社内エキスパートの育成による、ソフトウェアプロダクトライン(Software Product Lines)の全社展開

丹羽 徹　オムロン株式会社
グローバルものづくり革新本部
開発プロセス革新センタ SPILIT推進部

赤松 康至　オムロン株式会社
インダストリアルオートメーションビジネスカンパニー
商品事業本部 技術開発センタ 第3技術部

背景

オムロンの主な製品とソフトウェア

■ 組込み製品として機器にソフトウェアが組み込まれている

制御機器・FAシステム

車載電装部品

健康医療機器

社会システム

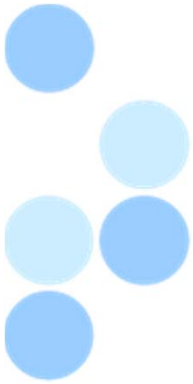
環境関連機器

オムロン製品のソフト開発規模の推移

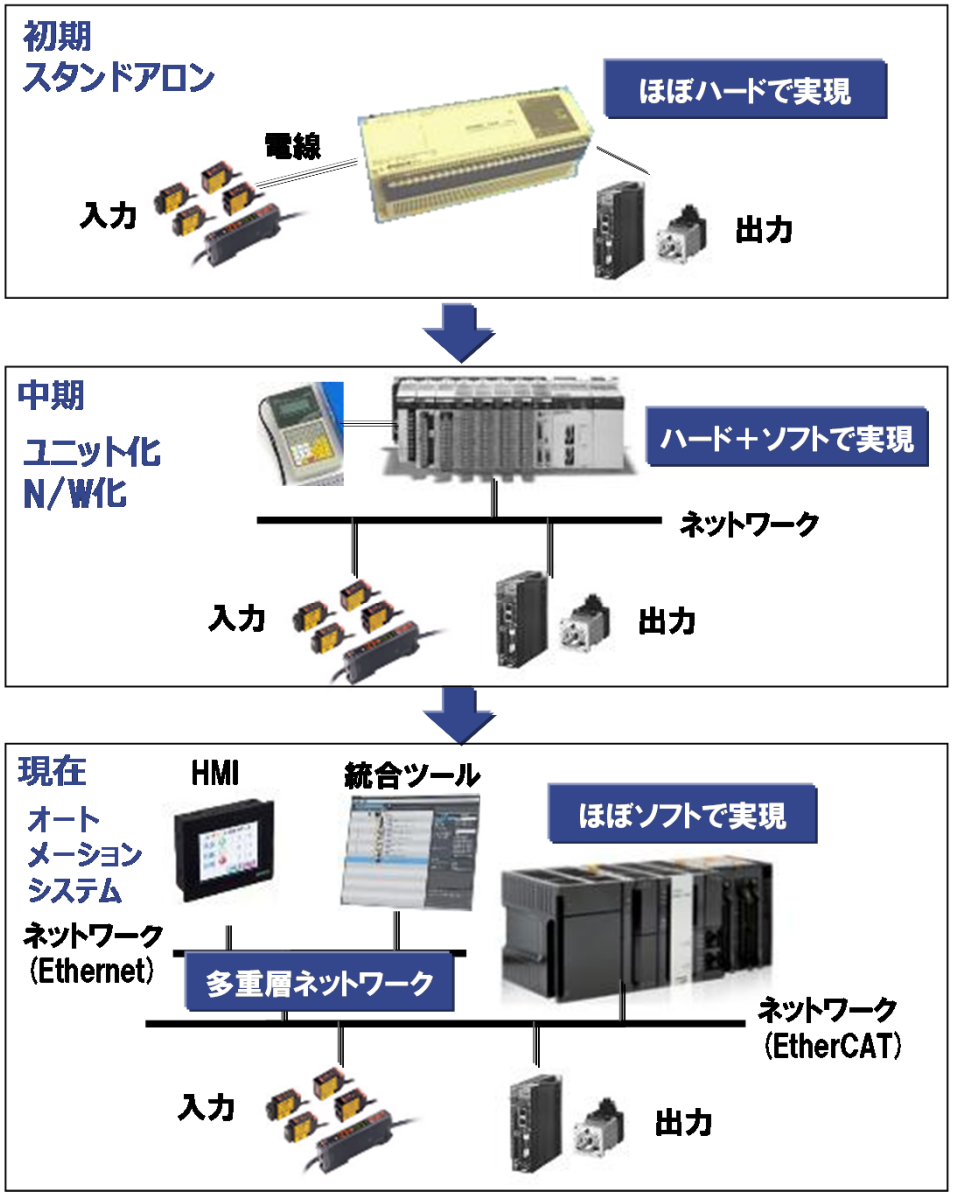
発表時の投影のみと致します

オムロン製品のソフト開発規模の推移

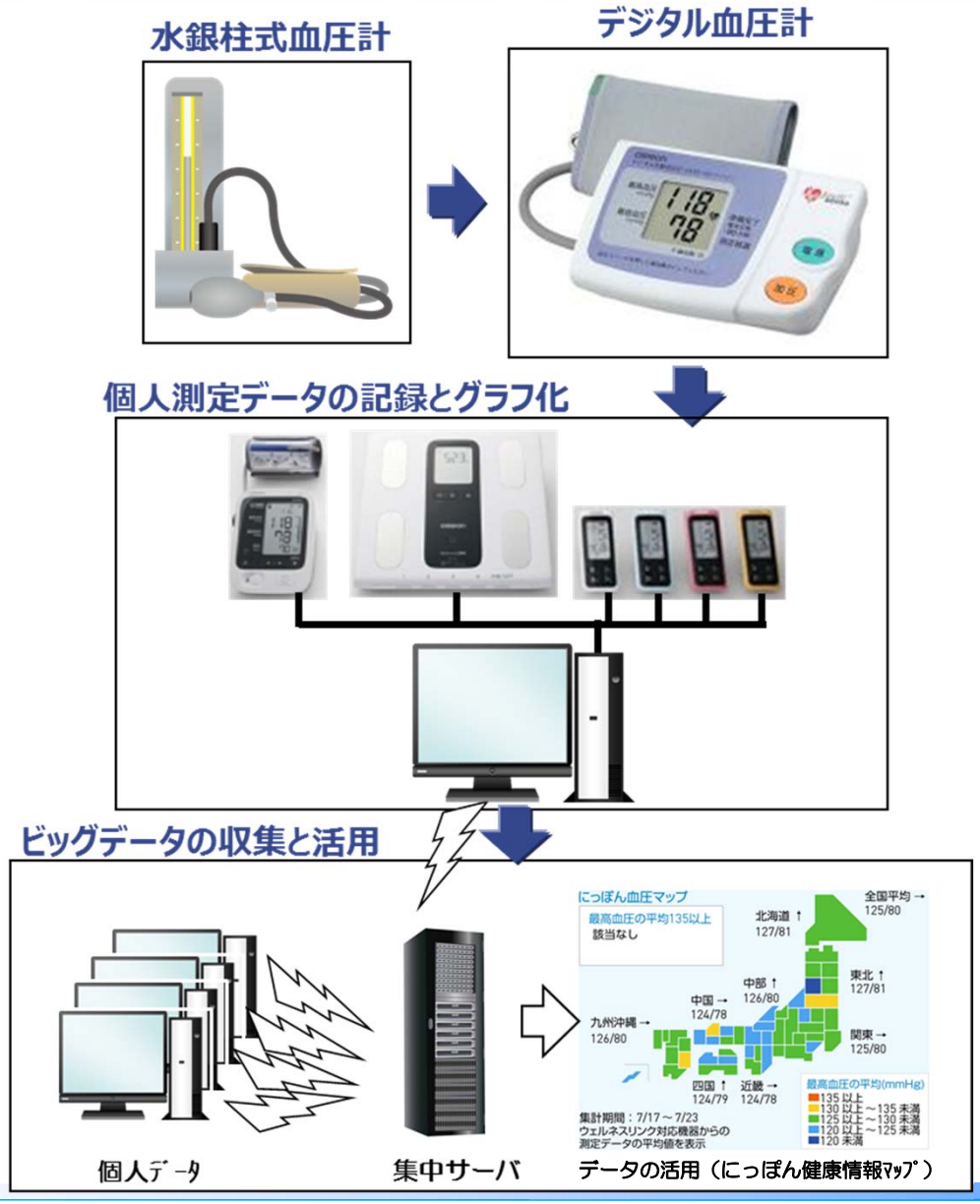
- 規模が増え続けている
- なぜ規模が増えるのか？



製品の進化の例（制御機器・FAシステム）



製品の進化の例（健康医療機器・サービス）



オムロン製品のソフト開発規模の推移

- 規模が増え続けている
- なぜ規模が増えるのか？

- 機能が増えている。客先の要望が増えるため。人間のニーズが増える、高度化、多様化する
- ハードで実現していた機能をソフトで実現。原価低減、対応スピード向上
- 製品のシステム化、複数の機器がつながって動作する
- CPUの性能・集積度の向上

ソフトの規模増大による課題

■製品機能のうちソフトウェアが占める割合が多くなり、ソフトウェアが製品開発のボトルネックとなっている

◆***ソフト開発がプロジェクトのクリティカルパスとなる***

・「ソフトのせいで製品リリースが遅れた」

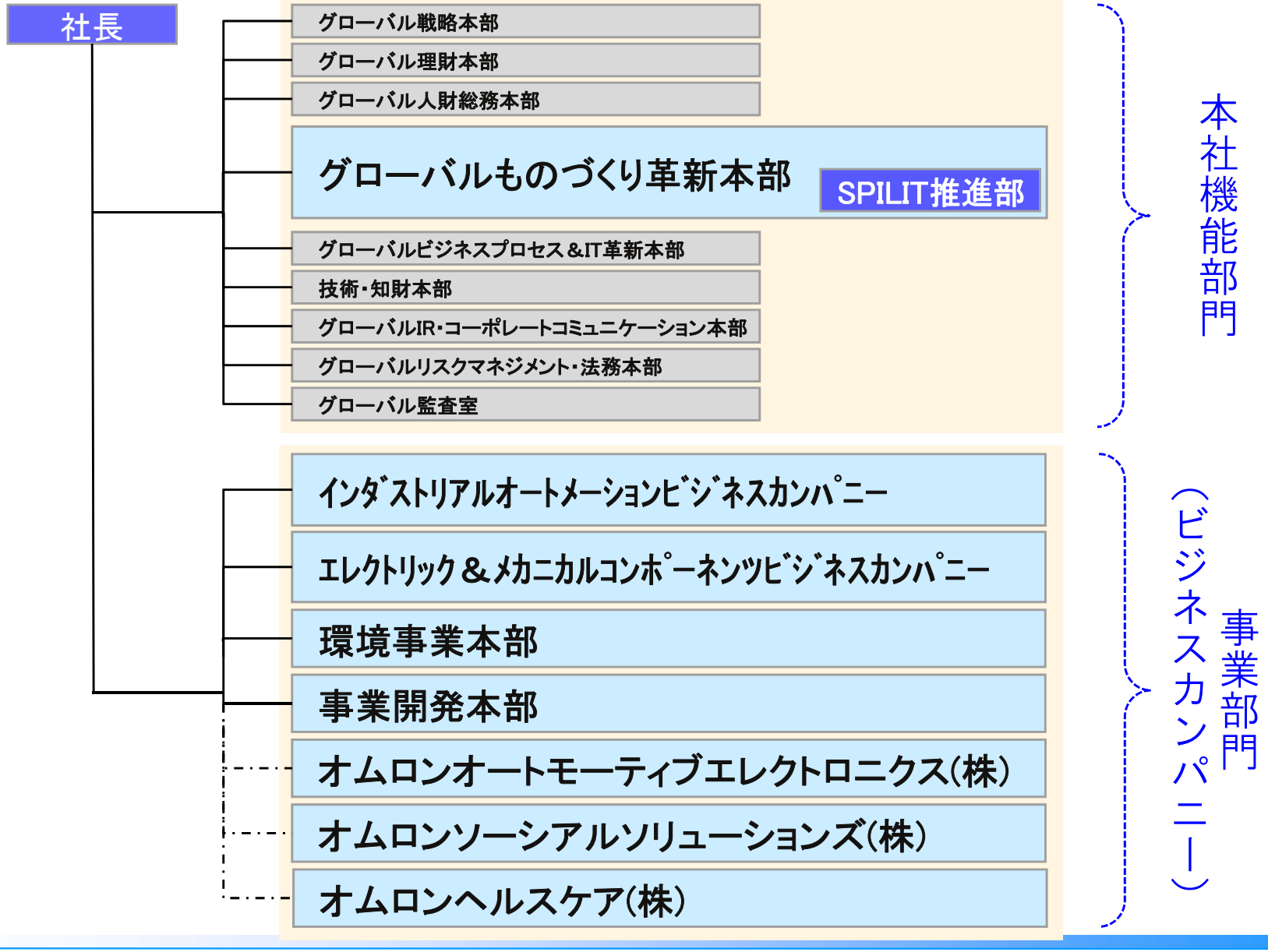
◆***ソフト開発費が高騰***

・「なんで(ソフトだけ)そんなにかかるんだ」

◆***ソフト不具合が多い***

・「またソフトの不具合か。。。」

オムロンの組織(概要)



改善の歴史と課題

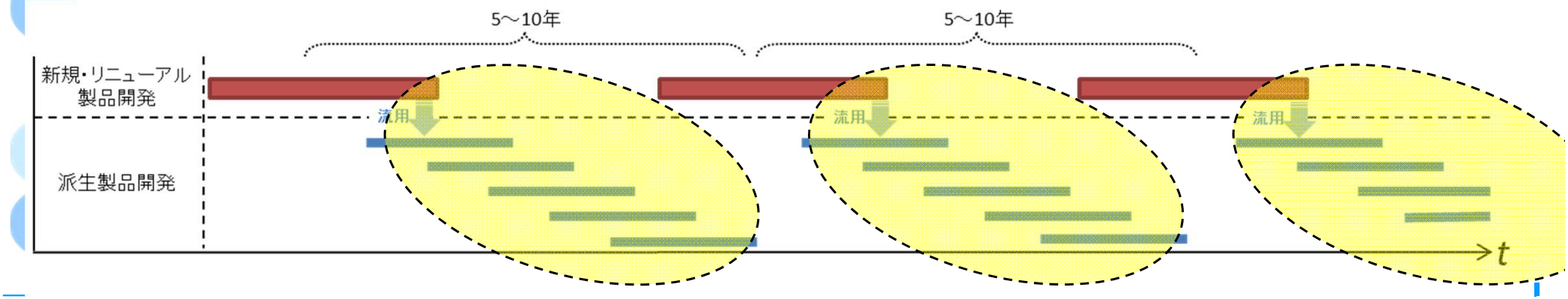
■ソフトウェアのQCD 向上を狙いにソフトウェアプロセス改善活動（SPI）に取り組んでいる

■プロセスリファレンスモデル（CMMIなど）を用いた改善活動を、個々の製品の事業体である各ビジネスカンパニーで実施している

改善の歴史と課題

- 弊社製品の特性上、製品群ごとに5～10年スパンでハードウェアも含め新規開発もしくは大きく製品リニューアルが行われ、その後複数の派生開発を行うというサイクルを繰り返している

- この新規・リニューアルでのソフトウェアの作りがまずいと、後の派生開発のQCDに大きく影響を与える構造になっている
  - ◆ *「なぜ派生開発なのにこんなに開発費がかかるのか？」*
  - ◆ *「なぜそれくらいの変更を最初から想定しておかなかったのか？」*
  - ◆ *「引き合いはあるのに（人が足りなくて）新たな派生開発を受注できない」*



解決策の方向性

規模増大の課題に対する打ち手、目指すべき姿

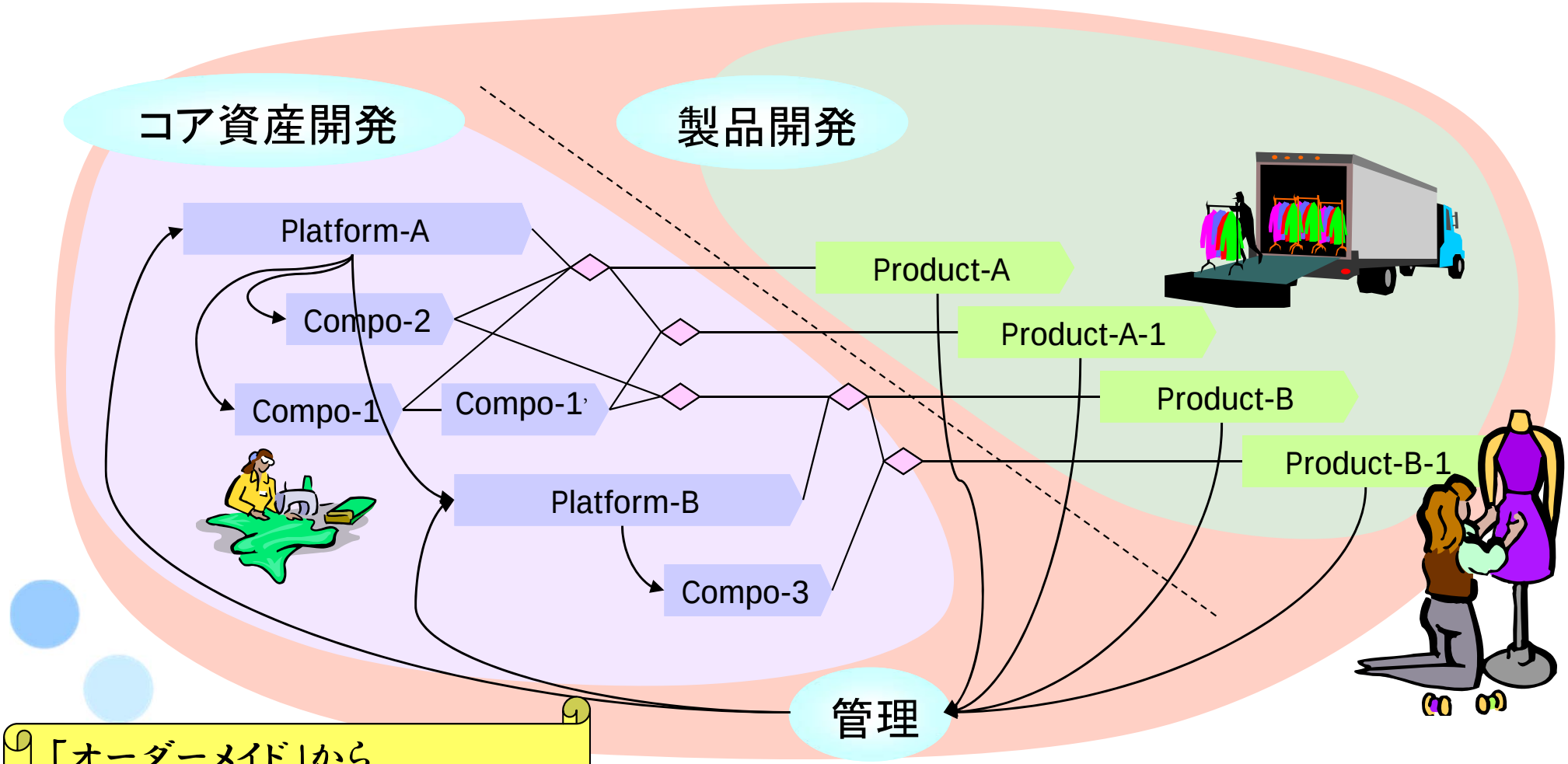
■ 打ち手

- ◆ *ソフトの規模増大の打ち手として、最も効果的なのは「作らないこと」*
- ◆ *そのために、ソフトの部品化（ソフトウェアプロダクトライン：SPL）が効果的と考えた*

■ ありたい姿（オムロン全社として）

- ◆ *カンパニーのビジネスゴールにマッチしたソフトの改善/革新活動が実施され続け、効果を出し続けている*
- ◆ *特にSPLについては革新的な打ち手の一つとして、適切なタイミングで取り入れられている*

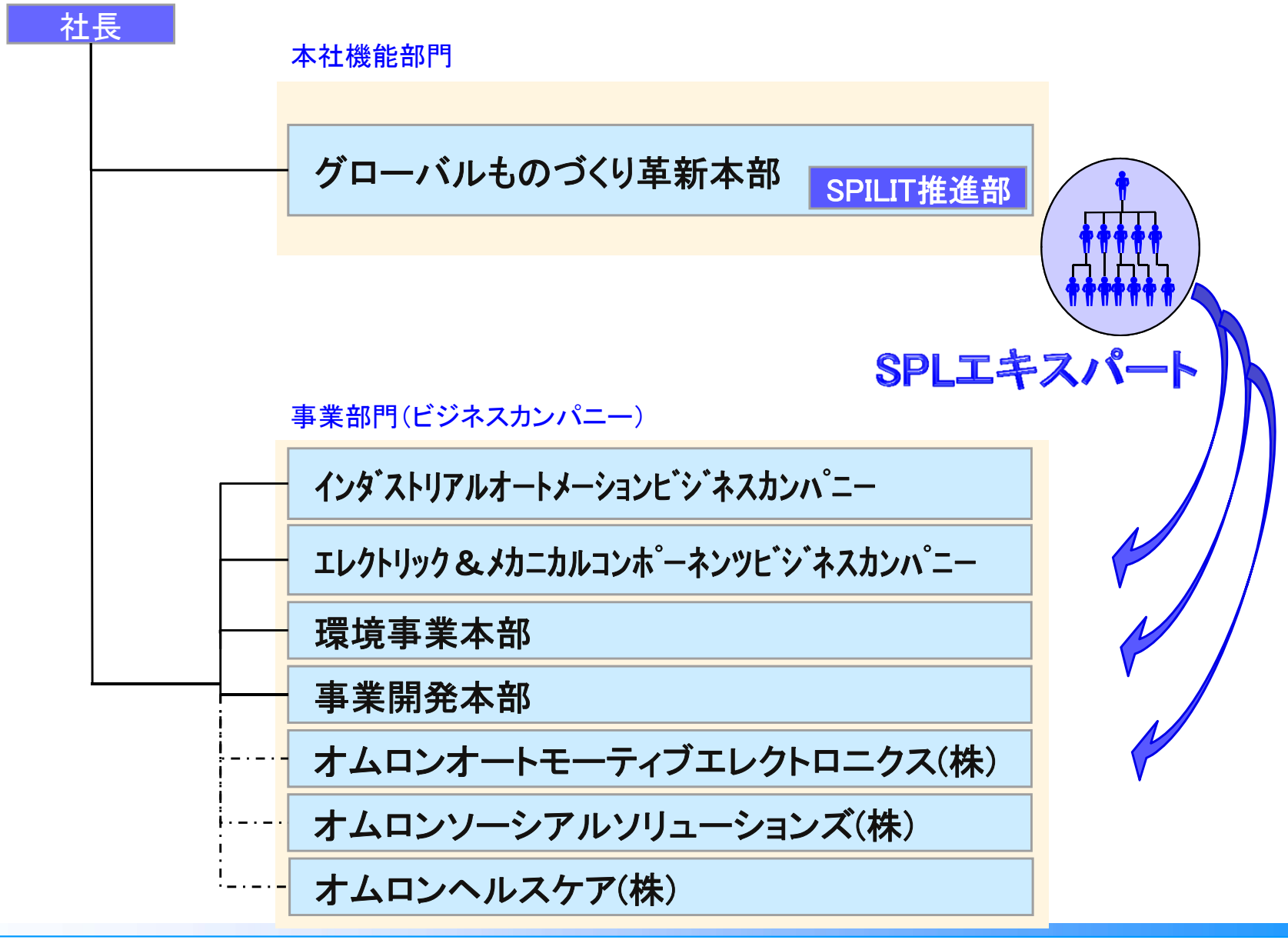
ソフトウェアプロダクトラインの考え方
（Software Product Lines = SPL）



「オーダーメイド」から
「カスタムメイド」、「レディメイド」
への変革！

改善策の実現方法

本社機能部門として“SPLエキスパート”を組織化
適切なタイミングで事業部門に展開



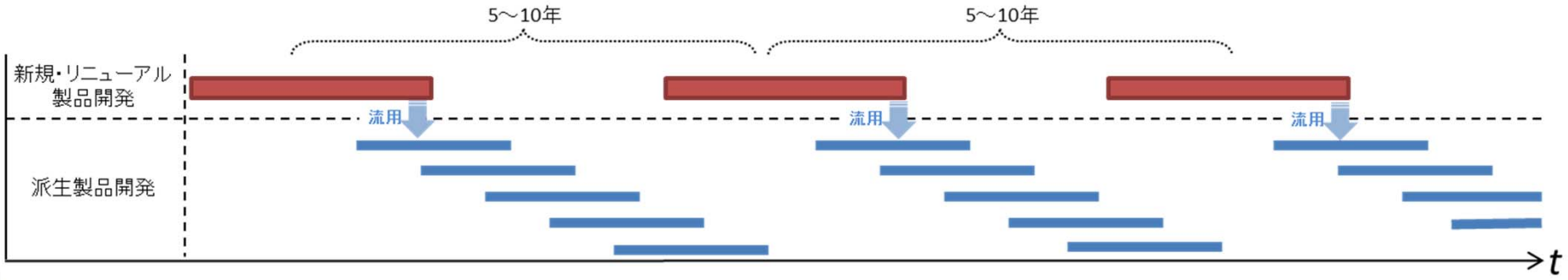
本社機能部門にSPLエキスパートを持つ理由

■ 各事業部門では5～10年に一度しか機会がないため

◆ ***プロセスを作っても陳腐化する***

◆ ***ノウハウを持った人材が開発現場からいなくなる***

・ 配置転換になる。マネージャになる



エキスパートの育成方法

育成のためのツール

① マテリアル

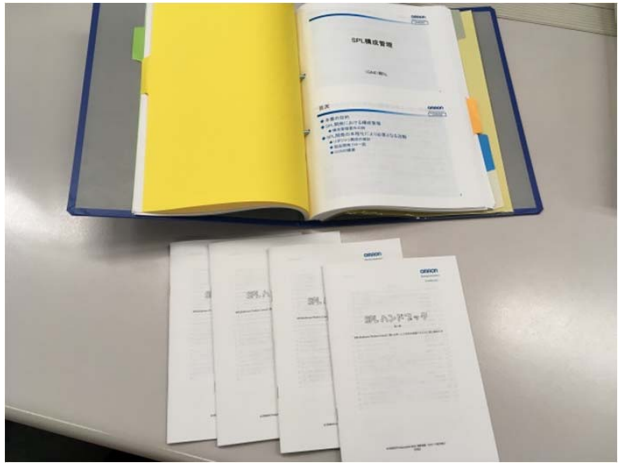
- ***エキスパートのスキルアップのためのトレーニング資料***

② 人材育成フレーム

- ***エキスパートが身に着けるべきスキル定義と獲得のための仕組み***

マテリアルの考え方

- 設計の方法だけでなく、SPL特有の管理方法や組織編制の方法まで含む
  - ***3つのカテゴリと29の活動領域に体系化****
- 全社共通で使える部分とそうでない部分を分離
- 事業部門への展開の中で新たな知見も適宜盛り込む
- 全約400ページ。定期的にエキスパート内でトレーニング



マテリアルとハンドブック

3つのカテゴリと29の活動領域

*: Paul Clements and Linda Northrop, “Software Product Lines: Practices and Patterns”, Addison- Wesley(2001)を参考に作成

マテリアルの例

発表時の投影のみと致します

当日会場に実物の一部を持参
しますので、興味のある方は
休憩時間にでもご覧ください

SPL人材育成フレーム（スキル定義）

【コア能力】

職種に限らず、プロフェッショナル人材に求められる共通的な能力

コア	1	開拓力
	2	情報収集力
	3	新規企画力
	4	提案・交渉力
	5	実行力
	6	自己開発力
	7	育成力

レベルⅠ：本組織における能力定義の意味、必要な理由を理解している
レベルⅡ：支援を受けつつ遂行できる
レベルⅢ：独力で遂行できる

【プロフェッショナル能力*】

SPLエキスパートの役割・責任を果たすために必要な専門能力

プロフェッショナル	1-1	ソフトウェア工学	アーキテクチャ定義
	1-2		アーキテクチャ評価
	1-3		コンポーネント開発
	1-4		COTSの利用
	1-5		既存資産の発掘
	1-6		要求エンジニアリング
	1-7		ソフトシステム統合
	1-8		試験
	1-9		関連ドメインの理解
	2-1	技術管理	構成管理
	2-2		データ収集/メトリクス/追跡
	2-3		作成/購入/発掘/委託の分析
	2-4		プロセス定義
	2-5		スコープ定義
	2-6		技術計画策定
	2-7		技術リスク管理
	2-8		ツールによる支援
	3-1	組織管理	ビジネスケースの作成
	3-2		顧客インタフェース管理
	3-3		調達戦略策定
	3-4		資金調達
	3-5		プロダクトラインの着手と制度化
	3-6		市場分析
	3-7		プロダクトライン運営
	3-8		組織計画策定
	3-9		組織リスク管理
	3-10		組織編成
	3-11		技術予測
	3-12		トレーニング

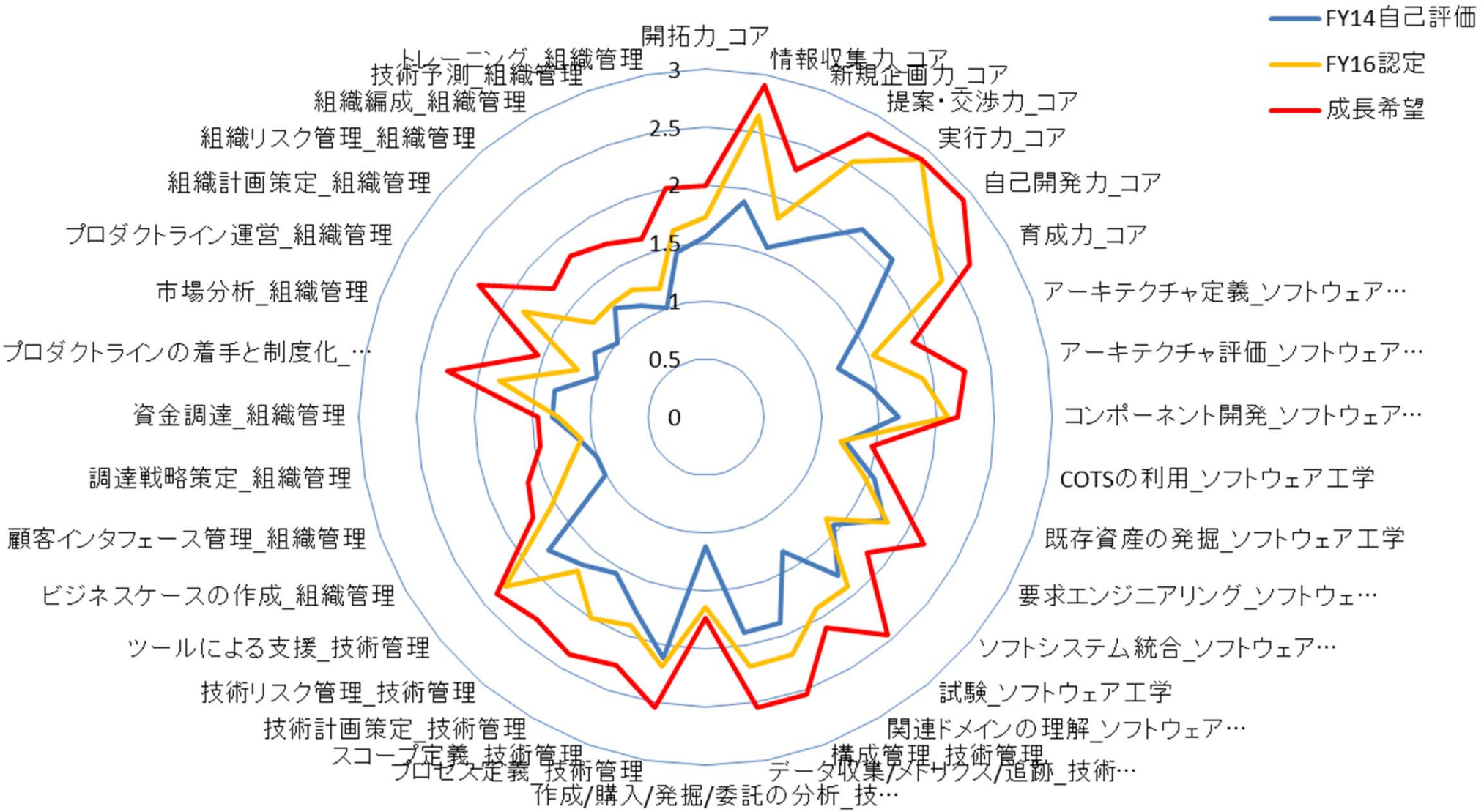
レベルⅠ：知識として知っている
レベルⅡ：自ら実践できる
レベルⅢ：相手が理解可能な手段で、指導や提案ができる

*: Paul Clements and Linda Northrop, “Software Product Lines: Practices and Patterns”, Addison- Wesley（2001）を参考に作成

SPL人材育成フレーム（獲得のための仕組み）
～個別育成計画書例～

発表時の投影のみと致します

SPL人材育成フレーム（能力向上の結果）



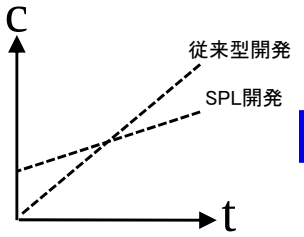
SPL人材育成フレーム（能力の内訳）

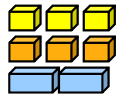
発表時の投影のみと致します

開発現場での活動

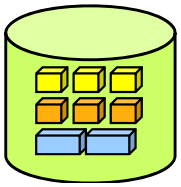
開発現場での活動ステップ概要

役割	SPL企画	コア資産開発	製品開発準備	製品開発
事業部門	SPLの実施判断	コア資産の開発	コア資産を活用した製品展開の仕組み準備	製品開発
本社機能部門	SPLの考え方、投資対効果説明	コア資産分析/開発方法のアドバイス	コア資産を活用した製品展開の仕組み提案	仕組みがうまく回っているかチェック

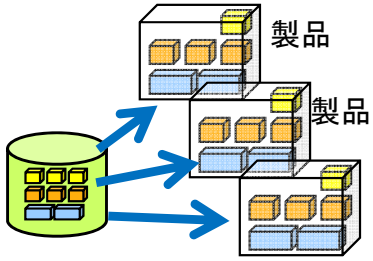




共通部品
（コア資産）



共通部品の入れ物
（リポジトリ）

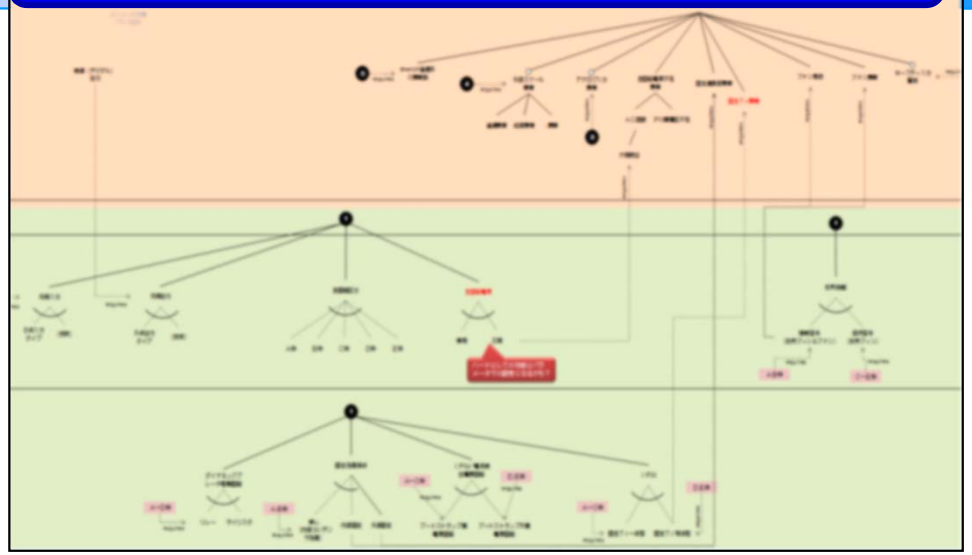


開発現場での活動例

SPL企画（スコープ・製品群の分析）

項目			仕様安定度	特性	A1	A2	A3	A4	C1
通信機能	上位通信機能	通信プロトコル	安定	共通	○	○	○	○	○
		Ethernet通信機能	安定	可変点	○	○	×	×	○
	メンテナンス通信機能	USB通信制御機能	安定	共通	○	○	○	○	○
		シリアル通信機能	安定	共通	○	○	○	○	×
ハードウェアインターフェース機能	IO	XX入力機能	安定	可変点	○	○	○	×	×
		XX出力機能	安定	可変点	○	○	○	×	×
	モニタ	XXモニタ	安定	可変点	×	×	○	×	×
	表示	XX表示	安定	共通	○	○	○	○	×
製品機能	基本機能	機能AA	安定	共通	○	○	○	○	○
		機能BB	安定	共通	○	○	○	○	○
		機能CC	安定	可変点	○	○	×	○	○
	応用機能	機能DD	変更可能性あり	共通	○	○	○	○	○
		機能EE	変更可能性あり	共通	○	○	○	○	○
		機能FF	変更可能性あり	共通	○	○	○	○	○

コア資産開発（フィーチャーモデリング）



コア資産開発（可変性実装ガイド）

			実行時	リンク時	コンパイル時	コーディング時
Link方式	推奨	makefileでリンクオブジェクトを変更		○		
インスタンスLink方式	推奨	makefileでインスタンスを生成するオブジェクトを変更		○		
関数マクロ方式	推奨	関数のテンプレートを用意しておき、マクロ定義で切り替える			○	
関数呼出If-Switch方式	推奨	関数を呼び出す側でif文、Switch文で処理を切り替える	○			
関数ポインタ方式	推奨	関数ポインタが指す実体関数の違いで処理を切り替える	○			
インスタンスIf-Switch方式	推奨	インスタンスの違いで処理を切り替える	○			
ステートメントIf-Switch方式	推奨	処理をif文、Switch文で切り替える	○			
ステートメントマクロ方式	推奨	ステートメントのテンプレートを用意しておき、マクロ定義で切り替える			○	
Define方式	推奨	ヘッダーファイルにて数値を#define定義する			○ 選択可	○ 選択可
変数方式	推奨	数値を変数で保持する。	○			
コンポーネント呼出If-Switch方式	非推奨	コンポーネントを呼び出す側でif文、Switch文で処理を切り替える	○			
Ifdef方式	非推奨	#ifdefで処理を切り替える			○	

製品開発（SPL成果物要件評価）

要件No.	ソフトウェア仕様要件評価	アーキテクチャ設計要件評価	製品バリアント決定管理要件評価	可変性実装要件評価	テスト要件評価	構成管理要件評価
1	LI	FI	FI	FI	FI	NA
2	LI	FI	FI	FI	FI	LI
3	LI	FI	LI	FI	FI	FI
4	FI	LI	FI	FI	FI	FI
5	LI	FI	FI		FI	FI
6	FI	FI	LI		FI	FI
7	FI	FI			FI	NA
8	FI	FI			LI	LI
9	NA	FI			FI	LI
10	LI	LI			FI	LI

開発現場での成果

対象	効果
製品群A	コア資産による客先での早期デモ実現による数十億円規模の商談獲得
製品群B	新製品群での派生開発費従来比30%減
製品群C	コア資産の活用により、新商品ラインナップの第1弾、第2弾を計画通りリリース。売上拡大（130%）へ貢献
製品群D	製品ごとに6種類あったソフトを1本化することで、生産ラインの段取り大幅改善
製品群E	ソフトウェアテストの工数従来比40%減（見込み）

おわりに

なぜできたか？

■自社を良く観察/理解し、自分達に合ったソリューションは何かをひたすら考える

◆*社外ベンチマークも必要だが、何をやったかではなく、なぜやったかを見極めることの方が重要*

■何としてでもやり遂げるという強い思い

◆*自分達に合ったソリューションが見つかったら、それを達成するために、社内のありとあらゆる障壁を乗り越える覚悟を持つ（他責にしない）*

ご清聴ありがとうございました