

名古屋大学における D-Case研究の状況と今後の展望

名古屋大学 情報連携統括本部 情報戦略室
教授 山本修一郎

Copyright Prof. Dr. Shuichiro Yamamoto 2014

1

主な話題

- 経緯
- 保証ケース超入門
- モデル図式保証方法論の構築
- モデル図式保証方法の一般化
- エンタープライズアーキテクチャの保証方法
- USDMの保証方法
- 例外試験項目の十分性保証方法

経緯

DEOSプロジェクト
2011- 2013

DEOS協会
2014

D-Case実証評価研究会
2012 —

D-Case部会
2014

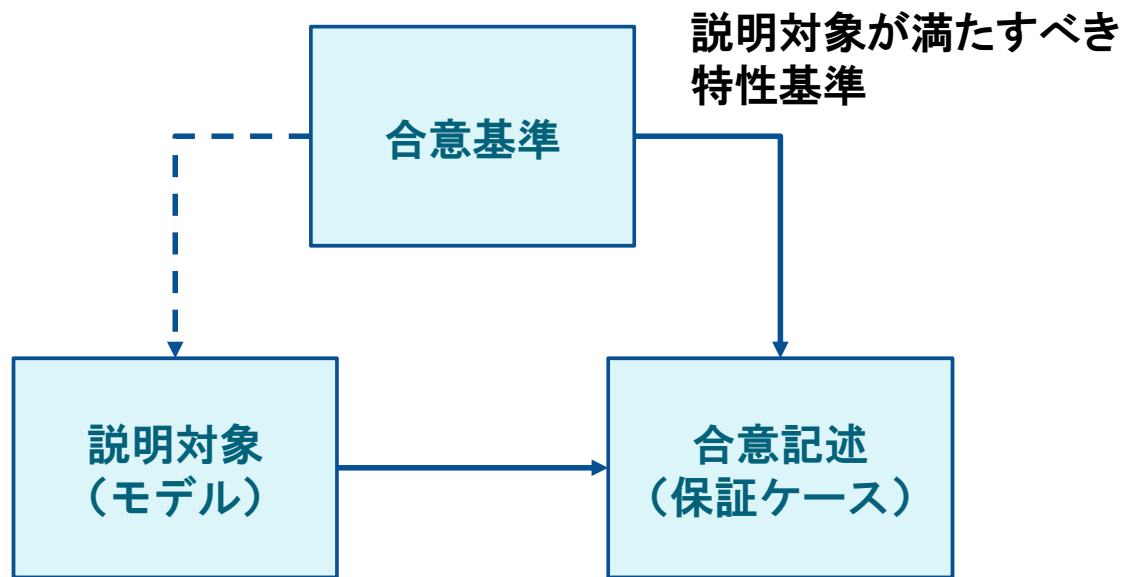
▲ O-DA標準 (TOG, 2013)

▲ D-Caseシラバス
(DEOS協会, 2014)

保証ケース超入門

- 保証ケースの位置づけ
- 保証ケースの基本概念
- 前提の重要性
- 例題
- 前提間の依存関係

基本概念



構成要素とその関係

説明対象が合意基準を満足することを、証拠に基づいて、説明

Copyright Prof. Dr. Shuichiro Yamamoto 2014

5

システム仕様が正しいことを説明する

前提(原則)



前提(対象生産物)



システムの構成要素の仕様が正しい

システム構成要素の仕様
についての確認結果

システム構成要素間の関係
についての仕様が正しい

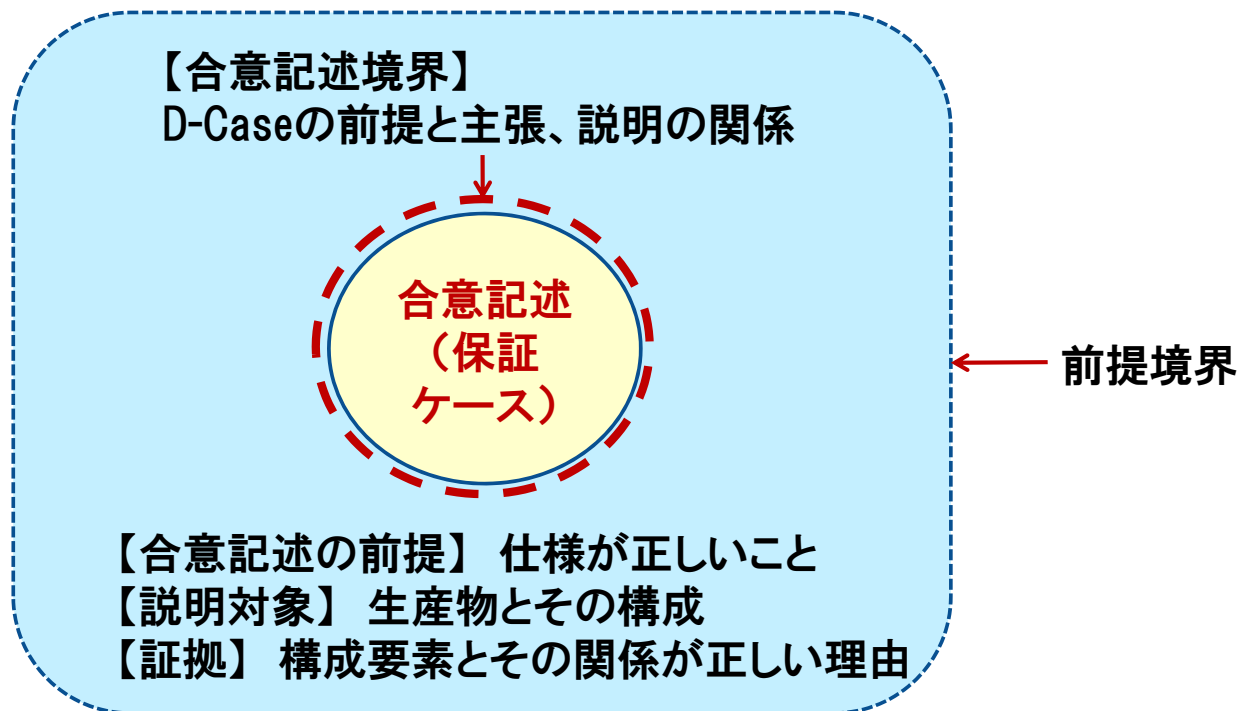
構成要素の
関係についての
確認結果

証拠

Copyright Prof. Dr. Shuichiro Yamamoto 2014

6

合意記述とは

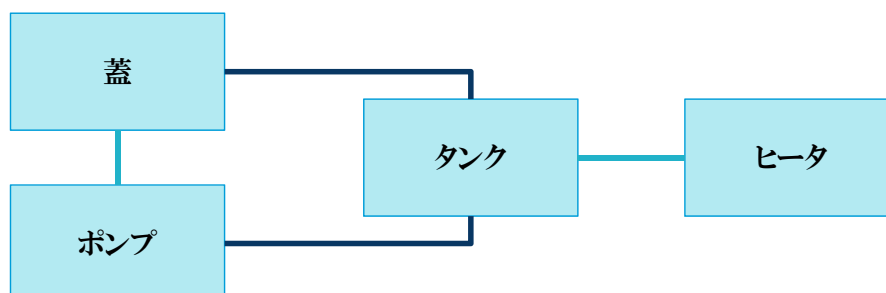


Copyright Prof. Dr. Shuichiro Yamamoto 2014

7

電気ポット・システムの仕様例

- S1:電気ポット・システムには、タンクと蓋、ポンプ、ヒータ、加熱制御があります。
- S2:タンクでは、タンク内に保管している湯量と湯温を測定します。
- S3:蓋は開閉します。
- S4:蓋が閉じているとき、ポンプのロック解除ボタンを押下すると、ロック解除ボタンが点灯します。
- S5:蓋が閉じているとき、ロック解除ボタン点灯中に、給湯ボタンの押下により、ポンプがタンク内の湯を給湯します。
- S6:蓋が閉じているとき、ロック解除ボタン点灯中に、給湯ボタンの押下しても、湯量が不足していると、給湯しません。
- S7:タンクの湯量が十分であれば、湯温が沸点になるまでヒータがタンクを加熱します。
- S8:タンクの湯量が不十分であれば、ヒータがタンクを加熱しません。
- S9:タンクの湯温が沸点になれば、ヒータがタンクの加熱を停止します。

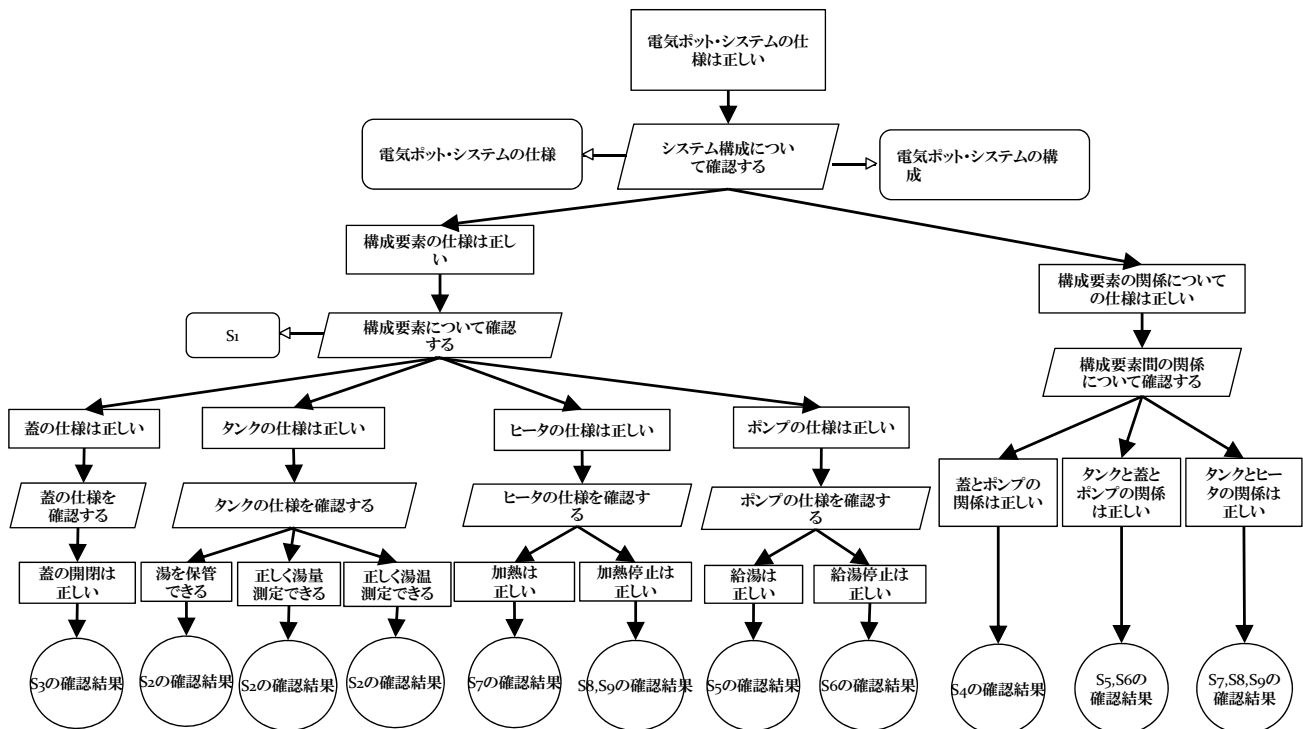


電気ポットのシステム構成

Copyright Prof. Dr. Shuichiro Yamamoto 2014

8

電気ポット・システムの仕様を確認するD-Case



Copyright Prof. Dr. Shuichiro Yamamoto 2014

9

電気ポット・システムの前提依存関係表の例

電気ポット・システムについて、前提としての仕様とシステム構成要素との関係を分析します。

	システム	タンク	蓋	ポンプ	ヒータ
S1	1	1	1	1	1
S2		1			
S3			1		
S4			1	1	
S5		1	1	1	
S6		1	1	1	
S7		1			1
S8		1			1
S9		1			1

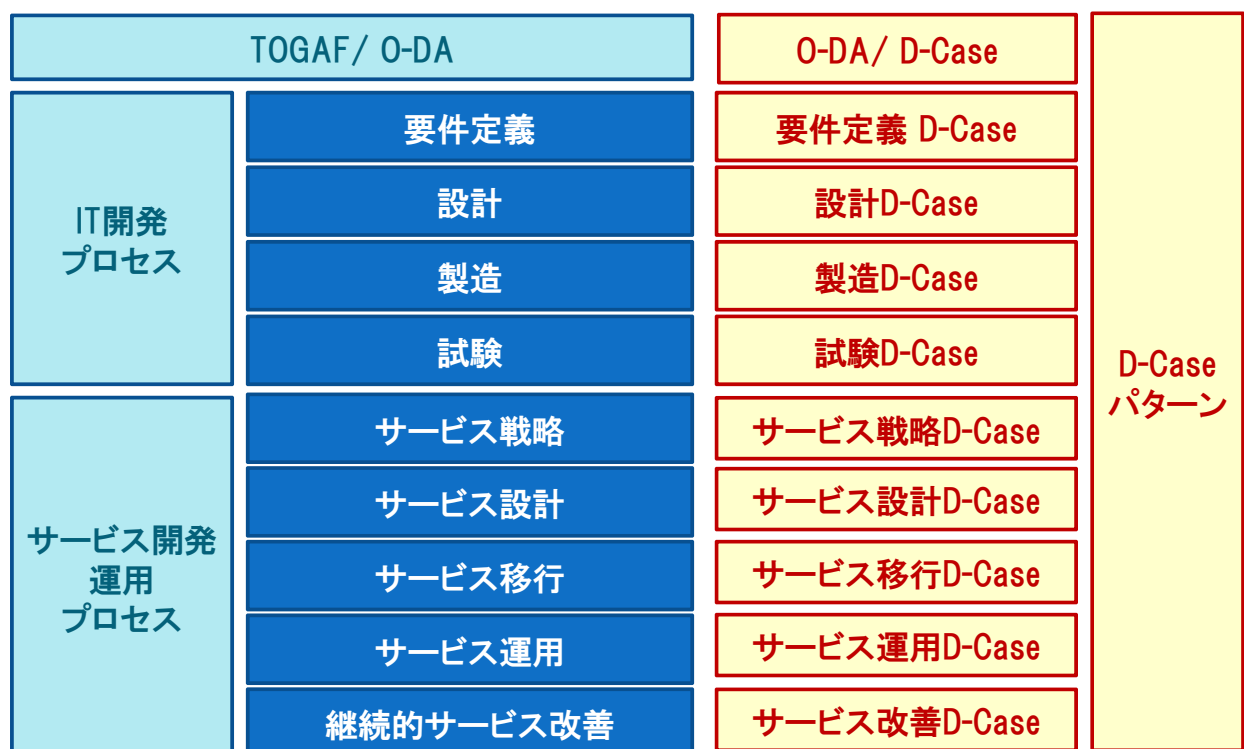
1: システム構成要素が仕様に出現する

Copyright Prof. Dr. Shuichiro Yamamoto 2014

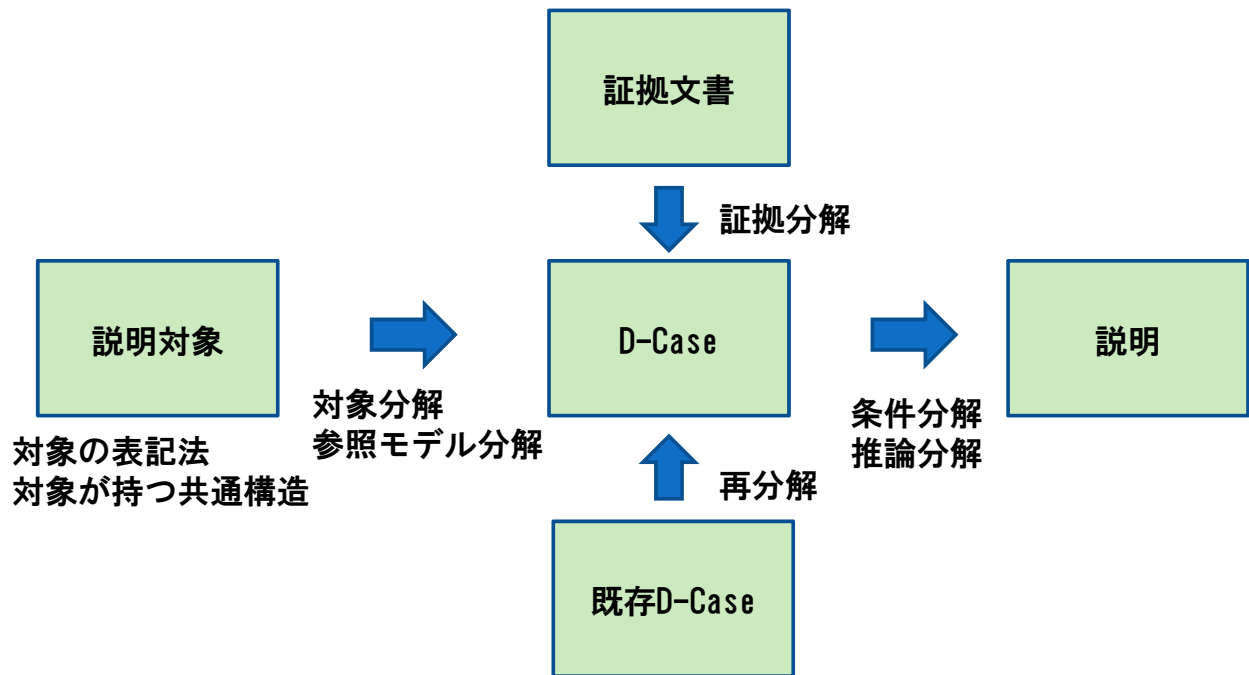
10

モデル図式保証方法論の構築

システム開発運用を保証するD-Caseの全体構成



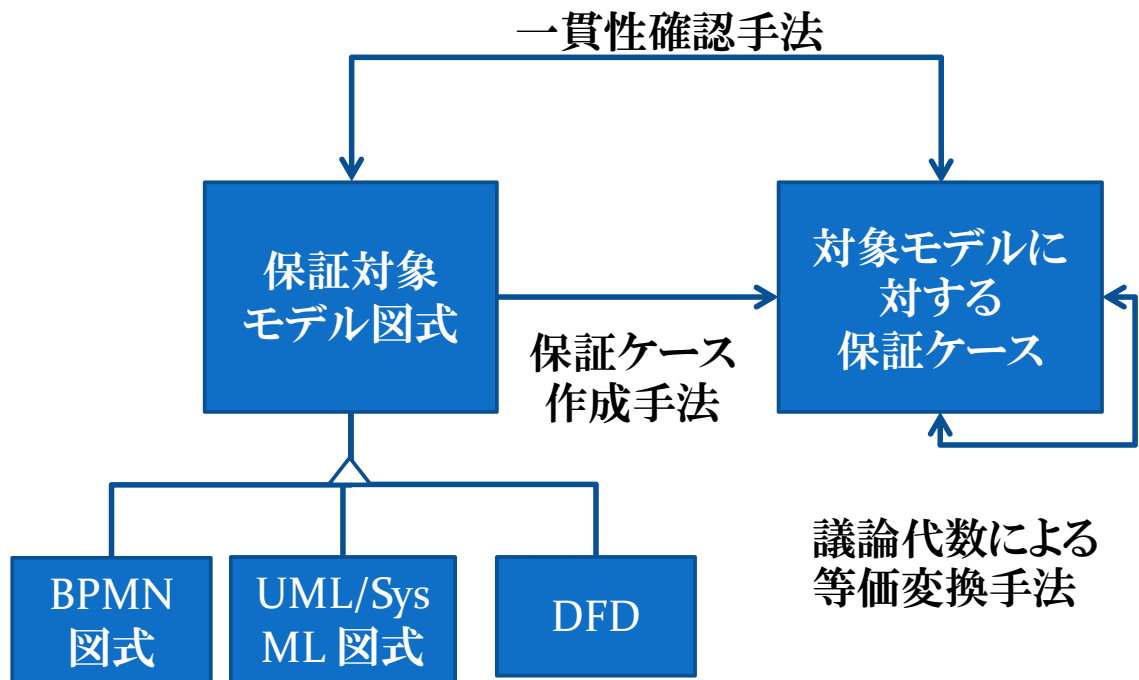
D-Caseパターンの構成



D-Caseパターンライブラリの構成

パターン 分類	説明
記述法 16	アーキテクチャ, 機能, 属性, 完全化, プロセス, プロセス依存関係, 階層化, データフロー図, ビュウ, ユースケース, 要求, 状態遷移, 運用要求, シーケンス図, クラス図, ビジネスモデル
参照モデル 13	DEOS プロセス, リスク, 組み込みシステム, コモンクライテリア, 要求テンプレート, システム境界, 欠陥モード, 非機能要求グレード, テストケース, 問題フレーム, ITILプロセス, ArchiMate動機拡張, テクノロジ層
条件 7	ECA, 条件判断, 代替案選択, 矛盾解消, 平衡化, 改善, 明確化
推論 5	帰納法, 消去法, 否定推論, 反駁
証拠 11	法制度, 形式的証明, モデル検査, 試験成績書, 合意文書, レビュー報告書, シミュレーション, 評価報告書, 説明書, モニタノード, 文書
再利用 2	水平分解, 垂直分解

手法の関係



Copyright Prof. Dr. Shuichiro Yamamoto 2014

15

DFD とGSN の一貫性



Nada Olayan, Shuichiro Yamamoto, A Consistency Check of Dependability Case (D-case) Produced from Data Flow Diagram (DFD), JCKBSE, Communications in Computer and Information Science, 466, (2014), 603-616

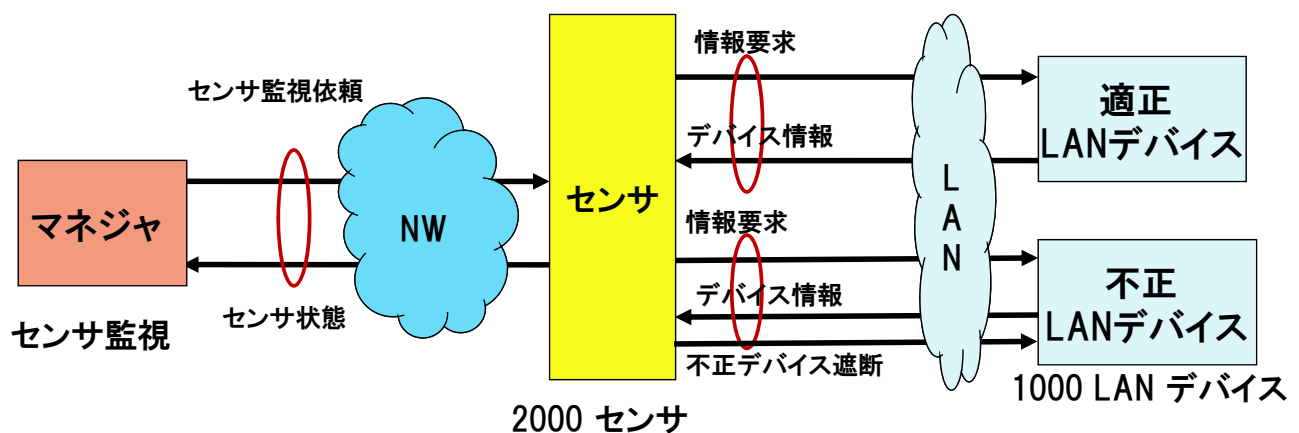
Copyright Prof. Dr. Shuichiro Yamamoto 2014

16

議論代数

Shuichiro Yamamoto, Argument Algebra: A Formalization of Assurance Case Development, JCKBSE, Communications in Computer and Information Science, vol.466, pp.717-725, 2014

LANデバイス監視システム



議論項

- S: LDMS,
 - T1: センサマネジャ,
 - T2: センサ,
 - T3: センサマネジャとセンサの相互作用
- C: DEOS プロセス,
 - C1: 通常運用プロセス,
 - C2: 変化適応サイクル,
 - C3: 障害対応サイクル
- [S, d]: “LDMSのサービス継続性が達成されている”
 ここで S: LDMS, d: “サービス継続性が達成されている”

議論代数による議論項の変換

$[S, d]$
 $\Rightarrow [S, C, d]$
 $\Rightarrow [S, C, d] \mid (C \rightarrow C1, C2, C3)$ ← プロセス分解
 $\Rightarrow [S, C1, d], [S, C2, d], [S, C3, d] \mid (C \rightarrow C1, C2, C3)$ ← アーキテクチャ分解
 $\Rightarrow [S, C1, d], [S, C2, d], [S, C3, d] \mid (C \rightarrow C1, C2, C3) \mid (S \rightarrow T1, T2, T3)$
 $\Rightarrow [T1, C1, d], [T2, C1, d], [T3, C1, d], [S, C2, d], [S, C3, d] \mid (C \rightarrow C1, C2, C3) \mid (S \rightarrow T1, T2, T3)$
 $\Rightarrow [T1, C1, d], [T2, C1, d], [T3, C1, d], [T1, C2, d], [T2, C2, d], [T3, C2, d], [T1, C3, d], [T2, C3, d], [T3, C3, d]$
 $\Rightarrow [T1, C1, d], [T1, C2, d], [T1, C3, d], [T2, C1, d], [T3, C1, d], [T2, C2, d], [T3, C2, d], [T2, C3, d], [T3, C3, d]$
 $\Rightarrow [T1, C1, d], [T1, C2, d], [T1, C3, d], [T2, C1, d], [T2, C2, d], [T2, C3, d], [T3, C1, d], [T3, C2, d], [T3, C3, d]$
 $\Rightarrow [T1, C, d], [T2, C1, d], [T2, C2, d], [T2, C3, d], [T3, C1, d], [T3, C2, d], [T3, C3, d] \mid (C \rightarrow C1, C2, C3) \mid (S \rightarrow T1, T2, T3)$
 $\Rightarrow [T1, C, d], [T2, C, d], [T3, C1, d], [T3, C2, d], [T3, C3, d] \mid (C \rightarrow C1, C2, C3) \mid (S \rightarrow T1, T2, T3)$
 $\Rightarrow [T1, C, d], [T2, C, d], [T3, C, d] \mid (S \rightarrow T1, T2, T3)$ ← プロセス分解
 $\Rightarrow [S, C, d]$ ← アーキテクチャ分解

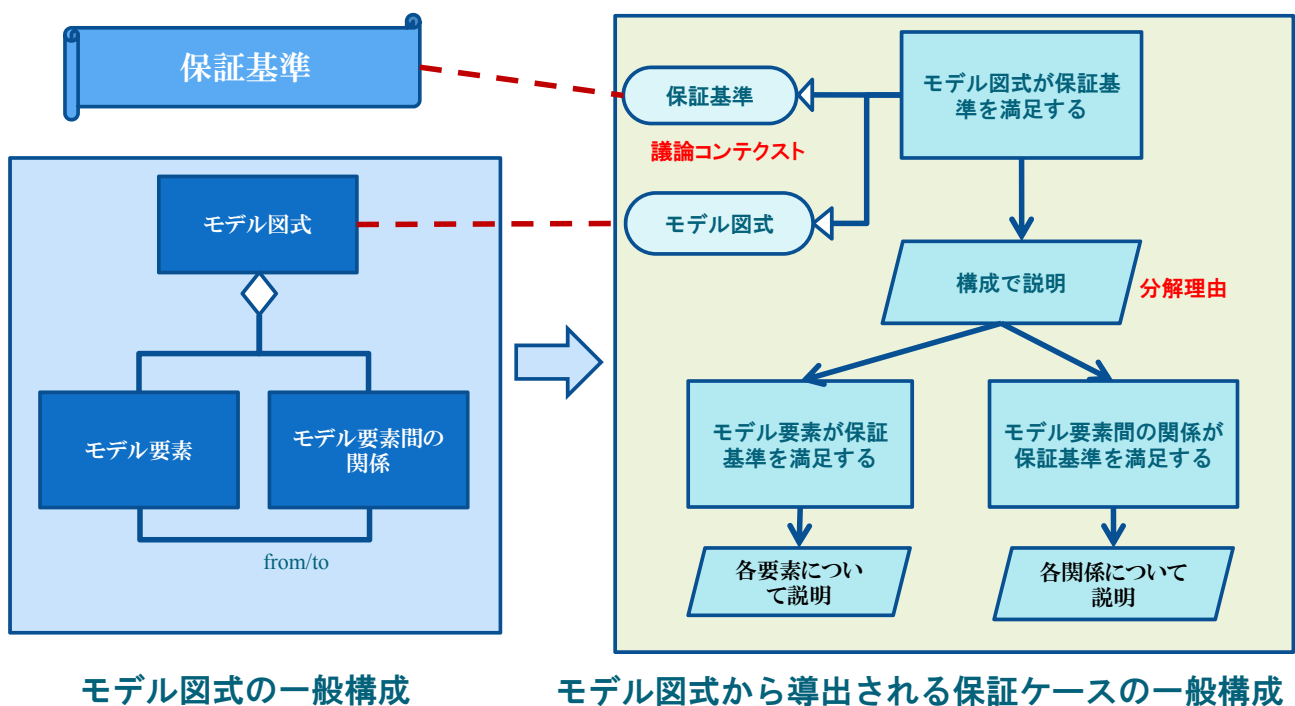
モデル図式保証方法の一般化

Shuichiro Yamamoto,
Interrelationship among Software Models and Assurance cases,
ISEE Symposia Series International Symposium on Practical Formal
Approaches to Software Development, 2014

Copyright Prof. Dr. Shuichiro Yamamoto 2014

21

モデル図式からの保証ケースの導出



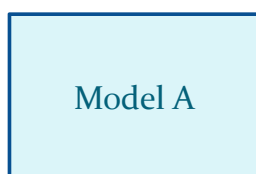
Copyright Prof. Dr. Shuichiro Yamamoto 2014

22

保証ケースの構成

頂点ゴール	主張:モデルが保証基準を満足する
概念と関係	モデルの構成要素と関係について、主張を分解する
概念と関係の種類	構成要素と関係の種類について、主張を分解する
概念と関係の実体	構成要素と関係の種類の実体について、主張を分解する
実体リスク	実体のリスクについて主張を分解する

モデルからのGSN自動作成方式



model $A = \langle \text{Concept Set}, \text{Relationship Set} \rangle$ に対して、以下の集合を作成。
 $\text{ConceptCategory}(A) = \{ C \mid \langle x, C \rangle \text{ is in ConceptSet of } A \}$
 $\text{RelationshipCategory}(A) = \{ C \mid \langle r, C \rangle \text{ is in RelationshipSet of } A \}$
 $\text{ConceptInstance}(C, A) = \{ x \mid \langle x, C \rangle \text{ is in ConceptSet of } A \}$
 $\text{RelationshipInstance}(C, A) = \{ r \mid \langle r, C \rangle \text{ is in RelationshipSet of } A \}$

ここで $\text{ConceptSet} = \{ \langle \text{Name}, Cc \rangle \mid Cc \text{ is a Concept category of Model} \}$
 $\text{RelationshipSet} = \{ \langle \text{Name}, Cr \rangle \mid Cr \text{ is a Relationship category of Model} \}$

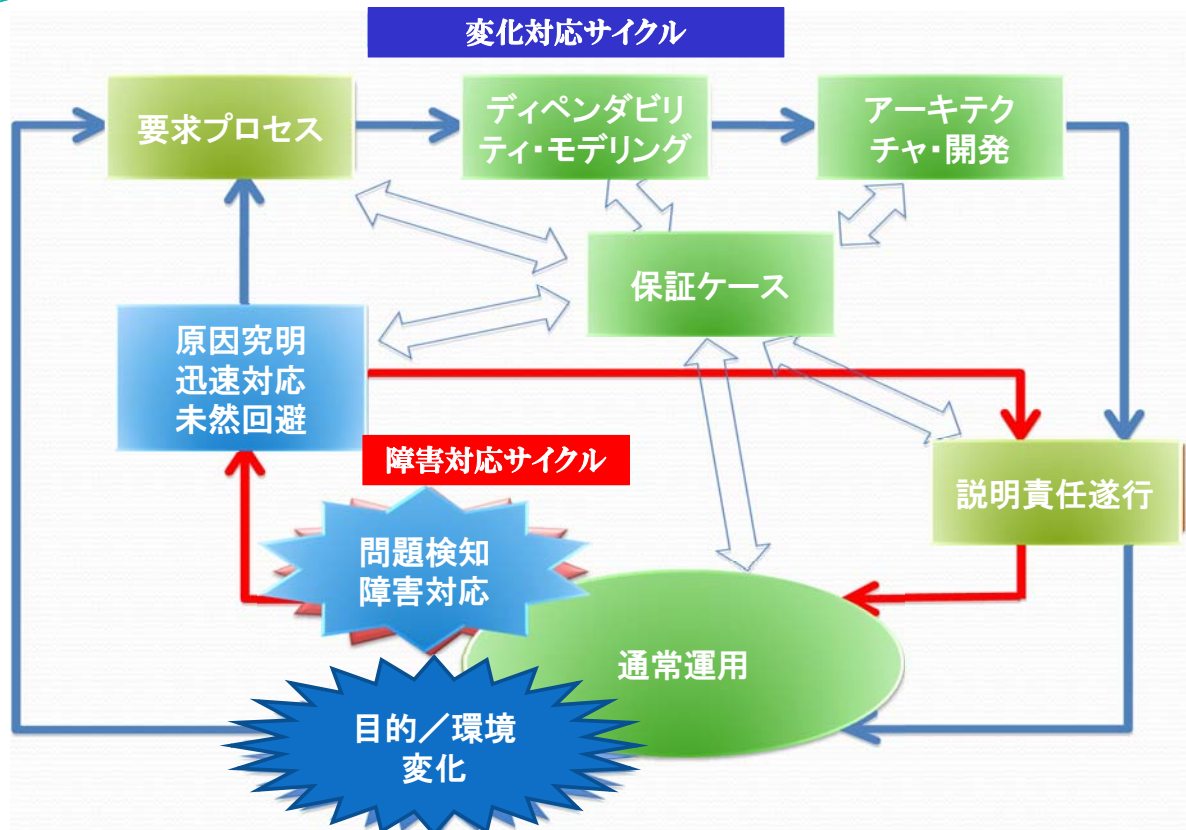
これらの集合に基づいて、GSN model D を以下の手順で作成
 頂点となる主張「モデルA が、基準を満足する」
 概念要素と関係で頂点主張を分解
 $\text{ConceptCategory}(A)$ と $\text{RelationshipCategory}(A)$ で、分解
 $\text{ConceptInstance}(C, A)$ と $\text{RelationshipInstance}(C, A)$ で、分解
 Aの実体要素ごとに、リスクを識別することにより、主張を分解

エンタープライズアーキテクチャの保証方法

- O-DA
- ArchiMate

O-DA標準の構成

章	内容	主な項目
1	はじめに	目的, 概要, 今後の方向性(改訂可能性)
2	定義	保証, 高保証アーキテクチャ, 保証ケース
3	O-DAフレームワーク	非機能要求としてのディペンダビリティ, 保証ケース開発, 説明責任, 障害対応サイクル, 変化適応サイクル
4	ガイドライン	保証ケース構造, 証拠と説明責任 , 保証ケース例 , 保証ケースの強化
5	形式手法	証拠としての形式手法
付録A	高保証アーキテクチャ開発手法	背景, TOGAFにおける高保証性フレームワーク , 保証内容メタモデル, アーキテクチャリポジトリ
付録B	DEOS	DEOSフレームワーク, DEOSサイクル, D-Case , 合成ディペンダビリティ , 合意形成と説明責任



Copyright Prof. Dr. Shuichiro Yamamoto 2014

27

高保証性 (Assuredness)

- 大規模で複雑なシステムのディペンダビリティを達成するために、O-DA標準では高保証性概念を導入している。
- アーキテクチャの実装が保証すべき要求に適合していることを確信するために、満足できる水準の証拠が提供されていることについて、システムのステークホルダが合意している状態を、**高保証性がある**という。

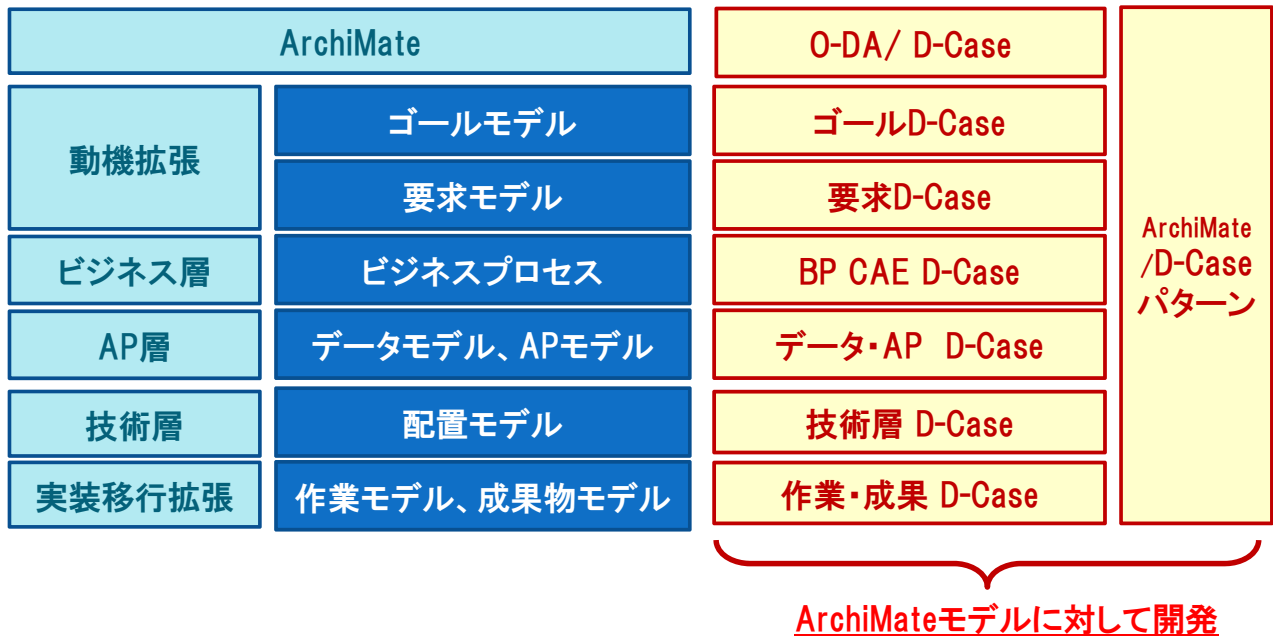
高保証アーキテクチャ

- 指定された保証すべき性質が実現されているアーキテクチャを高保証アーキテクチャ(Assured Architecture)という.
- 保証(Assurance)
 - アーキテクチャが高保証性を持つことを確認すること
- 保証ケース(Assurance case)
 - 保証するための議論構造を記録する成果物

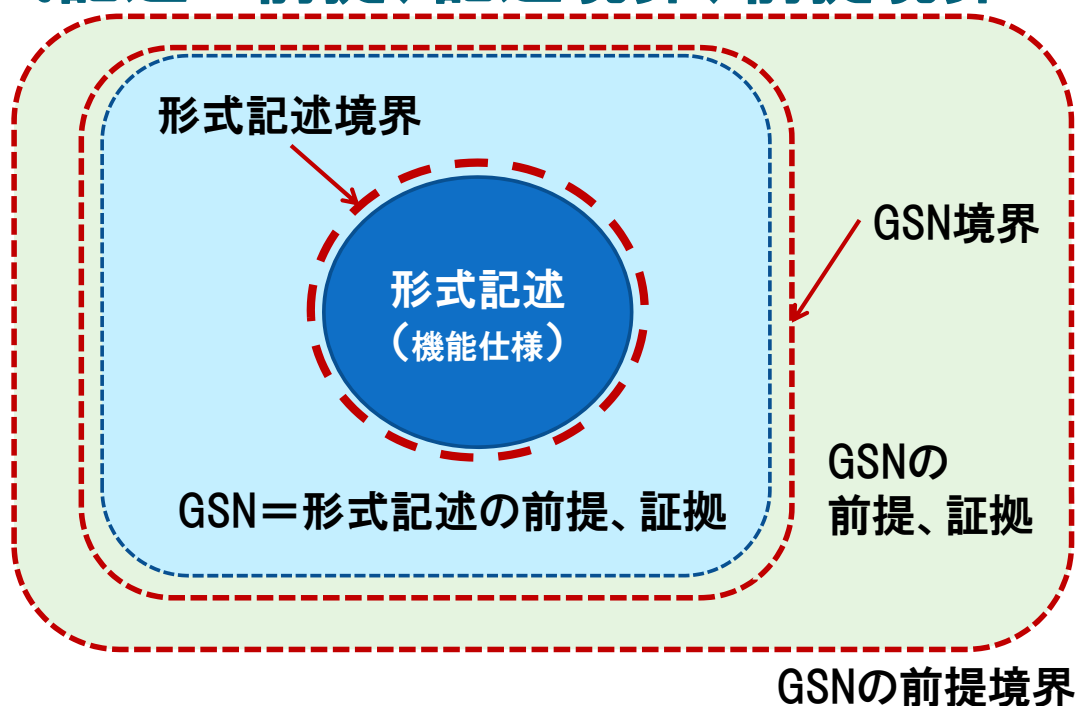
高信頼ADMの概要

工程	AADM
準備	①アーキテクチャリポジトリでの証拠文書と保証ケースの格納 ②ディペンダビリティ委員会での主張間の優先順位の合意
A.アーキテクチャビジョン	①ディペンダビリティスコープ定義 ②主張の定量評価尺度定義 ③保証ケース能力評価 ④ディペンダビリティパラメタ定義
B.ビジネスアーキテクチャ	①ディペンダビリティ原則定義 ②BA保証ケース作成 ③BA保証ケースレビュー
C.情報システムアーキテクチャ	①IA保証ケース作成 ②IA保証ケースレビュー
D.技術アーキテクチャ	①TA保証ケース作成 ②TA保証ケースレビュー
E.ソリューション	①BA, IA, TA保証ケースを統合 ②一貫性を確認
F.移行計画	①運用管理の保証ケース作成 ②ディペンダビリティパラメタ価値分析
G.実装監督	①保証ケースの証拠を作成 ②プロセス保証ケースの証拠を作成 ③網羅的に主張と証拠の関係を確認 ④運用に対する保証ケースをレビュー
H.アーキテクチャ変更管理	①運用保証ケースの証拠を管理 ②主張の不成立への対応策の確認 ③保証ケースによるリスク管理 ④保証ケースによる障害分析
要求管理	保証ケースの主張と要求の追跡管理

ArchiMateのためのD-Case方法論



形式記述の前提、記述境界、前提境界



形式手法とGSNによるアーキテクチャ保証方式

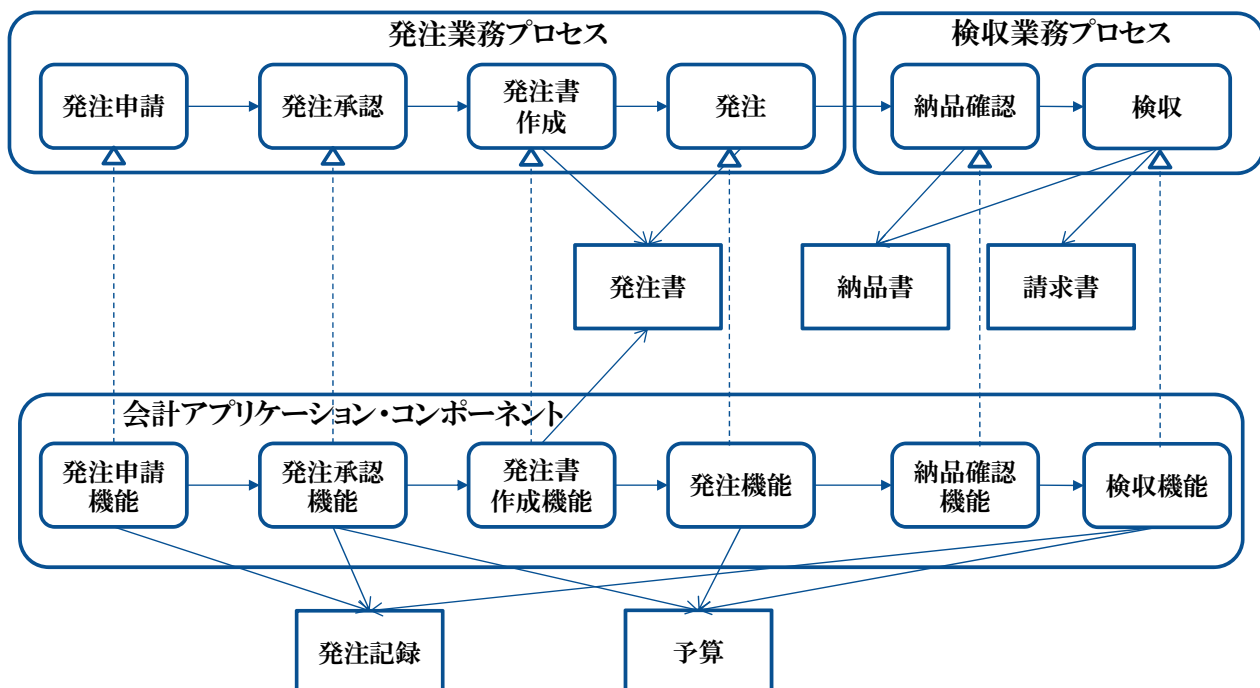
ArchiMate	FM (Formal Method)	GSN
ビジネス・アーキテクチャ		◎
BAとISAの関係		◎
情報システム・アーキテクチャ	◎AP機能仕様	○前提、証拠の説明
ISAとTAの関係		◎
技術アーキテクチャ		◎

BA: Business Architecture, ISA: Information System Architecture, TA: Technology Architecture

Copyright Prof. Dr. Shuichiro Yamamoto 2014

33

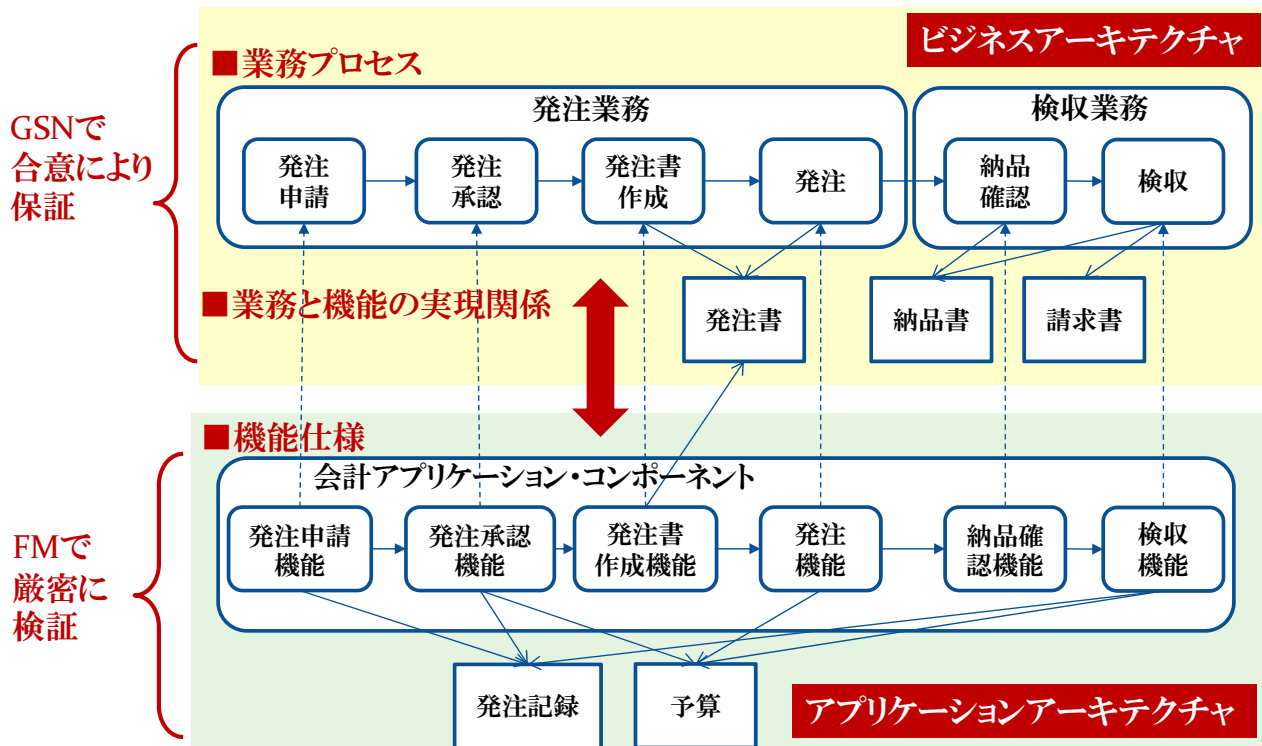
会計業務プロセスとアプリケーション



Copyright Prof. Dr. Shuichiro Yamamoto 2014

34

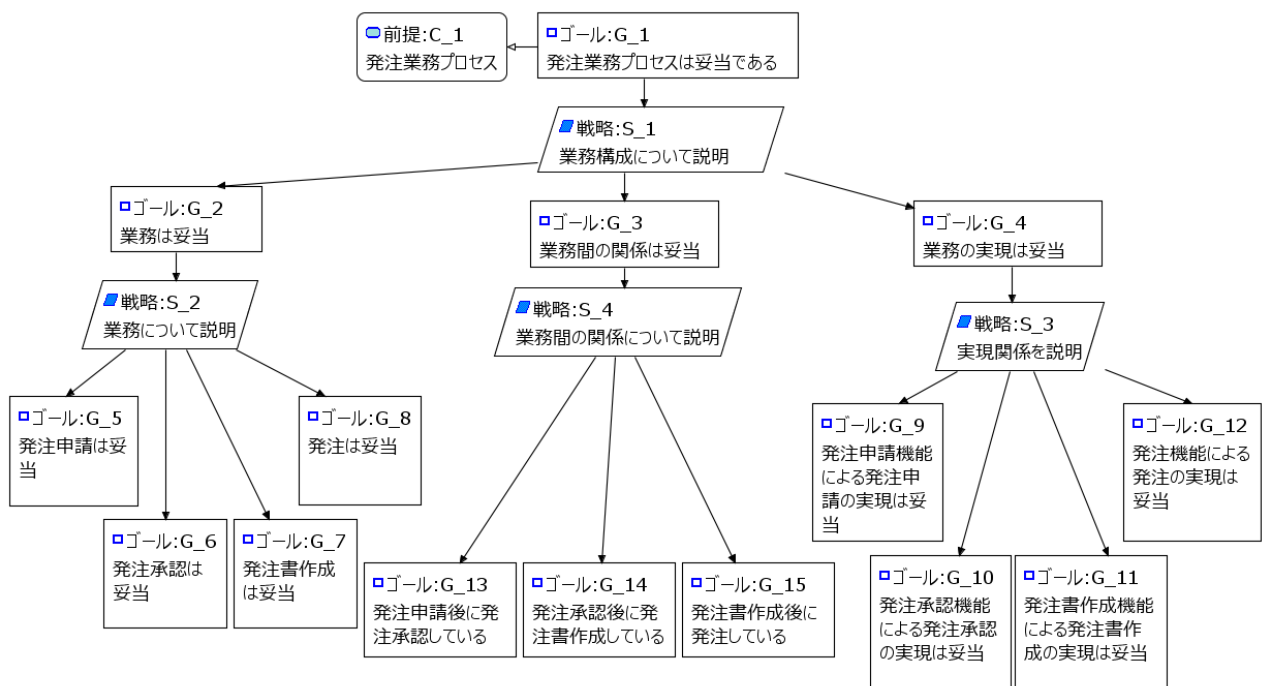
会計業務プロセスとアプリケーションの保証



Copyright Prof. Dr. Shuichiro Yamamoto 2014

35

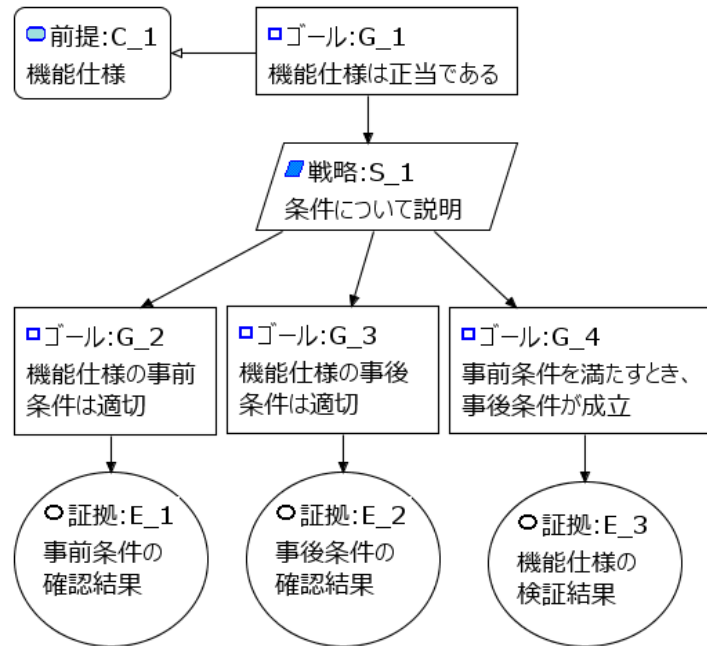
GSNによる会計業務プロセスの妥当性保証



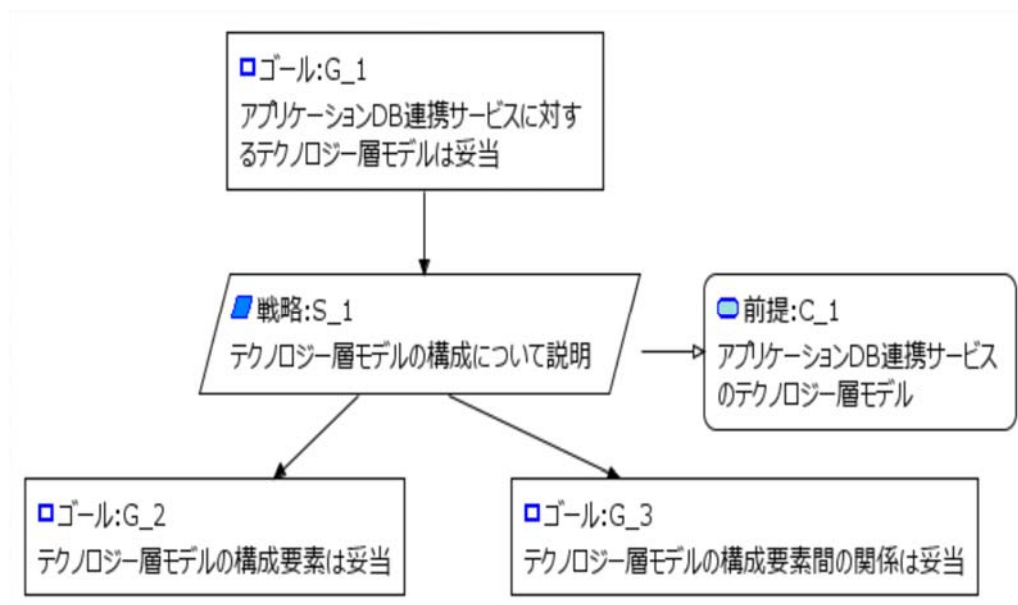
Copyright Prof. Dr. Shuichiro Yamamoto 2014

36

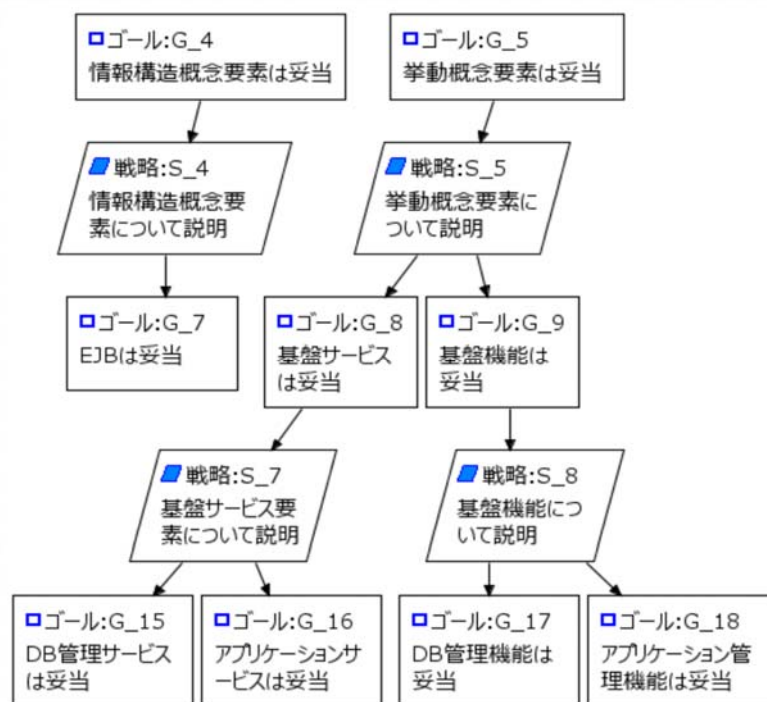
形式手法によるAP機能仕様の正当性検証



テクノロジー層モデル例の保証ケース(全体)



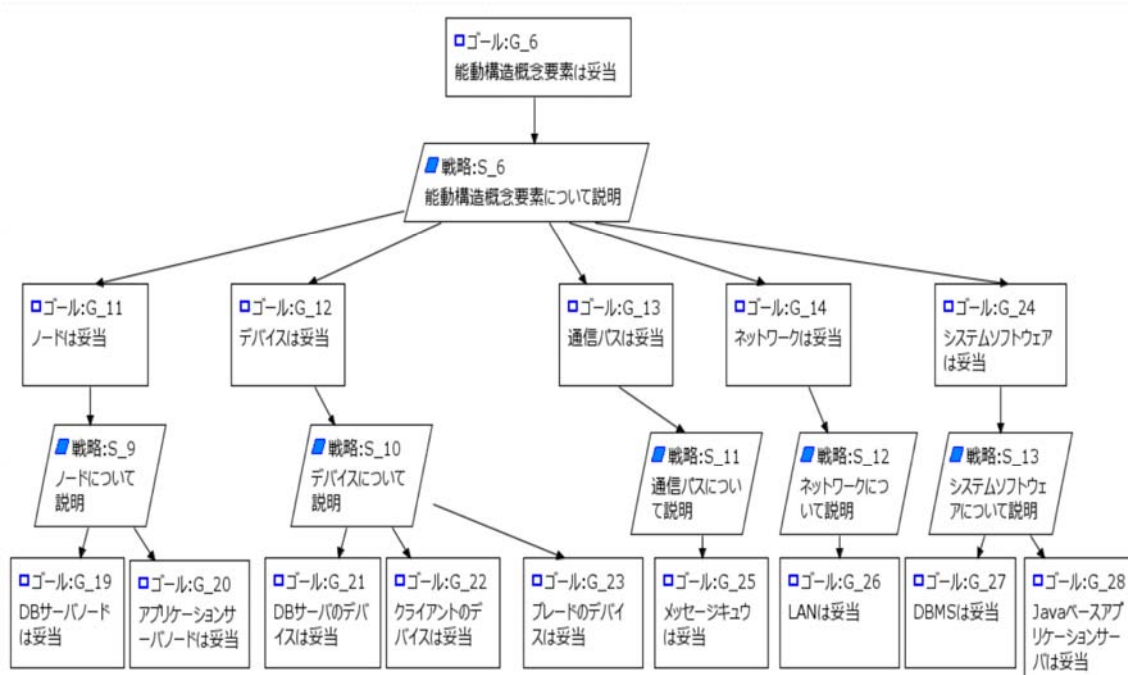
テクノロジー層モデル例の構成要素に対する分解(1)



Copyright Prof. Dr. Shuichiro Yamamoto 2014

39

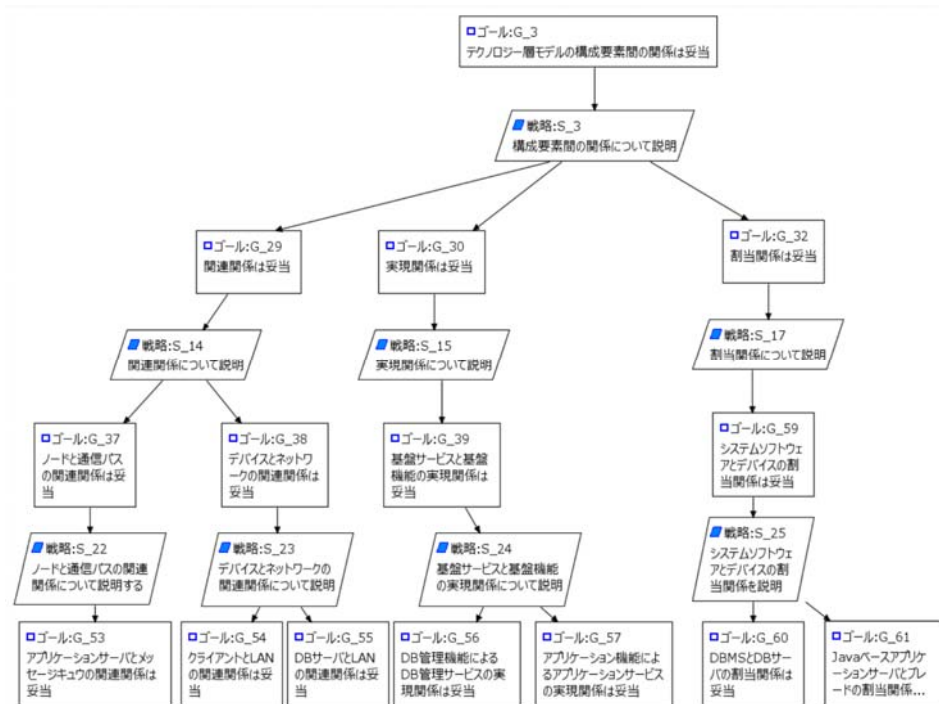
テクノロジー層モデル例の構成要素に対する分解(2)



Copyright Prof. Dr. Shuichiro Yamamoto 2014

40

テクノロジー層モデル例の構成要素関係に対する分解



Copyright Prof. Dr. Shuichiro Yamamoto 2014

41

USDMの保証方法

- USDMの例
- 要求理由一覧表
- 要求理由一覧表に基づく保証ケース

注) USDM: Universal Specification Description Manner

話題沸騰ポットの要求理由一覧表

対象	番号	要求	理由
蓋	p220	アイドル状態で蓋を閉じたら、水位を確認し、条件が合えば沸騰行為をする	蓋を閉めるという行為で加熱の指示をしたい
蓋	p221	蓋を開けたらロックが解除され、温度制御行為はしない	安全確保のため
沸騰ボタン	p230	保温行為中で沸騰ボタンを押すと、沸騰させる	保温状態から、沸騰させたいから
保温設定ボタン	p240	3種類の温度に設定できる	高温モード、保温モード、ミルクモード
解除ボタン	p250	給湯に対するロック/ロック解除機能を付ける	幼児などが給湯目的以外で誤って給湯ボタンを押してもお湯が出ないようにするため
給湯ボタン	p260	給湯ボタンを押すことによって給湯口から給湯する	ポットを持ち上げることなく、また給湯ボタンを無意識に押して期待していないタイミングでお湯が出て怪我(やけど)しないように、給湯したいから
タイマボタン	p270	タイマボタンを押すことで、時間を分でセットし、タイマを起動できる	簡単な操作でタイマを操作したいから
タイマボタン	p271	タイマボタンを長く押すことで、タイマをリセットし、タイマを停止できる	セットした時間をリセットしタイマを停止したい場合がある
タイマボタン	p272	ユーザーが指定した時間がきたら、ブザーを鳴らして知らせる	セットした時間がきたことに、ポットから離れたところにいても気が付きたいから
水位メータ	p280	ポット内の水量をインジケータで表示する	操作パネルを見て、ポット内の水量を知りたいから
沸騰行為	p310	水を沸騰させる	水を湯にしたいから
沸騰行為	p311	カルキ抜きをする	水道水に含まれる殺菌のための塩素を取り除くため
沸騰行為	p312	カルキ抜きが終わったら、保温行為をする	余計な電気を使用しないようにするため
保温行為	p320	設定されたモードの温度にポット内の水温を保持する	そのままにしておくと、水温は自然に気温近くまで下がってしまうので
温度制御停止行為	p330	沸騰行為も保温行為もできないときは温度制御はしない	温度制御ができないポットの状態ではヒータをon/offさせると危険だから
温度制御	p400	ヒータをon/offすることにより、水温を制御する。その制御方式は制御周期と操作量(%)により行う	水温を指定した温度に、一定に保ちたいから
エラー検知	p500	想定外の高温状態になった場合はエラーを検知し、ヒータ機能を停止する	加熱による火災等の危険を回避するため

Copyright Prof. Dr. Shuichiro Yamamoto 2014

43

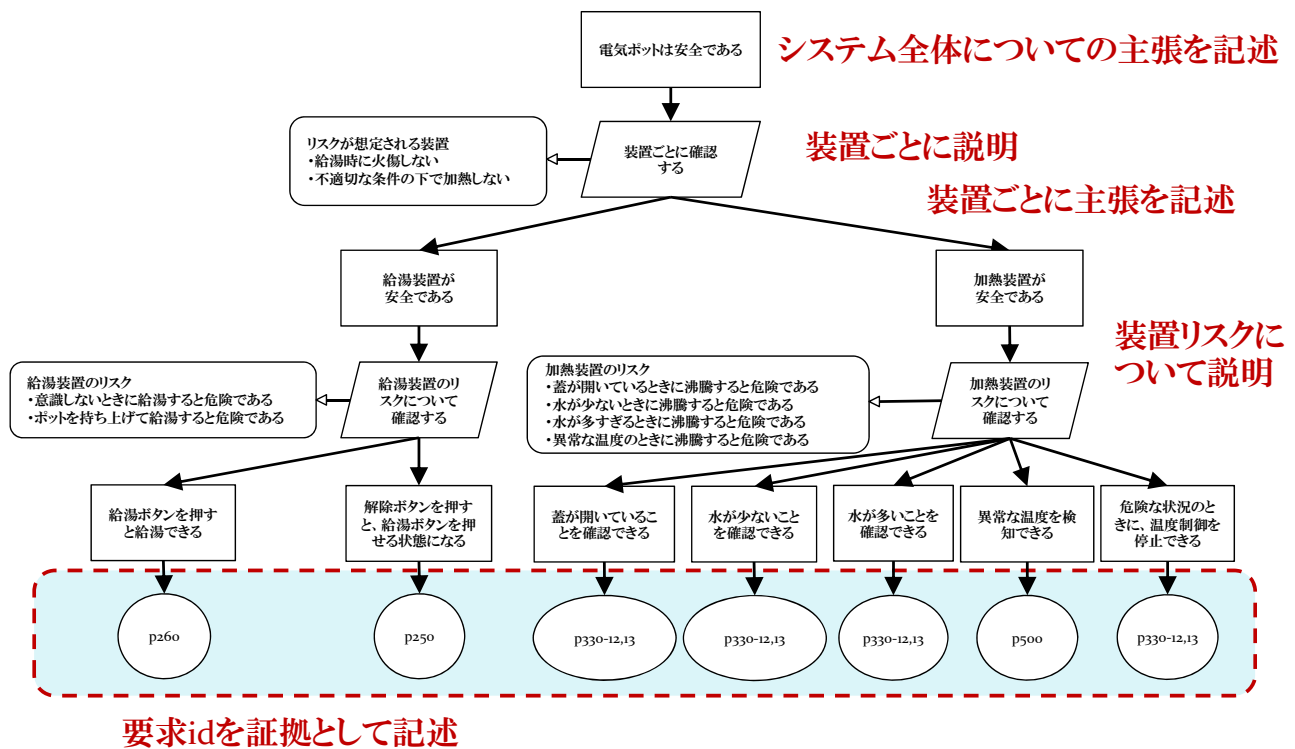
話題沸騰ポットの要求理由装置契機一覧表

対象	番号	理由	装置	契機
蓋	p221	安全確保のため	加熱装置	蓋が開く
解除ボタン	p250	幼児などが給湯目的以外で誤って給湯ボタンを押してもお湯が出ないようにするため	給湯装置	解除ボタンの押下
給湯ボタン	p260	ポットを持ち上げることなく、また給湯ボタンを無意識に押して期待していないタイミングでお湯が出て怪我(やけど)しないように、給湯したいから	給湯装置	給湯ボタンの押下
温度制御停止行為	p330	温度制御ができないポットの状態ではヒータをon/offさせると危険だから	加熱装置	貯水タンクの水量
エラー検知	p500	加熱による火災等の危険を回避するため	加熱装置	貯水タンクの湯温

Copyright Prof. Dr. Shuichiro Yamamoto 2014

44

保証ケースによる確認例 電気ポット



Copyright Prof. Dr. Shuichiro Yamamoto 2014

45

例外試験項目の十分性保証方法

- 関連研究
- 例外試験項目作成法
- 要求記述カード
- 要求逸脱分析表
- 例外試験項目作成実験

[1] Yamamoto, S., Morisaki, S., An Evaluation of Assuring Test Case Sufficiency using A D-Case Pattern, WOSD2014
 [2] Yamamoto, S., Morisaki, S., A Method to Design Exception Test Cases, ProMAC2014

Copyright Prof. Dr. Shuichiro Yamamoto 2014

46

例外試験の関連研究

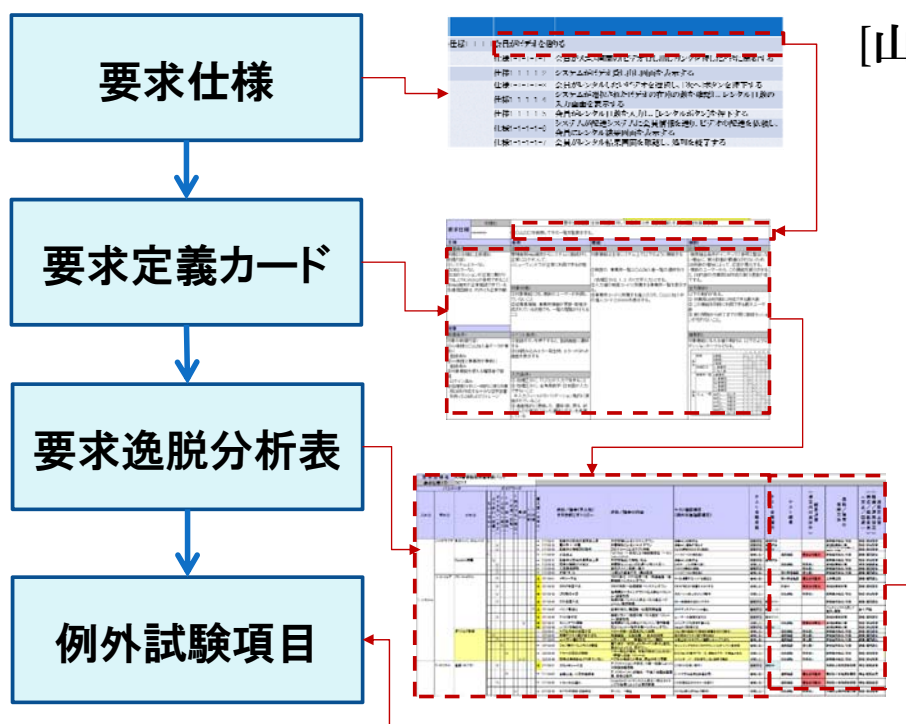
試験網羅性

要求に基づく試験項目作成

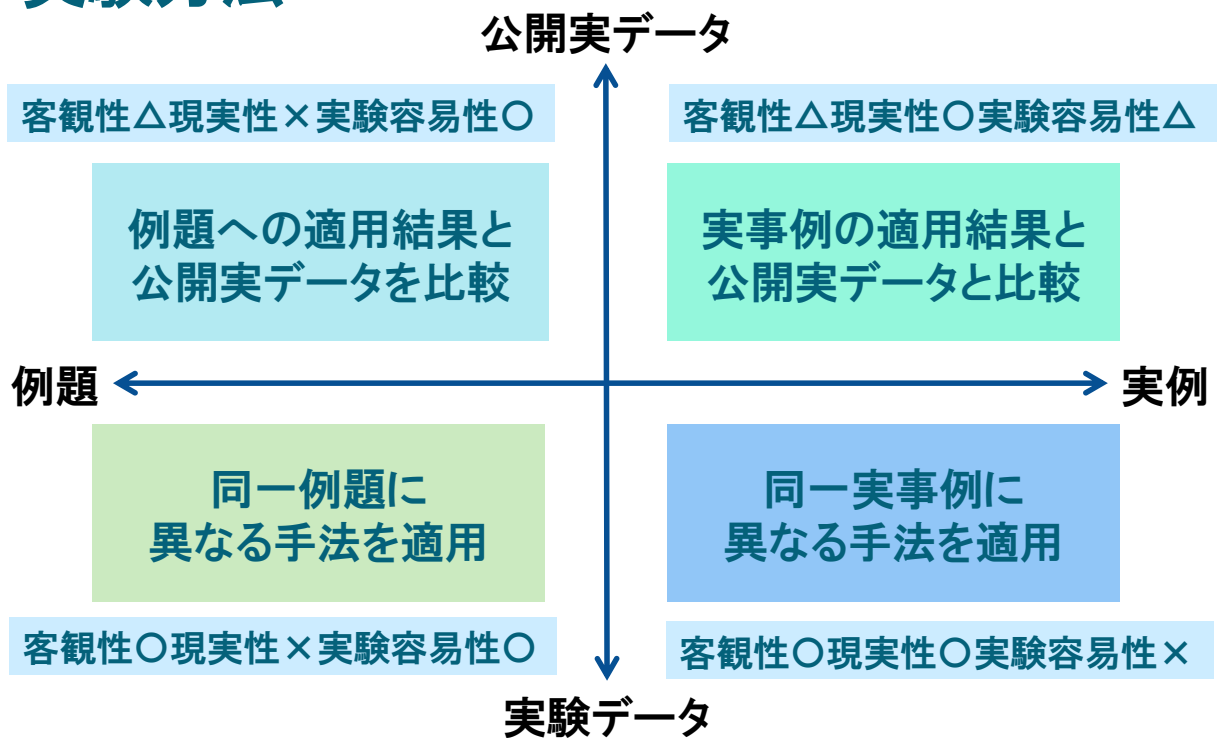
例外試験項目

関連研究	提案者[提案年]
試験網羅性実験	Maxison and others [2000]
ユーザ要求に基づくサービス試験項目	Pfaller and others [2006]
ユースケースによる対話試験項目作成	Benz [2007]
試験可能機能仕様によるシステム試験	Naik and others [2008]
試験項目の要求網羅性	Sommerville [2011]
要求制約による例外試験項目	Beque and others [2009]
事前条件と入力による例外試験項目	Atar and others [2010]
例外試験分類	Wang and others [2012]

例外試験項目の作成法



実験方法



Copyright Prof. Dr. Shuichiro Yamamoto 2014

49

実験概要

- 実験対象: 金融系情報システム
 - 年金管理パッケージの17機能
 - 要求記述
 - 要求逸脱分析表
 - 例外試験項目
- 実験参加者: 2名
 - 試験経験知識
 - 保証ケース知識
- 実験方法
 - 実事例に対する適用結果を公開データと比較
 - IPA/SEC, ソフトウェア開発データ白書2010-2011

Copyright Prof. Dr. Shuichiro Yamamoto 2014

50

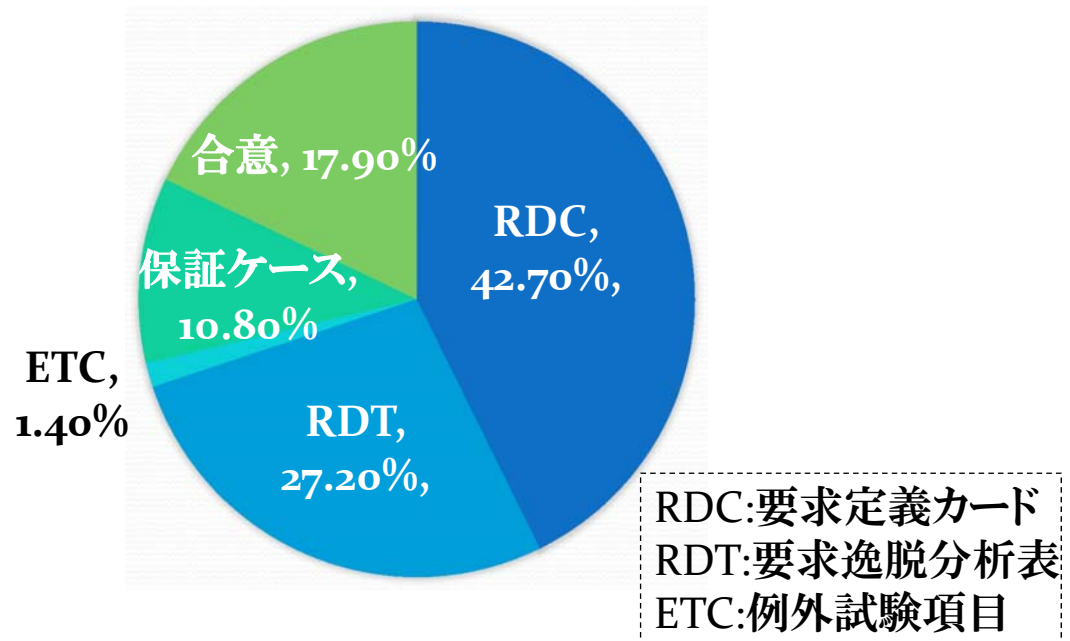
要求記述項目

項目	説明
主体	要求をみたそうとするシステム
対象	システムが操作する資源(データ, 被制御装置など)
前提条件	主体および対象の範囲と存在についての条件
条件	機能を実行するときの主体の状態と対象の状態 機能の実行契機となるイベント内容とタイミング条件 入力の内容についての条件
機能	入力に基づいて出力を生成する処理
制約	機能実行上のシステムが満たすべき挙動条件 応答制約: 応答事象内容とタイミングの制約条件 出力制約: 出力内容についての制約条件 値制約: 出力の値についての制約条件

Copyright Prof. Dr. Shuichiro Yamamoto 2014

51

工数の内訳

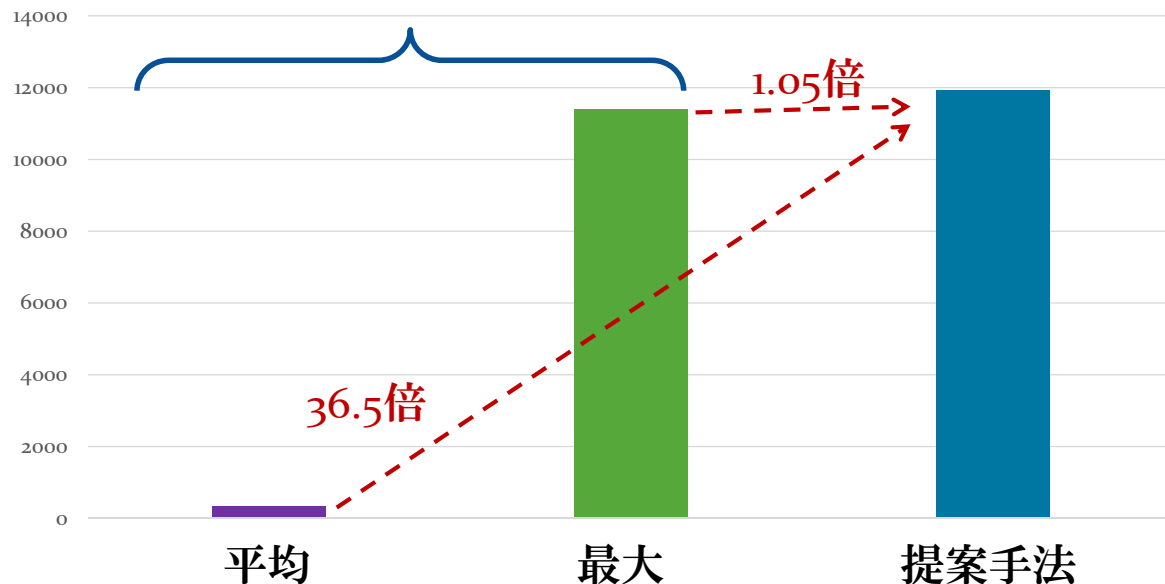


Copyright Prof. Dr. Shuichiro Yamamoto 2014

52

試験項目生産性の比較

テストケース生産性(K時間あたりの結合テスト項目数)の参考値
(IPA/SEC, ソフトウェア開発データ白書2010-2011)

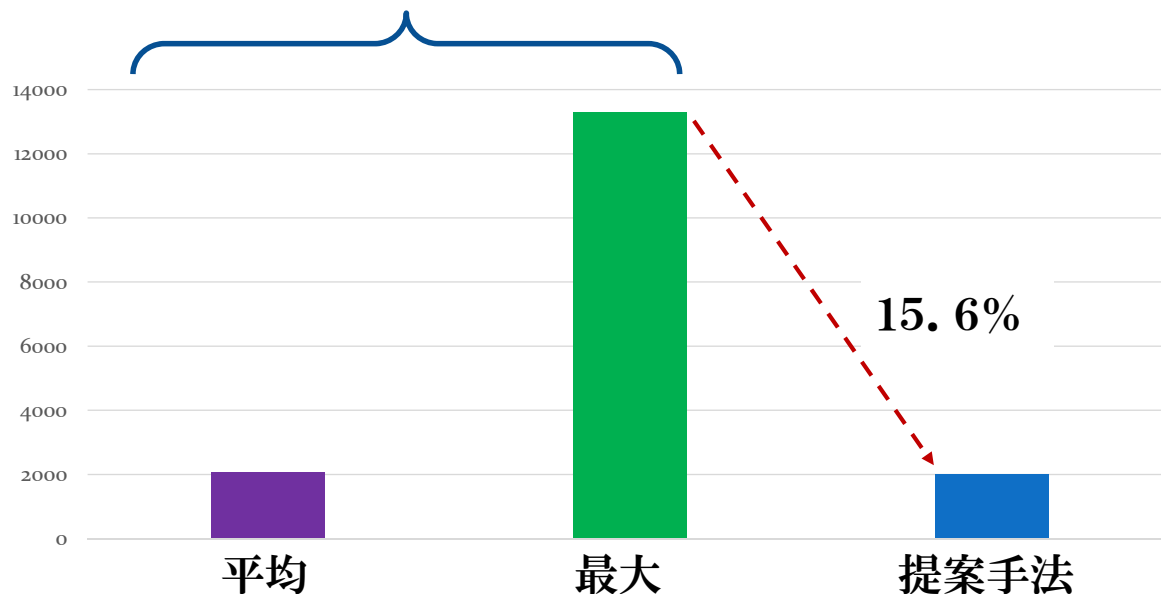


Copyright Prof. Dr. Shuichiro Yamamoto 2014

53

FPテストケース密度(件/KFP)の比較

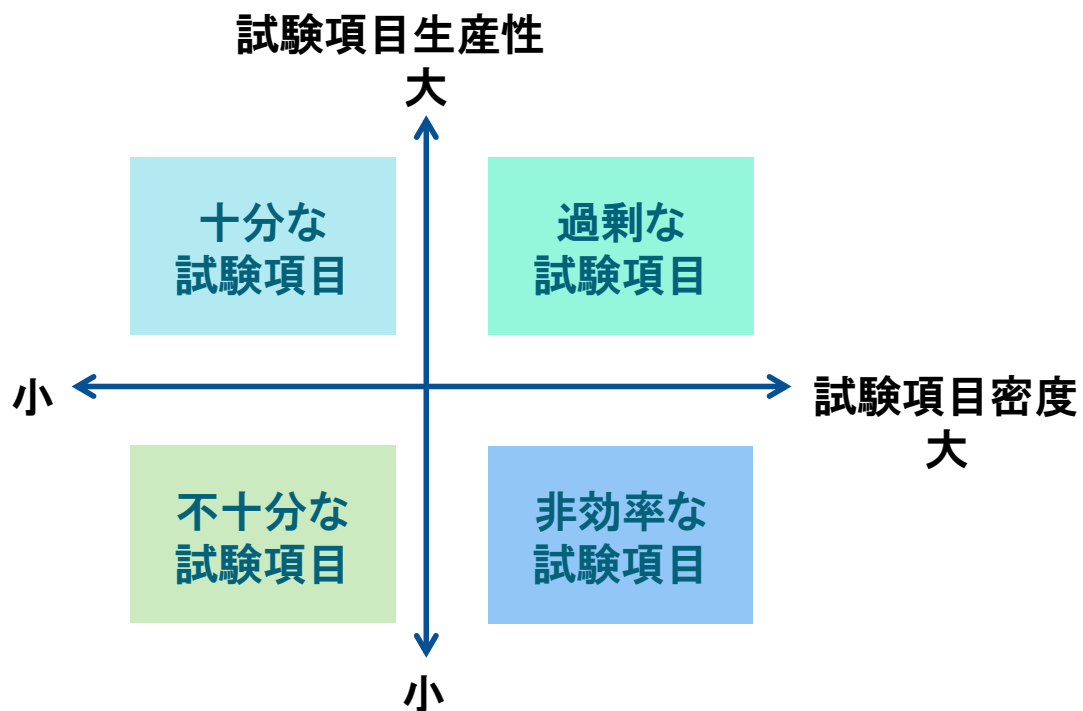
テストケース密度(KFPあたりの結合テスト項目数)の参考値
(IPA/SEC, ソフトウェア開発データ白書2010-2011)



Copyright Prof. Dr. Shuichiro Yamamoto 2014

54

試験項目の分類



Copyright Prof. Dr. Shuichiro Yamamoto 2014

55

まとめ

- 経緯
- 保証ケース超入門
- モデル図式保証方法論の構築
- モデル図式保証方法の一般化
- エンタープライズアーキテクチャの保証方法
- USDMの保証方法
- 例外試験項目の十分性保証方法

Copyright Prof. Dr. Shuichiro Yamamoto 2014

56