
MESXプロジェクト活動報告

～ 計画/実行/制御層の情報連携とその実証

MfgX MESXジョイントプロジェクト
三菱電機(株) 岩津 賢

内容

- MESXとは
- MESX検討成果
- 実証システム

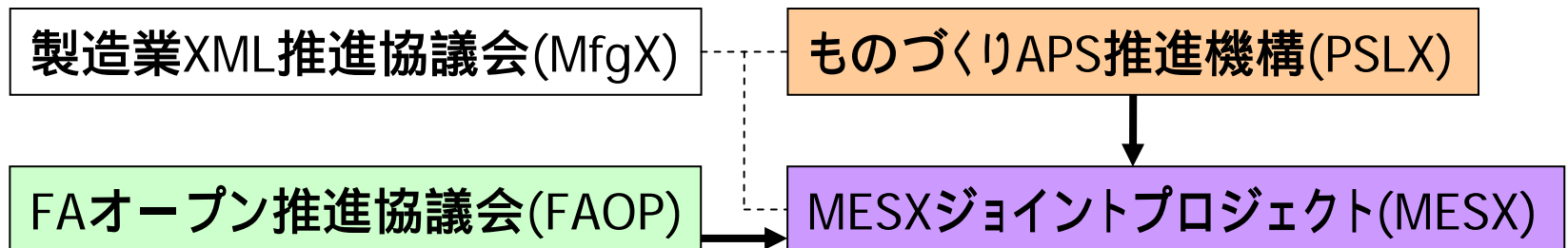
MESXジョイントプロジェクト

■ 目的

- APSとMESとをXMLにより連携させる仕組みの確立
- 仕様のオープン化によるインタフェース設計の一元化

■ 組織

- PSLXとFAOPによるMESX-JWGとして設立(03/4)
- MfgX内にMESXジョイントプロジェクトとして設立(06/4)
 - 名称は5/15のMfgX運営委員会にて決定



MESX



- PSLXおよびFAOPで実行系の上下の層を検討
 - PSLX(計画系)ではAPS間、APS/ERP間のI/Fを規定
 - FAOP(制御系)ではデバイス/コントローラ/装置のI/Fを中心に検討
- ERP/APSと装置/コントローラの間にはMESがある
この間を記述できなければシステム全体をトータルに表現できない
PSLXとFAOPのジョイントで、XMLを利用した
実用レベルの仕組みを構築したい

これまでの経過概要

- 設立趣意書(03/4発足)
 - PSLXとFAOP-MICXメンバを中心にスタート
- 他標準の動向や進め方の枠組みをまとめた白書発行
 - ホワイトペーパー完了(03/5)

各層の定義などはIEC 62264 (S95)に沿う
インタフェースはエージェント間/オブジェクト間
通信プロトコルとしてメッセージをXML定義する

- ユースケース/モデル/インタフェース/メッセージ設計
- 実証システム開発完了(06/2)
 - 製造業XMLシンポジウム2006(06/2)にてデモ実施
 - PSLX技術フォーラム2006(06/6)にてデモ実施

標準制定の手順

1. ユースケースの獲得

- アクタ、目的、機能の実行による変化、アクタとシステムの対話、などを含む「ユースケース記述」で表現

2. タイプ図の作成

- UMLのクラス図の表記法で記述
- 実装方法を規定しないクラス、データ属性と操作

3. インタフェースの設計

4. メッセージの設計

- KQML(Knowledge Query and Manipulation Language)を参考に、命令部とデータ部、制御部で構成
- XMLでエンコードする

ユースケースの例

ユースケース名：異物混入の出荷品を特定する

アクタ：

品質保証担当者、生産技術者

目的：

適正な製品だけを出荷したい。

事前条件：

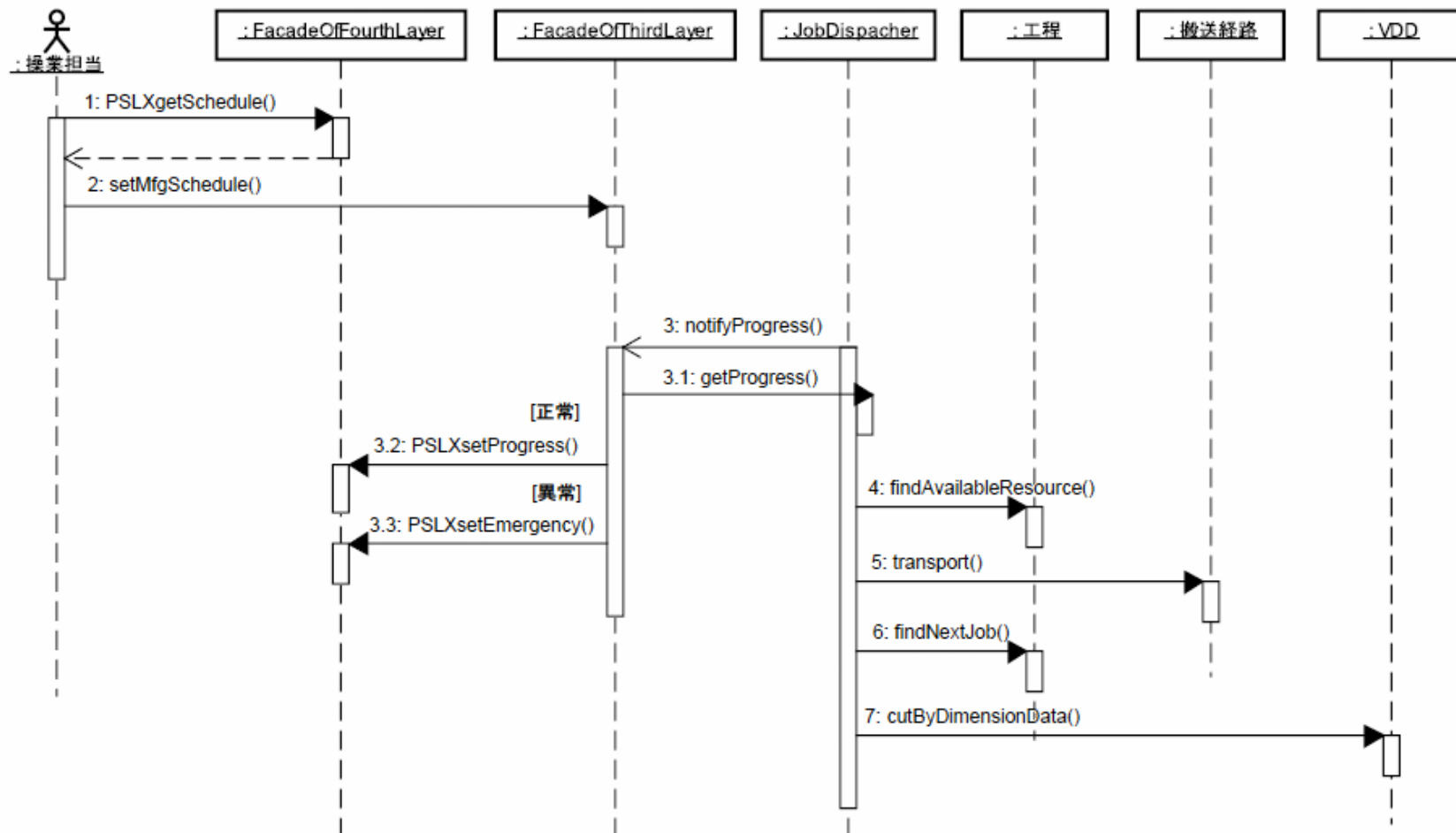
異物混入の発生した、系列、時刻(期間)、装置がわかっている。

手順：

- 1.アクタは異物混入の通知を受けて、このユースケースを起動する。
- 2.アクタは、系列、時刻(期間)、装置をシステムに入力する。
- 3.システムは、該当する時刻に該当装置内にあった仕掛品LOTを検索する。
- 4.システムは、該当する仕掛品LOTが使用された全出荷品と仕掛品を検索する。
- 5.システムは、該当する全出荷品、仕掛品を画面に表示し、印刷する。

⋮

シーケンス図



MESXインタフェース設計の原則

(1) インタフェースの命名

- **4-3層間**はPSLXの標準に従い“**PSLX**”なるprefixを付ける
例: PSLXgetSchedule 製造指示を受ける
- **3-2層間**は“**MESX**”なるprefixを付ける
例: MESXloadRecipe 指定した資源にレシピをロードする

(2) メッセージの設計

- KQMLで記述した後、PSLXの規約に準拠したXMLタグで実装する

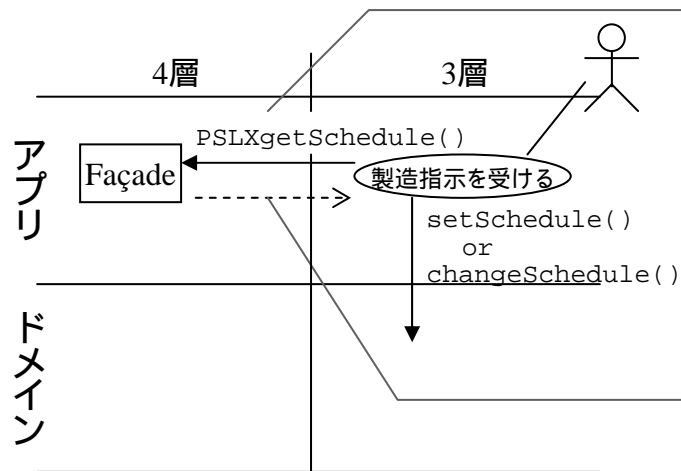
(3) データの渡し方

- プラットフォームおよびテクノロジーへの依存度を下げるために、
層間は粗結合とする
- オブジェクト渡しとせず、オブジェクトの名前で呼ぶ
 - 名前空間の管理が必要

(4) Façadeの想定

- 呼び出し元はアプリケーション(ユースケース実現)
- 呼び出し先は各層に設けられたFaçadeとする
オブジェクトの名前とオブジェクトそのものとの対応づけはFaçadeが行う

メッセージの設計例(1)



製造指示の送信を要求するメッセージ
(PSLXgetSchedule)

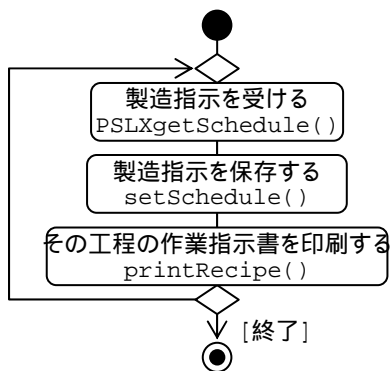
```

:message-type request
:sender usecase-toReceiveTheManufacturingOrders
:receiver facadeOf4thLayer
:request-number 0037
:content (ManufacturingOrder
          (profile,start,makeitem,consumeitem,resource,
           instruction))
:reply-with acknowledgment
    
```

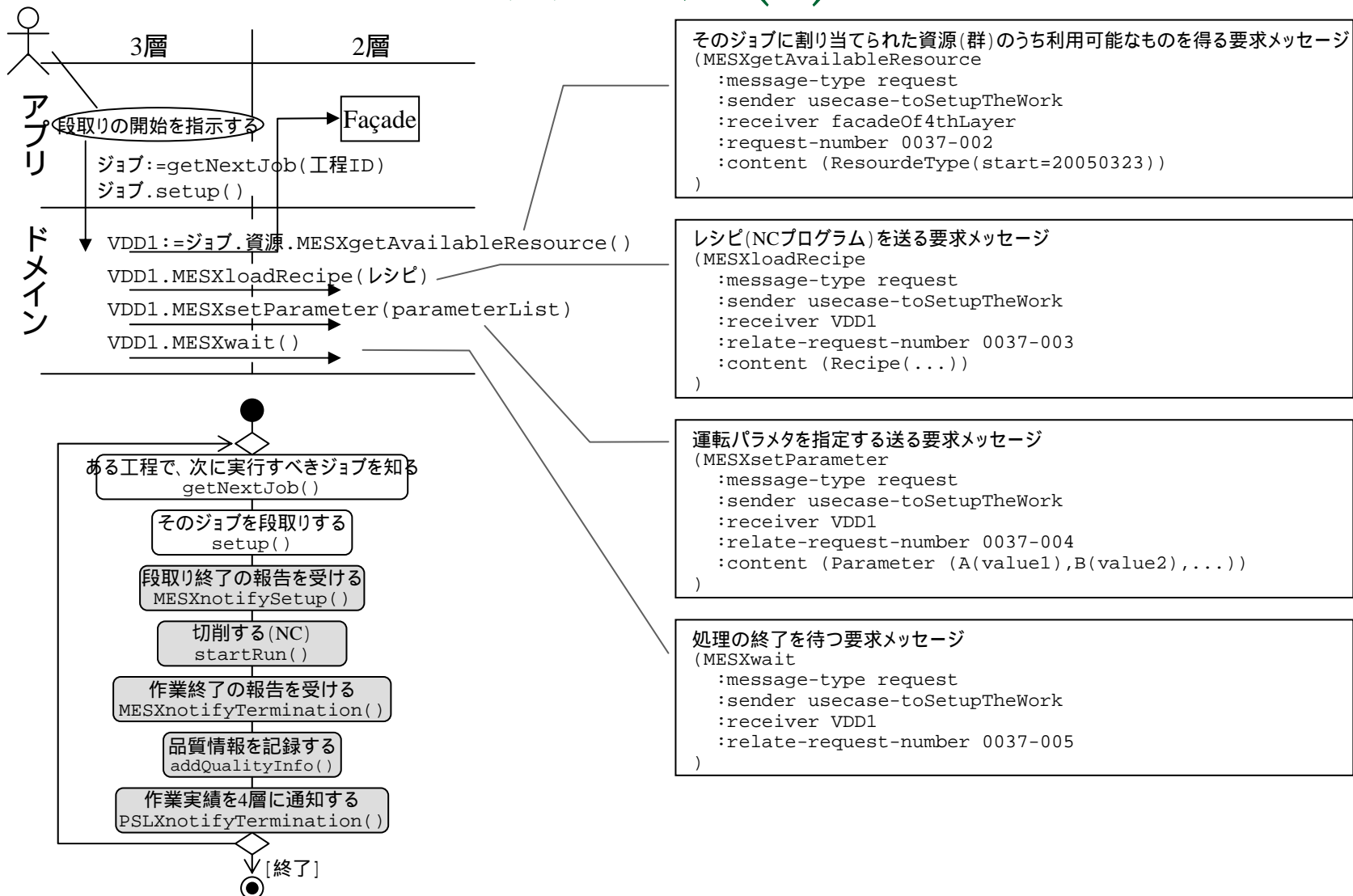
製造指示を送信する応答メッセージ
(PSLXgetSchedule)

```

:message-type response
:sender facadeOf4thLayer
:receiver usecase-toReceiveTheManufacturingOrders
:relate-request-number 0037
:content (
  ManufacturingOrder(#0001,
    start 20040303:182355,
    makeitem (rope(f=s w=10 l=2000m p=12 st=2t a=b),
             qty 32,container y),
    consumeitem (wire(f=s w=1 l=2222m),qty 320),
    resource(VDD001),
    instruction `do your best`)
)
    
```



メッセージの設計例(2)



定義したメッセージ

4層-3層

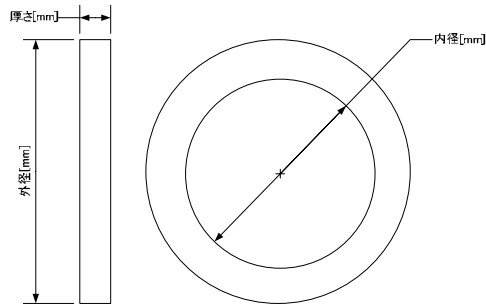
PSLX getSchedule	生産計画を得る
PSLX setProgress	製造実績を(ジョブ単位に)報告する
PSLX getProcess	製造手順を知る

3層-2層

MESX isAvailable	資源を指定して、現時点での使用可否を知る
MESX loadRecipe	資源を指定して、そこにレシピ情報を渡す。 資源はそのタイプに応じてポリモルフィックに振る舞う
MESX startExecution	資源を指定して、実行オーダ開始を指示する。 資源はそのタイプに応じてポリモルフィックに振る舞う
MESX getInformation	資源を指定して、実績情報を取得する
MESX wait	処理の終了通知を要求する(通知が来るまで非同期に待つ)
MESX notifyEvent	資源で発生した事象(終了/異常発生)を報告する(非同期)

実証デモ・対象工程

ワッシャー 製品概要

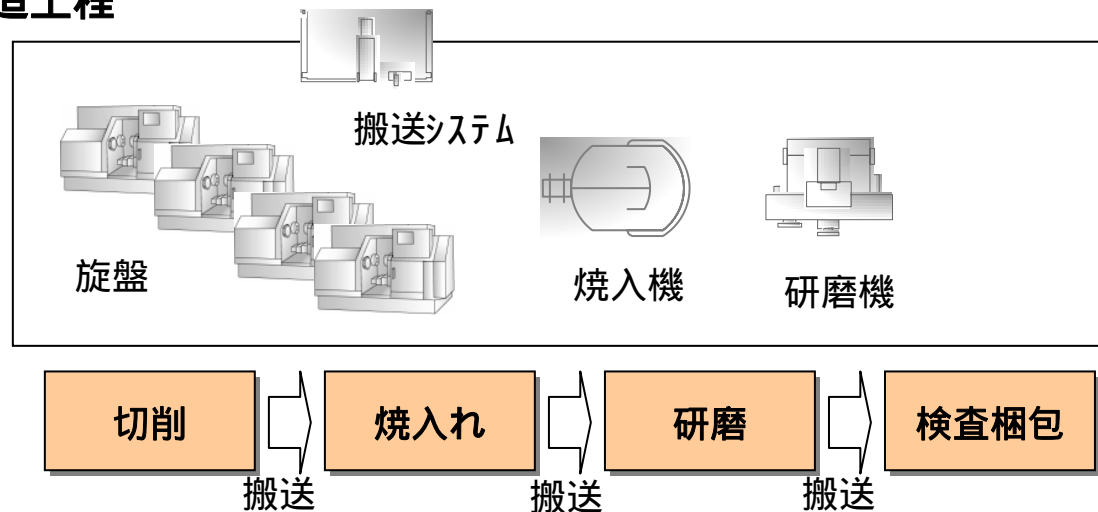


内径(mm)	10.0 ~ 40.0
外径(mm)	20.0 ~ 50.0
厚さ(mm)	1.0 ~ 5.0
箱入り数(個)	50 (一定)
強度(kg/mm ²)	50 ~ 80
面精度(μm)	10 ~ 100

ワッシャーオーダー例

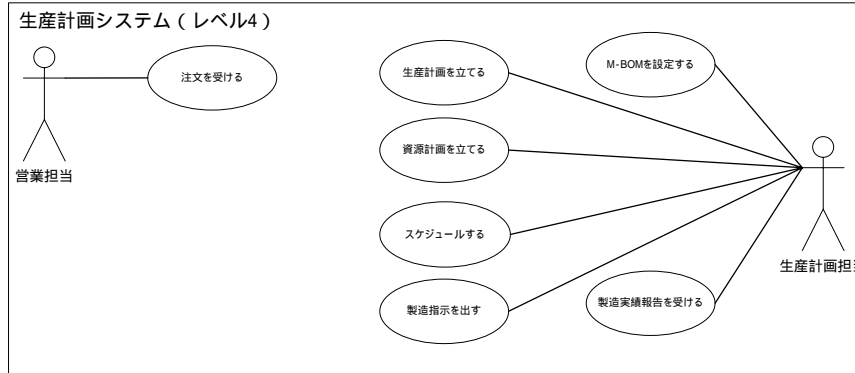
オーダー番号	個数	内径(mm)	外径(mm)	厚さ(mm)	強度(kg/mm ²)	精度(μm)
O0001	10	10.0	22.0	1.6	65	40
O0002	20	10.0	22.0	1.6	65	40
O0003	50	15.0	36.0	2.6	75	30
O0004	10	25.0	48.0	4.0	55	70
O0005	30	15.0	36.0	2.6	75	30
O0006	15	10.0	22.0	1.6	65	40
O0007	25	25.0	48.0	4.0	55	70
O0008	10	15.0	36.0	2.6	75	30
O0009	15	25.0	48.0	4.0	55	70
O0010	50	10.0	22.0	1.6	65	40

ワッシャー製造工程

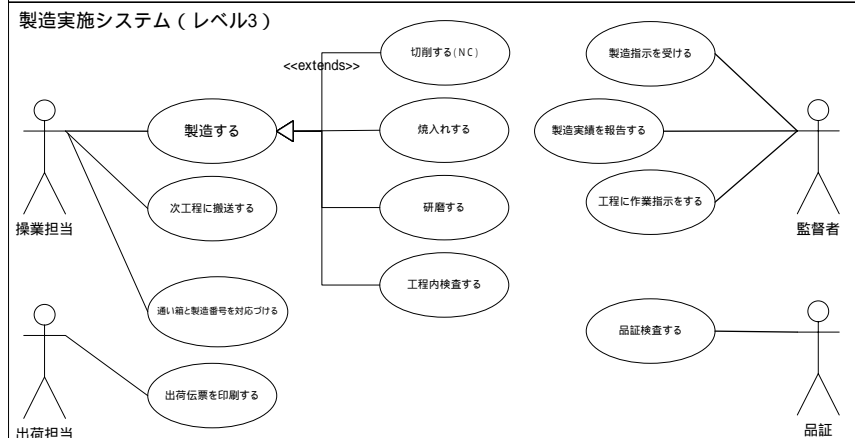


実証デモ・ユースケース

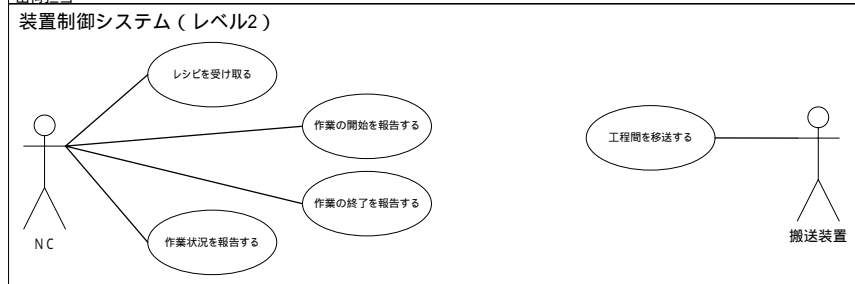
4層



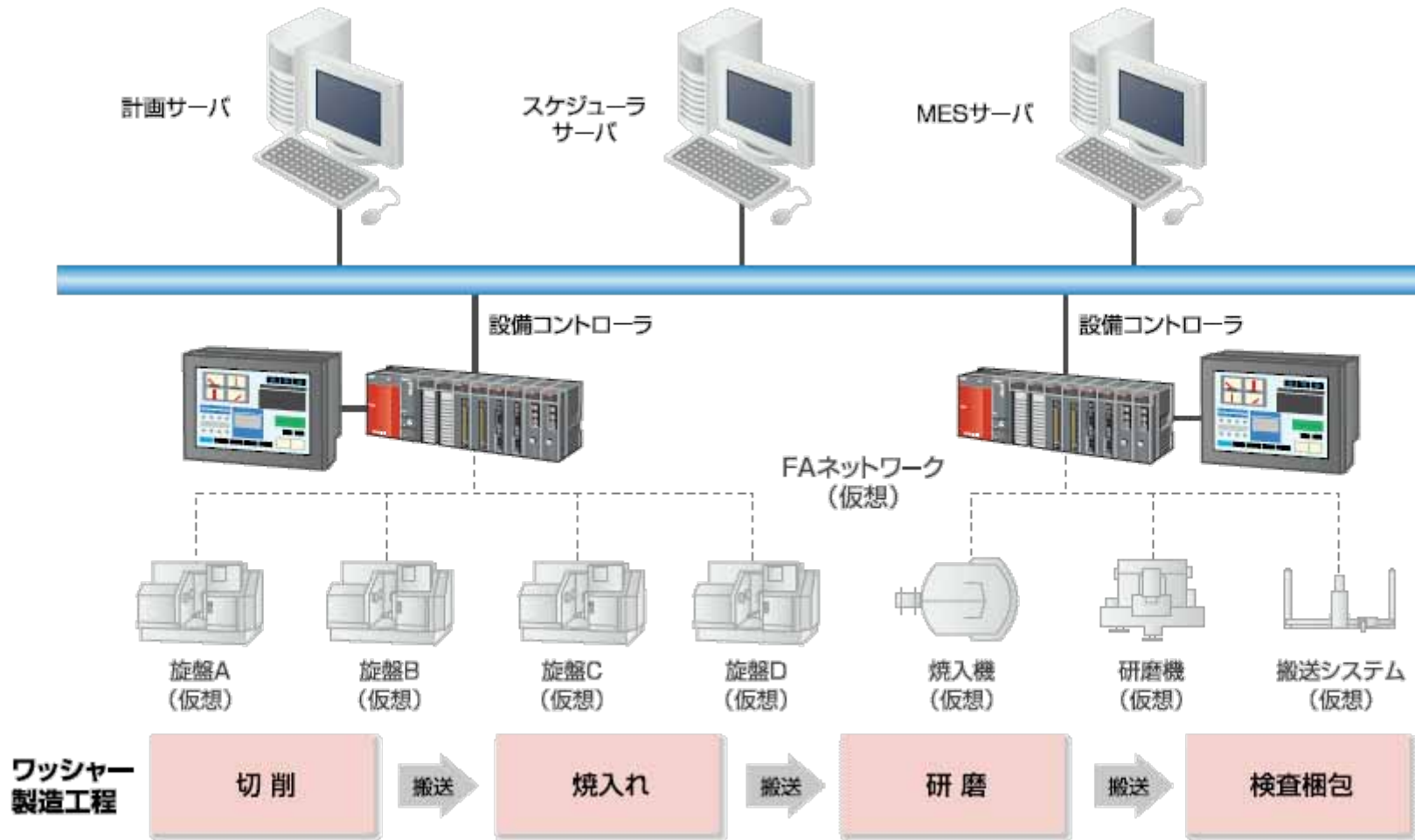
3層



2層



実証デモ・システム構成



実証デモ・機能構成

IEC 62264 (ISA-S95) (PSLX)

レベル 4B

エンタープライズ(企業)
サイト(拠点)
企業オーダー、拠点オーダー

レベル 4A

エリア(作業区)
作業区オーダー

レベル 3

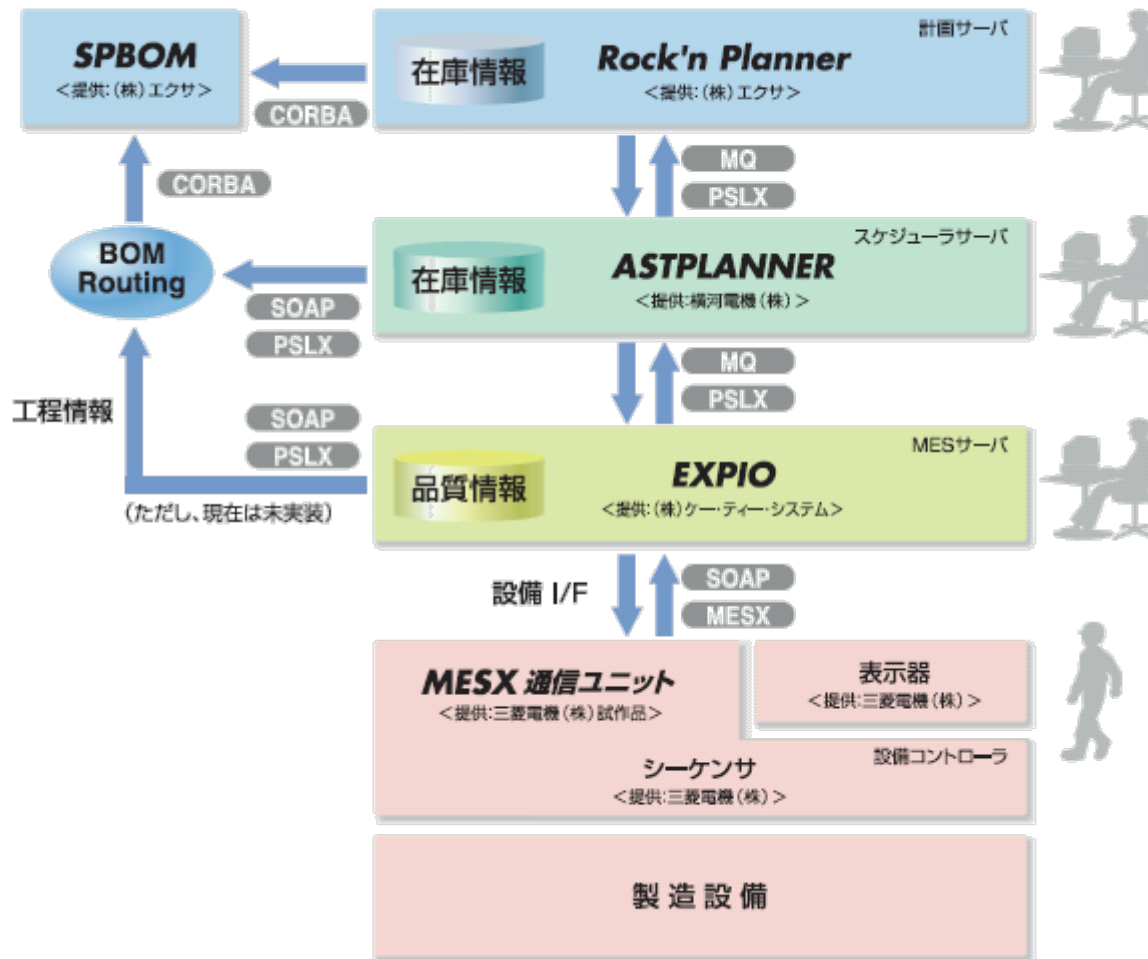
ワークセンタ(作業場)
ワークオーダー(作業指示)

レベル 2

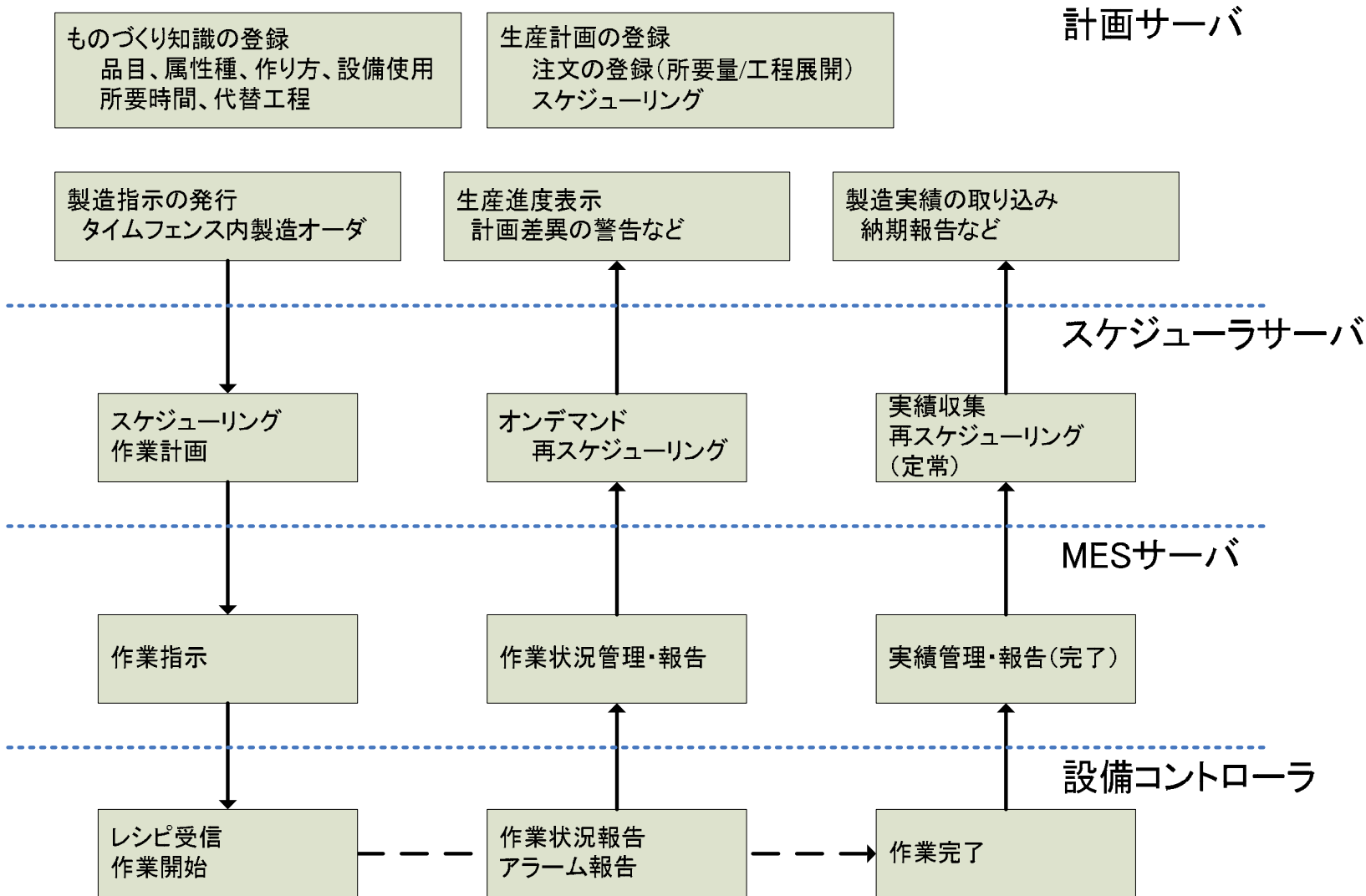
ワークユニット(設備)
エグゼキューションオーダー(実行指示)

レベル 1

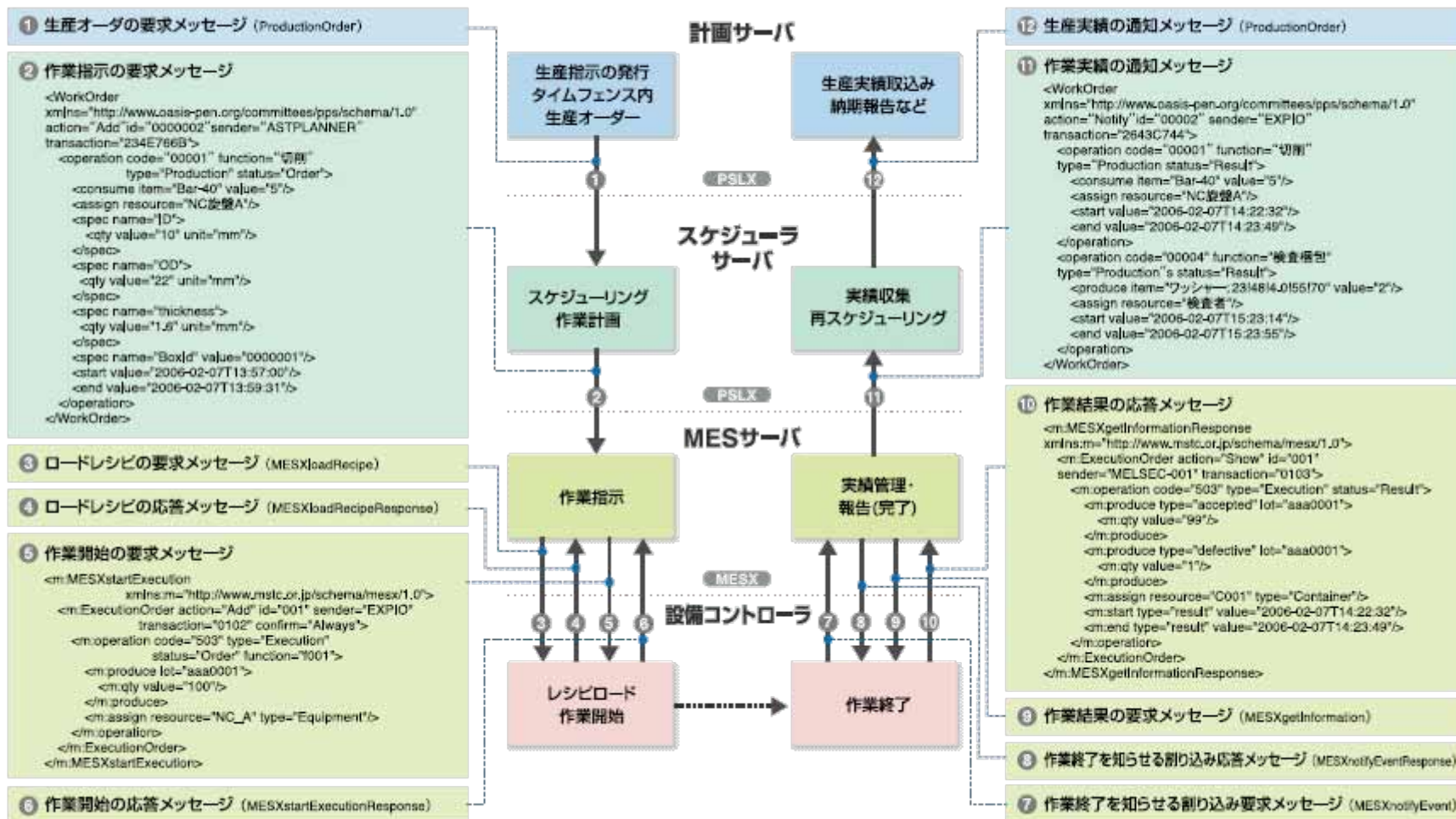
ワークステーション(装置)
インストラクション(動作指示)



実証システム・シナリオ



実証デモ・XMLメッセージ



メッセージ例(1) – 実行オーダ開始

要求メッセージ(3層 2層)

```
<m:MESXstartExecution
  xmlns:m="http://www.mstc.or.jp/schema/mesx/1.0">
  <m:ExecutionOrder action="Add" id="001"
    sender="EXPIO" transaction="0102" confirm="Always">
    <m:operation code="503" type="Execution"
      status="Order" function="f001">
      <m:produce lot="aaa0001">
        <m:qty value="100"/>
      </m:produce>
      <m:assign resource="NC_C"/>
    </m:operation>
  </m:ExecutionOrder>
</m:MESXstartExecution>
```

レシピのIDを
functionで指定

生産個数をqtyの
valueで指定

対象資源をassign
のresourceで指定

応答メッセージ(2層 3層)

```
<m:MESXstartExecutionResponse
  xmlns:m="http://www.mstc.or.jp/schema/mesx/1.0">
  <m:ExecutionOrder action="Confirm" id="001"
    sender="MELSEC-001" transaction="0102">
    <m:operation code="503" type="Execution"
      status="Order" function="f001"/>
  </m:ExecutionOrder>
</m:MESXstartExecutionResponse>
```

受け付けた
実行オーダのID

メッセージ例(2) – オーダ終了通知

要求メッセージ(2層 3層)

```
<m:MESXnotifyEvent
```

```
  xmlns:m="http://www.mstc.or.jp/schema/mesx/1.0">
```

```
  <m:ExecutionOrder action="Notify" id="001"
```

```
    sender="MELSEC-001" transaction="0234">
```

```
    <m:operation code="503" type="Execution"
```

```
      status="Result">
```

```
      <m:end/>
```

```
    </m:operation>
```

```
  </m:ExecutionOrder>
```

```
</m:MESXnotifyEvent>
```

発生した
事象の種別

MESが認識する
実行オーダのID

応答メッセージ(2層 3層)

```
<m:MESXnotifyEventResponse
```

```
  xmlns:m="http://www.mstc.or.jp/schema/mesx/1.0"/>
```

便宜上のレスポンス

実証システムによる評価

■ 規約化の効果

- 階層モデルの適用により、システム間の**結合度**を低くすることが可能(各システムの設計が独立)
- 階層ごとに共通概念に基づくモデルを使用したため、システム間の**インタフェース抽出、設計が容易**
- 共通のスキーマに基づくXMLメッセージの利用により、**プラットフォームの違いの意識が不要**
 - 接続に要した工数：10人日=5人×2日

■ 業務運用面での効果

- 設備やソフトウェア製品が容易に**入れ替え可能**
- 階層化と機能の明確化により、各層でのKPIによる**成績評価も可能**

KPI: Key Performance Index

まとめ

- 実証システムでメッセージ設計の妥当性を確認
 - IEC 62264の生産作業管理における情報連携を実現
- MESXジョイントプロジェクトの今後の活動
 - IEC 62264の品質・保守の各作業管理の領域に拡大
 - 事例収集によるユースケース分析、モデル化が必要
 - MESXジョイントプロジェクトへの参加募集中
 - 標準仕様の策定、IEC 62264-3以降へのインプット
 - PSLX (OASIS PPS)との整合性の維持
 - 機械振興協会との共同研究、常設展示、ORiN連携

おわり

ご静聴ありがとうございました