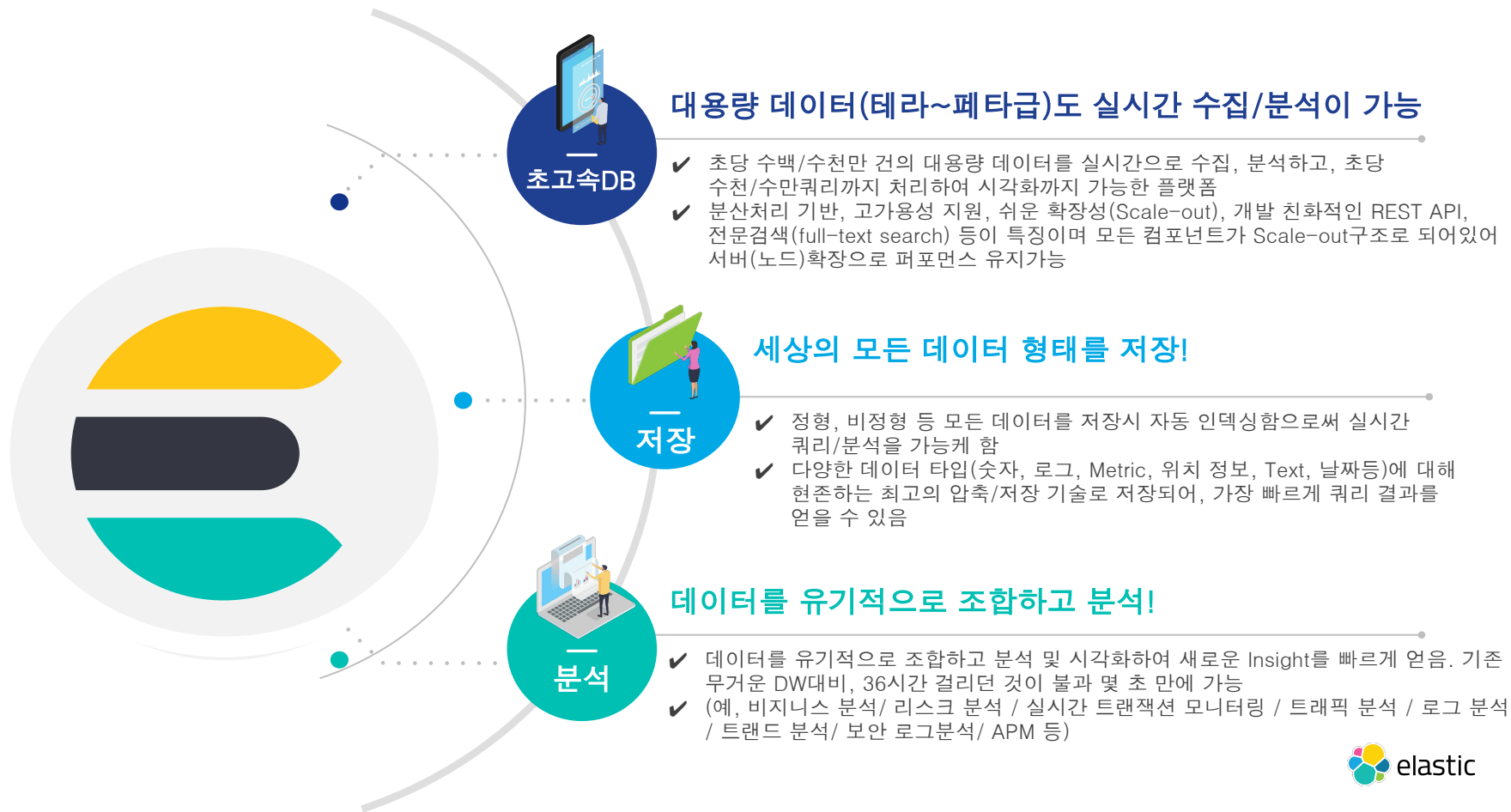




Elastic VS Splunk

Elasticsearch는 데이터를 수집, 검색(쿼리) 및 분석할 수 있는 데이터 플랫폼(초고속 DBMS)입니다
Elasticsearch를 이용한 연관 분석, 집계, 시각화를 통해 새로운 insight를 빠르게 얻을 수 있습니다





1. 엘라스틱 vs 스프링크

- a. 비용관점에서의 비교
- b. 퍼포먼스관점에서의 비교
- c. 기술관점에서의 비교
- d. 사용자(고객) 입장에서의 비교

2. 엘라스틱에 대한 오해와 진실

- a. 오픈소스는 안돼!
- b. 서버가 엄청나게 많이 필요해!
- c. 인덱싱할때 엄청 오래 걸려!
- d. A부터 Z까지 다 개발해야해!
- e. 검색엔진인데 무슨...대용량 데이터 분석?



3. 스프링크에서 엘라스틱으로 전환한 고객 사례





엘라스틱 vs 스프링크

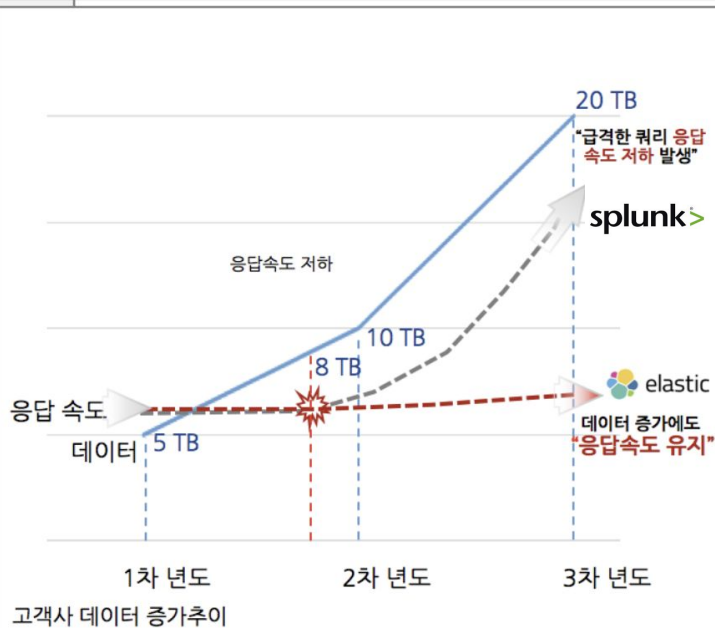
비용 관점

	과금방식	데이터 증가에 따른 비용 부담	비용 구조에 따른 특징	개발라이선스
스플링크	년간 라이선스 방식 과금 (일 볼륨에 기반) 과거엔 영구버전이 있었으나 현재는 연간 과금으로 변경	상당히 큼 (일 볼륨이 늘어남에 따라 비례적으로 금액 증가)	수집 해야할 대상이 늘어나거나 동일시스템에서 수집량을 늘리는부분에 부담이 늘어남에 따라 프로젝트의 목적 및 정체성이 흔들릴수 밖에 없음	유료
엘라스틱	년간 Subscription 방식 과금 (Node 수 기반)	유연함 (노드당 과금이기 때문에 hot/warm/cold 구조를 잘 조합하면 노드수를 예산에 맞게 조절가능함 예, 1노드에 1TB가 들어있든 20TB가 들어있든 동일 금액임)	수집대상이 늘어나거나 수집량이 늘어남에 전혀 부담스럽지 않음 금액에 대한 추가 부담이 당장 없기 때문에 프로젝트 목적에 맞게 유연하게 수집 대상 및 양을 늘릴수 있음	무료 (Subscription 구매시 무상으로 개발라이선스 제공됨)

Performance(쿼리 속도) 관점

성능 테스트 기준 : 대용량 데이터 수집 분석 (1일 8GB, ViaSat)

테스트 결과 : 하루 데이터 8GB 이상부터 급격한 성능저하



	데이터 볼륨에 따른 차이
스플링크	8TB(??) 이상부터 현저하게 쿼리 속도가 떨어지면서 응답시간이 현저히 올라감
엘라스틱	<p>엘라스틱은 모든 데이터가 Index(색인)되어 저장되어있기 때문에 데이터볼륨과 상관없이 1초이내로 쿼리속도를 유지 가능</p> <p>(구글에서 검색시 1초이내 결과 보여주는 것과 동일한 경험)</p>

기술 관점

Question	Elastic	S****
Platform vs Tool	<ul style="list-style-type: none"> 모든 데이터 분석 요구에 대한 솔루션 <ul style="list-style-type: none"> 비정형, 반정형 데이터를 유연하게 처리 가능 유연한 통합 로그, 메트릭 및 APM 분석 기존 로그 관련 UseCase 외에도 다양한 사례로 확장 가능 	<ul style="list-style-type: none"> 로그 데이터 분석을 위한 도구 <ul style="list-style-type: none"> 제한된 범위 -> 로그/시계열 데이터 제한 다양한 UseCase에 적용시, 기술적 문제 有
Speed over Disk	<ul style="list-style-type: none"> Elastic은 웹 스케일 쿼리 성능을 제공하도록 설계 Elastic 플랫폼에서 훨씬 더 빠른 실시간 쿼리를 통해 귀중한 분석 시간 절약할 수 있음 여러 사용 사례를 통해 시간 경과에 따른 TCO 절감 및 ROI 개선 동일한 플랫폼에서 여러 사용 사례를 결합하여 전체 TCO 절감 	<ul style="list-style-type: none"> S****는 쿼리속도 대비 한계 스토리지 효율성을 제공토록 설계 S**** 쿼리는 몇 시간이 걸릴 수 있지만, Elastic은 거의 즉각적인 결과를 제공 이러한 설계 방향은 패치나 업그레이드 등으로 쉽게 변경할 수 없음
Flexible Licensing Model	<ul style="list-style-type: none"> 무료 오픈 소스 버전인 Elastic stack으로 대규모 프로토타입 구축 프로토타입 구축 후 운영에 대한 준비가 되었을 때 Expert 투입 및 활용 	<ul style="list-style-type: none"> 일단위 데이터 용량에 의한 유연하지 않은 과금 정책 구성한 환경에 문제가 있고, 그 결과 더 많은 데이터가 생성된 경우 삭제하거나 초과 비용을 지불해야 함
End-to-End Solution	<ul style="list-style-type: none"> End-to-End 환경의 새로운 요건이나 운영 상황에 맞게 유연하게 확장이 가능함 	<ul style="list-style-type: none"> 사전 구축(Pre-Built)된 솔루션은 확장이 아닌 재구축의 방법이 필요하게 됨
Lower Risk	<ul style="list-style-type: none"> Elastic은 오픈소스 기반으로 기술(코드)의 투명성과 책임성 有 <ul style="list-style-type: none"> 코드 peer review로 오픈 소스 보안 및 성능 향상 오픈 소스 프로젝트에는 대규모 커뮤니티를 통해 새로운 활용 사례 등이 업데이트 됨 공급업체의 로드맵에만 의존하지 않음 	<ul style="list-style-type: none"> 특정 벤더(S****) 중심의 폐쇄 된 생태계에 시스템이 국한될 수 있음

기업(고객) 입장에서의 비교

	인프라부터 Application 까지 통합 모니터링	보안	국내 기술지원 서비스	개발라이선스
스플링크	APM이 통합되어있지 않음	SIEM에 국한됨	한국어 지원 불가	유료
엘라스틱	Infra부터 Application까지 하나의 플랫폼에서 처리 가능	SIEM + EDR(Endpoint)의 통합된 플랫폼	엘라스틱소속 한국어 지원 기술지원만 8명	무료 (Subscription 구매시 무상으로 개발라이선스 제공됨)



엘라스틱에 대한 오해와 진실

오픈소스는 안돼 !

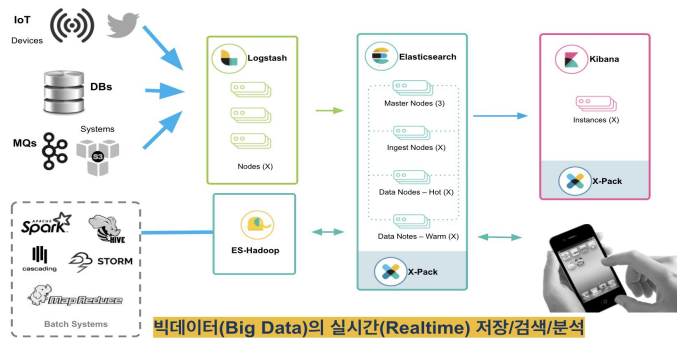
- ★ 엘라스틱은 R이나 아파치 하둡과 같이 단체나 커뮤니티가 관리하는 SW가 아닙니다
- ★ 엘라스틱은 엘라스틱사가 개발/테스트/검증을 거쳐 정식으로 Release하는 정식 SW입니다
- ★ 제품의 버전이 오픈소스의 커뮤니티버전과 상용라이선스기반의 기업용 버전 두가지가 존재합니다
- ★ 오픈소스는 단지 빠르게 확산시키기 위한 방식일뿐입니다
- ★ 이미 미 정부기관 금융사등 크리티컬한 운영환경에도 다양하게 활용되고 있고 국내에도 공공/금융/통신/제조등 크리티컬한 운영환경에 많은 Reference를 확보하고 있습니다

서버가 엄청나게 많이 필요해 !

- ★ 엘라스틱은 오픈소스(무료버전) 사용자가 많기 때문에 엘라스틱의 전문가의 지원없이 설치 및 사용하면서 생겨난 잘못된 오해이며, 오픈소스버전에는 대부분의 **default**로 설정된 값들이 튜닝요소이기때문에 엘라스틱 전문가를 통해 아키텍처 및 노드/샤드 설계를 하게되면 서버수를 크게는 30~40%까지 줄여도 빠른 퍼포먼스를 낼 수 있습니다
- ★ 일반적으로 경쟁사에서 엘라스틱의 상용버전이 아닌 오픈소스버전에 대한 잘못된 지식에 기반하여 주장하는 내용일뿐입니다

인덱싱할때 엄청 오래걸려 !

- ★ 이또한 오픈소스(무료버전) 사용자들 사이에서 엘라스틱 전문가의 도움없이 아키텍처 및 노드/샤드 설계를 하게되면 경험하게 되는 문제로 엘라스틱 컨설턴트의 지원 및 기술지원 서비스를 통해 인덱싱 속도를 폭발적으로 향상시킬수 있다
- ★ 이유는 아주 간단한데...아래 그림(모두 (X)라고 표현)에서 보듯, 엘라스틱의 비츠, 로그스태시, 엘라스틱서치 및 키바나 모두 **SCALE-OUT** 구조로 되어있기 때문에, 인스턴스만 Data Traffic에 맞춰 늘리면 인덱싱 또한 원하는 퍼포먼스를 얻을수 있다



A부터 Z까지 다 개발해야해 !

- ★ 아직도 오픈소스버전 사용자가 많다보니 엘라스틱 버전의 5점대나 2점대를 사용할때의 얘기이며, 현재 버전(7.6)의 엘라스틱 상용제품의 경우 많은 부분들이 **Solution**화 되어있어 예를들어 **Beats**를 설치해서 데이터를 수집하게되면 **Data parsing**부터 시각화 화면 및 이상징후탐지를 위한 머신러닝 **Rule**까지도 모두 자동으로 만들어준다
- ★ 물론 어떤 소프트웨어이든 기본적인 SI성 커스터마이제이션은 필요할것이며 이또한 엘라스틱 공식 파트너사의 도움을 받아 구축이 가능하다
- ★ 시장의 많은 개발 업체들이 엘라스틱 오픈소스버전으로 프로젝트를 구축하면서 엘라스틱 전문가로써 자신들을 포장하나 아직도 엘라스틱을 단순 빠른 **DB**로 이해하고 **RDB**처럼 설계하는 업체들이 많은 만큼 반드시 운영환경의 경우 엘라스틱사를 통해 공식파트너 소개를 받는것이 필요하다

검색엔진인데 무슨...대용량 데이터 분석 ?

- ★ 검색이 영어로 Search라고 하는데, 쿼리(Query) 또한 Search와 같다고 보면 된다
- ★ 이로써 엘라스틱은 우리가 다루고 있는 거의 대부분의 정형/비정형 데이터에 대해서 초고속의 쿼리속도를 기반으로 대용량 데이터에서 인사이트를 실시간으로 빠르게 얻을수 있는 빅데이터의 실시간 저장/검색(쿼리)/분석이 가능한 데이터 플랫폼이다
- ★ 이것이 국내에서 엘라스틱을 초고속 DB로 소개하는 이유이기도 하며 다음장에 각각의 기업들이 엘라스틱을 통해 경험하는 대용량 수집/쿼리/분석에 대한 속도가 이를 증명한다



- 180 billion documents in Elasticsearch
- 190 TB total data size
- 20 TB daily ingest rate
- 25 Queries/Sec



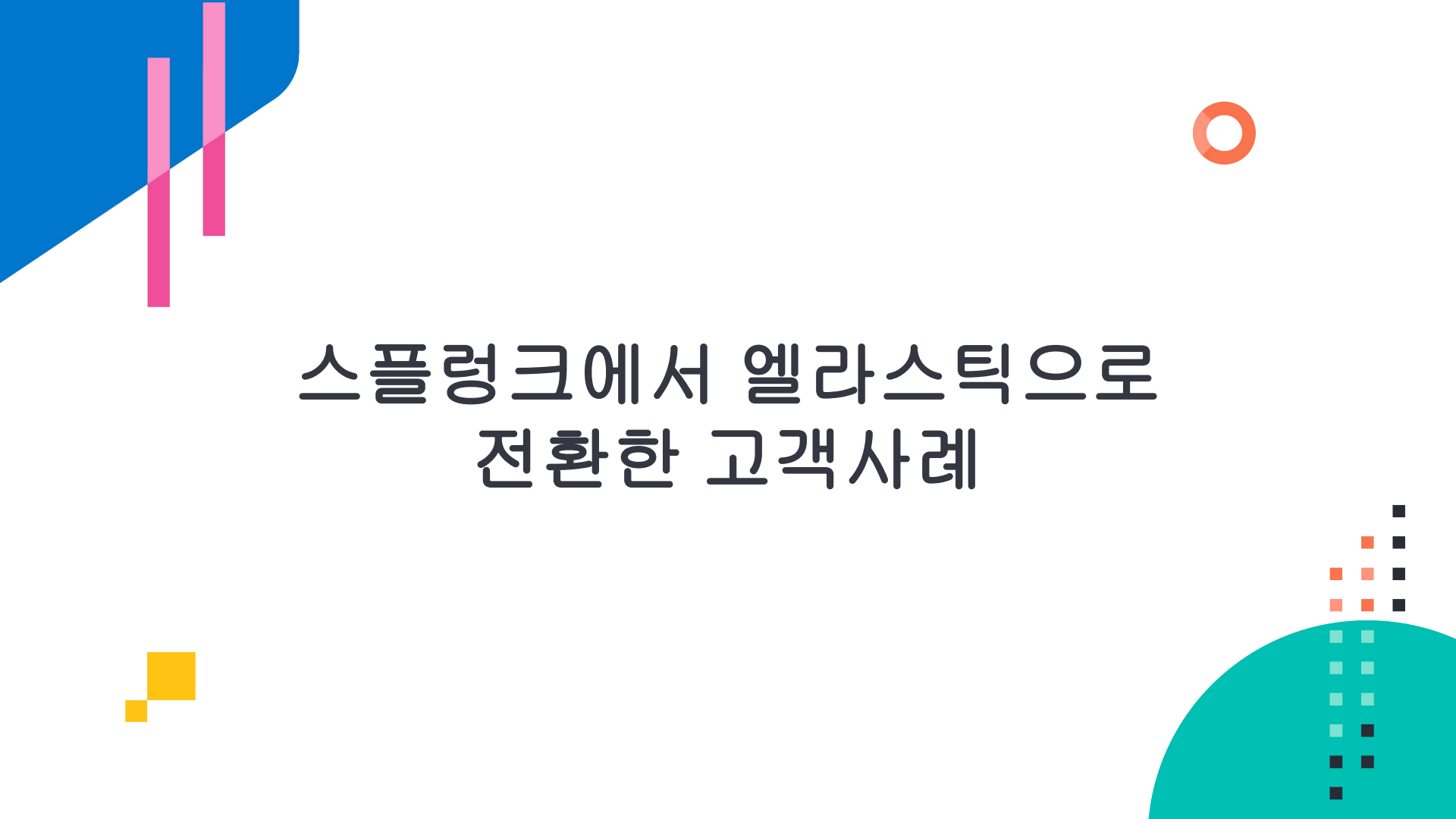
- 100,000+ computes
- 1000 억개의 URL Hits
- 1.2 PB daily application logs
- 5 M/Sec data metric points



- 하루 30억건의 레코드를 수집하며, 50TB 실시간으로 저장
- 200 개 대시보드에서 하루 30억개의 events from logs, DB, emails, syslogs, test messages, and internal and vendor application APIs.



- 하루 3억건의 쿼리(초당 3000개 쿼리 처리)
- 하루 수집데이터 양: 5천만 documents
- 77백만명의 고객을 대상으로 하루 3백5십만건의 탑승 서비스 제공



스플링크에서 엘라스틱으로 전환한 고객사례

Splunk에서 엘라스틱으로 전환



- ★ 최근 몇 년간, **Box**의 엔지니어링 팀은 보고용 레거시 백 엔드에 확장성이 없다는 사실에 우려가 커지고 있었습니다. **Box**의 로드맵이 성장하고 있었기 때문에, **Observability** 팀은 **Splunk**가 제공했던 것보다 애플리케이션과 운영 로깅을 위해 보다 안정적이고 비용 효과적인 솔루션을 채택하는 데 관심이 있었습니다.
- ★ 로깅은 중요성이 강화된 중요 업무용 프로세스였습니다. 특히, **Box**가 성장, 혁신, 그리고 새로운 고객 대면 기능 제공이 가능하도록 단일한 인프라를 수백 개의 마이크로서비스로 변환하는 작업을 계속 진행하면서 더욱 그러했습니다.
- ★ **Box**의 레거시 로깅 솔루션은 수집되는 데이터의 양에 따라 가격이 책정되었기 때문에, 비용을 제한하기 위해 때로는 **Box**가 로깅 프로젝트를 차단해야 하는 상황에 처하게 되기도 했습니다. 그렇지 않으면, **Box** 엔지니어가 새로 배포되는 마이크로서비스로부터 이벤트를 로그하지 않기로 결정하게 될 테니까요.
- ★ 당시에는 그것이 피할 수 없는 현실이었고, 선도적인 **Cloud Content Management** 플랫폼으로 빠르게 혁신해가려는 **Box**의 사명과는 잘 맞지 않았습니다. 이 혁신을 위해 **Box**는 단일 구조를 마이크로서비스로 분할해야 했으며, 이는 로깅이 줄어드는 게 아니라 더 많은 그리고 포괄적인 로깅이 요구되는 방향이었습니다.

SPLUNK: 비용절감 = 수집데이터 볼륨 제한



"비용 절감을 위해서는 로깅 데이터 볼륨을 줄여야 할 것인데, 이는 **Box** 시스템이 좀더 관측가능하게 만들려는 우리의 사명에 반하는 것이었습니다. 우리는 보다 비용 효과적인 시스템을 구축하고, 보다 가시성을 높인다는 사명을 위해 계속 작업하고 싶었습니다. 그래서 우리는 **Elastic**을 선택했습니다. 이것은 개발자가 개발자를 돕기 위해 구축한 것입니다."

– Deepak Wadhvani, Observability 팀의 엔지니어링 관리자 | Box



Elastic = Happy

"우리 엔지니어들은 지금 훨씬 더 행복해하고 있으며, 쿼리는 거의 즉시 완료됩니다. 우리의 만족도 지수는 훨씬 더 높습니다."

– Salman Ahmed, 데이터 플랫폼 및 Observability SRE 팀 엔지니어링 디렉터 | Box



"우리는 이렇게 생각하고 있었습니다. 우리는 계속 규모가 커지고 있는데, 이 모든 것이 5년 후에는 어떤 의미가 될 것인가? **Elastic**으로 전환하면서 여러 가지로 도움이 되었습니다. 테라바이트당 비용은 반으로 줄었고, 우리 개발자들에게는 한결 작업이 편해졌으며, 이들이 구축하고 있는 마이크로서비스에 대해 **Observability** 기능을 제공했습니다. 우리는 더 이상 비용 때문에 로깅 프로젝트를 거절하고 돌려보낼 필요가 없게 되었습니다."

– Deepak Wadhvani, Observability 팀의 엔지니어링 관리자 | Box

Box 클러스터

클러스터 수

1개

노드 수

데이터 85개, 마스터 3개, 클라이언트 6개

LS 인스턴스/Beats 수

Logstash 20개

총 문서 수

1,800억 개

총 데이터 크기

190TB

일일 수집 비율

20TB

인덱스 수

250개

쿼리 비율

초당 쿼리 25개

복제

1개

시계열 인덱스

매일

노드 사양: 총 메모리,
CPU, 디스크 유형(SSD,
HDD)

AWS i3.4XLarge

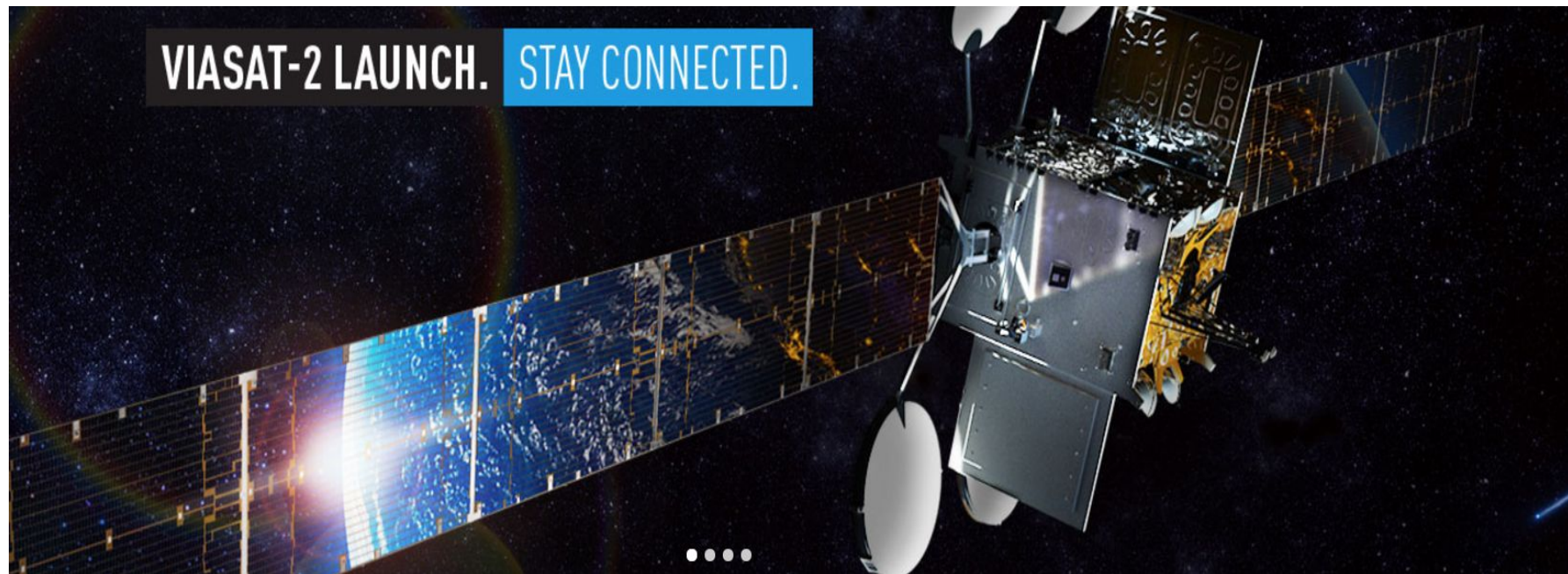
CDK's Story (기존 Splunk에서 엘라스틱으로 전환)

- 운영 로그 모니터링 용도로 엘라스틱 활용
- 수백명의 사용자들이 사용할 수 있도록 환경 구축
 - 키바나는 쉽고 다양한 부서의 사람들의 목적에 충분히 부합하고 있음
- 원하는 데이터 전체를 모두 수집
 - 데이터가 늘어나도 비용에 큰 부담이 없어, 데이터를 일부러 버릴필요가 없음
 - 기존에는 비용에 대한 부담이 너무 커서 전체 데이터가 아닌 일부분만 수집하든지 집계데이터만 수집했었음
- 모든 데이터에 접근이 가능함은 그만큼의 가치를 증대시켜줌
 - 현상에 대한 이해가 훨씬 빨라짐(필요한 전체 데이터에 대한 접근이 가능해짐)

“필요에 의해 더 다양하고 많은 데이터를 추가로 수집을 해서 모니터링 해야함에도, 늘어나는 비용때문에 할 수 없는 상황이 너무 짜증이 날 뿐이다”

- 시스템 운영 담당

미국 대형 위성 통신사 VIASAT의 스프링크에서 엘라스틱 전환



ViaSat Problem Statement and Elastic Why



Network Ops

- Need for Scale Speed
- Splunk tooling too slow
- Requirement for Proactive Monitoring and Alerting



DevOps

- Monitor app lifecycles
- Desire to improve application cycles



Management Team

- Desire for centralized Logging Solution

Elastic Products at ViaSat

- **Elastic Stack at ViaSat:** Elasticsearch, Logstash, Beats, Kibana, X-Pack: Security, Alerting, Monitoring, Reporting
- **Tech Challenges:** Large metrics use case, up to 8TB per day
- **Elastic Advantage:** Speed, large community, we opened their eyes to new use cases and the Elastic Stack as a platform
- **Support engagement model:** COE who will engage support (monitoring/metrics)
- **Growth?** Possibility for more daily ingest (TB) as adoption increases and value realized (we had proposals on table for up to 113 nodes during process)



We conducted a bake-off of several datastores, only Elasticsearch could scale by tuning it to do everything we need. The others by comparison:

Druid was difficult to setup, manage; doesn't support the queries we need.

Splunk query performance was awful.

Kudu is just a file system, we'd have to build too much

Anthony Kinsley
Software Engineer, Metrics-as-a-Service Owner, ViaSat