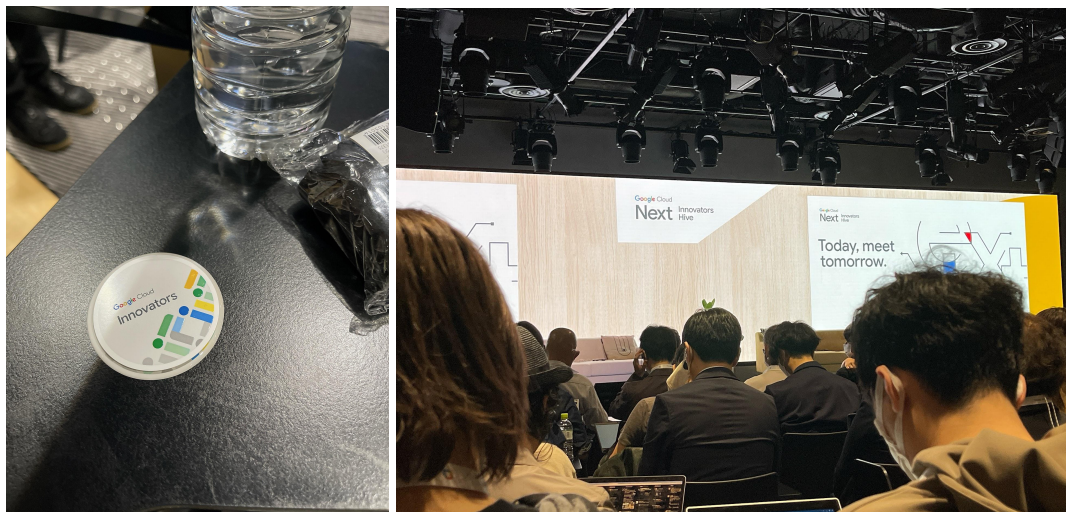


## 基調講演前の様子



データ活用から管理までを知るエキスパートが語る **BigQuery** 使いこなしの秘訣 ～初入門の方法からこれからのデータ基盤運用まで～

共有することが大切

(クエリをつけて返したら、興味を持ってもらえる)

BigQueryは共有が楽

コネクテッドシートとBQでSQL書かずに分析可能に

11:16  
Amazon | amazon.co.jp

Google BigQuery: The Definitive Guide:  
Data Warehousing, Analytics, and  
Machine Learning at Scale  
English Edition  
★★★★☆ (91)  
O'REILLY

**Google BigQuery**  
**The Definitive Guide**  
Data Warehousing, Analytics, and Machine  
Learning at Scale

Valliappa Lakshmanan  
& Jordan Tigani

Kindle (Digital) ¥4,570 (46pt)	Paperback ¥6,845 (68pt)
-----------------------------------	----------------------------

Shop all formats >

As an alternative, the Kindle eBook is available now and can be read on any device with the free Kindle app.

おすすめ書籍

## お題2

**BigQuery**をどこで使っている？どうやって使っている？

LookerStudio(旧 Data Portal)

スプレッドシート

データコネクト

GAS

Cloud Composer

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

- ETLからELTが主流に  
(BQに入れる前に加工から、BQに入れてから加工に。)  
[https://www.ashisuto.co.jp/eai\\_blog/article/201811\\_etl.html](https://www.ashisuto.co.jp/eai_blog/article/201811_etl.html)

- BQのSQLでデータクレンジング可能  
pandas不要に
- BQで加工してから、SSで見ると良い感じに。

## お題3

どうやって、管理していきますか？

- テーブルのユニークな名付け
- 権限管理  
グループ付与。基本、個人にはつけない

Googleグループでデータセット単位で

個人、サービスアカウントへの付与はしない

Googleグループの管理権限さえ渡せば、人の出入りを自由に管理してもらえる。

オンプレからクラウドにするには？

オンプレSQLのリードレプリカをクラウドSQL

## セッション3

**MLOps と Vertex AI** について語る会

Vertex AI

TFXのコンポーネント群をサービスとして展開

GoogleではSREがDevOpsに取り組む。

VertexAI Pipelineのあれこれ

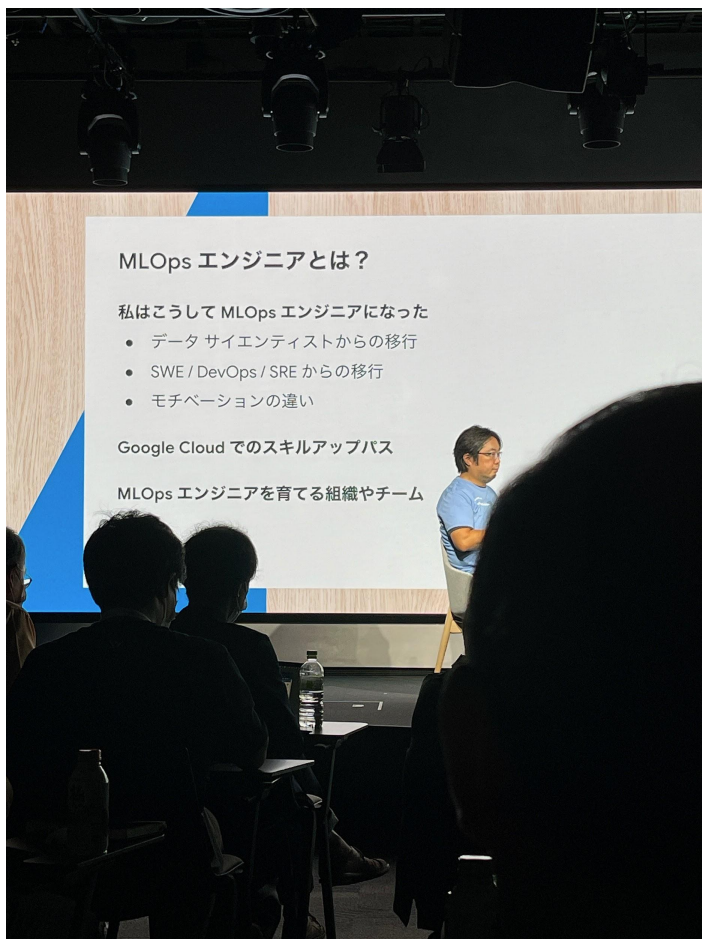
がんばってほしいところ  
サーバーレスゆえ、エラー検知が難しい

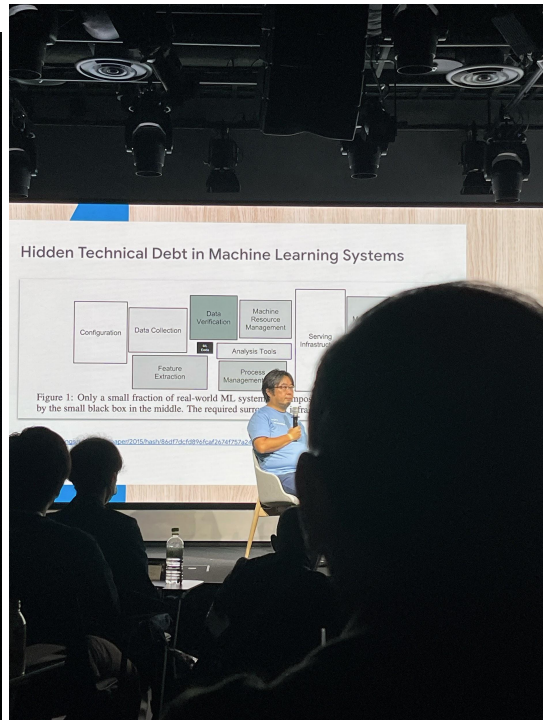
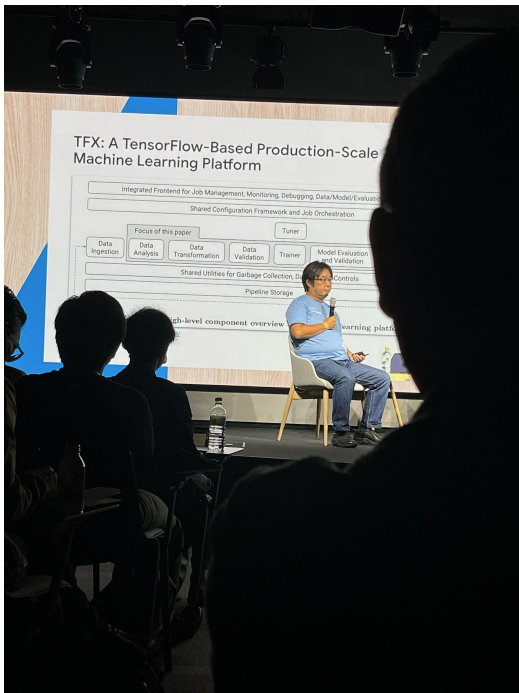
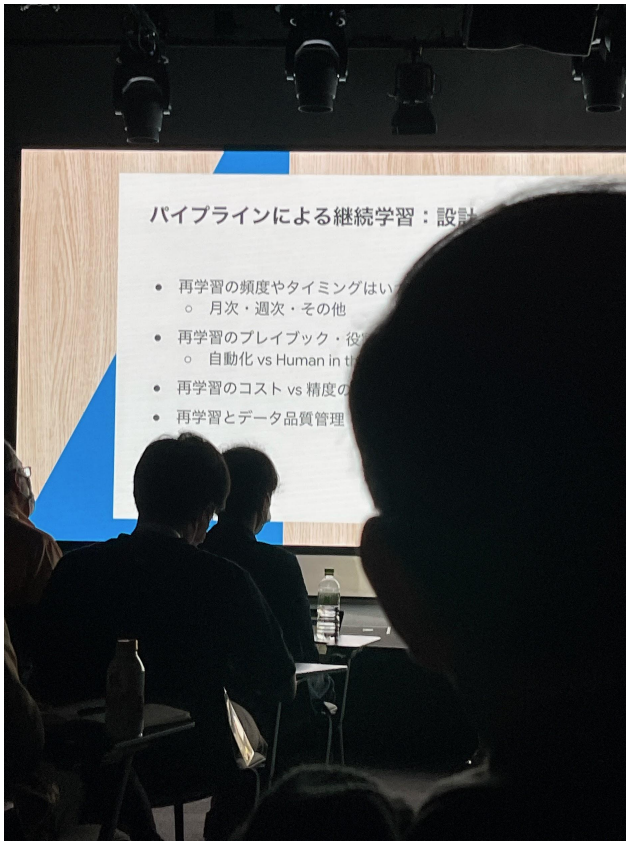
MLOpsをどう始めるか？

BQMLはおすすめ。手軽。  
SQLを書ければできてしまう。  
インフラの面倒がない。  
試してみようかなと。

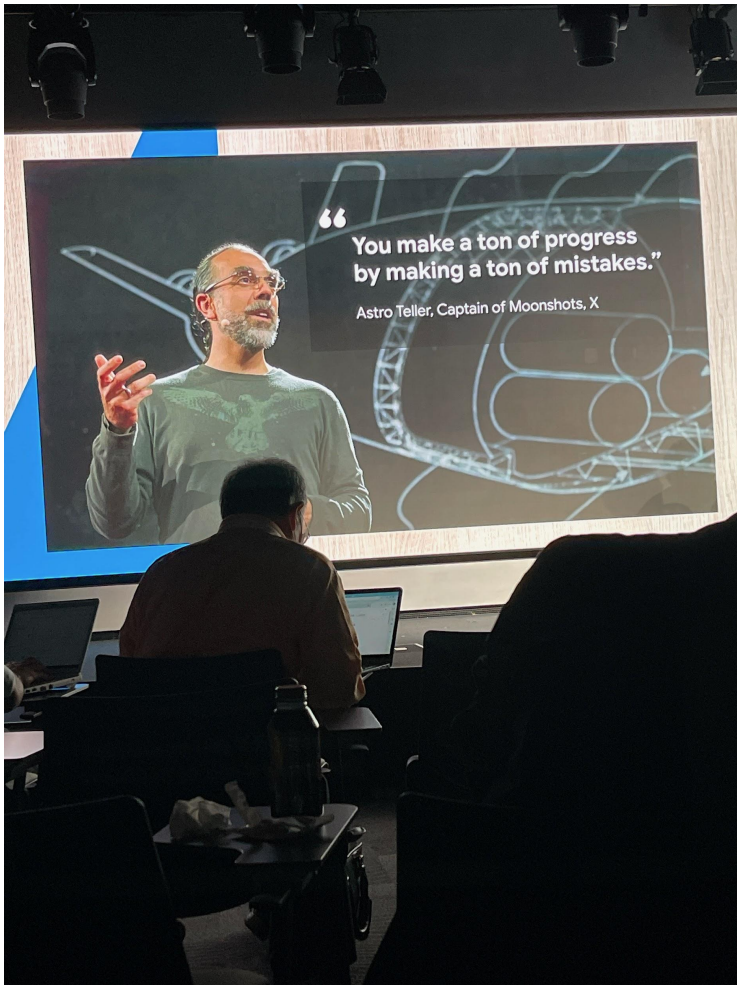
MLOpsエンジニアとは？

もともとアルゴリズム屋から  
アルゴリズム屋だと事業貢献、評価されにくい。  
インフラまで手が回ることはない。

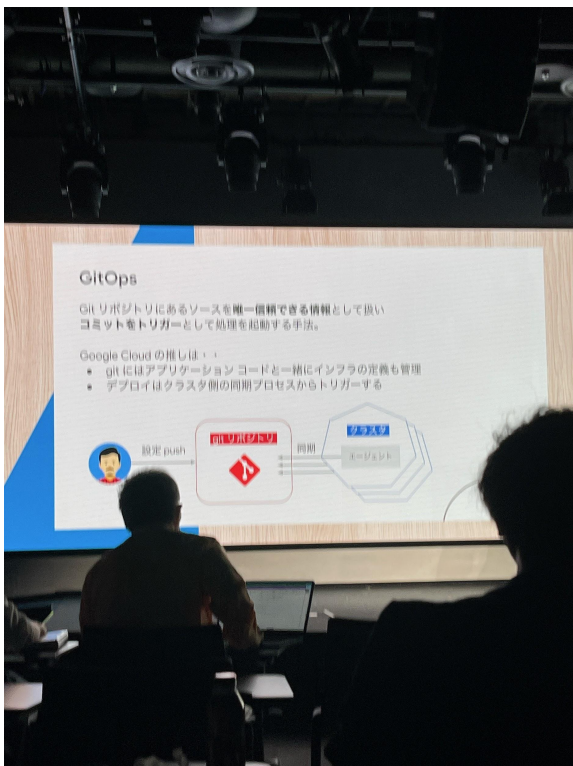


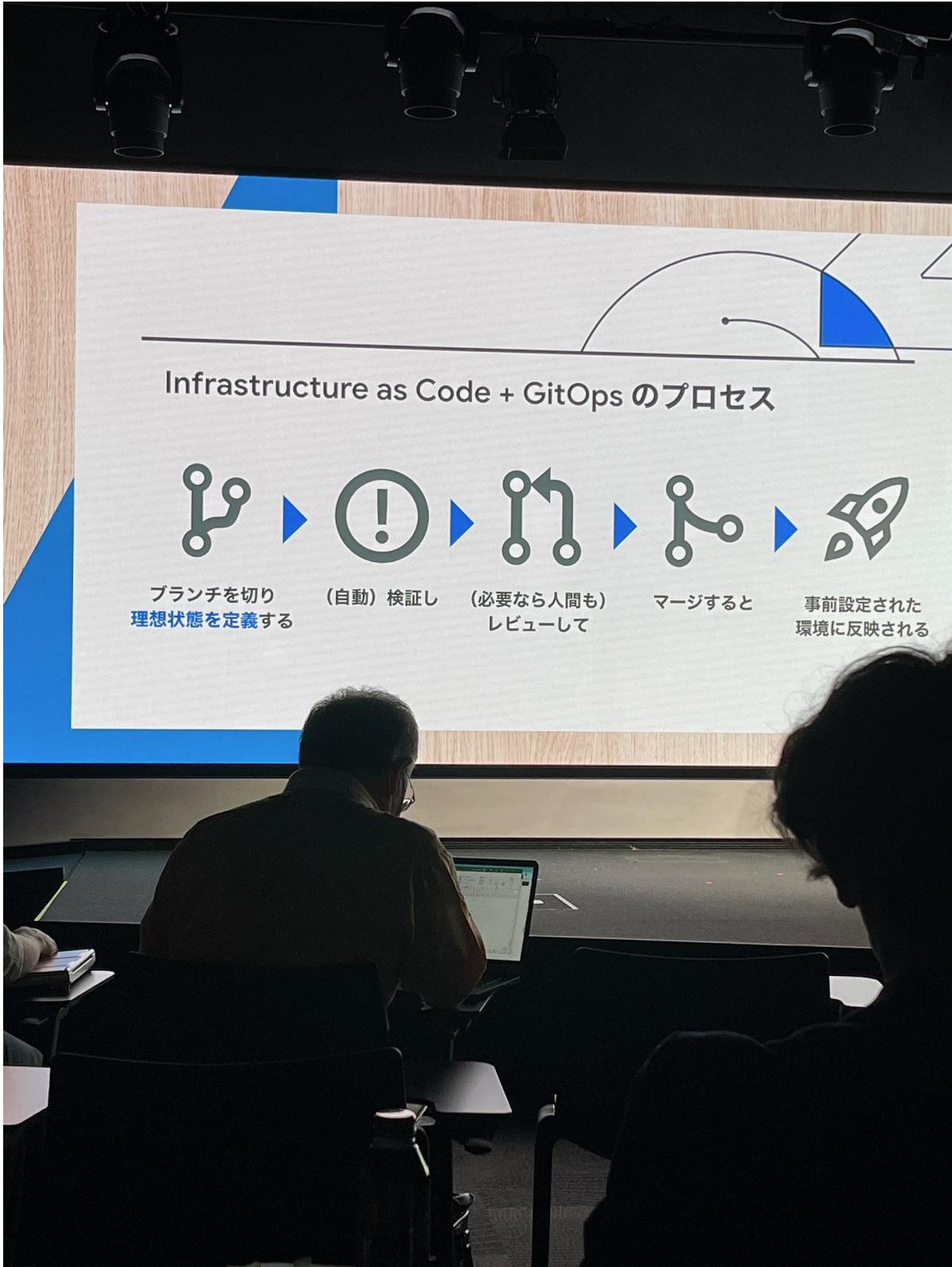


ソフトウェア デリバリー：クラウド ネイティブな開発プロセスの解説



膨大な数の失敗が、大きな進歩をもたらす





Configuration as Data  
環境設定もGit管理

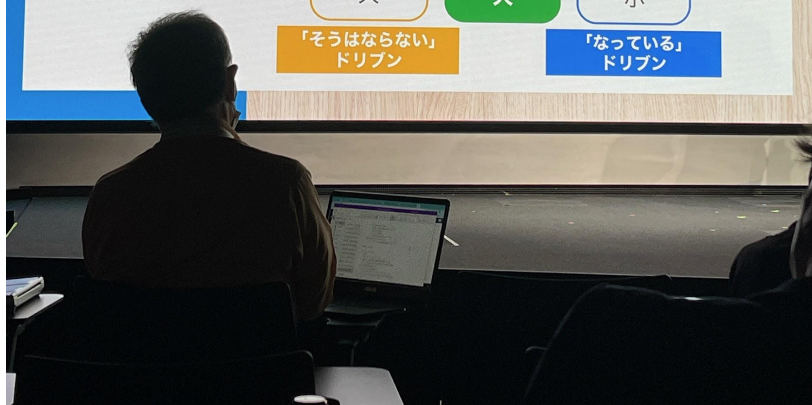


# スタートアップ or エンタープライズって 単純な話でもありませんが

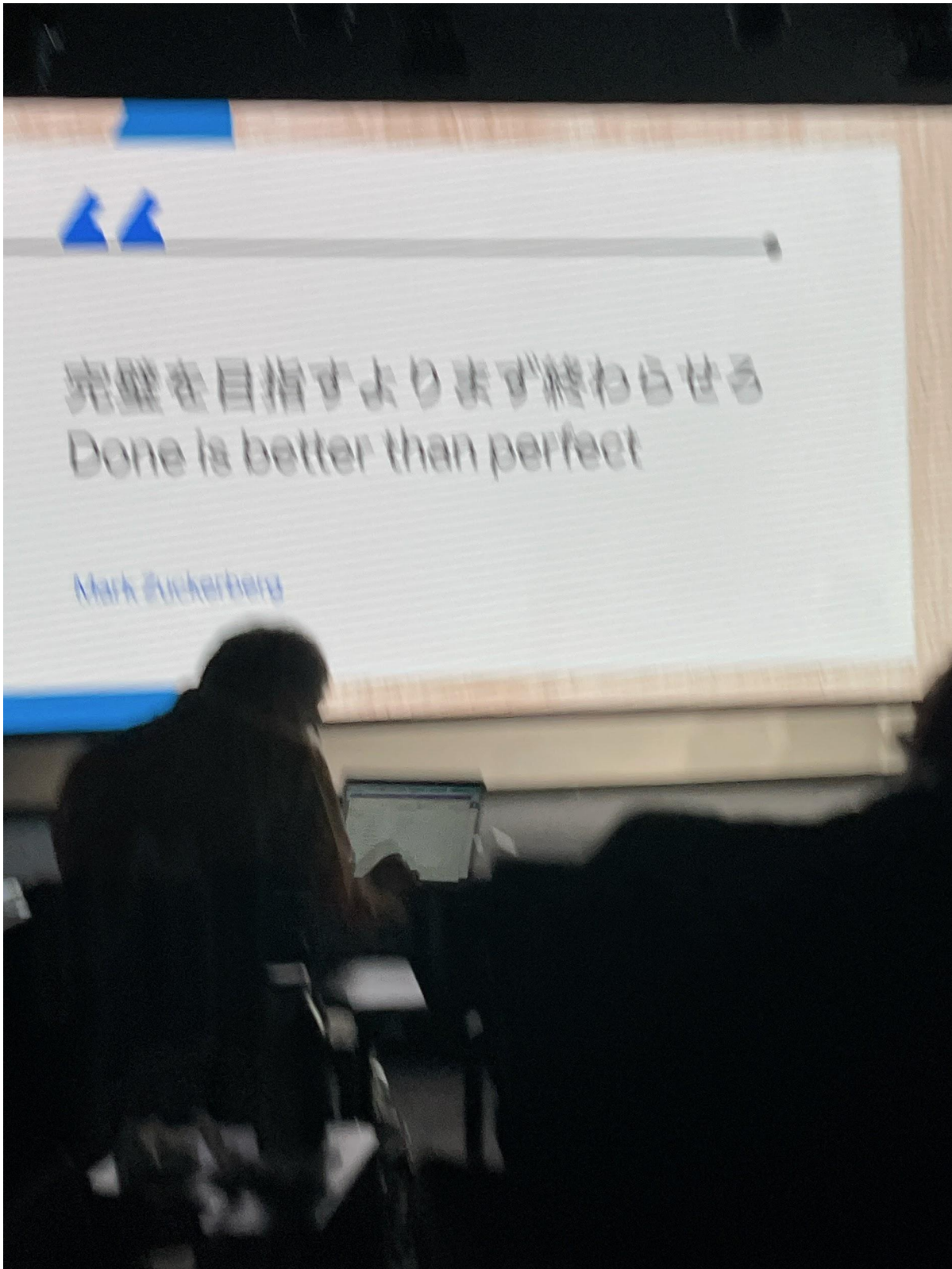
	Startup	Cloud Native Enterprise	Enterprise
プロダクト数	小	大	大
プロダクトあたりの開発規模	小	大	大
社会的責任	小	大	大
変化のスピード	大	大	小

「そうはならない」ドリブン

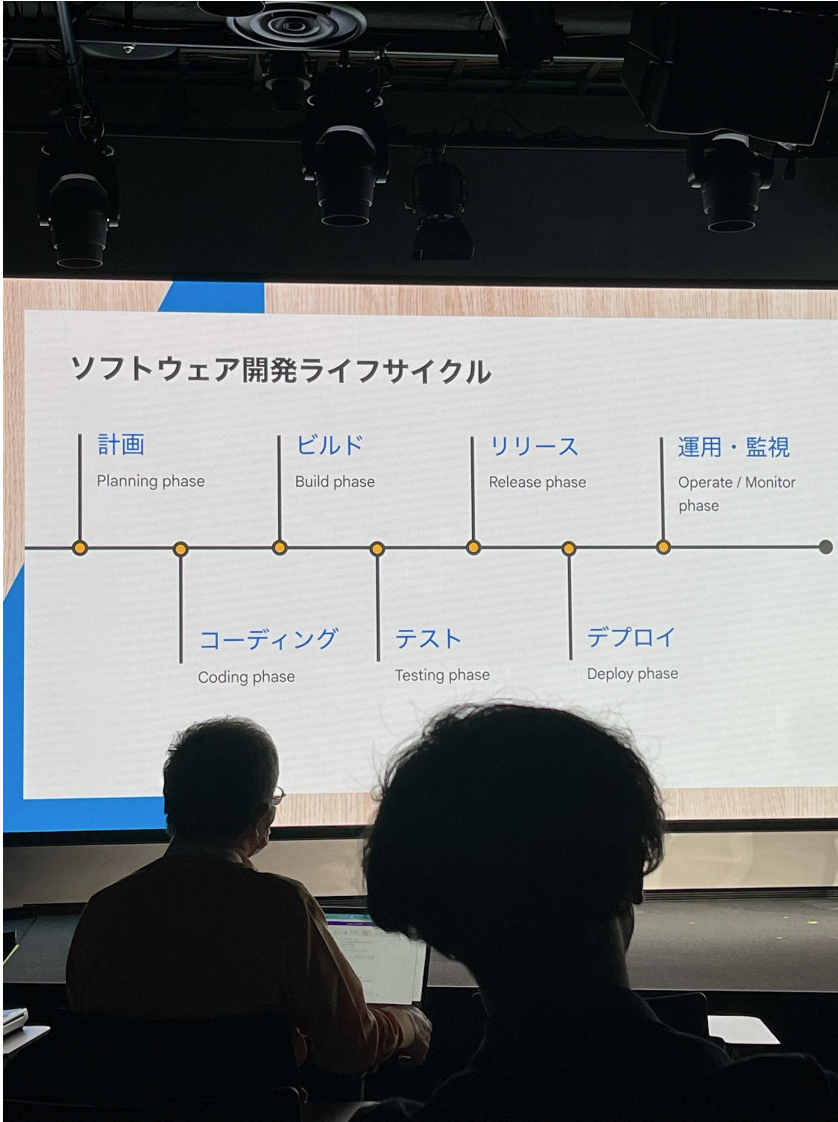
「なっている」ドリブン

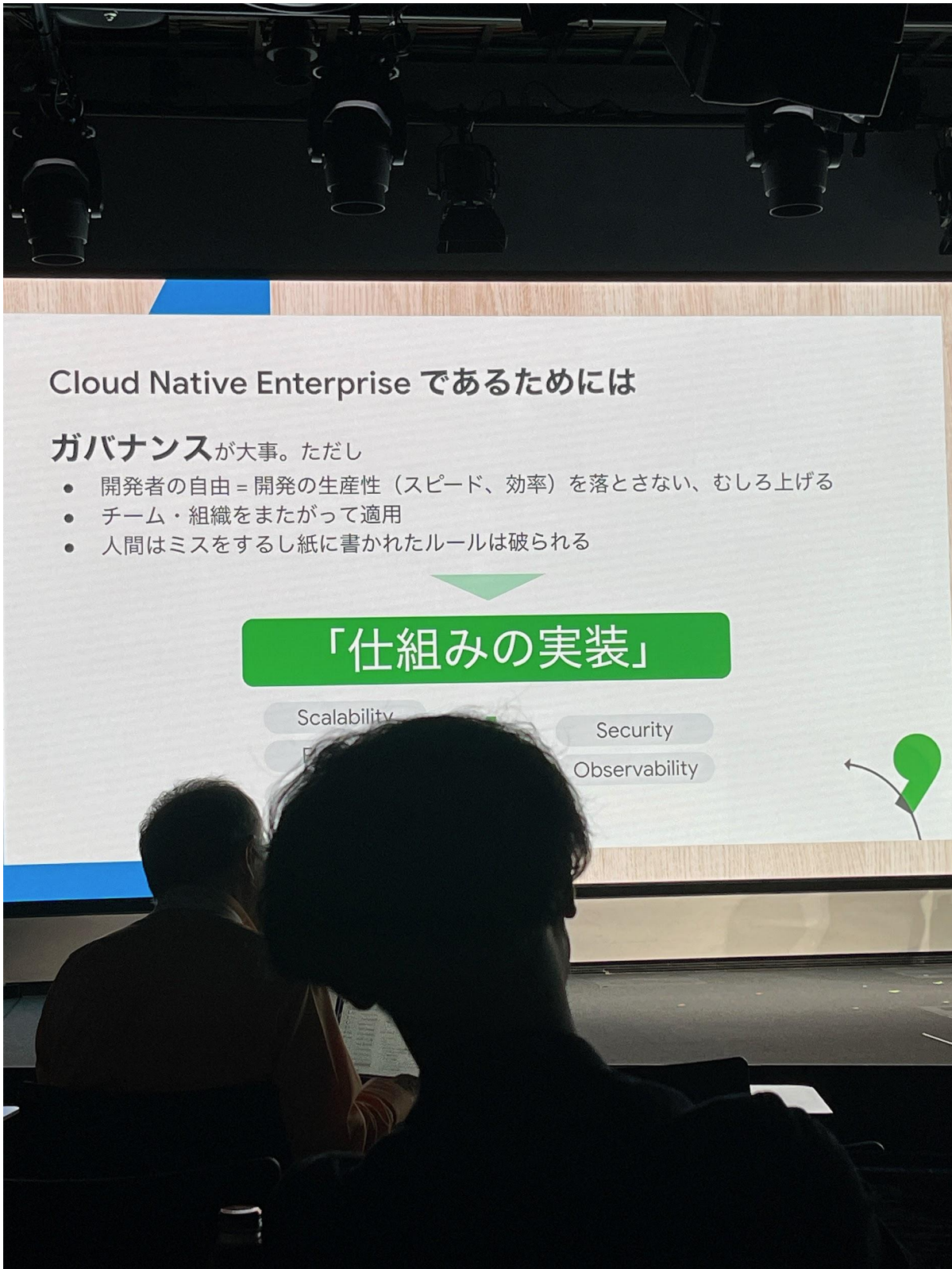






「完璧を目指すよりもまず終わらせろ」Mark Zuckerberg(フェイスブック創業者)





人間はミスするし、ルールを破る

## 仕組みを実装する上での考慮

### クラウド リソースの利用

- ポリシーの適切な設計と適用、一元化
- 複雑になりすぎてもいけない

### laC

- そのコードは本当に Single Source of Truth か？
- Configuration と State の分離

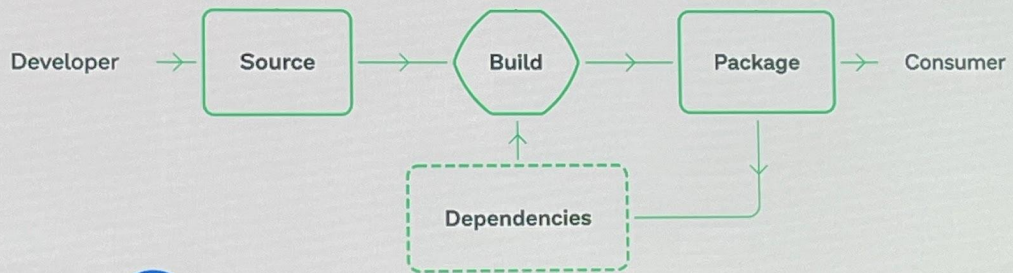
### CI / CD

- ソースコードからデプロイまで一貫したセキュリティ

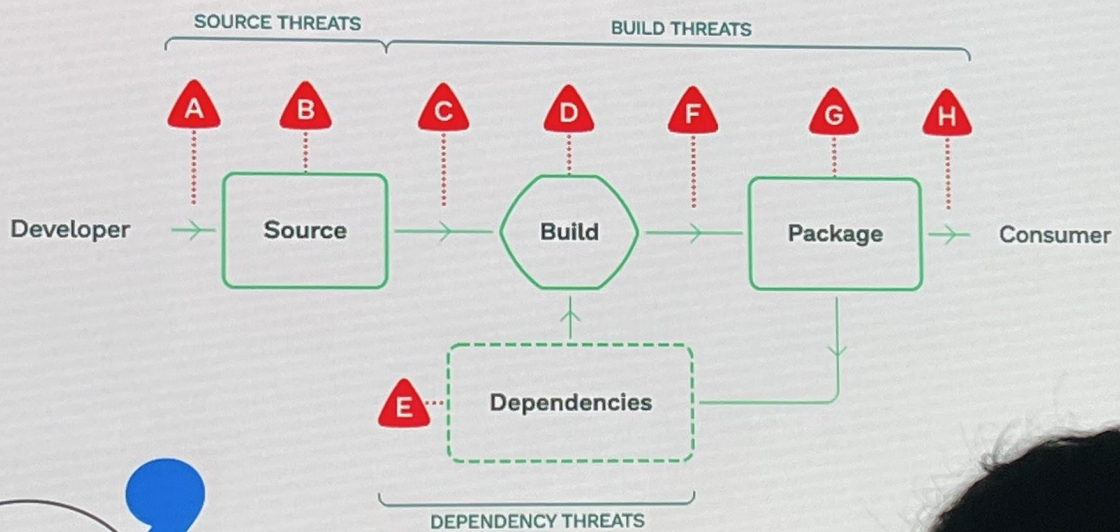
### Visualization / Reporting

- 仕組みが正しく機能していることを誰もが納得し  
安心できなければならない

## ソフトウェア サプライチェーン



# ソフトウェアサプライチェーン



セキュリティがますます重要に

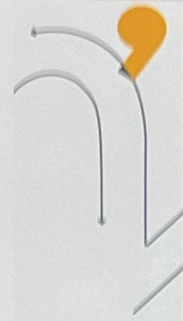
## ソフトウェア サプライチェーン セキュリティ

**650%** OSS サプライチェーン攻撃の急増\*

**84%** エンタプライズコードベースはOSSを含む\*\*

**45%** の組織は2025までサプライチェーン攻撃を受ける予想\*\*\*

\*Sonatype \*\*Synopsys \*\*\*Gartner







## Google Cloudとの出会いと技術選定

- コンテナベースアーキテクチャ  
コンテナベースオリエントドで考えた時にGoogleCloudが挙げられた。  
(Cloud Run等)

### ベンダーロックイン回避

k8sが使えるほど人員がいない

ステートフルなワークロードはさほどない  
それらが増えてきたら、他の技術検討が必要になる

- **BigQuery**が理由  
50人の社員全員がクエリを叩ける(営業の人も)  
採用基準にクエリが叩けるということもない

- 従量課金、**Firestore**を利用した認証利用  
基本ステートレスで構築  
スポットVMでコスト削減、ワークロードが終了したら、インスタンスはシャットダウン。

Cloud SQLをいま使っているが、今後はAlloyDBなども。  
Firestoreはもともとのエンジニアが使っていた。

IoT系企業なので、  
16000+/rpsを捌くのが至上命題

当初はAppEngineとDatastoreを使っていた  
0アクセスのときもあった

Cloud Run上でAPIとして実装すると、開発コストが下がる。  
pubsubのsubもcloud functionsではなく、Cloud Runで受ける。gcfでやると、監視系の構築が必要になる。

Cloud Runはポートがひとつなのが問題。  
HTTPSサービスをホストするが、たとえば監視系サービス用にポートを開けることはできない。  
外部接続用ポートは443のみ。他のポートは応答しない。

<https://qiita.com/shuichiro/items/8e3f3e870481cac9aec9>

## Google Cloudの評価点

- サービスがリリースされたらすぐ試せる
- スポットVMで格安にできる
- コンテナのフルマネージドサービスがある

