

DENSO

Crafting the Core

ネットワーク型データ構造による SW部品関係の可視化

～SW部品選択におけるグラフDBの適用と評価～

株式会社デンソー

川井 隆之* 小川 雄太 水藤 倫彰

アジェンダ

1.開発の現状

2.課題

3.アプローチ

4.実施方法

5.実施結果

6.考察

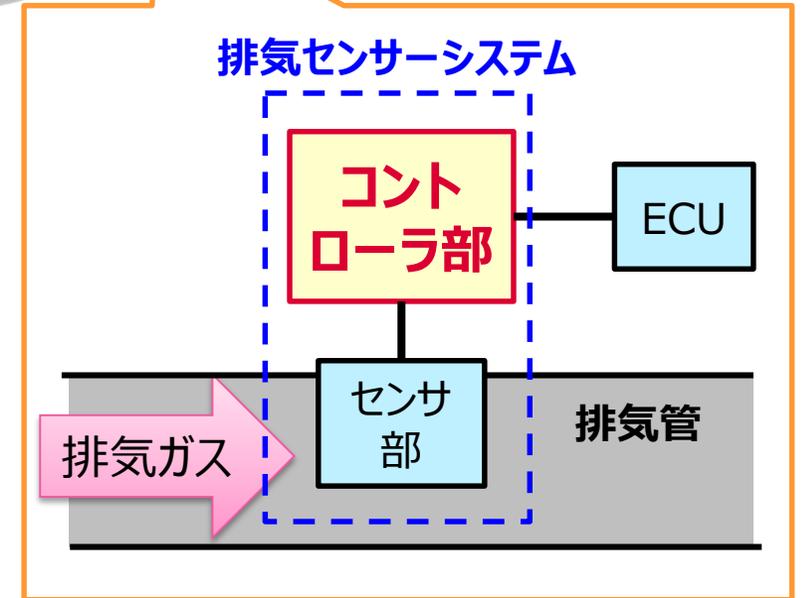
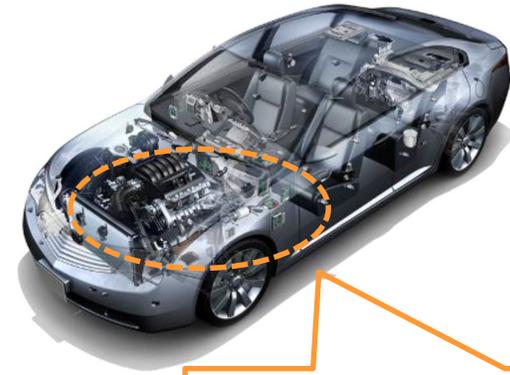
7.まとめ

1. 開発の現状（1）： 対象とするシステム

■ 開発製品の概要

排気センサーシステム

- 機能
 - 排気ガス中の成分を測定、成分情報へ変換
 - 成分情報をECU(Engine Control Unit)へ送信
- 構成
 - センサ部 …… 排気ガス中の成分と反応
 - **コントローラ部** …… センサ部の制御とECUとの通信
- 顧客
 - 国内外の主要自動車メーカー
 - メーカーによって要求は多種多様



排気ガス中の成分情報をECUへ伝えるシステム

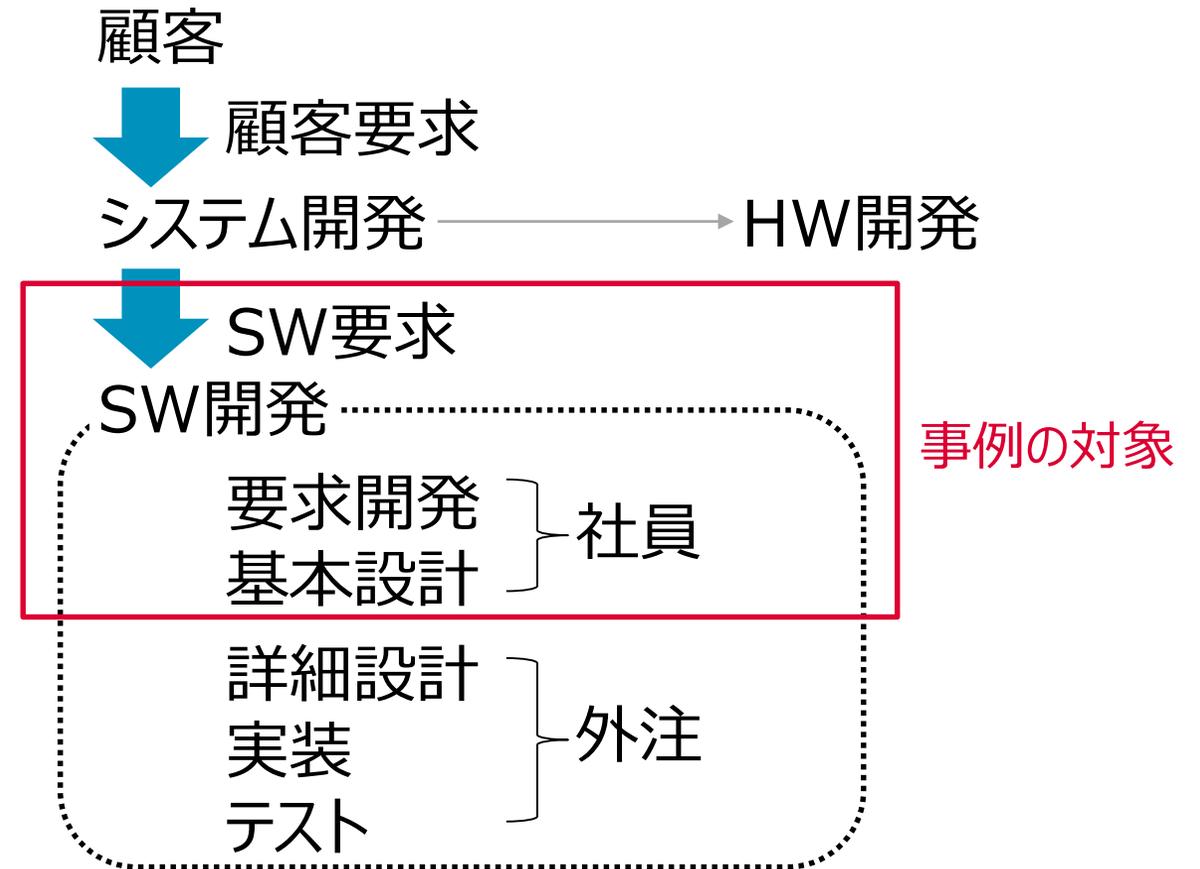
1. 開発の現状（2）： 組織と開発の流れ

■ 組織構成

- 排気システム技術部
 - システム開発Gr
 - HW開発Gr

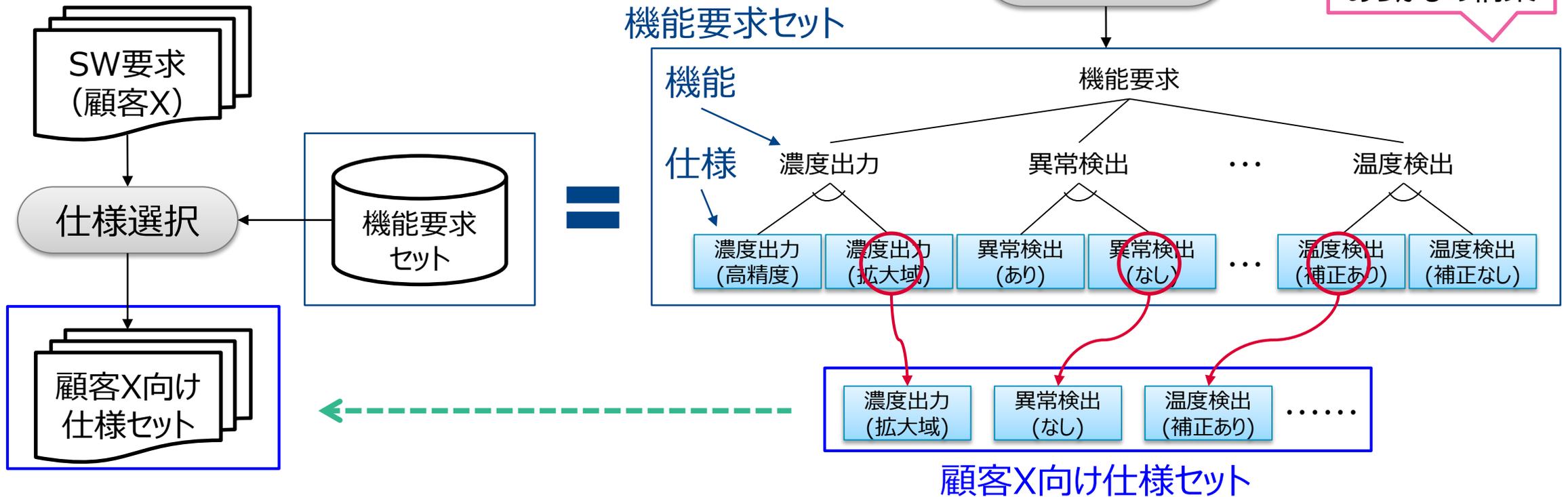
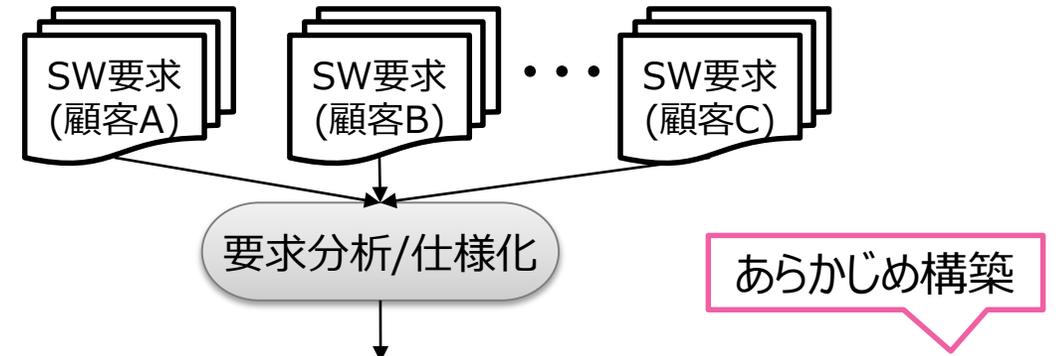
- SW開発Gr
 - 管理者
 - 社員 4 人（熟練者 2 人）
 - 外注 15 人

■ 開発フロー



1. 開発の現状（3）： 要求開発

- 複数の顧客（車両メーカー）への対応
 - 顧客毎に異なる要求へ対応



リリース先顧客のSW要求に合わせて仕様を選択する

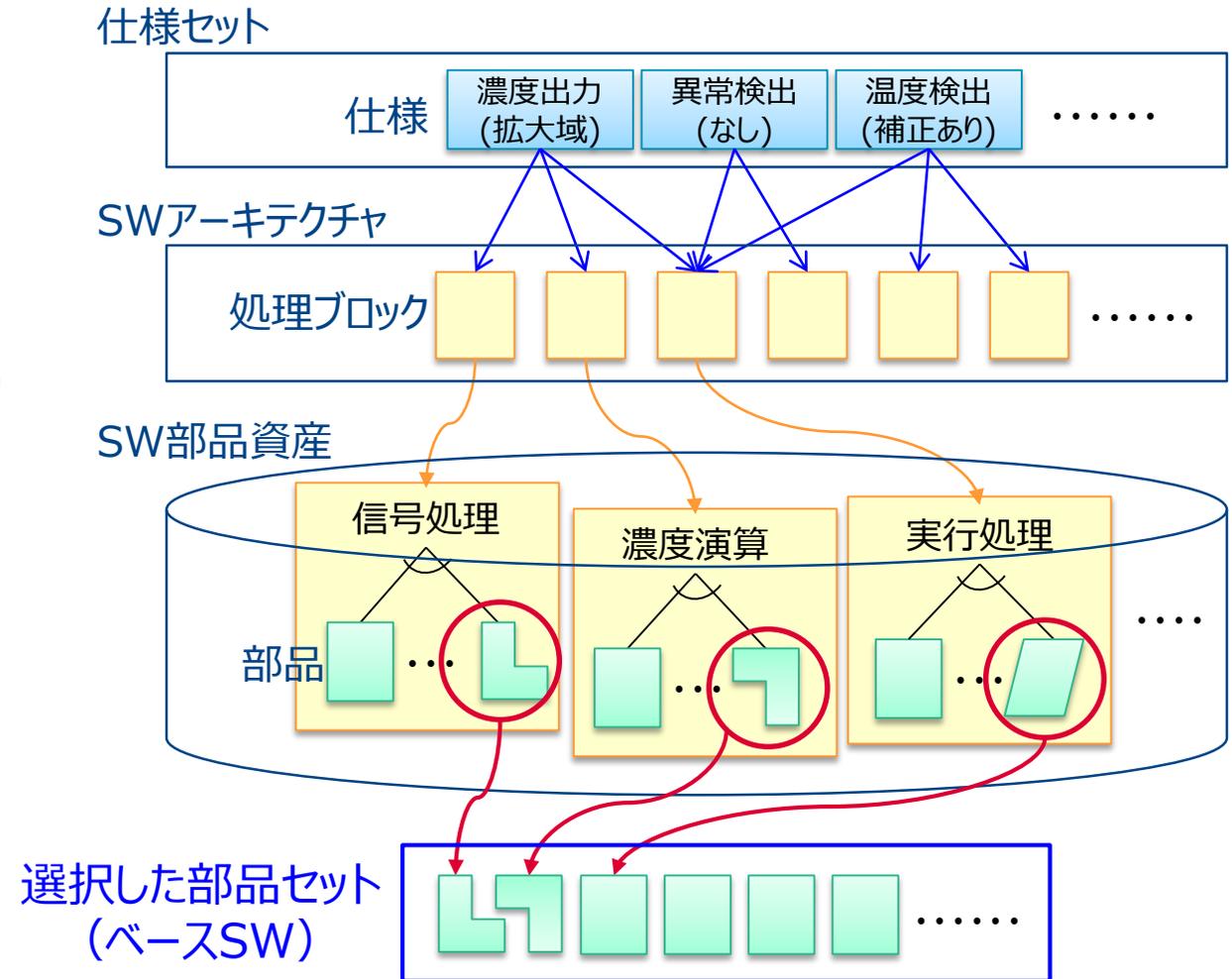
1. 開発の現状（4）：基本設計 -SW部品を選択-

■ SW部品の選択手順

1. 仕様セットの各仕様を複数の処理ブロックで実現
2. 各処理ブロックにSW部品資産から仕様を満たす部品を1つ選択

■ 選択した部品セット

- 開発開始時のベースSW
- 必要に応じて追加変更



仕様を充足するSW部品を選択しベースSWとする

2. 課題（1）： SW部品選択の問題

■ 部品の選択結果

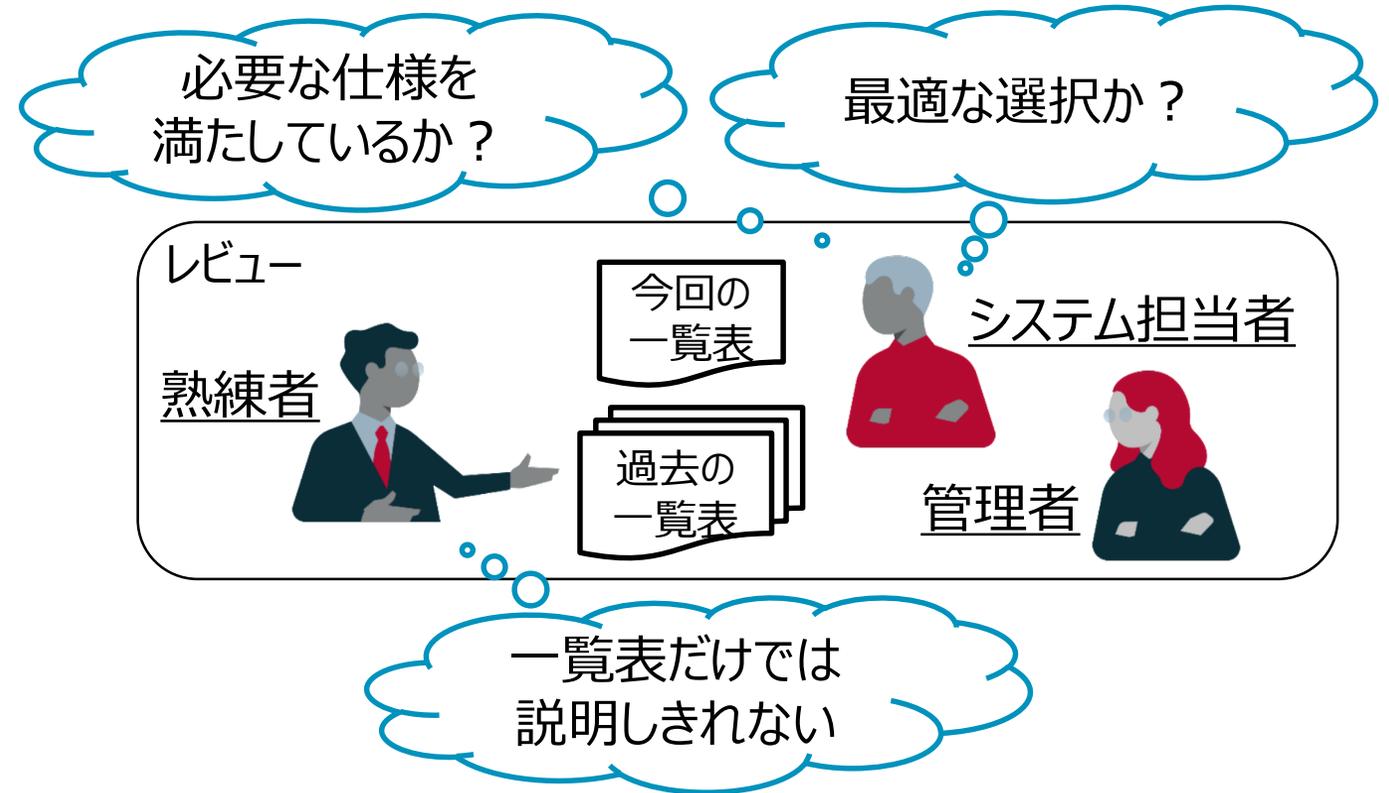
- 熟練者が部品を選択
- 選択結果：一覧表

<SW部品一覧表>

処理ブロック	SW部品	変更区分
信号処理	ADC_001	なし
濃度演算	GAS_002	あり
実行処理	TCM_002	なし
異常判定	DAG_003	あり
通信	COM_004	なし
：	：	：

■ レビューでの問題

- 説明しても理解してもらえない



部品の組合せの妥当性確認が困難なこと

2. 課題（2）： 妥当性確認ができない理由と課題

■ SW部品一覧表の問題点

<SW部品一覧表>

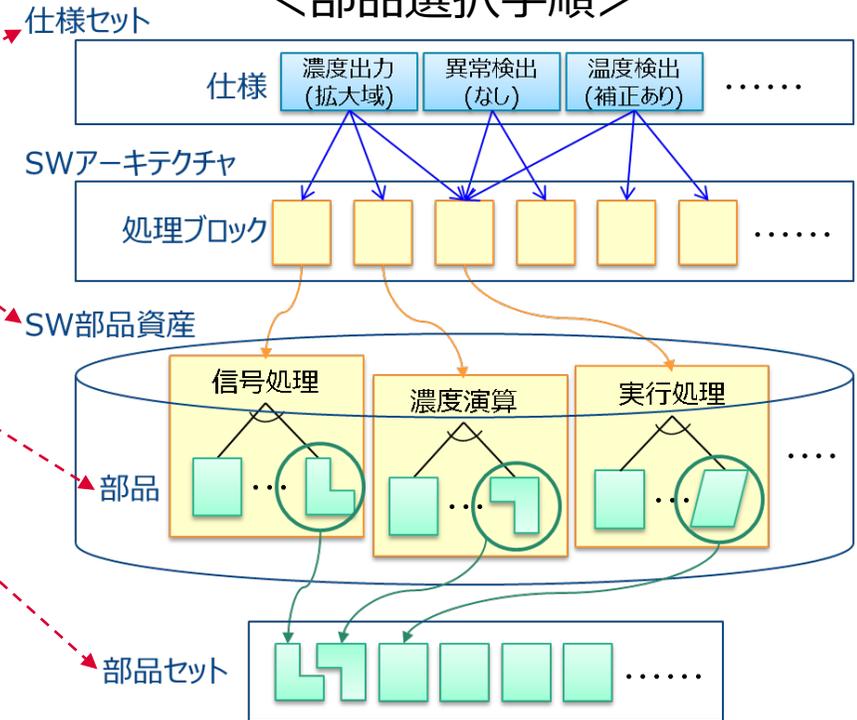
処理ブロック	SW部品	変更区分
信号処理	ADC_001	なし
濃度演算	GAS_002	あり
実行処理	TCM_002	なし
異常判定	DAG_003	あり
通信	COM_004	なし
⋮	⋮	⋮



<部品選択理由>

- ①仕様を満たすこと
- ②最新版であること
- ③同時選択可能なこと

<部品選択手順>



組合せ結果のみの一覧表では
部品の選択理由を読み取れない

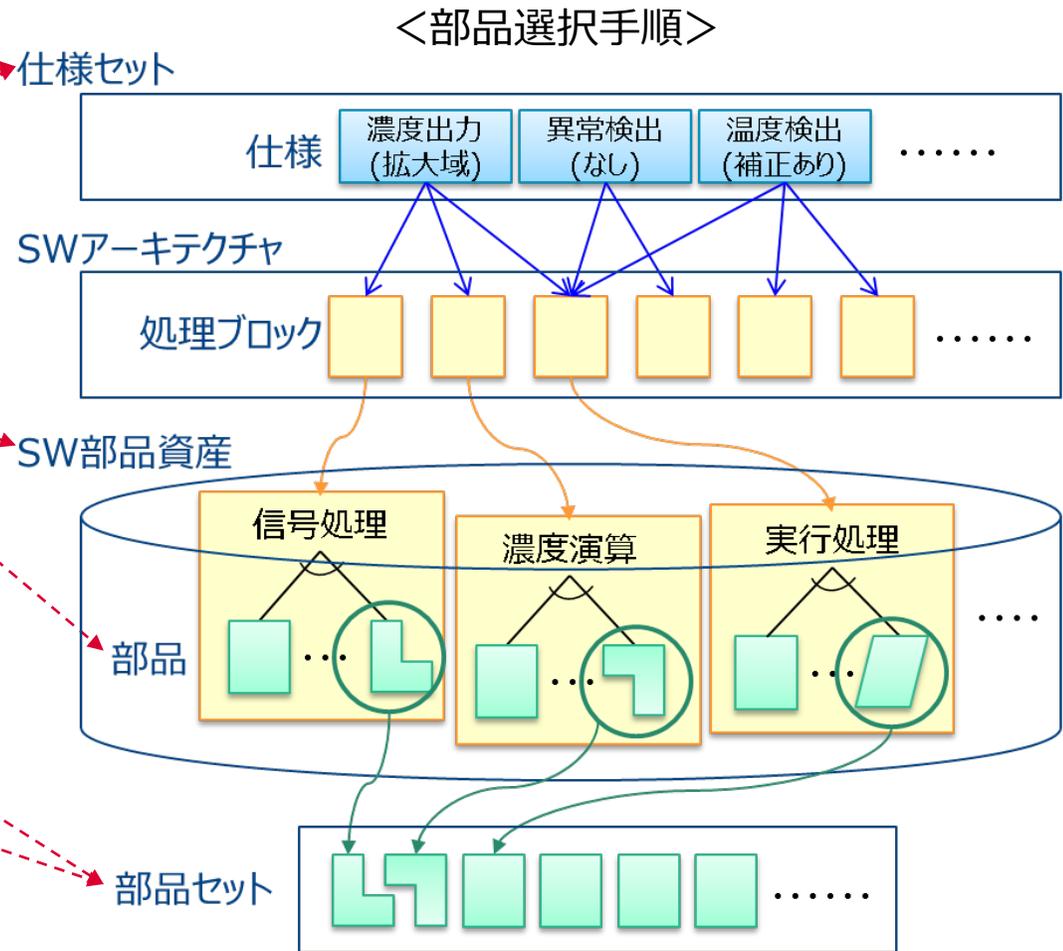
課題： 選択理由を示す情報をレビューで提示できるようにする

3. アプローチ (1)

■ SW部品の選択理由を示す情報

1. 仕様セット
2. 仕様と部品の対応
3. 部品バリエーションとバージョン
4. 同時選択の制約
5. 部品セット

各情報をつなげて示す必要がある



表形式では表現できない関係を示す必要がある

3. アプローチ（2）

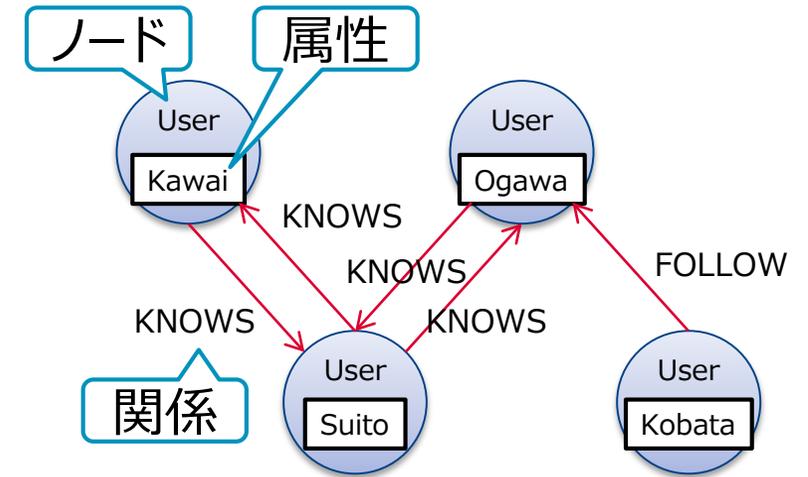
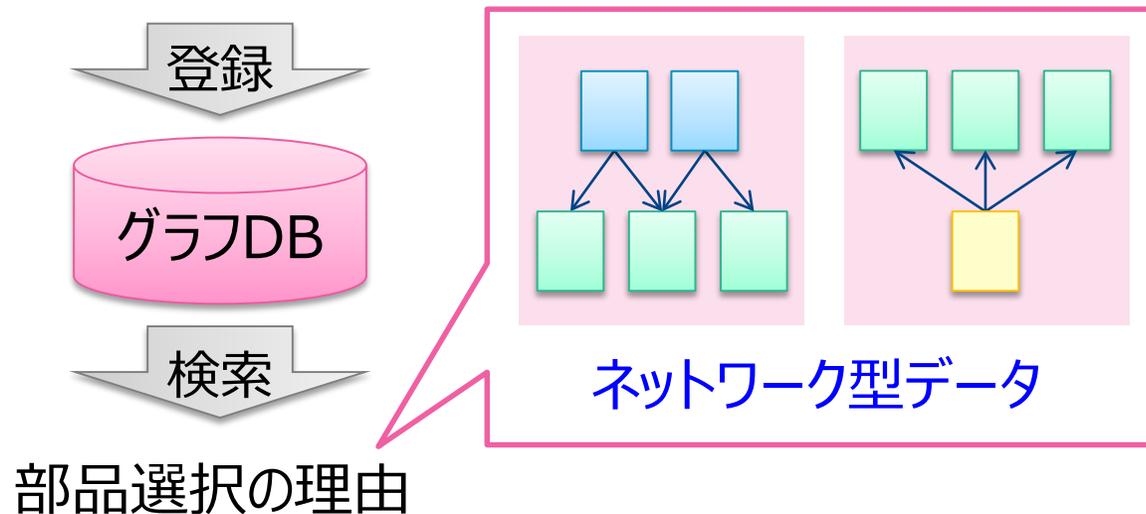
■ 解決策

- グラフDBを活用
- 仕様とSW部品の複雑な関係をDB化

■ グラフDB

- ネットワーク型のデータ構造を扱うDB
- つながりを持つ関係の扱いに優れる

仕様や部品の情報

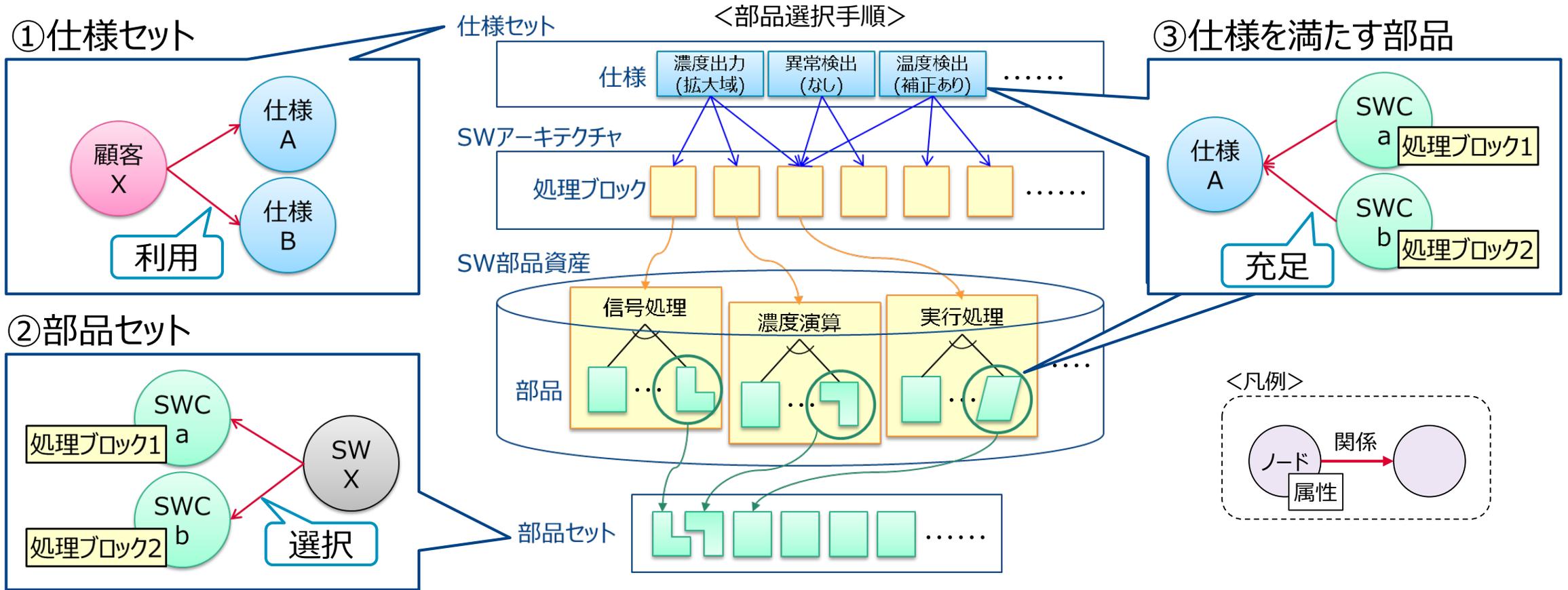


グラフDBで部品選択の理由を示す情報を可視化する

4. 実施方法 (1)

開発データの一部でグラフDBを構築し、仕様セットと部品セットの対応を確認する

■ Step1 ノード、関係、属性を定義



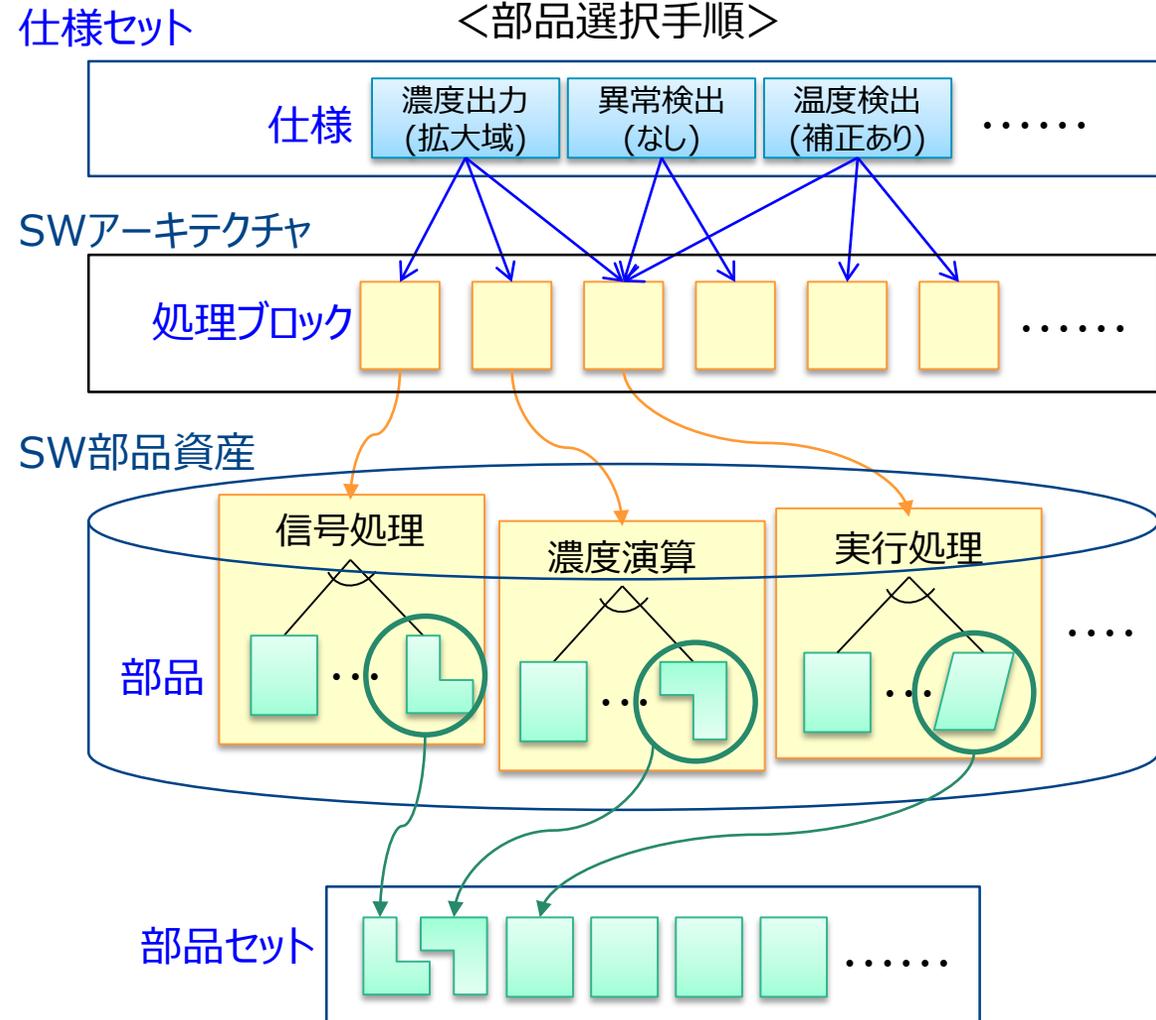
4. 実施方法（2）

■ Step2 グラフDBの作成

- 実施環境 … Neo4j
- 実施条件
 - 仕様セット：3種類
 - 仕様：7種類
 - 処理ブロック：5種類
 - 部品：10種類
 - 部品セット：3種類

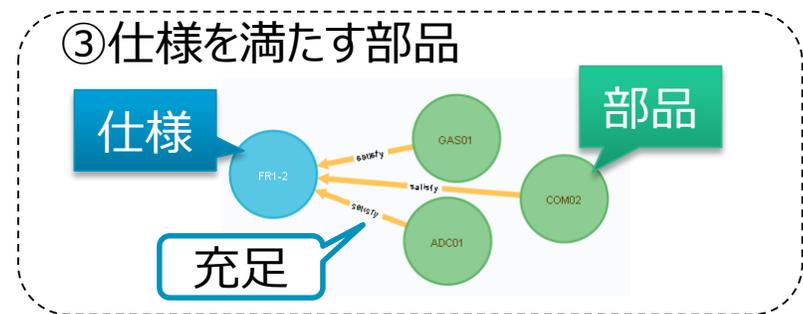
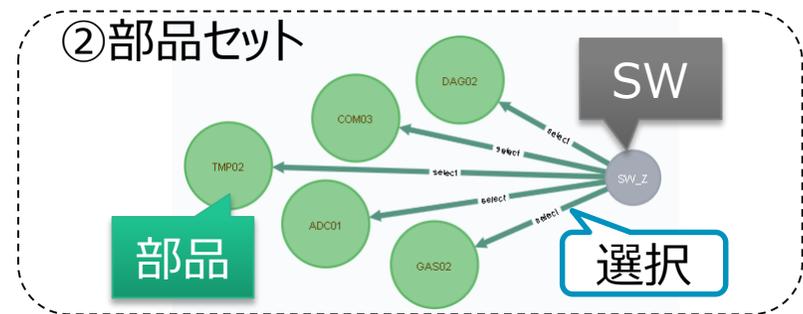
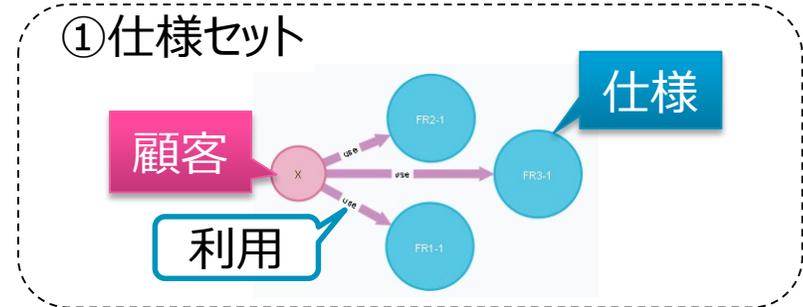
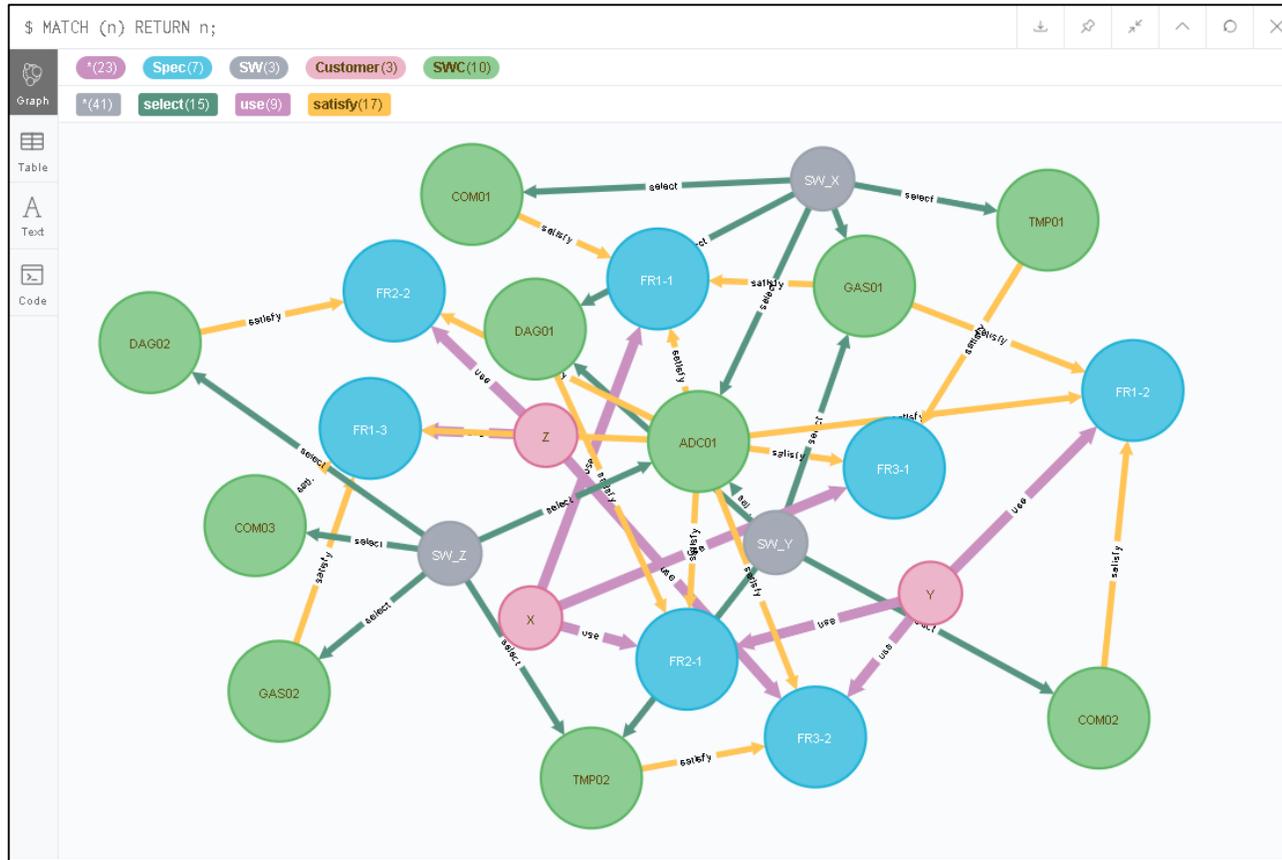
■ Step3 必要な情報を検索

- 出力結果の確認



5. 実施結果（1）： グラフDBの作成結果

■ 実施結果 1： 全ての情報の関係図



SW部品の関係をネットワーク型データ構造で表現できた

5. 実施結果（2）：有効性の確認結果

■ 実施結果 2

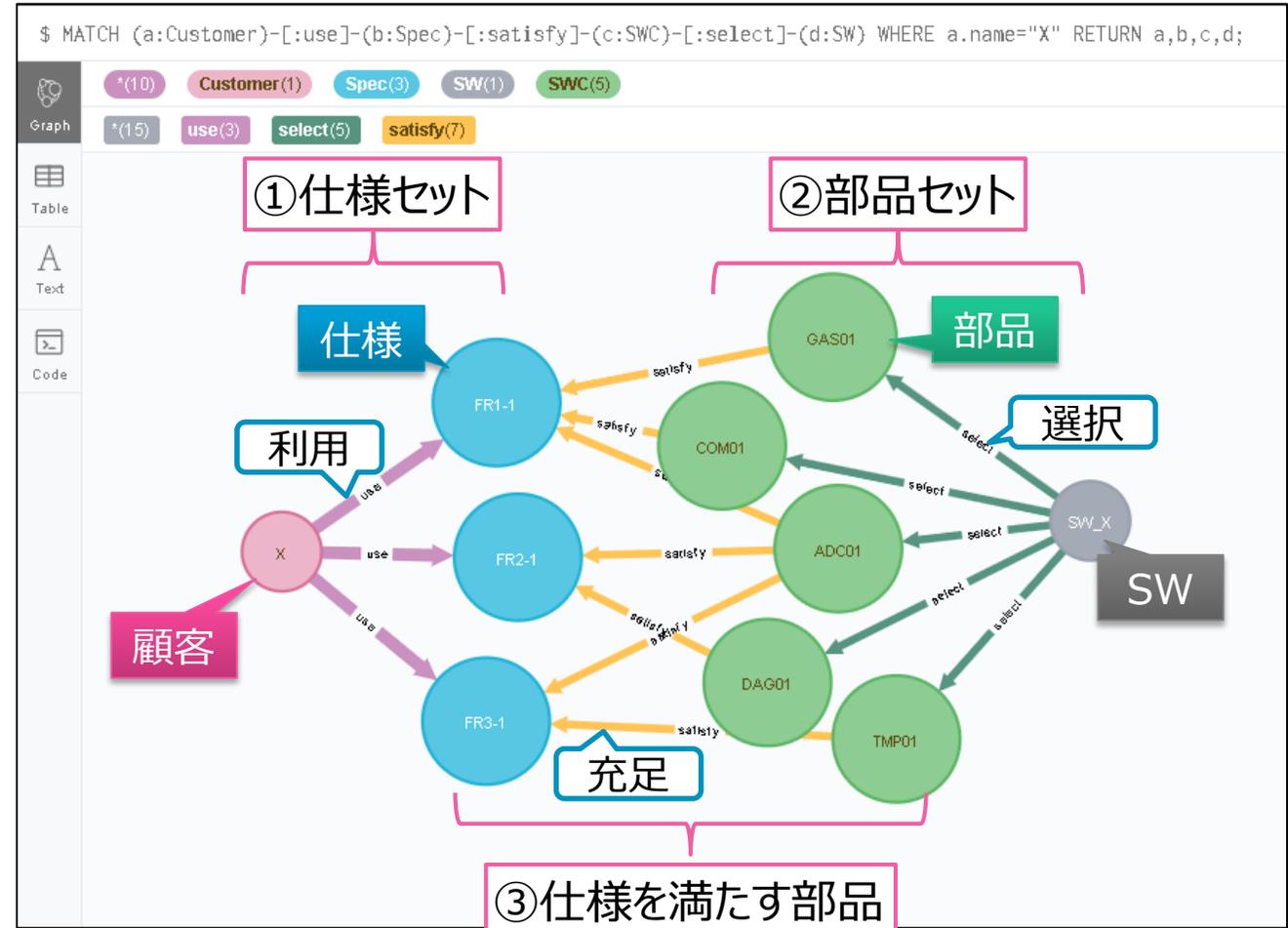
- 顧客X向けSWの関係図

以下を検索、つなげて表示

- ① 仕様セット
- ② 部品セット
- ③ 仕様を満たす部品



全てつながっていること
⇒ SW部品の組合せが妥当



選択した部品セットが対象の仕様セットを満たすことを確認できる

6. 考察（1）

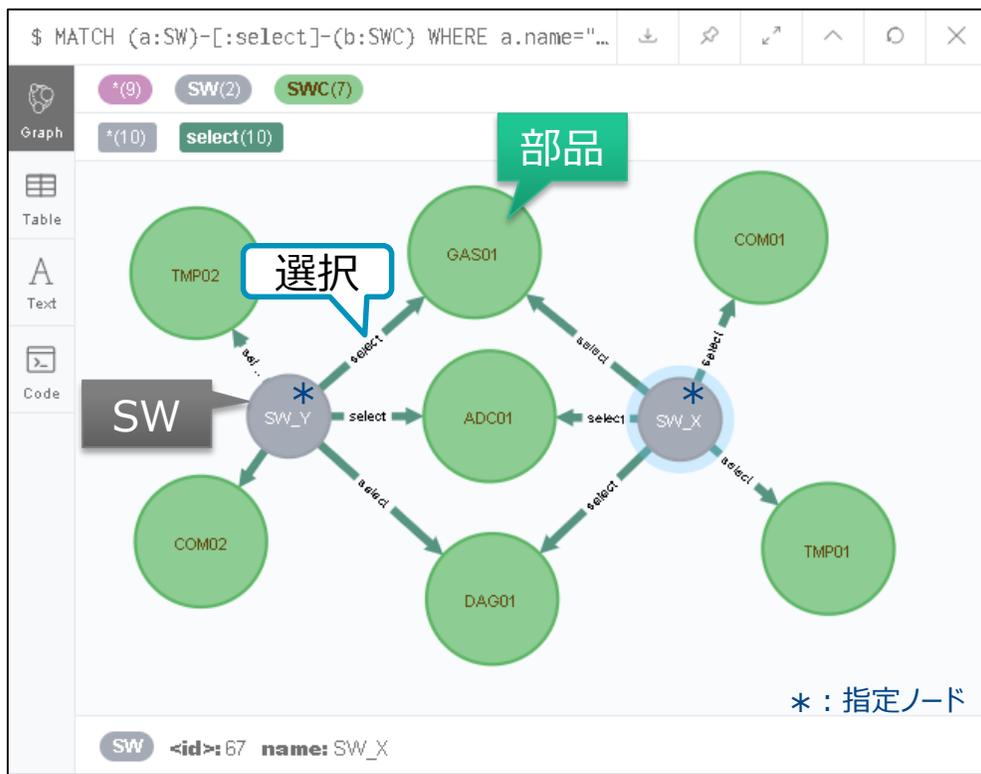
■ 実施結果の考察

- SW部品の選択理由を示す情報を可視化し、組合せの妥当性確認ができた
- ネットワーク型のデータ構造とすることで直感的にわかりやすく表現できる
- ブラウザ上で対話的に自由に視点を切り替えられることも利点である

6. 考察（2）：他の開発プロセスへの活用②

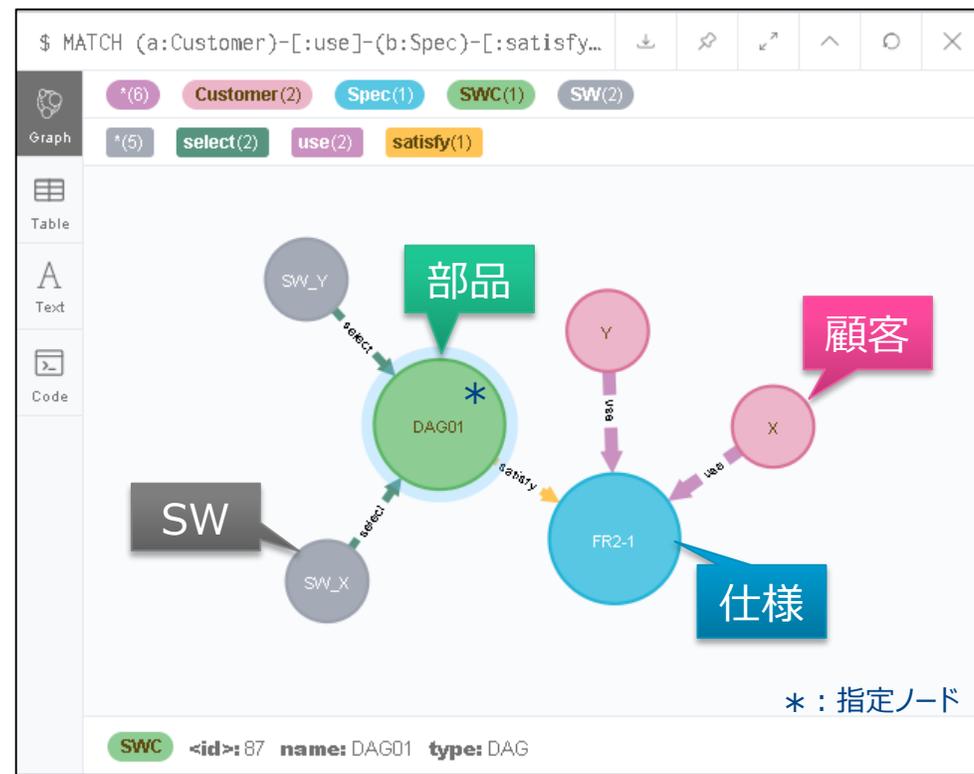
■ 部品セットの比較

- 複数のSWを指定し選択部品を検索



■ SWバグの波及先調査

- 指定した部品と関係があるノードを検索



差分分析、不具合調査時にも有効活用できる

7. まとめ

- 現状の問題
 - SW部品の組合せの妥当性確認が困難である
- 課題
 - SW部品の選択理由を示す情報を提示できるようにする
- アプローチ
 - グラフDBを適応し、検索によりネットワーク型のデータ構造で可視化する
- 結果
 - 選択した部品セットが対象の仕様セットを満たすことを確認できた
- 考察
 - SW部品の組合せの妥当性確認において有効である
 - それ以外の開発プロセスにおいても活用できる

DENSO

Crafting the Core