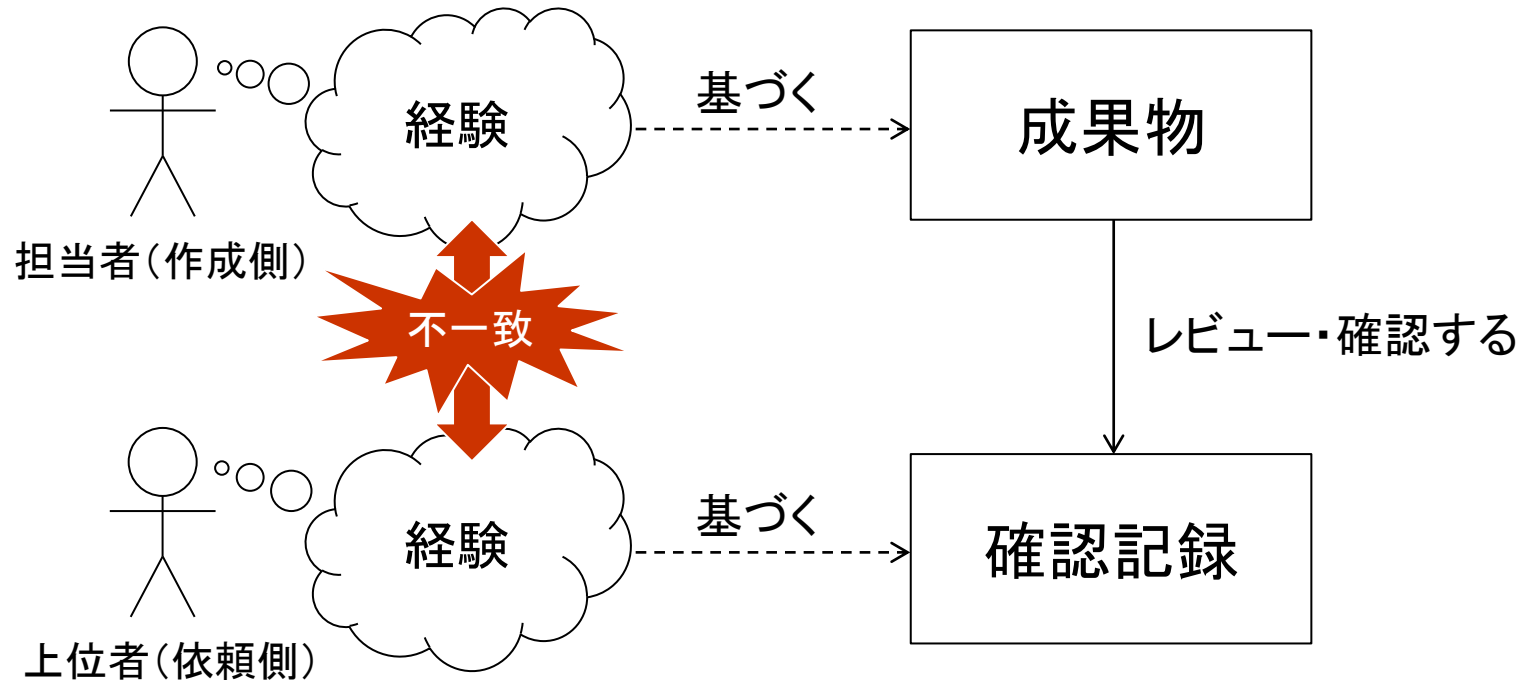


開発現場へのD-Case導入事例の紹介

第7回 D-Case研究会
(株)デンソークリエイト 宇都宮浩之

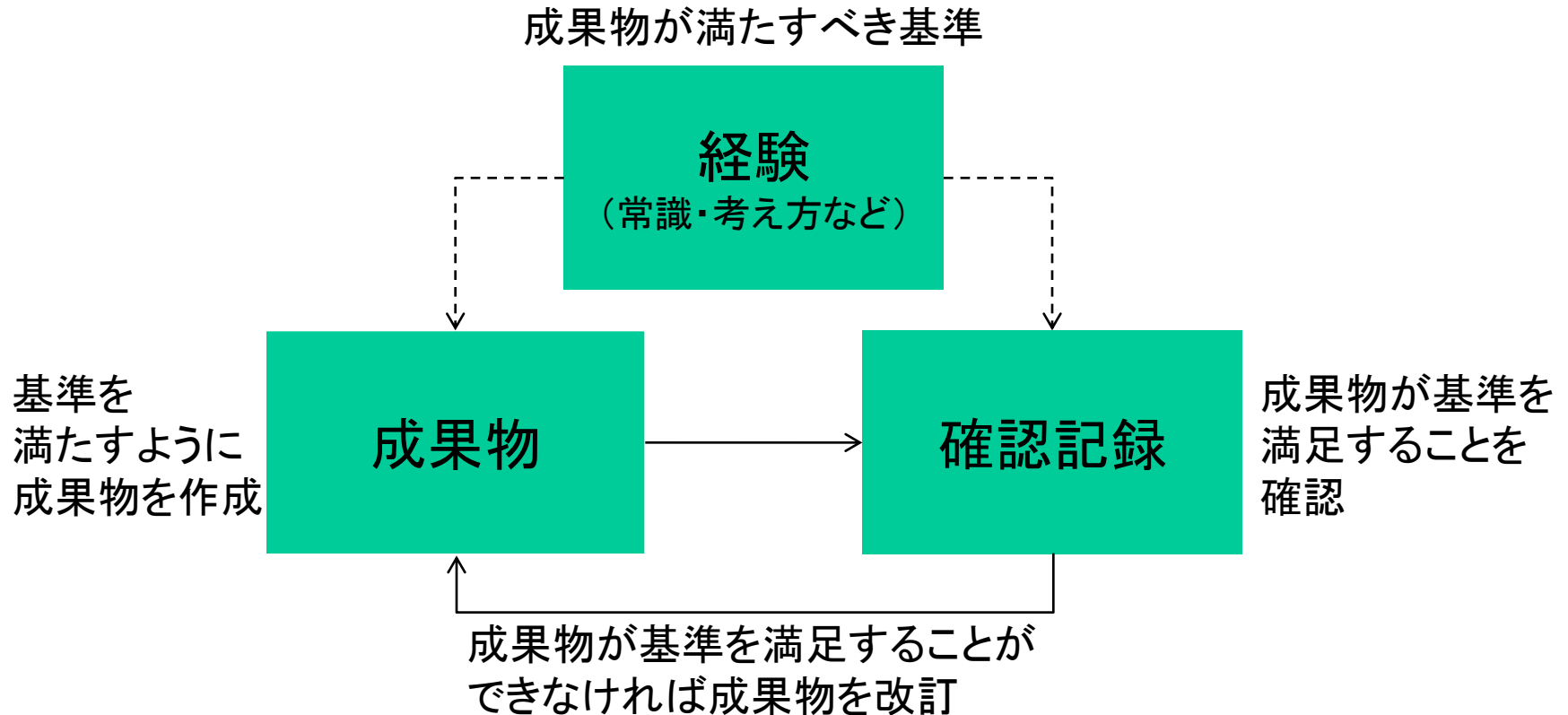
開発現場が抱える問題

- 上位者の経験が成果物に上手く反映できていない
 - 担当者は成果物の妥当性を論理的に説明できず、上位者は明確な判断基準に基づいた指摘ができていないため、もぐら叩きのレビューになってしまう



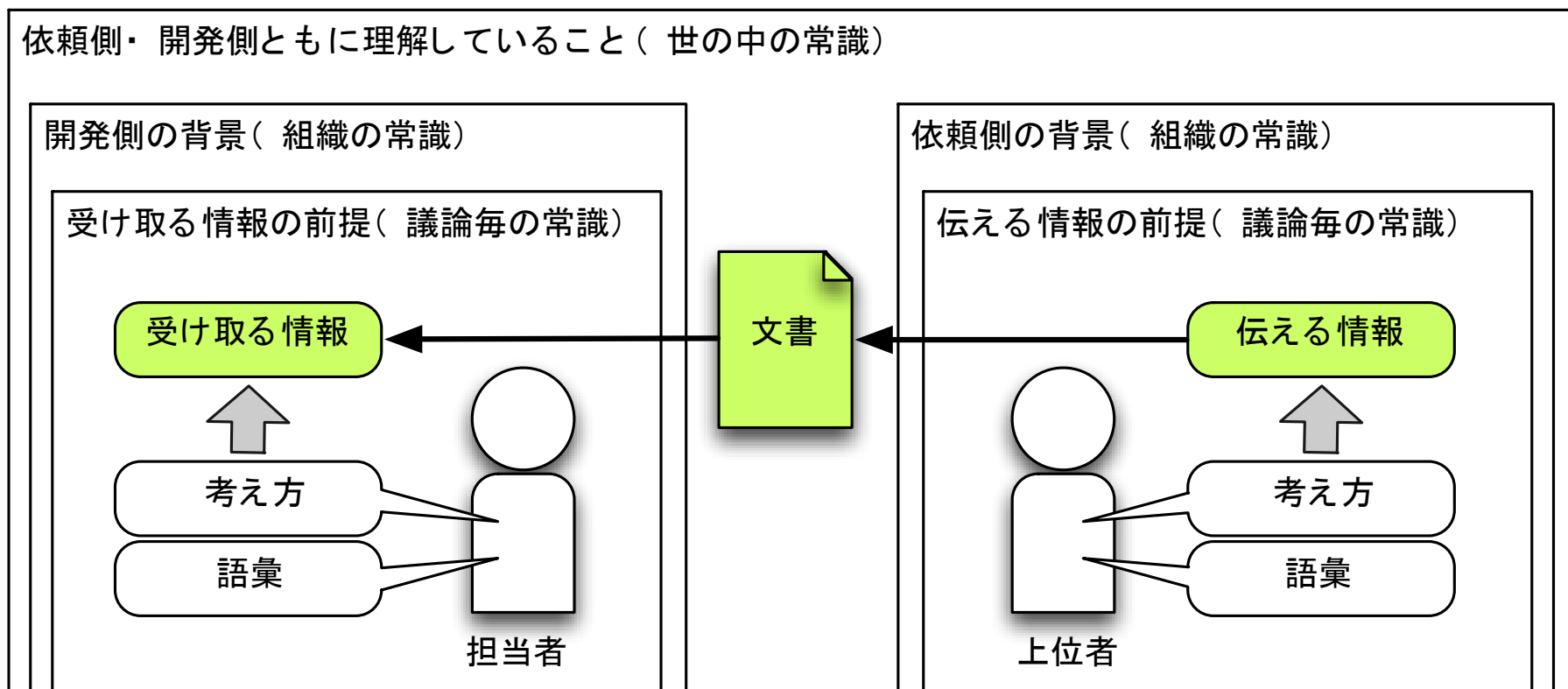
問題の解決方針

- 上位者と担当者が同じ経験に基づいて仕事に取り組む環境を整えることで問題解消を図る

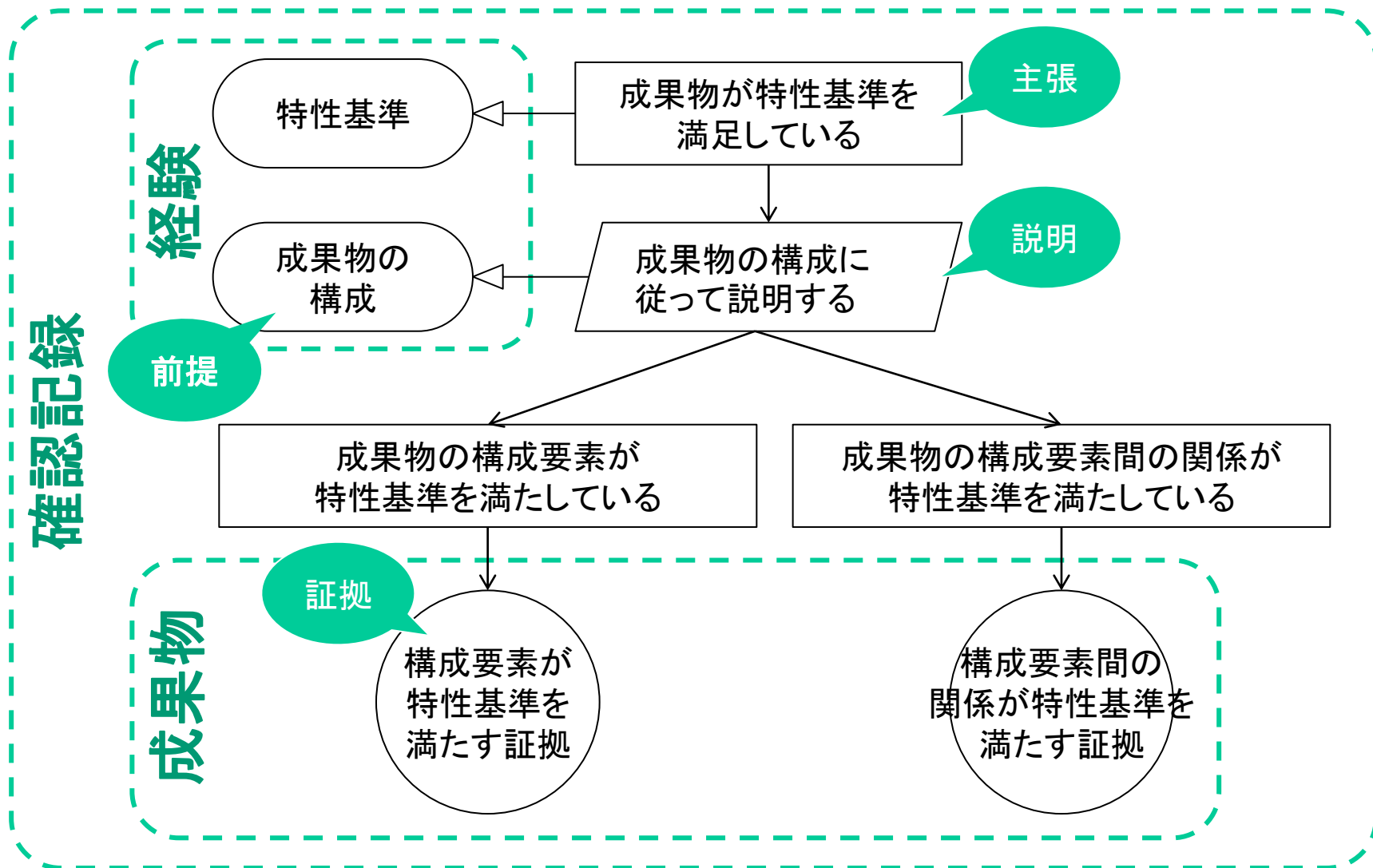


上位者の経験を担当者と正しく共有できる手段が必要

- 双方の語彙や常識、考え方が揃えば相手と正しい共通理解を持てるはず



前提となる語彙や常識、考え方を見える化する手段としてD-Caseを導入する



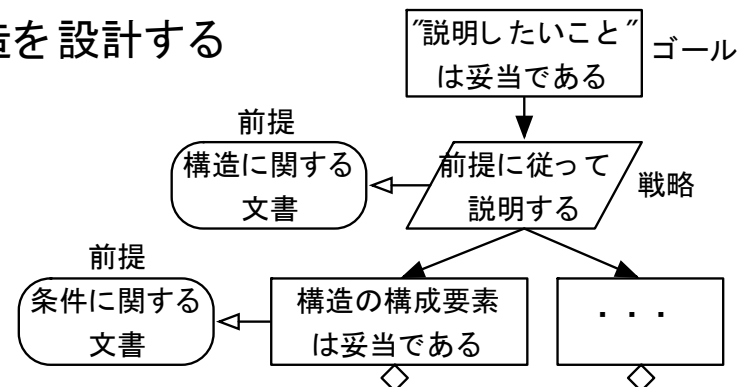
手順1： 前提となる文書を揃える

- ・ 相手と合意形成できている文書を揃える
- ・ 不足する場合は右記の4 状態を念頭に新規に用意する

相手	知らない	③文書化して理解を得る	④一緒に文書化する
	知っている	①合意形成できている	②教えてもらって文書化
		知っている	知らない
		自分	

手順2： 説明の構造を設計する

- ・ 手順1 で揃えた文書を前提として説明の構造を設計する

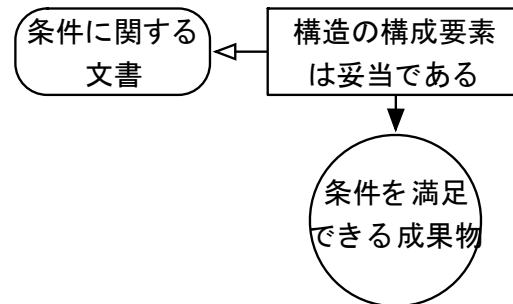


手順4： D-Caseで説明する

- ・ “構造に関する文書”に基づいて説明したいことを手頃な大きさに分解して説明する
- ・ 説明の妥当性は、“条件に関する文書”を成果物が満足していることで納得を得る
- ・ 相手の納得が得られなければ手順を戻って再チャレンジ！

手順3： 成果物を証拠に紐づける

- ・ 説明の妥当性を最終的に支える証拠として条件を満足できる成果物を紐付ける



- ・ 下記赤字部分の仕様変更に対応した結果を判断する設計品質レビュー

<インタフェース仕様>

ret = ConvMsec(unsigned short sec, unsigned long *msec);

<変換規則>

- ・ secを1000倍した値を*msecに格納し、E_OKを戻り値とする。
- ・ **secが255より大きい場合は*msecに255000を格納し、E_OVERを戻り値とする。**

<設計品質レビューの報告内容>

確認事項	妥当性判断	証拠	妥当性判断の根拠
テスト項目数は十分であるか？	OK	テスト仕様書	既存処理をコードレビューで押さえ、追加処理の境界値に関するテスト項目を抽出。
：	：	：	：
仕様を満足することが確認されているか？	OK	テスト報告書	テスト仕様書の該当項目がすべて合格している

- 下記赤字部分の仕様変更に対応した結果を判断する設計品質レビュー

<インタフェース仕様>

```
ret = ConvMsec(unsigned short sec, unsigned long *msec);
```

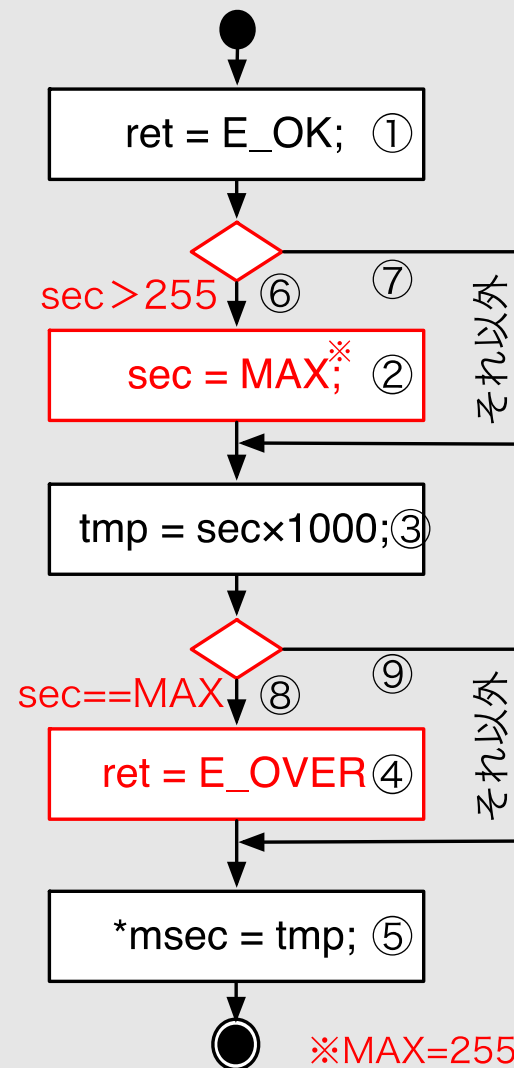
<変換規則>

- secを1000倍した値を*msecに格納し、E_OKを戻り値とする。
- secが255より大きい場合は*msecに255000を格納し、E_OVERを戻り値とする。**

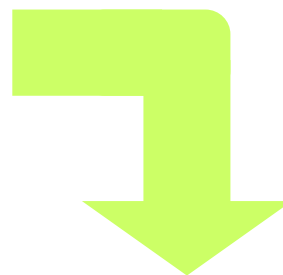
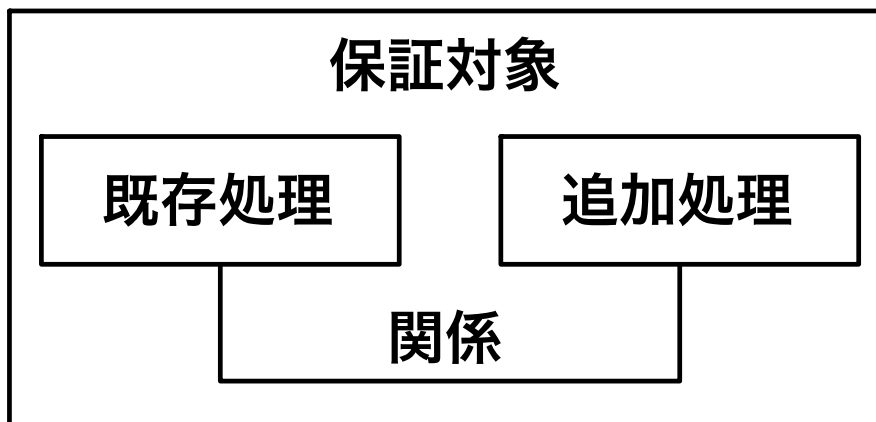
<設計品質レビューの報告内容>

確認事項	妥当性判断	証拠	妥当性判断の根拠
テスト項目数は十分であるか？	OK	テスト仕様書	既存処理をコードレビューで押さえ、追加処理の境界値に関するテスト項目を抽出。
:	:	:	:
仕様を満足することが確認されているか？	OK	テスト報告書	テスト仕様書の該当項目がすべて合格している

<処理の流れ>

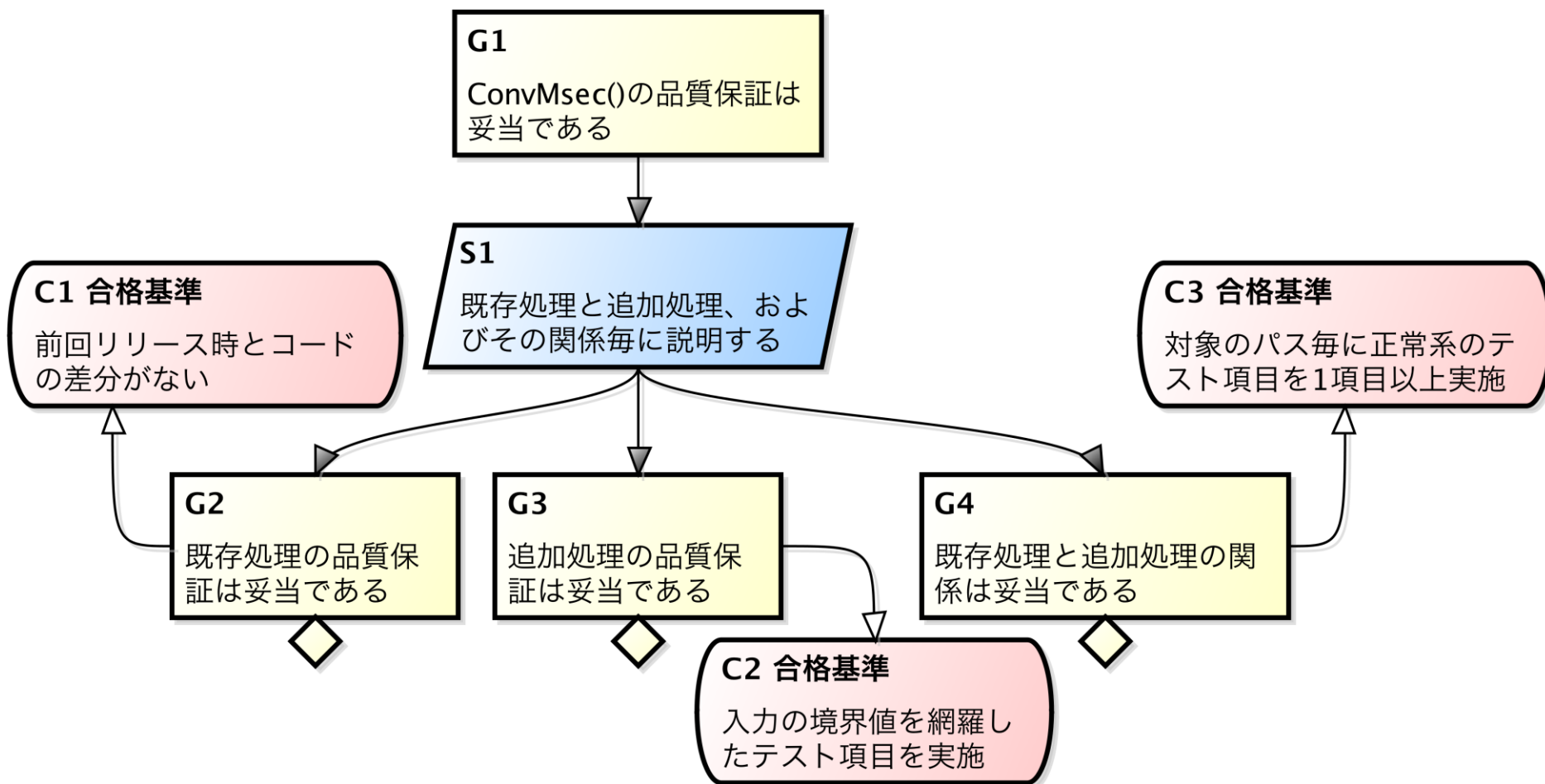


- 下記の「保証対象の構造」に従って上位者の考える合格基準を明文化



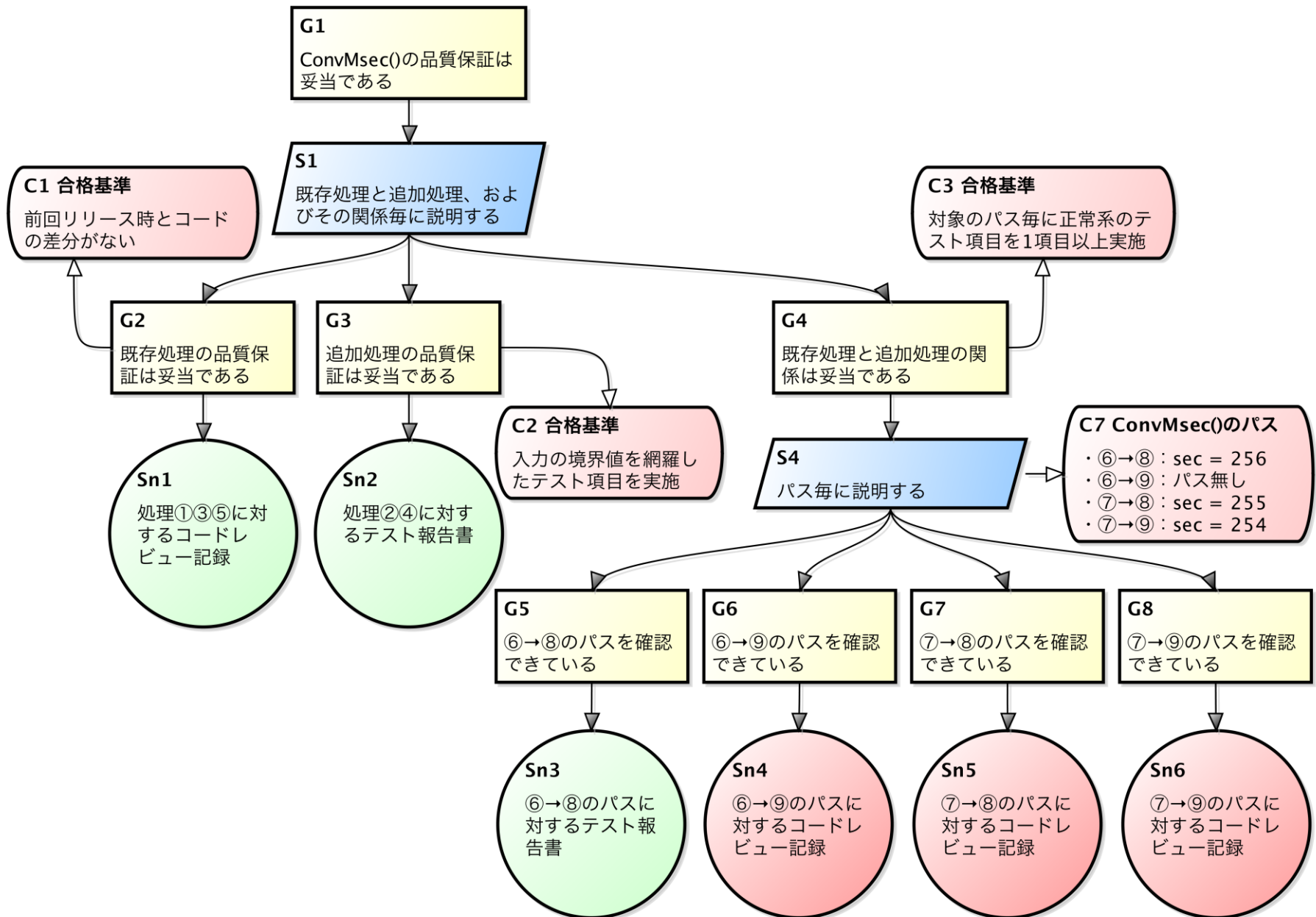
品質保証対象	合格基準
既存処理	(C1) 前回リリース時とコードの差分がない
追加処理	(C2) 入力の境界値を網羅したテスト項目を実施
既存処理と追加処理の関係	(C3) 対象のパス毎に正常系のテスト項目を1項目以上実施

- 上位者の考える合格基準を前提として説明の構造を見える化



手順3: 成果物を証拠に紐付ける

10 / 25



G1ConvMsec()の品質保証は
妥当である**S1**既存処理と追加処理、およ
びその関係毎に説明する**C1 合格基準**前回リリース時とコード
の差分がない

手順4: D-Caseで説明(その1)

既存処理の品質保
証は妥当である追加処理の品質保
証は妥当である**Sn1**処理①③⑤に対
するコードレ
ビュー記録**Sn2**処理②④に対す
るテスト報告書**C2 合格基準**入力の境界値を網羅し
たテスト項目を実施**C3 合格基準**対象のパス毎に正常系のテ
スト項目を1項目以上実施**G4**既存処理と追加処理の関
係は妥当である**S4**

パス毎に説明する

C7 ConvMsec()のパス

- ・⑥→⑧: sec = 256
- ・⑥→⑨: パス無し
- ・⑦→⑧: sec = 255
- ・⑦→⑨: sec = 254

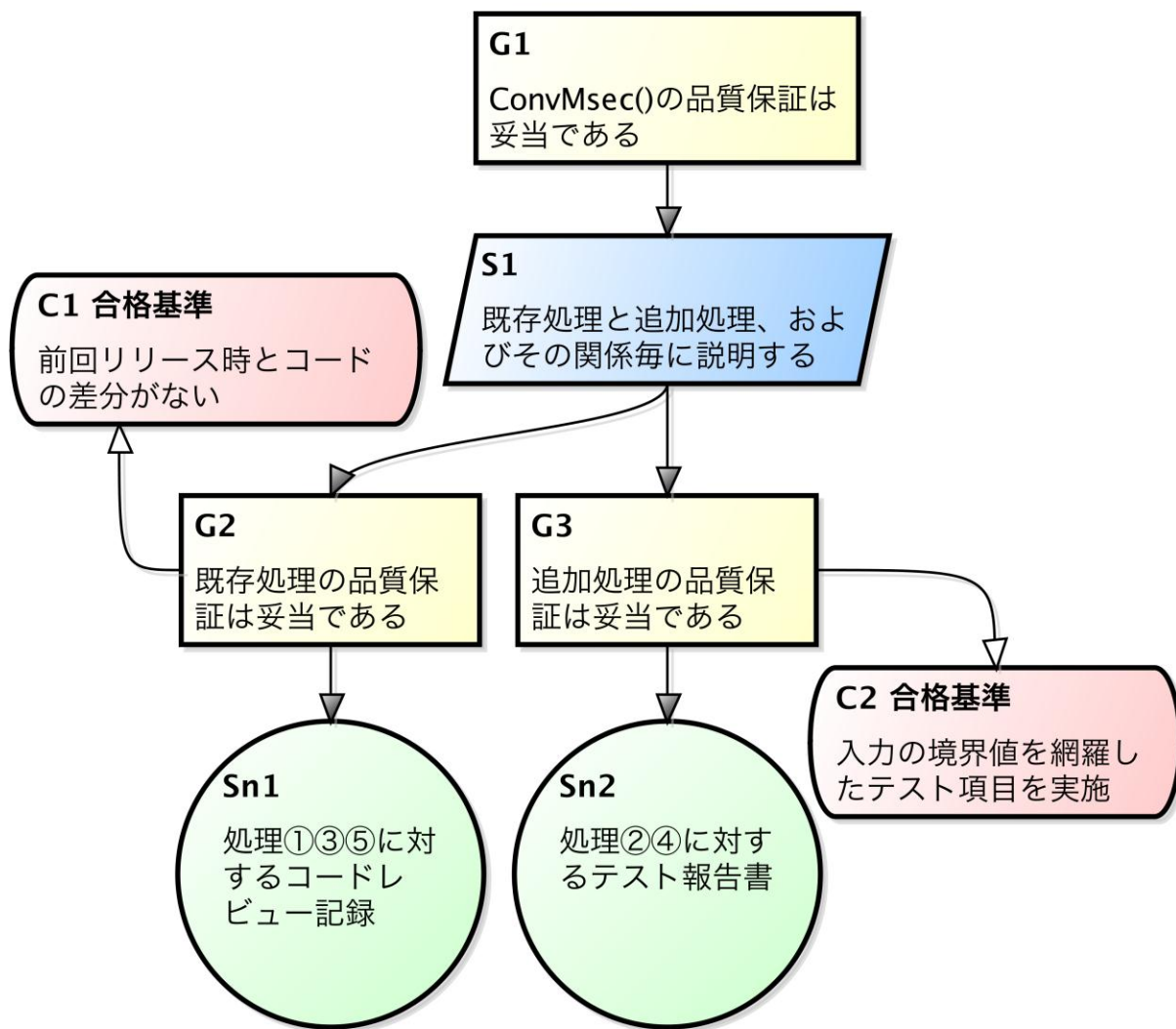
手順4: D-Caseで説明(その2)

G5⑥→⑧のパスを網羅
できている**G6**⑥→⑨のパスを網羅
できている**G7**⑦→⑧のパスを網羅
できている**G8**⑦→⑨のパスを網羅
できている**Sn3**⑥→⑧のパスに
対するテスト報
告書**Sn4**⑥→⑨のパスに
対するコードレ
ビュー記録**Sn5**⑦→⑧のパスに
対するコードレ
ビュー記録**Sn6**⑦→⑨のパスに
対するコードレ
ビュー記録

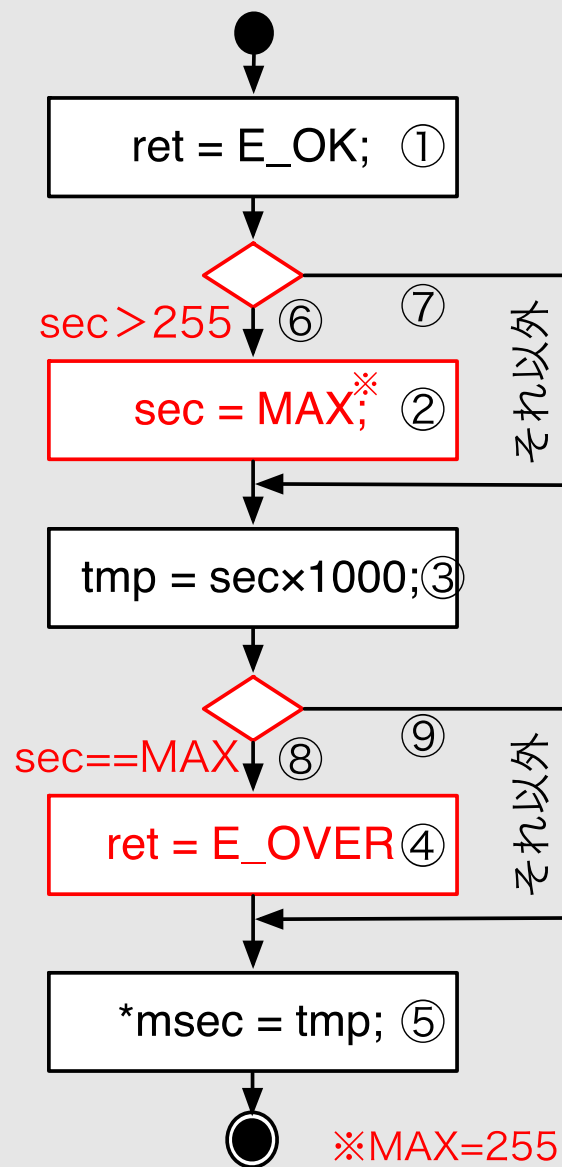
手順4: D-Caseで説明(その1)

12 / 25

- 上位者の合格基準(C1,2)を満足しており問題無し



<ConvMsec実装>



手順4: D-Caseで説明(その2)

13 / 25

- 上位者の合格基準(C3)未達の証拠(Sn4, 5, 6)を検出
- Sn5, 6は合格基準を満たすため追加テストを実施
⇒ Sn5の追加テスト時に不具合発見!

C3 合格基準

対象のパス毎に正常系のテスト項目を1項目以上実施

G4

既存処理と追加処理の関係は妥当である

sec=255時に
E_OVERが返る

S4

パス毎に説明する

C7 ConvMsec()のパス

- ⑥→⑧: sec = 256
- ⑥→⑨: パス無し
- ⑦→⑧: sec = 255
- ⑦→⑨: sec = 254

G5

⑥→⑧のパスを確認できている

G6

⑥→⑨のパスを確認できている

G7

⑦→⑧のパスを確認できている

G8

⑦→⑨のパスを確認できている

Sn3

⑥→⑧のパスに対するテスト報告書

Sn4

⑥→⑨のパスに対するコードレビュー記録

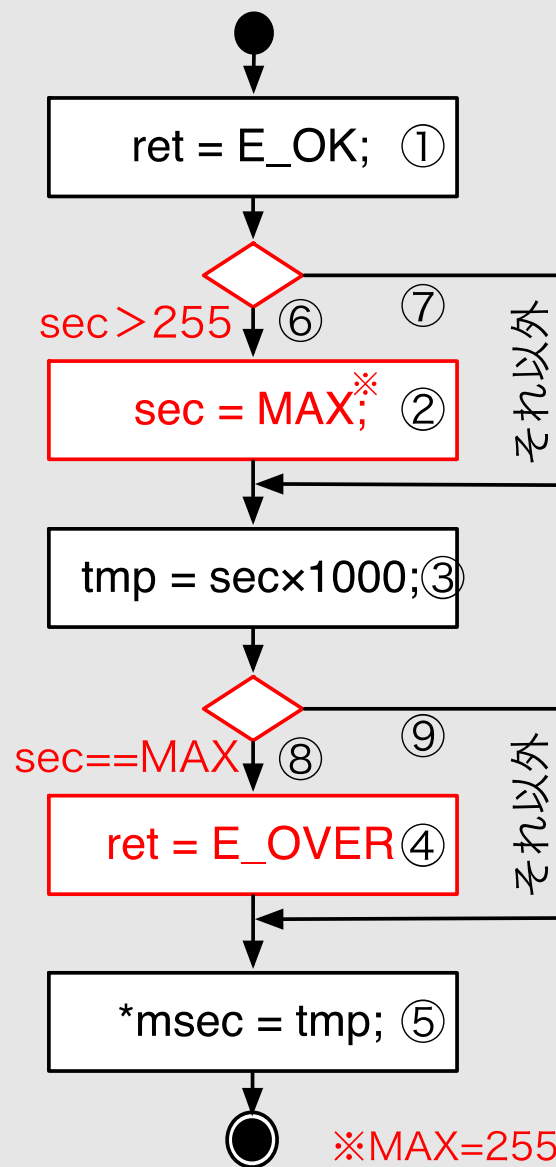
Sn5

⑦→⑧のパスに対するコードレビュー記録

Sn6

⑦→⑨のパスに対するコードレビュー記録

<ConvMsec実装>



上位者 - 担当者の理解のズレが発生した原因

14 / 25

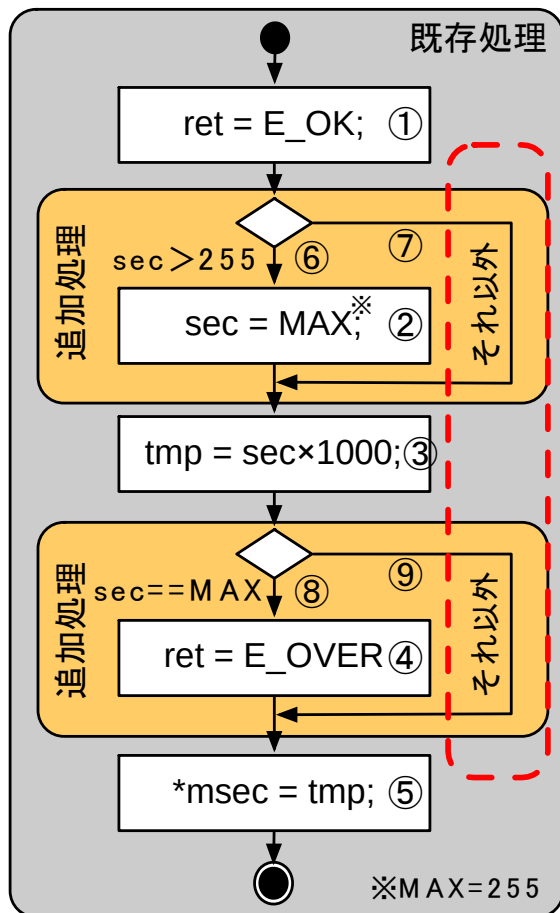
<設計品質レビューの報告内容>

確認事項	妥当性判断	妥当性判断の根拠
テスト項目数は十分であるか？	OK	既存処理をコードレビューで押さえ、追加処理の境界値に関するテスト項目を抽出。

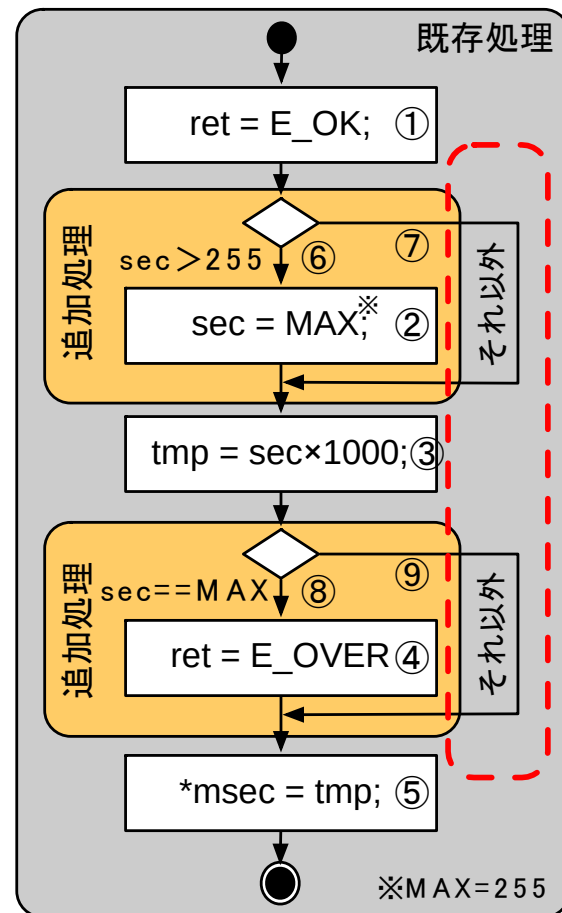


担当者は⑦⑨の処理が空で追加がないため既存処理と理解していた
⇒ 赤点線枠の理解のズレが発生

<上位者の理解>

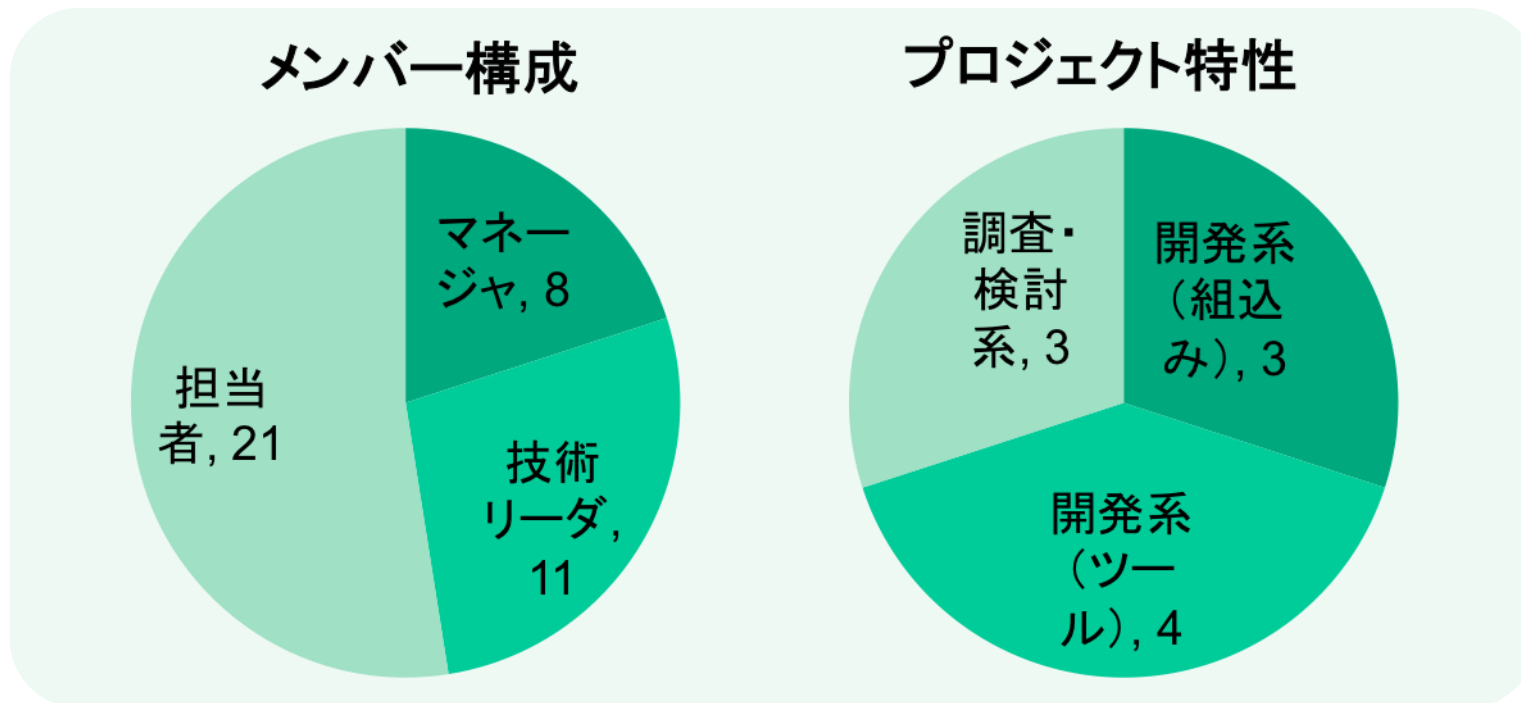


<担当者の理解>



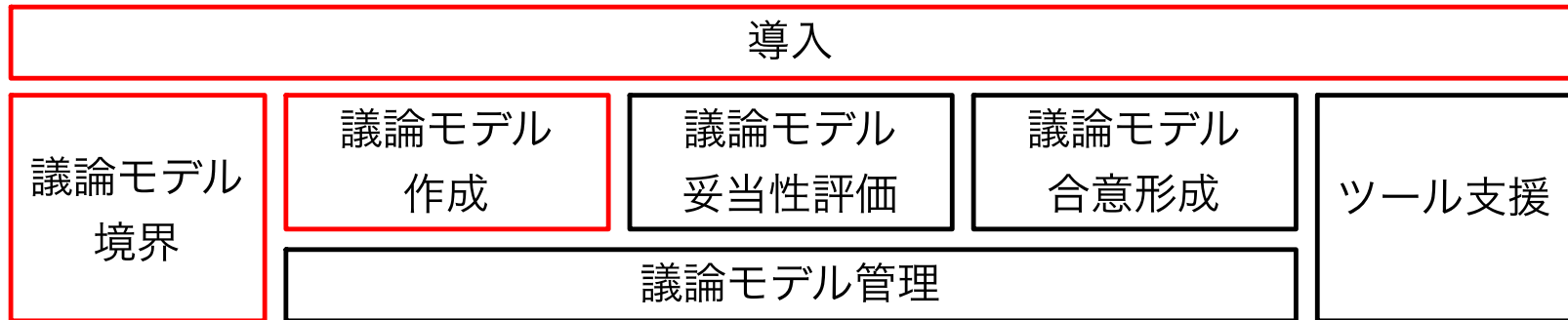
D-Caseを用いることで理解のズレを見える化できた

- メンバー総数40名で10プロジェクトを担当する部署へ導入



- D-Caseを導入した工程：
 - 成果物(仕様書、設計書等)の品質を審議する「設計品質レビュー」
- 導入前の準備：
 - D-Case教育シラバスに基づいたトレーニングを全メンバー向けに実施
 - D-Case作成ツールとしてastah GSN(ChangeVision)を導入

- レビューの説明で使えるD-Caseを描けるよう導入、議論モデル境界、議論モデル作成に対応したトレーニングを実施(赤枠部分)



No.	科目名称	概要
1.	導入	①D-Caseの必要性、②D-Caseの基本概念、③活用事例の共有
2.	議論モデル境界	①議論範囲の決定、②議論範囲の妥当性評価
3.	議論モデル作成	①主張の設定、②主張を分解する説明の設定、③主張・説明に対する前提の設定、④主張を支える証拠の設定、⑤議論モデルの作成手順
4.	議論モデル妥当性評価	①用語の評価基準、②主張の評価基準、③説明の評価基準、④前提の評価基準、⑤証拠の評価基準
5.	議論モデル合意形成	①合意形成の基礎、②合意形成の手順
6.	議論モデル管理	①議論モデルの変更要因、②変更に対する影響範囲分析、③変更に対する追跡性の版管理、④変更管理の手順
7.	ツール支援	①ツール要求、②ツール評価、③ツール導入手順

- 68回のレビューのうち16回の会議で“考慮不足”を検出できた

No.	コスト種別	内容	工数
1.	運用コスト	D-Caseの作成時間	105h
2.	削減コスト	考慮不足の見落としで発生する手戻り予想時間	186h
3.	効果コスト	(削減コスト) - (運用コスト)	81h

《“考慮不足”の検出事例》

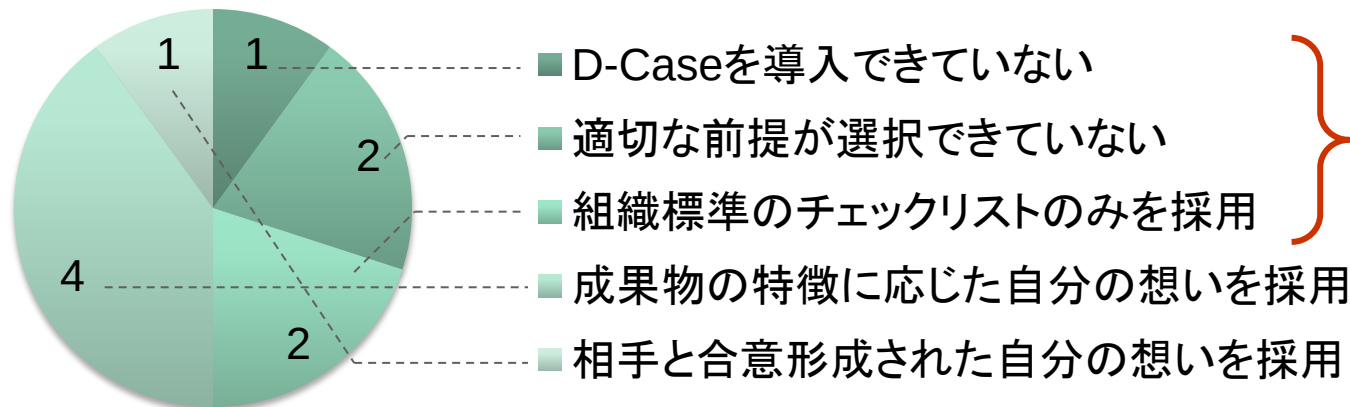
- 影響範囲の判断ミス
- 設計情報の記載漏れ
(入力パターンの漏れ等)
- リリース基準の未達箇所、等

- その他の効果として...

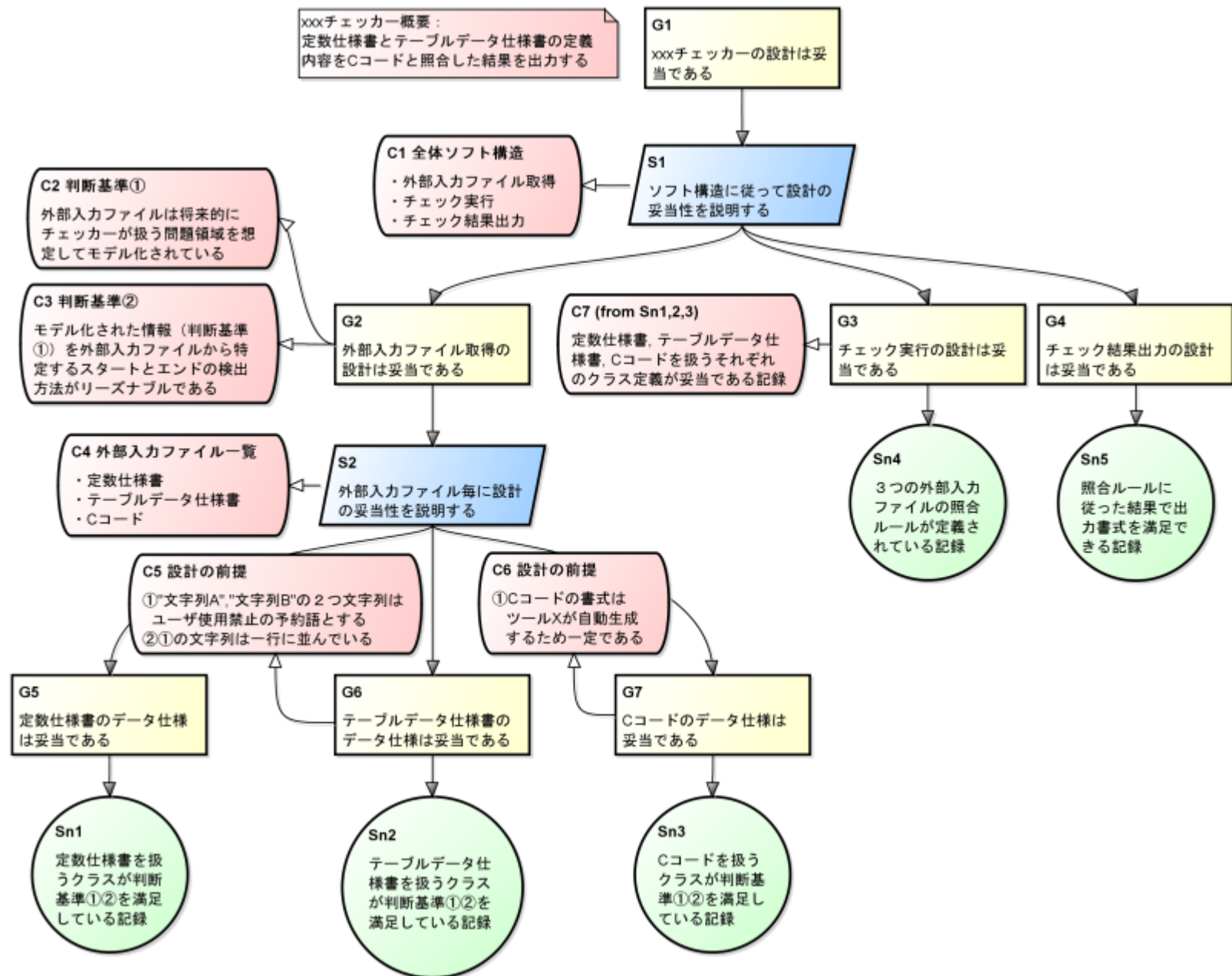
- 成果物に対するレビューの網羅状況が見える化できるようになってきた
- レビュー時の判断に役立つD-Caseの記述パターンが見えてきた

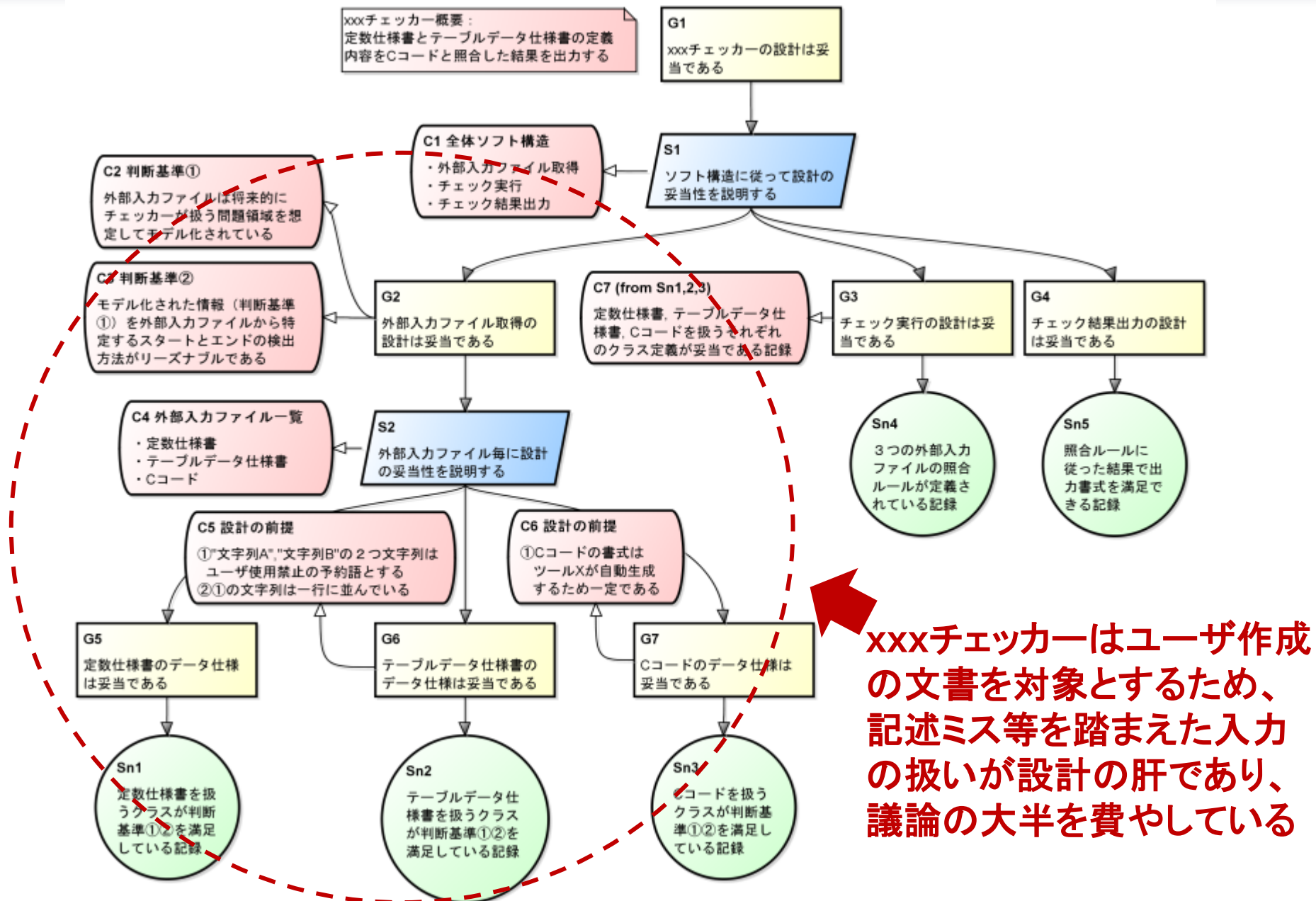
プロジェクト毎の活用状況

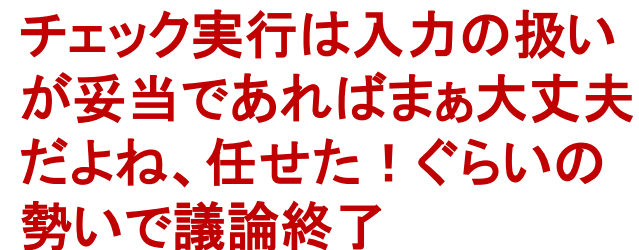
《前提の状態でプロジェクト毎の活用状況を評価》

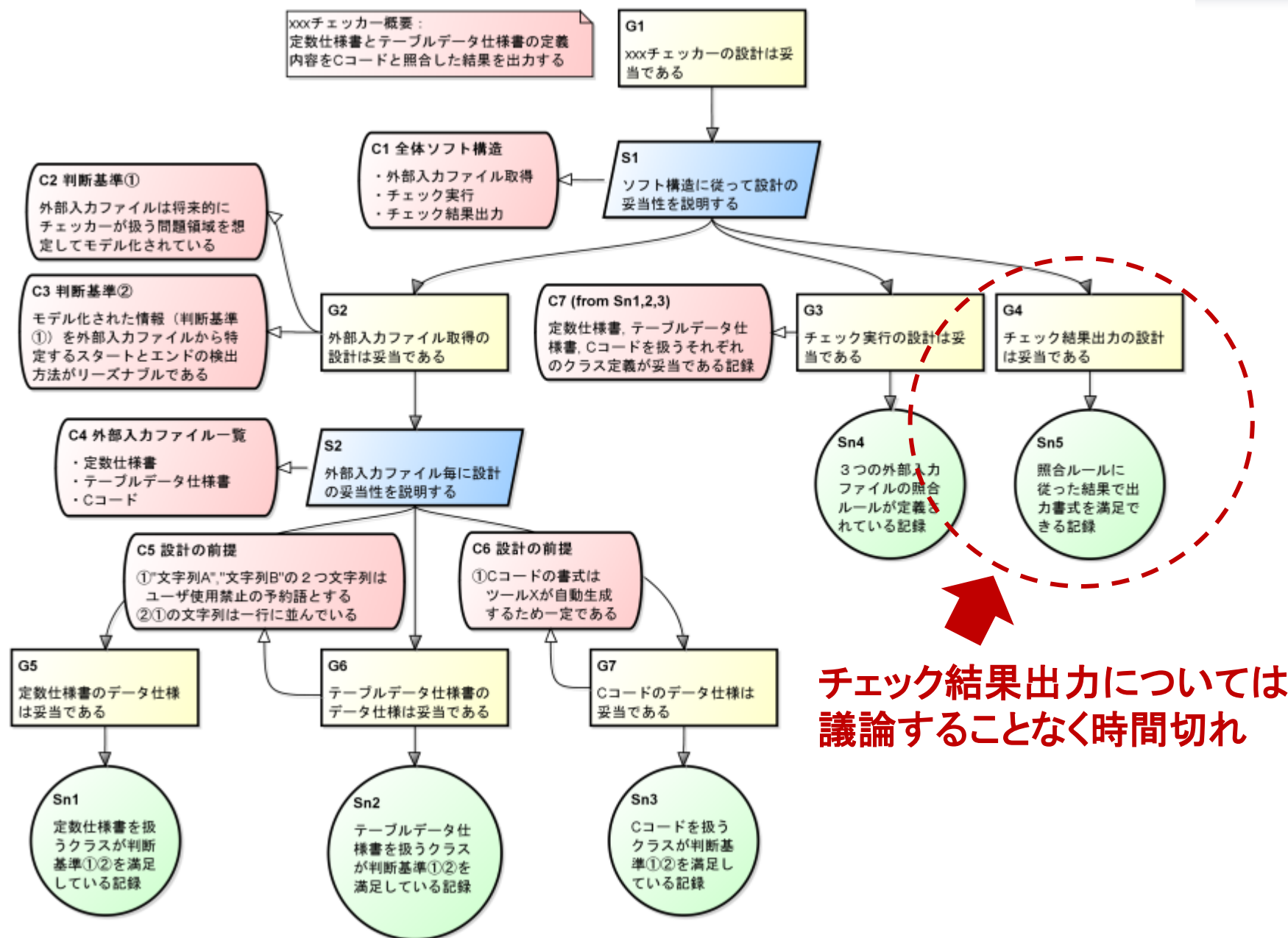


この状況のプロジェクトが上手く活用し始めると削減コストの増加が期待できる

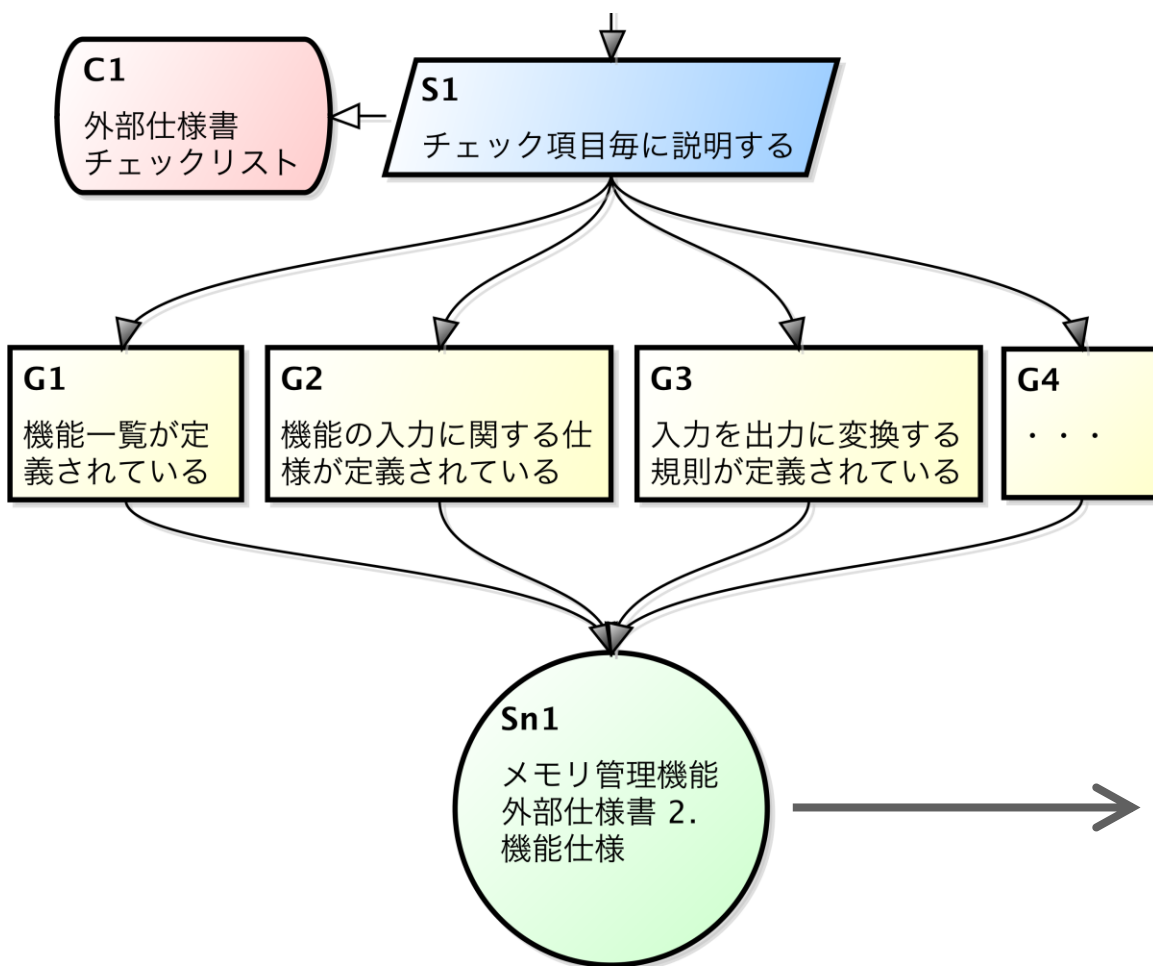








- 複数のゴールが一つの証拠に集約している場合、複数の視点が一箇所に混在した文書になっている場合が多い(文書品質が低い可能性がある)



《メモリ管理機能外部仕様書》

2. 機能仕様

2.1. 初期化

メモリ管理機能はECUの初期化要求時に不揮発メモリを初期化する。要求時に指定された開始アドレスからデータ長までの範囲を初期化対象とする。また、初期化時に指定された値を不揮発メモリの初期値とする。データ長の値域は0～100までとし、それ以外の値が設定された場合にはエラーを返す。初期値に使用できる値域は0～255とする。値域外の値を渡された場合のエラーはE_INVALIDを戻り値として返す。関数の詳細はx.x節参照。

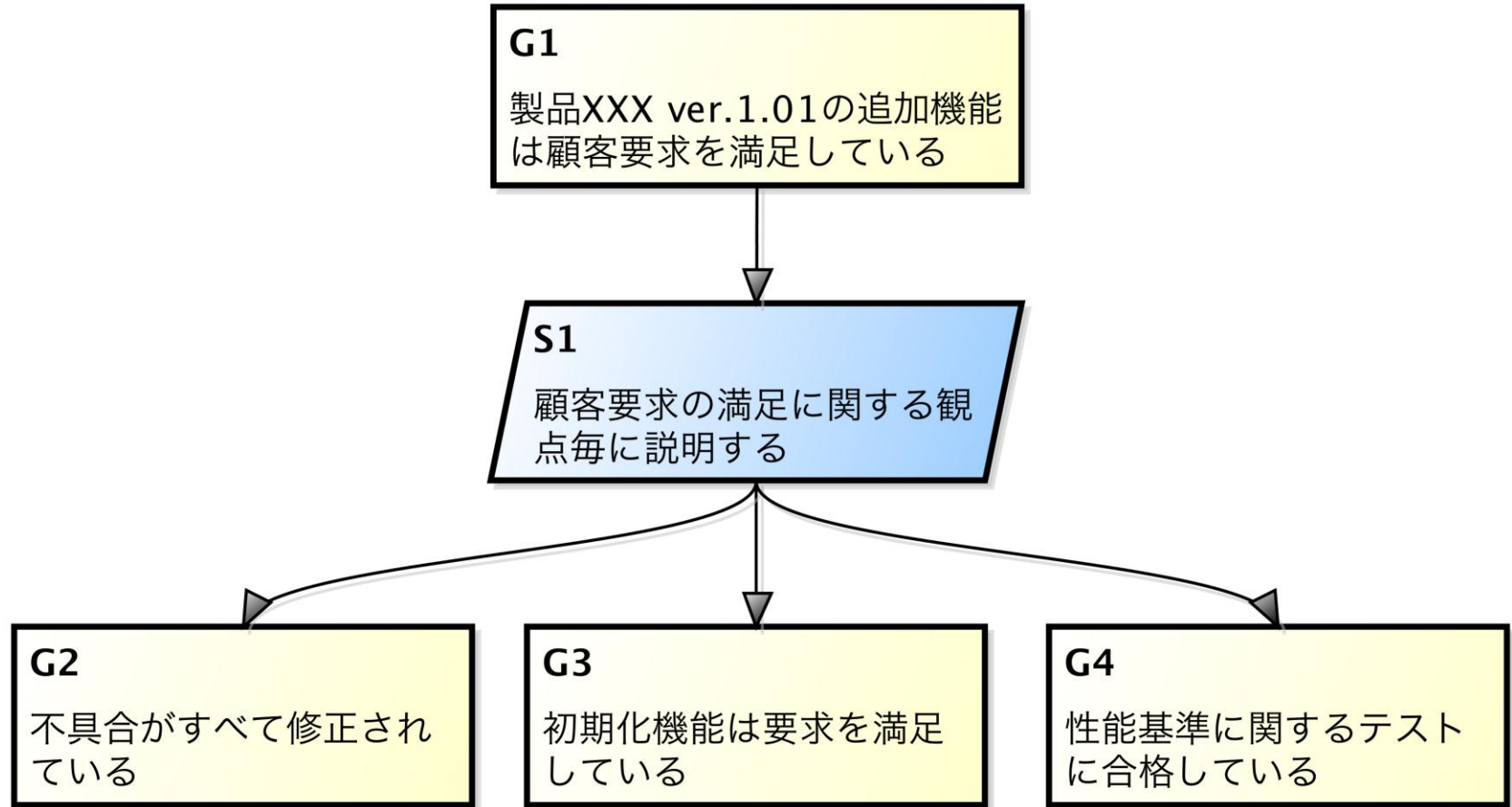
2.2. 不揮発メモリ書込み

:

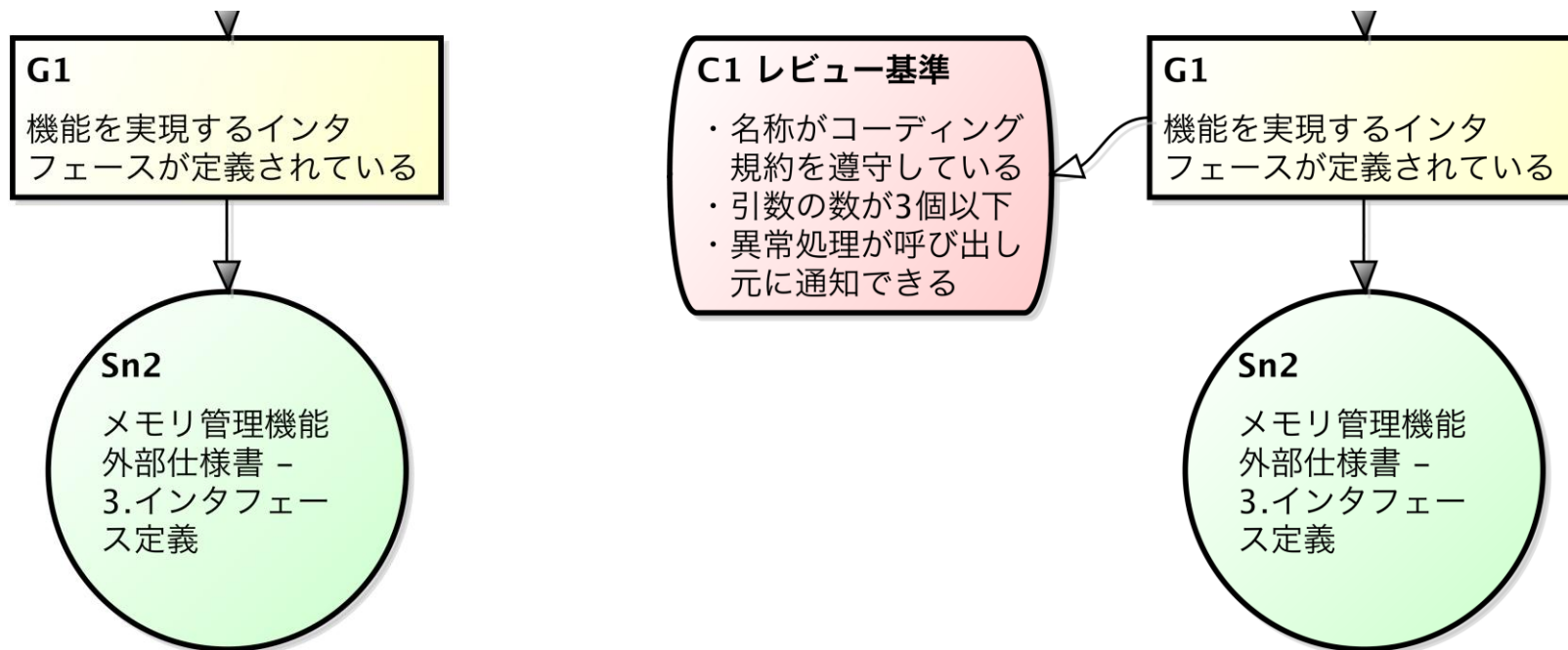
2.3. 不揮発メモリ読み出し

:

- 戦略にレビュー対象の構造や属性を示した前提が関連付けられていない場合、分解されたゴールが網羅的でない場合が多い(納得感が薄い)



- ゴールに証拠の判断基準となる前提が関連付けられていない場合、証拠のレビュー基準がレビューアの間感の場合が多い(客観的な根拠がない)



インタフェースが文書化されていることは分かるが...

インタフェースの妥当性を判断した根拠が明文化されている

プロジェクトで活用しているレビュー基準の整備状況が見える化できる

(現場が抱える問題)

上位者の経験が成果物に上手く反映できていない

- 前提ノードで見える化された上位者の経験に基づいて成果物の品質を確認できるようになった
- 構造化された図表現の特性から表形式では見落としがちな不備(横並び要素の不揃い感等)を検出できるようになった
- 成果物の品質推測に活用できる勘所が導出できた

今後取り組んでいきたいこと

- D-Caseから得られる勘所のパターン整備
- プロジェクトの状態の見える化(レビュー基準の整備状況等)
- 導入工程の拡大(計画レビューへの導入を検討中)