

「機能－作業マトリクス」を用いた、 プロジェクトの簡潔な進捗管理手法の適用事例

- 田中 裕大 (株式会社 東芝 ソフトウェア技術センター)
- 小森 真紀 (株式会社 東芝 ソフトウェア技術センター)
- 黒川 久美子 (株式会社 東芝 府中事業所 技術・情報システム部)
- 大淵 一彦 (株式会社 東芝 府中事業所 交通ドライブシステム部)

1. 背景と課題

- － 背景
- － 進捗管理の課題
- － 課題の根本原因

2. 改善策の検討

- － 既存手法
- － 機能－作業マトリクス
- － 根本原因への対策

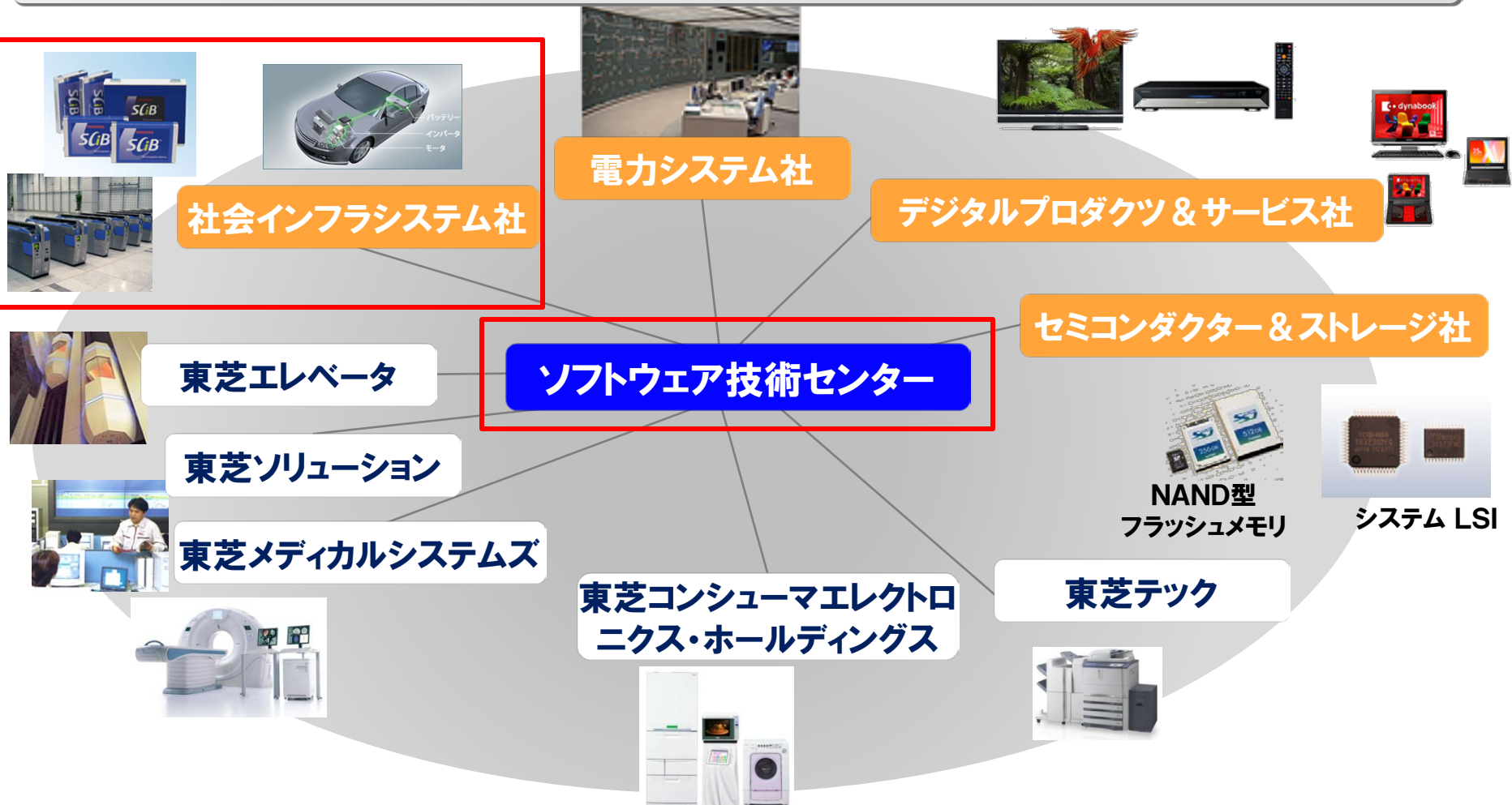
3. まとめ

- － 試行評価
- － 課題と今後の予定

東芝ソフトウェア技術センターのご紹介

SQiP2013

社内カンパニー・関係会社と連携し、
東芝グループのソフトウェア開発力を強化



1. 背景と課題

- － 背景
- － 進捗管理の課題
- － 課題の根本原因

2. 改善策の検討

- － 既存手法
- － 機能－作業マトリクス
- － 根本原因への対策

3. まとめ

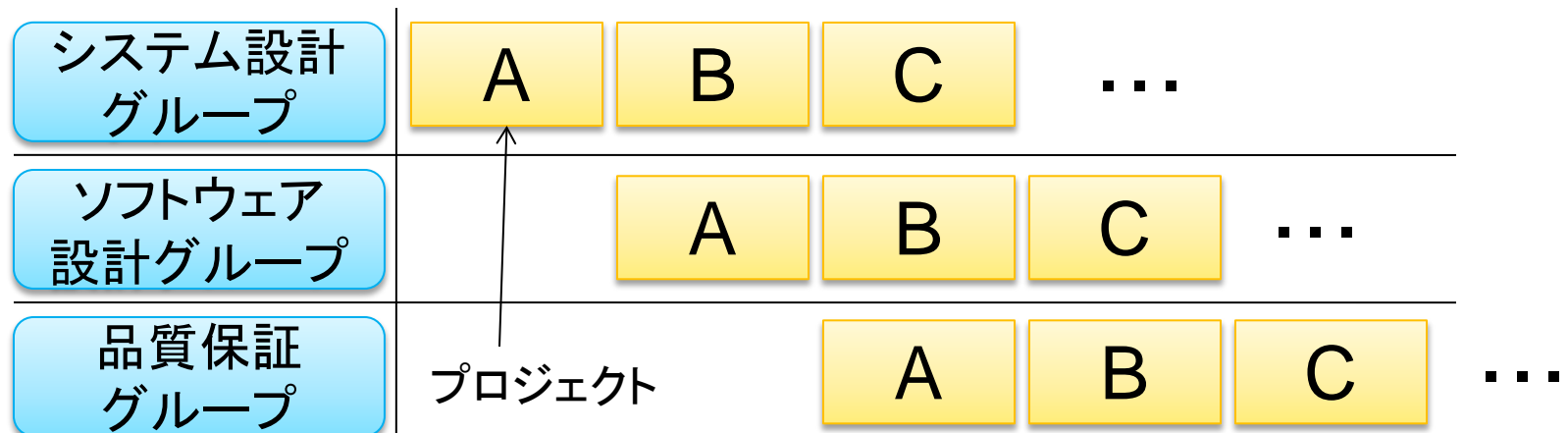
- － 試行評価
- － 課題と今後の予定

- 改善の対象部門について

- 社会インフラ事業の一部門で、組込製品のHW/SW開発を行っている

- 対象部門のプロジェクトの特徴

- 既存製品からの派生開発が多く、小規模なものが多い
- 分業化されており、一つのプロジェクトを複数のグループで行う
 - プロジェクトの責任が工程ごとに引き継がれながら進んでいく
 - 同時並行に多数のプロジェクトが処理されている



並行開発に向いているが、進捗管理面の課題もある

• 現状の進捗管理

- ガントチャートを用いた進捗管理を行っているが、様々な課題が発生し、定着していない



進捗管理作業に
時間がかかる



進捗が感覚と
合っていない

作業者

ガントチャートを見ても
進捗がよくわからない

対策がうまく
取れない

進捗管理作業が
定着しない



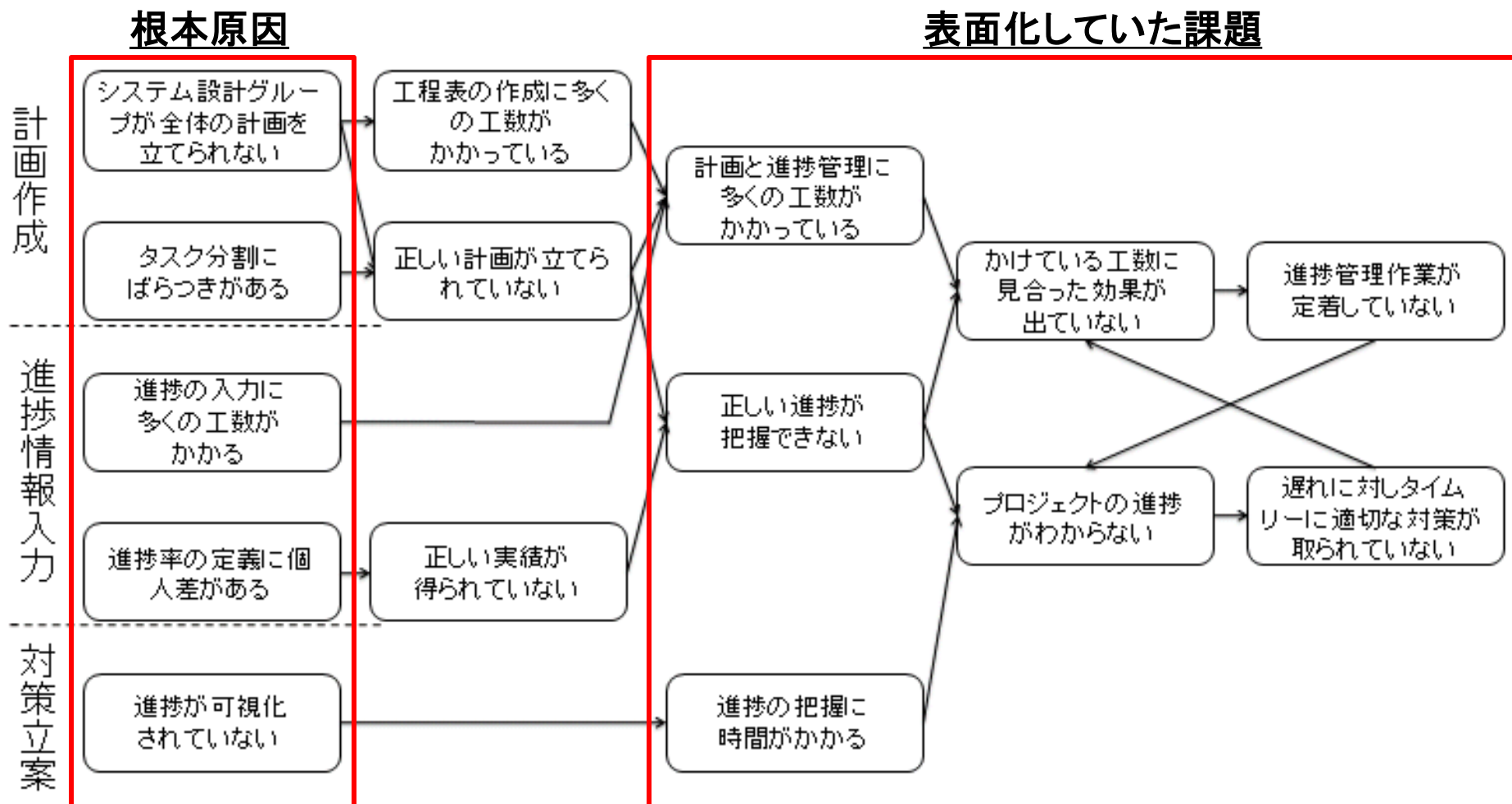
管理者

これらの進捗管理に関する課題を改善する

課題のブレークダウン

SQIP2013

- 課題の因果関係を分析し、根本原因の洗出しを行った



5つの根本原因

SQIP2013

A. 計画立案

- システム設計グループが全体の計画を立てられない
(他のグループの作業内容がよくわからない)
- タスク分割にばらつきがある

対策方針：作業者に依存せず、負荷なく、計画の立案ができる

B. 進捗情報入力

- 進捗の入力に多くの工数がかかっている
- 進捗率の定義に個人差がある

対策方針：正しい進捗を簡単に入力できる

C. 対策立案

- 進捗が可視化されていない

対策方針：プロジェクトの進捗がひと目で分かる

作業者に依存しない簡単で直感的な進捗管理方法が必要

1. 背景と課題

- － 背景
- － 進捗管理の課題
- － 課題の根本原因

2. 改善策の検討

- － 既存手法
- － 機能－作業マトリクス
- － 根本原因への対策

3. まとめ

- － 試行評価
- － 課題と今後の予定

● 調査した手法

- タスク分割方法、進捗の可視化など、根本原因に効果のある手法を調査

● 株式会社 富士通関西システムズ の手法(※)

- 各工程を「準備・実行・レビュー」を基本とする「共通枠」へと分解
 - 右図の「作業内容」
- 各工程で開発する機能ごとに、「作業内容」でブレイクダウンしたものをタスクとして進捗管理

各工程の「作業内容」を固定化
計画立案作業の効率化

	略号	作業内容	
論理設計 (SS)	SSP	論理設計前の準備作業	準備
	SS	論理設計	実行
	SSR	論理設計書のレビュー作業	レビュー
プログラム開発 (PG)	PS	詳細設計作成作業	準備
	PG	プログラミング作業	実行
	PTP	プログラム単体テストの準備	準備
	PT	プログラム単体テスト	実行
結合テスト (QT)	PTR	プログラム単体テスト結果レビュー	レビュー
	ITP	結合テスト準備作業	準備
	IT	結合テスト	実行
システムテスト (ST)	ITR	結合テスト結果レビュー	レビュー
	STP	システムテスト準備作業	準備
	ST	システムテスト	実行
	STR	システムテスト結果レビュー	レビュー

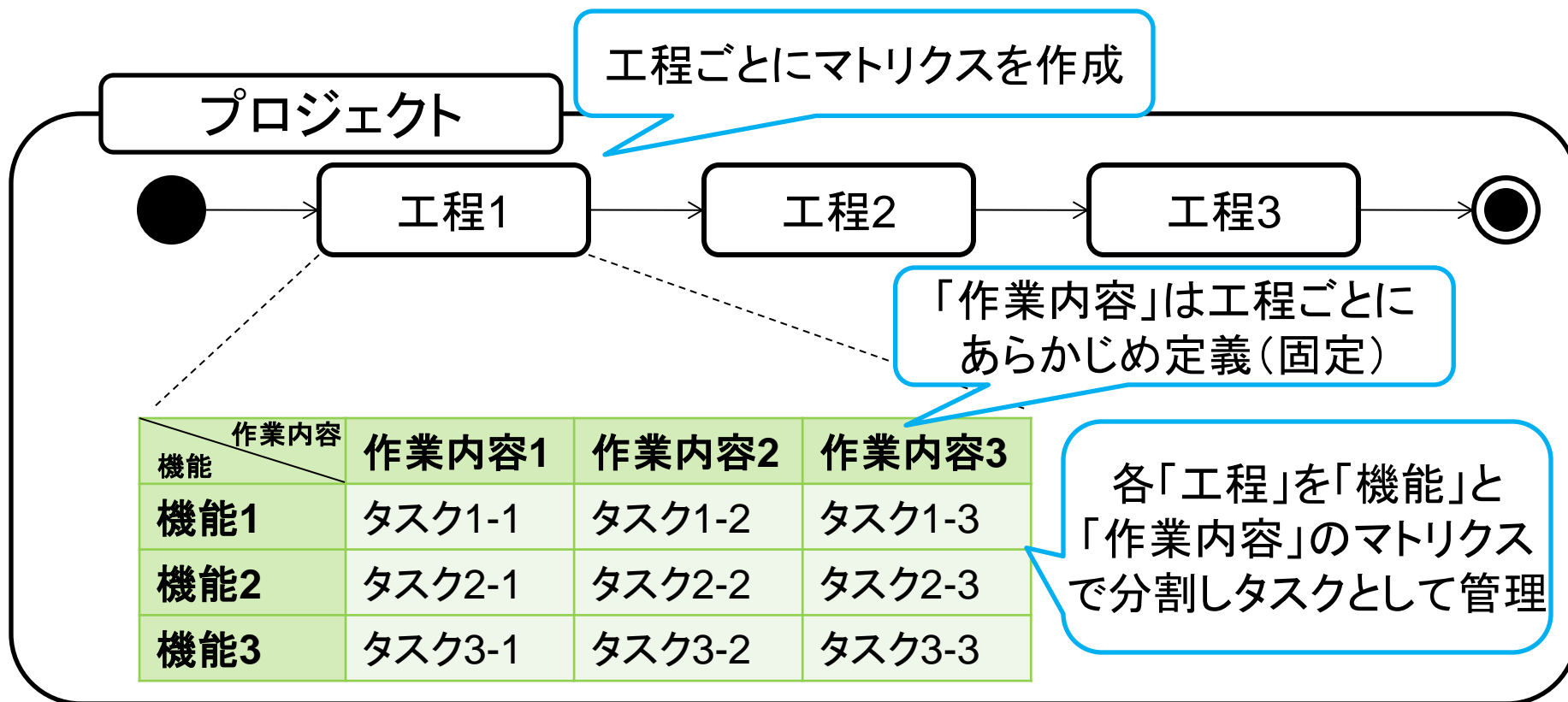
※プロジェクト進捗管理における定量化及び
可視化とそのコントロールについて
武智 英記

Journal of the Society of Project Management Vol6, No.3, 2004

機能－作業マトリクス

SQIP2013

- 株式会社 富士通関西システムズの手法をマトリクス状にした「機能－作業マトリクス」を提案



- この手法をもとに、各「根本原因」への対策を行う

根本原因への対策

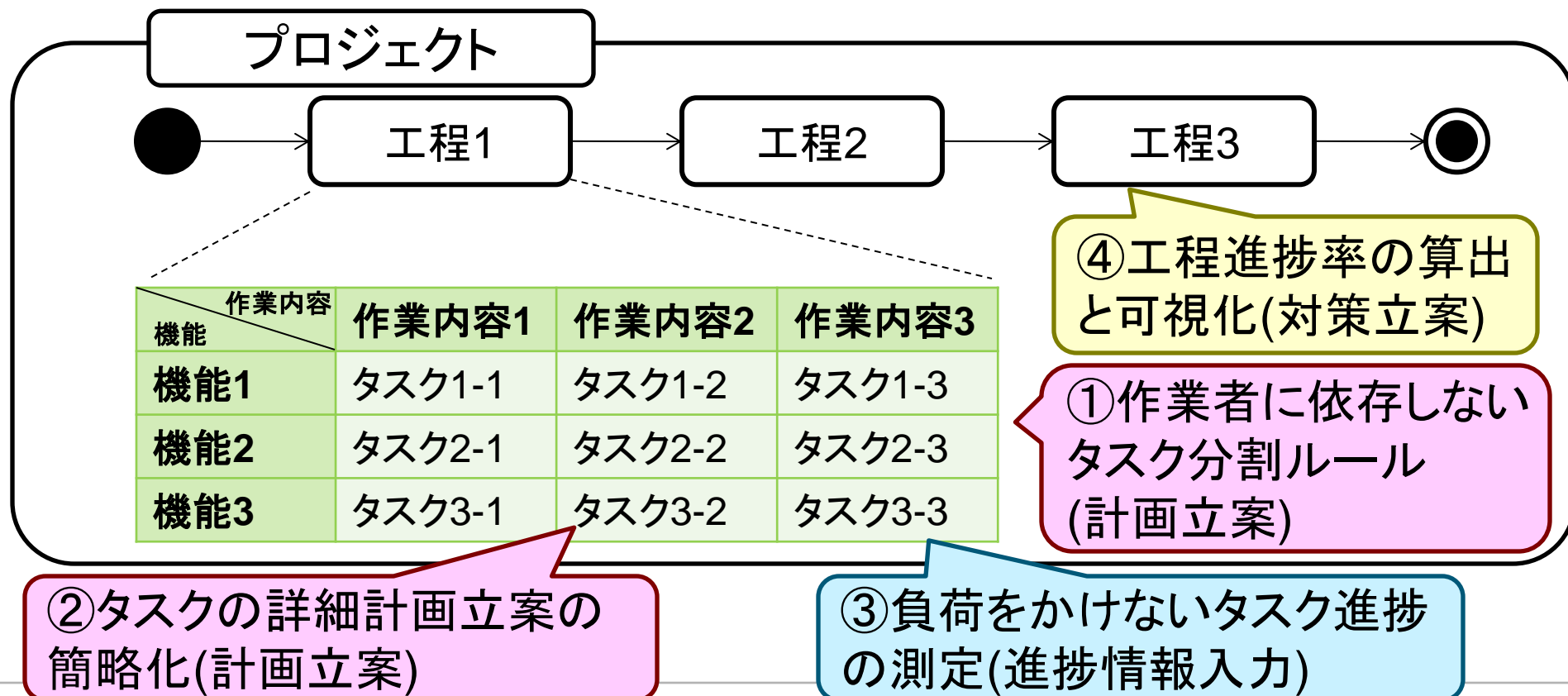
SQIP2013

• 根本原因を解決するために必要な4つの対策

A. 計画立案 : 作業者に依存せず、負荷なく、計画の立案ができる

B. 進捗情報入力 : 正しい進捗を簡単に入力できる

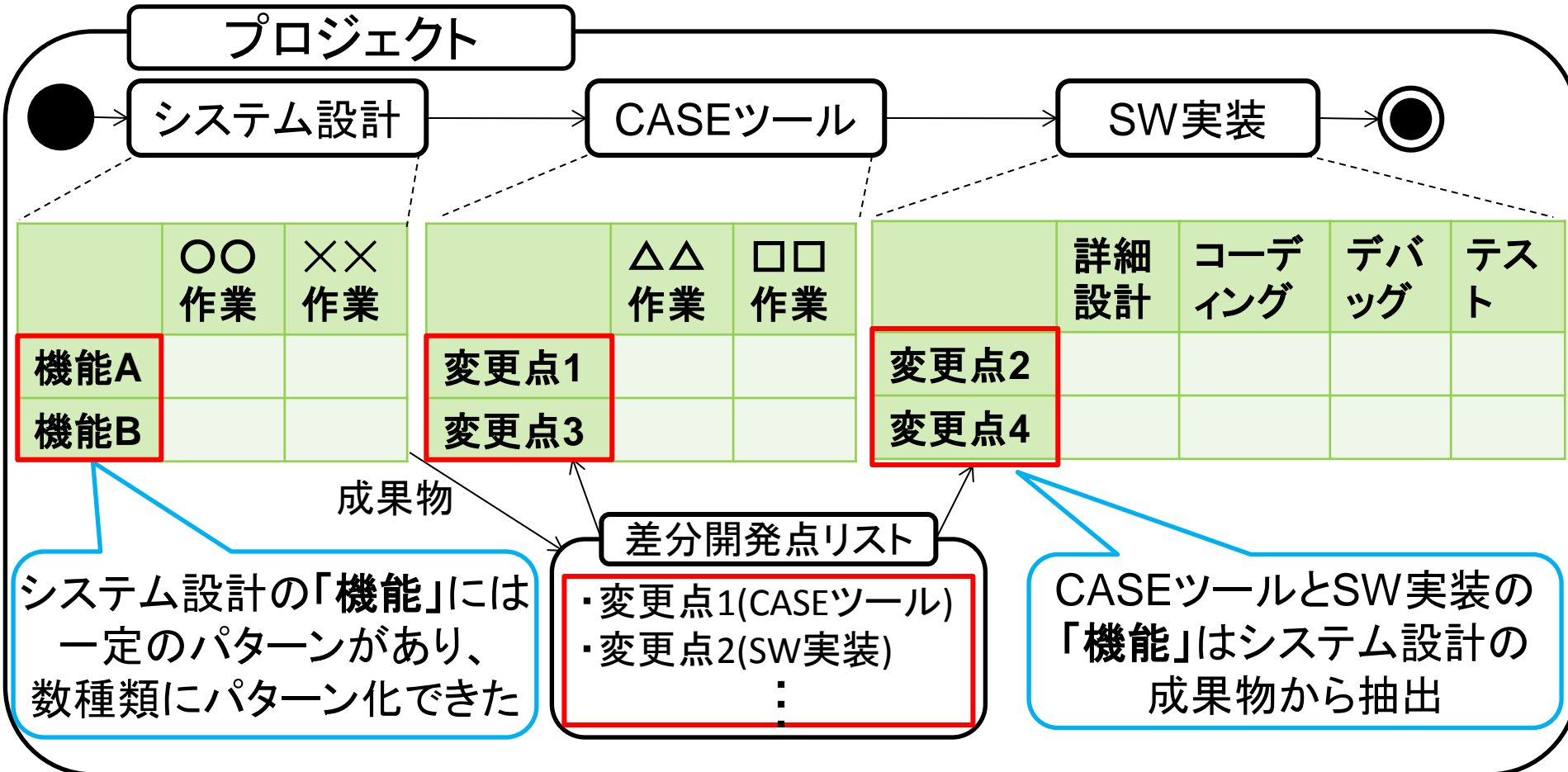
C. 対策立案 : プロジェクトの進捗がひと目でわかる



① 作業者に依存しないタスク分割ルール

・ 解決方法

- － 「機能」の定義付けを行う
- － 作業者に依存せず、一意に決定できるようにする



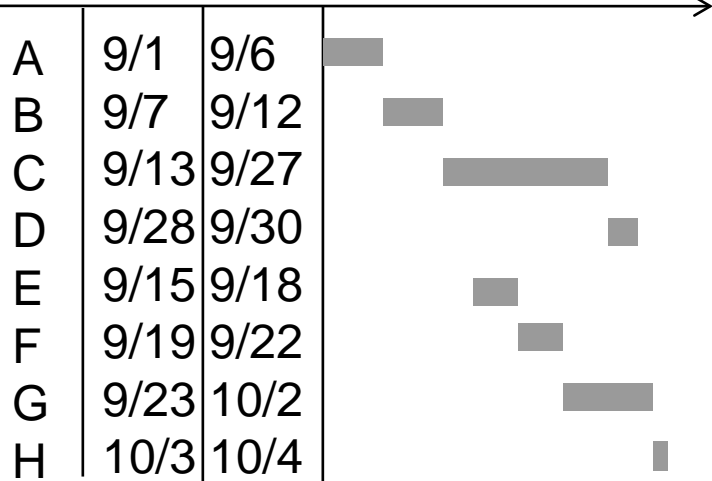
②タスクの詳細計画立案の簡略化

• 解決方法

- 「作業内容」ごとの「重み」と、「機能」毎の開始・終了予定日から各タスクの開始・終了予定日を算出
- 計画の立案に掛かる負荷を軽減

ガントチャート

開始 終了



従来は全タスクの
開始・終了予定を指定

機能－作業マトリクス

	開始	終了	詳細設計	コーディング	デバッグ	テスト
重み			2	2	5	1
機能1	9/1	9/30	A	B	C	D
機能2	9/15	10/4	E	F	G	H

「機能」毎の開始・終了予定と
各「作業内容」の重みから
各タスクの開始・終了予定を算出

	開始	終了
A	9/1	9/6
B	9/7	9/12
C	9/13	9/27
D	9/28	9/30
...

③ 負荷をかけないタスク進捗の測定

- 解決方法

- 各タスクの進捗は完了・未完了の二値で管理
 - 作業者毎の進捗の定義は揃う
 - 作業者の負荷は最小限に抑えられる

- 進捗の精度

- 各タスクの細かい進捗率を測定する手法を比較して精度は下がる
- しかし、比較的小さいプロジェクトが多く、タスクは十分に詳細化されていると考え、今回は作業者の負荷軽減を優先



④ 工程進捗率の算出と可視化

• 解決方法

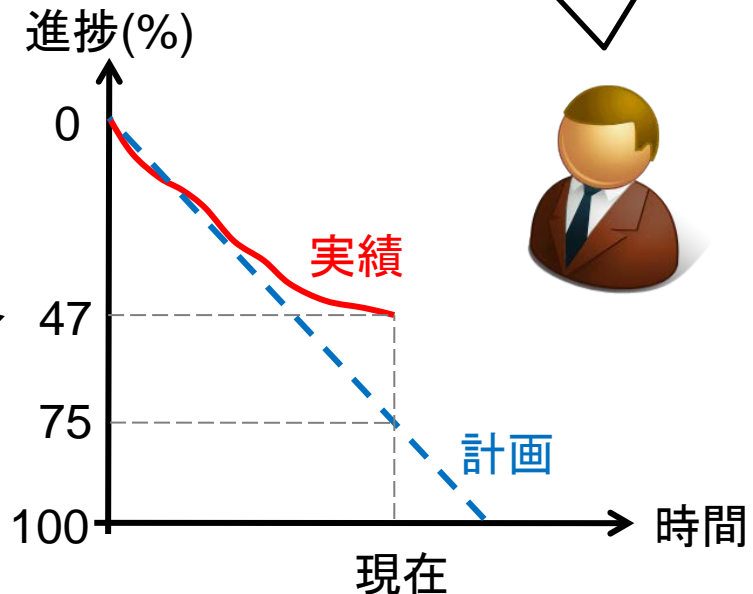
- 前述の「重み」を用いて、工程の進捗率を算出
- バンダウンチャートで日単位の進捗率を可視化
 - 計画は、開始・終了予定日から算出

• 工程進捗率の算出

- 「重み」と「終了状態」から進捗率を算出

	詳細設計	コーディング	デバッグ	テスト
重み	2	2	5	1
機能1	✓	✓	✓	✓
機能2	✓	✓		
機能3				

$$\text{進捗率} = \frac{(2 \times 2) + (2 \times 2) + (5 \times 1) + (1 \times 1)}{(2 \times 3) + (2 \times 3) + (5 \times 3) + (1 \times 3)} = 47(\%)$$



バンダウンチャート



A. 計画立案

- システム設計グループが全体の計画を立てられない
(他のグループの作業内容がよくわからない)
 - タスク分割にばらつきがある
- 計画立案方法を確立した。だれでも**一意の計画が立案できる**ため、
作業者間の計画のバラツキをおさえることができる

B. 進捗情報入力

- 進捗の入力に多くの工数がかかっている
 - 進捗率の定義に個人差がある
- 各タスクの進捗を二値化したことにより、タスクの進捗率を考える必要が
なくなり、最小限の労力で**個人差のない進捗の入力**が可能

C. 対策立案

- 進捗が可視化されていない
- 重みを用いて、工程の進捗率を算出し、バーンダウンチャートによる
可視化を行ったことで、**プロジェクトの情報を簡潔に知る**ことができる

1. 背景と課題

- － 背景
- － 進捗管理の課題
- － 課題の根本原因

2. 改善策の検討

- － 既存手法
- － 機能－作業マトリクス
- － 根本原因への対策

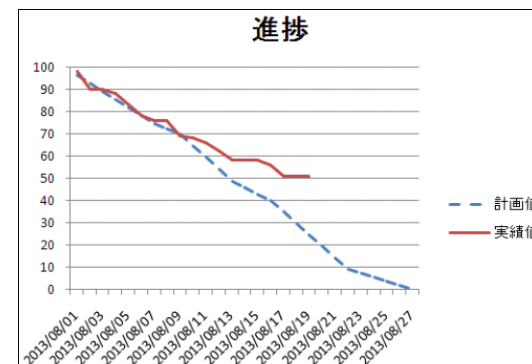
3. まとめ

- － 試行評価
- － 課題と今後の予定

・ システムの実装

－ 上述の機能を持つシステムをExcelマクロを用いて実装

作業項目	開始予定日	終了予定日	詳細設計	コーディング	デバッグ	テスト結果記録
重み			2	2	5	1
変更点1	2013/8/1	2013/8/7	■ 2013/8/2	■ 2013/8/2	■ 2013/8/5	■ 2013/8/7
変更点2	2013/8/1	2013/8/9	■ 2013/8/1	■ 2013/8/2	■ 2013/8/6	■ 2013/8/7
変更点3	2013/8/1	2013/8/9	■ 2013/8/2	■ 2013/8/3	■ 2013/8/9	■ 2013/8/10
変更点4	2013/8/10	2013/8/13	■ 2013/8/9	■ 2013/8/12	■ 2013/8/17	□
変更点5	2013/8/10	2013/8/16	■ 2013/8/12	■ 2013/8/13	□	□
変更点6	2013/8/10	2013/8/16	■ 2013/8/11	■ 2013/8/13	□	□
変更点7	2013/8/17	2013/8/22	■ 2013/8/16	□	□	□
変更点8	2013/8/17	2013/8/22	□	□	□	□
変更点9	2013/8/17	2013/8/27	□	□	□	□
変更点10	2013/8/17	2013/8/27	□	□	□	□



・ 試行結果

－ 本格導入前に、数プロジェクトで先行的に試行を行った



作業員

チェックするだけなので
進捗入力がとても簡単。
感覚では、進捗入力の
工数が1/10程度になっ
たように思える

工程表の作成に
かかる工数が
1/6程度になった

進捗管理の仕組みが
実情に合っている



管理者

- **進捗率の正確さの評価**

- － 負荷軽減のために正確さを排した部分が、進捗率に与える影響の評価
 - 進捗を完了/未完了へ二値化

- **対象部門への定着**

- － 試行を経て、ルール化および教育を行った
- － 今後は定着に向けたサポートや改善につながる機能追加などを行う

- **適用範囲の拡大**

- － 今はシステムとソフトウェアの設計・実装のみを対象にしている
今後はハードウェア設計や品質保証などにも広げたい



ご清聴ありがとうございました