

開発プロジェクトを成功に導く 「開発計画書」とは

室 修治

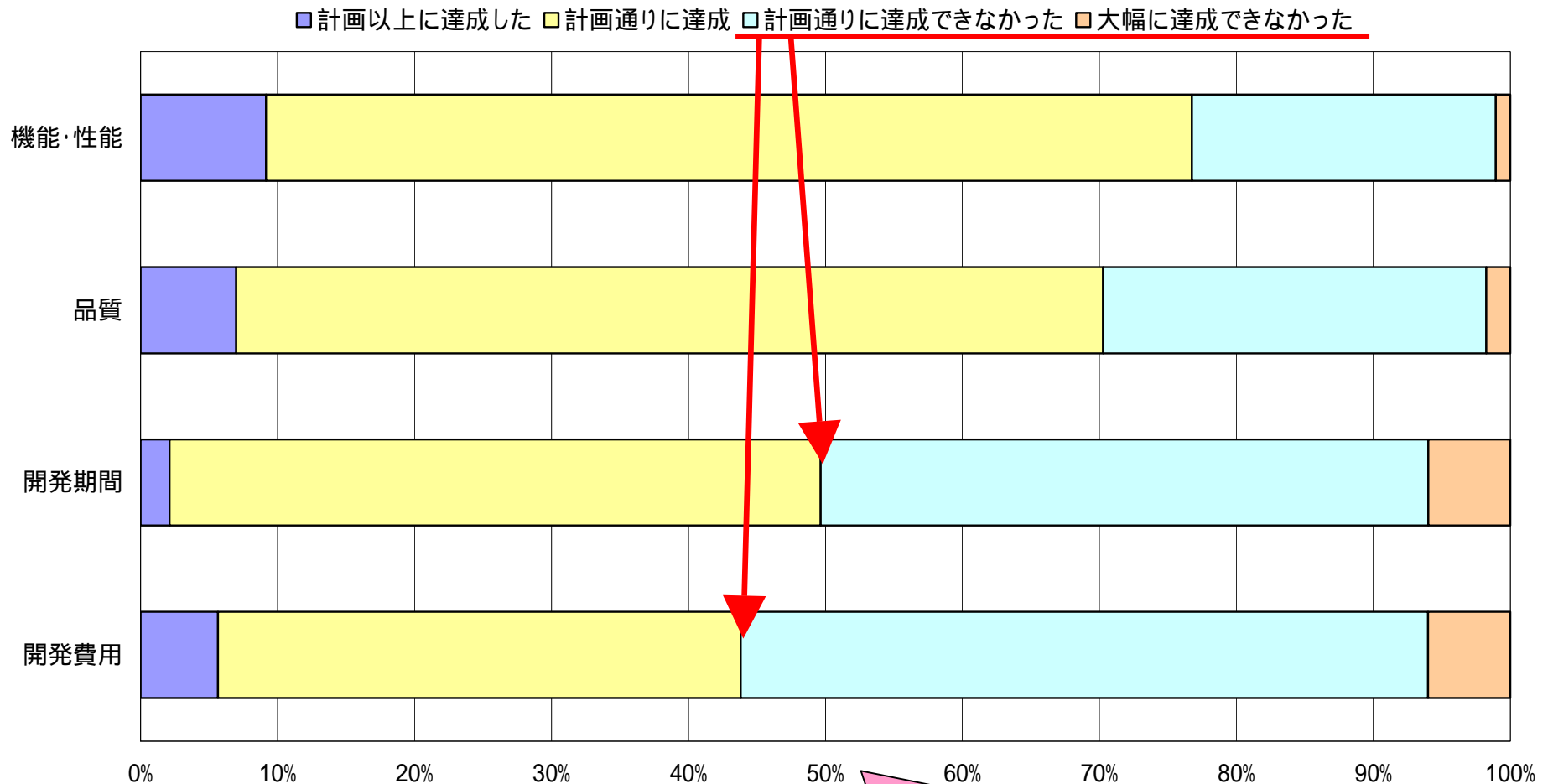
横河デジタルコンピュータ(株)

独立行政法人 情報処理推進機構

ソフトウェア・エンジニアリング・センター (SEC)

2006.5.11

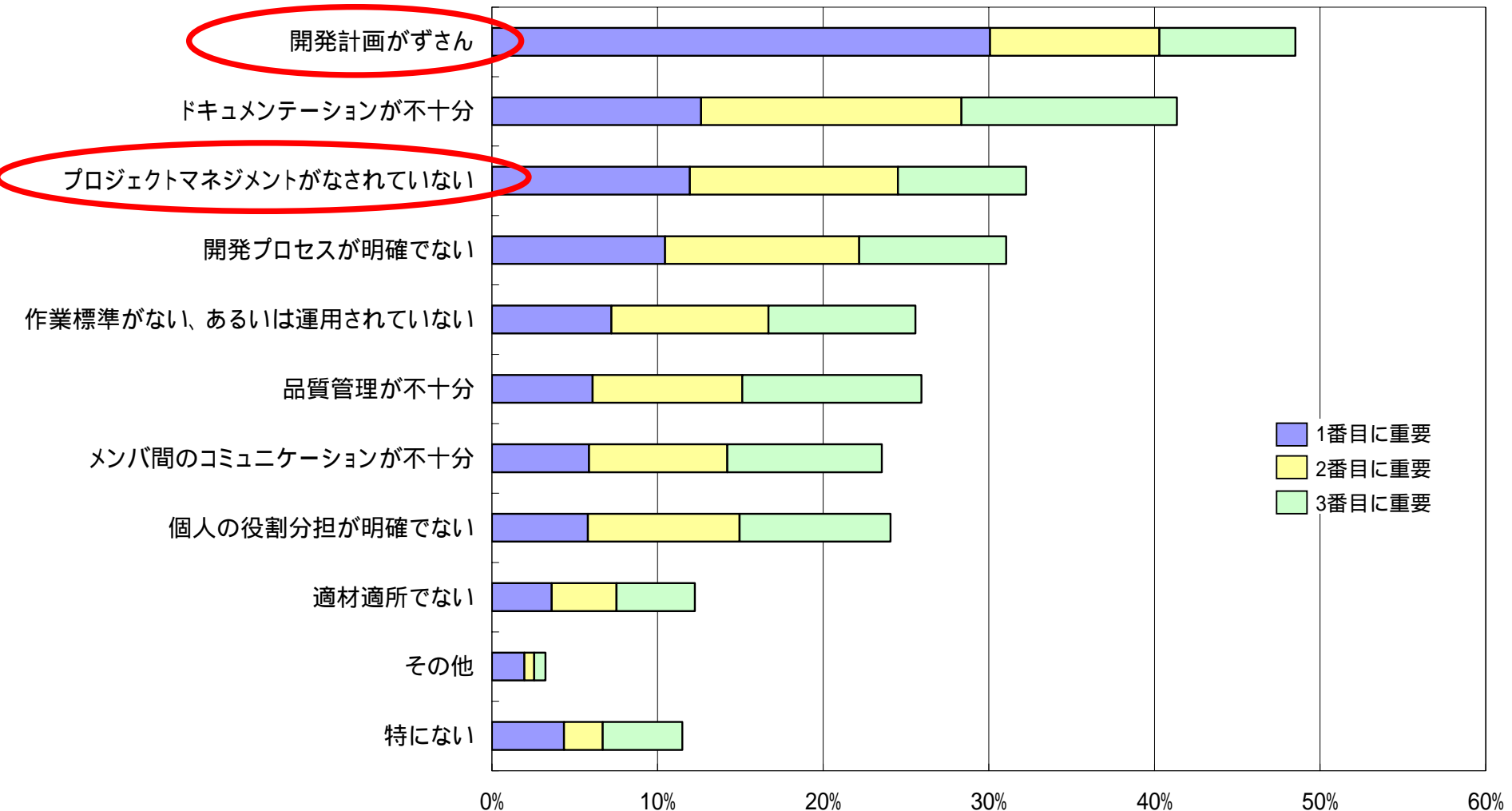
組込みプロジェクトの計画達成状況



期間・費用は半数以上が計画未達！

2005年版組込みソフトウェア産業実態調査：開発プロジェクト責任者向け調査より

開発現場の想い(開発の進め方で課題と感じていること)



2005年版組込みソフトウェア産業実態調査:技術者個人向け調査より

上位要求の納期

実際に必要な作業の積み上げ

溢れ

開発組織

メンバ

作業

作業

作業

作業

作業

作業

メンバ

作業

作業

作業

作業

作業

作業

作業

⋮

メンバ

作業

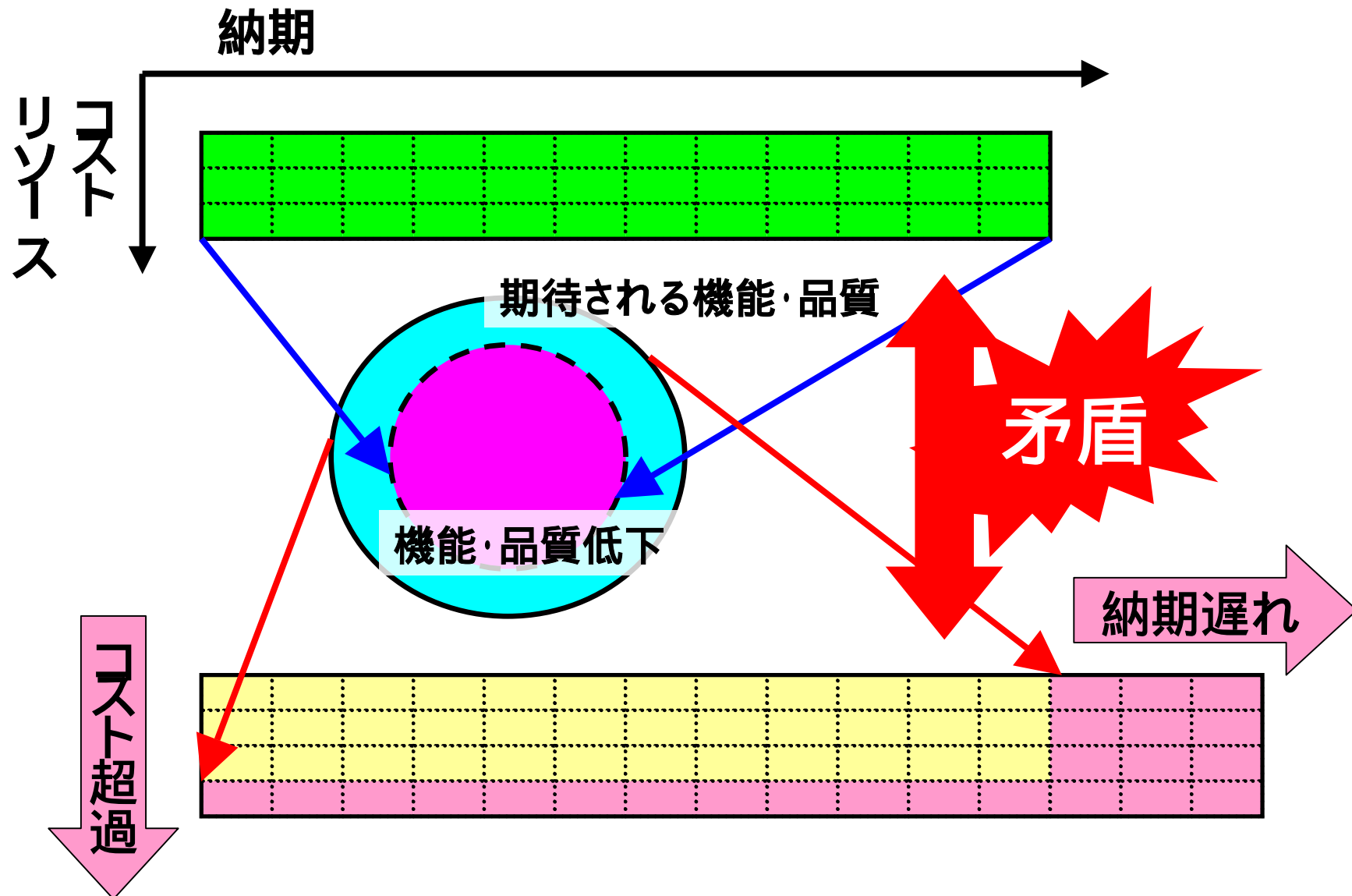
作業

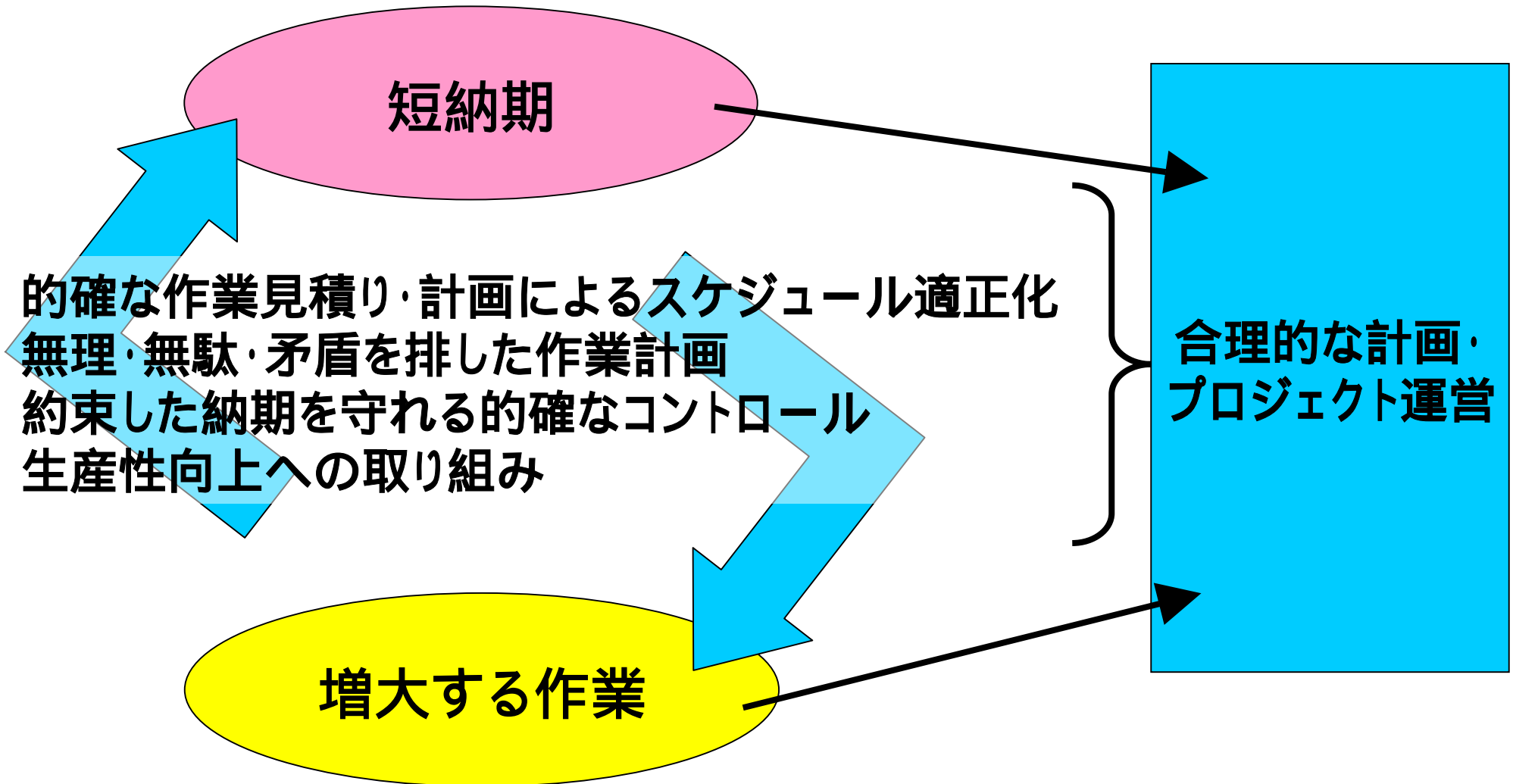
作業

作業

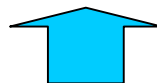
作業

実現性の裏付けなく納期を設定すると

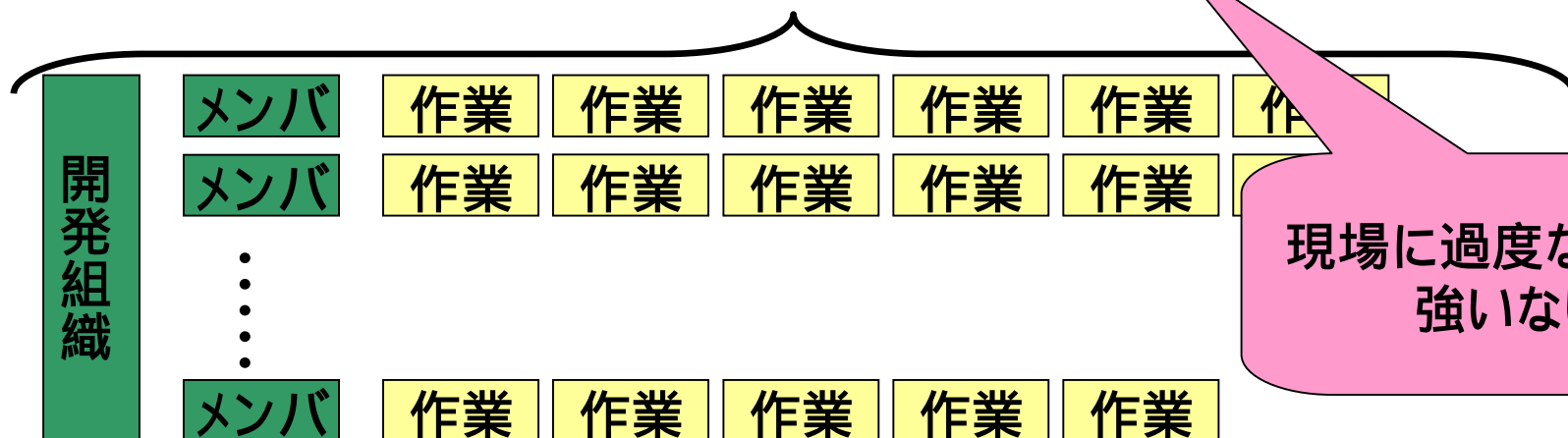




上位要求の実現確率向上



- 精度の高い実現計画
- 効果的な開発プロセス
- 効果的な開発マネジメント
- 生産性向上にも寄与できる品質向上技術



現場に過度な無理を
強いない

組込みソフトウェアの品質・生産性向上

設計品質
向上技術

実装品質
向上技術

プロジェクト
マネジメント
技術

開発
プロセス
技術

組込みソフトウェア設計力強化のための
技術整備と普及

利用品質向上技術

高性能指向実装技術

ガイド：
“規定”でとどまるのではなく適用・定着のための指針までを提供

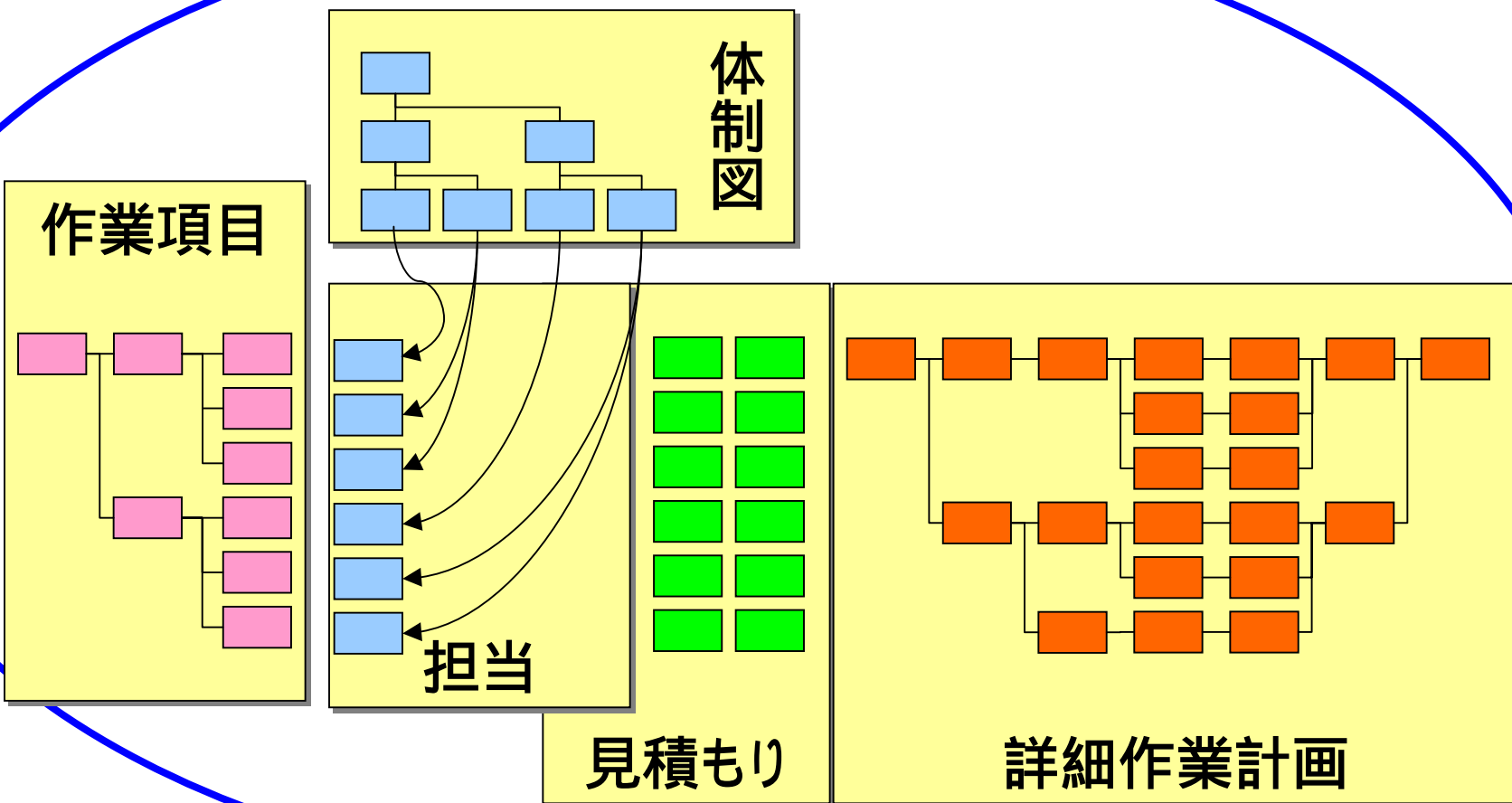
きちんとした計画に基づく
適切なマネジメントの実践

組込みソフトウェア向け
開発計画作成ガイド(V0.8)

ハードウェア開発を考慮した
最適なソフトウェア開発プロセスの実現

組込みソフトウェア向け
開発プロセス標準ガイドの策定

プロジェクトの目的・範囲・目標



『組み込み』ソフトウェアの制約・特徴

競争の激しい市場を対象

ソフトウェアによる付加価値増大が製品のポイント

ソフトウェアの規模
複雑さが増大

多人数開発による
開発プロジェクトの肥大化

仕様を決めきれない

仕様の曖昧さ
開発途中の仕様変更の頻発

様々な周辺領域の技術(知識)が開発に必要

短い製品寿命

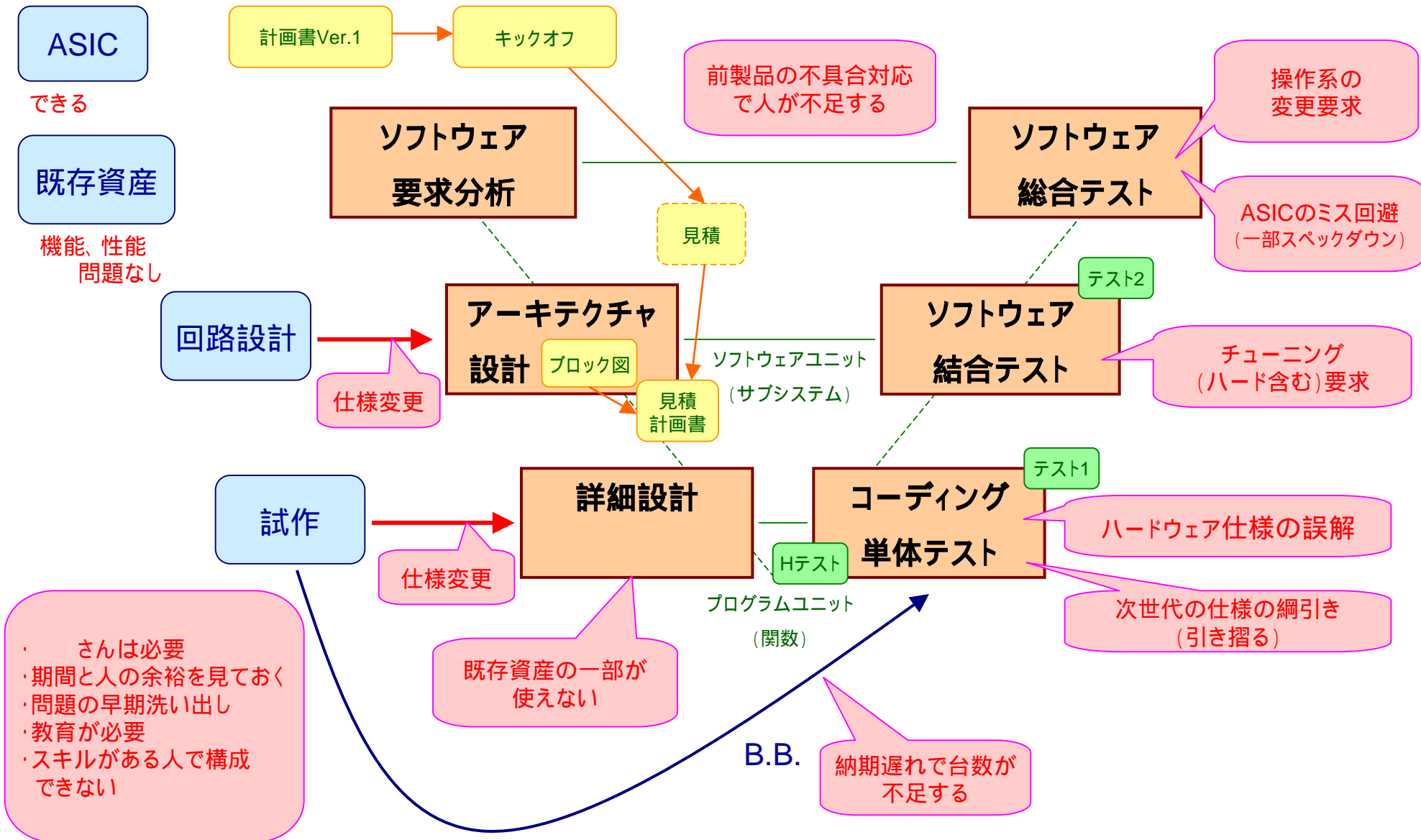
短期開発が必須

HW/SWのコンカレント開発

Evolving型(進化型)開発がベース

機能以外の要素が
製品価値に影響

信頼性の要求
高信頼性が必須
品質・信頼性の要求が曖昧



『組み込み』ソフトウェアの制約・特徴の影響

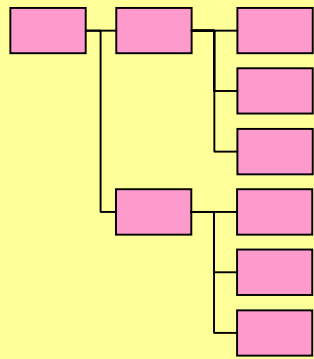
仕様を決めきれない

HW/SWのコンカレント開発

仕様の曖昧さ
開発途中の仕様変更の頻発

作業項目の追加、作業量の変化、やり直し等の発生

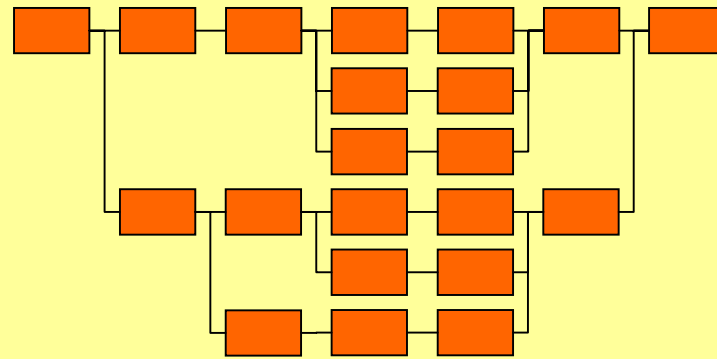
作業項目



見積もり



詳細作業計画



変更対応方法の明確化
↑
HWとのすりあわせ作業の
明確化
↑
作業項目の詳細化

■ 対象読者

- プロジェクトマネージャ(プロジェクトを実践している人)
- 有効な計画書を持たないプロジェクト

■ 狙い

- 参考にできる計画書例。
- 参考にできるテンプレートを用意。
- ベストサンプルとしての運用例を掲載。
- 各項目がなぜ必要であるかその理由も示す。

Part-1: 解説編

プロジェクト計画の基本的な考え方
関連規格など

Part-2: 計画書標準フォーム編

計画書標準フォーム
標準フォーム解説
計画のテクニック

Part-3: 計画チェックリスト編

開発計画チェック項目

Part-4: サンプルプロジェクト例

1. プロジェクトの概要

- 1.1 プロジェクトの目的
- 1.2 プロジェクトの目標
- 1.3 目標達成のための方針・手段
- 1.4 プロジェクトの範囲
- 1.5 プロジェクトの前提条件
- 1.6 プロジェクトの成果物
- 1.7 スケジュールと予算
- 1.8 計画の更新

2. 参照・定義

- 2.1 参照
- 2.2 定義

3. 体制

- 3.1 プロジェクトの体制
- 3.2 外部インタフェース
- 3.3 内部体制
- 3.4 役割と責任

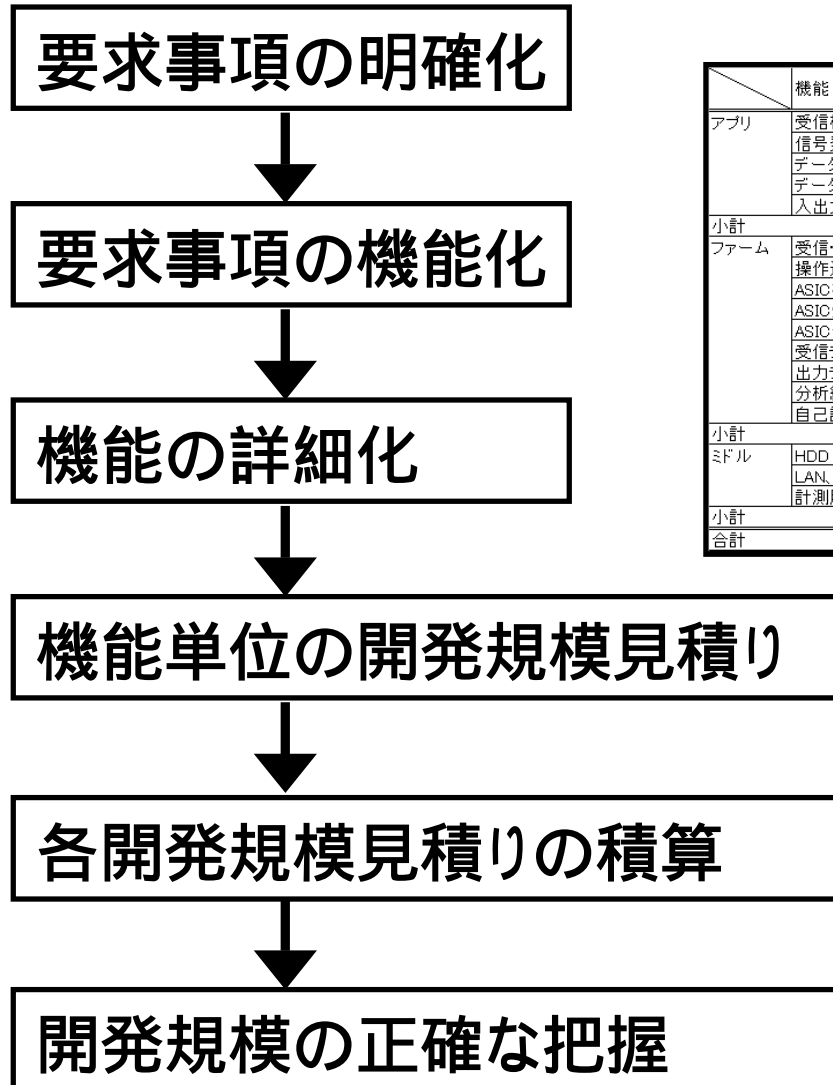
4. スタートアップ計画

- 4.1 見積り計画
- 4.2 人員計画
- 4.3 設備、機器等調達計画
- 4.4 プロジェクトの人員研修計画

5. 作業計画

- 5.1 作業計画
- 5.2 作業項目
- 5.3 スケジュール割当て
- 5.4 人員リソース割当て
- 5.5 予算リソース割当て

- 参考とできるフォーム例
- 記述すべき内容の説明
- 記述の際の注意事項、考慮点
- 作成事例紹介



	機能	サイズ			新規換算	詳細設計	コーディング 単体テスト	結合テスト	総合テスト
		新規	流用	再利用					
アプリ	受信機能設定					xx人日	xx人日	xx人日	xx人月
	信号発生機能設定								
	データ表示機能設定								
	データ編集機能設定								
	入出力設定								
小計									
ファーム	受信・編集データ表示								
	操作追従動作								
	ASIO機能設定								
	ASIO受信動作								
	ASIO信号出力動作								
	受信データ編集								
	出力データ編集								
	分析編集								
小計	自己診断、製造検査プログラム								
ミドル	HDD								
	LAN、USB、シリアル								
	計測用信号入出力								
小計									
合計									

規模 ÷ 生産性 = 作業量

No.	大分類	中分類	小分類 (Ver.2 以降)	生産規模見積り [K LOC]			開発規模 [KLOC]	生産性 & 開発工数	
				$\alpha = 1.0$	$\beta = 0.5$	$\gamma = 0.1$		生産性 [KLOC/人月]	工数 [人月]
				新規	流用	再利用			
1	アプリ	受信機能設定	---	1	1		2	1.43	1.05
		信号発生機能設定	---	1	1		2	1.43	1.05
		データ表示機能設定	---	2	1		3	1.43	1.75
		データ編集機能設定	---	4	1		5	1.43	3.15
		入出力設定	---	2	1		3	1.43	1.75
							0		
							0		
		小計		10	5	0	13	1.43	8.75
			生産規模 :	15					

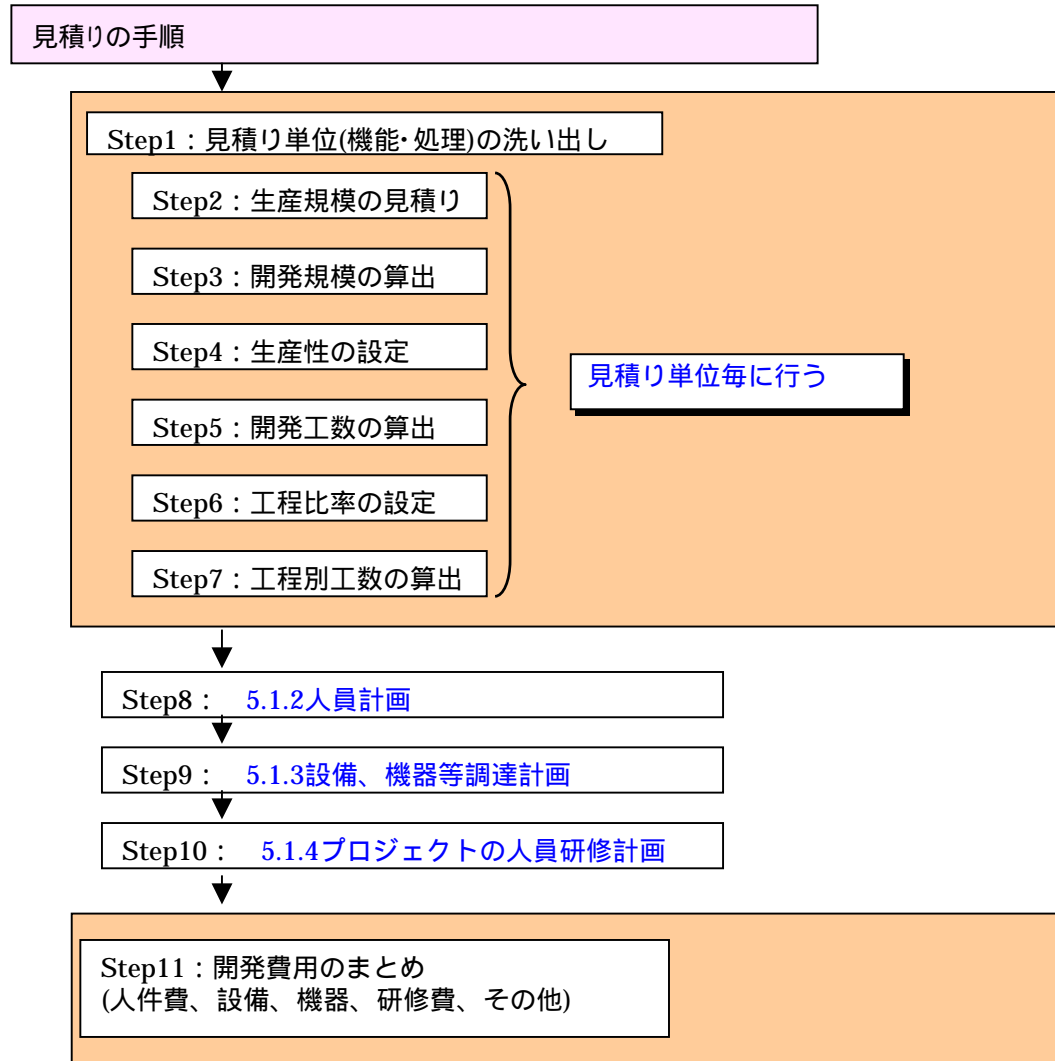
No.	チーム	職種	スキルレベル	アクティビティ	技術要素
1		プロジェクトマネージャ	レベル4 最上級	ソフトウェア要件定義 ～ ソフトウェア・アーキテクチャ設計	
2					
6					
7	Aチーム	ソフトウェアエンジニア	レベル3 上級	ソフトウェア詳細設計 ～ 実装及び単体テスト	
.....					
	合計	---	---	---	---

計画書作成のポイント、計画書標準フォーム紹介

プロジェクト名											承認		立案													
【概算 開発規模見積り】 Ver.1																										
No.	大分類	中分類	小分類 (Ver.2 以降)	生産規模見積り [K LOC]			開発 規模 [KLOC]	生産性 & 開発工数		流用元	ソフトウェア 要件定義		ソフトウェア アーキテクチャ 設計		ソフトウェア 詳細設計		実装及び 単体テスト		ソフトウェア結合 及び結合テスト		ソフトウェア 総合テスト		備考			
								生産性 [KLOC/ 人月]	工数 [人月]		工数 [人月]	工程 比率	工数 [人月]	工程 比率	工数 [人月]	工程 比率	工数 [人月]	工程 比率	工数 [人月]	工程 比率						
				= 1.0 新規	= 0.5 改造	= 0.1 流用																				
1							0																			
							0																			
							0																			
							0																			
							0																			
							0																			
							0																			
							0																			
							0																			
							0																			
小計				0	0	0	0	0.00	0.00	---	0.00	0.0%	0.00	0.0%	0.00	0.0%	0.00	0.0%	0.00	0.0%	0.00	0.0%				
生産規模 :				0																						
2							0																			
							0																			
							0																			
							0																			
							0																			
							0																			
							0																			
							0																			
							0																			
							0																			
小計				0	0	0	0	0.00	0.00	---	0.00	0.0%	0.00	0.0%	0.00	0.0%	0.00	0.0%	0.00	0.0%	0.00	0.0%				
生産規模 :				0																						
3							0																			
							0																			
							0																			
							0																			
							0																			
							0																			
							0																			
							0																			
							0																			
							0																			
小計				0	0	0	0	0.00	0.00	---	0.00	0.0%	0.00	0.0%	0.00	0.0%	0.00	0.0%	0.00	0.0%	0.00	0.0%				
生産規模 :				0																						
合計				0	0	0	0	0.00	0.00	---	0.00	0.0%	0.00	0.0%	0.00	0.0%	0.00	0.0%	0.00	0.0%	0.00	0.0%				
生産規模 :				0																						

開発規模 = × 新規 + × 改造 + × 流用
生産規模 = 新規 + 改造 + 流用

計画書作成のポイント、計画書標準フォーム紹介



記述の際に注意 / 考慮すべき内容

(1) 開発規模見積り

- ・類似プロジェクトでの生産規模、開発規模や見積り根拠を確認しておく。
- ・類似プロジェクトや社内標準ライブラリの改造や流用が出来ないかを検討する。但し、改造の場合は、改造が入る事により生産性が低下し、流用の場合は、改造が無いがテストは必要である事を認識する。また、開発ドキュメントの有無も確認しておく。
- ・改造する場合は、ベースとなるソースにどの位の改造が必要かを見極めること。
- ・開発規模を縮小することが可能な、市販ツールの利用を考慮する。
- ・類似プロジェクトの見積り担当者とのレビューやアドバイスを受けること。
- ・人員計画に基づく工数と開発規模見積りで算出された工数の差異および妥当性を確認し、整合性を保つように見直しする。
- ・プロジェクトの目的を達成するための困難度、ハードウェア開発遅延、ハードウェアとの擦りあわせ結果によるソフトウェアによる実現部分の増加、・・・等々が想定できる場合は、規模、生産性や係数にてリスクヘッジすること。
- ・今後のプロジェクトの見積りをスムーズに行うためにも、開発規模の算出根拠を蓄積することが重要である事を認識する。

■ WBS (Work Breakdown Structure)

- プロジェクト全体を詳細な作業に分割する手法
- プロジェクトに関する全作業を洗い出し詳細化するため、プロジェクトの範囲(スコープ)を確認するためにも使用される

■ PERT (Program Evaluation and Review Technique)

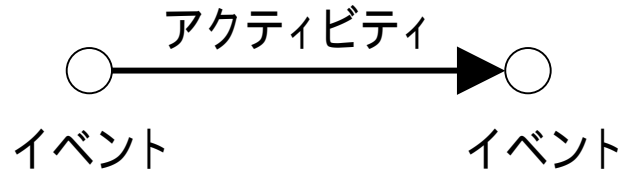
- WBSで洗い出された作業を組み合わせ、プロジェクトを与えられた資源で最短に終了させるためのスケジュールを立案する手法
- オペレーションズ・リサーチと呼ばれる数理的アプローチから生まれ、ネットワークを利用してプロジェクト計画を数值的に分析できる

プロジェクトマネジメント:WBS

作業項目			作業内容	作業規定	開始条件		終了条件		優先度		確定 度合い
ID: 作業名称					入力	前作業 項目	成果物	検証方法	項目 自身	後作業 項目	
A:											
A1:											
A1.1:											
A1.1.1											
A1.1.2											
A1.1.3											
A1.1.4											
A1.1.5											
A1.2:											
A1.2.1											
...											
A1.2.4											
A1.2.5											
A1.3:											
A1.3.1											
...											
A1.3.5											

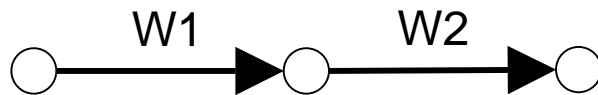
作業の詳細化

アローダイアグラムの例



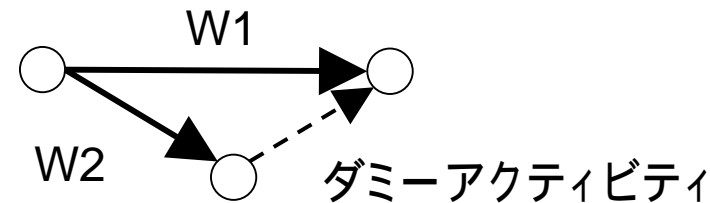
基本

- ・ 作業はアクティビティと呼ばれ矢印で表現する
- ・ 開始と終了はイベントと呼ばれ丸印で表現する



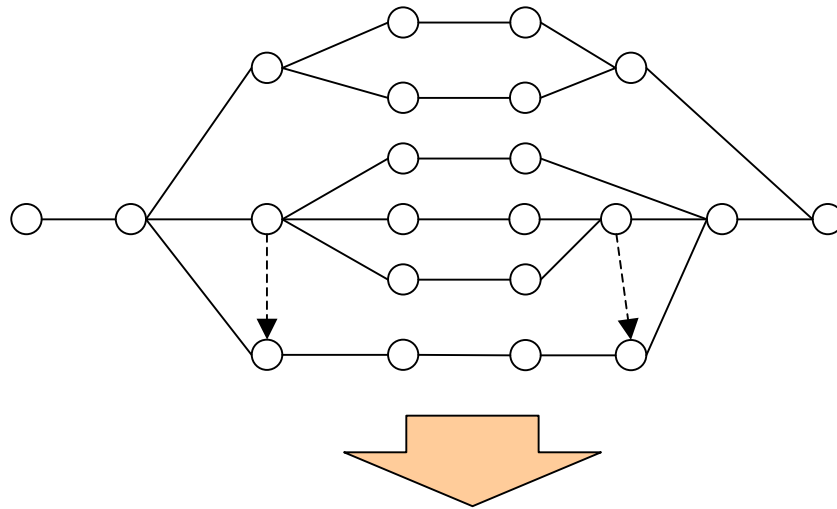
直列作業

- ・ W2はW1が完了しないと着手できないことを示す

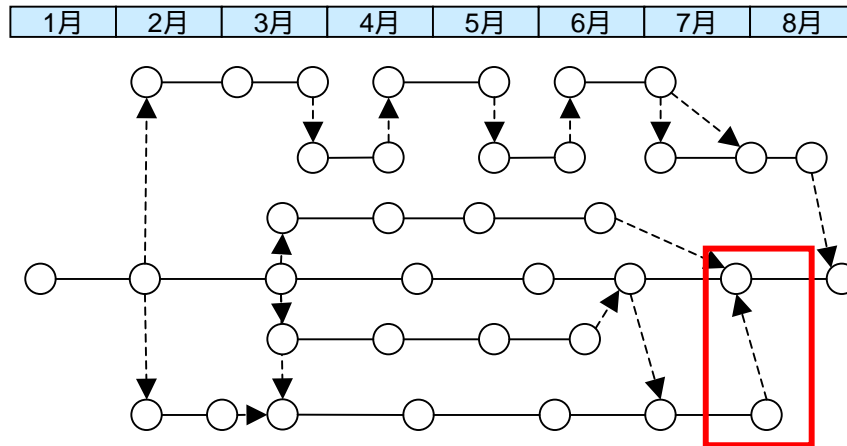


並列作業

- ・ W1とW2は並行して作業できることを示す
- ・ 破線はダミーアクティビティと呼ばれ期間ゼロで作業の順だけを示す



まず、WBSからみた理想的な
(期間が最短になる)順序を
整理してみる



作業順序の逆転

その後、担当者を割り当て、
必要期間からカレンダー日程に
置き換える
このとき、作業順序の逆転が
起こっていないかチェックする
また、クリティカルパスに着目し、
積極的に人材を登用することで、
期間短縮を行う

作業の明確化・詳細化の例(開発規模見積り)

【開発規模見積り】

No.	大分類	中分類	小分類	規模 [K LOC]	算出根拠	
1	新規	〇〇		1,000		
		〇〇		600		
		〇〇		1,500		
		〇〇		700		
		〇〇		2,000		
		小計		5,800		
2	流用(要改造)	△△		100	15.0%	・・・プロジェクト・・・機能の・・・処理を流用
		△△		200	30.0%	・・・プロジェクト・・・機能の・・・処理を流用
		小計		300	改造率	
3	再利用	××		100	社内標準ライブラリを利用	
		××		300	・・・プロジェクト・・・機能を再利用	
		××		500	・・・プロジェクト・・・機能の・・・処理を再利用	
		小計		900		
		合計		7,000		

流用ソースがどの位、
改造が必要かの目安を示す

作業の明確化・詳細化の例(役割分担表)

担当者と役割

担当者	役割名	役割
個人名	プロジェクトマネージャ	開発全体の技術リーダー。 開発プロジェクト全体のプロジェクトマネジメントを行う。 上位マネジメント、企画部門、製造部門、SQA部門との連携を行う。
個人名	OS・DD開発サブチームのリーダー	OS・DD部の技術リーダー。 メカ部門とエレキ部門との連携を行う。 サブチーム内の派遣社員管理を行う。
個人名	ミドル開発サブチームのリーダー	ミドルウェア部の技術リーダー。 買入れするミドルウェアの選定や導入を行う。 サブチーム内の派遣社員管理を行う。
個人名	アプリ開発サブチームのリーダー	アプリケーション部の技術リーダー。 請負会社との連携や管理を行う。
個人名	OS・DD開発サブチームメンバー	OS・DD部の開発を行う。
個人名	OS・DD開発サブチームメンバー	OS・DD部の開発を行う。
個人名	OS・DD開発サブチームメンバー	OS・DD部の開発を行う。
個人名	OS・DD開発サブチームメンバー	OS・DD部の開発を行う。
個人名	OS・DD開発サブチームメンバー(派遣)	OS・DD部の開発を行う。
個人名	OS・DD開発サブチームメンバー(派遣)	OS・DD部の開発を行う。
個人名	ミドル開発サブチームメンバー	ミドルウェア部の開発を行う。
個人名	ミドル開発サブチームメンバー	ミドルウェア部の開発を行う。
個人名	ミドル開発サブチームメンバー	ミドルウェア部の開発を行う。
個人名	ミドル開発サブチームメンバー(派遣)	ミドルウェア部の開発を行う。
個人名	ミドル開発サブチームメンバー(派遣)	ミドルウェア部の開発を行う。

支援プロセスの担当者と役割

担当者	役割名	役割
個人名	構成管理	
個人名	検証と有効性確認	
個人名	文書化	
個人名	品質保証	
個人名	レビューと監査	
個人名	問題解決	
個人名	外注管理	
個人名	プロセス改善	

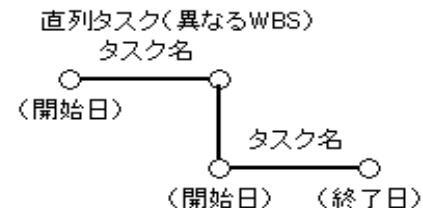
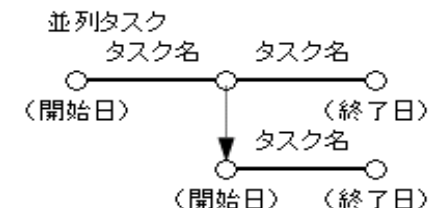
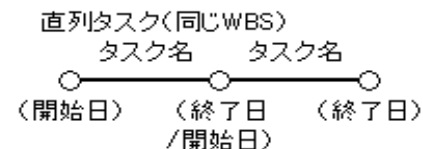
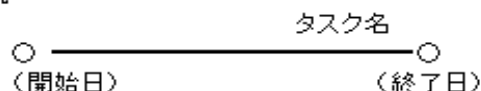
作業の明確化・詳細化の例(スケジュール割当て)

5. 2. 2スケジュール割当て

WBS				TRM			見積もり												
番号	第1レベル	第2レベル	第3レベル	担当	照査	認可	LOC/頁数	1週	2週	3週	4週	5週	6週	7週	8週	9週	10週	11週	12週
Ext.	外部マイルストーン										▲			▼					
											リリース			受け取り					
s1.0	WBS名			名前	名前	名前	??頁	○	作業名	○									
m1.0	プロジェクトマネジメント																		
		プロジェクト計画書作成																	
		プロジェクト終了報告書作成																	

凡例

基本表現



5.2.2 スケジュール割当て

- 担当部分以外のレビューに必要な工数も考慮して計画する。
- 長期休暇など、長期に作業できない期間を考慮しておく。
- PERTの作成後に机上で作業をシミュレーションし、無理のない作業計画とする。
- PERTのシミュレーションにより、計画上のリスクを特定しておく。
- リスクの発生確率を軽減するために必要な作業も計画書に含めておく。
- 外部依存関係(ハードウェアの受け入れ遅れ)など、十分に軽減することのできないリスクは、Contingencyを事前に計画しておく。
- 仕様を決める際に必要な擦り合わせ期間をレビュー期間として計画しておく。
- ハードウェアとの結合は、WBSごとに結合の順番を十分考慮して計画する。

(1) プロジェクトの立ち上げ

- ・製品企画
- ・技術的実現性の判断
- ・戦略的市場投入時期決定(経営的)
- ・開発に係わる組織リソースの対応可能性判断

プロジェクト立ち上げの決定

(2) プロジェクトの発足

- ・プロジェクトマネージャの任命
- ・プロジェクトの登録
- ・プロジェクトチーム主要メンバーの決定

開発計画書
暫定版作成

(3) 具体的見積りの開始

- ・プロジェクトチームでの具体的開発作業を開始できるまでの分析・設計

開発計画書
第1版作成

ベースライン化

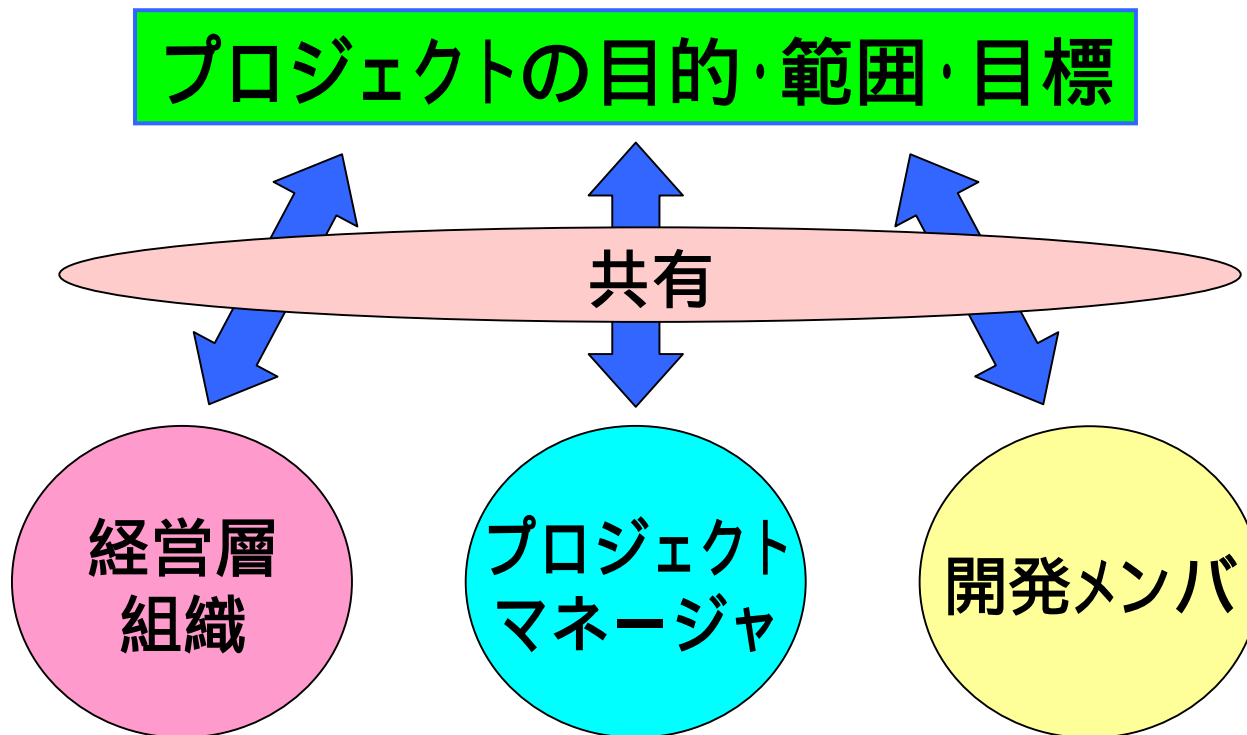
(4) 具体的開発作業の開始

- ・プロジェクトチームメンバーの決定

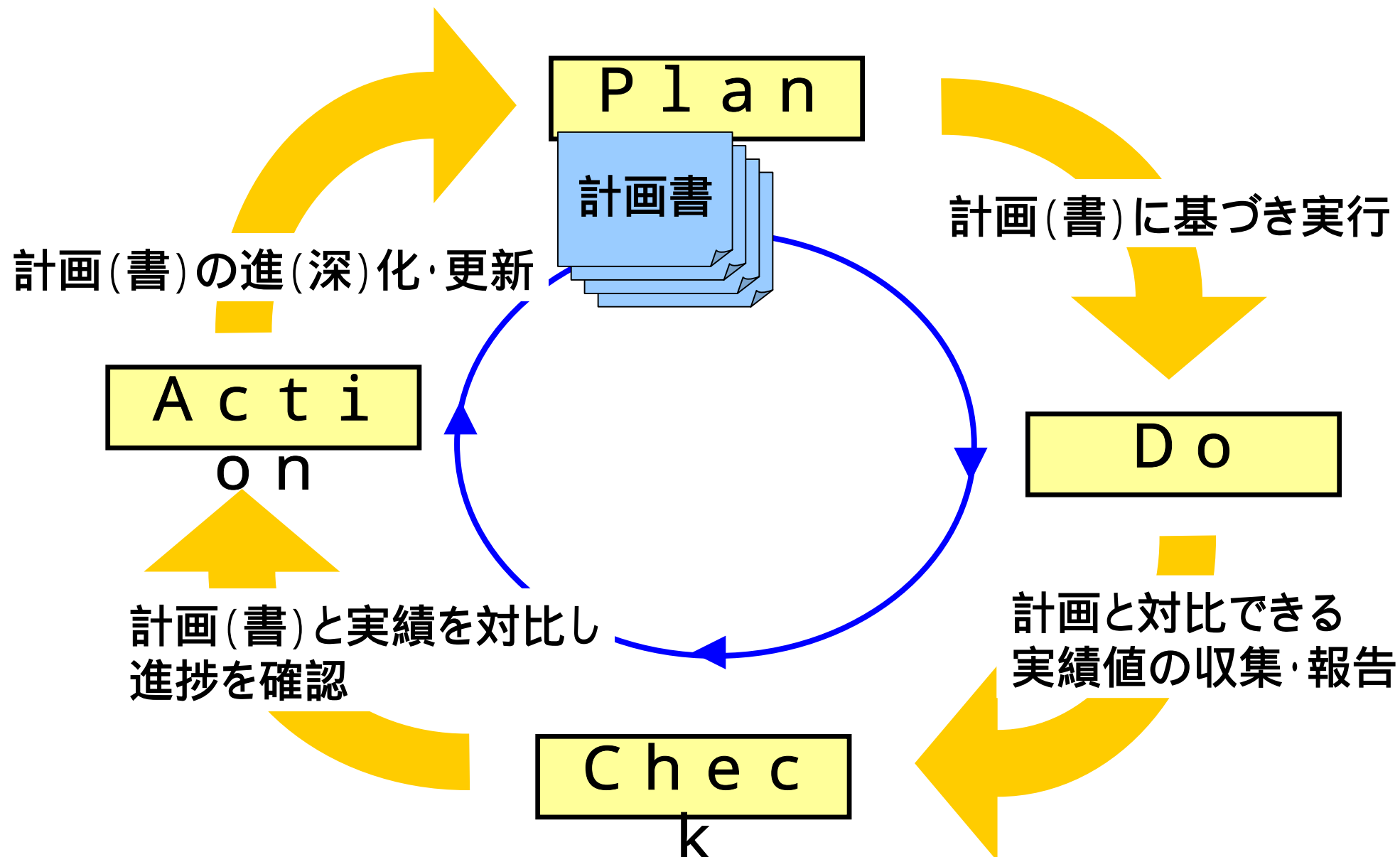
具体化 詳細化は
一気にできないのが普通

プロジェクトの立ち上げルール、開発計画作成、プロジェクト開始条件は組織により異なる(組織によるルール化が必要)

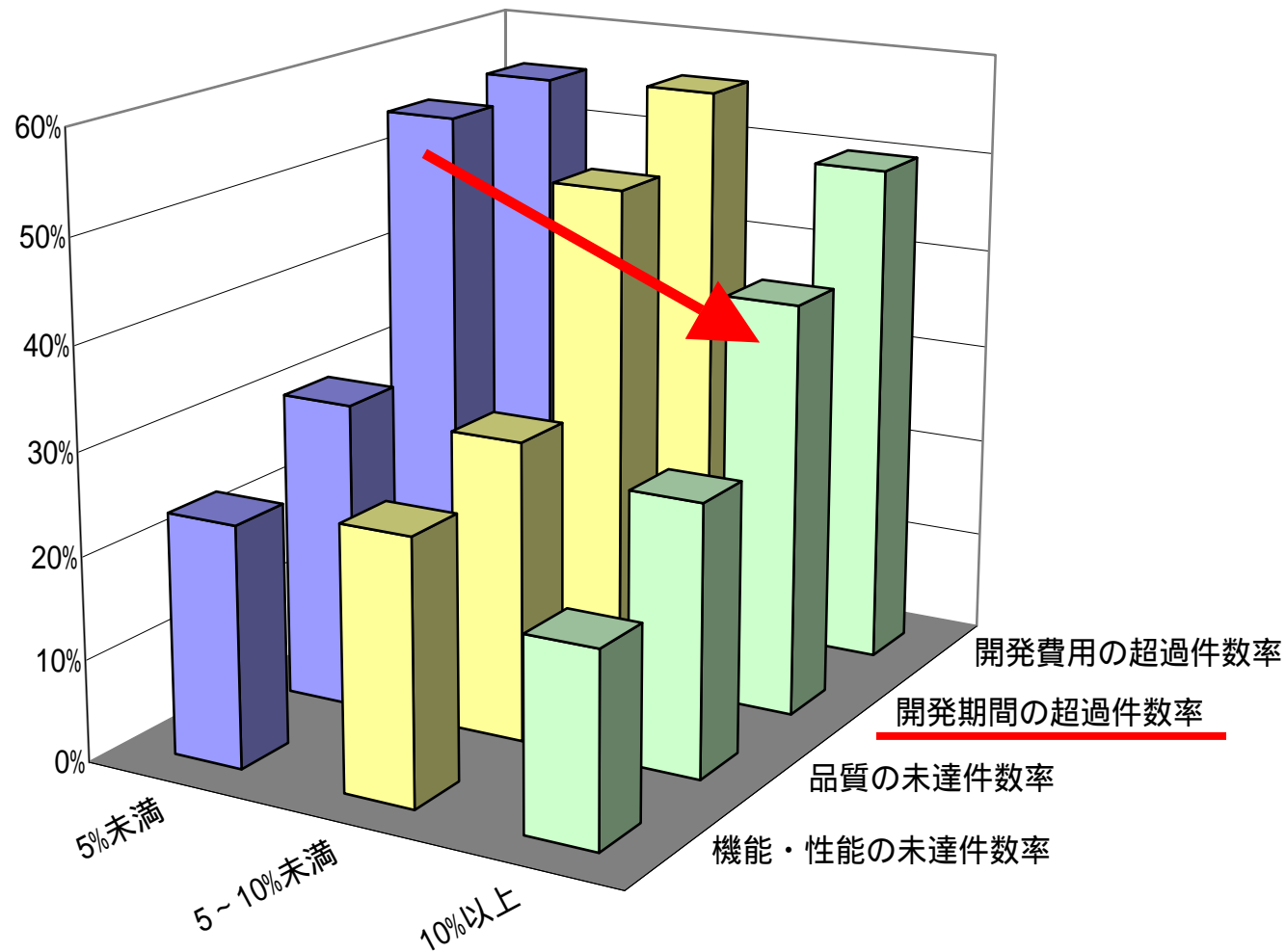
全ての「実行」、「コントロール」、「評価」の源



特に問題発生時のジャッジメント、コントロールの一貫性に必須

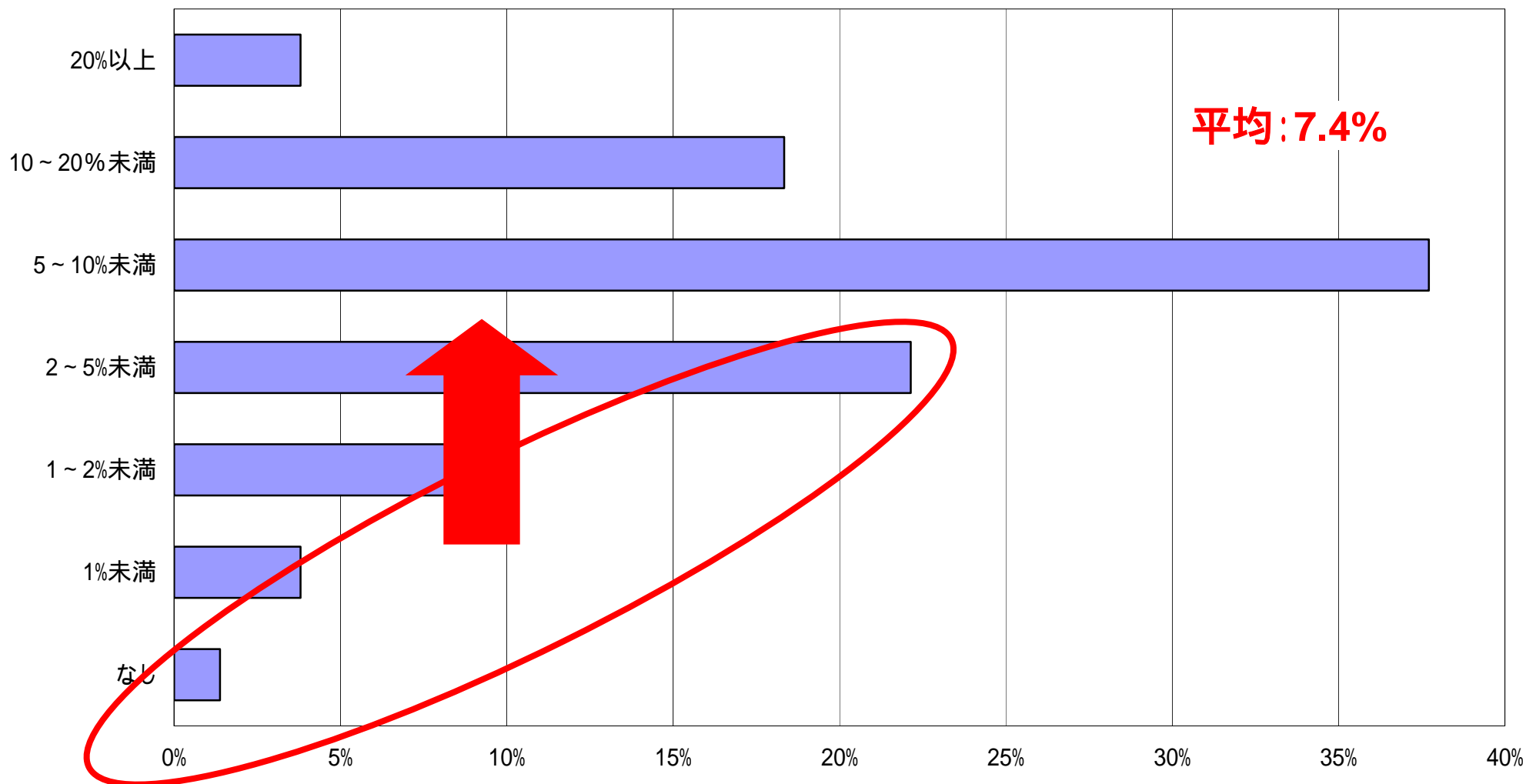


プロジェクトマネジメントの工数比率と プロジェクトの計画未達成率



2005年版組込みソフトウェア産業実態調査: 開発プロジェクト責任者向け調査より

プロジェクト全体工数のうちプロジェクトマネジメントのためにかけられた割合



2005年版組込みソフトウェア産業実態調査: 開発プロジェクト責任者向け調査より

ご清聴ありがとうございました
IPA / SECの活動にご協力をお願い致します

<http://sec.ipa.go.jp/>