UMTPモデリング技術部会 2013年3月15日
モデリング技術セミナー『匠のモデリング技術の伝承と普及』

EPSON
EXCEED YOUR VISION

セイコーエプソンの モデリングトレーニング

～センスに依存しないモデリングを実現する～

セイコーエプソン株式会社

機器IT推進本部 機器ソフトウェア品質・生産技術部
ソフトウェア生産技術トレーニンググループ

萩原 豊隆

自己紹介

EPSON
EXCEED YOUR VISION

担当業務:ソフトウェア工学の導入推進

- 名前: 萩原 豊隆
- 所属部門
  - IT推進本部 機器ソフトウェア品質・生産技術部
  - ソフトウェア生産技術トレーニンググループ
- 経歴
  - IEEE1394 WDMドライバ開発
  - 業務用小型プリンタのアーキテクチャ設計
- 社外活動
  - 一般社団法人 電子情報技術産業協会(JEITA)
    情報システム・ディスラプティブ技術調査委員会
    ソフトウェアエンジニアリング技術専門委員会 委員

モデリングトレーニング

EPSON
EXCEED YOUR VISION

センスに依存しないモデリングを実現する

1. 教えられないという問題
2. 明確な手順と規準を用意する
3. モデリングの具体論
4. 成果と課題
5. まとめ

Ⅰ．教えられないという問題

Ⅰ．教えられないという問題

EPSON
EXCEED YOUR VISION

センスに依存する部分が多いと教えられない

➔ “できる人”なら“できる”では業務で使えない

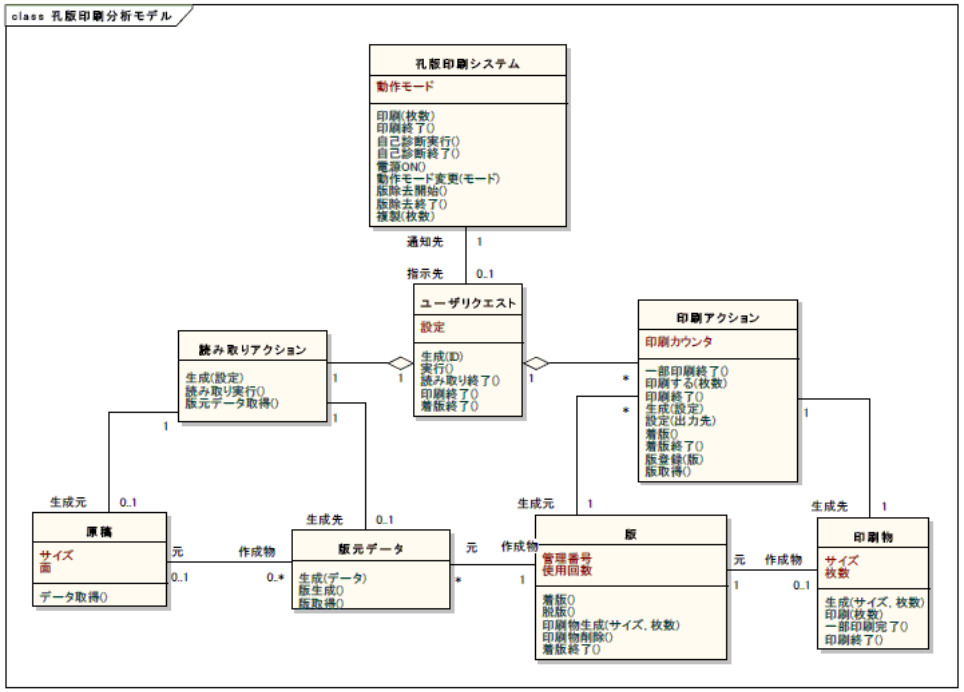
1. 手順を言語化できない
2. 良いと悪いとを判断できない
3. 真似して習得ではアートと同じ

1. 手順を言語化できない

EPSON
EXCEED YOUR VISION

「できるけど説明できない」では展開できない

➔ 手順を言語化して、誰でも実践可能にする



お手本を見ながらのスキル向上は良い方法！

しかし習得スキルは属人的な技能となり展開できない

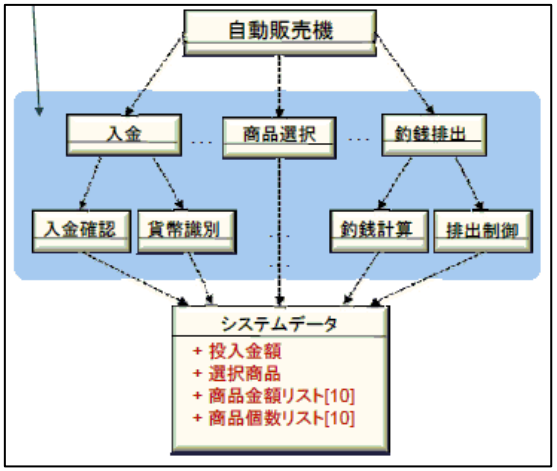
モデルの出典:「組み込み分野のためのUMLモデルカタログ」2010年10月1日 UMTP, Japan

2. 良いと悪いとを判断できない

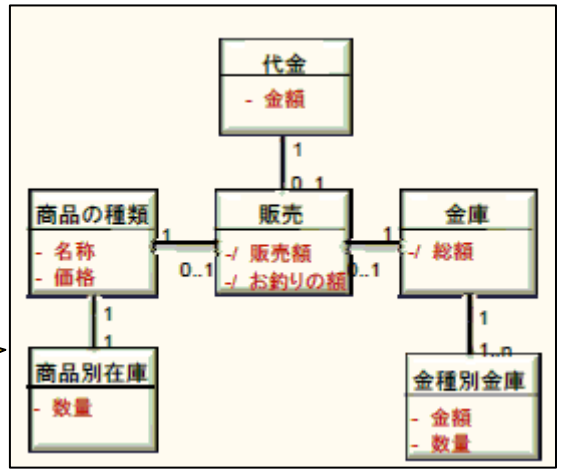
EPSON
EXCEED YOUR VISION

良し悪しをセンスの有無にしてはいけない

➔ 規準が無いとセンスの問題になってしまう



これがダメで



これが良いは、
なぜなのか？

モデルの出典：UMTPModelingForum2010「組み込み分野のためのモデルカタログ」講演資料
組み込み分科会 副主査 (株)エクスモーション 芳村美紀 氏

3. 真似して習得ではアートと同じ

EPSON
EXCEED YOUR VISION

技術(Art)から工学技術(Engineering)へ移行する

➔ 科学的な理論に基き、誰でも実践できること

必要なこと

1. 解くべき問題を捉える方法
   - 解くべき問題＝情報処理の目的を的確に捉える
   - 目的を俯瞰することで、整理し本質化する
2. 問題からソフトウェア構造を作る方法
   - 情報処理の目的を反映する構造を作る
   - 目的を反映するだけでなく、良い構造にする

保守性と移植性が高い

4. まとめ

EPSON
EXCEED YOUR VISION

センスに依存する部分が多いと教えられない

ポイント

1. 「できるけど説明できない」では展開できない
2. 良し悪しをセンスの有無にしてはいけない
3. 技術(Art)から工学技術(Engineering)へ移行する

Ⅱ．明確な手順と規準を用意する

Ⅰ．教えられないという問題

Ⅱ．明確な手順と規準を用意する

Ⅲ．モデリングの具体論

Ⅳ．成果と課題

Ⅴ．まとめ

Ⅱ．明確な手順と規準を用意する

EPSON
EXCEED YOUR VISION

誰でもできる技法を用意してトレーニングする

➔ 技法(method)＝手順(procedure)＋規準(criteria) ※

1. 誰でもできる手順
2. 妥当性を確認する規準
3. 変換の正しさを検証する規準

※定義の出典：「ずっと受けたかったソフトウェア設計の授業」（飯泉純子、大槻繁 共著, 2011年）

1. 誰でもできる手順

EPSON
EXCEED YOUR VISION

BeforeからAfterへの変換手順を用意する

➔ 冗長であっても、一歩ずつ進める手順とする

特徴

1. 変換手順である

   正しいものを正しく変換すれば最後まで正しい

   ➔ トップダウン的に進める手順は分かりやすい

   構造化分析と同じ

2. 途中で間違いを修正できる

   とは言え、間違いに気づき修正できることは重要！

   ➔ 早期に修正しながらボトムアップ的に作り上げる

実際に役に立つ設計手法にトップダウンのものはない

※株式会社 一
大槻繁氏の言葉

1.1 分かりやすい規準も設定する

EPSON
EXCEED YOUR VISION

正しいものを正しく作っているかを判断する

➔ 間違いにすぐに気づくことが重要

気づいたら、その場で
すぐに直す

規準の種類

1. 妥当性確認(validation)の規準

   正しいことを確認する

   ➔ 作るものは目的を満たしているか？

2. 検証(verification)の規準

   正しく行っていることを検証する

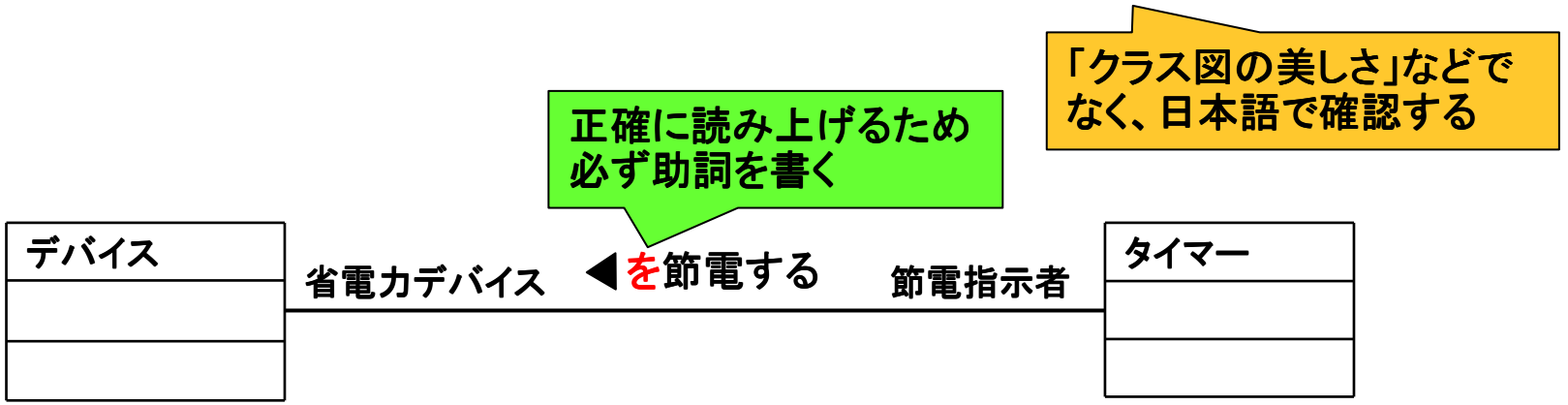
   ➔ 途中で間違いが混入していないか？

2. 妥当性を確認する規準

EPSON
EXCEED YOUR VISION

読み上げたとき機能が目的を満たしている

➔ 読み上げると、機能を表現する文章になる



手順

読み上げて、クラス名、関連名、ロール名の正しさを確認する

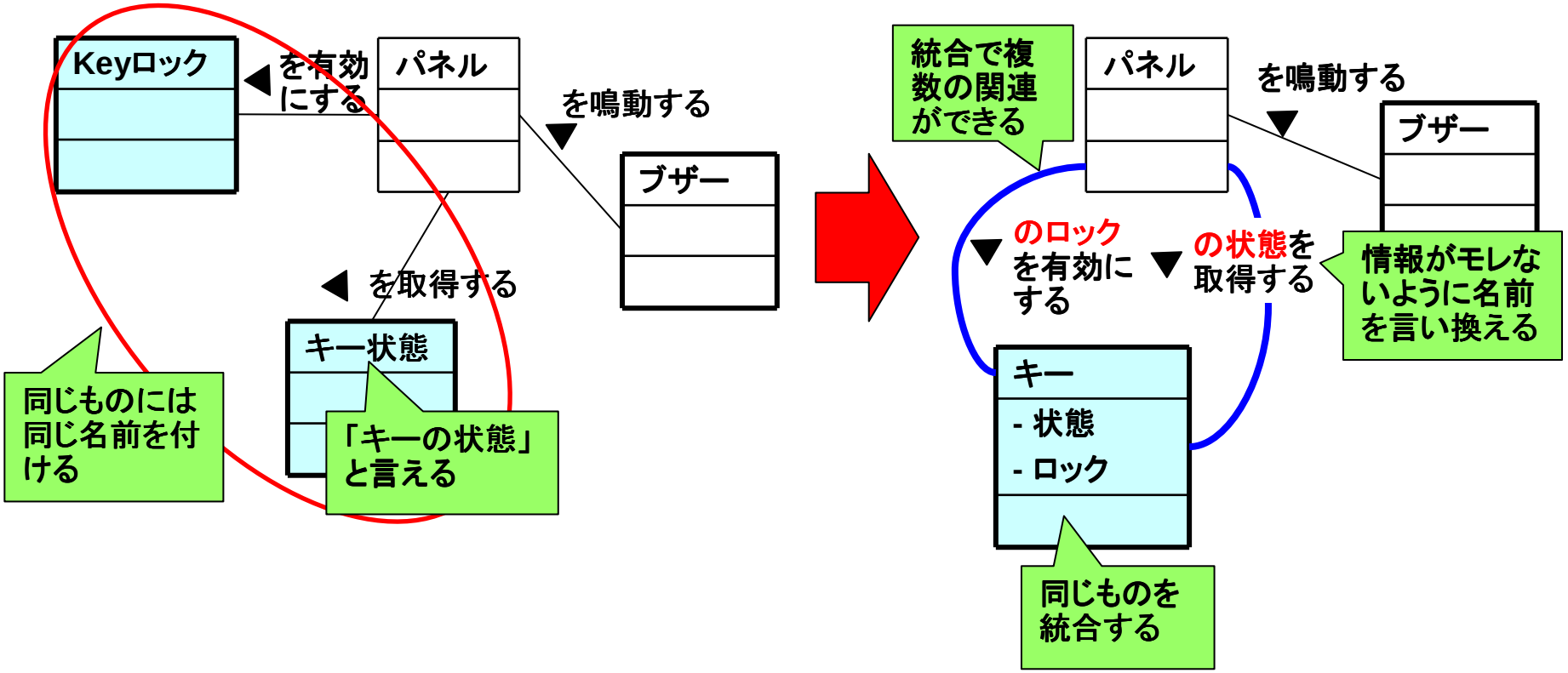
- 「タイマーは、デバイス を節電する」と読む。
- 「タイマーは、節電指示者の役割で、デバイス を節電する」
- 「タイマーが節電するとき、デバイスは省電力デバイスの役割になる」

3. 変換の正しさを検証する規準

EPSON
EXCEED YOUR VISION

変換前後で読み上げ結果が一致している

➔ 読み上げ結果が同じなら変換に間違いはない



4. まとめ

EPSON
EXCEED YOUR VISION

誰でもできる技法を用意してトレーニングする

ポイント

1. BeforeからAfterへの変換手順を用意する
2. 読み上げたとき機能が目的を満たしている
3. 変換の前後で読み上げ結果が一致している

Ⅲ. モデリングの具体論

Ⅰ. 教えられないという問題

Ⅱ. 明確な手順と規準を用意する

Ⅲ. モデリングの具体論

Ⅳ. 成果と課題

Ⅴ. まとめ

Ⅲ. モデリングの具体論

EPSON
EXCEED YOUR VISION

機能一覧をクラス図へ変換する手順を作った

➔ 要求を機能（目的語＋動詞）として捉える

1. 機能を定義する
2. 機能をクラス図に対応付ける
3. クラスの粒度をそろえる
4. 空のクラスを削除する
5. 本質的な関連に整理する

1. 機能を定義する

EPSON
EXCEED YOUR VISION

目的語と動詞で、目的となる機能を捉える

情報処理の

➔ 機能とは、目的語＋動詞である

機能の表現 ※

＜定義の対象＞	＜目的語＞	＜動詞＞	＜制約条件＞
腕時計は	時刻	を示す	±10秒／月差
機能の割付対象	機能（言葉のモデル）		

記述のポイント

- 定義の対象を一般化して捉えないで、対象特有のはたらきを定義する

「を制御する」ではダメ

※出典：「新・VEの基本 価値分析の考え方とプロセス」（土屋裕 監修、産能大学VE研究グループ 著、1998年）を元に作成

1.1 機能一覧を作る

EPSON
EXCEED YOUR VISION

クラス図に対応づける機能を一覧表にしておく

➔ 構造に反映する機能をすべて洗い出す

機能一覧の記述例

目的語	動詞	制約条件
ホームタイムの都市情報	を設定する	アジャストボタンで
時刻	をデジタル表示する	12/24時モードで表示方法を決める
時刻	をアナログ表示する	
アラーム	を鳴らす	アラーム時刻の場合
ストップウォッチ	を計測開始する	

Who(だれが)、What(何を)、
When(いつ)、Where(どこで)、
How mach(どの程度)
How to (どのように) など

※この機能一覧の記述例は、「ずっと受けたかったソフトウェア設計の授業」(飯泉紀子 、大槻繁 共著)の第5章「振舞い」の腕時計の例題を参考に作成した

1.2 機能を事前に整理する

EPSON
EXCEED YOUR VISION

USDMで整理しておくと機能一覧を作りやすい

➔ 要求に現れる目的語と動詞を一覧に反映する

USDMによる要求と要求仕様の整理例 ※ USDM : Universal Specification Describing Manner

要求	M01-02	複数のキーワードを組み合わせてメールを検索できる
	理由	可能性のあるキーワードで確実にメールを見つけた
□	M01-02-1	検索したいキーワードを入力できる
□	M01-02-2	複数のキーワードを「AND」と「OR」でつなぐことができる
□	M01-02-3	キーワードは最大８個まで指定できる
要求	M01-03	検索されたメールをリスト表示して、そこからメールを選択して表示する
	理由	該当するメールが複数あるときは内容を確認して絞り込みたい
□	M01-03-1	検索されたメールの「Subject」を一覧で見せる
□	M01-03-2	メールが10件を超えるときはスクロールバーを表示する
		・・・・

主に構造（クラス図）に反映される機能

主に関数内に反映される機能

※ 出典：「要求を仕様化する技術表現する技術 改定第2版」（清水吉男 著、2010年）の要求仕様の記述例

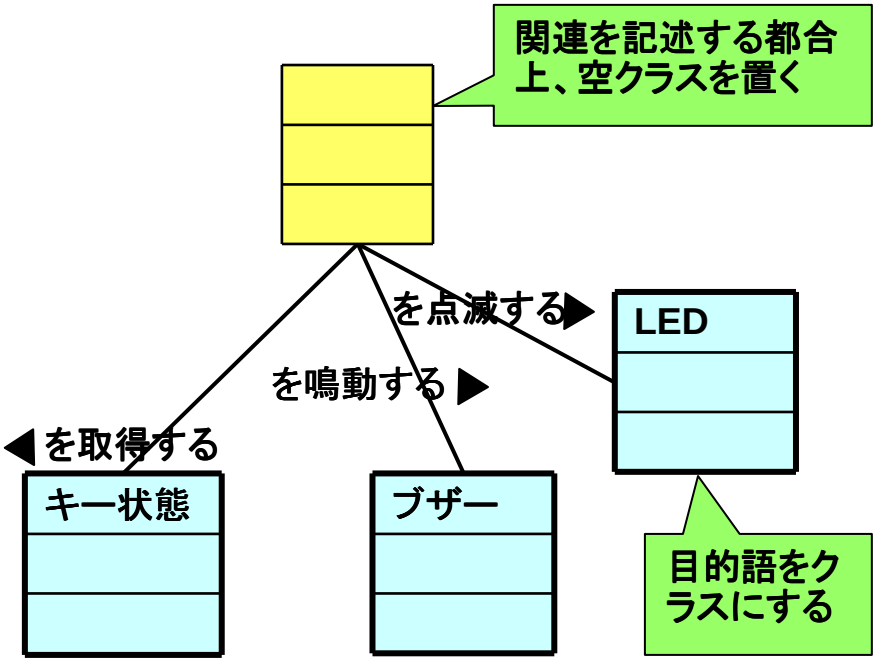
2. 機能をクラス図に対応づける

EPSON
EXCEED YOUR VISION

目的語はクラス、動詞は関連名にする

機能一覧表

目的語	動詞
キー状態	を取得する
ブザー	を鳴動する
LED	を点滅する



規準

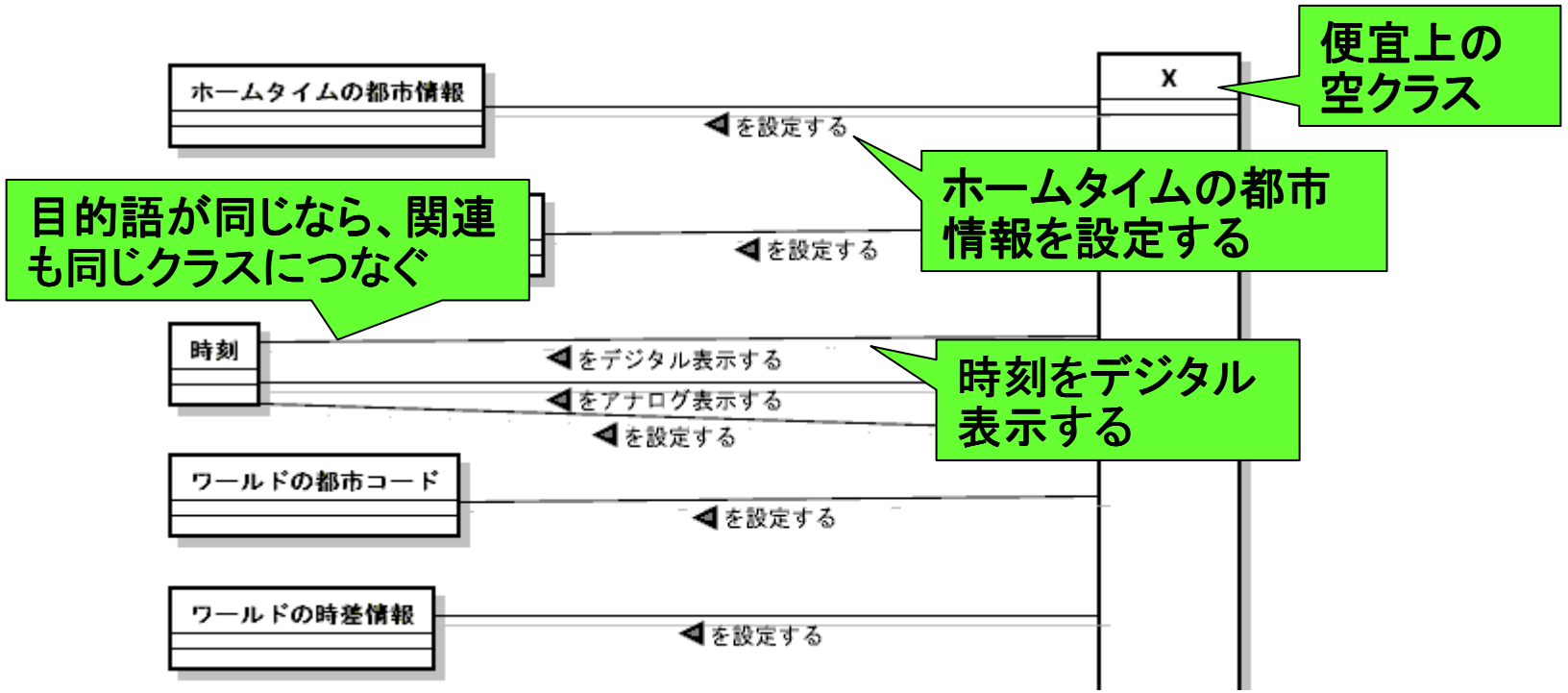
- 目的語をクラスにしている
- 助詞と動詞を関連にしている

2.1 機能一覧から作ったクラス図

EPSON
EXCEED YOUR VISION

機能一覧をクラス図に対応づける

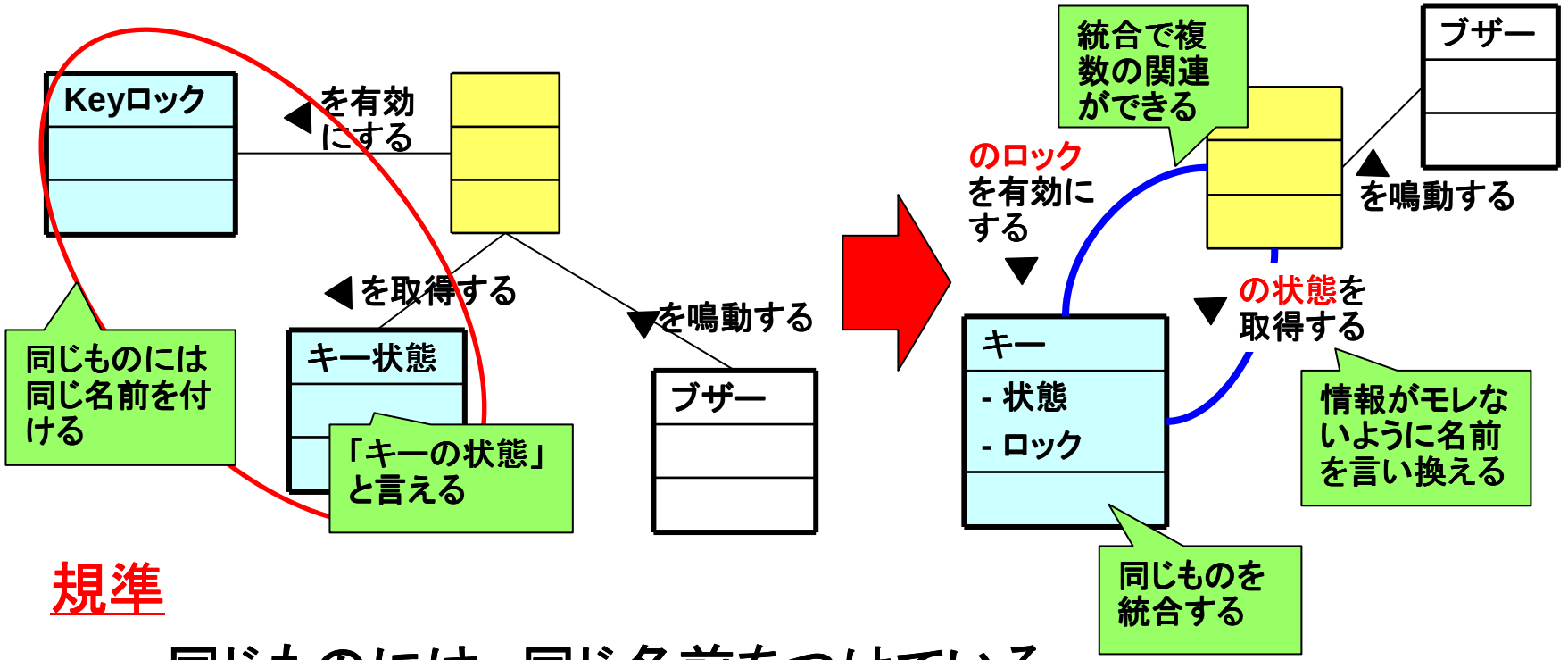
➔ クラスと関連を読み上げると機能一覧に戻る



3. クラスの粒度をそろえる

EPSON
EXCEED YOUR VISION

細かくなったクラスを統合する



規準

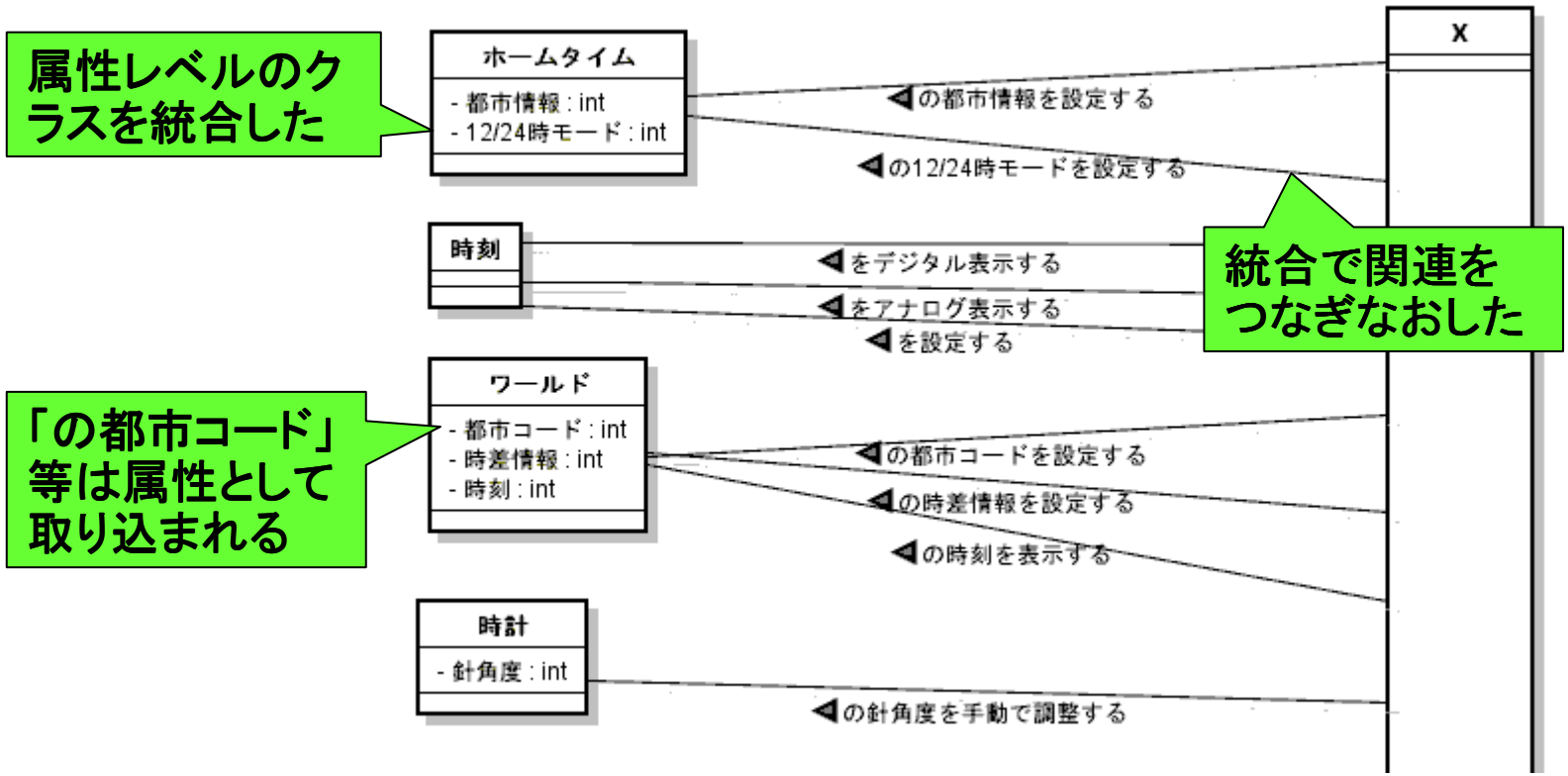
- 同じものには、同じ名前をつけている
- 属性レベルのクラスを統合している
- 統合したクラスの関連をつなぎ直している

3.1 粒度が適正化されたクラス図

EPSON
EXCEED YOUR VISION

クラスの粒度を適正化すると属性が現れる

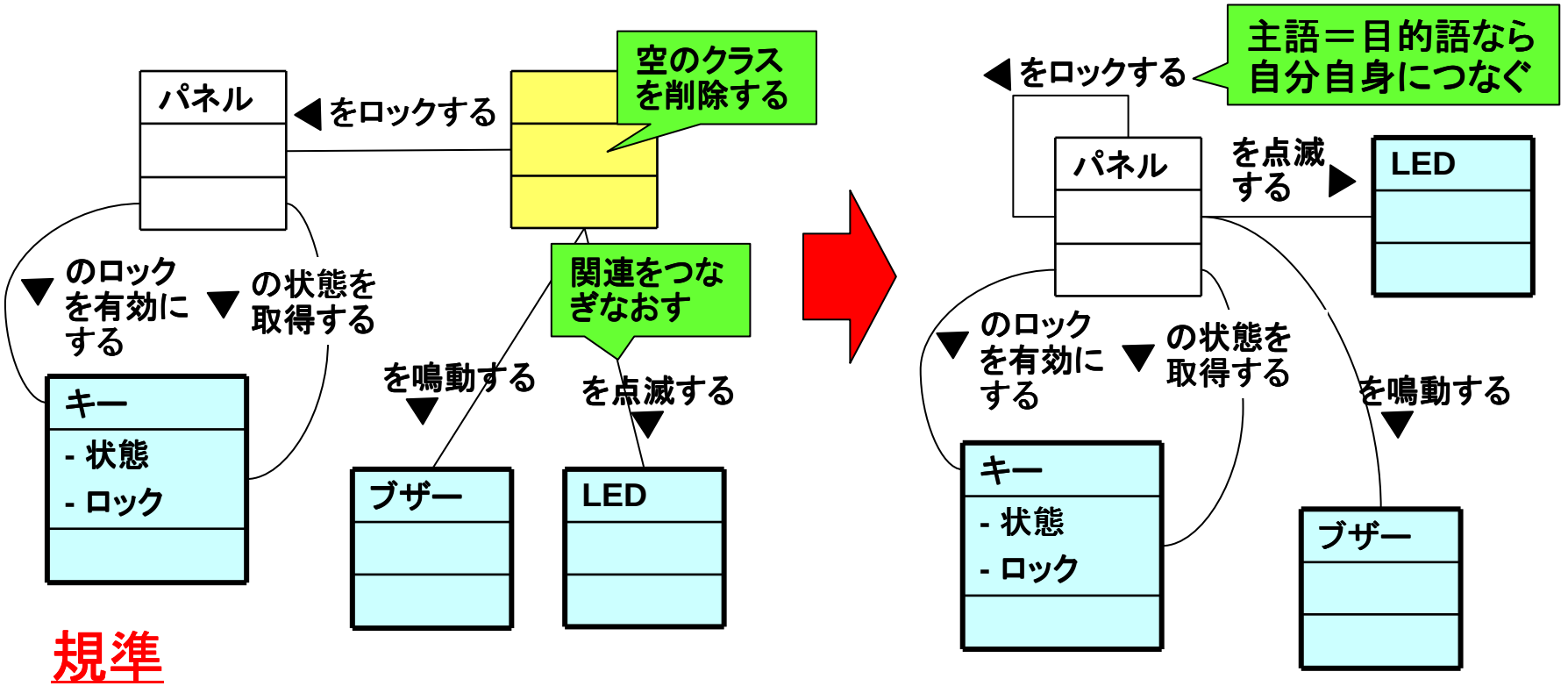
➔ 属性化で言葉の散らばりも、少し解消する



4. 空のクラスを削除する

EPSON
EXCEED YOUR VISION

関連をつなぎ直して、空クラスを削除する



規準

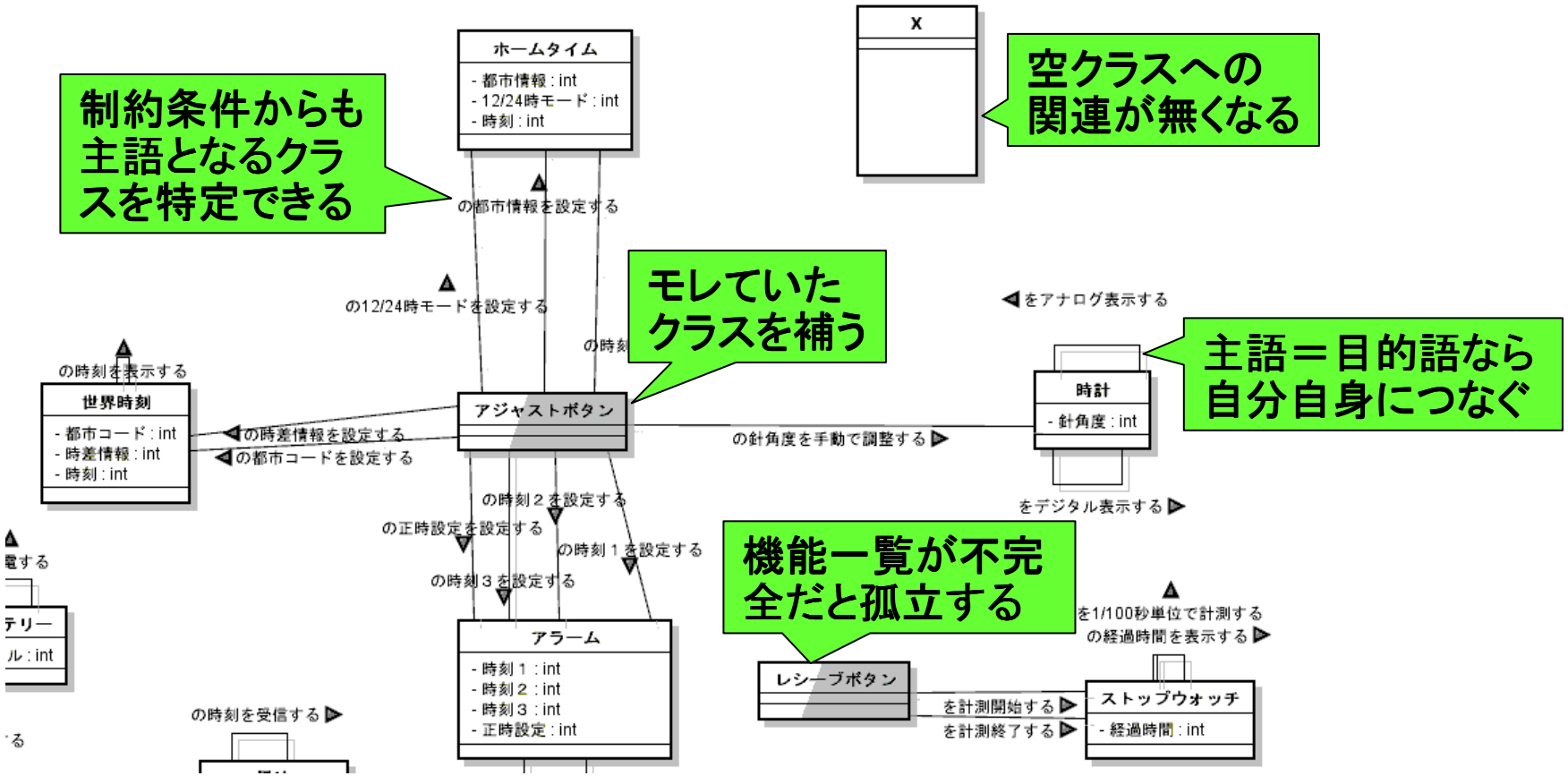
- 読み上げたとき主語となるクラスに関連をつなぎ直している
- 関連が無くなった空のクラスを削除している

4.1 空クラスを削除したクラス図

EPSON
EXCEED YOUR VISION

空クラスを削除すると、クラス図らしくなる

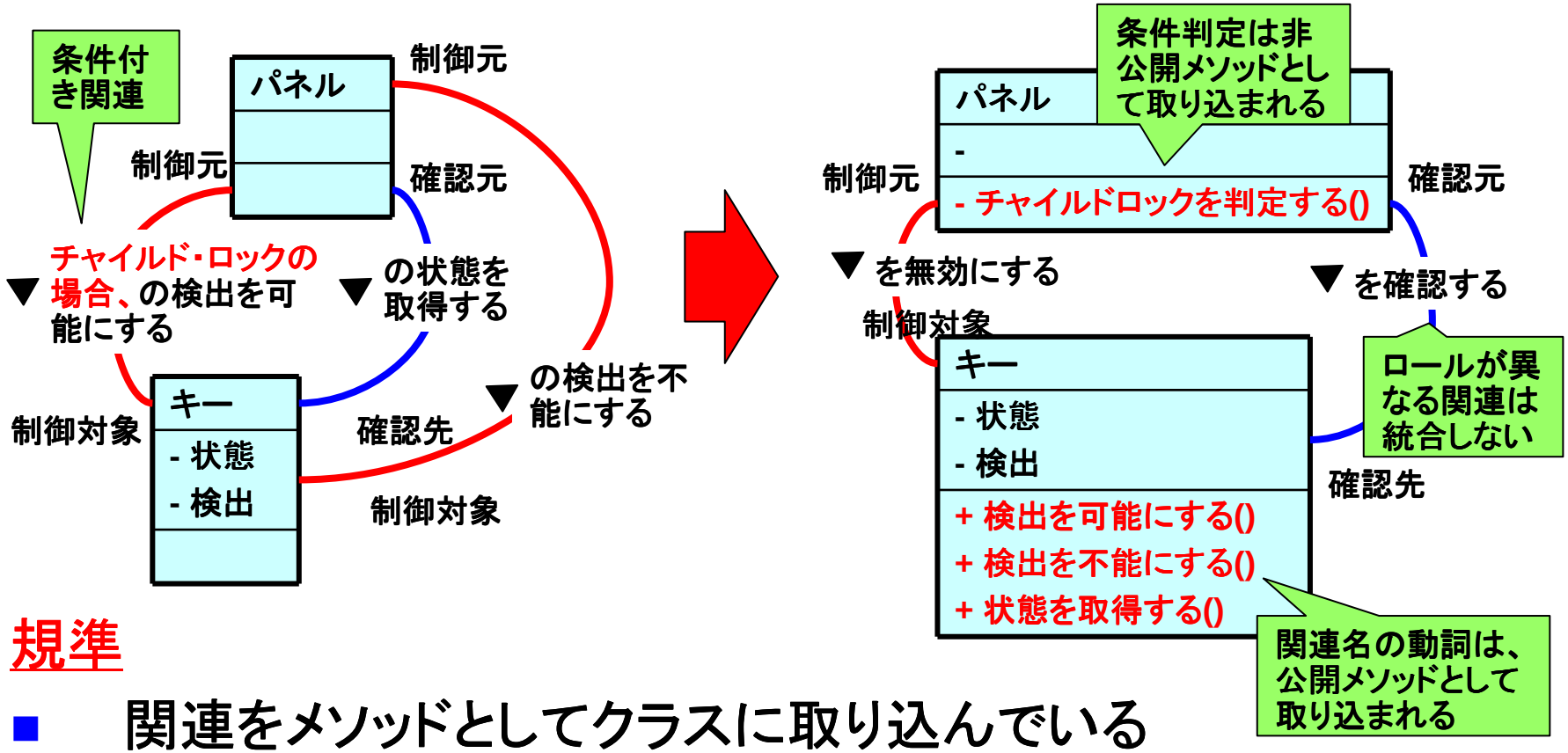
➔ つなぎなおすときに、モレが見つかる



5. 本質的な関連に整理する

EPSON
EXCEED YOUR VISION

細かい関連はメソッドとしてクラスに取り込む



規準

- 関連をメソッドとしてクラスに取り込んでいる
- 取り込んだ関連を本質的な関連に置き換えている

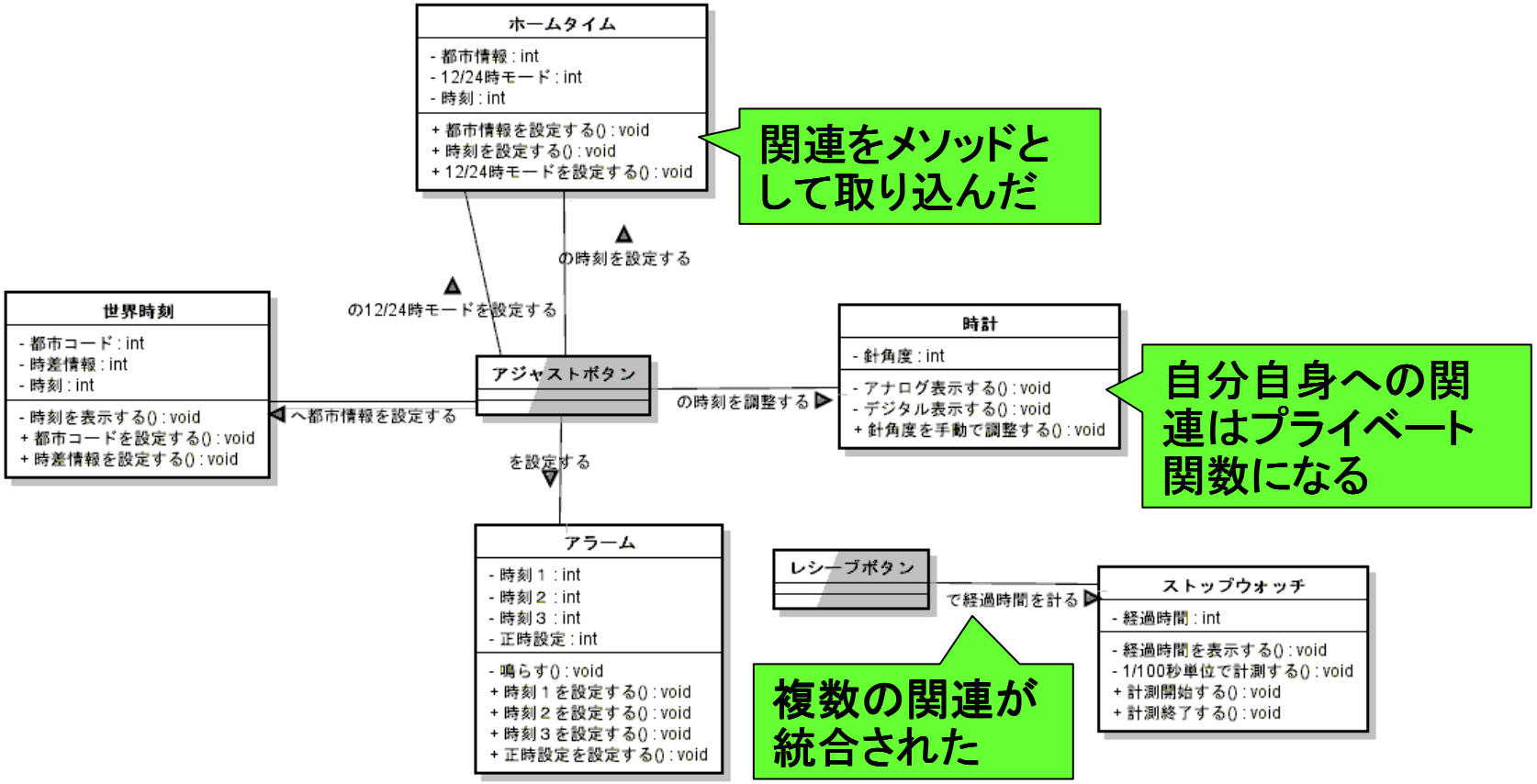
5.1 関連を整理したクラス図

EPSON
EXCEED YOUR VISION

関連を整理すると、クラス図として完成する

➔ あとは洗練させるだけ！

とりあえず



5.2 関連の統合

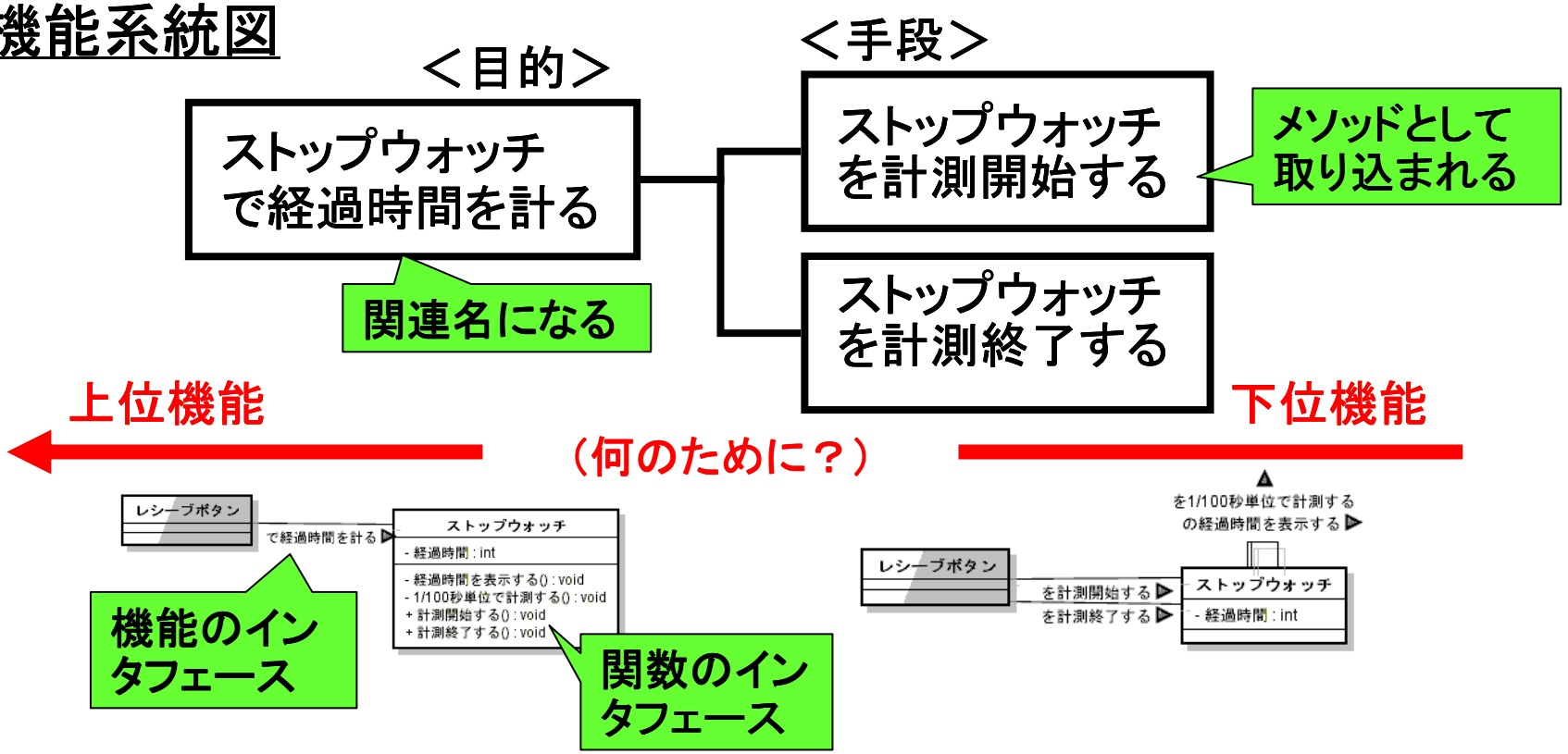
VEで活用する図の一種

EPSON
EXCEED YOUR VISION

関連の統合は、機能系統図で考える

➔ 「何のために？」と問い、上位機能を抽出する

機能系統図



6. まとめ

EPSON
EXCEED YOUR VISION

機能一覧をクラス図へ変換する手順を作った

ポイント

1. 目的語と動詞で、目的となる機能を捉える
2. 目的語はクラス、動詞は関連名にする
3. 細かくなったクラスを統合する
4. 関連をつなぎ直して、空クラスを削除する
5. 細かい関連はメソッドとしてクラスに取り込む

Ⅳ. 成果と課題

Ⅰ. 教えられないという問題

Ⅱ. 明確な手順と規準を用意する

Ⅲ. モデリングの具体論

Ⅳ. 成果と課題

Ⅴ. まとめ

Ⅳ. 成果と課題

EPSON
EXCEED YOUR VISION

満足度は高く、気づきのポイントも多い

➔ 適用事例も出始め定量的な効果も上がっている

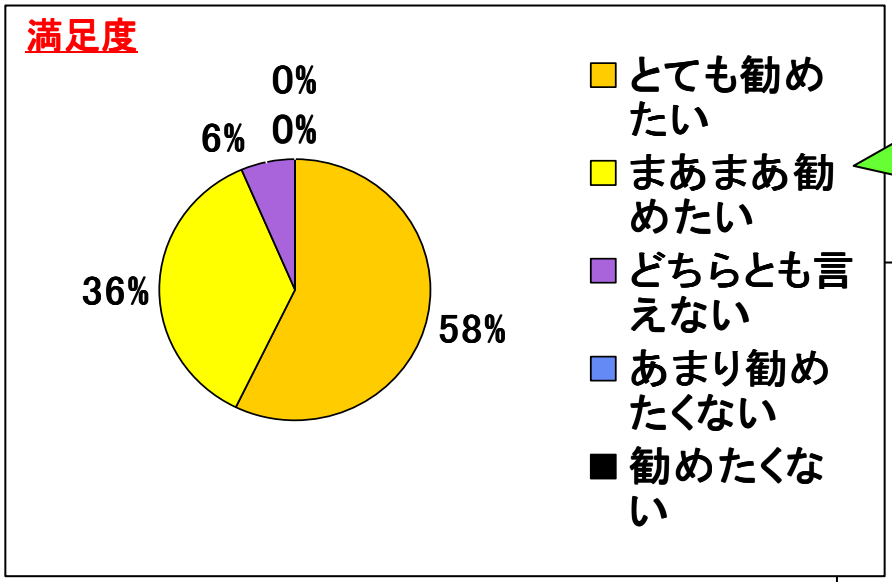
1. トレーニング実績
2. 事例1：C言語でのモデル駆動開発
3. 事例2：低リソース環境でのSPLE挑戦
4. 課題と、当面の対策

1. トレーニングの実績

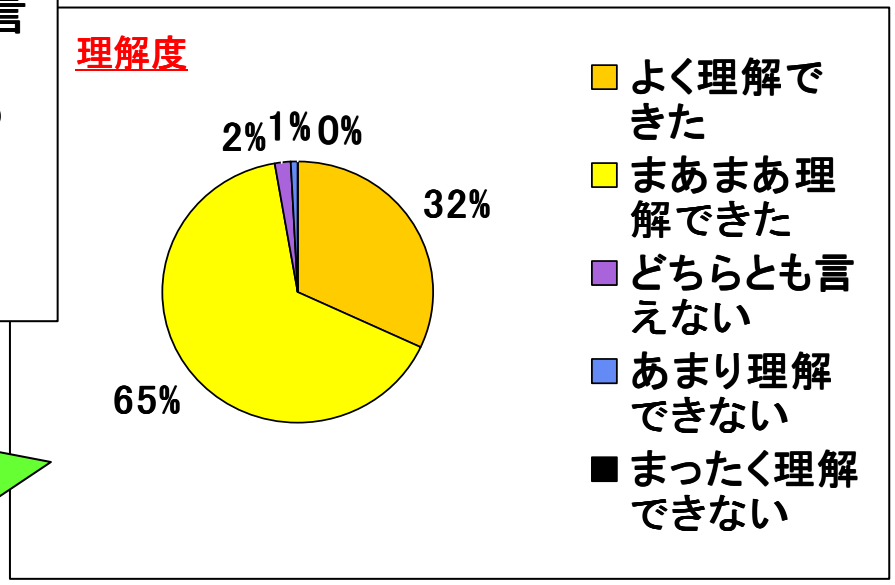
EPSON
EXCEED YOUR VISION

延べ200名以上が受講している

➔ トレーニングの満足度と理解度は高い



《質問》
本トレーニングの受講を他の人に勧めたいと思いますか？



《質問》
本トレーニングの内容は理解できましたか？

※調査期間：2011年3月～2012年7月、回答数：110名

1.1 手順と基準に関する感想

EPSON
EXCEED YOUR VISION

分かりやすい手順と規準が評価されている

➔ 勘と経験からの脱却に手ごたえを感じている

＜アンケートの感想＞

- 機械的にモデリングができることは、複数人でまわすプロジェクトをこなす上で大変重要なことだと思います。それをある程度実現できるというところはすごく有用性がある仕組みだと感じました。
- モデルとは経験と勘によるものが多いと思っていましたが、機械的にできる部分が多いことを実感しました。

1.2 要求仕様に関する気づき

EPSON
EXCEED YOUR VISION

要求仕様作成の動機付けにもなっている

➔ USDMとの連携にも手ごたえを感じている

＜アンケートの感想＞

- **USDMからの言葉の洗い出しをきっちりやっておくことでその後のクラス分析のやり易さにつながると感じた。**
- **しっかりとUSDMで記載された要求仕様書をまず用意する必要があり、そのための準備が前もって行ってあれば尚効率的に設計できたと感じました。**

2. C言語でのモデル駆動開発

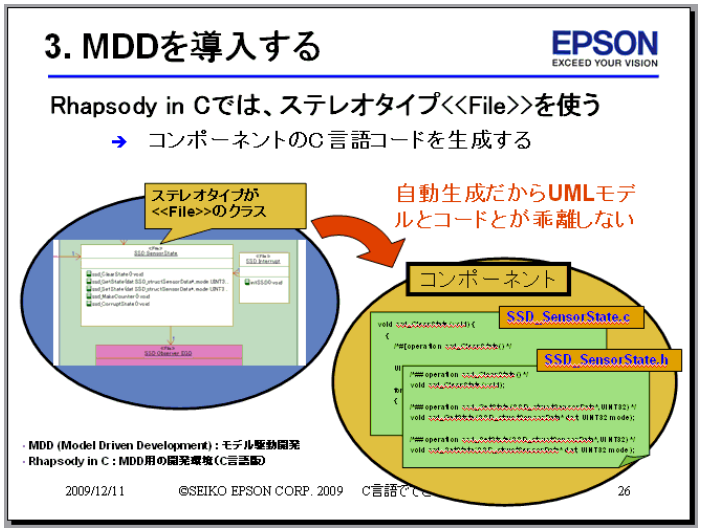
EPSON
EXCEED YOUR VISION

モデル駆動開発(MDD)へつなげることができた

➔ C言語を使う技術者でもMDDを実践できた

オブジェクト指向
開発が未経験の

JEITAソフトウェアエンジニアリング技術分科会
2009年度ワークショップ発表で発表



2.1 定量的な効果

EPSON
EXCEED YOUR VISION

後続機種において部品化の効果も出ている

➔ 再利用しやすい構造も作りこむことができた

1. 再利用性

   後続機種で高い再利用率を実現できた

   - コンポーネント再利用率＝91%

   ※ コンポーネント再利用率=.hと.cのペアを修正無く再利用できた割合

2. 生産性

   再利用性の向上が生産性の向上にもつながった

   - 開発工数の20%削減

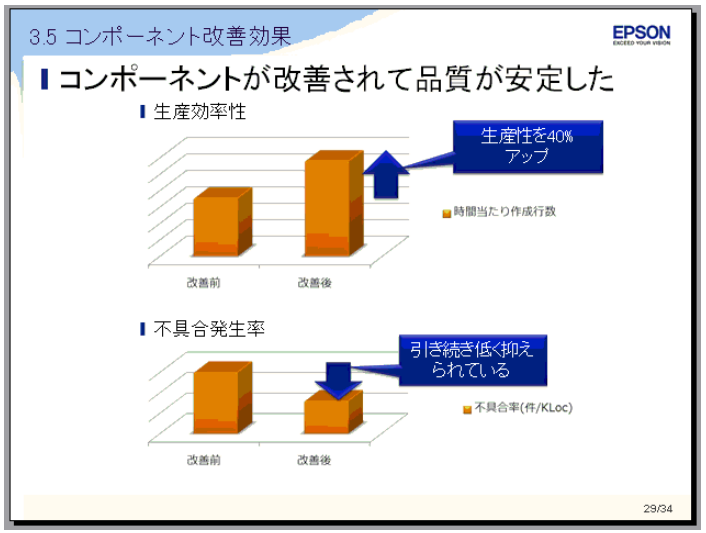
3. 低リソース環境でのSPLE挑戦

EPSON
EXCEED YOUR VISION

厳しいリソース環境でも適用できた

➔ SPLEの基盤となるコンポーネントも作った

JEITAソフトウェアエンジニアリング技術分科会
2011年度ワークショップ発表で発表



※SPLE : Software Product Line Engineering

3.1 定量的な効果

EPSON
EXCEED YOUR VISION

トレーニング後にさらに品質と生産性が向上した

➔ 単にモデル化するよりも高い効果がでた

1. 部門独自で取り組んだとき

   品質悪化に歯止めができ生産性も向上した

   - 不具合を80%削減、開発期間を30%削減した

2. トレーニング受講後の継続開発

   コンポーネントが改善されて品質が安定した

   - 不具合発生率は、引き続き低く抑えられている
   - 生産性がさらに40%向上した

4. 課題と、当面の対策

EPSON
EXCEED YOUR VISION

名前付けに必要な語彙力が足りない

モデル要素に対する

➔ 関連名が「を制御する」では意味がない

ポイント

1. 振る舞いの範囲が分かる動詞が使えるか？
   例えば「を制御する」を具体的に区別できること
   ➔ 「を加熱する」「を回転する」など・・
2. 語彙は5万語が必要となる

   専門用語を含めると7万語

   少なくとも大学卒業レベルとなる5万語は欲しい
   ➔ 語彙力推定テスト（NTTコミュニケーション科学基礎研究所）
   <http://www.kecl.ntt.co.jp/icl/lirg/resources/goitokusei/goi-test.html>

小学生：5千～2万語
中学生：2万～4万語
高校生：4万～4万5千語
大学生：4万5千～5万語

日経IT PROの記事「SEよ、豊かな語彙を持て」によると、20代で平均2万5000語、30代で3万8000語、40代で4万1000語、50代で4万5000語しかない
<http://itpro.nikkeibp.co.jp/article/Watcher/20120607/400862/>

4.1 グループワークを活用する

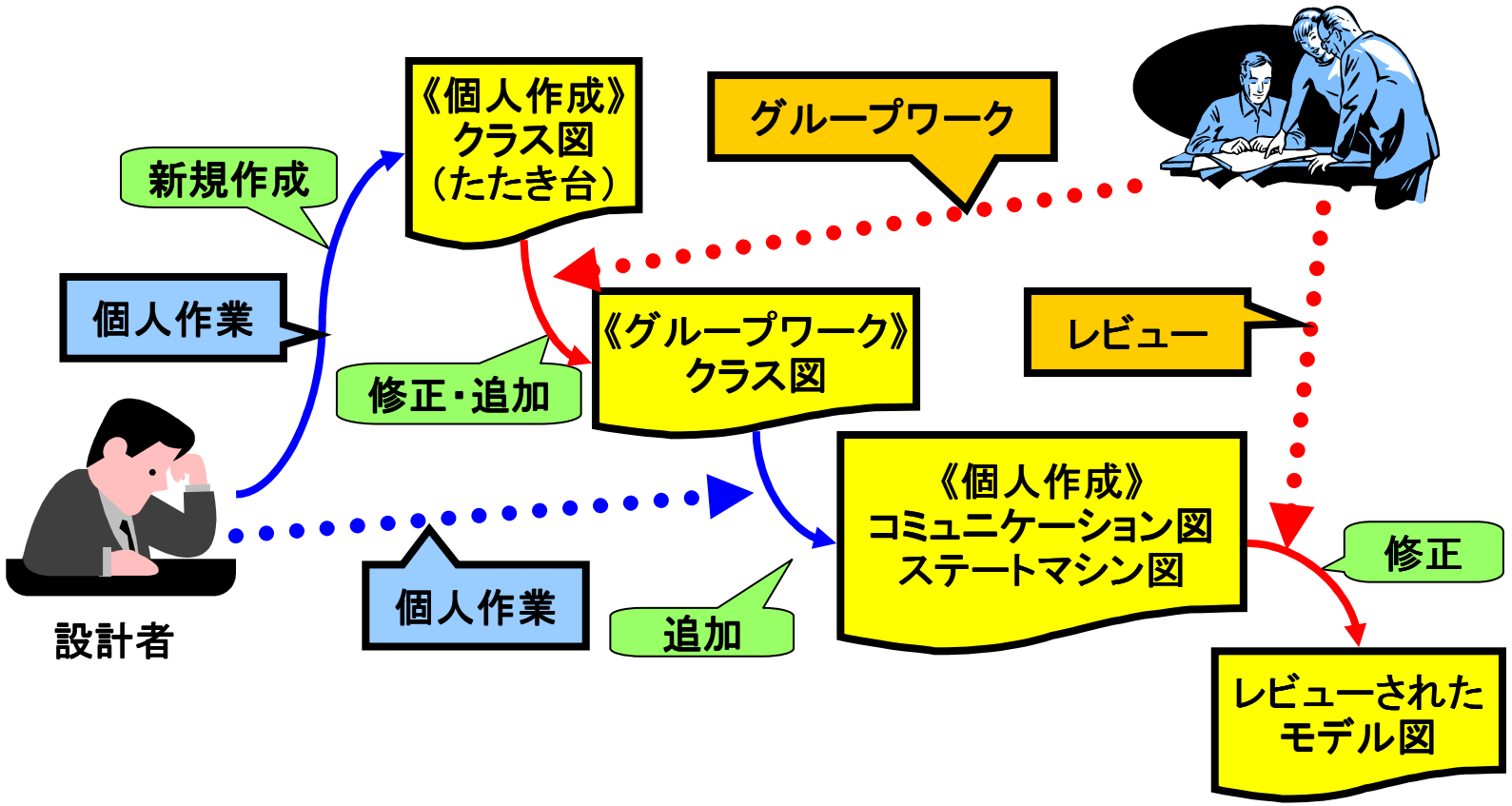
EPSON
EXCEED YOUR VISION

グループワークで語彙力の不足を補う

とりあえず

トレーニング
できないが

➔ とは言え、各人の語彙力向上も必要になる



4.2 名前をつけて詳細を無視する

EPSON
EXCEED YOUR VISION

名前付けはソフトウェア工学の基本である

➔ 黎明期の指摘をやり切ることができていない

抽象化の定義

※E.W.Dijkstra, C.A.R Hoare, O-J.Dahl,
“構造化プログラミング”,1972 (日本語訳 1975)

- 演算に名前をつけて、
  その動作の仕組みを無視すること
- データに名前をつけて、
  その構造の詳細を無視すること

ルーチンが行うことをすべて表現する。
ルーチン名には、すべての出力とすべての副作用が記述されていなければならない。
・・・「コードコンプリート－完全なプログラミングを目指して」ステーブ マコネル 著,1994

5. まとめ

EPSON
EXCEED YOUR VISION

満足度は高く、気づきのポイントも多い

ポイント

1. 延べ200名以上が受講している
2. モデル駆動開発へつなげることができた
3. 厳しいリソース環境でも適用できた
4. 名前付けに必要な語彙力が足りない

Ⅴ. まとめ

Ⅰ. 教えられないという問題

Ⅱ. 明確な手順と規準を用意する

Ⅲ. モデリングの具体論

Ⅳ. 成果と課題

Ⅴ. まとめ

Ⅴ. まとめ

EPSON
EXCEED YOUR VISION

センスに依存しないモデリングを実現する

ポイント

1. センスに依存する部分が多いと教えられない
2. 誰でもできる技法を用意してトレーニングする
3. 機能一覧をクラス図へ変換する手順を作った
4. 満足度は高く、気づきのポイントも多い

Tips. やり切っていないだけ

EPSON
EXCEED YOUR VISION

パルナスは語る

Q. What are the most exciting/promising software engineering ideas or techniques on the horizon?

A. I don't think that the most promising ideas are on the horizon. They are already here and have been here for years but are not being used properly.

➔ 既にやるべきエンジニアリングはある。ただそれをきちんとやっていないだけ

※D.L.Parnas "ACMのフェローインタビュー",2007 URL:http://www.sigsoft.org/SEN/parnas.html

EPSON

EXCEED YOUR VISION

モデ脳の問題をモデリングする

EPSON
EXCEED YOUR VISION

モデ脳検定に受かることはできるのか？

1. 問題文
2. 機能一覧を作る
3. 機能一覧からクラス図を作る
4. クラスの粒度を適正化する
5. 空クラスを削除する
6. 関連を整理する
7. モデリングの結果

1. 問題文

EPSON
EXCEED YOUR VISION

モデ脳の“漁夫の利”をモデリングしてみる

➔ 漁師が利益が得ることを示せるか？

問題文

蛤（ハマグリ）が日向ぼっこをしていたところに、鷸（シギ）がやってきて、蛤の肉をつかみました。蛤も、負けじと殻をとじて、鷸のくちばしをはさみました。どっちも離そうとせず、ずっと争っていたところに漁師が来て、鷸と蛤をいっぺんに捕まえてしまいました。

※出典：UMTP モデ脳検定 <http://www.umtp-japan.org/modules/modeno/>

2. 機能一覧を作る

EPSON
EXCEED YOUR VISION

クラス図に対応づける機能を一覧表にしておく

➔ 構造に反映する機能をすべて洗い出す

機能一覧の記述例

目的語	動詞	制約条件
ハマグリの肉	を掴む	シギが掴む
ハマグリの殻	を閉じる	ハマグリが閉じる
シギのクチバシ	を挟む	殻を閉じるとき
ハマグリ	を捕まえる	争っているとき
シギ	を捕まえる	争っているとき

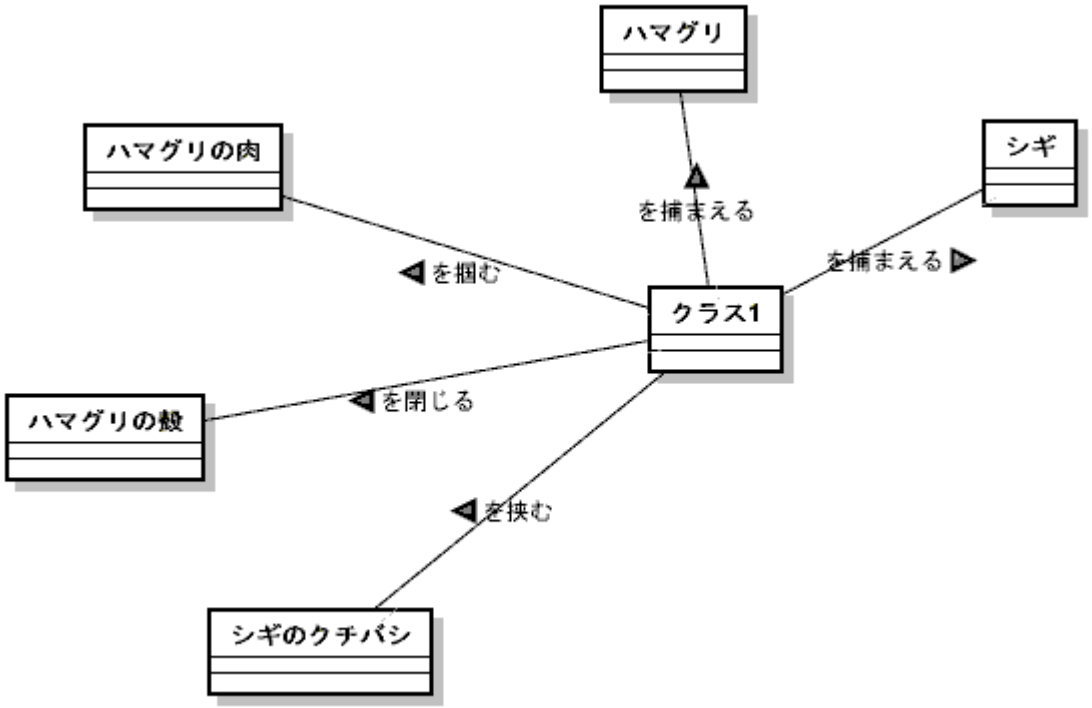
Who(だれが)、What(何を)、
When(いつ)、Where(どこで)、
How mach（どの程度）
How to (どのように) など

3. 機能一覧からクラス図を作る

EPSON
EXCEED YOUR VISION

機能一覧をクラス図に対応づける

➔ クラスと関連を読み上げると機能一覧に戻る

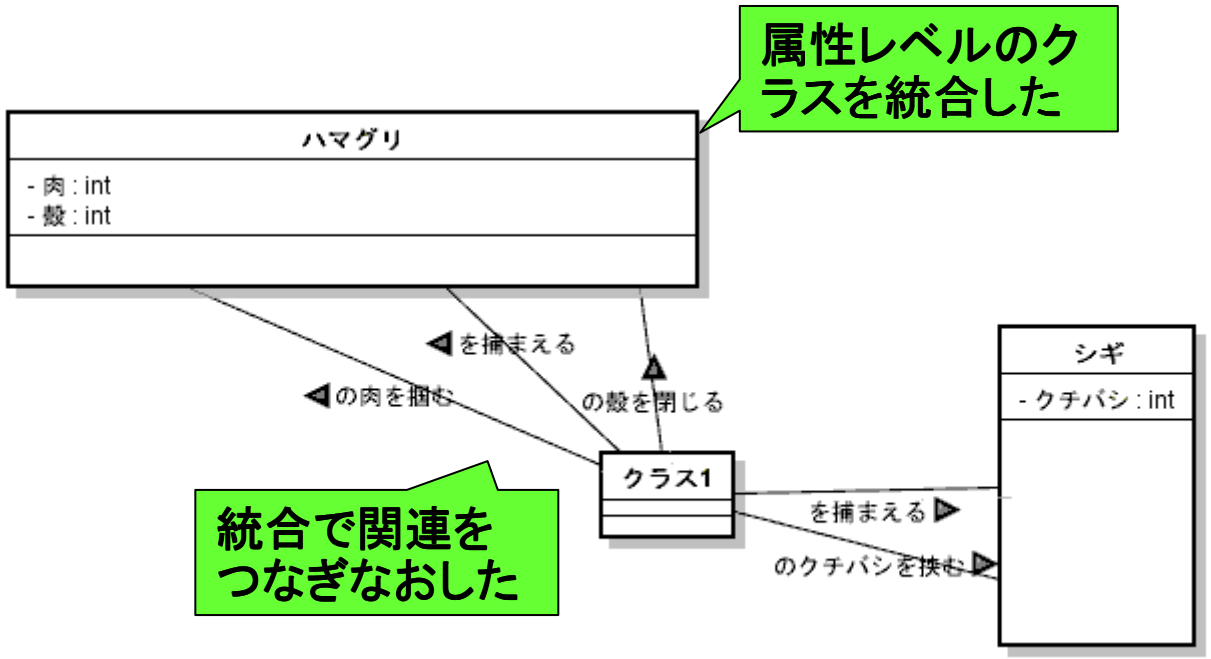


4. クラスの粒度を適正化する

EPSON
EXCEED YOUR VISION

クラスの粒度を適正化すると属性が現れる

➔ 属性化で言葉の散らばりも、少し解消する

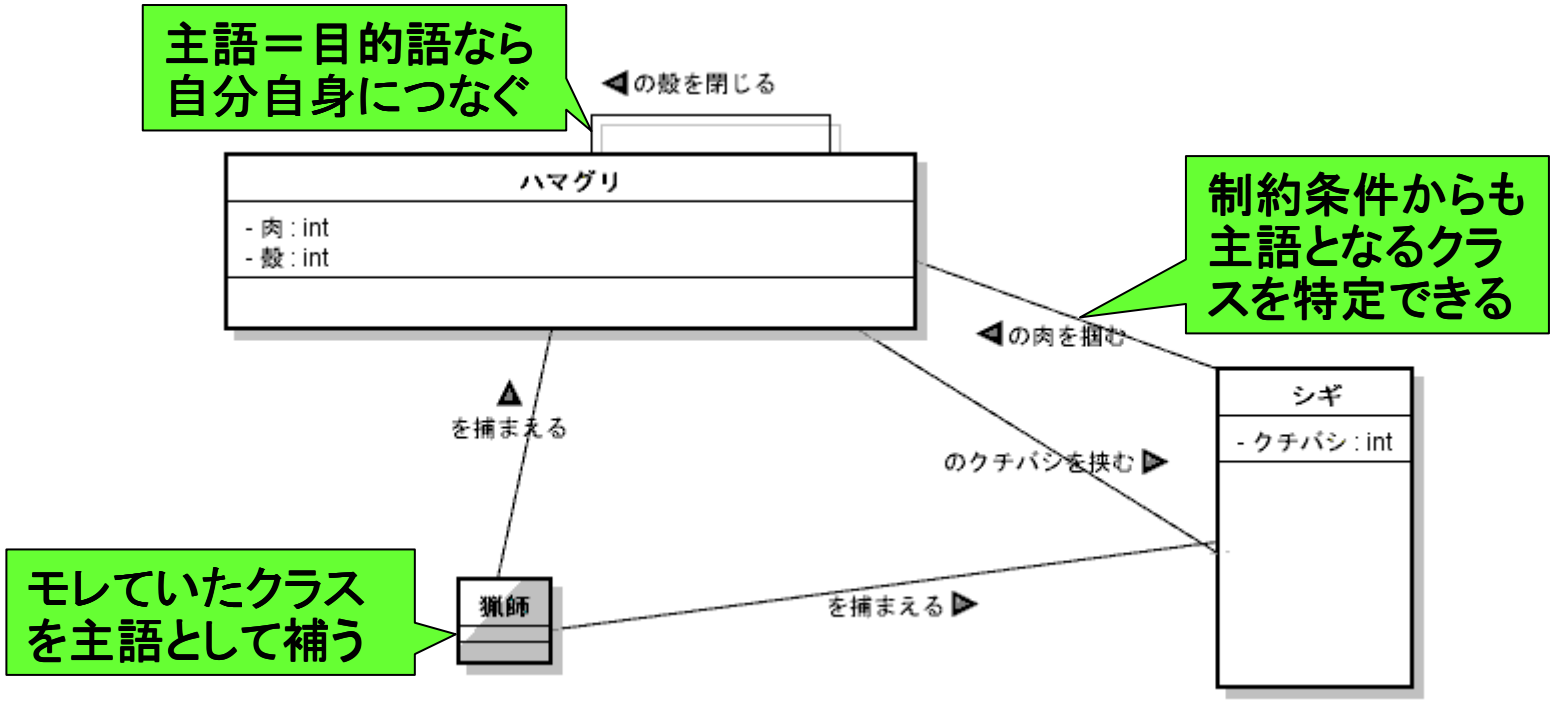


5. 空クラスを削除する

EPSON
EXCEED YOUR VISION

空クラスを削除すると、クラス図らしくなる

➔ つなぎなおすときに、<u>モレが見つかる</u>

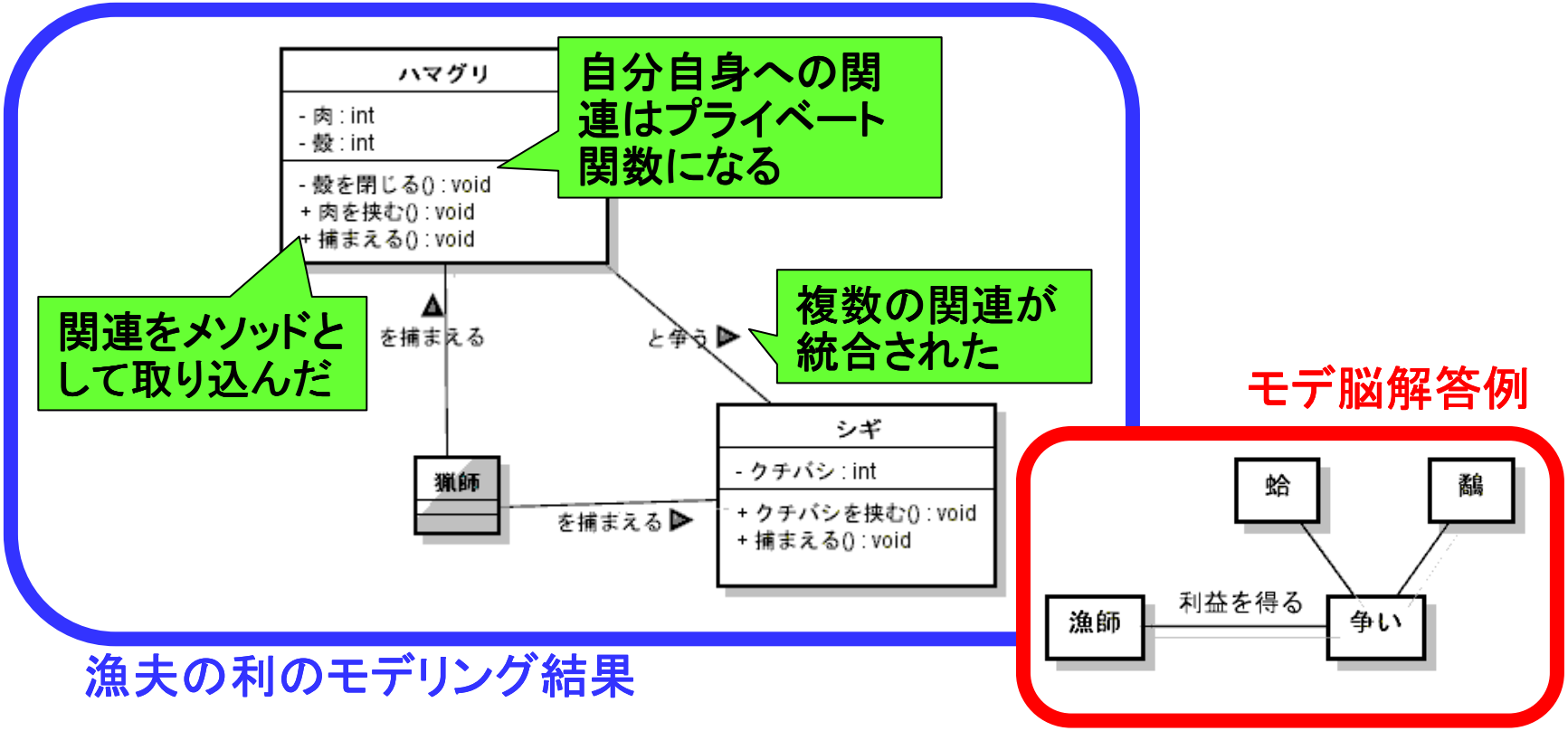


6. 関連を整理する

EPSON
EXCEED YOUR VISION

関連を整理すると、クラス図として完成する

➔ モデ脳解答例とは異なる結果になる



※解答例の出典：UMTP モデ脳検定 <http://www.umtp-japan.org/modules/modeno/>

7. 結果

EPSON
EXCEED YOUR VISION

モデルは作れるが、漁夫の利は見出せない

ポイント

1. 問題文にない機能はモデリングできない
   「漁師が利益を得る」という機能は問題文にない
   - ➔ 問題文にない機能はモデルに表現できない
   - ➔ とは言え、途中で気づく可能性はある
2. “コト”に関するモデリング力が弱い
   「争い」のようなコトはクラスに表現できない
   - ➔ コトが重要なドメインには向かないかも