「見える化」の次を検討するための

「儲かる化」（最適化）条件シミュレータ SAS/Optimization

～　マテハン・IoTを先に進める　IoT++　～

SAS Institute Japan株式会社
インダストリーソリューショングループ　地主修一

2019/6/11

facebook.com/SASJapan　twitter.com/sasjapan　sascom.jp/blog

Copyright © SAS Institute Inc. All rights reserved.

AGENDA

- 「見える化」の次の「儲かる化」とは？

- What’s and Why SAS/Optimization ？

- ユースケース＆スケジューラとして使った例

Copyright © SAS Institute Inc. All rights reserved.

「儲かる化」とはなにか？(1)

儲かる化＝問題の見える化から解決へ

【問題の見える化】

- マテリアルハンドリングやIoTの取り組みで製造・物流の問題が見えた

↓

【問題の解決】

- 取り組みで見えた製造・物流の問題を解決して「儲け」につながってる

見えるだけじゃダメ・・・また儲けに関係ないことがいくら見えてもダメ

製造・物流の「儲かる」とは何か？　⇒**要は無駄取りと詰め込み**

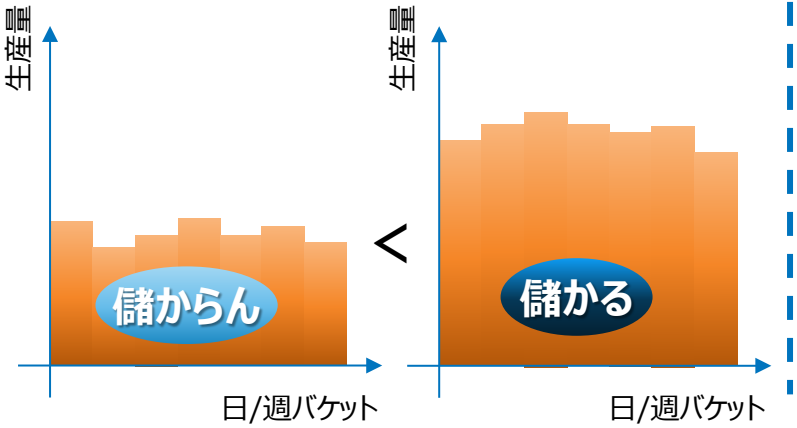
Copyright © SAS Institute Inc. All rights reserved.

「儲かる化」とはなにか？(2)

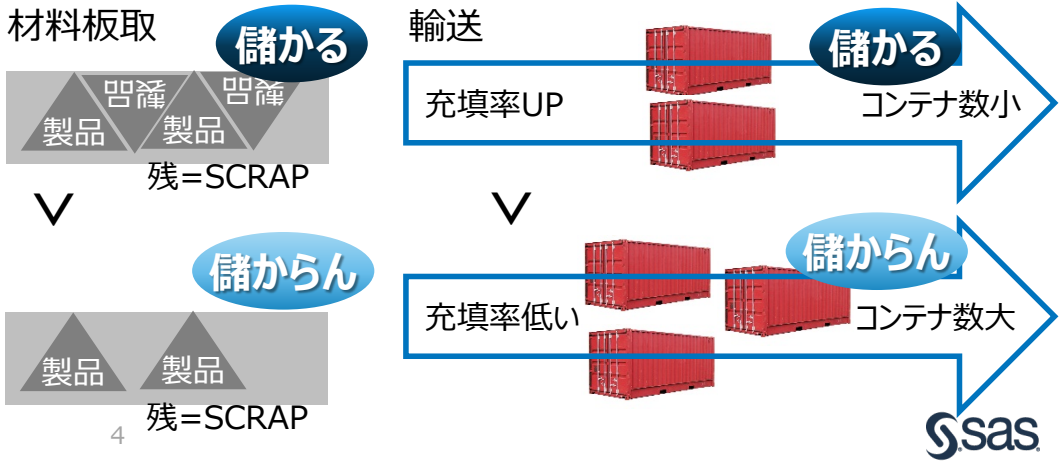
製造業・物流業で「儲かる」とは結局こういうことです

1．【投入した原料・労力に対して無駄のない生産】

2．【時間当たりより多く生産】



3．【空間当たりより多く製品をとる・詰込む】



Copyright © SAS Institute Inc. All rights reserved.

AGENDA

- 「見える化」の次の「儲かる化」とは？
- What’s and Why SAS/Optimization ？
- ユースケース＆スケジューラとして使った例

Copyright © SAS Institute Inc. All rights reserved.

儲かる化の具体例

SAS/Optimizationはこれらを支える計画手段を提供します。

1．【投入した原料・労力に対して無駄のない生産】・・・・「線形計画法(LP)」
- 連産品／副産物のある製品の製造量計画内訳
- 外部環境（仕入れ値／販価／ユーティリティ費）を考慮した最適在庫計画
- 複数拠点から供給できる製品の供給量計画

2．【時間当たりより多く生産】・・・・・・・・・・・・・・・・・・・・・「混合整数計画法(MILP)」
- 多数工程の機械工業や鍛造などで最適（歩留・生産性・機械稼働率）なスケジュールの策定
- スキルセットにばらつきがあるサービス作業員の組合せを考慮したシフトスケジュール立案

3．【空間当たりより多く製品をとる・詰込む】・・・・・・「混合整数計画法(MILP)」
- コンテナ/荷物箱への充填率考慮した梱包リスト作成（荷卸地順にコンテナ内の配置も検討可）
- 鉄鋼板からもっともスクラップの少ない板取の計算/製紙業での幅抱合せ抄紙計画立案

Copyright © SAS Institute Inc. All rights reserved.

SAS/Optimization®とは「全体が儲かる（最適）条件シミュレータ」
必要なモノ＝儲かる指標・制約条件・計算手段

課題解決の儲かる指標（全体最適化関数）を設定し、モノの流れや条件のモデルを組む

儲かる指標の例
- 生産性をあげたい⇒スループット
- 機械の遊びをなくしたい⇒稼働率

実際に運用できるようにするための現実的な制約条件を設定

制約条件の例
- 仕掛かってる機械に重ねてオーダを入れない。
- 一日〇〇時間×２シフトなど

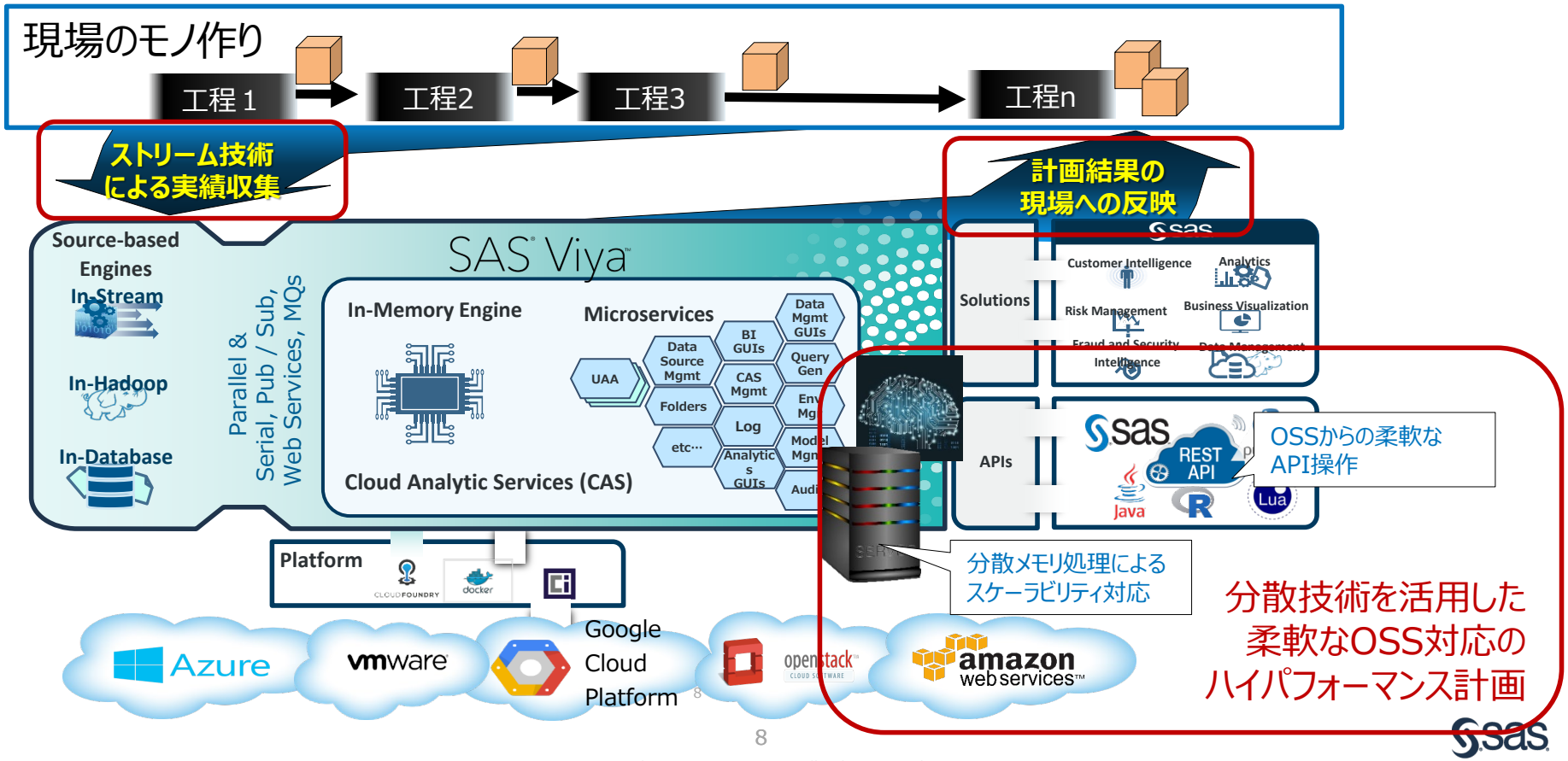
儲かる指標が「最適になる」条件(最大or最小)をコンピュータで計算して、現場の運用に取込

得られる運用の例
- 遊んでる機械を探して仕事を埋める
  ⇒効率的なジョブスケジュールができる

Copyright © SAS Institute Inc. All rights reserved.

SASは現場状況データを受けて「儲かる化」の解析結果を計画に反映します

現場連携実績収集＆ハイパフォーマンスな計画反映のECOSYSTEMが完備



Copyright © SAS Institute Inc. All rights reserved.

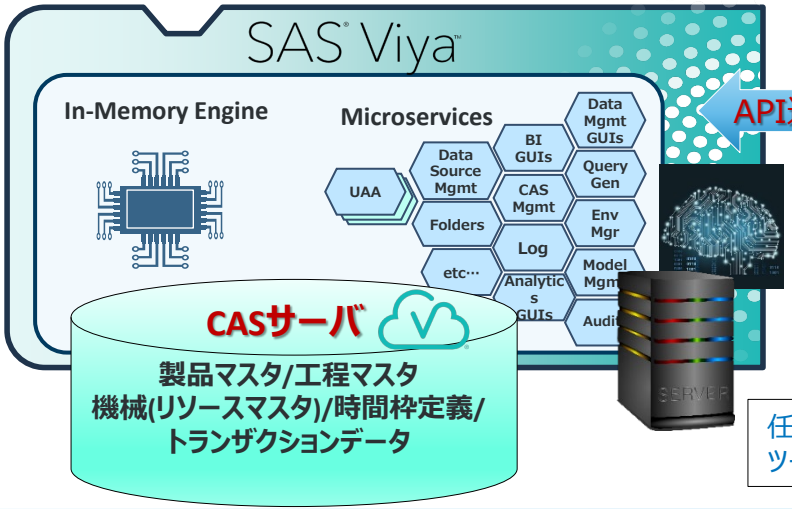
AGENDA

- 「見える化」の次の「儲かる化」とは？
- What’s and Why SAS/OR ？
- ユースケース＆スケジューラとして使った例

Copyright © SAS Institute Inc. All rights reserved.

今回のDemo環境についてのご説明

データはCASサーバー／ロジックはローカル／結果はOSS（Pythonのライブラリ）



API送受信

Ex.Pythonから利用すれば
Pandas.DataFrame
など柔軟なPython
Objectを利用可能

Program類はすべてローカル
(microservice call)
SAS9/Optimization
&SASOPTPY
&(OSS)plotly

SASOptは「儲かる指標」や「制約・変数」をセットしてSOLVEとするだけで容易に分散技術を活用した解析が利用可能
（SASが作成した豊富なサンプルも提供されている）

任意の計画可視化（GANTT等）ツールと連携可

Resource Links（Githubs）　すべてPIPからインストール可能

【SAS】　swat :　https://github.com/sassoftware/python-swat

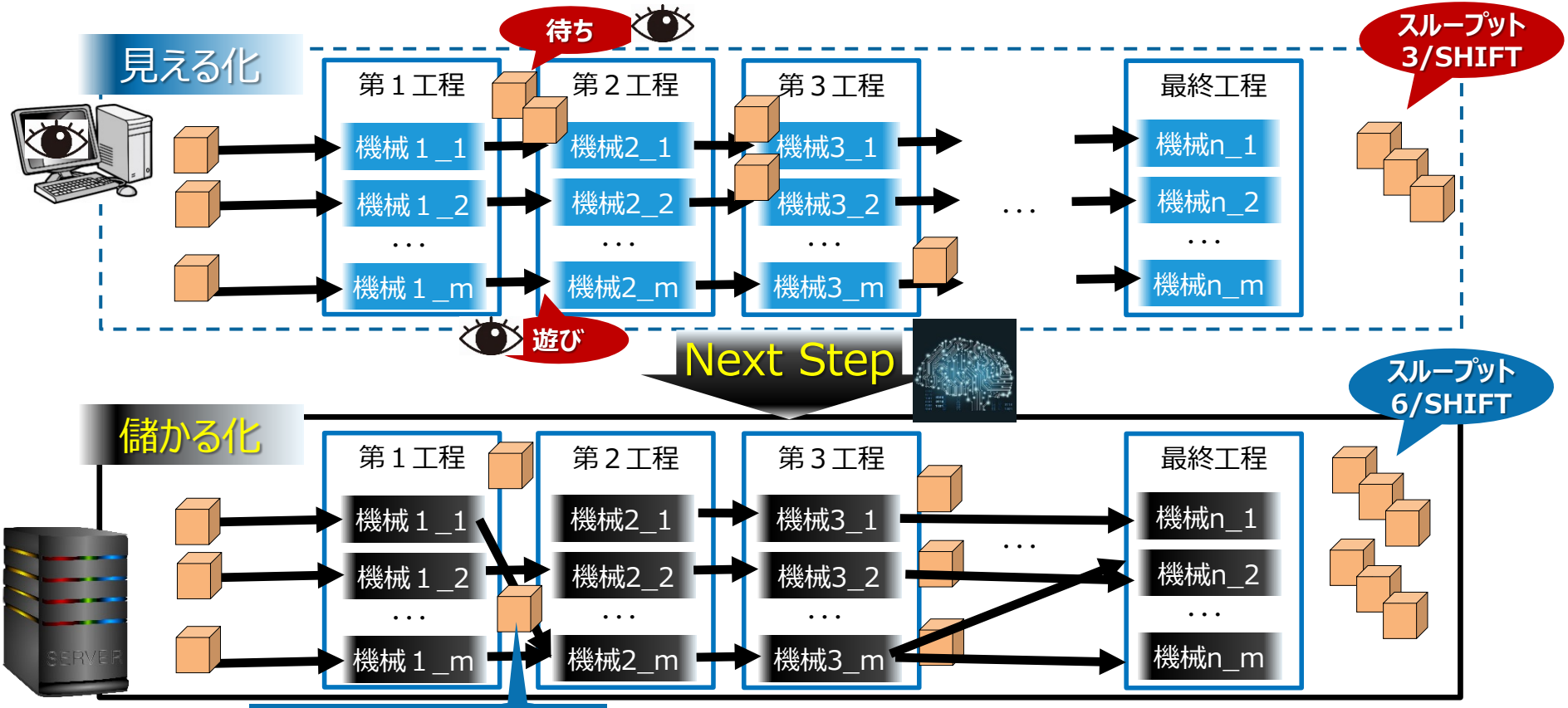
sasoptpy :　https://github.com/sassoftware/sasoptpy

サンプル集⇒ https://github.com/sassoftware/sasoptpy/tree/master/examples

【OSS】　Plotly :　https://github.com/plotly/plotly.py

Copyright © SAS Institute Inc. All rights reserved.

IoT取組みでサプライチェーンや生産プロセス「見える化」はできた⇒次は「儲かる化」



Copyright © SAS Institute Inc. All rights reserved.

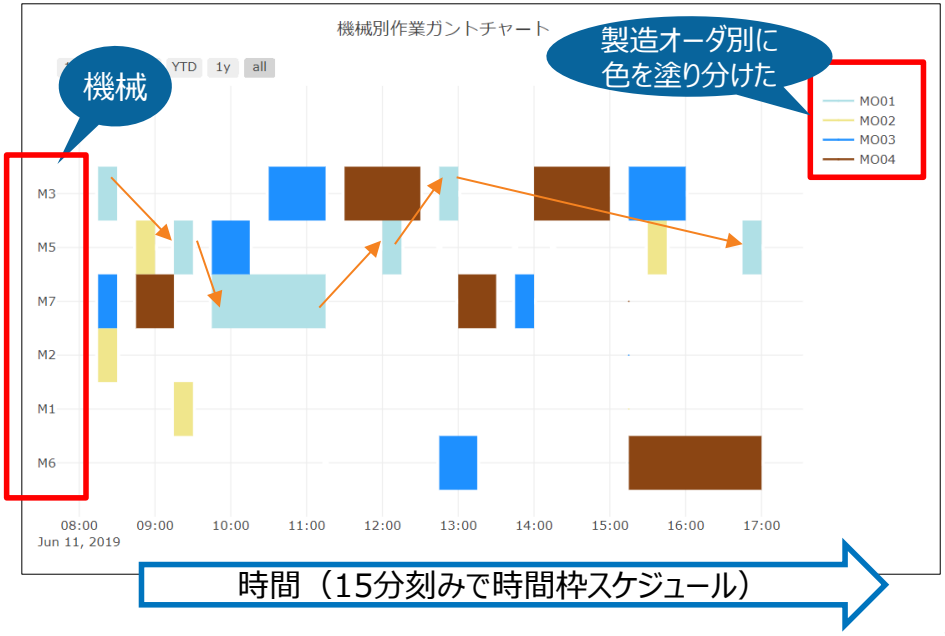
SAS/Optimizationによるスケジューリング最適化例

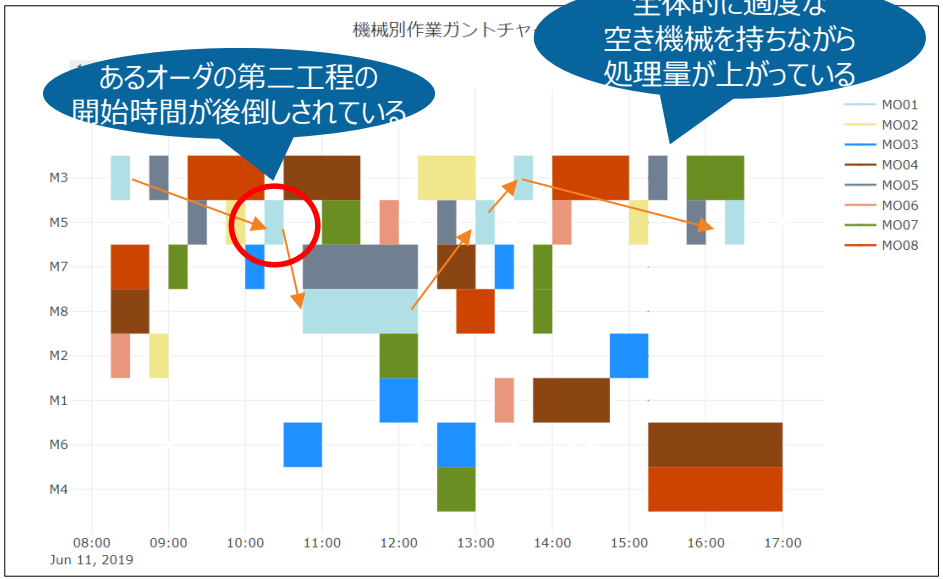
SAS/Optimaiationの出力をplotlyライブラリ(FREEWARE)で可視化(赤矢印は説明のために上書き)

4-6工程を持つ４つの製造オーダMOにさらに5-8工程を持つ４つのMOを追加しリスケジュール

必要に応じて後工程の処理タイミングを再計画し、全体のスループット向上

先入先出（FIFO）のスケジューラでは待ちと遊びができやすい





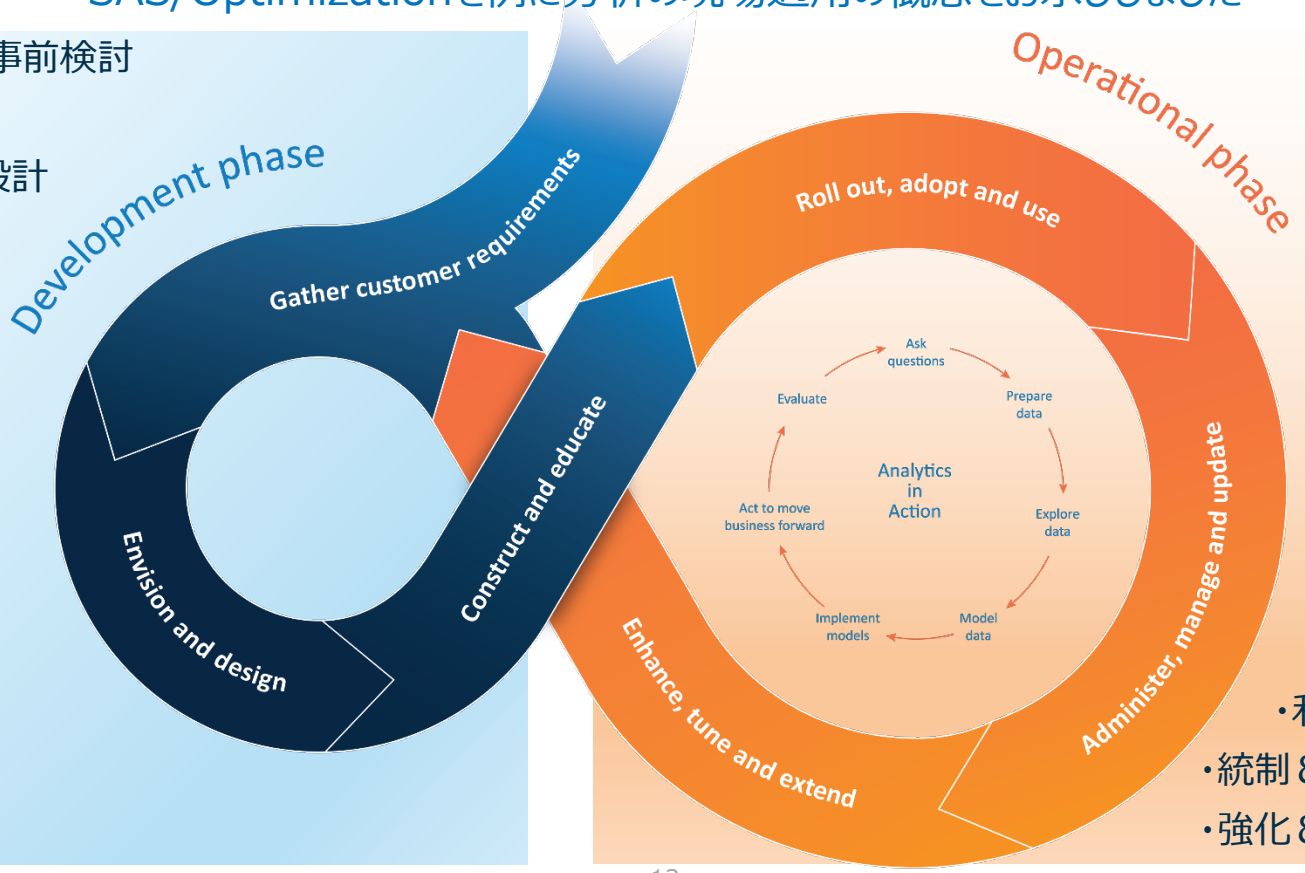
Copyright © SAS Institute Inc. All rights reserved.

まとめ　SASによる分析と現場への活用（Analytics in Action）

SAS/Optimizationを例に分析の現場適用の概念をお示ししました

「儲かる化」への事前検討

- 要求理解
- ビジョン構築と設計
- 実装＆教育



現場適用

- 利活用＆展開
- 統制＆管理＆更新
- 強化＆調整＆拡張

Copyright © SAS Institute Inc. All rights reserved.