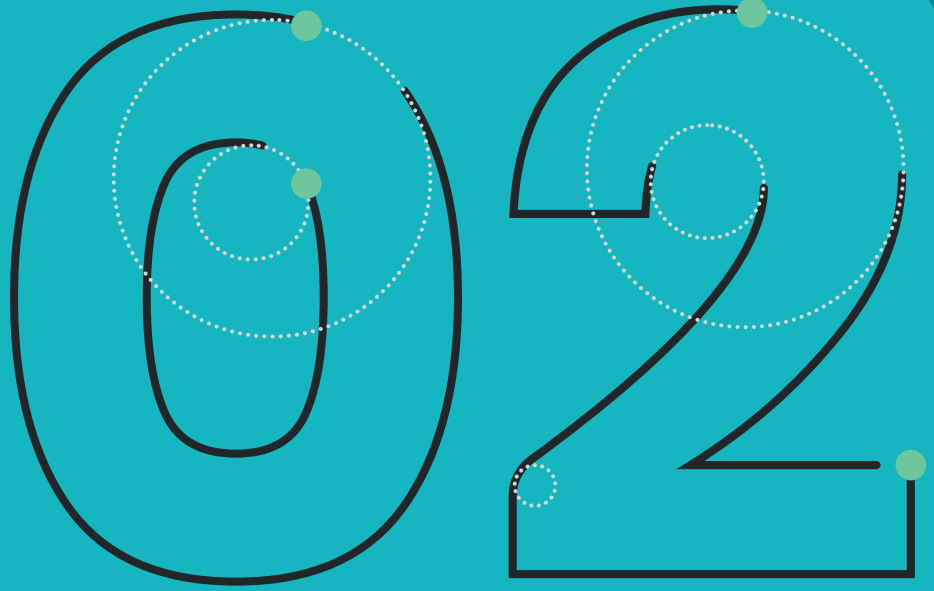
이 모든 것으로 인해 문제를 해결하는 데 시간이 더 오래 걸립니다. 일부 문제의 경우, 데이터가 너무 분산되어 근본 원인을 파악할 수 없으므로, 엔드유저 경험에 부정적인 영향을 줄 수 있는 문제들이 드러나기 시작합니다. 무엇보다도, 사용자들이 먼저 이러한 문제를 경험하고 보고를 합니다.

다운타임과 비즈니스 비용 간에는 직접적인 연관성이 존재하며, 그 비용은 상당할 수 있습니다.

Gartner에 따르면, 다운타임 1시간의 평균 비용은 3억 달러이며, 33%의 기업은 다운타임 1시간으로 인해 실제로 1~5백만 달러의 비용이 발생했다고 밝혔습니다.

무엇이 잘못되었는지를 명확하게 파악해야 문제를 방지할 수 있습니다.

그리고 이는 문맥이 있어야 가능합니다.



완전한 문맥이 있다면 어떨까요?

관찰과 모니터링은 그 자체만으로는 충분하지 않습니다.

성공적인 모니터링 및 관찰 관행에는 세 가지 주요 목표가 있습니다.

- 매출 향상
- 고객의 관심 향상
- 운영 효율성 향상

이 모든 것이 *비즈니스와 관련됩니다.*

최대한 많은 데이터를 수집하는 것만으로는 이러한 목표를 달성할 수 없습니다. 데이터 간의 점들을 직관적으로 연결할 수 있어야 하며, 무엇보다 시스템에 대해 핵심적인 질문을 할 수 있어야 합니다.

데이터가 많다는 것은 잠재적으로 더 많은 인사이트를 얻을 수 있다는 뜻이지만, 더 많은 툴을 사용해야 한다는 뜻이기도 하기 때문에, 더욱 인사이트를 얻기가 더 어려워집니다.

즉, '더 많은 툴 ≠ 더 많은 정보'라는 공식은 성립되지 않습니다.

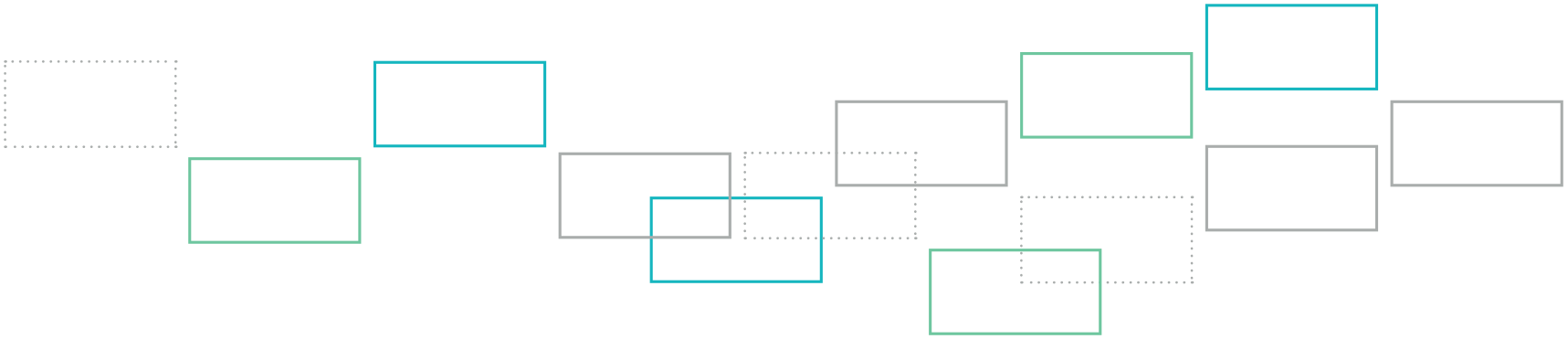
모니터링이 목표를 위한 수단이라면, 진정으로 중요한 요소는 다음과 같습니다.

- 모니터링 솔루션이 제공하는 비즈니스 가치는 얼마나 되는가?
- 문제 해결에 얼마나 효과적이고 유용한가?
- 핵심적인 문제를 파악하고 해결하는 데 데이터를 얼마나 잘 활용할 수 있는가?

툴 간의 전환을 위해 손실되는 시간과 문제 진단 및 해결에 소요되는 시간은 매우 중요합니다.

모니터링이 전체 스택에서 지표를 수집할 수 있게 해주지만, 비즈니스에 핵심적인 문제를 해결하는 데 도움이 되지 않는다면 리소스 낭비일 뿐입니다.

문맥이 중요해지는 부분이 바로 여기입니다. 단일 관찰 플랫폼이 모든 것을 변화시킬 수 있는 이유이기도 합니다.



완전한 문맥이 있다면 어떨까요?

문맥의 힘

단일 플랫폼으로 전체 스택을 관찰하면 문맥이 포함된 관찰성을 확보할 수 있습니다.

관찰성은 웹 및 모바일 애플리케이션의 엔드유저 경험에서 인프라와 애플리케이션까지 포괄하는 것은 물론, 모든 유형의 텔레메트리 데이터 (**메트릭, 이벤트, 로그, 트레이스**)를 한 곳에 통합해줍니다.

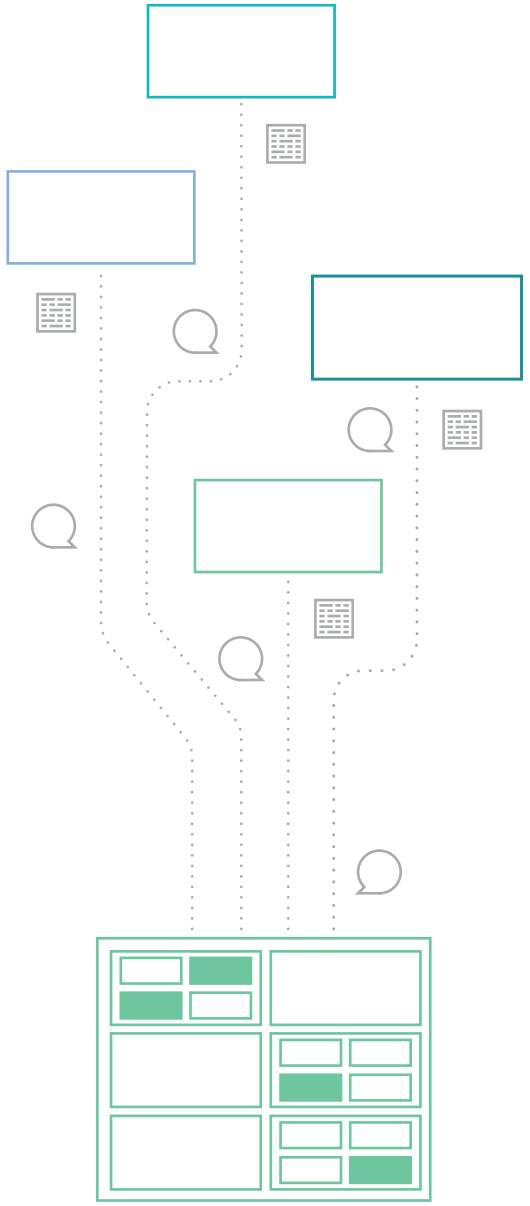
데이터를 의미 있게 연결해 이해할 수 있기 때문에, 팀들은 문제를 보다 빠르게 찾아 해결할 수 있습니다.

기업은 민첩성을 확보해, 데이터로 팀들을 위해 유용한 애플리케이션을 구축하여, 인프라 상태와 성능을 비즈니스 결과 및 고객 경험에 연결할 수 있게 됩니다.

데이터를 양방향 상호 작용이 가능하도록 시각화하여 각 팀이 원하는 방식으로 가장 관련성이 높은 데이터를 볼 수 있습니다.

단순한 문맥이 아니라 비즈니스의 니즈에 정확히 들어맞는 문맥을 통해 비즈니스와 관련된 모든 정보를 강력한 애플리케이션 구축 기능과 결합할 수 있습니다. 때문에, 문제 해결이 아니라 문제 방지에 힘쓸 수 있습니다.

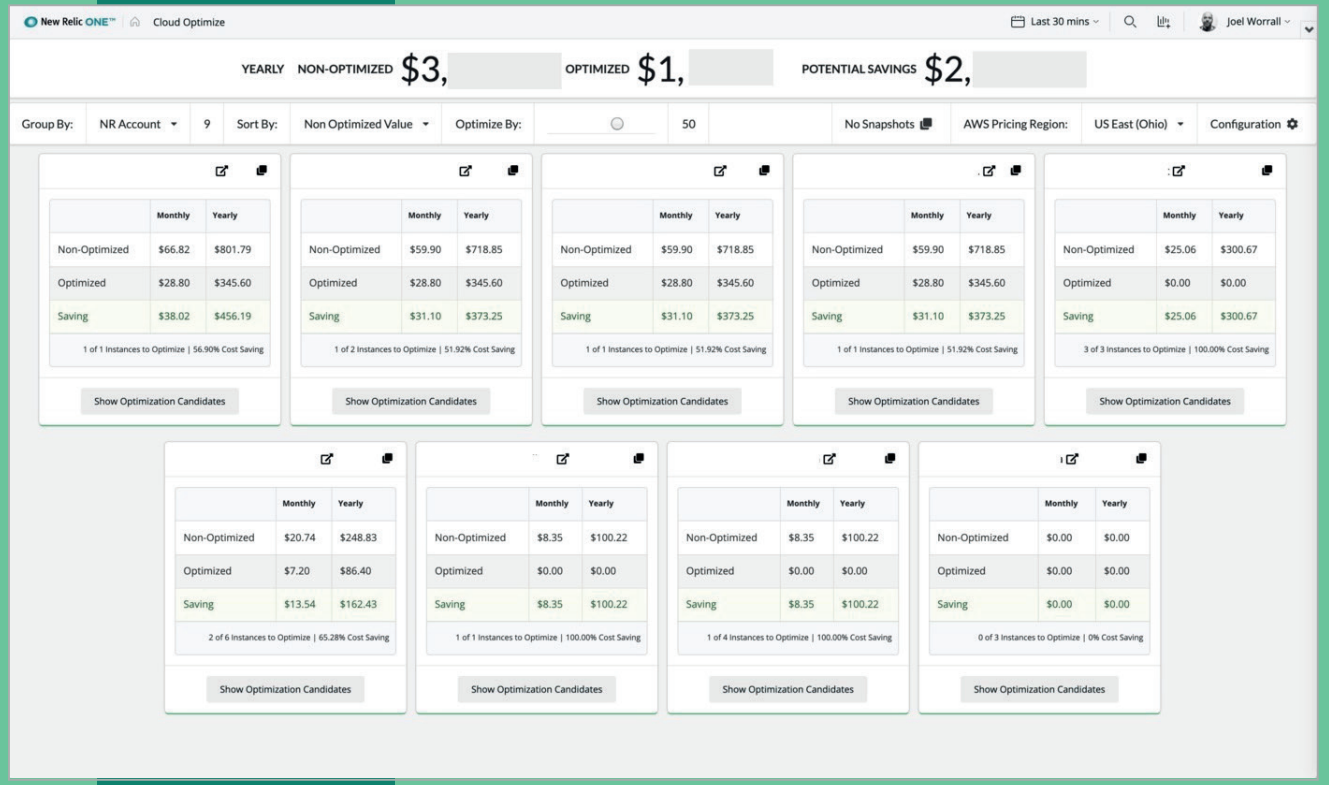
몇 가지 예를 통해 **그것이 어떤 모습일지** 한번 확인해보고, 비즈니스에 핵심적인 문제를 해결하려면 **무엇이 필요한지** 고려해 보는 것도 가치가 있을 것입니다.



완전한 문맥이 있다면 어떨까요?

클라우드 비용에 대한 완벽한 관리

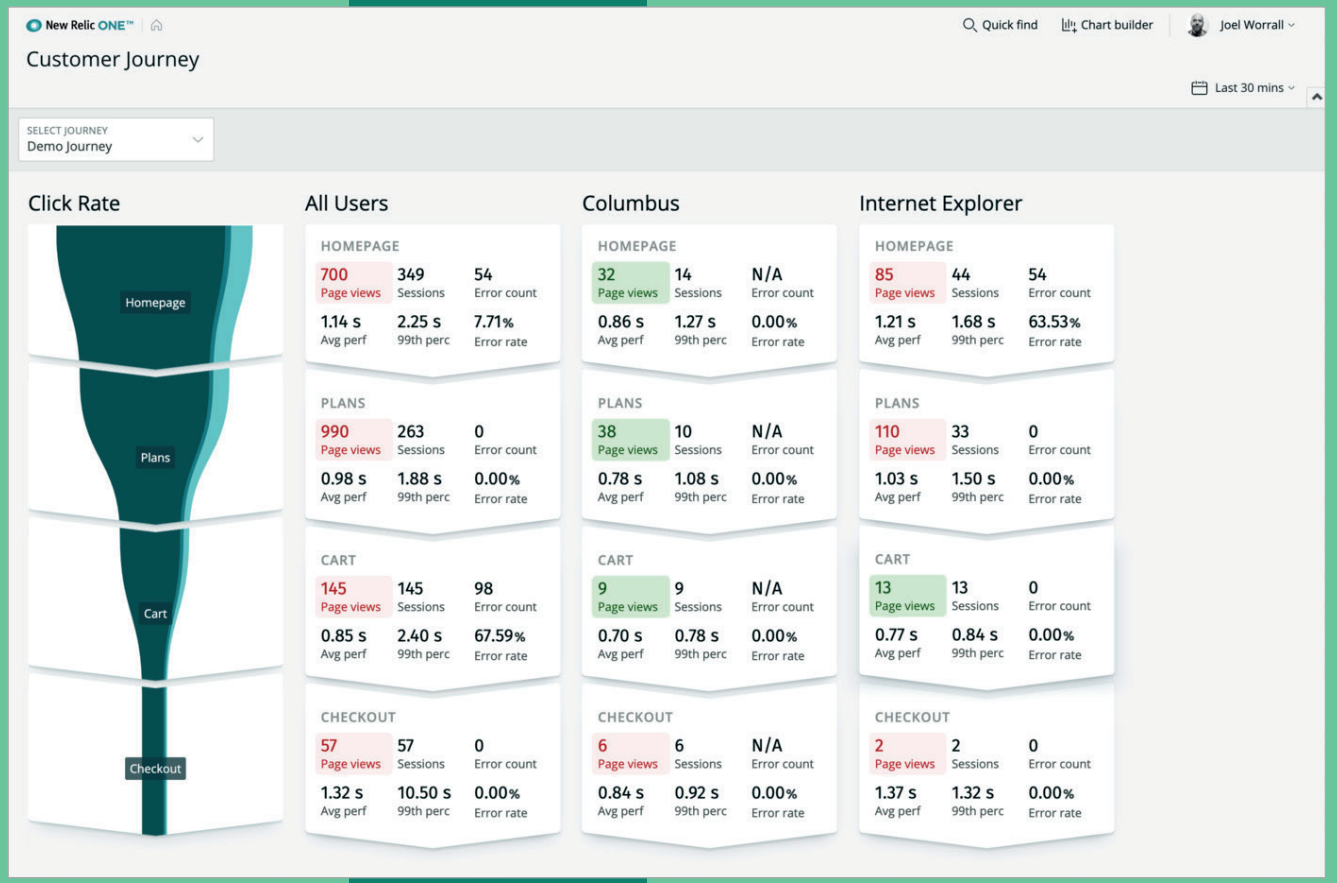
클라우드 인스턴스의 규모와 활용도를 비교하여, 팀들이 오버프로비저닝 가능성이 있는 리소스를 신속하게 파악할 수 있습니다. 보다 세분화된 호스트, 지역 및 설정을 선택하여 고유한 비즈니스 활용 사례를 구축할 수 있습니다.



완전한 문맥이 있다면 어떨까요?

고객의 구매 전환에 대한 이해

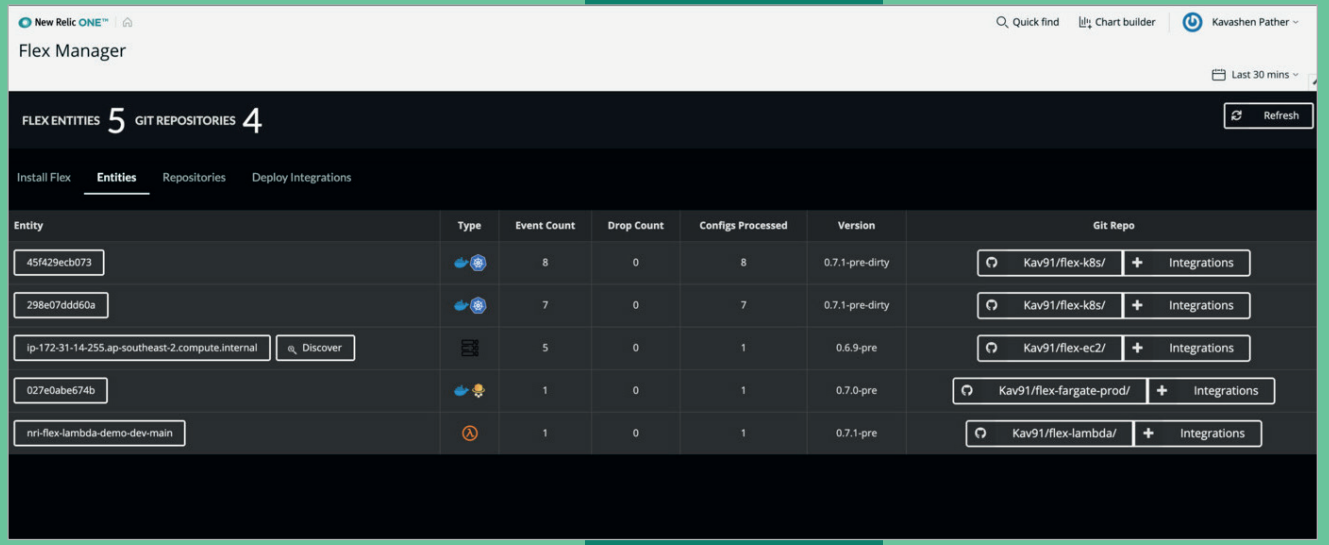
양방향 인터페이스를 통해 구매자의 여정 단계를 분석하고 맞춤화할 수 있습니다. 페이지 뷰, 오류율, 오류 수 등 각 단계의 표준 데이터를 볼 수 있습니다. 마케팅 퍼널의 각 단계에 대한 지표를 보다 자세히 확인할 수 있습니다.



완전한 문맥이 있다면 어떨까요?

새로운 인프라의 원활한 통합

운영 팀이 사용하는 엔드포인트는 기하급수적으로 증가하고 있습니다. 전문 지식이 필요하지 않은 올인원 통합 기능을 활용해 타사 소스의 데이터를 쉽게 수집할 수 있기 때문에, 팀들은 특정 인프라 환경과 관련된 문제를 해결하는 데 필요한 문맥을 확보할 수 있습니다.



결론

인프라에는 문맥이 필요합니다.

관찰성과 마찬가지로, 인프라는 그 자체만으로는 존재할 수 없습니다. 인프라는 더 광범위한 스택이라는 문맥으로 존재합니다. 비즈니스가 성공하려면 고객 경험이라는 광범위한 맥락에서 존재하는 인프라를 제대로 구축해야 합니다.

현대의 인프라 관리는 지속적으로 변화하기 때문에 애플리케이션 문제를 빠르게 감지하고 해결하기가 쉽지 않지만, 이를 올바르게 수행하는 것이 그 어느 때보다 중요해졌습니다.

로깅이나 Linux 인프라 같이 스택의 한 부분만을 모니터링하고 고립된 데이터에 의존하는 포인트 솔루션이 모순이 될 수 밖에 없는 이유도 이 때문입니다. 이러한 솔루션은 사용하는 팀을 위해 필요한 역할을 수행하지만, 특정 구성 요소를 벗어나는 즉시 큰 혼란을 야기합니다.

MTTR을 개선하고 다운타임을 줄이며 끊김 없는 엔드유저 경험을 제공하려고 노력하는 과정에서 충돌이 발생하면, 두 가지 질문에 즉시 대답할 수 있어야 합니다. "무엇이 고장났고, 그 이유는 무엇인가?"

옵저버빌리티, 즉 관찰 플랫폼이 제공하는 전체적인 문맥이 있어야 '무엇'의 단계에서 신속하게 '이유'의 단계로 이동할 수 있으므로, 이러한 질문에 빠르게 답변하고 다운타임이나 엔드유저의 경험 저하를 최소화할 수 있도록 팀들에게 역량을 제공해야 합니다.

최신 인프라스트럭처와 관찰 플랫폼은 인프라스트럭처, 로그, 설정 변경, 애플리케이션 및 프런트엔드 서비스의 다양한 데이터를 한 곳에 수집해 상호연관시켜 적절한 문맥으로 표시해줍니다.

이는 운영 팀이 인프라스트럭처가 애플리케이션에 미치는 영향이나 그 반대의 경우도 정확히 파악할 수 있다는 뜻입니다.

데이터에서 더 의미 있는 인사이트를 얻길 원한다면,

New Relic One을 고려해 보십시오. 개발 및 운영 팀이 단일 플랫폼에서 동일한 데이터에 액세스하고, 애플리케이션과 인프라에 대해 신속하게 상관관계를 수립해 문제를 보다 빠르게 파악하고 해결할 수 있도록 지원해야 합니다.

그래야 소프트웨어에서 어떤 일이 일어나도 엔드유저 경험에 영향을 미치기 전에 문제를 찾아 해결할 수 있습니다.

New Relic®