


# モデル検査SPINの実システムに対する適用と フォーマルメソッド導入ガイダンス

---

2010年7月10日  
第5回 CSP研究会

 株式  
会社 三菱総合研究所

石黒 正揮, Ph.D.

## ■ 内容

### ■ プロジェクトの背景と概要

- ガイダンスの位置付け
- ガイダンスの対象範囲
- ガイダンス構成と作成状況

### ■ ケーススタディの紹介

- ICカードベース入退室管理システム
- ブルーレイディスク

### ■ 「フォーマルメソッド導入ガイダンス」の紹介

- モデリングと形式検証の概念のイントロダクション
- プロセスと成果物の関係
- 要求仕様と検証性質
- モデリングプロセスの手順化
- 費用対効果に関する分析
- 形式手法選択のためのガイド

### ■ まとめ

# 背景

---

## ■ 背景

### ■ ソフトウェアの不具合等に起因する大規模な損害事故の増加

- 証券取引所の株式売買システムの障害による市場の大混乱。
- 携帯電話の不具合による大量回収騒ぎ。回収費用は120億円。
- 航空会社のシステム障害。約7万人に影響。計63便が欠航、357便が遅延。
- NASA火星探査衛星のシステム障害。予算規模150億円。

### ■ 情報システム事故に係る潜在リスクは日本の上場企業全体で29兆円

- 経済産業省「グローバル情報セキュリティ戦略」報告書<sup>※</sup>において試算。

### ■ 欧米におけるフォーマルメソッドの産業応用の進展

- パリ地下鉄、ロワシー空港のシャトルの制御システムにおいてBメソッドを用いた設計仕様の詳細化検証とAdaプログラムの自動生成。
- 船舶向け通信システムに対して、CHARMYを用いて検証性質を抽出し、SPINを用いてモデル検査により不具合を検出(イタリア)

### ■ 国際標準、政府調達基準への採用の動き

- IT製品の情報セキュリティに関する国際標準コモン・クライテリア(ISO/IEC 15408)、電子製品の機能安全(IEC 61508)等において形式手法が認定基準に採用。
- WTO GPA政府調達協定で国際標準の採用が推奨。
- 欧米への製品輸出における非関税障壁の懸念が拡大。
- ハイブリッド車のブレーキ制御ソフトウェアに関する問題。ISO26262の標準化目前。

※ [1] Masaki Ishiguro, Hideyuki Tanaka, Kanta Matsuura, The Effect of Information Security Incidents on Corporate Values in the Japanese Stock Market, Workshop on the Economics of Securing the Information Infrastructure (WESII 2006) の研究成果に基づき経済産業省が試算。

[2] 経済産業省 リスク定量化ワークショップ, 2007年, “IT事故と情報セキュリティ対策が企業価値に与える影響分析”, 石黒正揮 (三菱総合研究所), 松浦幹太 (東京大学), 田中秀幸 (東京大学)

## ■プロジェクトの概要

**名称:** 経済産業省 新世代情報セキュリティ研究開発事業

「モデル検査による組込みソフトウェア検証とモデリング・パターン化の研究開発」

2008年10月～2011年3月(予定)

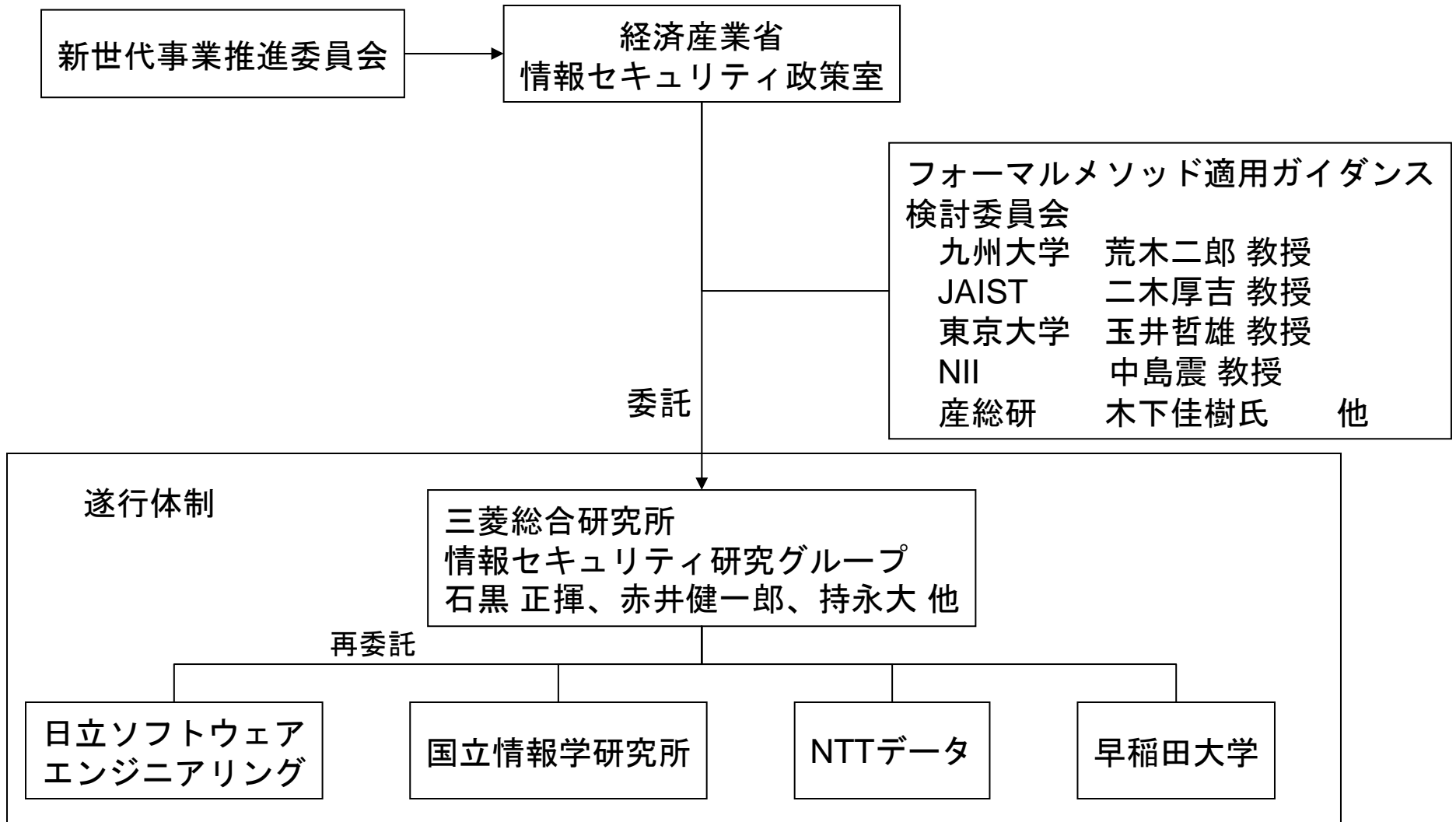
**目的:** ソフトウェア開発における形式手法の導入を促進するために、導入の障害に対する解決アプローチや実用化のノウハウを示した「フォーマルメソッド導入ガイダンス」を作成する。ガイダンスを作成するために実ソフトウェアに対するケーススタディを実施する。開発プロセスへの形式手法の導入方法

### 実施内容

- 実システムのケーススタディの実施
  - ブルーレイディスクの操作制御ミドルウェア
  - ICカードベースの入退室管理システム
- 形式モデリングのノウハウ
- ツール・手法の比較
- 形式手法導入の費用対効果

**実施者:** 株式会社三菱総合研究所、日立ソフト、NTTデータ、  
国立情報学研究所、早稲田大学

## ■ 実施体制



# フォーマルメソッド導入ガイダンスの位置付け

---

■ 形式手法の普及に係わる課題と本ガイダンスの位置付け  
 ～取っ付き難さを解消することが最大の解決策～

	課題	本ガイダンス	既存文献の例
管理面	導入の意思決定に必要な情報の不足	<ul style="list-style-type: none"> <li>●費用対効果分析</li> <li>●効果影響分析</li> <li>●具体的な効果を例示</li> </ul> (効果の見える化)	(なし)
	形式手法に対する認識と現実のズレ	<ul style="list-style-type: none"> <li>●形式手法の導入における留意点</li> </ul> (意識のギャップを解消)	Bowen: Ten Commandment Revisited(形式手法の十戒) NASA Guidebook
	形式手法選択の情報の入手の困難さ	<ul style="list-style-type: none"> <li>●解説文献のガイド</li> </ul> (既存の文献を活用して、適切な手法の選択を支援)	ソフトウェア科学 解説論文、日経BP組込みソフトウェア形式手法特集
技術面	モデリングに対する理解の困難性	<ul style="list-style-type: none"> <li>●モデリング・アプローチの枠組みを整理</li> <li>●形式検証プロセスの具体的な手順化</li> </ul> (既存の文献を補強)	(SPINによる設計モデル検証)
	状態爆発等の技術的障害 (モデル検査の場合)	<ul style="list-style-type: none"> <li>●検証性質を意識した抽象モデリングの手順</li> <li>●状態爆発の発生原因と解決策の例の提示</li> </ul> (既存の文献を補強)	(Ben Ari, Principles of SPIN Model Checker)



■ 本ガイドのスコープ(モデル検査のパート)  
 ~ 管理者および開発上流工程に重点を置く ~

業務プロセス		対象者	
		管理者向け	技術者向け
管理	事前準備	<ul style="list-style-type: none"> <li>● 形式手法の概念と特徴</li> <li>● 費用対効果</li> <li>● 手法の比較情報</li> </ul>	<ul style="list-style-type: none"> <li>● 形式手法の概念と特徴</li> <li>● 手法の比較情報</li> <li>● ケーススタディ</li> </ul>
	プロジェクト管理	<ul style="list-style-type: none"> <li>● 費用対効果</li> <li>● 導入効果の具体例</li> <li>● 形式手法導入の留意点</li> <li>● 設計検証プロセス・アプローチ</li> </ul>	該当なし
開発工程	要求分析、 設計	該当なし	<ul style="list-style-type: none"> <li>● 設計検証プロセス・アプローチ</li> <li>● モデルの抽象化</li> <li>● 検証性質の記述パターン</li> <li>● SPINオプションの使い方</li> </ul>
	実装		(メルコパワー:モデル検査ガイドブック)
	テスト		(テストカバレッジの計測(VDMtools))

## ■「フォーマルメソッド導入ガイダンス」全体構成と作成状況 ～来年度の完成および普及促進を目指して作成中～

<b>I. 組織管理者編</b>		}	導入検討者用
(1) フォーマルメソッド導入の効果・メリット ソフトウェアに求められる信頼性の要件 フォーマルメソッド独自の効果			
モデル検査導入のコスト・効果	(一部、2008年度)		
<b>II. プロジェクト管理編</b>		}	開発工程への 導入方針検討用
(2) フォーマルメソッドの俯瞰と本ガイドラインの位置付け	(2009年度)		
SPINと他のフォーマルメソッドの特徴/適用対象の比較	(2009年度)		
モデル検査系の特徴比較	(2009年度)		
ツールの選択法			
(3) ソフトウェア開発プロセスへの導入方法 (モデル検査の場合)			
(4) 導入研修の進め方			
<b>III. 開発技術者編</b>		}	適用時のノウハウ提供
(5) 検証手法の適用法 (Cook Book)			
(5.1) モデリングのパターン化	(2009年度)		
(5.2) 検証性質の記述法	(一部、2008年度)		
(5.3) モデルの抽象化・有界化	(一部、2008年度)		
(5.4) 不具合解析			
(5.5) ツールの使い方			
(6) ケーススタディ			
(1) ブルーレイディスク	(2008年度)		
(2) 入退室管理システム	(2009年度)		
<b>IV 付録</b>			
関連文献			
リンク集			

## ケーススタディに関する現状と成果

---

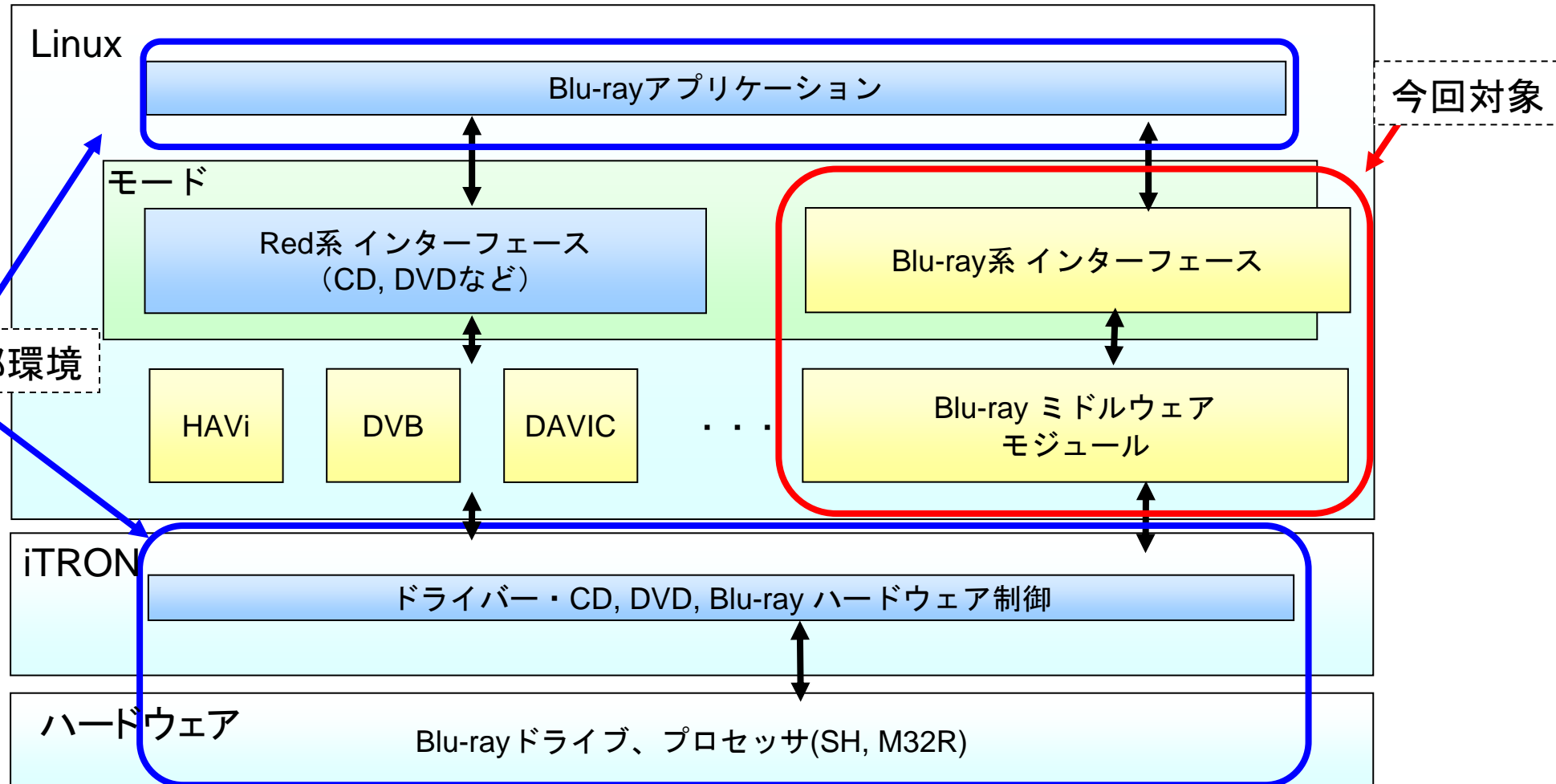
## ■ ケーススタディの概要

事例	検証の観点	検証性質(例)	検証結果・ 不具合発見等	システムの規模
ブルーレイ ディスク (2008年度)	<ul style="list-style-type: none"> <li>● ディスク操作APIの制御アルゴリズムの安全性。モジュールレイヤー間の整合性検査。</li> </ul>	<ul style="list-style-type: none"> <li>● 要求されたAPIは常に実行される。</li> <li>● APIが実行された後、常に待機状態に戻る。</li> </ul>	<ul style="list-style-type: none"> <li>● API要求と割込みの同時発生に関わるデッドロックの検出。</li> <li>● 再生API要求の不実行の発見</li> </ul>	システム全体： 31万2000ステップ 検証対象： 20万5000ステップ
入退室管理 システム (2009年度)	<ul style="list-style-type: none"> <li>● 要求仕様の妥当性確認</li> <li>● 運用ポリシーの明確化</li> </ul>	<ul style="list-style-type: none"> <li>● 待合室に入った人はいずれは閲覧室に案内される。</li> <li>● 閲覧室が空室の時以外決して案内は変更されない。</li> </ul>	<ul style="list-style-type: none"> <li>● 入室待ちリストのバッファ溢れの発見。</li> <li>● ある時点で、許可されない人が閲覧室のICカード認証をパスしない。</li> </ul>	システム全体： 3万1000ステップ 検証対象： 1万ステップ

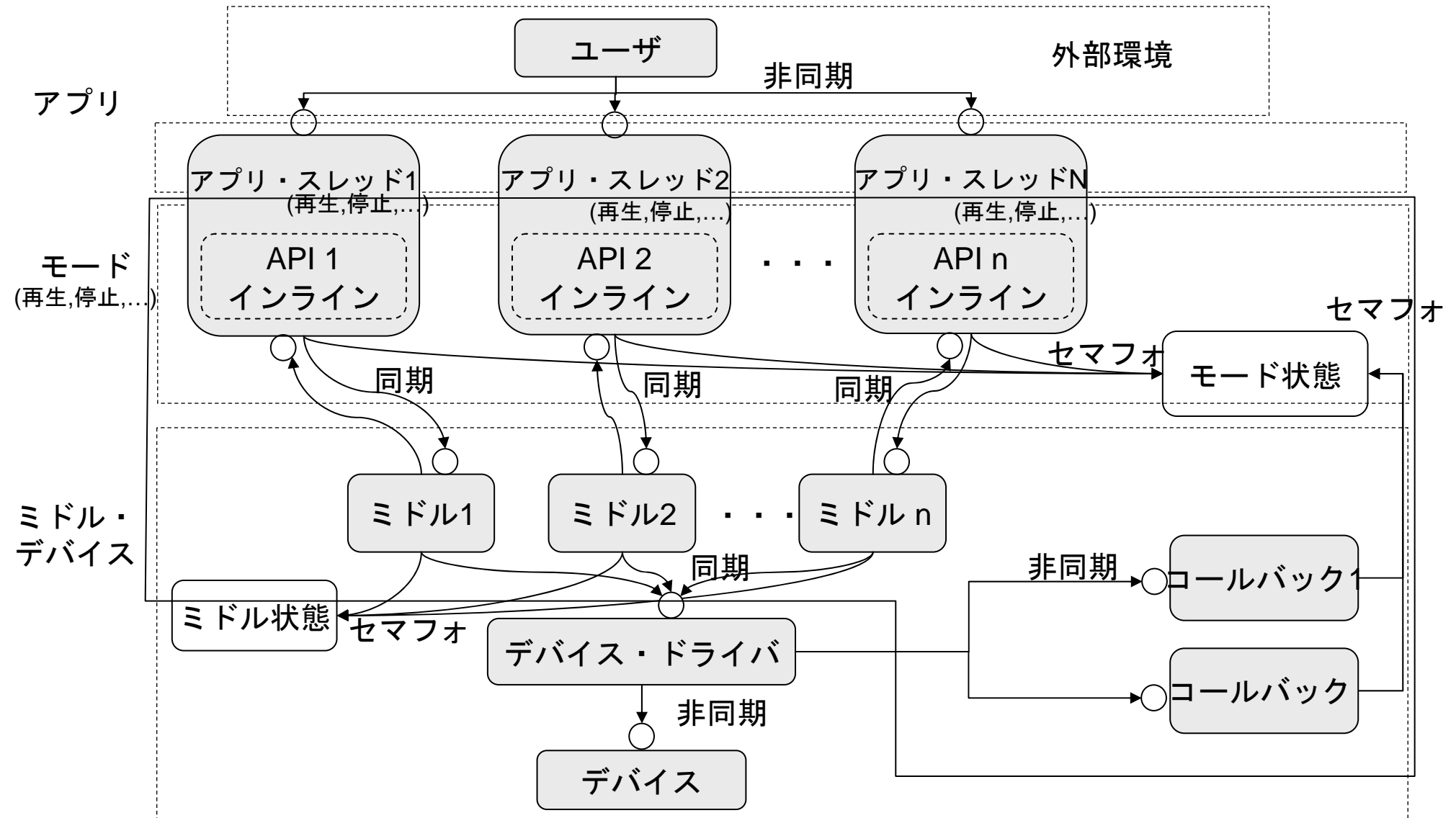
## ■ 検証対象システム(ブルーレイディスク)

日立ソフトウェアで開発中のブルーレイディスク・ミドルウェア

Blu-Rayアプリケーションからの操作APIの制御部のモデル検査



## ■ Blu-Rayディスクのモード制御プロセスの構成



## ■ミドルモジュールの状態遷移表

- ミドル関数の呼出しとミドル状態に応じて状態遷移が決まる。
- デバイス割込みにより非決定的な状態遷移も発生する。

### ミドルの状態

割込み	ミドル関数の一覧	ミドル状態							
		停止状態	再生状態	再生移行中	ポーズ中	早送り中	正スロー中	早戻し中	逆スロー中
デバイス割込み		/	☆	☆	/	☆	☆	☆	☆
first_playback 再生処理CALL		☆	☆	/	☆	☆	☆	☆	☆
再生(再開)処理CALL		☆	×	×	×	×	×	×	×
Chapter Search実行時処理CALL		×	☆	/	☆	☆	☆	☆	☆
停止処理CALL		/	☆	☆	☆	☆	☆	☆	☆
Pause押下処理CALL		×	☆	×	☆	☆	☆	☆	☆
順方向再生速度変更処理CALL		×	☆	×	☆	/	☆	☆	☆
逆方向再生速度変更処理CALL		×	☆	×	☆	☆	☆	/	☆
順方向再生速度変更処理(スロー)CALL		×	☆	×	☆	☆	/	☆	☆
逆方向再生速度変更処理(スロー)CALL		×	☆	×	☆	☆	☆	☆	/
順方向コマ再生処理CALL		×	/	×	/	/	/	/	/
トリックプレイ解除処理CALL		×	/	×	☆	☆	☆	☆	☆
アングル変更処理CALL		×	/	×	×	/	×	/	×
Audio Language セットCALL		/	×	×	×	×	×	×	×

## ■ 検証内容と結果

### 想定外の不具合を発見:

#### ■ 到達性解析によりにデッドロックを発見。

APIの操作要求とデバイスからミドルモジュールへの割込み処理の並行処理においてデッドロックが発生。

#### ■ 検証性質1

モード状態が「停止中」の時、いつでも「再生」を要求すれば、いずれは再生が実行される。

一般的な機能要求だが、反例を発見→ライブロックの問題。

ユーザの操作イベントによっては、期待した機能に到達しない。(「再生移行中」のタイミング)

### 期待通りの性質を証明:

#### ■ 検証性質2

目的: モードモジュールとミドルモジュールの状態のズレにより問題が生じないか検証

性質一般形: ミドルが処理できない要求をモードが流しても、いつかは待機状態に戻る。

具体形: いつでも、ミドル状態が「停止中」で、かつ、ミドルに「ポーズ」要求が来ても、いつかは待機状態に戻る

$\square((\text{Mid\_state} == \text{mid\_stop}) \ \&\& \ (\text{Mode\_req} ? [\text{midf\_pause}(\text{ch1})]) \rightarrow \langle \rangle (\text{Mode\_state} == \text{mode\_stop}))$

→ 正しさを証明。

#### ■ 検証性質3

性質一般形: ミドル状態において処理できない要求を、モードは流さない。

具体形: ミドル状態が「再生移行中」で、かつ、モードがミドルに「ポーズ」要求を出していることはない。

$\square!(\text{mid\_state} == \text{mid\_to\_playing}) \ \&\& \ (\text{Mode\_req} ? [\text{midf\_pause}(\text{ch1})])$

→ 反例検出を目的としたが、最初、反例が検出されなかった。

LTLの記述に誤りを発見(要求イベントと状態種別の混同)

#### ■ 検証性質4

モード状態とミドル状態はずれることが無い。

→ モデルの記述の誤りがあり、修正により反例を検出することができた。



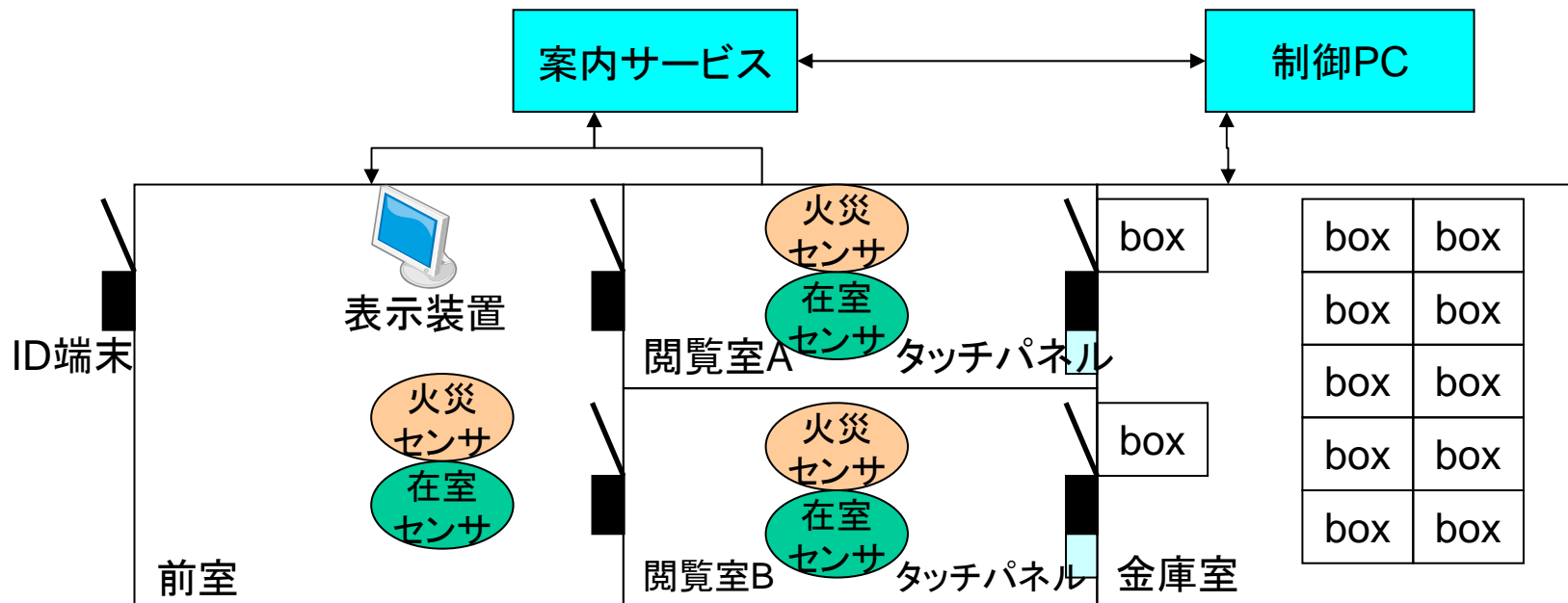
# 入退室管理システムへの適用

---

## ■ 検証対象システム概要 (ICカード入退室管理システム)

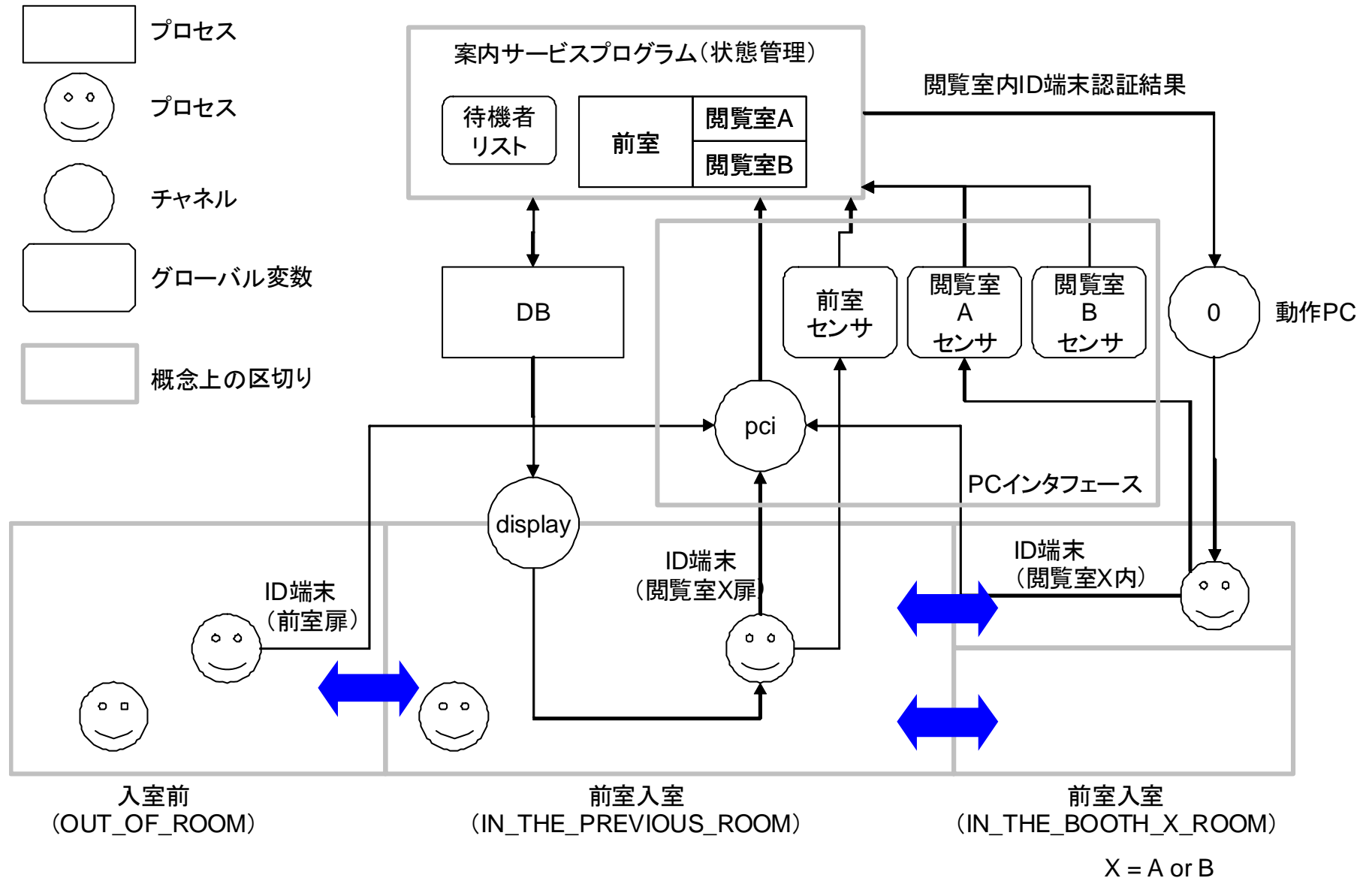
前室(待合室)、閲覧室、金庫室の3つの部屋で構成

- 前室、閲覧室への入室にはIDカードを用いた認証が行われる
- 閲覧室では、室内でのIDカード認証とタッチパネル操作により、金庫室内の該当するボックスを閲覧することができる



検証対象システムの概略図

## ■対象システムの構成 ～概念的なモデルの構成～



# ■ 状態遷移表の例(案内サービス・プロセス) イベント

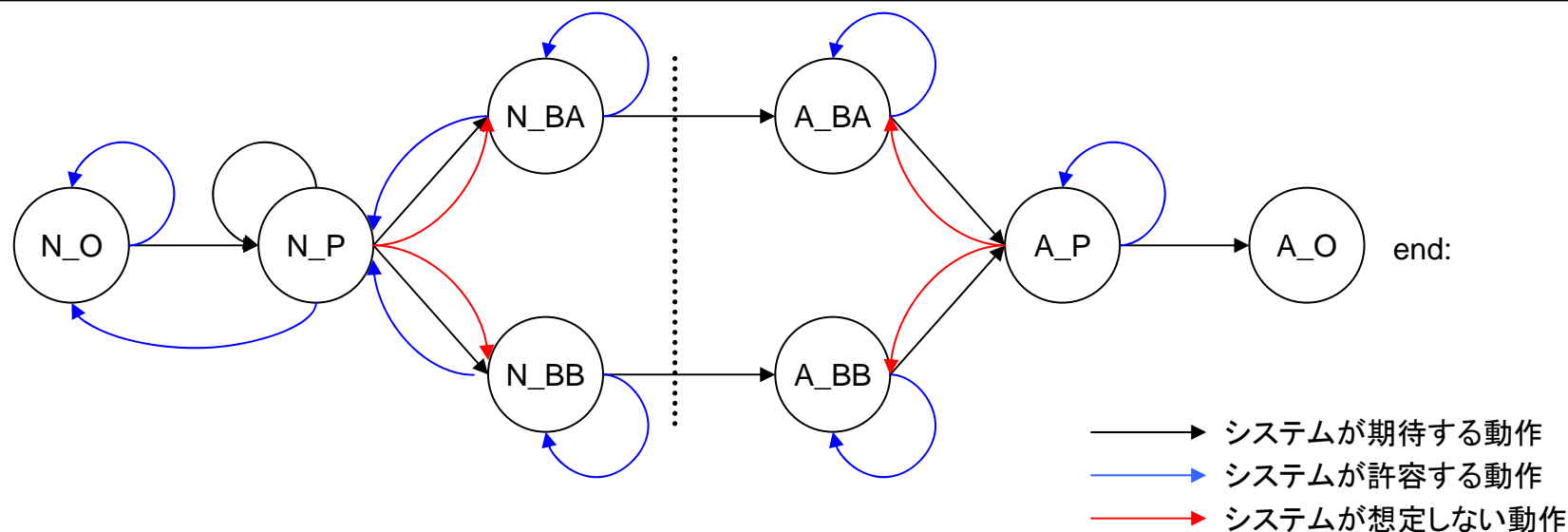
状態

		種表示 (種別表示されている NAI)	配室案内内	配室案内内	乗客の チェックイン/パス 類のがある、種別 が案内されている 場合	乗客の乗船 開始OK	乗客の乗船 終了OK
P.F.D	乗客・前室乗降車 到着						
	乗客・L10P						
	乗客・スキャップ						
	乗客・L10D						
	乗客・前室セン サーON						
	乗客・L10WAT						
INTO	乗客・L10WAT						
	乗客・L10WAT						
WAIT	乗客・WATCH						
	乗客・L10.DOOR.A 内からの						
WATCH	乗客・L10.DOOR.A 配室案内内からの						
	乗客・L10.DOOR.B 配室案内内からの						
P.F.P	乗客・LEAVE						
	乗客・表示状況確認 (表示されているNAI →監視 WAIT)						
TO.DOOR.A	乗客・配室案内確認 乗客・L10.DOOR.A 配室案内内からの						
	乗客・L10.DOOR.A 配室案内内からの						
TO.DOOR.B	乗客・配室案内確認 乗客・L10.DOOR.B 配室案内内からの						
	乗客・L10.DOOR.B 配室案内内からの						
LEAVE	乗客・L10WAT						
	乗客・L10WAT						
N.L.B.A	乗客・前室センサ ーOFF(乗客)						
	乗客・前室センサ ーON						
N.L.B.B	乗客・前室センサ ーOFF(乗客)						
	乗客・前室センサ ーON						
A.L.B.A	乗客・L10P						
	乗客・スキャップ						
A.L.B.B	乗客・L10P						
	乗客・スキャップ						
A.L.P	乗客・L10D						
	乗客・L10D						
A.L.O	乗客・L10D						
	乗客・スキャップ						

## ■例) 閲覧者の状態遷移表

閲覧者の状態を、存在する部屋×ボックスアクセス(閲覧室内タッチパネル・アクセス)の有無で細分化

- 存在する部屋だけで状態を分けると制御構造が複雑になり、閲覧者が選択可能な行動を見落とす可能性がある
- 一旦タッチパネル脇ID端末で認証OKとなると、状態が戻ることはない



ラベル	状態		ラベル	状態	
	ボックス	部屋		ボックス	部屋
N_O	×	OUT_OF_ROOM	A_O	○	OUT_OF_ROOM
N_P	×	PREVIOUS_ROOM	A_P	○	PREVIOUS_ROOM
N_BA	×	BOOTH_ROOM_A	A_BA	○	BOOTH_ROOM_A
N_BB	×	BOOTH_ROOM_B	A_BB	○	BOOTH_ROOM_B

## ■ ケーススタディにおける要求仕様と検証性質

分類	要求仕様	No.	検証性質	備考	検証対象
ライブネス	IDカードを登録している人は、前室に入ることができる			簡略版モデルでは、全ての閲覧者は登録済み(前提1-5)	×
	前室に入った人は、必ず案内される	L01	<u>前室入室した人は、いつか必ず案内される</u>	—	○
	閲覧室の扉認証は、閲覧室内が空室なら誰でも成功する			センサを閲覧者のモデルに組み込んでいるため	×
	案内された人は、一定時間、閲覧室の金庫ボックス取り出しの認証を自分のIDカードでパスできる。	L02	閲覧室に案内された人だけが、タッチパネル脇ID端末で、いつか必ず認証OKとなる	簡略版モデルでは、時間を取り扱っていない(前提1-1)	○
		L03	閲覧室から退出した閲覧者は、いつか必ず待機者リストから削除される	システム構築ベンダ殿からの要請	○
安全性	閲覧室に人が案内されると、その人以外は、一定時間、閲覧室の金庫ボックスの認証に成功しない。 (最終的に保証すべき安全性)	S01	閲覧室に案内されていない人が、(タッチパネル脇)ID端末で認証を行っても、常に認証をパスしない	簡略版モデルでは、時間を取り扱っていない(前提1-1)	◎
	閲覧室が空室の時以外、閲覧室に人を案内しない。	S02	<u>閲覧室が空室のとき以外、決して案内は変更されない</u>	—	◎
	利用者は、どの部屋にも閉じ込められない。(常に、退出することが出来る。)			簡略版モデルでは閲覧室の扉を取り扱っていない(前提1-3)	×
		S03	<u>待機者リストはオーバーフローしない</u>	コードデバック中にindexエラーを見つけたため	◎

◎:検証済み ○:検証対象

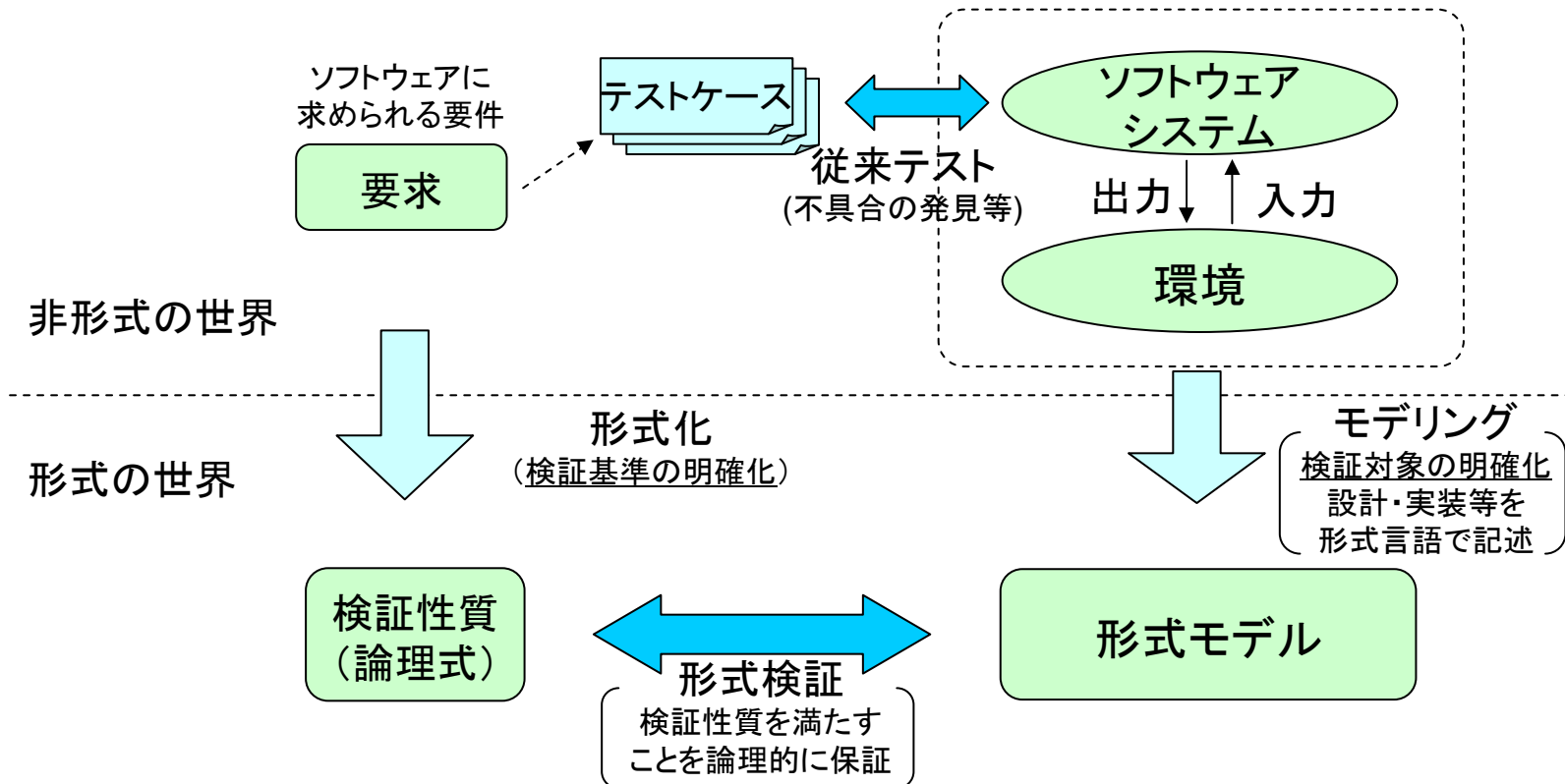
# 「フォーマルメソッド導入ガイダンス」(現状)の紹介

---

～ 形式手法の導入とモデリングのガイド ～

## ■ 形式手法とは？

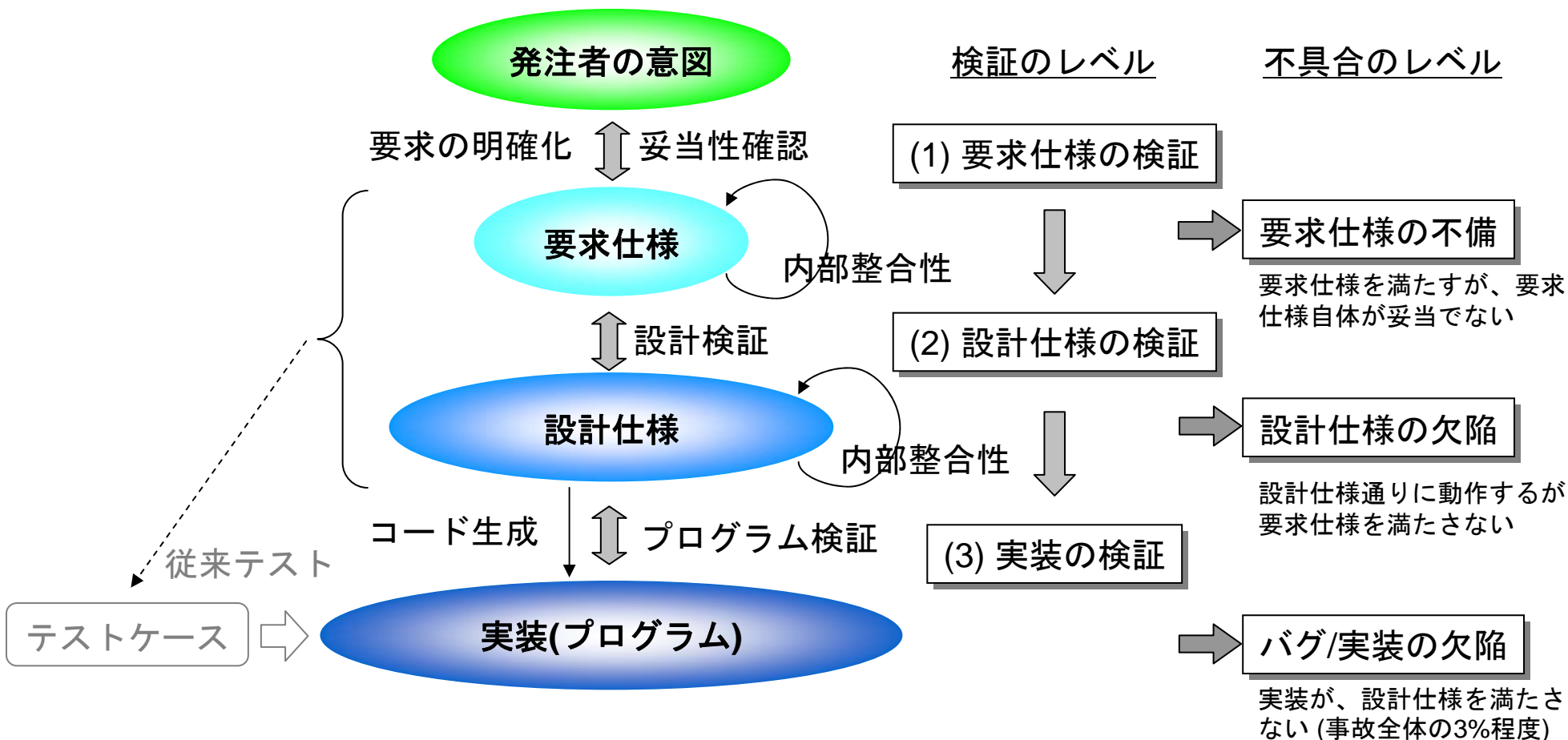
- ソフトウェアの要求や設計等の仕様を、数学理論に基づく厳密な言語で記述し、仕様が要求を満たすことを機械的に検証する技術。
- 世の中で100以上の異なる形式手法が存在し、それぞれベースとする数学理論に応じて異なる言語やツールが提供された技術の総称。
- テストケースに基づく従来のテスト法では、ソフトウェアの不具合の発見はできても、不具合が無いことを保証する点で無力。形式手法はその弱みを補う特徴を備えている。





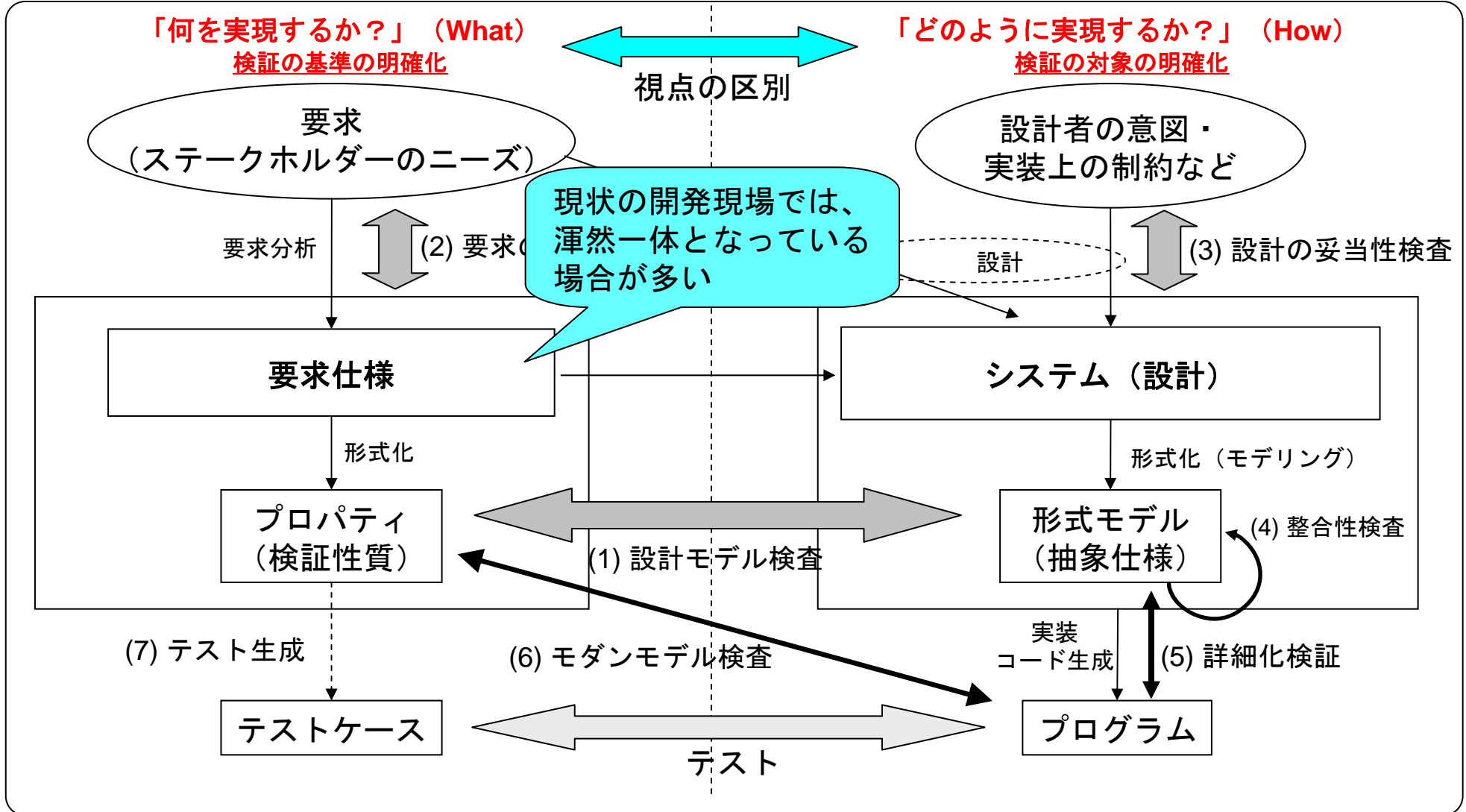
## ■ 形式検証のレベル分けと意義

- テストケースに基づく従来のテスト法では、バグを発見することはできても、バグが無いことを示す点では無力。
- ソフトウェアに関わる事故や損害の多くは、要求仕様や設計仕様の欠陥が原因である。

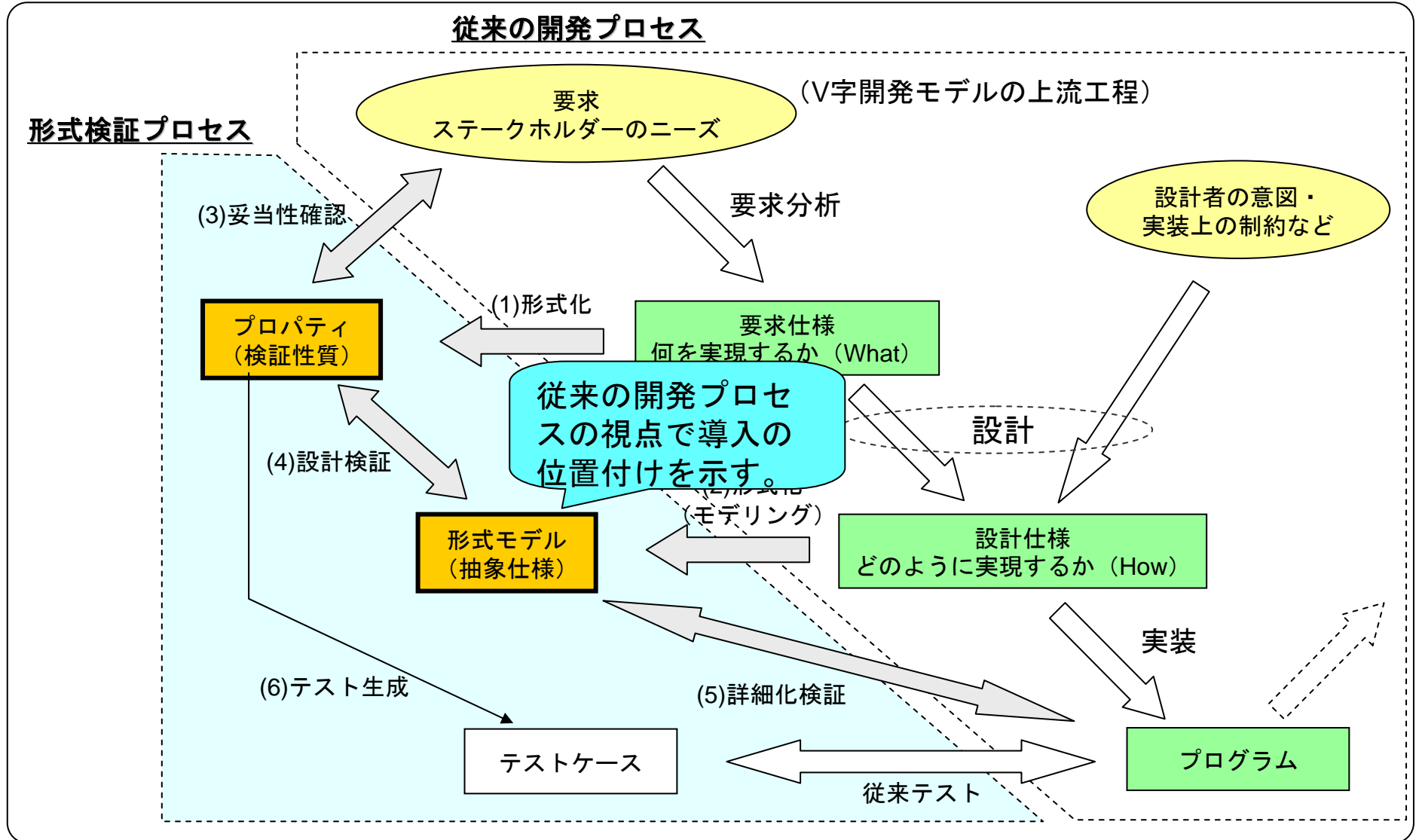


## ■ 形式検証において重要なことを強調

～「何を実現するか」と「どのように実現するか」の区別が重要～

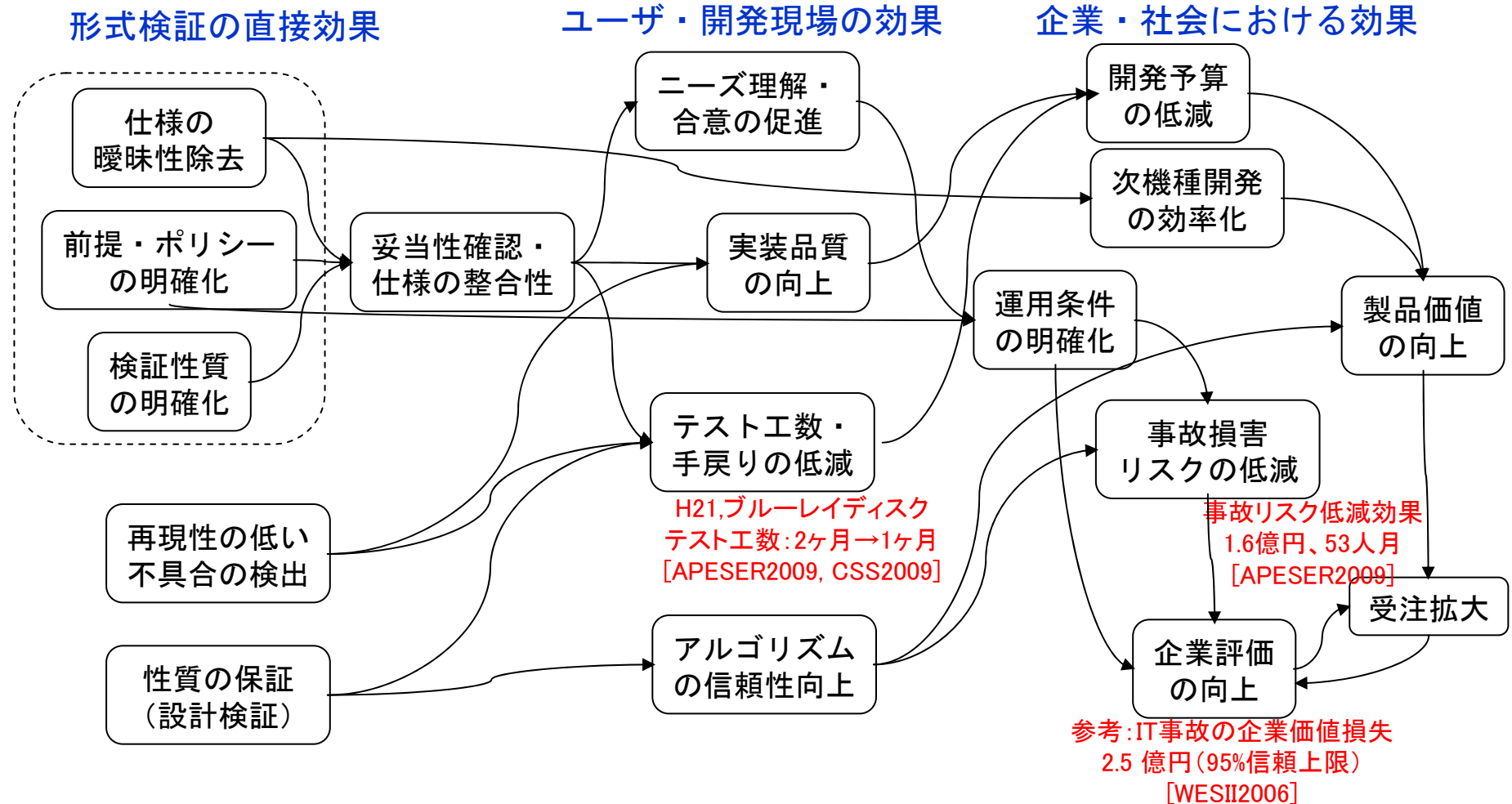


■ 従来の開発プロセス(V字モデル)に対する形式検証の位置付けを説明  
 ～開発者の視点から形式検証プロセスを捉え直す～



## ■ 形式手法の効果とそれらの間の関係(全体像) ～ 定性的な効果と定量化可能な効果の洗出し ～

※ 電中研の投資効果評価法を形式手法に応用



[1] Masaki Ishiguro, Hideyuki Tanaka, Kanta Matsuura, The Effect of Information Security Incidents on Corporate Values in the Japanese Stock Market, Workshop on the Economics of Securing the Information Infrastructure (WESII 2006)

[2] Masaki Ishiguro, Kazuyuki Tanaka, Shin Nakajima, Akihiro Umemura, Tomoji Kishi, A Guidance and Methodology for Employing Model-Checking in Software Development, APESER: Asia-Pacific Embedded Systems Education and Research Conference, December 14-15, 2009

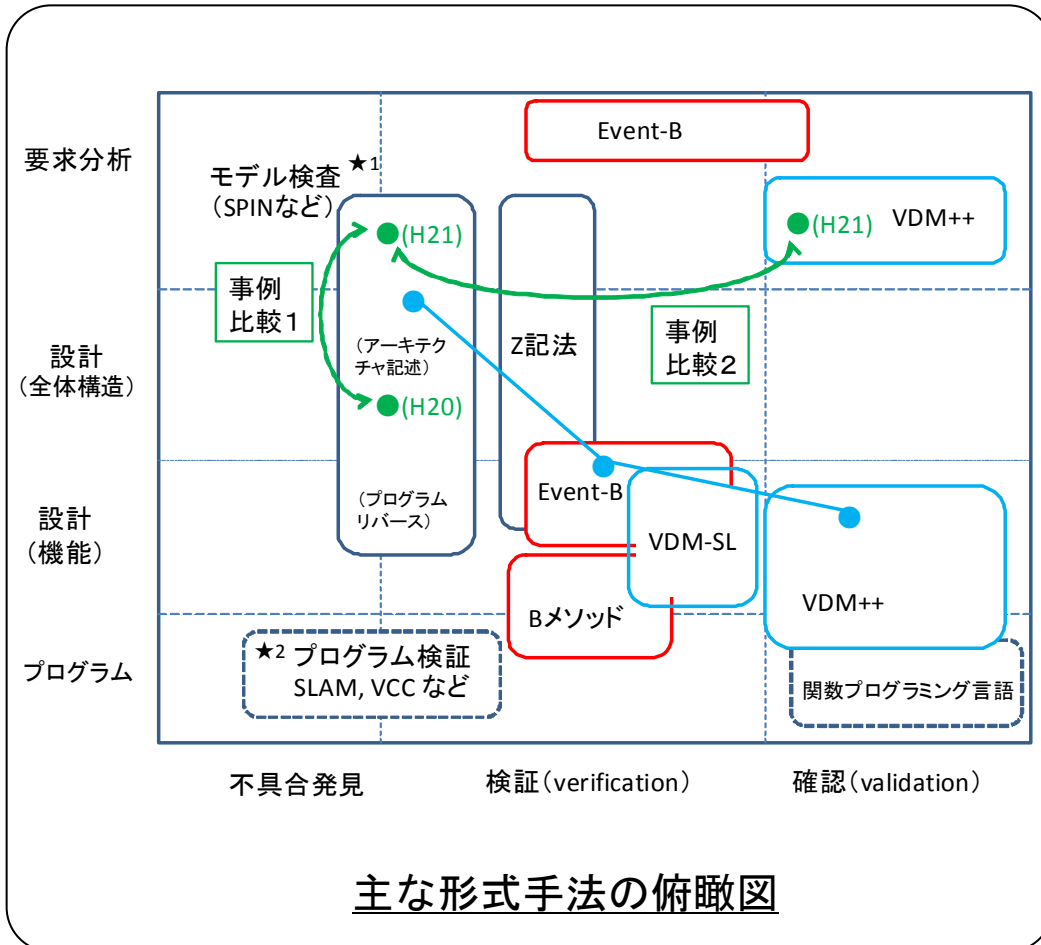
## ■ 開発現場における問題点と形式手法の適用による効果

～ 管理者、開発者が抱えるリアルな悩みと形式手法の効果 ～

効果の分類 不具合対策	開発現場における現状の問題点	形式手法により期待される効果	具体例
不具合の発見	<ul style="list-style-type: none"> <li>・並行プロセスのタイミングに依存する不具合は再現性が低く、発見が困難。</li> <li>・検査したい性質について、系統的に場合分けを網羅するテストケースの生成が困難。</li> <li>・そもそも、タイミングに係わる不具合の可能性さえも気が付かない場合が多い。</li> <li>・バグを減らすために、並行プロセスや条件分岐を減らすなど消極的なプログラミングを強いられる。</li> <li>・テストツールの自主開発は、コストがかさみ、また、転用が難しい。</li> </ul>	<ul style="list-style-type: none"> <li>・網羅的に検査するために、検査で不具合が見つからなければ、正しさが保証される。</li> <li>・モデル検査では、系統的、網羅的に自動検査が可能であるため、効率的。</li> <li>・不具合が存在する場合、不具合に至る実行トレースが提示されるため、原因を究明できる。</li> </ul>	<ul style="list-style-type: none"> <li>・ブルーレイディスクにおいて、操作要求とデバイス割込みが並行して発生する場合に、デッドロックが発生することがある。</li> <li>・再生実行が、処理されず停止に戻る場合がある。(通常の操作では発生しないことを確認)</li> </ul>
不具合の原因特定	<ul style="list-style-type: none"> <li>・不具合の存在が分かっても、再現性が低い場合、毎回挙動が異なるため、原因の特定が困難。</li> <li>・めったに再現しない不具合は、コストの制約から修正できないまま出荷せざるを得ない場合もある。</li> <li>・サイクルマチック数が30を越えると、設計した本人でも修正が困難となる。</li> </ul>	<ul style="list-style-type: none"> <li>・再現性が低い不具合であっても、網羅的な検査を行うため、不具合の原因特定が可能。</li> <li>・処理の分岐が複雑であっても、系統的に自動検査可能である。</li> </ul>	<ul style="list-style-type: none"> <li>・ブルーレイディスクの動作におけるデッドロックに至るトレースを追跡することで、原因が特定できた。</li> </ul>
不具合の修正の影響分析	バグの修正の影響範囲は人手によるコードレビューに頼っている。	一度、検証性質を記述すれば、モデルの修正後、繰返し検査に利用することができる。	
<b>検証</b>			
与えられた性質を満たすことを保証	<ul style="list-style-type: none"> <li>・並行動作のタイミングに依存する不具合は、テスト回数をいくら増やしても、完全な保証が得られない。</li> </ul>	<ul style="list-style-type: none"> <li>・網羅的な検査を行うため、不具合が検出されなければ、与えられた性質に関して正しさが保証される。</li> </ul>	<ul style="list-style-type: none"> <li>・ブルーレイディスクの操作制御モデルに、デッドロックが発生しないことを保証できる。</li> </ul>
検証結果の評価に対する科学的根拠	<ul style="list-style-type: none"> <li>・テストツールを開発しても、網羅性の保証は困難で、テスト結果に対する評価は、ベテランの感に頼ることが多く、科学的な根拠が示せない。</li> <li>・テストカバレッジを計測する方法が普及しておらず、パスを通った結果が正しいか人手に依存する。</li> </ul>	<ul style="list-style-type: none"> <li>・検査結果に網羅性が保証される。</li> <li>・形式仕様から、テストカバレッジを自動的に計測することができる(VDM)。</li> </ul>	<ul style="list-style-type: none"> <li>・ブルーレイディスクの制御モデルに関して、デッドロックが発生しないという性質を科学的に保証できる。</li> <li>・ファイルシステムにおいて、ファイルオープン操作のテストカバレッジが計測できる。(VDM)</li> </ul>
従来テストでは扱えない性質(安全性、到達性等)の検査	<ul style="list-style-type: none"> <li>・従来テストでは、状態の網羅的な検査が必要な安全性や、実行の無限系列に関する到達性に関する性質が扱えない。</li> </ul>	従来テストでは、扱いにくい性質を時相論理式として表現し、検証が可能。	「どのようなリクエストに対しても、いずれは待機状態に戻る」という性質は、従来のテストでは扱いが困難。
<b>妥当性確認</b>			
仕様の曖昧性排除	<ul style="list-style-type: none"> <li>・仕様書の曖昧性により、設計者と開発者の異なる思い込みにより、バグが入り込む。</li> <li>・要求仕様の曖昧性により、発注者の意図と設計者の理解にズレが生じることがある。</li> </ul>	<ul style="list-style-type: none"> <li>・性質に関する記述を、条件の論理的な組合わせ(かつ、または、否定などの構造化)で表現することにより、条件の範囲や掛かり方が明確化される。</li> <li>・条件が満たされる時間の範囲や、検証対象とする実行経路の範囲や、係り受けが明確化され、誤解が回避される。</li> </ul>	<ul style="list-style-type: none"> <li>入退室管理システムで「閲覧室が空室のとき以外、常に案内は変更されない」という表現は、「閲覧室が空室の時以外、決して案内は変更されない」の方が意味が明確。</li> <li>「常に性質pを満たさない」という表現は、『「いずれは、性質pを満たす。」の否定』、または、『「常に性質pを満たす」の否定』のいずれであるか明確化が必要。</li> </ul>

## ■ 形式手法選択のためのガイド

- 実応用の多い形式手法を中心に、手法の位置付けを俯瞰
- 形式手法の詳しい学習の前に、手法の概要を理解するための解説文献のガイドを整理



### 1.1.1.1. モデル検査

[1] モデル検査法のソフトウェアデザイン検証への応用

文献情報	モデル検査法のソフトウェアデザイン検証への応用, 中島 震, コンピュータ ソフトウェア, Vol. 23 (2006), No. 2, 2_72-2_86
入手先	<a href="http://www.istage.jst.go.jp/article/jssst/23/2/2_72/pdf-char/ja/">http://www.istage.jst.go.jp/article/jssst/23/2/2_72/pdf-char/ja/</a>
対象手法	SPIN を中心とする。SMV, NuSMV など他のモデル検査手法については応用動向についてまとめている。
想定読者	モデル検査など形式手法の前提知識の無い技術者

#### 概要:

モデル検査法およびそのツールなどに関して、技術の適用方法を伝えることに主眼をおいて分かりやすく解説している。また、ソフトウェア開発のいろいろな局面での利用例を紹介することで実用性について示している。簡単な例を用いてソフトウェアなどを状態遷移システムによるモデリングする方法と検証の基本原則について説明している。また、各種モデル検査ツールによる設計検証の応用例を概観している。幅広くモデル検査ツールについて比較一覧をまとめている。また、モデル検査の書籍に関する概要紹介を行っている。

本文献により、始めて学習する人にとってもわかりやすくモデル検査の基本概念が理解でき、また、具体的な目的に応じてツールの選択を行う際の参考情報が得られる。

## 解説文献ガイド抜粋

## ■今年度の研究発表等(一部)

- [1] Masaki Ishiguro, Kazuyuki Tanaka, Shin Nakajima, Akihiro Umemura, Tomoji Kishi, A Guidance and Methodology for Employing Model-Checking in Software Development, APESER: Asia-Pacific Embedded Systems Education and Research Conference, December 14-15, 2009
- [2] 石黒 正揮, 中島 震, 田中 一之, 梅村 晃広, 組込みソフトウェアの制御機構に対するモデル検査の適用に関する評価実験, 情報処理学会 コンピュータセキュリティシンポジウム CSS2009
- [3] 石黒正揮, フォーマルメソッド実用化に関する背景, ソフトウェアの信頼性およびセキュリティを確保するためのフォーマルメソッド適用のパターン化, 情報処理学会 ソフトウェア工学シンポジウム 形式手法ワークショップ 2009/9/7
- [4] 石黒正揮, ソフトウェア開発における形式手法導入に関する課題と解決アプローチ, 先端ソフトウェア工学に関するGRACE国際シンポジウム, 形式手法の産業応用ワークショップ2010, WIAFM2010: Workshop on Industrial Applications of Formal Methods, p.41-48, 2010年3月15日, 国立情報学研究所, Grace-TR-2010-03
- [5] 石黒正揮, ESEC第12回組込みシステム開発技術展 ソフトウェアエンジニアリングトラック 専門セミナー「形式手法の概観とモデル検査法の応用」
- [6] 中島震, 「形式手法」の「適用」について, ソフトウェアシンポジウム2009「形式手法適用」WG, 2009年6月
- [7] 中島震, ソフトウェア工学からみたモデル検査法, 第22回 回路とシステム軽井沢ワークショップ, 2009年4月
- [8] 中島震, モデル検査とパターン, SES2009併設ワークショップ, 2009年9月
- [9] 中島震, 形式手法とモデリング, 情報処理学会ソフトウェア工学研究会ウインターワークショップ, 2010年1月
- [10] 中島震, 谷津弘一, 野中 哲, 佐原伸, 検証モデリングの比較検討 ～ 組込みソフトウェアの事例 ～, 電子情報通信学会コンカレント工学研究会, 2010年1月
- [11] 梅村晃広, 中島震, 「SATソルバ・SMTソルバの技術と応用」, コンピュータソフトウェア, 2010 May(予定)

## ■ 今後の取り組み

- 「フォーマルメソッド導入ガイドンス」の作成  
ソフトウェア開発現場の実態や情報に基づき、形式手法をソフトウェア開発工程に導入するための勘所や留意点をまとめる。
  
- 「フォーマルメソッド導入ガイドンス」の普及促進  
組込み系、エンタープライズ系、重要インフラ系など幅広いソフトウェア産業およびユーザの業界団体と連携し、「フォーマルメソッド導入ガイドンス」を生かしつつ、形式手法の実応用の支援、セミナーの開催等を行う。
  
- フォーマルメソッド導入支援
  - 形式手法の適用を規定した国際標準「コモン・クライテリア(ISO/IEC 15408)」、「機能安全(IEC61508)」、「自動車分野の機能安全(ISO26262)」等への認定取得支援。
  - 「コスト見積りモデル契約書」における形式手法の導入支援(受託企業向け支援)、外注先企業等への形式手法の導入支援(発注企業向け支援)の実施。



～ご静聴ありがとうございました。～

お問合せ先：

**MRI** 株式会社 三菱総合研究所

情報セキュリティ研究グループ

主任研究員 石黒 正揮, Ph.D.

TEL : 03-3277-5609 (直通)

FAX : 03-3277-0567

E-mail : masa@mri.co.jp

お気軽にお問合せ下さい！