

# 失敗しないAIトランスフォーメーションの 検討ステップと実現手法

日本ヒューレット・パッカード株式会社  
Pointnext事業統括  
テクノロジーアーキテクト部  
惣道 哲也

# 自己紹介: 惣道 哲也 (そうどう てつや)

## ✓所属

日本ヒューレット・パッカード株式会社

Pointnext事業統括 テクノロジーアーキテクト部

## ✓職務領域

オープンソースに関する調査・検証・提案・構築 (OSSに関すること何でも)

主にCloud/Container/AI&DataAnalyticsなどに関する技術アーキテクト活動

## ✓経歴

ソフトウェア開発R&D部門

通信・金融・放送業界などのお客様向け案件への従事 (インフラ構築・アプリ開発)

2012年11月よりOSS技術を専門にする今の部署へ

## ✓その他活動領域

HPE社内技術コミュニティのJapan Lead

外部講演、書籍執筆など

JDLA Deep Learning for GENERAL 2019 #1 / NVIDIA Deep Learning Institute Certified Instructor



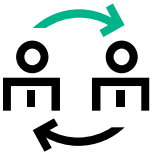
# 本日本話する内容 (40分)



① AI/データ分析に必要な5つのステップ



② コンテナ基盤を活用するメリット



③ AI/データ分析基盤のセルフサービス化に向けて(デモ)



④ まとめ・サービスのご紹介



# 本日のサマリ: AIトランスフォーメーションで検討すべき3項目

## 「3つのP」の観点



### Process

全体像の把握と  
5つのステップの理解

ビジネス成果を追求するエンタープライズにおいては、やみくもに始めるのは悪手

ビジネスゴール・ロードマップの策定、データ収集、といった一見泥臭いステップをきちんと踏むことが重要

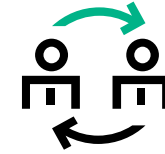


### Platform

コンテナを活用した  
統合基盤の構築

扱うデータは種類、量とも大量になり、さまざまな処理・加工を柔軟、かつ、迅速に行う必要がある

各処理を行うシステムをサイロ化せずに、コンテナを活用した統合化・近代化を目指すべき



### People

セルフサービス化による  
データ分析の民主化

コンテナ基盤の活用により、End-to-Endの分析タスクは分析者自身が「セルフサービス」で準備・実施が可能に

迅速なフィードバックが得られることにより大きな成果につながる



# はじめに

～ The state of AI in 2020 ～



# 人工知能の進化 (1/4)

## AlphaGo (2016)

- 韓国のイ・セドル九段と対戦し、4勝1敗で勝利(2016年3月)
  - AlphaGoは人間のプログラミングによって打ち方のルールを設計されたわけではない
  - 機械学習のテクニックを使って、過去の膨大な棋譜をもとにどうやって囲碁の試合に勝つか学んでいく



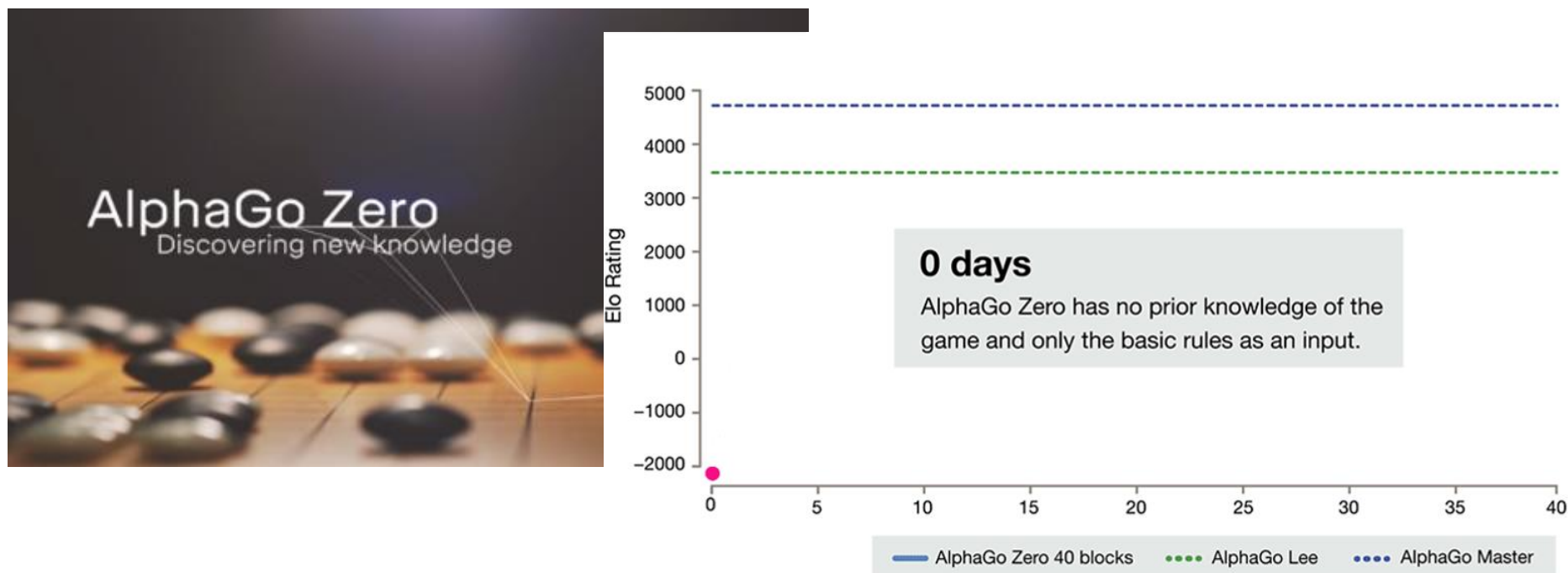
Google社のAlphaGoが囲碁の対戦でイ・セドル九段に勝利

# 人工知能の進化 (2/4)

## AlphaGo Zero (2017)

### – AlphaGo Zeroの登場 (2017年11月)

- これまでのAlphaGoが過去の人間の棋譜を入力としていたのとは異なり、完全に自己学習のみを行う方式。人間を真似ることをやめることで人間の限界を超えた。



<http://www.zdnet.com/article/deepmind-alphago-zero-learns-on-its-own-without-meatbag-intervention/>

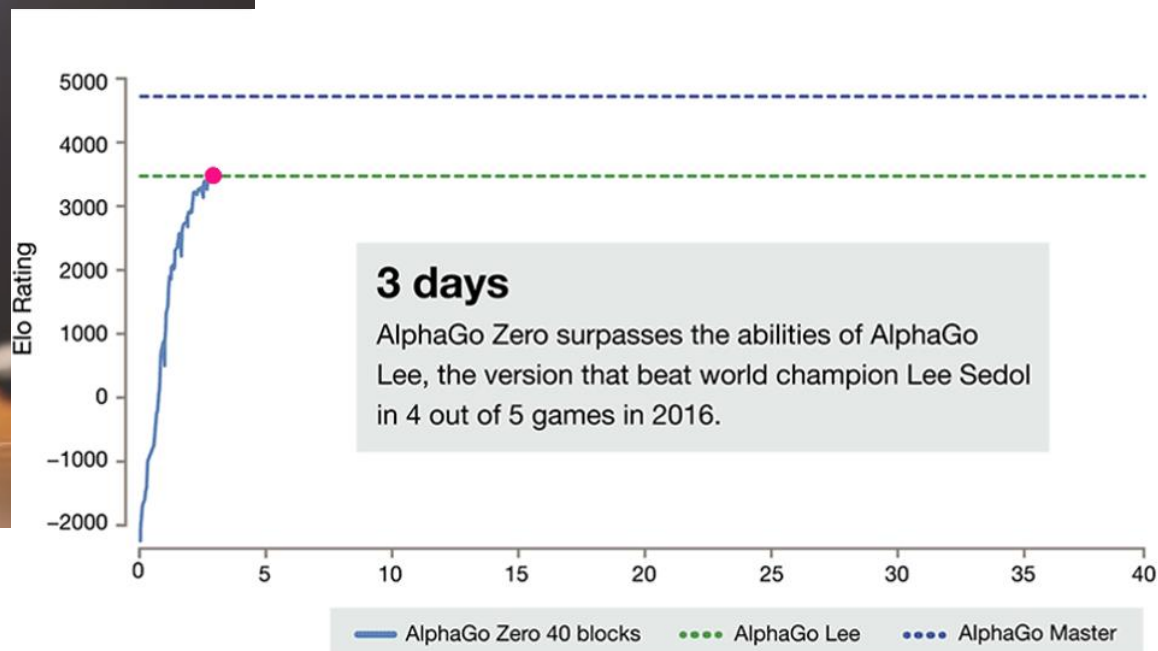


# 人工知能の進化 (2/4)

## AlphaGo Zero (2017)

### – AlphaGo Zeroの登場 (2017年11月)

- これまでのAlphaGoが過去の人間の棋譜を入力としていたのとは異なり、完全に自己学習のみを行う方式。人間を真似ることをやめることで人間の限界を超えた。



<http://www.zdnet.com/article/deepmind-alphago-zero-learns-on-its-own-without-meatbag-intervention/>



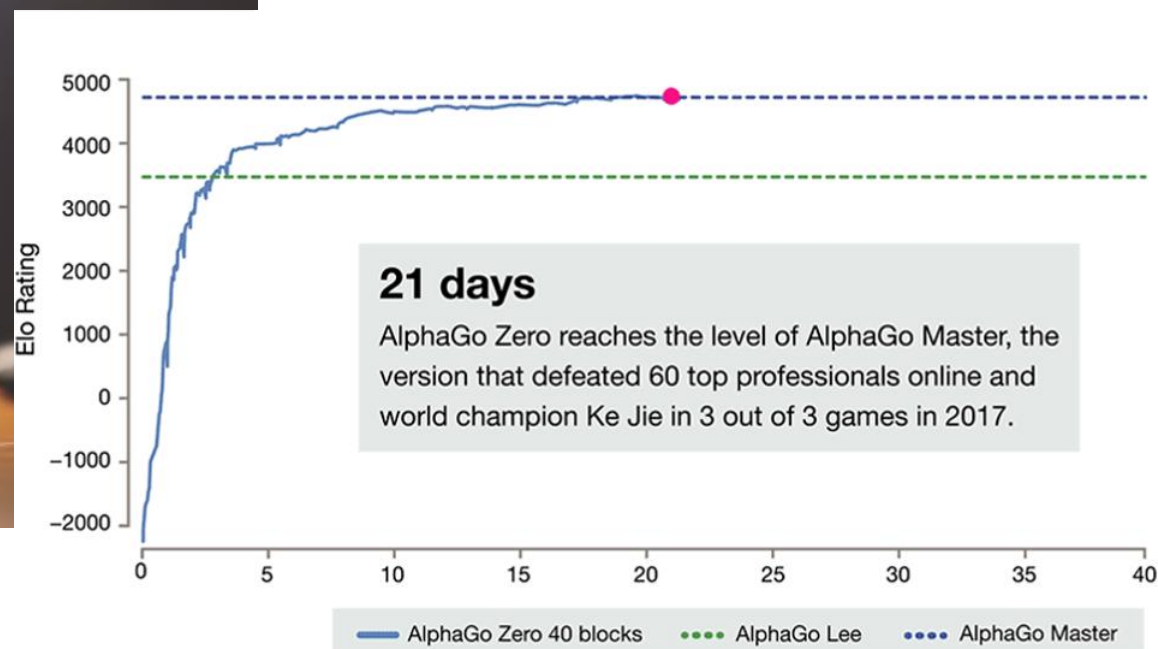


# 人工知能の進化 (2/4)

## AlphaGo Zero (2017)

### – AlphaGo Zeroの登場 (2017年11月)

- これまでのAlphaGoが過去の人間の棋譜を入力としていたのとは異なり、完全に自己学習のみを行う方式。人間を真似ることをやめることで人間の限界を超えた。



<http://www.zdnet.com/article/deepmind-alphago-zero-learns-on-its-own-without-meatbag-intervention/>

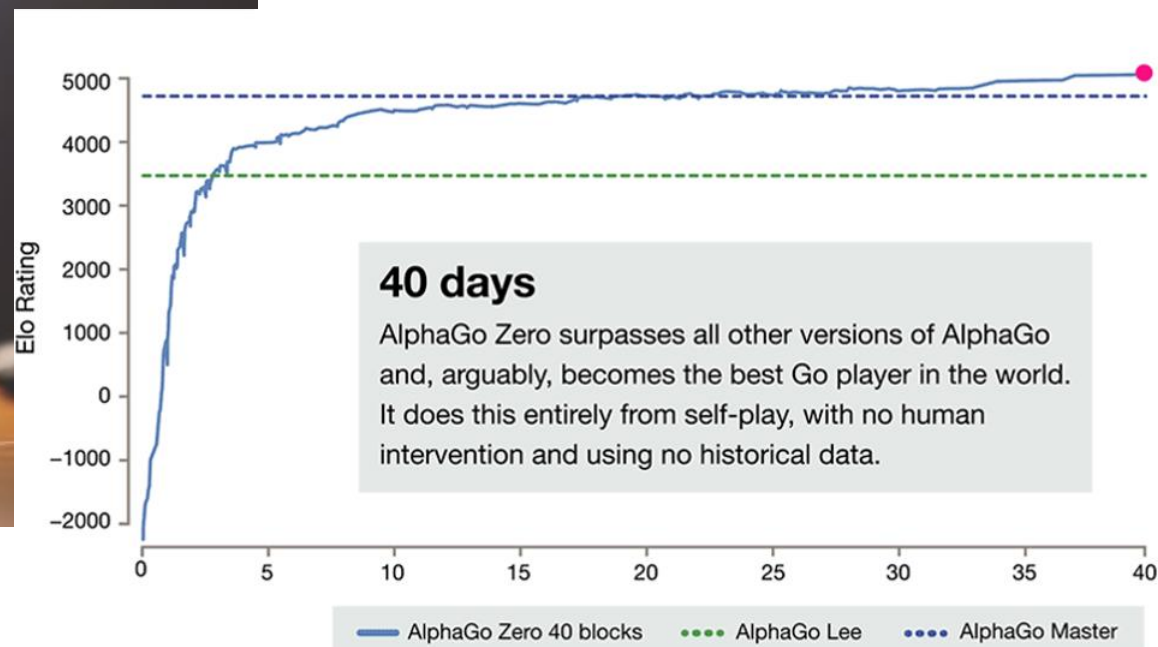


# 人工知能の進化 (2/4)

## AlphaGo Zero (2017)

### – AlphaGo Zeroの登場 (2017年11月)

- これまでのAlphaGoが過去の人間の棋譜を入力としていたのとは異なり、完全に自己学習のみを行う方式。人間を真似ることをやめることで人間の限界を超えた。



<http://www.zdnet.com/article/deepmind-alphago-zero-learns-on-its-own-without-meatbag-intervention/>



# 人工知能の進化 (3/4)

## AlphaZero (2018)

### – さらに汎用化されたAlphaZeroへ (2018年12月)

- AlphaZeroは、チェスを9時間学習した後チェスの世界チャンピオンAI「Stockfish」を、将棋を12時間学習した後世界コンピュータ将棋選手権の2017年の勝者「elmo」に、囲碁を13日間学習した後AlphaGo Zeroに勝利した
- 「AlphaZeroの目的はチェスや将棋、囲碁に勝つことの先にあり、様々な実世界の問題を解決するシステムを構築することにある」



# 人工知能の進化 (4/4)

## Microsoft Suphx (2019)

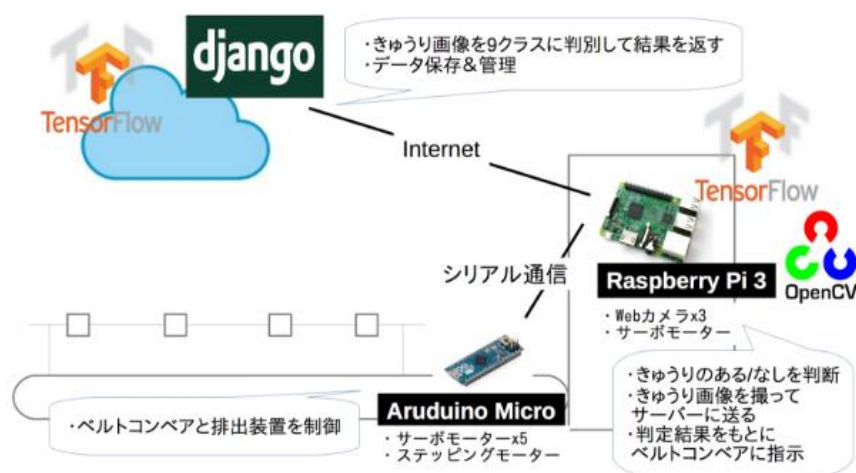
- 相手の手がわからない「**不完全情報ゲーム**」でもトッププレイヤー並みの強さを獲得 (2019年9月)
  - Microsoft Research Asiaは、「Suphx(スーパーフェニックス)」という麻雀AIを開発し、オンライン麻雀サービス「天鳳」で、最高位にあたる「十段」(180人/330,000人)を獲得した。
  - 相手の手のように隠された不確定情報を含めて学習できるAIは金融取引など現実的な応用も期待される。



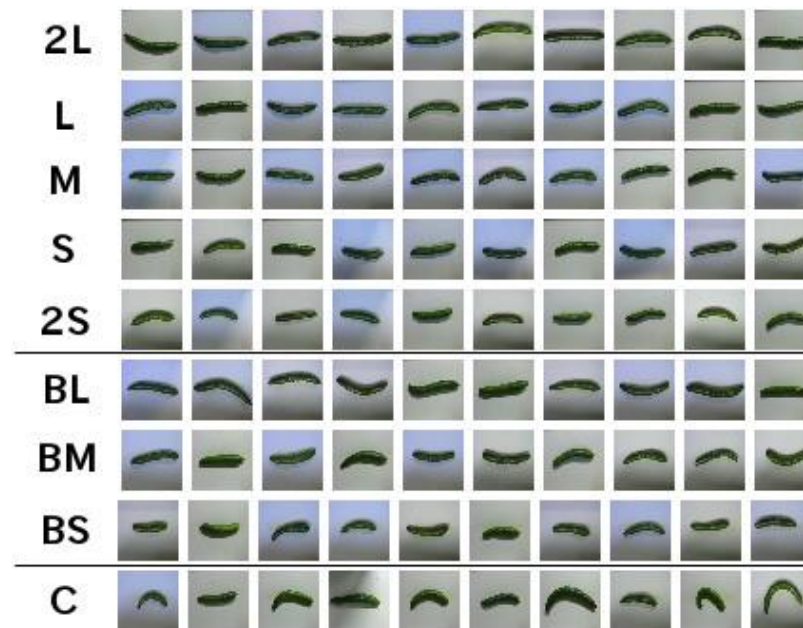
<https://www.businessinsider.jp/post-197865>

# AIの民主化 (1/2): 農家におけるキュウリの自動仕分け

- 農家でのキュウリの仕分け(出荷時に形や色合い、大きさにより選別する作業)を自動化する例
  - Webカメラで画像撮影し、画像をディープラーニングによる認識エンジンに送り、排出装置で振り分ける
  - 精度は80%以上



試作機のシステム構成図



大きさ・形により等級を決定する

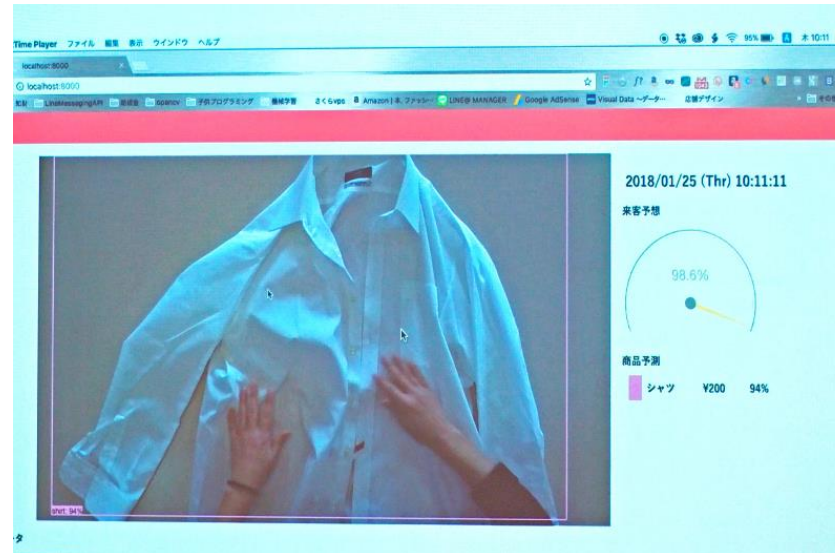
[http://googlecloudplatform-japan.blogspot.com/2016/08/tensorflow\\_5.html](http://googlecloudplatform-japan.blogspot.com/2016/08/tensorflow_5.html)

# AIの民主化 (2/2): クリーニング店における商品分類

- クリーニング業界全体の市場縮小に直面し、無人での商品分類を目指して地方クリーニング店の店主が個人でAI開発
  - 店に設置したカメラからリアルタイムで画像を取得し、24種類の商品に自動分類を行う
  - 開発費はPCとタブレットなど50万円以内におさめる



テーブルに商品を置くとAIが分類・検知する



学習済みモデルによりシャツと識別された例

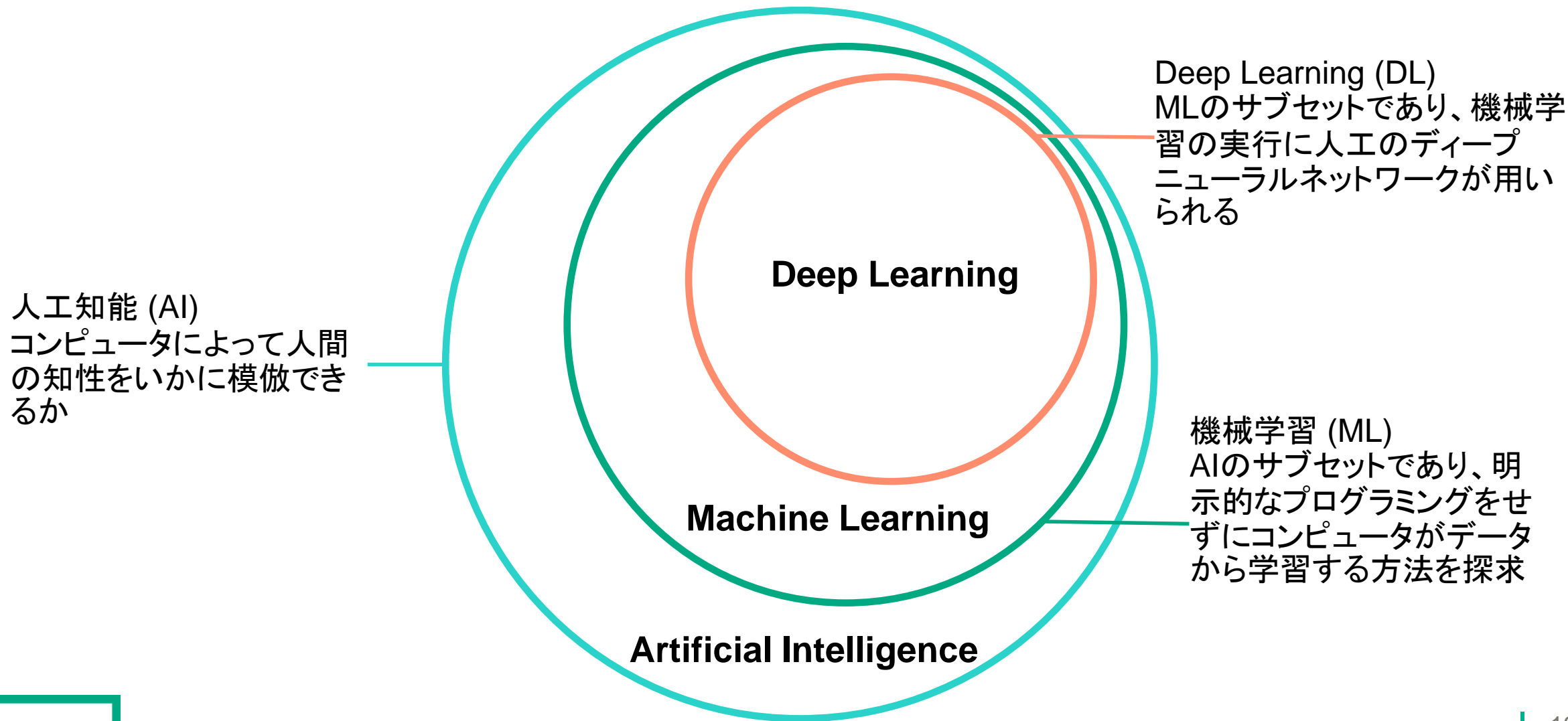
<http://www.itmedia.co.jp/enterprise/articles/1804/12/news021.html>



# 1. AI/データ分析に必要な5つのステップ

～ The 5 steps of “enterprise data analysis” ～

# 人工知能(AI)、機械学習(ML) と Deep Learning(DL)





# ルールベースのAIから、機械学習、ディープラーニングへ

タスク: 住宅価格を近隣の学校のランク(s)、寝室(be)とトイレ(ba)の数、広さ(f)から予測する

## ルールベース AI

専門家がルールを定義し、明示的にプログラムで表現する:

```
if (s==9 and be==2
    and ba==2 and f==1000)
then
    price = $1000000;
```

```
else if (...) then ...
else if (...) then ...
```

## 機械学習

ラベル化されたデータセットを集める

```
House 1: s = 9, be = 2,
          ba = 2, f = 1000,
          price = $1000000
```

```
House 1: s = 4, be = 2,
          ba = 1, f = 700,
          price = $600000
```

関数(モデル)を定義する

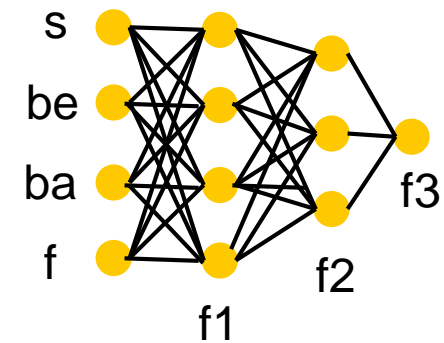
```
F: (s,be,ba,f) -> price
price = F(s,be,ba,f) =
w1*s + w2*be + w3*ba + w4*f
```

モデルを使い推測する:

w1~w4の最適解を求める





## Deep Learning

機械学習と同様だがより複雑  
(「関数の関数」の形式へ)






```
F: (s,be,ba,f) -> price
price = F(s,be,ba,f) =
f3(f2(f1(s,be,ba,f)))
```

# Machine Learning と Deep Learning の違い

 データベース ステータブル	 統計データ
 センサ データ	 ログデータ

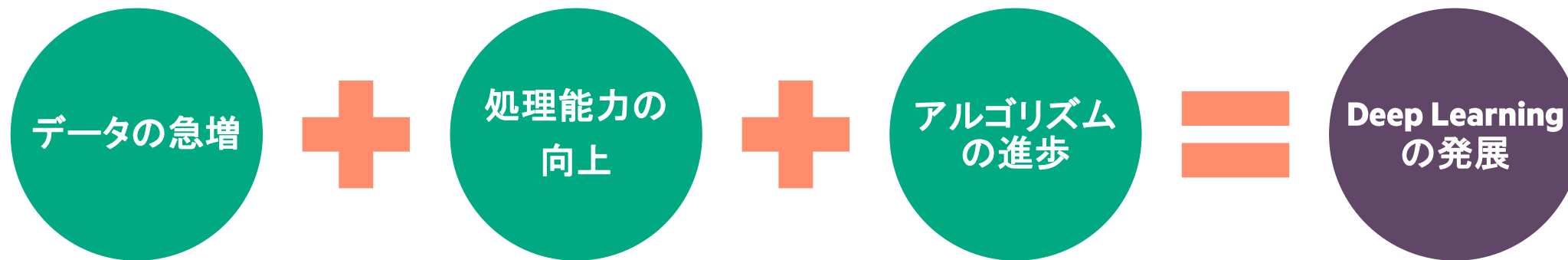
**Machine Learning**  
(主に)構造化データ

 画像	 テキスト
 ビデオ	 スピーチ

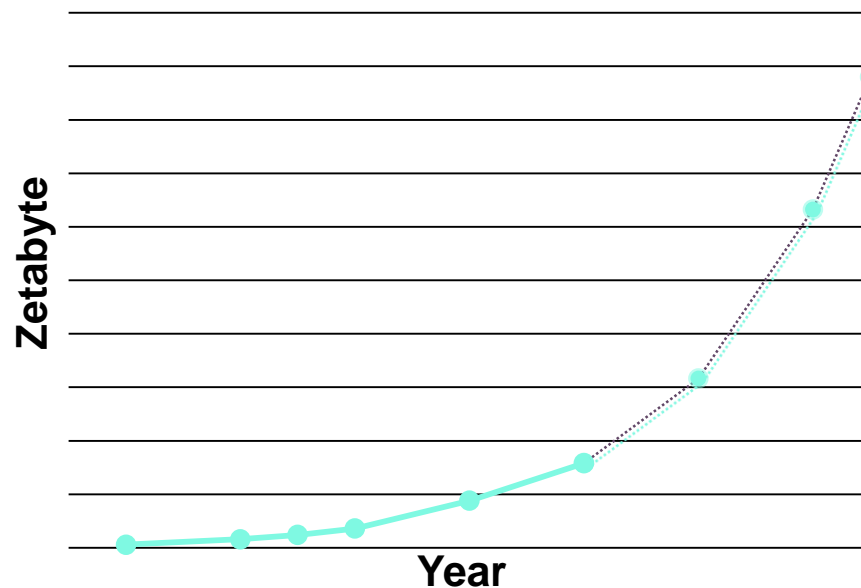
**Deep Learning**  
(主に)非構造化データ



# Deep Learningの発展の背景



データ量の指数関数的な増大



# AIの応用例



予知保全



自律運転



アルゴリズム取引



信用格付け



レコメンドシステム



チャットボット



不正検知










監視



# 目的とスコープを定める

- ユースケース分析
- データソースの特定
- Critical success factorと制約について議論しておく
- 期待値に関する合意
- ロードマップを描いておく

Image	Video	Data	Web Log	Text	Sensor	Transaction
						
<b>Healthcare:</b> Radiology Diagnosis	<b>Smart City:</b> Video Surveillance	<b>Agriculture:</b> Crop Disease Study	<b>Marketing:</b> Recommender System	<b>Media:</b> Sentiment Analysis	<b>Oil Refinery:</b> Predictive Maintenance	<b>Banking:</b> Fraud Detection

# データ取得



- 組織内のデータ資産
- ユーザ生成データ
- 研究用の無償データ
- 商業目的でのデータ取得



# データ前処理

## Image Preprocessing

- 解像度調整
- アスペクト比調整
- ピクセル正規化

## Text Preprocessing

- 単語分割
- タグ付け
- 語幹抽出
- ストップワード除去
- 大文字小文字、ひらがなカタカナの名寄せ
- 未知語の扱い



# データラベリング



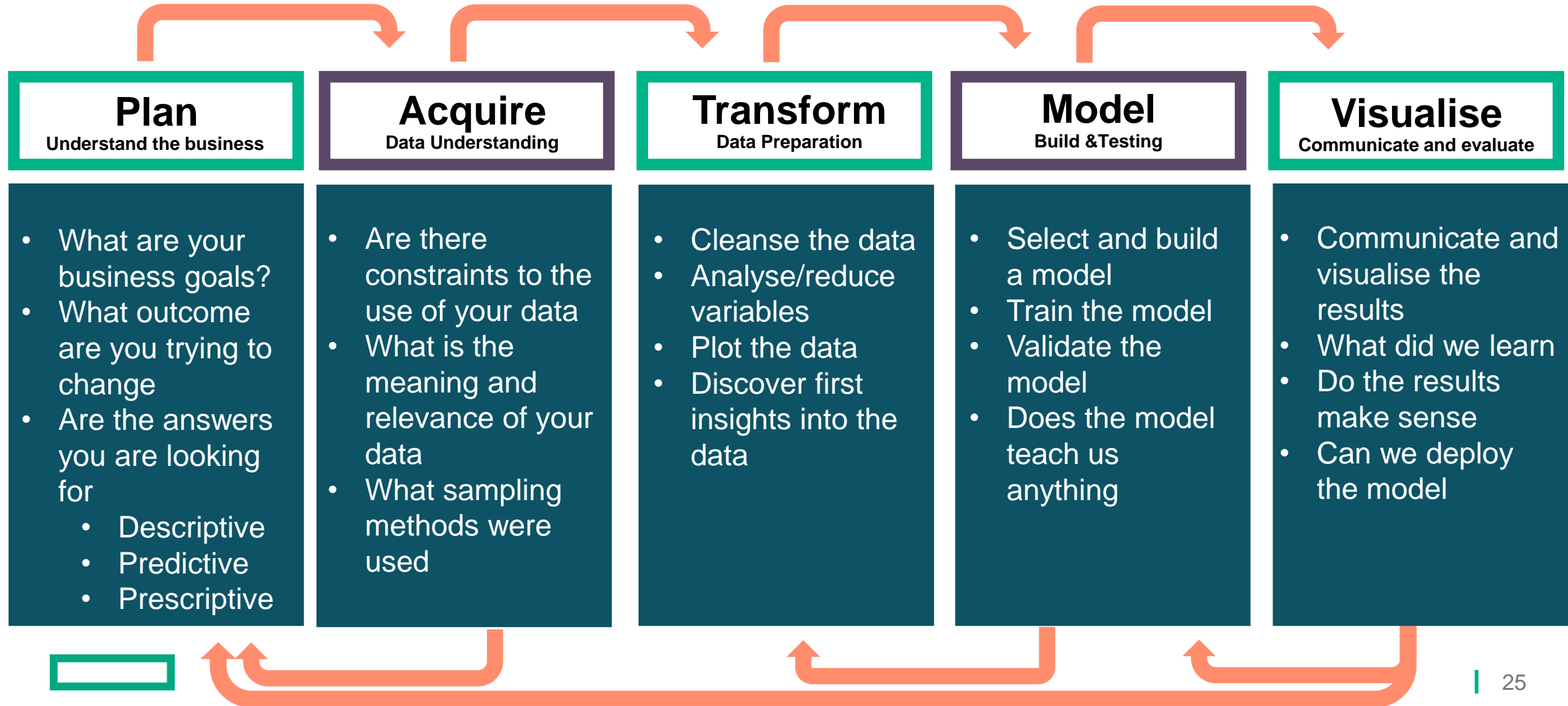
- 専門家の助言
- ユーザ生成ラベル
- ラベル付きデータセットの利用
- 外部へのアウトソース






# データ分析パイプラインの「5つのステップ」

Based on Crisp-DM  
and data Harvard data science course





## 2. コンテナ基盤を活用するメリット

～ Container and Kubernetes as modern platform ～



# コンテナ活用のユースケースは幅広く存在



[Gartner, Mar. 2016](#)

コンテナのメリットは  
セルフサービスの実現

## CICD Automation

- Seamless application development and deployment could be accelerated and automated by the use of Container
- **Benefit** : Agility, IP Protection



## Replacement of Virtualization

- Use container technology instead of virtualization to minimize infrastructure tax
- **Benefit** : Performance, TCO Optimization, Flexibility



## AI / Deep Learning Platform

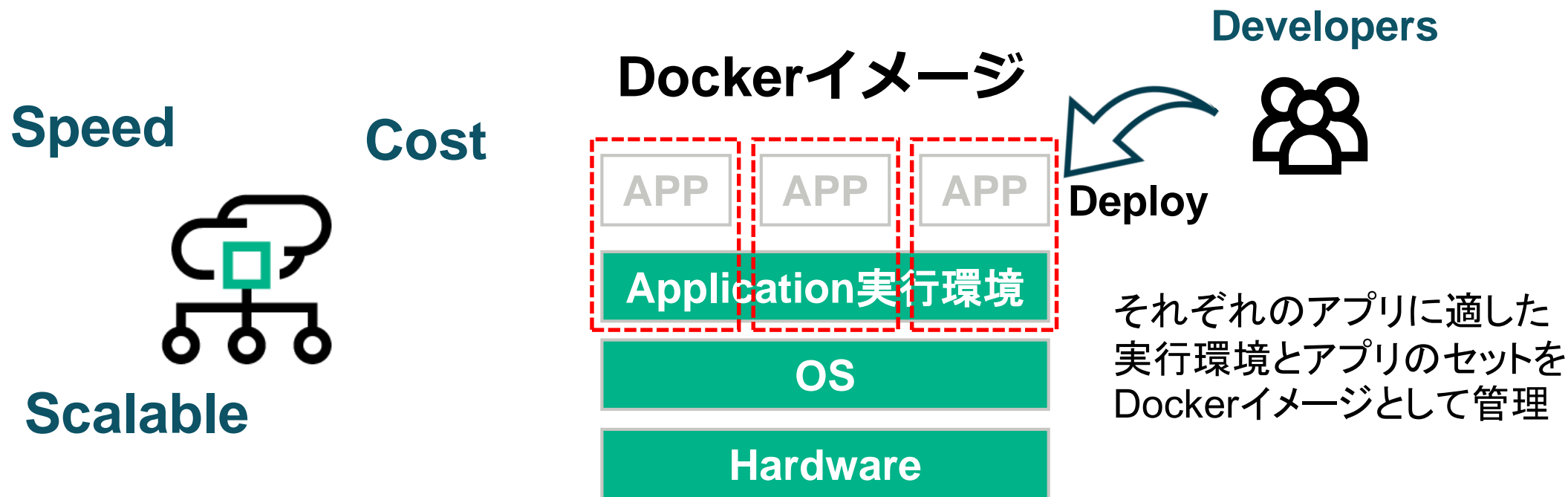
- Higher utilization and performance of AI / DL frameworks such as Tensorflow could be provided
- **Benefit** : Efficiency, Agility, Innovation



コンテナ活用により、データ分析の「民主化」が可能に

# Dockerの生まれた背景

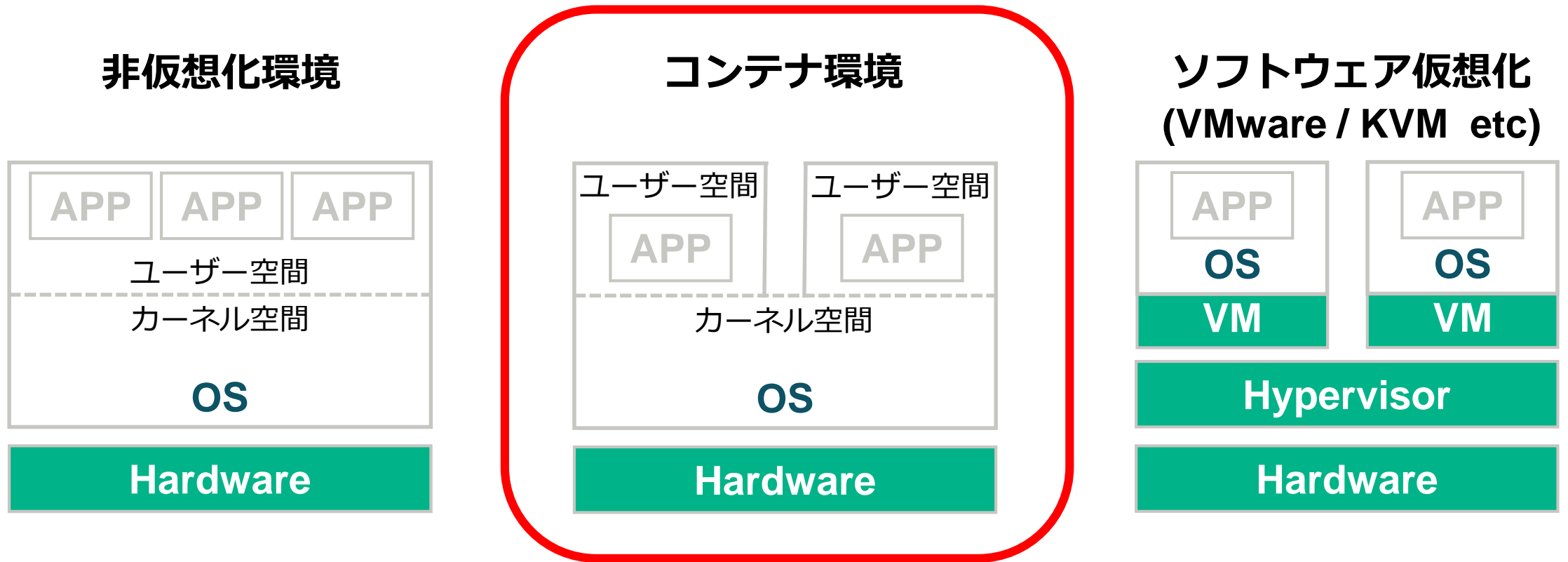
アプリ(APP)は実行環境(ライブラリ、ミドルウェア等)と密結合している



アプリケーションと実行環境を「Dockerイメージ」に固める技術が開発された

# Dockerコンテナ環境の中身

コンテナはユーザー空間プロセスを隔離している



ユーザ空間を複数に分けて、それぞれのプロセス(アプリケーション)から見えるリソースを制限する

# エンタープライズアプリケーション基盤の要件

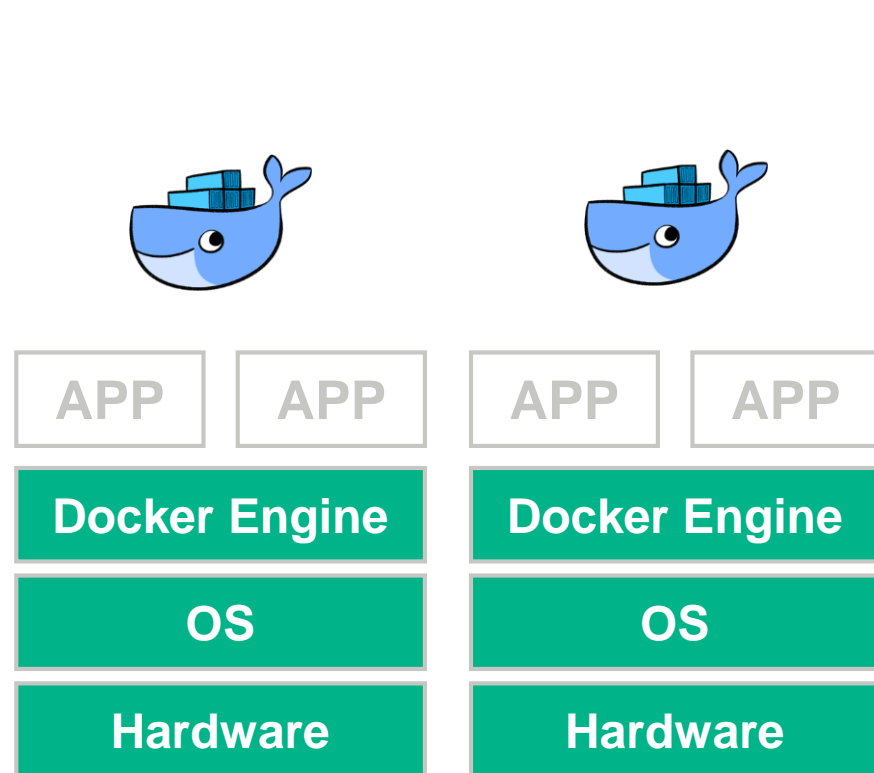
少なくとも以下のような考慮が必要

- スケーラビリティ
- 可用性
- 既存のインフラとの相互運用性
- データ保全
- セキュリティ
- ライフサイクル管理(パッチ、アップグレード)
- 認証/認可
- 監視
- ログ管理
- 運用体制
- サポート

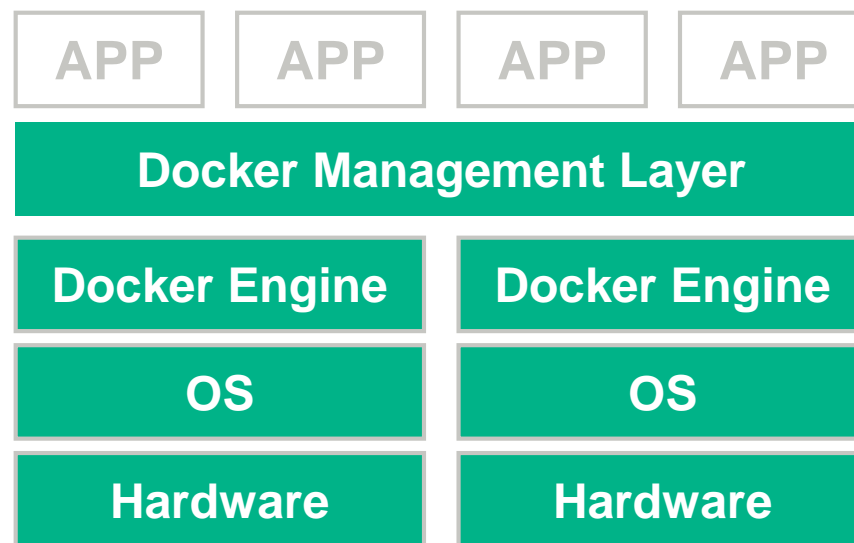


# Kubernetesとは

Dockerを分散環境で構築するためのオーケストレーションツール



素のDocker = 単体ノードの管理

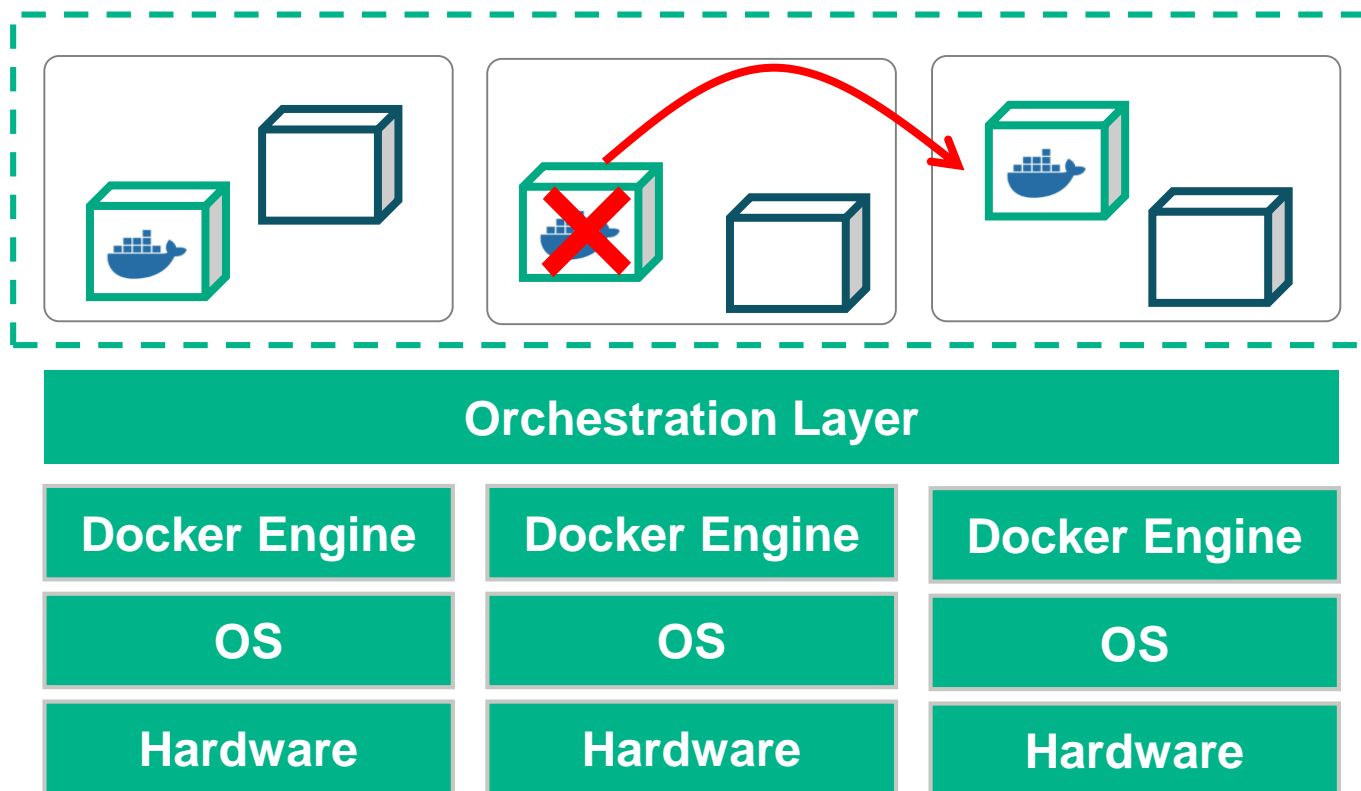
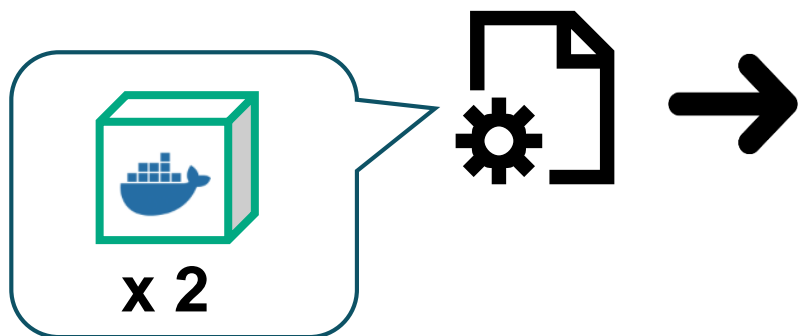


複数ノードに分散したノードを  
対象とした運用管理レイヤ

# Kubernetesとは

Dockerを分散環境で構築するためのオーケストレーションツール

- ・複数のノードを一つのサーバのように管理できる
- ・コンテナはどこで動いていてもOK
- ・レプリカ数のスケールアウトも容易
- ・レプリカ数を一定に保つよう監視

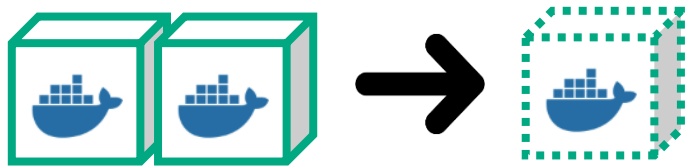




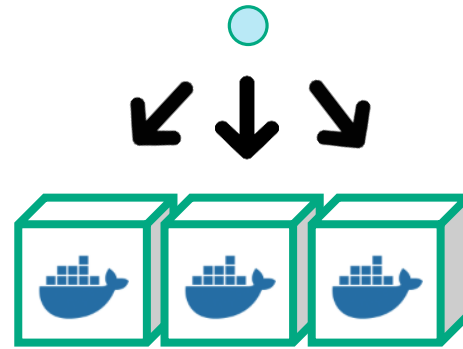
# Kubernetesとは

主要な機能を整理すると...

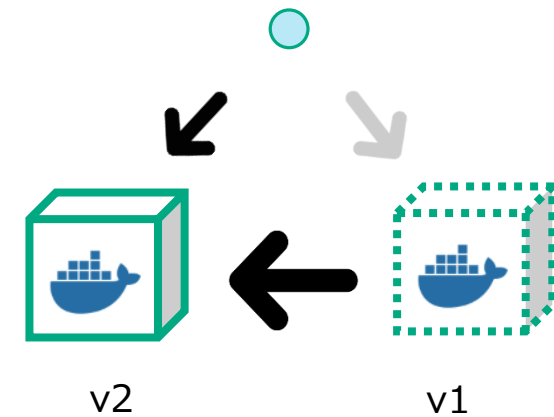
## Workloads



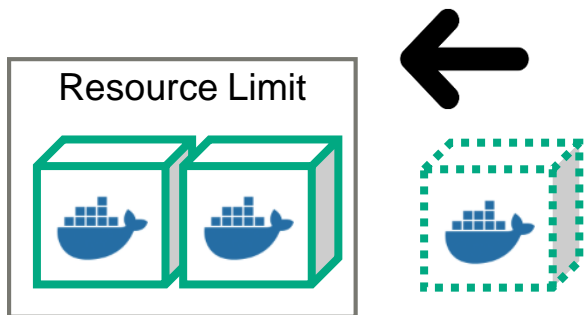
## Load Balancing



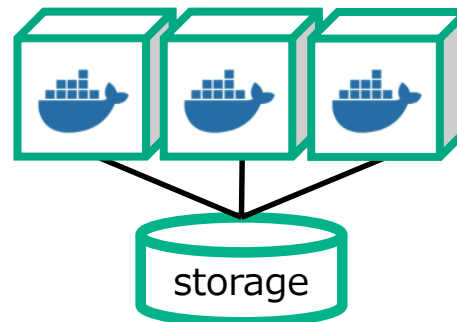
## Rolling Update



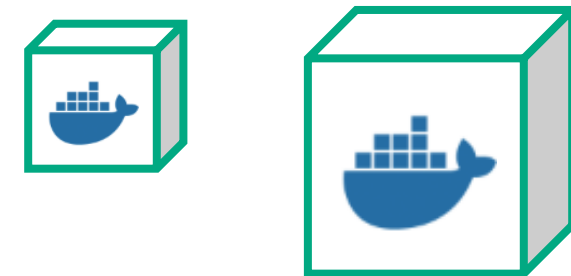
## Policy Mgmt



## Storage/NW Mgmt



## Resource Mgmt



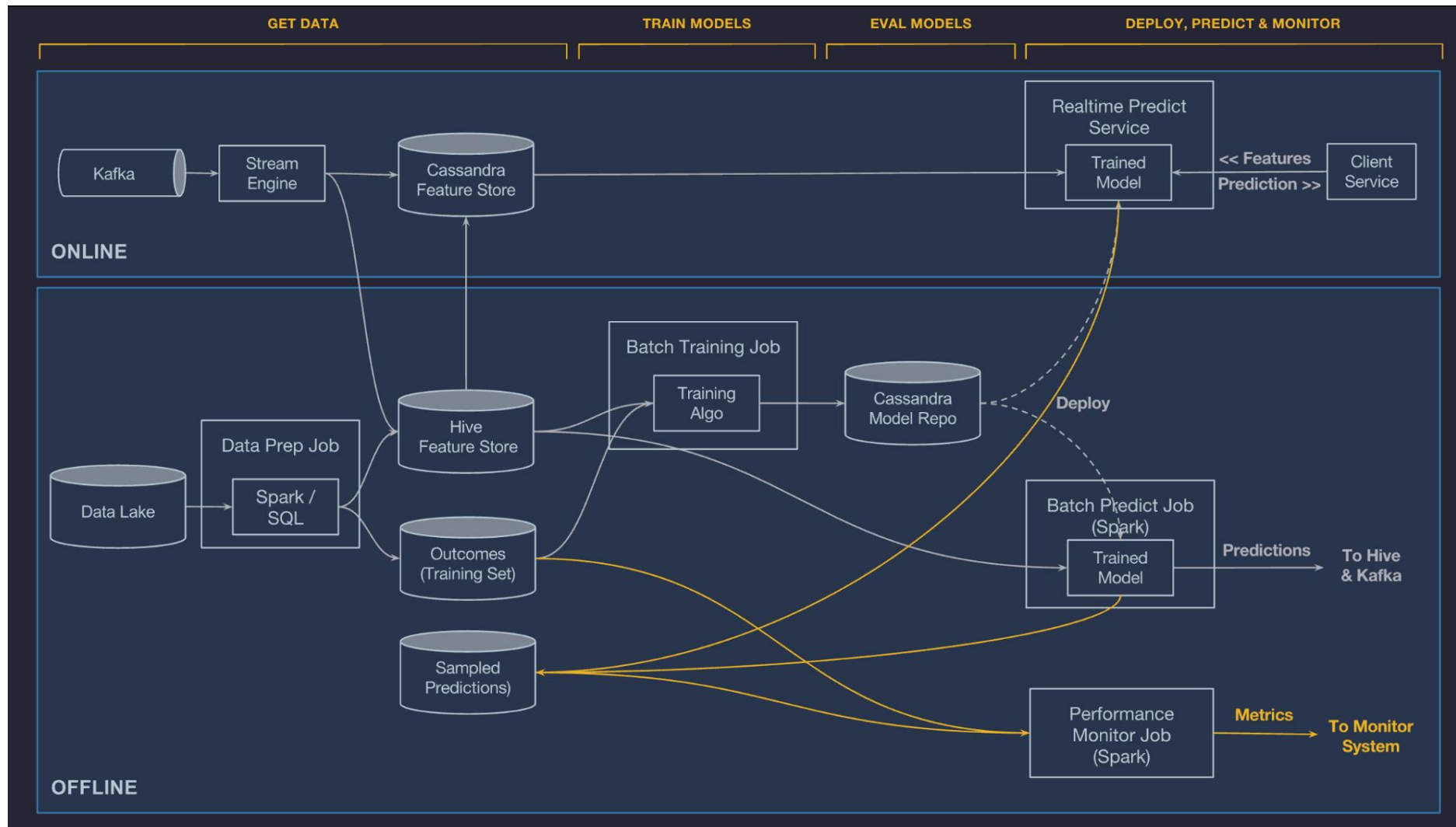
# 3. AI/データ分析基盤のセルフサービス化に向けて

～ “MLaaS” for enterprise data analysis ～



# MLaaSの例: Uber

<https://eng.uber.com/michelangelo/>



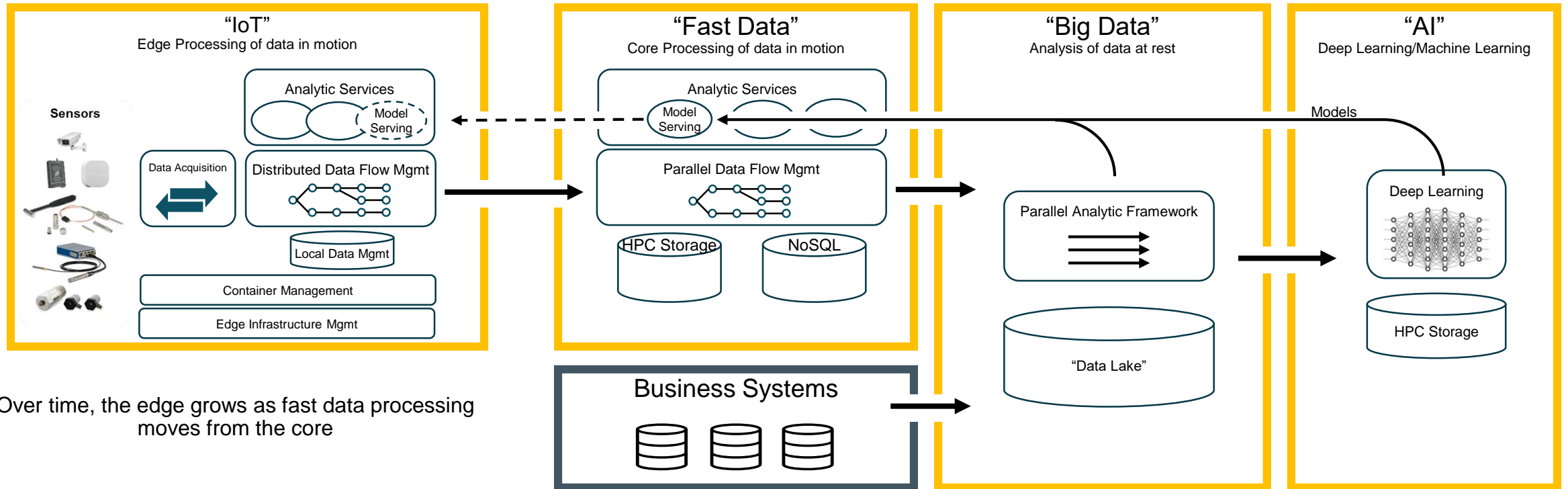
# An End to End Data Pipeline

## Functional View

Services and Solutions

Data Science Toolchains

Data Flow Design, Data Science Workbench, Model Management, Application Deployment



Over time, the edge grows as fast data processing moves from the core


On Prem

On Prem or Cloud

# An End to End Data Pipeline

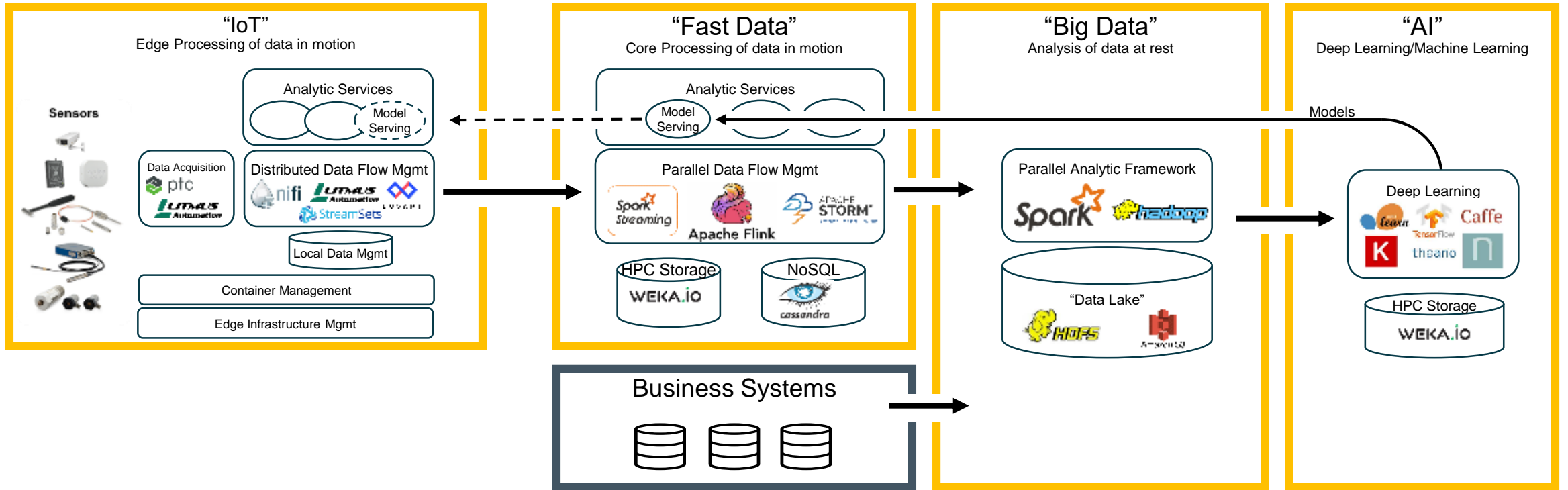
## Applications View

### Services and Solutions

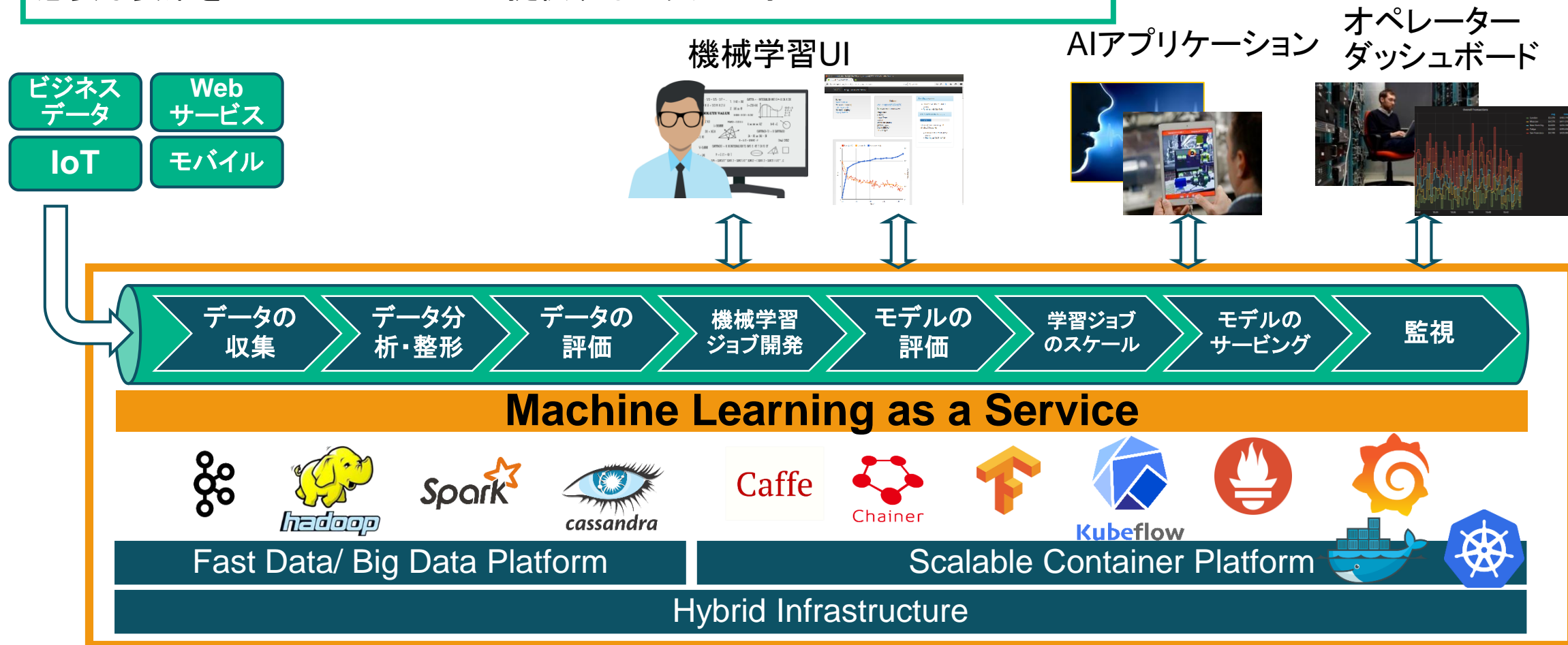




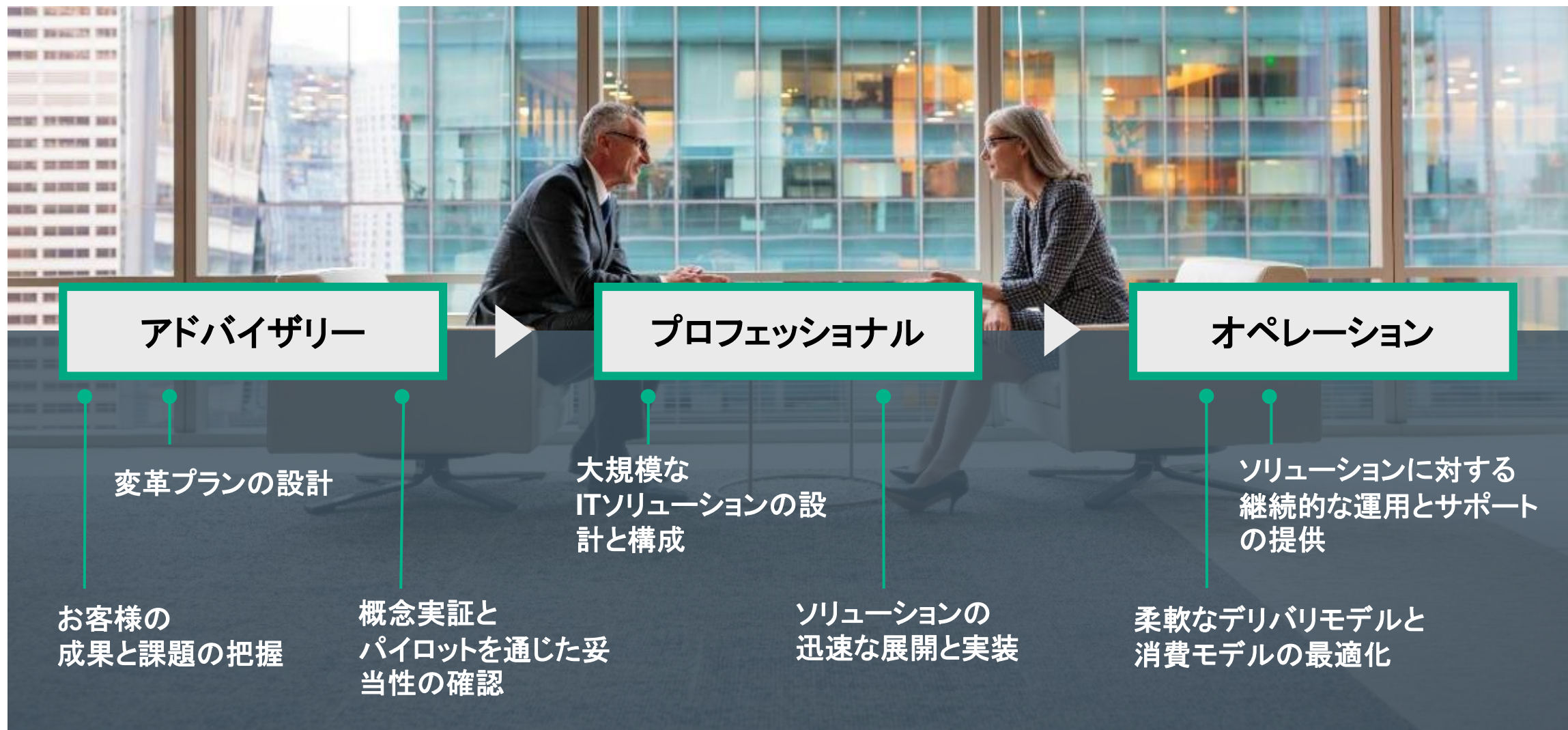
# 企業のデータ分析者に必要な”MLaaS”とは？

データの入り口から学習済みモデルのビジネスアプリケーションへの応用まで、必要な要素を”as a Service”として提供するプラットフォーム



## 4.まとめとHPEサービスのご紹介

# お客様を支援する専門知識 – HPE Pointnext –





# HPE AIワークショップの概要

## 内容



1日ワークショップを通じて、AI/ML、データ分析のトピックをカバー

### ベネフィット

- AIプロジェクトの迅速なスタート
- ビジネス、データ管理、ITチームを連携
- 優先度に応じて関連したユースケースを選択
- データソースをもとに依存性や利用可否を判断
- 顧客のインテリジェントなデータ戦略のためのハイレベルロードマップを策定

# HPE AIワークショップの概要

## 成果物

### 1. Use case analysis

重要な目的と戦略を定める



目的

### 2. Outcomes and scope

目的のためにどのような気づきと判断が必要か？



気づきと判断

### 3. Data Sources & Project Roadmap

気づきと判断のためにどのような情報が必要か？



情報

# Deep Learning開発用コンテナ環境構築サービス

## 概要

本サービスは、Deep Learning開発環境としてNVIDIA GPUを搭載したHPE Apollo 6500 Gen10 サーバーをRed Hat OpenShift Container Platform (以降OpenShift)のNodeサーバーとして構成し、その上でTensorFlowやChainer等のDeep Learningアプリケーションコンテナを稼働する環境を導入します。OpenShiftの特長でもある堅牢なテナント分離を実現するだけでなく、特定のGPUリソースを柔軟に各利用者のコンテナに割り当てることが可能となる、リソース、開発環境構築時間、コスト等あらゆる面で開発効率が飛躍的に向上しうる環境を提供いたします。

## HPEが提供するDeep Learning開発用コンテナ基盤環境



## HPE Apollo 6500 serversの特徴

HPE Apollo Systemは、ラックあたり最高レベルのパフォーマンスと効率性を実現する、Deep Learning 向けに最適化されたスケールアウト型GPUシステムです。

### 最高のGPU密度

NVIDIA Tesla GPU (PCIe もしくは NVLINK 2.0) を最大8基搭載可能

### 柔軟なストレージ構成

16本までのSATA/SAS/SSDもしくは4本までのNVMEを構成可能

### GPUパフォーマンスを活かす

1または2CPUあたり最大8GPUを実現し、アプリケーションに最適化

## OpenShiftによる開発環境のメリット

### エンタープライズ向けコンテナオーケストレーション

Dockerと採用実績豊富なKubernetesをネイティブに統合、エンタープライズ向け認証・SDN・Webコンソール・運用管理等の機能も充実

### GPUリソースのマルチテナント毎割り当てが可能

最大8基のNVIDIA GPUのワークロードに合わせた柔軟な割り当て、テナント毎のリソース分離とアクセス制御を実現。セキュリティ、パフォーマンスの双方において、開発効率が飛躍的に向上します。

### 異種・複数バージョンDeep Learningフレームワークの混在可

Deep LearningフレームワークとCUDA Toolkitライブラリをコンテナ化。複数フレームワークの開発環境を同一プラットフォーム上で利用可能です。

# Deep Learning開発用コンテナ環境構築サービス

## 概要

本サービスは、Deep Learning開発環境としてNVIDIA GPUを搭載したHPE Apollo 6500 Gen10 サーバーをRed Hat OpenShift Container Platform (以降OpenShift)のNodeサーバーとして構成し、その上でTensorFlowやChainer等のDeep Learningアプリケーションコンテナを稼働する環境を導入します。OpenShiftの特長でもある堅牢なテナント分離を実現するだけでなく、特定のGPUリソースを柔軟に各利用者のコンテナに割り当てることが可能となる、リソース、開発環境構築時間、コスト等あらゆる面で開発効率が飛躍的に向上しうる環境を提供いたします。

## HPEが提供するDeep Learning開発用コンテナ基盤環境



## HPE Apollo 6500 serversの特徴

HPE Apollo Systemは、ラックあたり最高レベルのパフォーマンスと効率性を実現する、Deep Learning 向けに最適化されたスケールアウト型GPUシステムです。

### 最高のGPU密度

NVIDIA Tesla GPU (PCIe もしくは NVLINK 2.0) を最大8基搭載可能

### GPUパフォーマンスを活かす

1または2CPUあたり最大8GPUを実現し、アプリケーションに最適化

### 柔軟なストレージ構成

16本までのSATA/SAS/SSDもしくは4本までのNVMe SSDを構成可能

## OpenShiftによる開発環境のメリット

### エンタープライズ向けコンテナオーケストレーション

Dockerと採用実績豊富なKubernetesをネイティブに統合、エンタープライズ向け認証・SDN・Webコンソール・運用管理等の機能も充実

### GPUリソースのマルチテナント毎割り当てが可能

最大8基のNVIDIA GPUのロードに合わせた柔軟な割り当て、テナント毎のリソース分離とアプリケーション毎のGPUリソースの混在可  
双方において、開発効率が飛躍的に向上します。

### フレームワークの混在可

TensorFlowやChainer等のDeep Learningアプリケーションをコンテナ化。複数フレームワークの開発

# Deep Learning コンテナ HPE

# 本日のまとめ: AIトランスフォーメーションのための3つの「P」

## Process



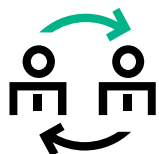
エンタープライズにおけるデータ分析では、やみくもに始めるのは悪手  
ビジネスゴール・ロードマップの策定、データ収集といったプロセスが重要

## Platform



収集した大量データを加工・分析、モデル学習する必要があるが、  
これら異なる処理基盤をサイロ化せず統合化・近代化すべき

## People



「コンテナ」の活用により、End-to-Endの分析基盤を柔軟・迅速に構築可能  
データサイエンティストにも優しい「データ分析の民主化」を実現しましょう！

# アンケートのご協力をお願い

アンケートと引き換えにノベルティをお渡しいたします。

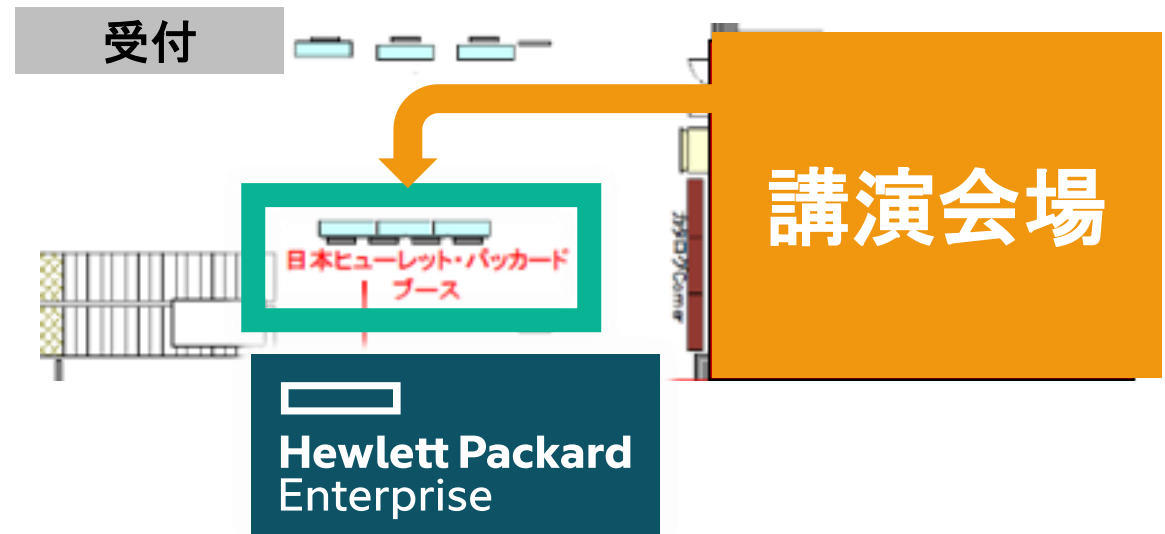
Q1. 本日の講演資料をご希望の場合は、チェックを入れてください。  
後日ダウンロードのURLをお知らせします。  希望する

Q2. 貴社のシステム導入を検討する立場	<input type="checkbox"/> 調査/検討中	<input type="checkbox"/> 導入済み			
Q3. 貴社の内容は、お客様の課題にどの程度対応していますか？	5	4	3	2	1
Q4. 貴社のシステム導入を検討している理由	[自由記述欄]				
Q5. AIの導入を検討している理由	[自由記述欄]				
Q6. AIシステムの導入を検討している理由	[自由記述欄]				
Q7. AIのシステム導入の時期についてお答えください	1年以内	1-3年以内	3-5年以内	5年以上	不明
Q8. 本日に参加した講演に対して、お返事がしたい(質問/回答)	[自由記述欄]				
Q9. ご質問したいメールアドレスに日本ヒューレット・パッカード株式会社からEメールをお送りさせていただきます。よろしいでしょうか？	<input type="checkbox"/> はい				

← 本日の講演資料  
をご希望の場合、  
資料希望の欄に  
チェックを入れて  
ください。

# HPE展示ブース案内

ぜひお立ち寄りをお待ちしております。





**Hewlett Packard**  
Enterprise

**Thank you**

