

APS Summit 2017

階層間連携言語 *iHCL*
ISO TR16161の審議経過と
レベル2 実証評価について

2017年12月6日

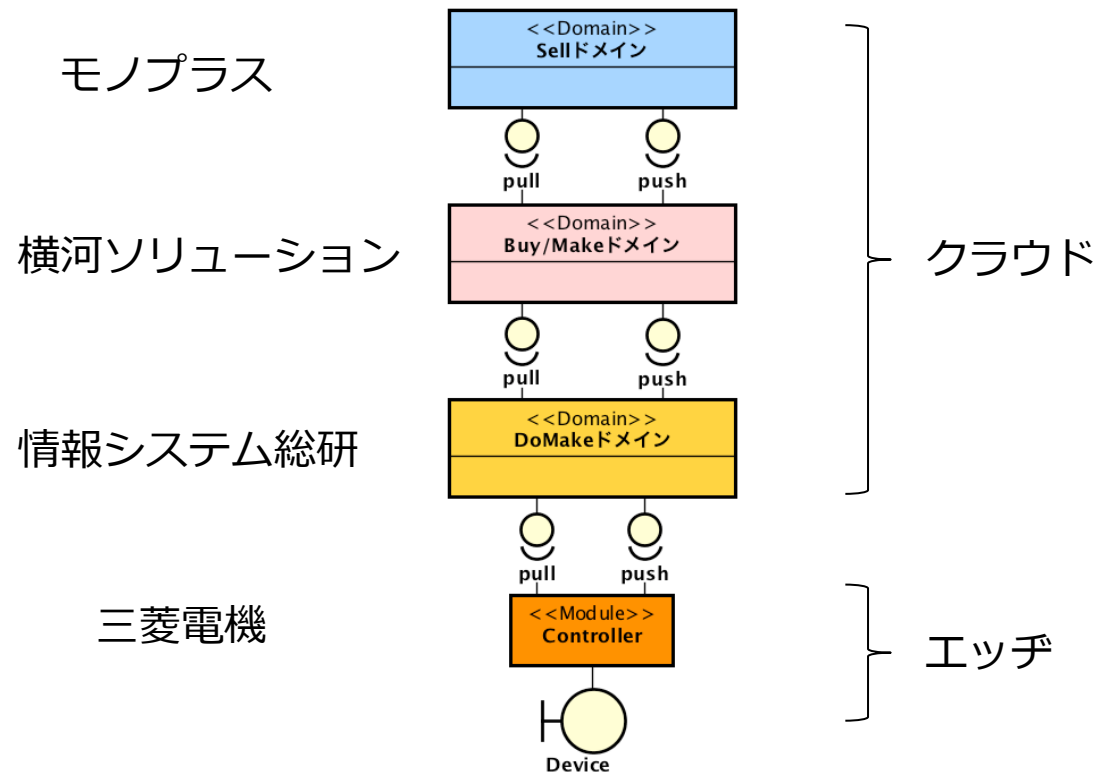
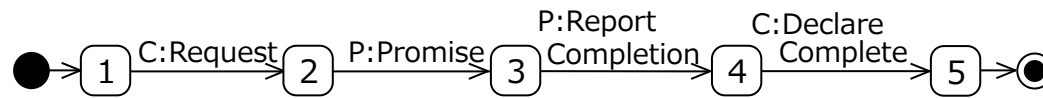
MESX-JP
APSOM

お話の流れ

- 去年から何が変わったか
- *iHCl* の概要
- *iHCl* 2.0
- *iHCl* アダプタの実装公募
- 実証評価レベル2のお誘い

去年から何が変わったか

- レベル1の実証評価を終了
- 規格書を国際標準（TR）へ
- 評価結果を反映して*iHCl* 2.0の設計と公募

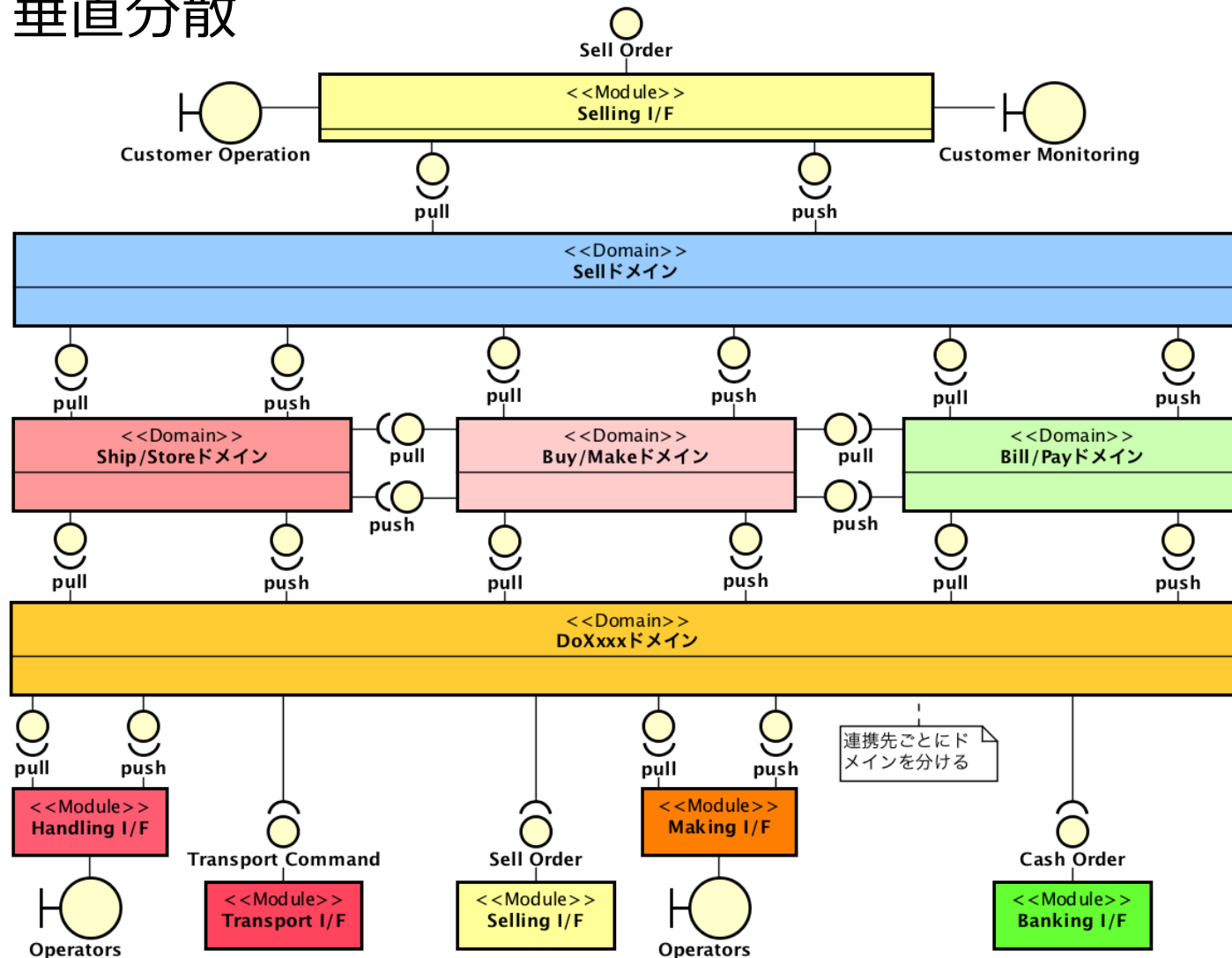


*iHCl*の概要

- 生産システムを 機能要素を組み合わせで作る
 - 機能要素を 定義して識別する
 - MSU (Manufacturing Software Unit)
 - Capability Profile (ISO 16100-*)
 - 機能要素間の対話構造
 - Customer-Performerモデルに基づく
 - メッセージ言語とプロトコル
 - MSUのモデルにニュートラルな言語
iHCl (inter-Hierarchy Collaboration Language)
 - 疎結合 (一方向参照の徹底, 非同期通信)
 - リアルタイム連携
 - 通信アダプタの提供
- 一枚岩 (モノリシックな) システムを砕く
 - 合理的な機能要素に分解する (どこで切るか)
 - アプリケーションインタフェース (どうつなぐか)
 - 部分的かつ段階的にシステムを再構築する

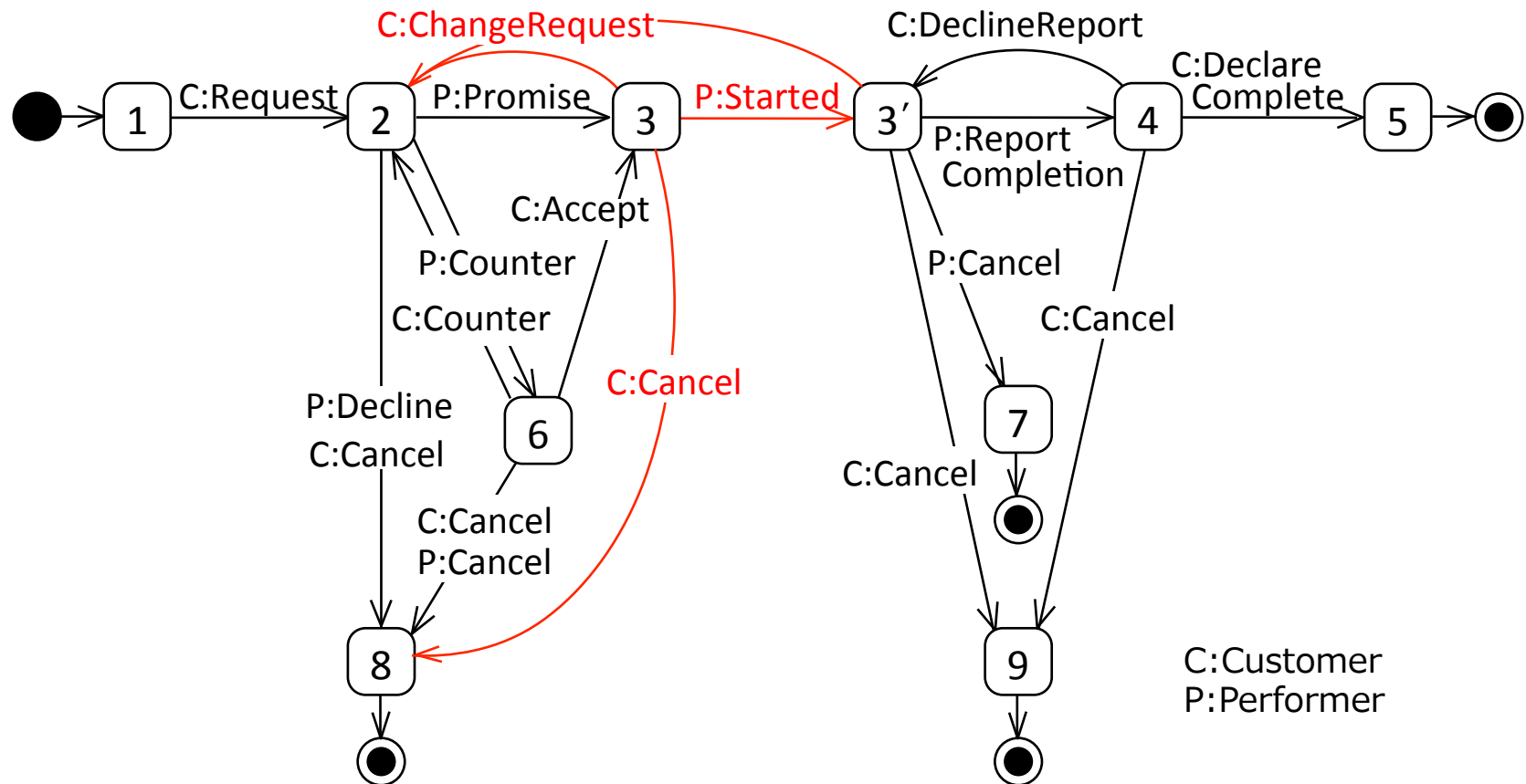
iHClの概要

- 生産システムを構成する機能要素 (MSU)
 - 垂直分散



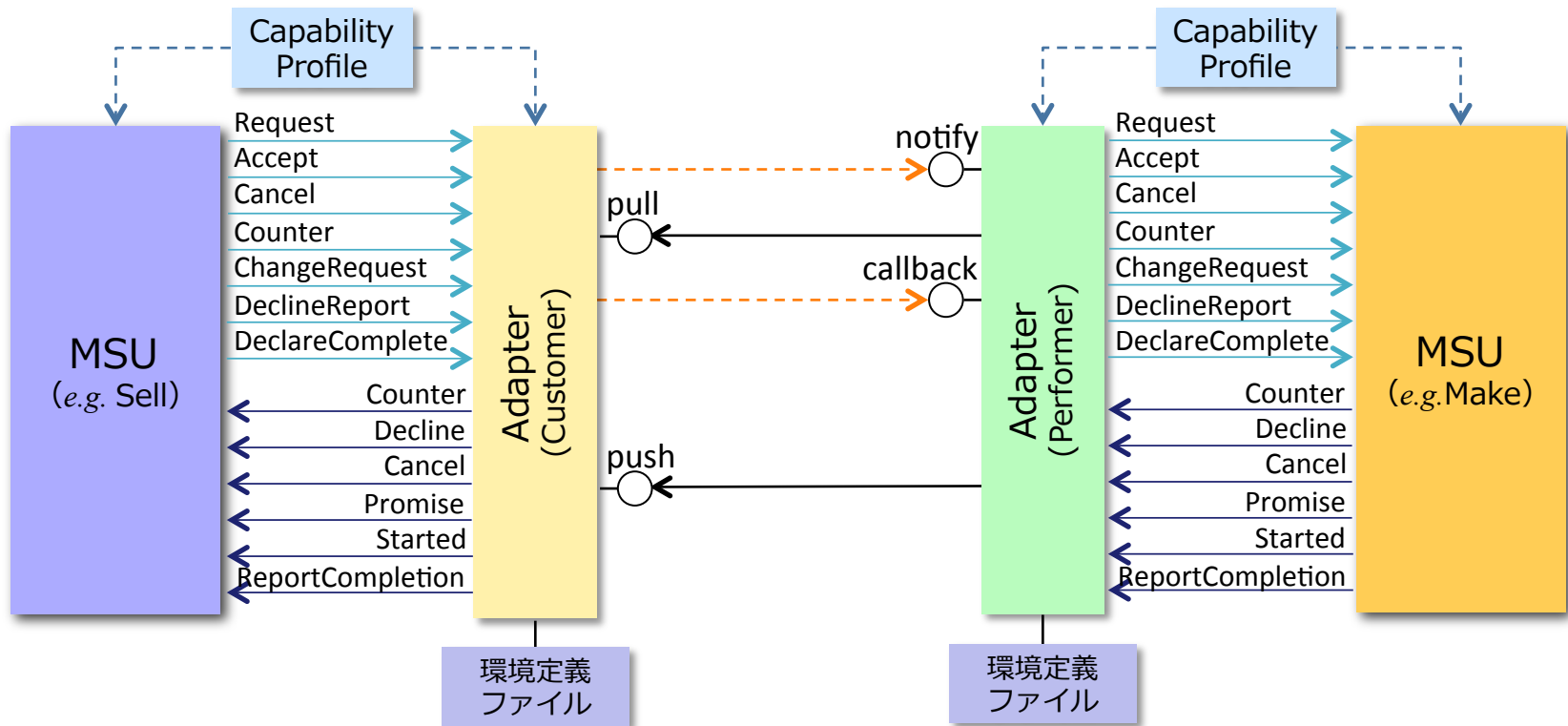
*iHCl*の概要

- 機能要素間の対話構造
 - ビジネスプロトコル：9状態モデル
 - Customer (Requestし, 結果を受け取る機能要素)
 - Performer (Promiseし, 実行する機能要素)



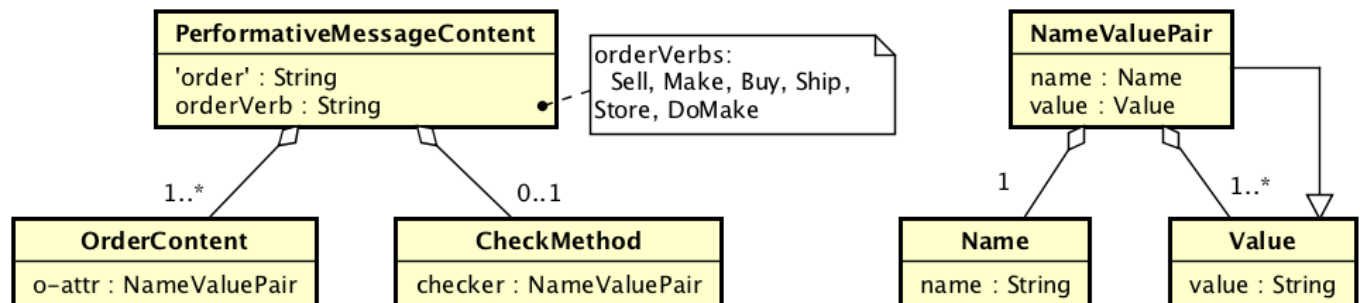
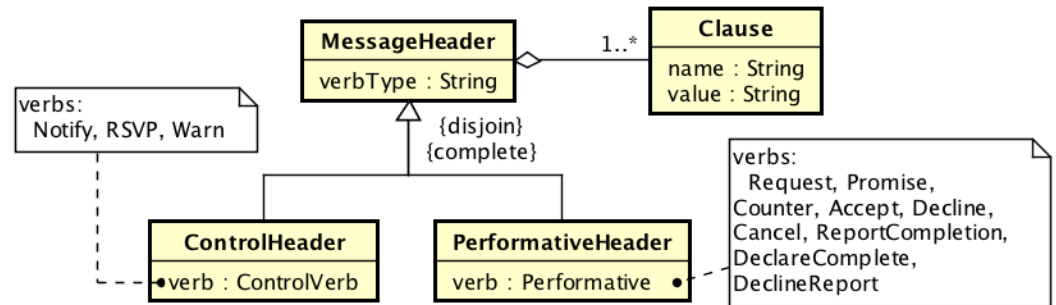
iHClの概要

- 対話の実装
 - Capability Profile
 - 機能要素のCapability (Manufacturing Activity)
 - 対話のためのチャネル定義 (Information Exchange)
 - 対話の確立
 - 相手のチャネル定義を自分の環境定義ファイルに取り込む



iHCLの概要

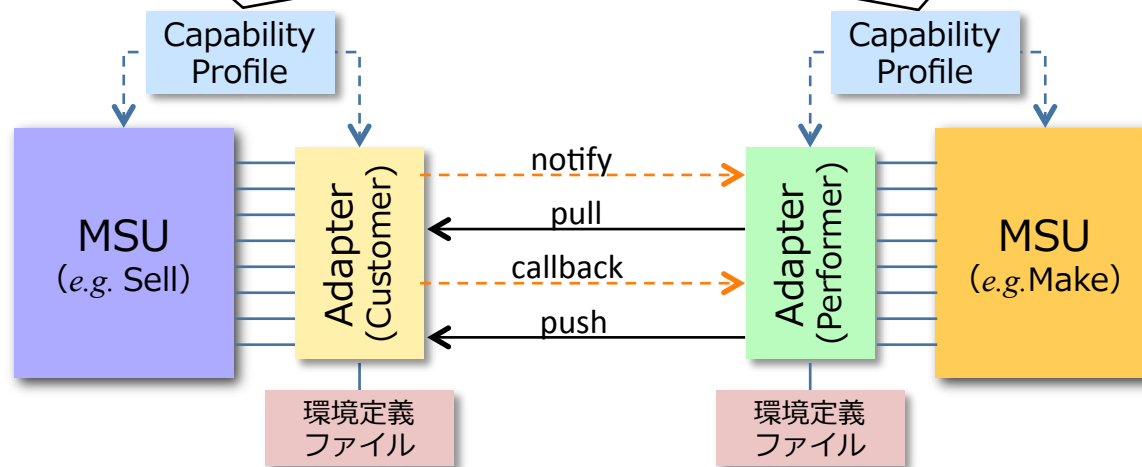
- メッセージの種類
 - 制御メッセージ
 - 業務メッセージ
- メッセージの構文
 - メッセージヘッダ
 - メッセージ本文
 - 人が読める
 - 自動テスト可能



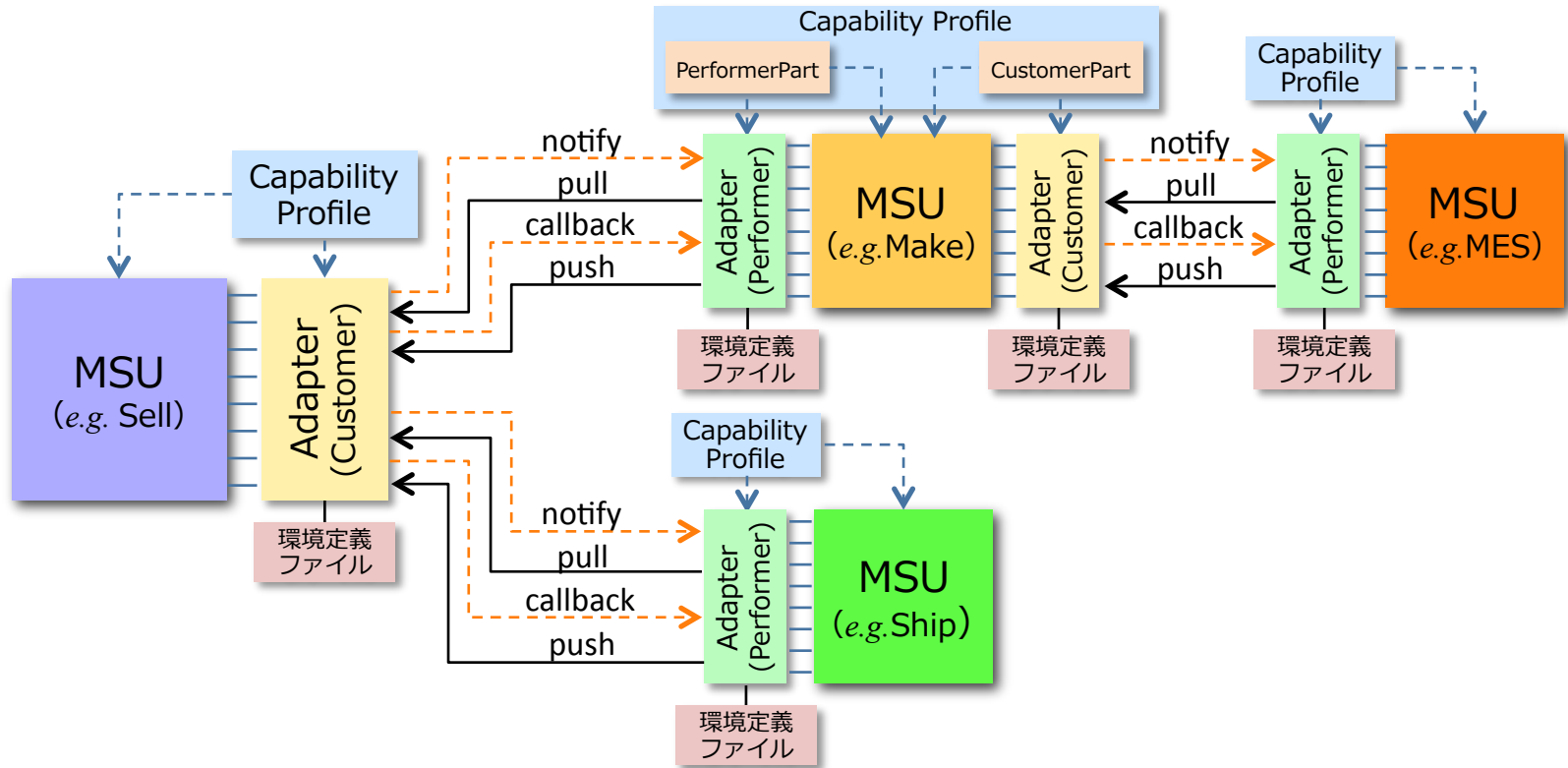
- CapabilityProfileの例
 - 機能と接続情報の定義

```
<CapabilityProfiling xmlns:xsi=...">
  <CapabilityProfile date="2018-08-10">
    :
    <Specific>
      <Activity id="Sell">
        <InformationExchange>
          <ApplicationProtocol id="Customer-Performer"/>
          <BasicProtocol id="REST"/>
          <Part type="Customer">
            <Channel type="pull" address="...."/>
            <Channel type="push" address="...."/>
          </Part>
        </InformationExchange>
      </Activity>
    </Specific>
  </CapabilityProfile>
</CapabilityProfiling>
```

```
<CapabilityProfiling xmlns:xsi=...">
  <CapabilityProfile date="2018-08-10">
    :
    <Specific>
      <Activity id="Make">
        <InformationExchange>
          <ApplicationProtocol id="Customer-Performer"/>
          <BasicProtocol id="REST"/>
          <Part type="Performer">
            <Channel type="pull" address="...."/>
            <Channel type="push" address="...."/>
          </Part>
        </InformationExchange>
      </Activity>
    </Specific>
  </CapabilityProfile>
</CapabilityProfiling>
```



- 複数のPerformerと対話する
 - 現状の*iHCl*の仕様で可能→実証する
 - 通信アダプタ実装仕様書に明記する



- 機能要素間で受け渡す注文内容が異なる
 - CapabilityProfileでチャネル別にデータ項目を定義する
 - MSUでの対応は必要

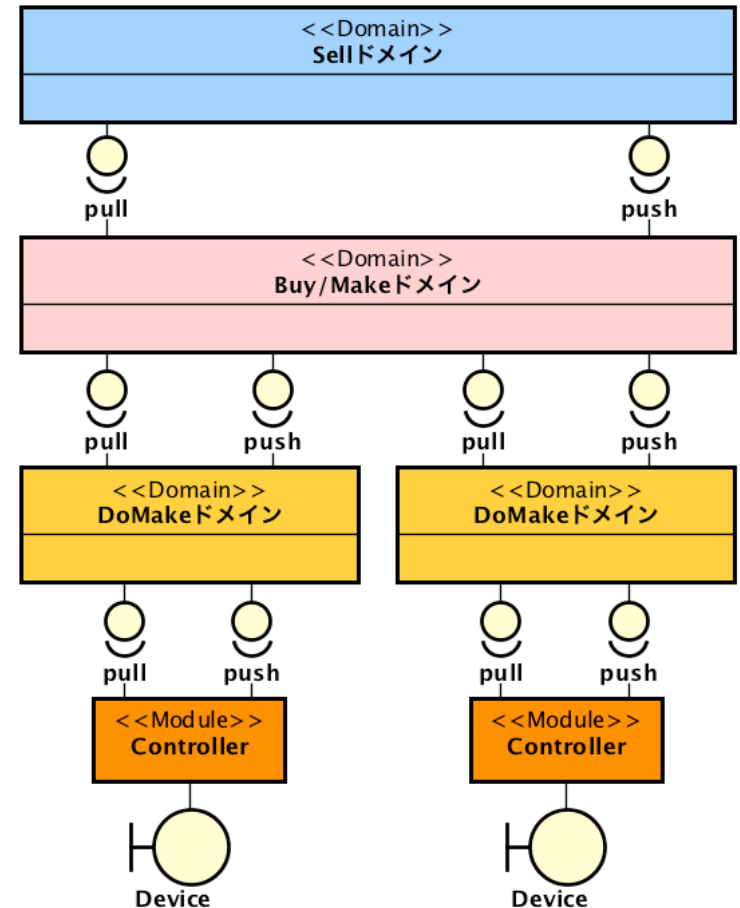
```
<Part id="Customer" />
  <Channel type="pull" address="http://pullのアドレス">
    <Method type="POST" name="..." />
    <Message name="Request">
      <Data name="o-id" type="Identifier"/>      <!-- 注文識別 -->
      <Data name="what" type="Service"/>        <!-- 操作名 -->
      <Data name="spec" type="JSON"/>           <!-- 注文品仕様 -->
      <Data name="qty" type="Size"/>             <!-- 量 -->
      <Data name="start_from" type="DateTime"/> <!-- 開始時刻 -->
      <Data name="end_to" type="DateTime"/>     <!-- 終了時刻 -->
      <Data name="option" type="JSON"/>         <!-- 追加情報 -->
    </Message>
  </Channel>
  <Channel type="push" address="http://pushのアドレス">
    <Method type="POST" name="..." />
  </Channel>
</Part>
```

iHCl アダプタの実装公募

- アダプタの実装を公募する
 - RFP（公募条件）
 - APSOMが改変可能，公開配布可能とする
 - 実装者に瑕疵担保責任なし
 - V1.0のソースコード開示
 - 公募内容
 - 規格書
 - 実装仕様書
 - 検収テスト仕様書
- オープンソース化
 - APSOMメンバにはフリー
 - 当面MESX-JPで保守

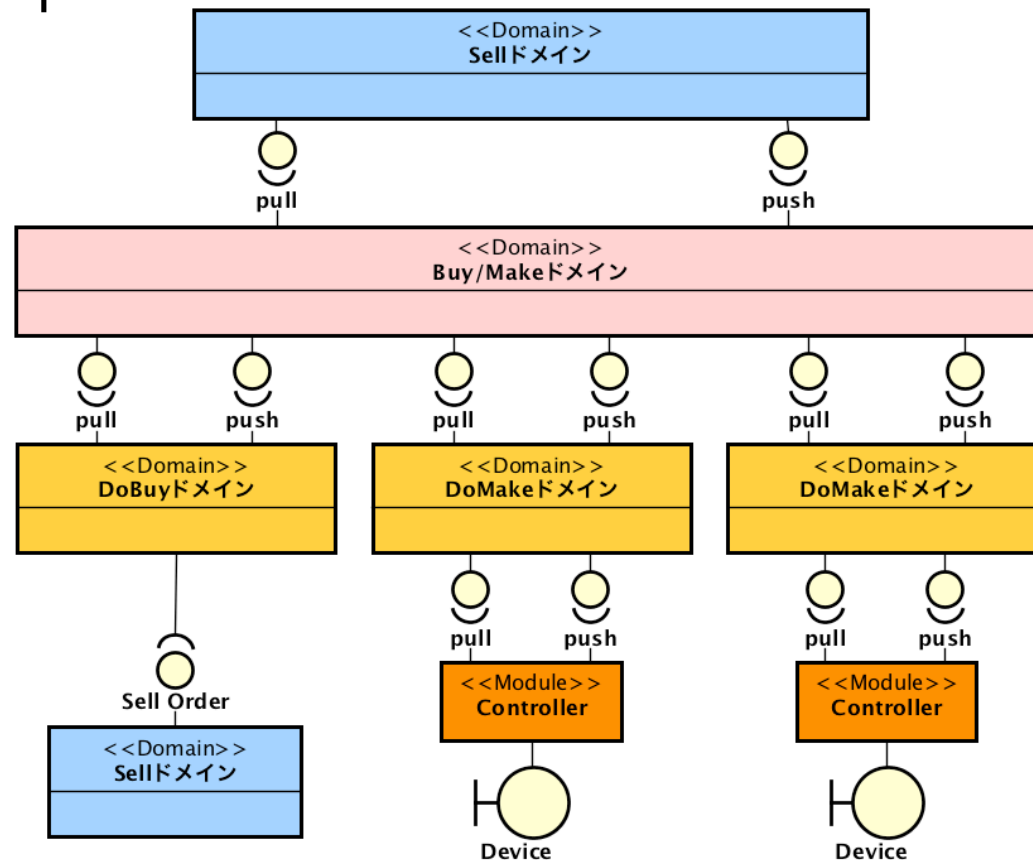
実証評価レベル2のお誘い

- レベル2
 - 複数のPerformer, 工程間連携
 - 異常系メッセージ
 - 多様なメッセージ
- ベンダ
 - MSUの提供を受ける
- ユーザ
 - ユースケースの提供を受ける
 - リアルな例外処理



実証評価レベル2のお誘い

- ちなみにレベル3では
 - 自動テスト可能に
 - 外部連携する
 - セキュリティ



まとめーMESXのめざすこと

- 変化を抱擁する
- *iHCl*の採用によるメリット
 - ベンダ
 - － よいコンポーネント→品質の向上
 - － 機能に集中→投資効率の向上
 - － 国際標準による接続性の向上→ビジネスチャンス
 - － SaaSとしての新規ビジネス
 - 利用者
 - － 開発費用の縮小
 - － リアルタイム連携
 - － システムの拡張性（生産システム以外にも）
 - － 段階的構築，短期立ち上げ可能
 - － 機能拡張・縮小が容易
 - － 国際標準によるMSUのとっかえひっかえ