

目指せコーディングレス！ 「繋げる」が実現するクラウド活用による高速アプリケーション開発の魅力

講演サブタイトル

日本オラクル株式会社
クラウド事業戦略統括
クラウドプラットフォームソリューション部
山成 慎吾

Modern Cloud Day Tokyo

次世代クラウドが変える日本のビジネス



以下の事項は、弊社の一般的な製品の方向性に関する概要を説明するものです。また、情報提供を唯一の目的とするものであり、いかなる契約にも組み込むことはできません。以下の事項は、マテリアルやコード、機能を提供することをコミットメント（確約）するものではないため、購買決定を行う際の判断材料になさらないで下さい。オラクル製品に関して記載されている機能の開発、リリースおよび時期については、弊社の裁量により決定されます。

OracleとJavaは、Oracle Corporation 及びその子会社、関連会社の米国及びその他の国における登録商標です。文中の社名、商品名等は各社の商標または登録商標である場合があります。

Agenda

- 1 ITトレンドとしての高速リリースへのモチベーション
- 2 ニーズを具現化するための現実解
- 3 コードレス開発を実現するOracle Integration Cloud
- 4 Oracle Integration Cloudによる高速開発

Agenda

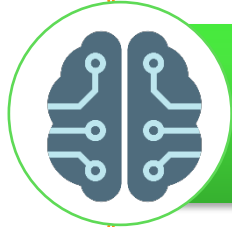
- 1 ITトレンドとしての高速リリースへのモチベーション
- 2 ニーズを具現化するための現実解
- 3 コードレス開発を実現するOracle Integration Cloud
- 4 Oracle Integration Cloudによる高速開発

ITトレンドのビジネス活用への模索

ビジネスに対するITトレンドの活用の模索と見極め



IoT/BigData : "モノ" から流れるデータの分析や利活用



AI : 機械学習によるインテリジェンスやRPAによるオートメーション



Blockchain : 分散台帳をベースとしたコンソーシアム型の提携ビジネス

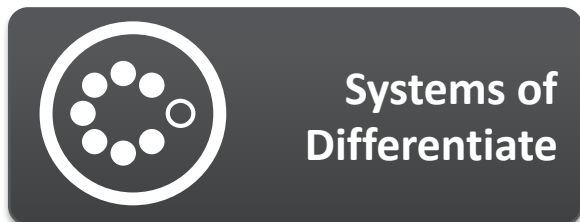
“ペースレイヤーモデル” から見たサービスの更新頻度

早期に取り組み・判断するフットワークが求められる領域



新しいビジネスの発見

- 新たなビジネス要件や機会の変化に対応し**即座に開始し頻繁に更新する必要**のあるサービス
- **ライフサイクルは数ヶ月～1年未満と短い**



既存ビジネスの遂行

- 企業や業界固有プロセスに対応するサービス
- **新規ビジネスと連動**する可能性
- **ライフサイクルは1～3年程度だが、頻繁に変更・強化**して変化に対応する必要がある



バックオフィス業務

- 業務の根幹となる中核的なトランザクション処理を担い企業の重要なマスターデータを管理
- 変更のペースは遅い（基盤の刷新など）

更新頻度

高



更新頻度

低

“Pace-Layered Application Strategy and IT Organizational Design”

<https://www.gartner.com/binaries/content/assets/events/keywords/applications/apn30/pace-layered-applications-research-report.pdf>

注目を浴びるDevOpsと主な期待

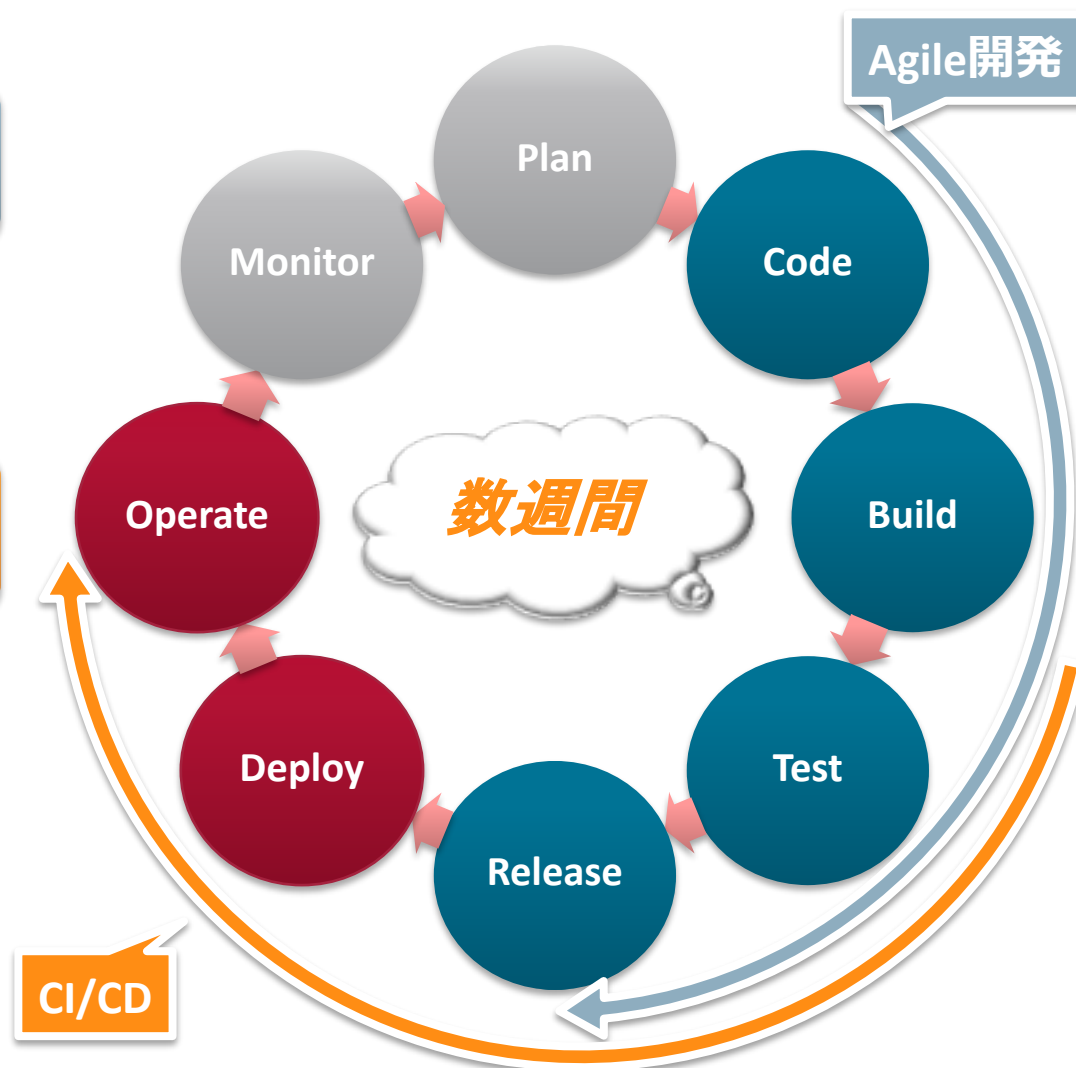
サービス開発の生産性(即応性/柔軟性)に対する一般的な課題意識

Agile開発

- スクラム開発を中心とした開発サイクルの小規模化
- 開発チームによる意思決定

CI/CDによる自動化

- ビルド/リリース/運用タスクの自動化
- 環境構築コストの極小化と確実性の確保
- アセット・ドリブンのリリースサイクルの自動化



注目を浴びるDevOpsと主な期待

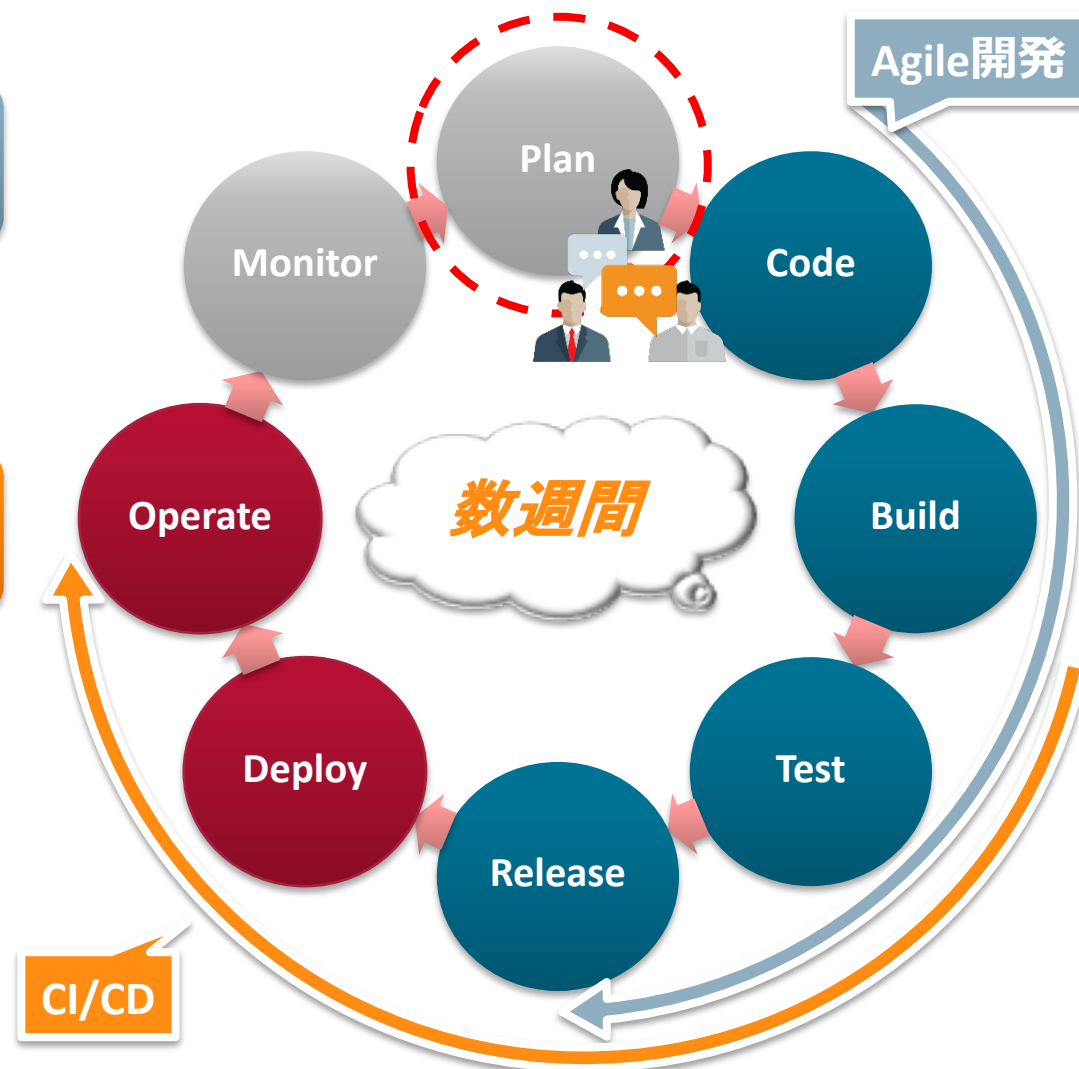
最も重要な計画・設計への言及は比較的少ない。。。

Agile開発

- スクラム開発を中心とした開発サイクルの小規模化
- 開発チームによる意思決定

CI/CDによる自動化

- ビルド/リリース/運用タスクの自動化
- 環境構築コストの極小化と確実性の確保
- アセット・ドリブンのリリースサイクルの自動化



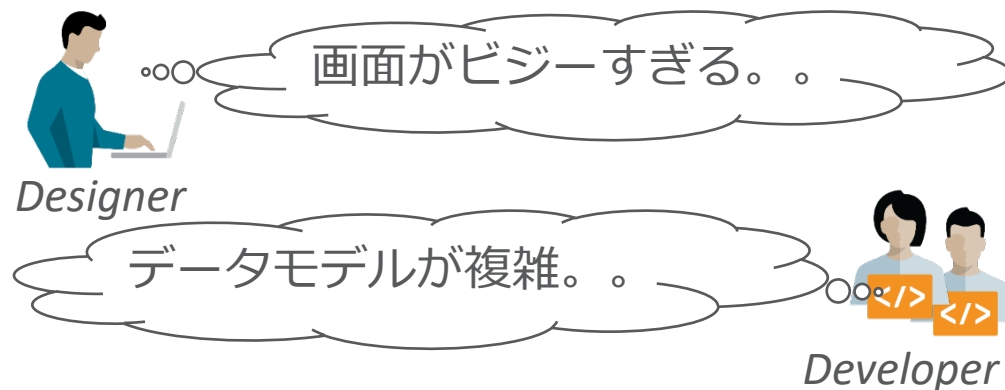
Agenda

- 1 ITトレンドとしての高速リリースへのモチベーション
- 2 ニーズを具現化するための現実解
- 3 コードレス開発を実現するOracle Integration Cloud
- 4 Oracle Integration Cloudによる高速開発

ニーズを具現化するための重要なプロセス プロトタイピングやトライアルによるサービス設計の精緻化

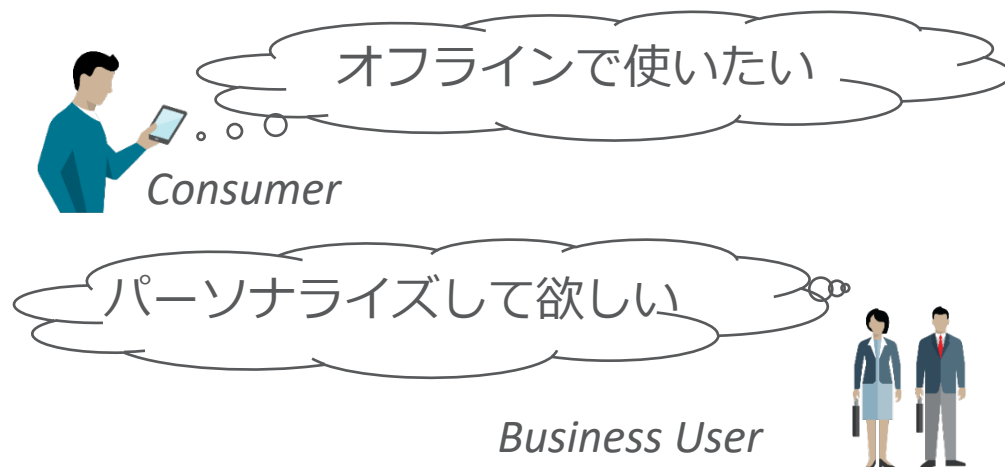
開発チーム内でのフィードバック

- プロトタイプ開発による**実際に動くモノによる評価**
- **体感によって得られるフィードバック**の重要性



実ユーザからのフィードバック

- 特定(セグメント、地域、等)ユーザへの限定的開放と**フィードバックの収集**
- 提供される**サービス自体に対するフィードバック**



模索しながら開発するサービスの重要タスク

想定ニーズと実ニーズのギャップにいかに迅速に対応するか

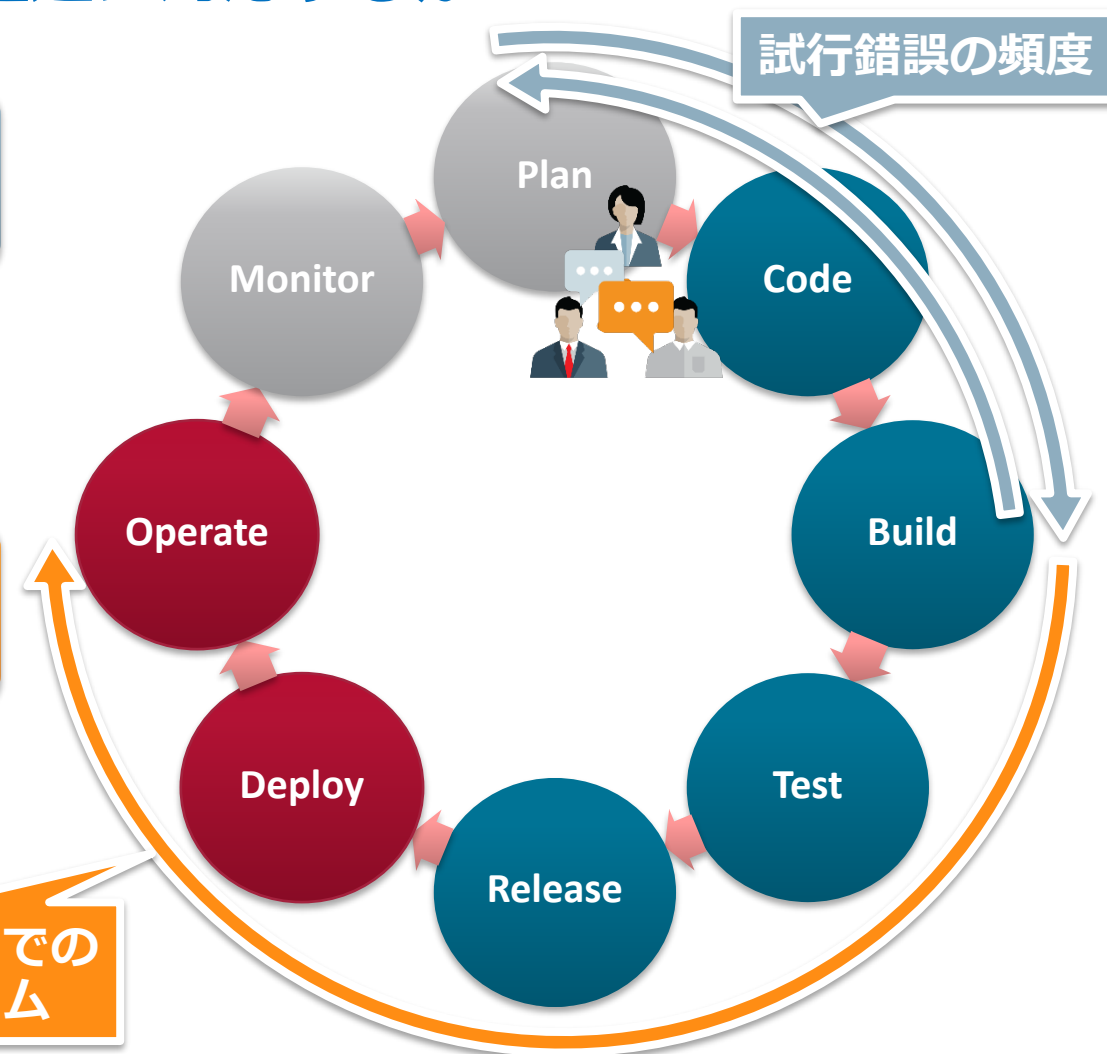
試行錯誤：想定ニーズの具現化

- プロトタイピングを踏まえた計画と設計
- いかに**頻繁かつ容易に試行錯誤**ができるか

トライアル：実ニーズの把握

- 実ユーザーに対する限定公開による利用
- いかに**素早く本番移行**が可能か

トライアルまでの
リードタイム



プロトタイピング/トライアルで注目すべき視点

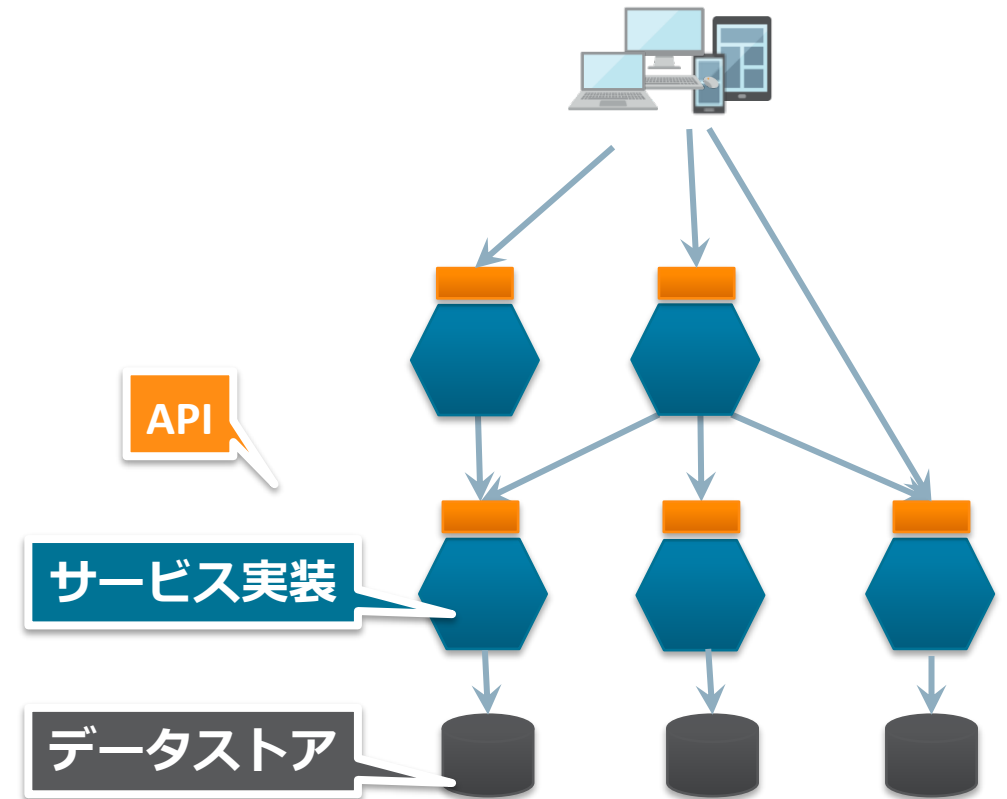
「既存サービス」を使い「重要なサービス」にフォーカス

プロトタイピングにおける要点	開発時のプロトタイプ	ユーザによるトライアル
仮説を立てたサービスの設計	○ プロトタイプの試行による設計の精緻化	○ サービス内容に対するフィードバックを正式版に反映
重要な機能のみを実装	○ プロトタイプの試行による重要な機能の確認	○ 機能改善/追加に対するフィードバックを正式版に反映
ユーザ・エクスペリエンスにフォーカス	○ 可視化/体感を踏まえたサービスの機能や質の確認	○ 操作性・デザインに対するフィードバックを正式版に反映
既存のサービス/ツールを活用	○ カスタム開発/構築を極小化して試行頻度と速度を優先	○ カスタム開発・構築を極小化して早期のトライアルを優先
モックアップ/静的データによる実装	○ 機能/エクスペリエンスの妥当性を優先的に評価	✗ 実利用可能な機能とデータセットによる提供が必須
品質よりも実装を優先	○ 機能/エクスペリエンスの妥当性を優先的に評価	✗ 実利用可能な最低限の品質の(エラー処理、可用性等)確保が必須

柔軟性を維持するためのアーキテクチャ #1

APIで提供される小規模な機能群を組み合わせ疎結合なシステムを実現

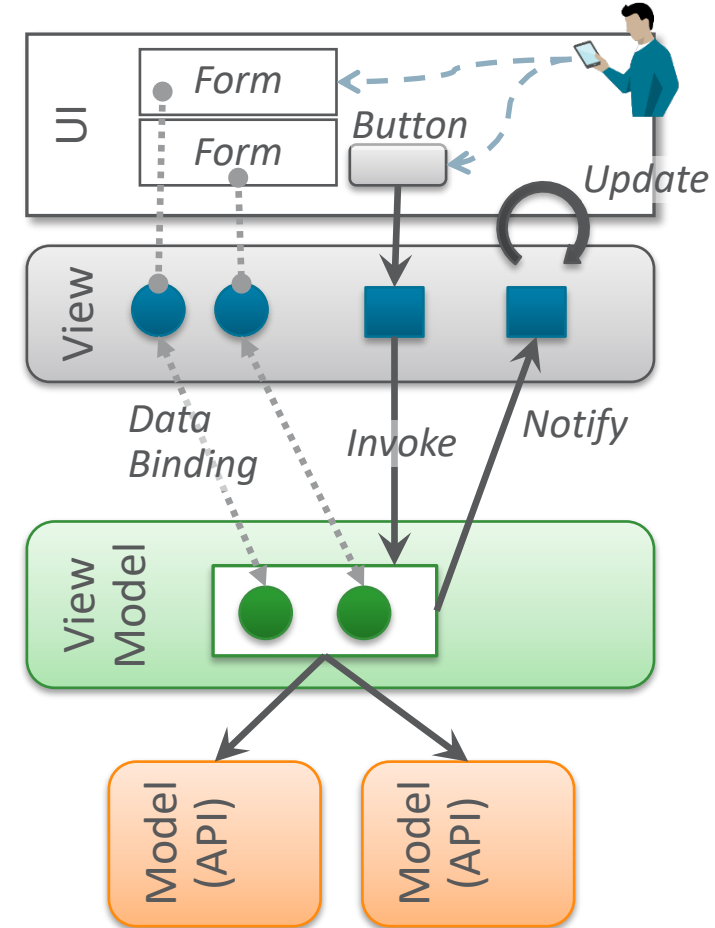
- マイクロサービス・アーキテクチャ
 - SOAから派生した**疎結合**を前提としたシステム・アーキテクチャ
 - **API等でサービス間を分離**することによりサービス間の依存度を極小化
 - サービス単位でデータ/インフラストラクチャを独立させ**サービス間の密結合を防止**
 - データソース間の複雑なトランザクションの排除
 - データソース共有によるサービス間依存性の排除



柔軟性を維持するためのアーキテクチャ #2

ユーザ・インタフェース(UI)側でAPIをマッシュアップして各サービスを制御

- Model-View-ViewModel(MVVM)
 - MVC(Model-View-Controller)から派生した**UIと業務ロジックを分離**したプレゼンテーション・レイヤの設計モデル
 - UIの操作と機能の呼出しの**双方向連携をデータ・バインディング**により実現
 - Viewが管理するUI上のデータ(フォームデータ等)とViewModelが管理するデータの同期
 - UI上の操作に対するModelに対する操作の割当て
 - SPA(Single Page Application)のようなUIコンポーネント単位での機能呼び出しの実装と相性が良い



各レイヤで実装すべきアプリケーションの特徴

UIコンポーネントのデザイン/APIエンドポイント実装/APIマッシュアップ

画面コンポーネントのレイアウト

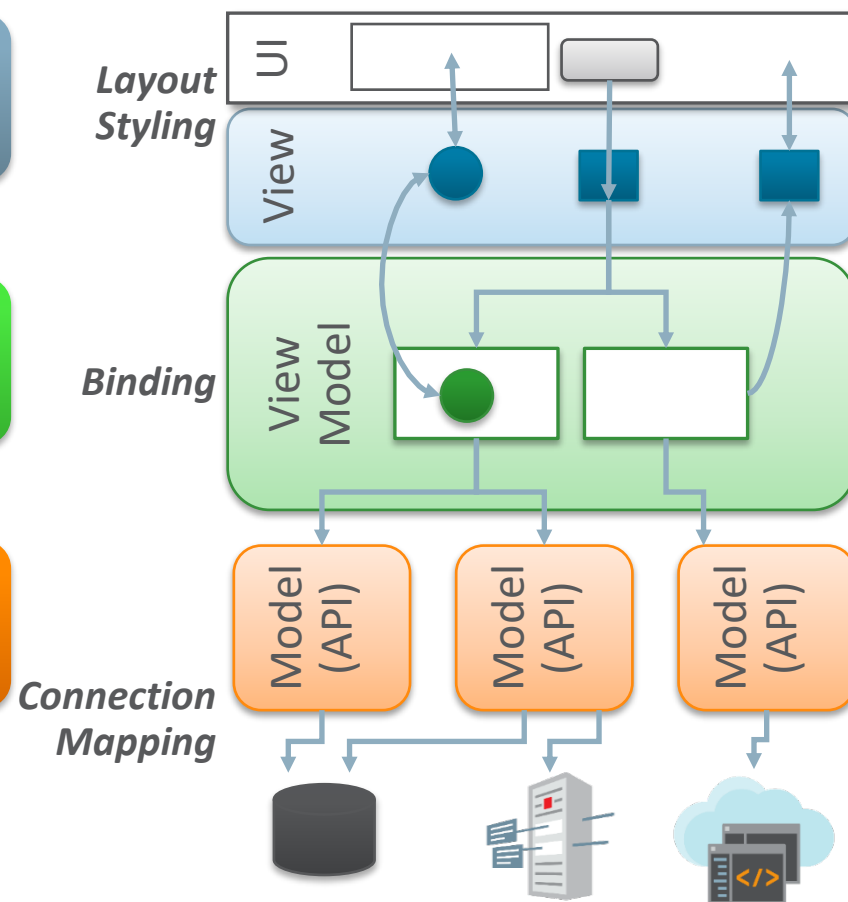
- ユーザインタフェースの配置とスタイルのデザイン

画面コンポーネントとAPIのバインディング

- UIコンポーネントとAPIの間の双方向バインディング

データソースや他システムのAPI化

- 既存システムやデータベース等の接続制御・管理
- 呼出し順序や結果のマッピング等の逐次的処理



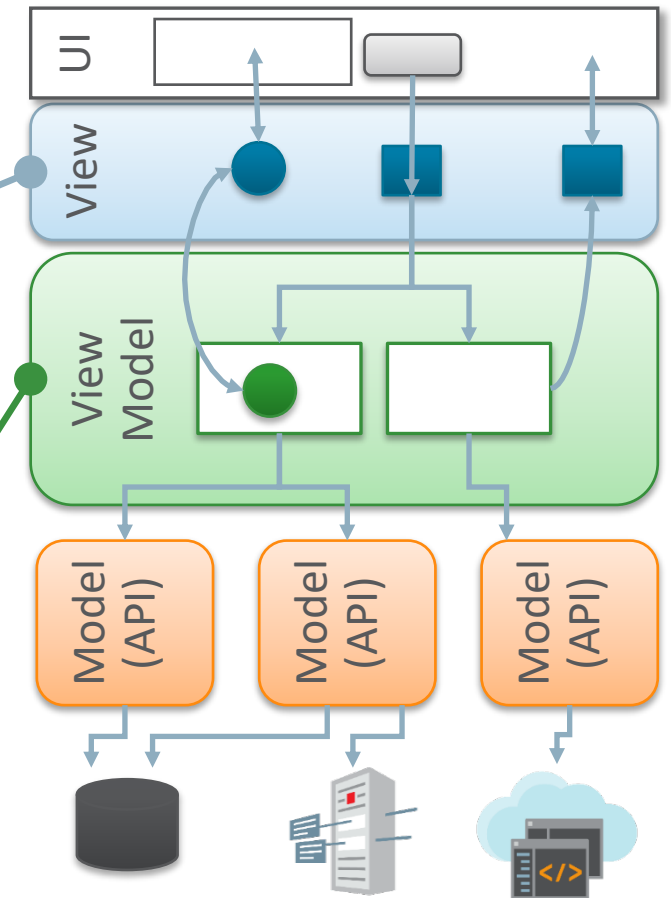

コードレス開発のメリット #1

成果物を想像し辛いプログラミングによる実装の現実

```
1 <oj-ext-mobile-page-template id="page-template" page-title="お客様のプロフィール">
2
3 <div class="oj-flex-item oj-flex oj-sm-flex-direction-column oj-sm-flex-wrap-nowrap">
4 <div class="oj-panel oj-flex-items-pad oj-flex-item" id="div-516204537-1">
5 <div class="oj-flex">
6 <oj-label id="oj-label-516204537-1" for="oj-input-text-516204537-1" class="
7 <oj-input-text id="oj-input-text-516204537-1" class="oj-flex-item" require
8 </div>
9 <div class="oj-flex">
10 <oj-label id="oj-label-516204537-2" for="oj-input-text-516204537-2" class="
11 <oj-input-text id="oj-input-text-516204537-2" class="oj-flex-item" require
12 </div>
13 <div class="oj-flex">
14 <oj-label id="oj-label-516204537-3" for="oj-select-one-516204537-1" class="
15 <oj-select-one id="oj-select-one-516204537-1" class="oj-flex-item" value="
16 <oj-option value="M">男</oj-option>
17 <oj-option value="F">女</oj-option>
18 <oj-option value="O">その他</oj-option>
19 </div>
20 </div>
```

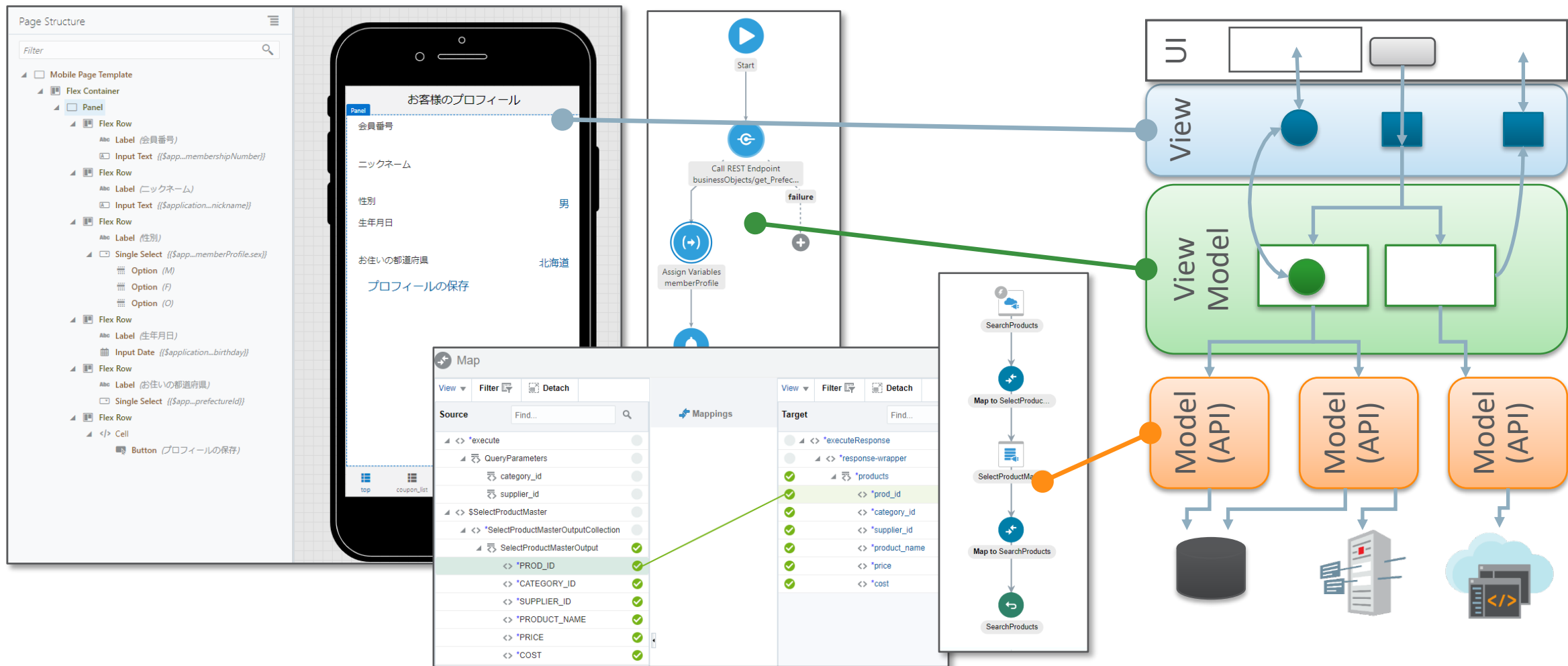
???

```
31 "chains": {
32 "ButtonActionChain": {
33 "description": "",
34 "variables": {
35 "detail": {
36 "required": true,
37 "type": "any",
38 "input": "fromCaller"
39 }
40 },
41 "detail": {
42 "required": true,
43 "type": "any",
44 "input": "fromCaller"
45 }
46 }
47 }
```



コードレス開発のメリット #2

アプリケーション構造の可視化と定型処理のプログラミングによる誤実装の防止



模索しながら開発するサービスの重要タスク

想定ニーズと実ニーズのギャップにいかに迅速に対応するか

プロトタイピングによる想定ニーズの具現化

- 頻繁かつ容易に試行錯誤によるサービス設計・実装

トライアルによる実ニーズの把握

- 素早い本番移行により早期フィードバックの実現

コードレス開発によるサービス開発の加速

- 確実なイメージを掴みながらアプリケーション設計
- 生産性の低い技術的/単純作業の省力化

素早く本番移行

頻繁な試行錯誤



Agenda

- 1 ITトレンドとしての高速リリースへのモチベーション
- 2 ニーズを具現化するための現実解
- 3 コードレス開発を実現するOracle Integration Cloud
- 4 Oracle Integration Cloudによる高速開発

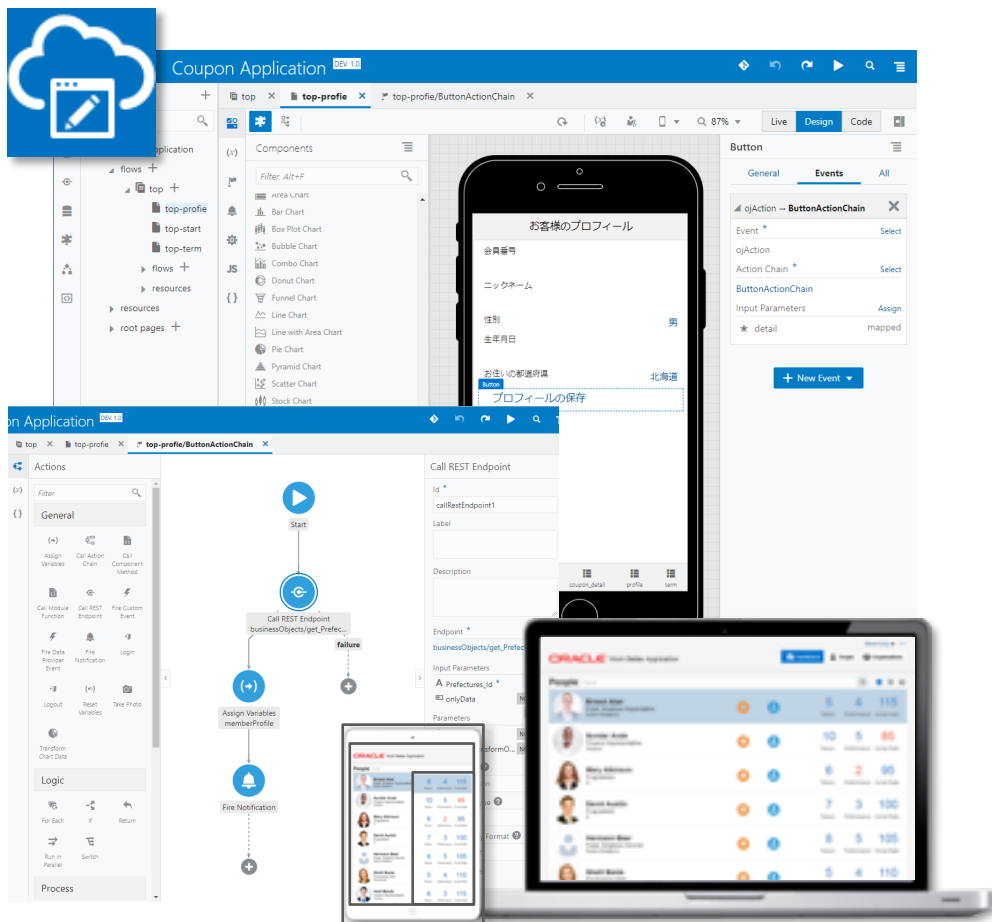
Oracle Integration Cloud

UI/システム/業務プロセスをデザインタイムで即座にマッシュアップ



Oracle Integration Cloud – Visual Builder

Web/モバイルUIの直感的なデザインタイム開発を実現！



誰にでもできる簡単Web UI開発

- Web UIで利用可能な直観的 & リッチなデザイン
タイムUI開発

マルチデバイスに標準で対応

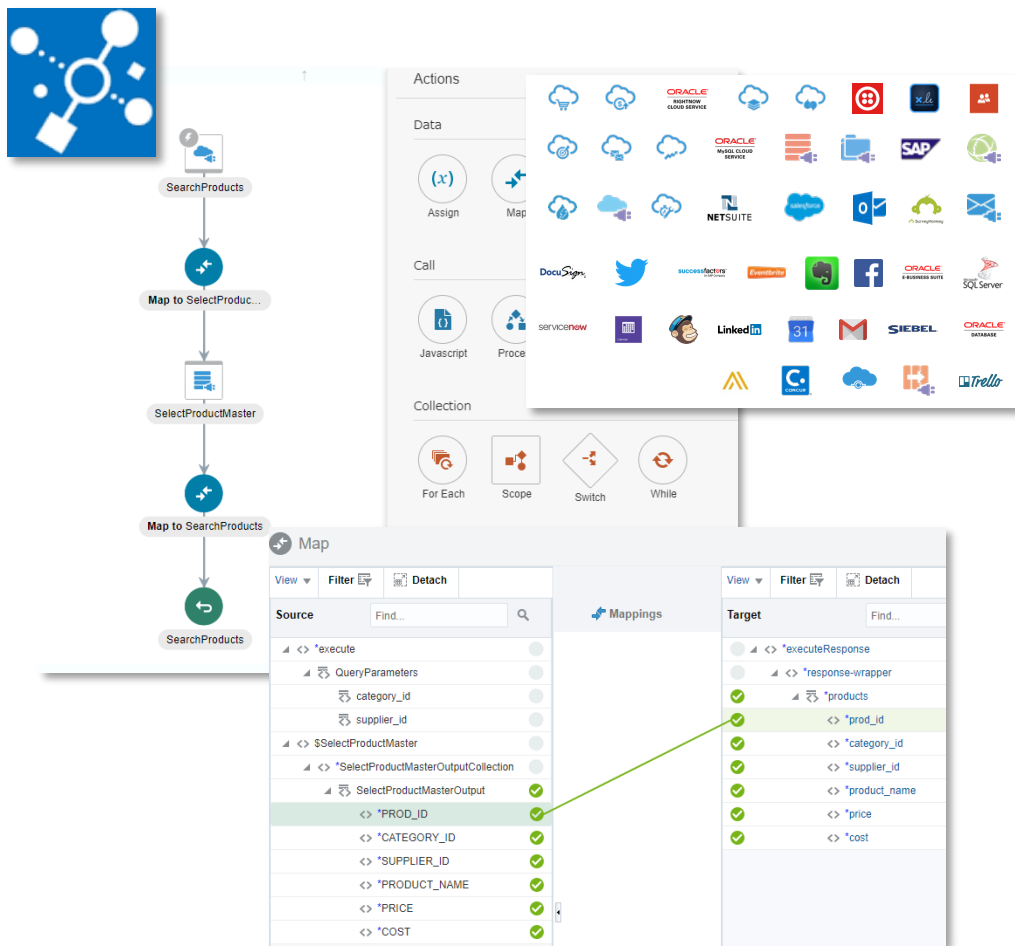
- 豊富なUIコンポーネントを組み合わせ素早く簡
単にレスポンスUI開発

多様なデータ/システムと連携

- Integrationで作成した連携から外部APIまで様々な
システムやデータソースとマッシュアップ

Oracle Integration Cloud - Integration

接続アダプタをコードレスでマッシュアップしてシステム間連携を素早く実現！



直感的なデザイナー

- シンプルな接続から複雑なオーケストレーションまでドラッグ&ドロップでシステム連携

簡潔なデータ構造の変換

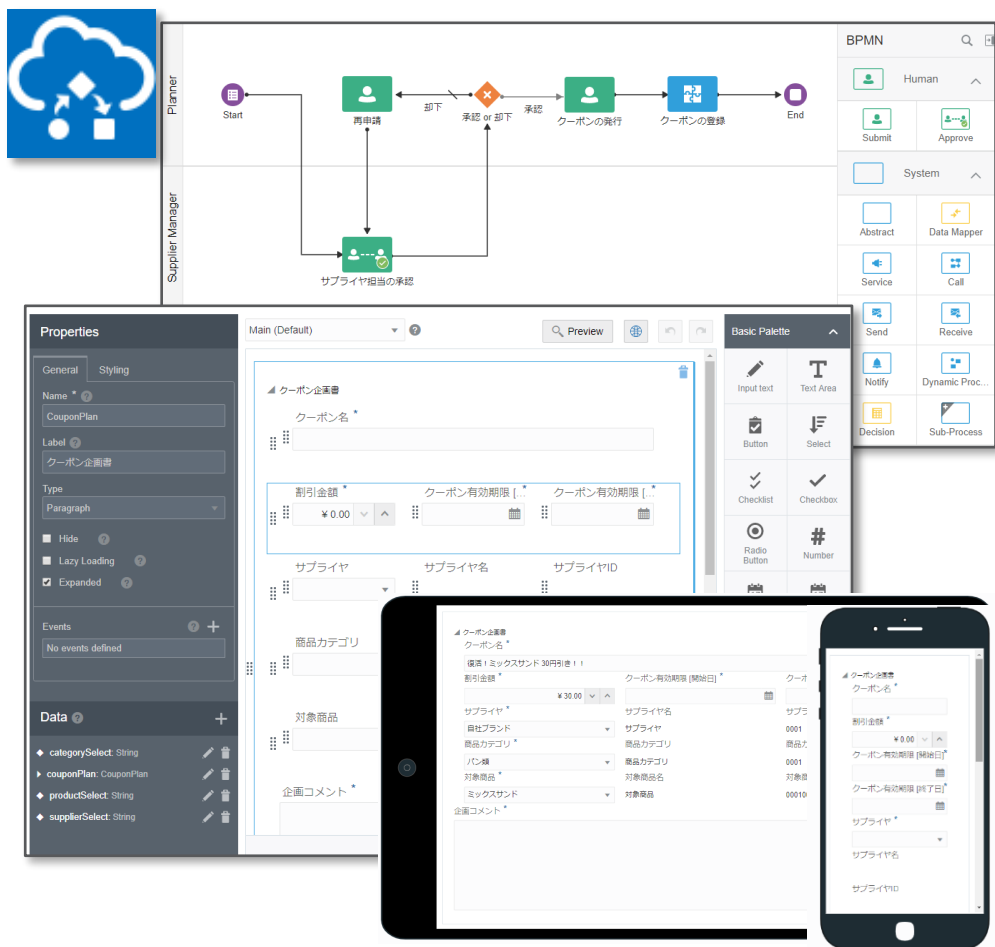
- RESTやデータベース、SNSからSaaSまで多岐に渡るシステムとクイックに接続

連携の制御も簡単

- セットアップの容易な接続エージェントでオンプレミス環境とセキュアに接続

Oracle Integration Cloud - Process

業務プロセスの設計・実装をデザインタイムで手早く実現！



即座に利用可能なワークフロー

- デザインタイムでワークフローを設計するだけで即座に業務プロセスを実行可能

デザインタイムのフォーム開発

- 業務プロセスで利用するフォームをアクティビティと連携させながらデザインタイムで開発

即座に他システムと連携

- Integrationと統合済みで社内外の様々なシステムとワークフローの連携を容易に実現

Agenda

- 1 ITトレンドとしての高速リリースへのモチベーション
- 2 ニーズを具現化するための現実解
- 3 コードレス開発を実現するOracle Integration Cloud
- 4 Oracle Integration Cloudによる高速開発**

モデルケース：クーポンによる顧客プロフィール収集

- 想定業種と背景
 - 全国展開する特定セグメント向けのスーパーマーケット
 - **顧客プロフィールを利用した購買傾向から商品/仕入れ計画に生かせるかどうか**の検証する
- 目的
 - **分析に足りる顧客プロフィールを集める**事にフォーカス
 - 一定期間収集した上で状況に応じて分析の手段や方式を計画する
- 手段
 - **割引クーポン**を材料に顧客プロフィールを収集
 - 既に配布済みの会員カード（個人情報不要）に紐づけて顧客プロフィールを収集
 - 主婦層にも浸透してきたスマートフォン向けのペーパーレス・クーポン
 - これまで紙ベースのQRコード・クーポンを運用した実績あり

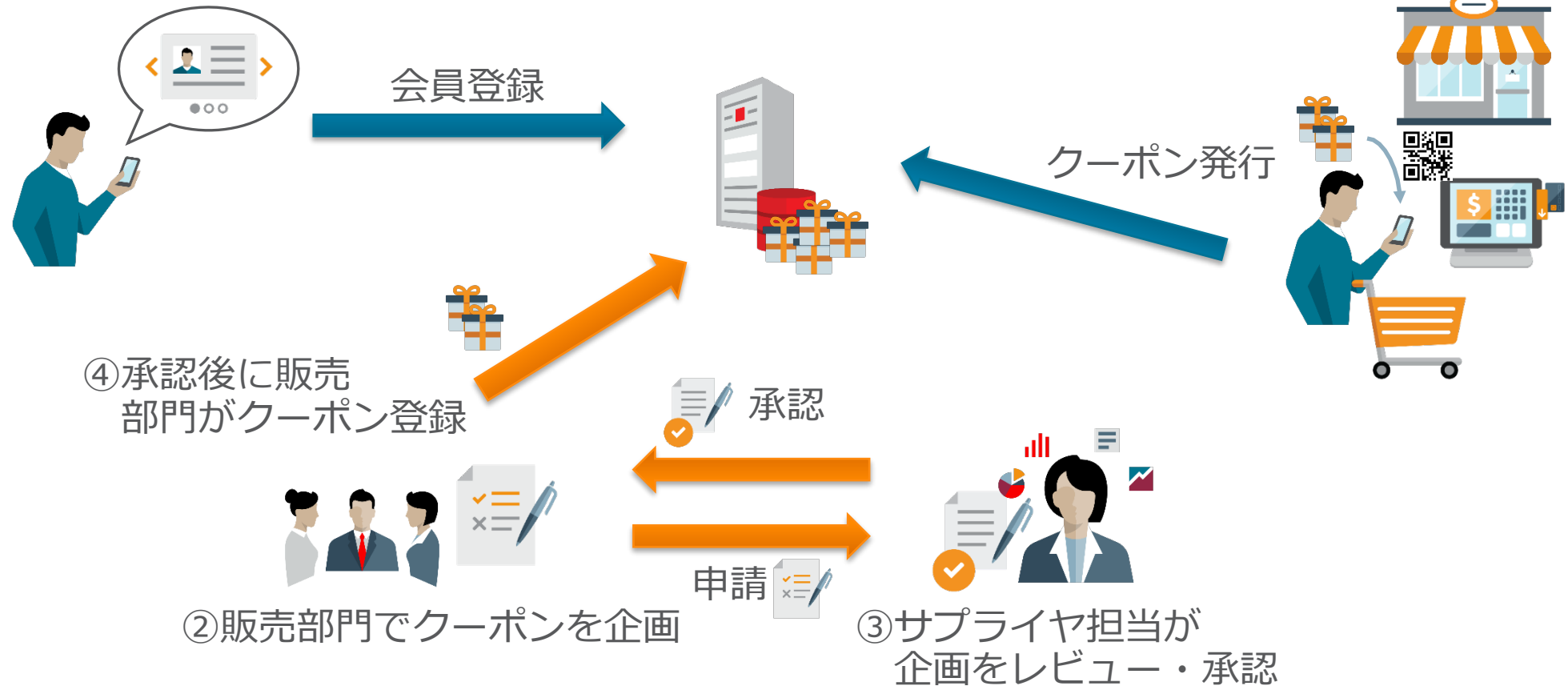
モデルケース：想定する諸要件

- ユーザの利用のモチベーションを落とさないために、**個人が特定できる属性情報は扱わない**
 - 会員カード番号、生年月日、性別、居住する都道府県
- モバイル・アプリケーションは極力シンプルに
 - **会計時にすぐに出せる**ように
 - **会員番号と顧客プロフィールは初回時のみ**入力すれば使えるように
- オンプレミス環境の変更は必要最小限に
 - **変更が不要**なのがベスト
- クーポンの発行にはサプライヤ部門の商品担当の許可が必要
 - クーポンの発行は**販売部門で企画**
 - **サプライヤ部門での承認**を経た上で、**販売部門が発行**(マスタDBに反映)する

モデルケース：システムの利用イメージ

① アプリをダウンロード
会員番号とプロフィールを登録

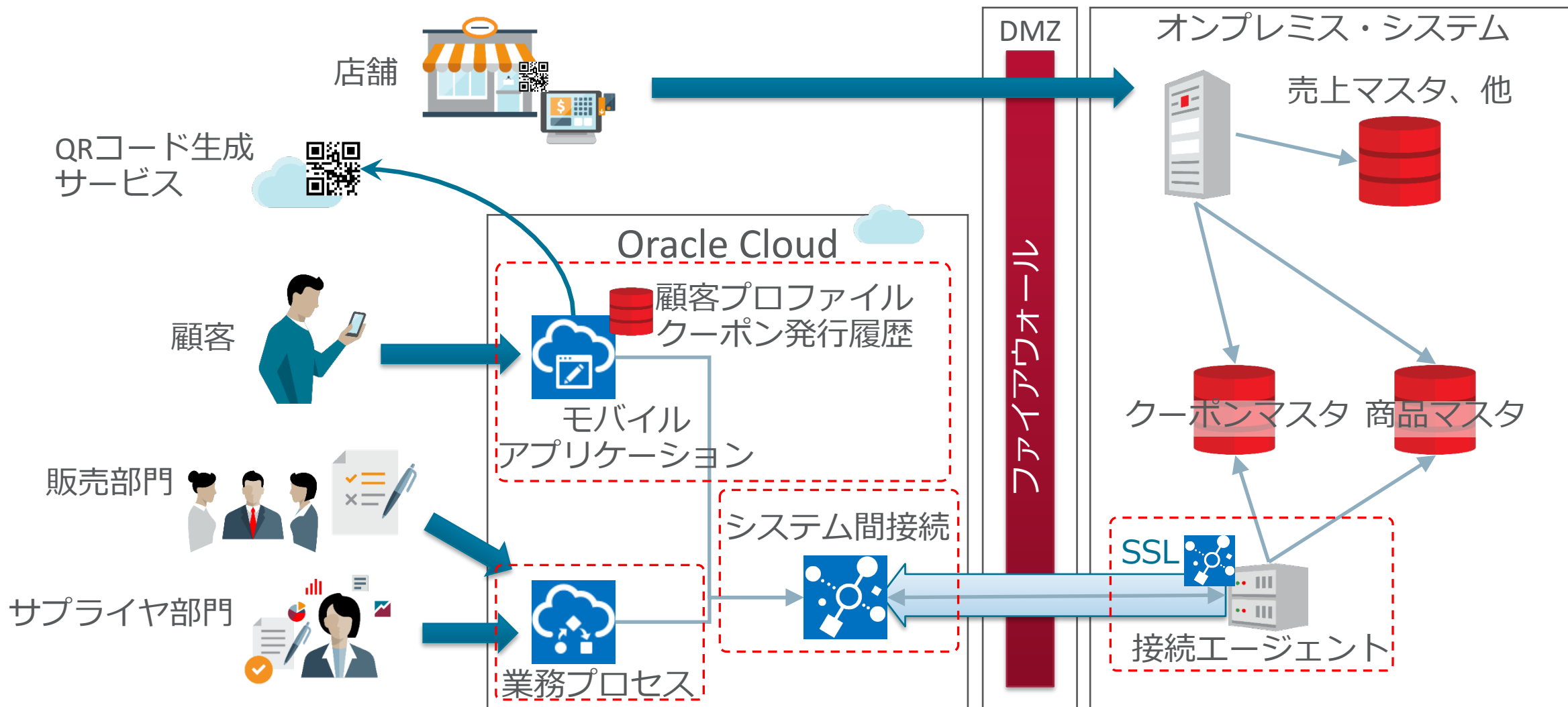
⑤ 会計時にクーポンのQRコード
を選び会計金額から割引



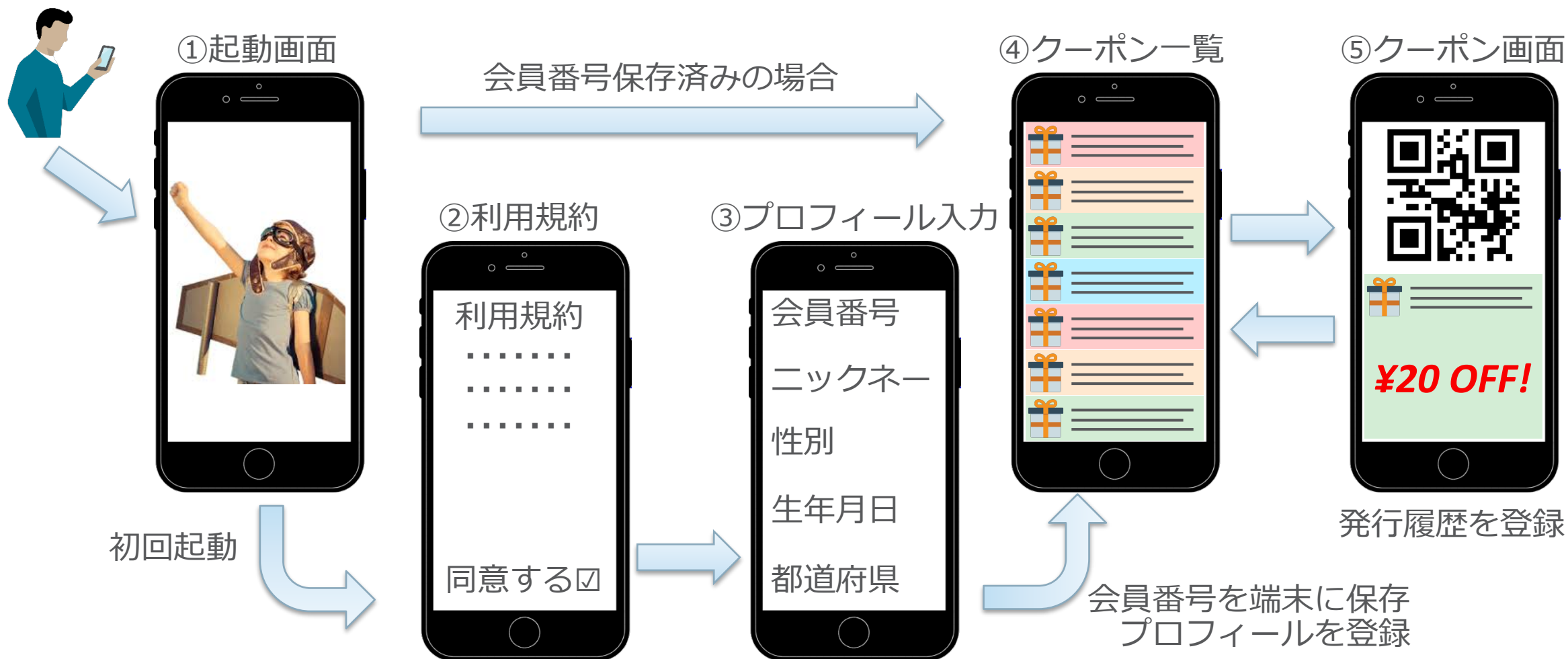
モデルケース：既存システムとの関連性

- 店舗系システム
 - 紙ベースのQRコードのクーポン(割引商品扱い)を過去取り扱っており、**販売系システム側の修正は不要**
- オンプレミス上のシステムとデータ
 - 商品マスタ、クーポンマスタ等の**マスタ情報はオンプレミス側の既存データベース**を利用
 - マスタ情報参照のためのIntegration Cloudの**接続エージェントの環境のみ追加**で配置
- 新規で作成するシステム
 - Visual Builder：QRコードのクーポンを発券して利用する**モバイル・アプリケーション**
 - Integration：オンプレミス上の**マスタ参照のためのシステム間接続**
 - Process：クーポンの発行プロセスにかかる**業務プロセスのアプリケーション**
 - QRコードの生成は商用利用可能な**外部サービスを活用し開発しない**
- 収集した顧客プロファイルやクーポン発行履歴のデータ
 - 分析時に取り出せれば良いため**Visual Builderが内包するデータベースに格納**し適宜エクスポート

モデルケース：システムの構成イメージ



モデルケース：モバイルアプリケーションのイメージ

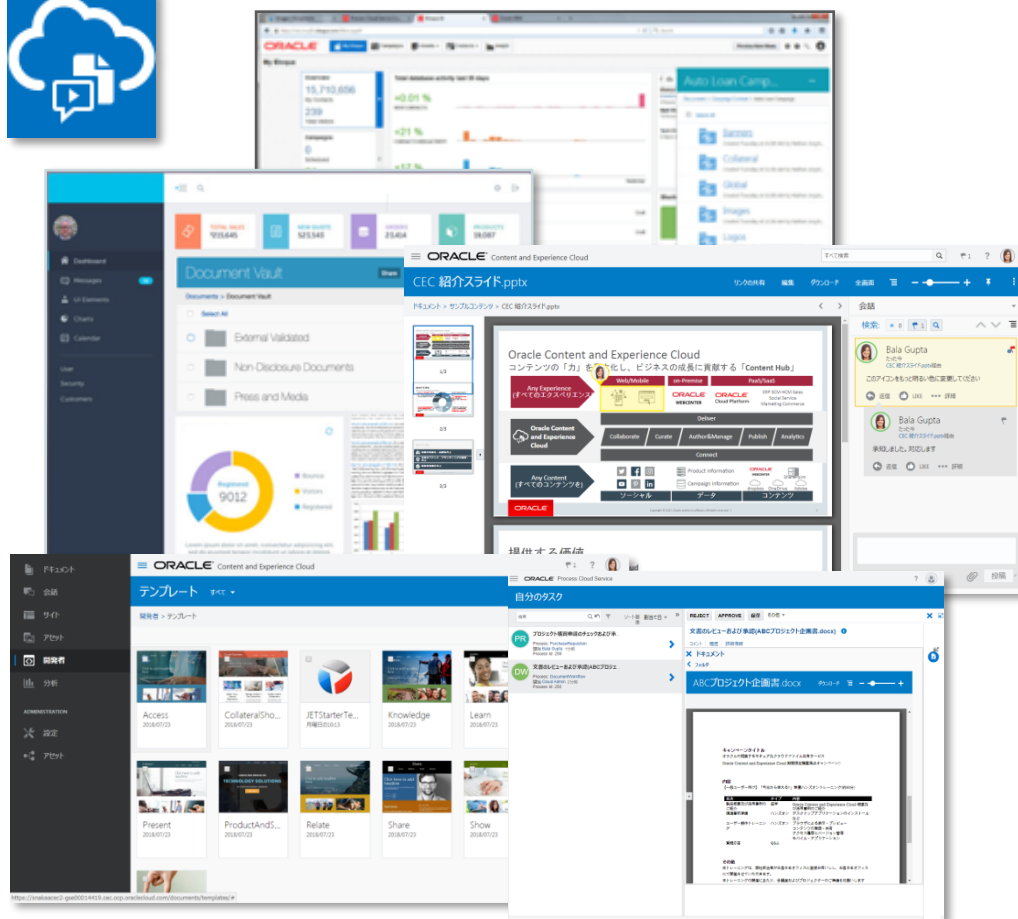


Demonstration

モデルケース：今後の展望

- 売上げ状況/利益状況を見ながらクーポンを企画したい
 - 企画・レビュー画面にオンプレミス上の売上げマスタの情報をグラフで可視化する
- 会計時に複数のクーポンをスムーズに選びたい
 - 発行したクーポンをスワイプで切り替えて表示する
- QRコード生成を自社内で完結させたい
 - QRコード生成ライブラリを使ってカスタム・コンポーネントを作りアプリケーションに組み込む
- **クーポンの画像のデザインを作りたい**
 - 商品画像に合わせたクーポン画像を外部のデザイナーに依頼
 - レビューしながらコンテンツの内容を精査する
- **集めた顧客プロフィールと売上げ情報を傾向分析したい**
 - 顧客プロフィールと売上げ/利益率の傾向の関連性を可視化しながら抽出したい

Oracle Contents and Experience Cloud コンテンツ/サイト/プロセスをシームレスに統合した



コンテンツ作成の迅速化を追求

- 版管理/レビュー/サイト公開までコンテンツ作成の一連タスクを直観的なインタフェースで実施

開発不要の洗練されたサイト構築

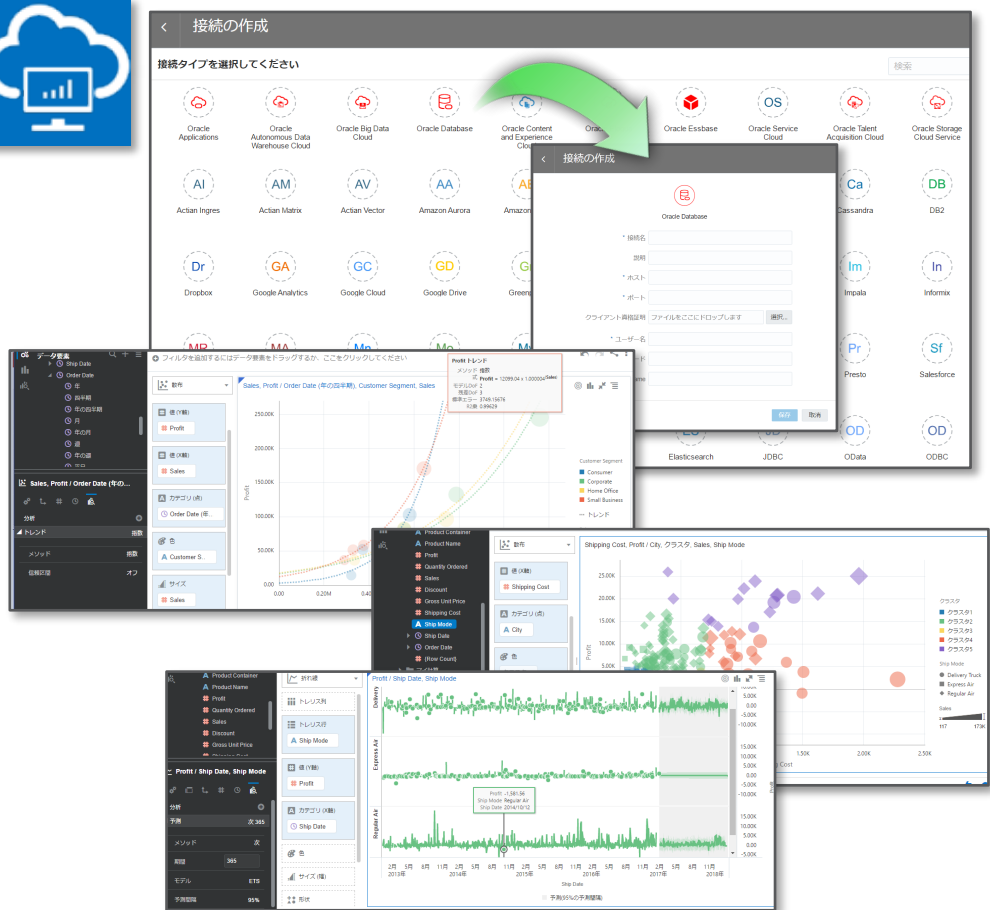
- マルチチャネル向けWebサイトをコンテンツやデザインフォームをデザインタイムで即座に構築

業務タスクとのシームレスな連携

- 業務アプリケーションへのUI統合や業務プロセス連携、SaaS連携まで幅広い業務連携を実現

Oracle Analytics Cloud

直観的なデザインタイムとAIによる傾向分析で実現するセルフサービスBI



デザインタイムで即座に分析

- 分析の準備から可視化まで全工程をデザインタイムで実現するセルフサービスBI

様々なデータソースを横断分析

- データベース、ファイルからSaaSまで多種多様のデータソースの横断分析を即座に実現

機械学習による傾向分析

- 線形回帰・時系列・クラスタリング等トレーニングに基づくAI型分析から傾向分析の精度を向上

模索しながら開発するサービスの重要タスク

計画・設計の頻度を高め確実にニーズを捉える機会を逃さない！

プロトタイピングによる想定ニーズの具現化

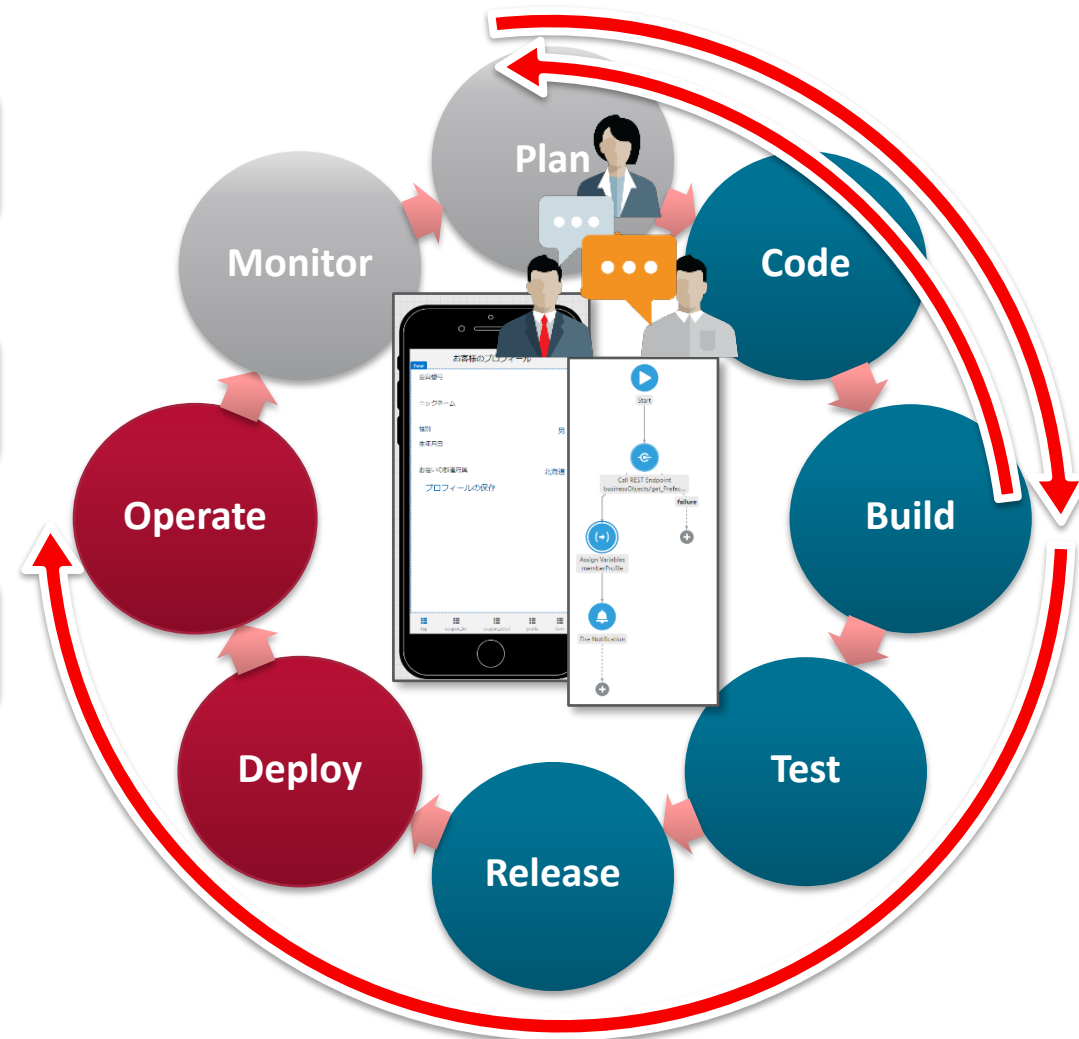
- 頻繁かつ容易に試行錯誤によるサービス設計・実装

トライアルによる実ニーズの把握

- 素早い本番移行により早期フィードバックの実現

コードレス開発によるサービス開発の加速

- 確実なイメージを掴みながらアプリケーション設計
- 生産性の低い技術的/単純作業の省力化



Oracle Integration Cloud

コードレス開発によるサービス開発のDevOpsイテレーションの迅速化



こんな時、かけこむ会社が増えています。



ビジネスプロセスを
改善したい!



今のシステムは
使いにくい!



システムコストを
下げたい!



パフォーマンスを
良くしたい!



経営分析を
したいのだが...



どんなソリューションが
あるの?



見積りはどれくらい
なんだろう?



楽に管理を
したい!

Oracle Digitalは、オラクル製品の導入をご検討いただく際の総合窓口。
電話とインターネットによる直接的なコミュニケーションで、どんなお問い合わせにもすばやく対応します。
もちろん、無償。どんなことでも、ご相談ください。



お問い合わせは電話またはWebフォーム

☎ 0120-155-096

受付時間 月～金 9:00-12:00 / 13:00-17:00
(祝日および年末年始休業日を除きます)

<http://www.oracle.com/jp/contact-us>

ORACLE®