発表資料

GSNを用いた欠陥情報からのシナリオテスト作成の取り組み

国立研究開発法人　宇宙航空研究開発機構（JAXA）

研究開発部門　第三研究ユニット

◎波平晃佑（発表者）　梅田浩貴　植田泰士　片平真史

namihira.kohsuke@jaxa.jp

JaSST'20 Hokkaido　2020年7月3日

目次

- 背景説明（テスト対象の特性、課題）
- 提案手法の概要
- 適用事例
- まとめ

本プレゼンのターゲット

- システムテスト設計者として
  異常時のシナリオテストの作成に課題をお持ちの方

- 過去の欠陥情報（不具合情報）の活用に
  課題をお持ちの方

- Goal Structuring Notation（GSN）を使った分析を
  業務への適用を検討している方

発表の概要

課題

外部環境に強く影響を受け、複数のコンポーネントが協調しながら動作を行う、またV字モデル開発を行う宇宙機ソフトウェア開発で、シナリオテスト設計を実施した場合に以下の課題があった。

- 複数の発生条件をもつシナリオテストを検討することが困難
  - 主な理由：外部環境やコンポーネントの状態の組み合わせは無数にある
- 過去の欠陥情報を参考にシナリオテストを作成しても、類似シナリオの識別漏れが発生する
  - 主な理由：過去の欠陥情報（不具合情報）は、条件の1つの組み合わせの例にすぎない

効果

- 提案手法は、膨大な条件組み合わせから、欠陥情報を活用し、外部環境やSW処理等の発生可能性のある複数の条件の組み合わせの可視化を誘導する
- 提案手法は、ガイドワード適用の促進によって網羅性を向上できる

宇宙機システムの特徴

- 外部環境の状態の影響を強く受ける
  - 「太陽や星の位置」等を観測しながら制御する
  - システム/SWからみると、衛星自体の状態（「衛星の姿勢」等）も外部環境の1つとなる。

- 複数のコンポーネントと協調しながら動作を行う
  - 多数センサのデータ処理機能、姿勢計算処理、冗長切り替え の連携等

- 故障/異常があっても出来るだけ動作継続が求められる
  - 代替シナリオが多い

    打ち上げ時に速度不足が発生したとしても分離タイミングを変化することでミッションを達成する等

- ロケット

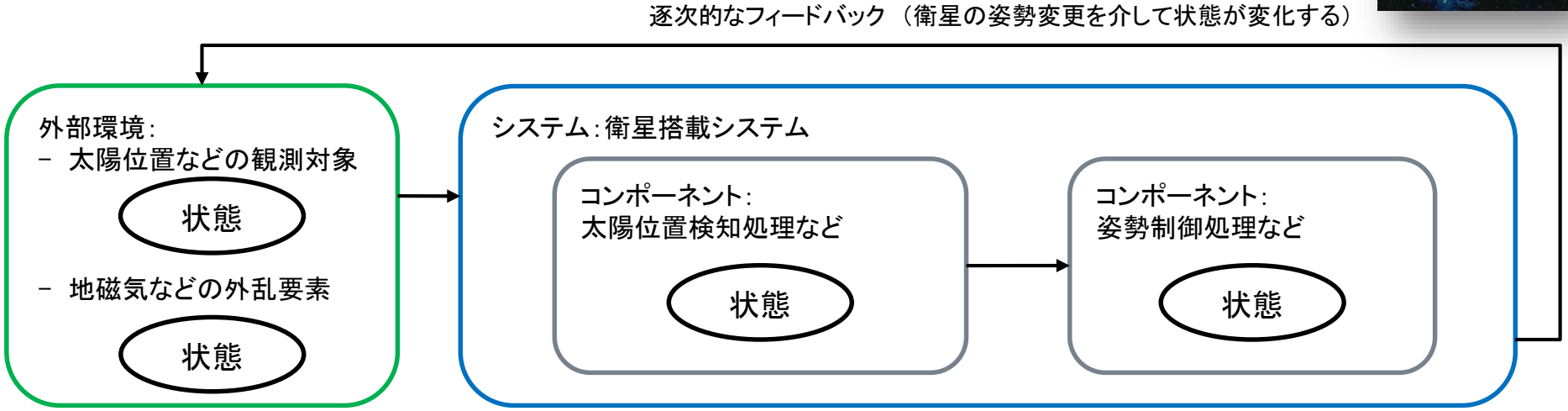
- 人工衛星

- 宇宙ステーション

- 地上管制システム

リスクシナリオテスト設計への要件(1/2)

宇宙機システムの動作の流れ



＜外部環境＞
時間経過に伴う状態変化
例：慣性運動など

＜システム内の振る舞い＞
時間経過を考慮した動作
例：過去の増減量をもとに姿勢制御量を決定する計算　など

リスクシナリオの発生原因を識別するには
時間経過を考慮する必要がある

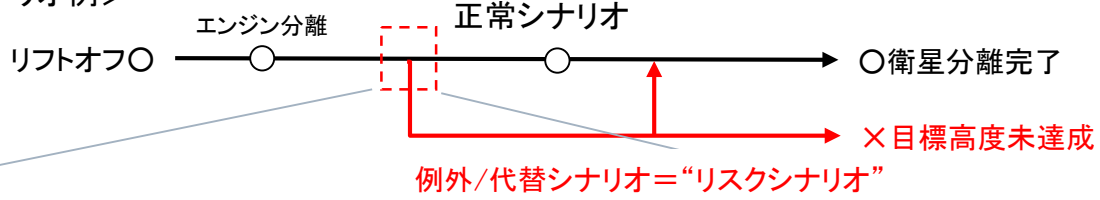
例：状態変化の過渡状態、制御実施タイミングのズレ、誤った過去データの蓄積など

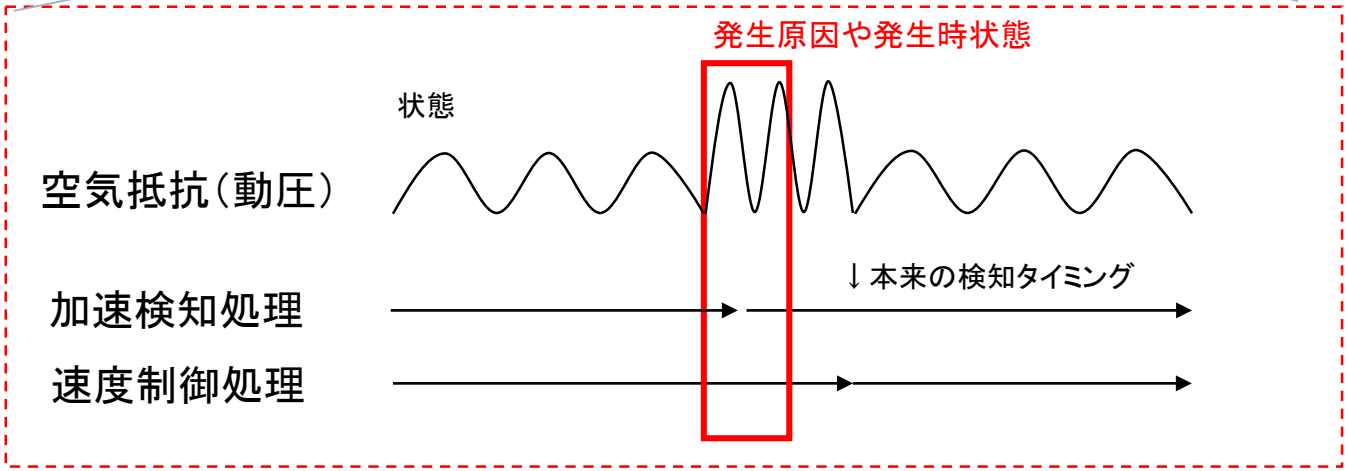
リスクシナリオテスト設計への要件(2/2)

リスクシナリオのテストを作成するためには
- 正常シナリオ上のどのタイミングで発生するか
- 外部環境やコンポーネントがどんな状態か
- 他のコンポーネントからどんな入出力があったか

等の複数の条件を設定する必要がある

＜ロケット打ち上げのシナリオ例＞





発生条件

正常シナリオ上のタイミング

＋

外部環境の状態

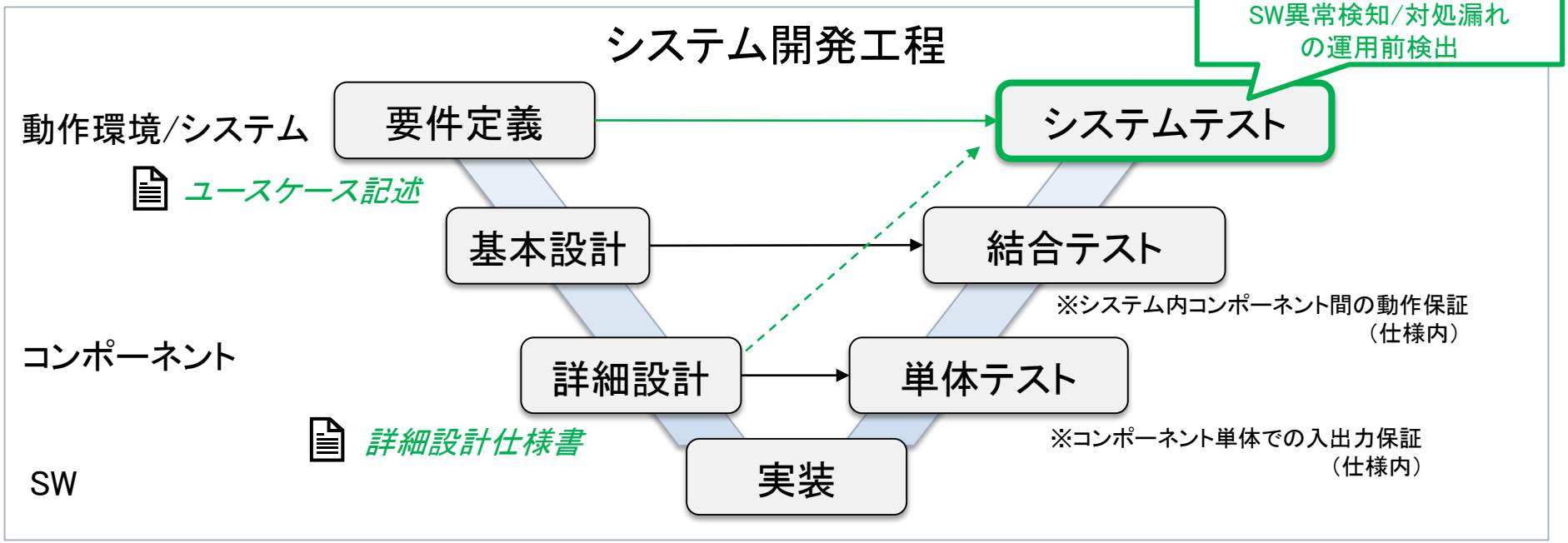
＋

各コンポーネントの状態や入出力値

リスクシナリオの発生条件の例:
“エンジン分離(重量減少)後に、空気抵抗の変動が大きい場合に加速検知処理が誤検知する”

少なくとも3つ以上の発生条件

V字開発モデルに依る課題



【シナリオテストの目的】
ユースケース上の**異常/故障時の振る舞い**に関する検証（耐故障性能の向上）
→例外/代替シナリオに関するテストの作成（“リスクシナリオテスト”）

【リスクシナリオテスト設計への要件】
リスクシナリオテストを効果的に実施するためには
上位仕様だけでなく、SW詳細設計に含まれる異常検知や対処に関する条件も考慮したほうが良い

【課題】
テスト設計者が複数の開発工程の情報を個別に参照して関係有無の分析をしなければならないため、
リスクシナリオとなる発生条件の識別の難易度が高い。組み合わせが膨大となる。

テスト設計方法に依る課題

従来使われる「表」や「単一観点」を使った発生条件の識別

テスト設計のフォーマット①

大項目	中項目	小項目
○○機能	起動	
	終了	

■説明
テスト項目を大/中/小などの項目で項目を分割している。
（仕様書の項番がそのまま大/中/小の項目に使われる場合が多い）

■課題
- 2つの要素が関係する条件、時間経過に関する条件の表現が困難
  - 協調動作/並列動作、復旧処理 など
  - タイミングのズレ など

テスト設計のフォーマット②

テスト対象	テスト観点 もしくは 品質特性	テスト観点 もしくは 品質特性
○○機能	○	
××機能		○

■説明
テスト対象とテスト観点/品質特性を組み合わせている。
例：○○機能と「入力」　[テスト対象×テスト観点]
　　××機能と「正確性」　[テスト対象×品質特性]

■課題
3つ以上の組合せは、2項関係で表現が難しい。
- {環境、システム、処理}の状態の組み合わせ

組み合わせによっては動作が変化する場合の考慮が困難
- 入力センサーの故障状態によって振る舞いが異なる など

シナリオテスト設計からみた欠陥情報(不具合情報)の特徴

3つ以上の条件の組合せを考慮しようとすると、作成されるリスクシナリオの数は膨大となる

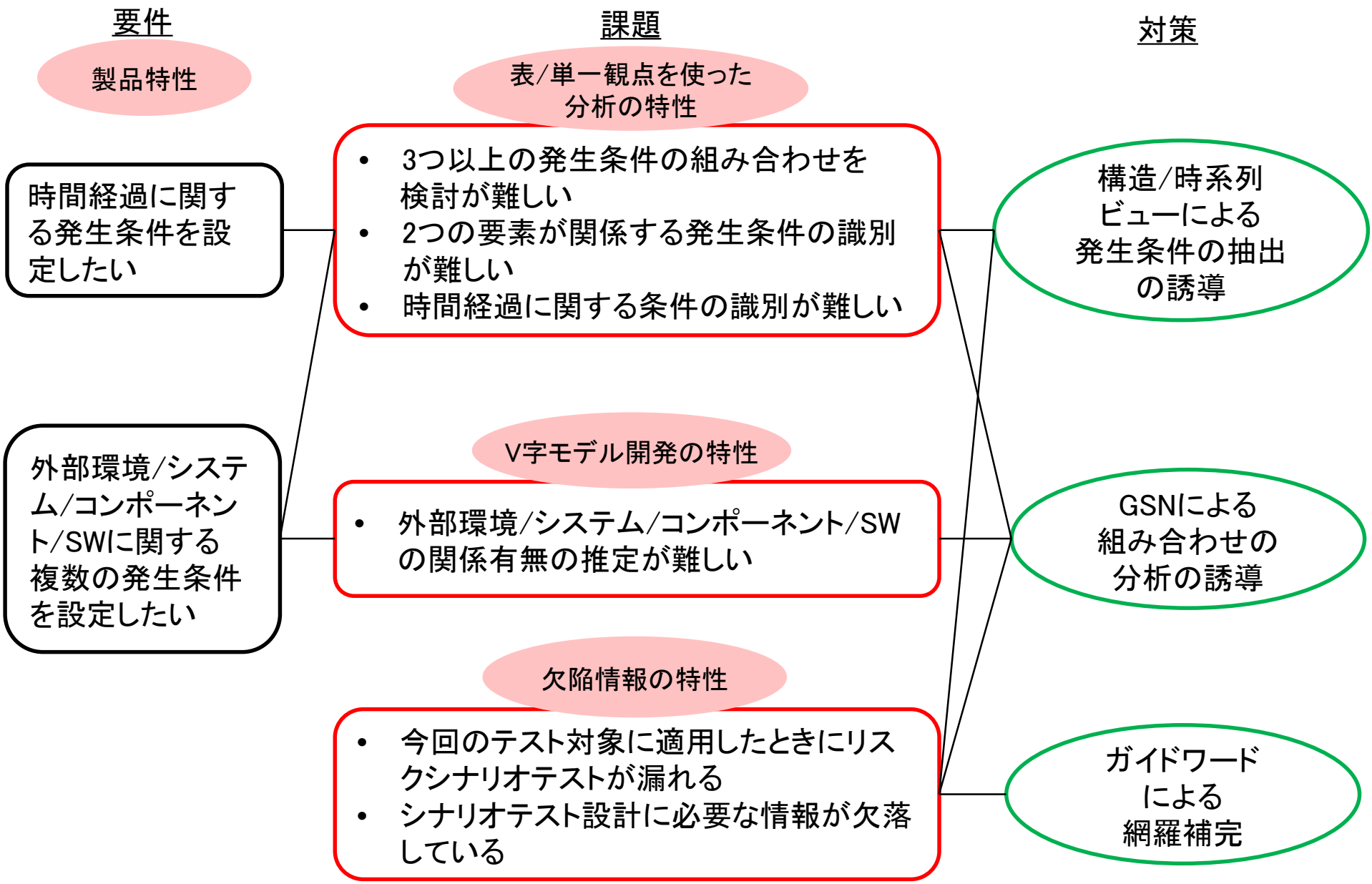
提案手法では
過去の実際に発生した欠陥(不具合)を利用し
テストすべき発生条件の組み合わせを識別する

■欠陥情報を利用するときの課題

- 1つの欠陥情報は、その発生時の条件の1つの組み合わせのみを表現している。多数の欠陥情報を利用した場合でも、テスト条件の識別漏れが発生する可能性がある。
  -> 今回のテスト対象に合わせ、発生可能性がある&テスト可能な条件範囲を向上させる必要がある(網羅性向上)

- 記載内容自体に、動作環境やコンポーネント間の入出力値の条件に関する情報が欠落している場合、その解釈や活用判断は属人的/暗黙知となる。
  -> 欠陥情報からシナリオテスト設計に必要な情報を補完/可視化する必要がある

シナリオテスト設計への要件、課題と対策

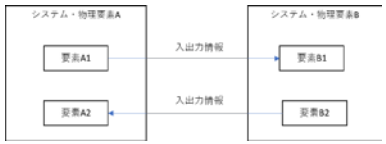


提案手法の概要

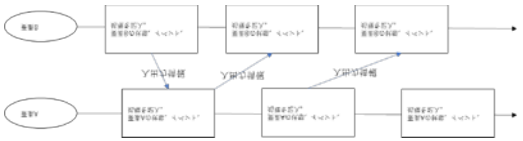
①欠陥情報から発生条件の抽出

【工夫点1】
入出力関係と時系列の視点で
欠陥の発生条件を抽出

構造ビュー

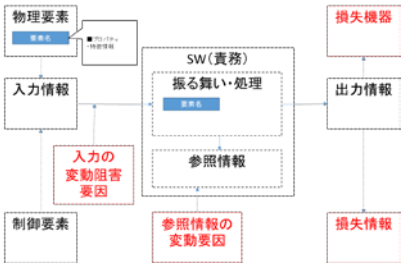


時系列ビュー



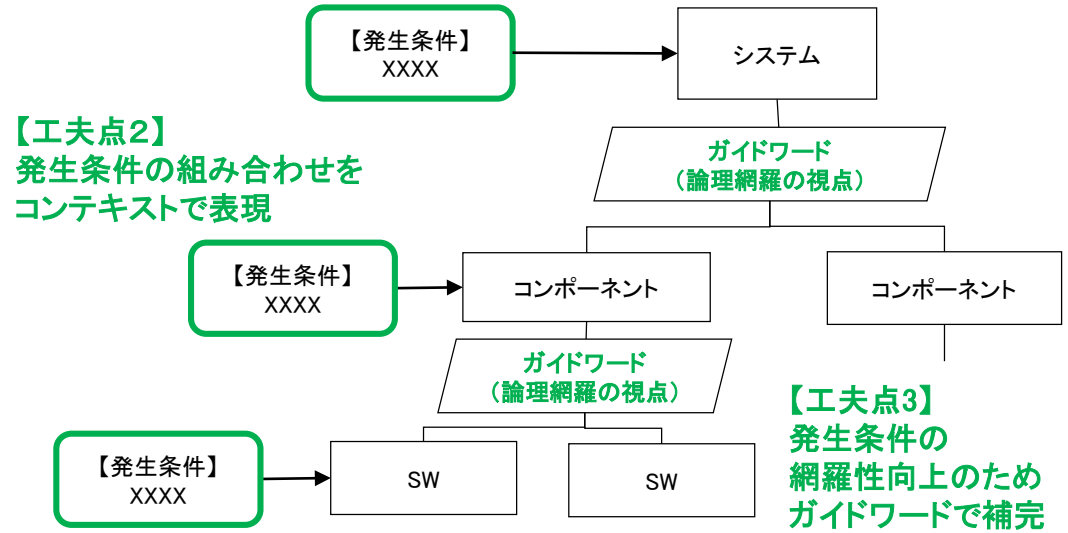
各位置づけで特徴情報を抽出

構造ビュー



②リスクシナリオ発生条件の導出

GSN

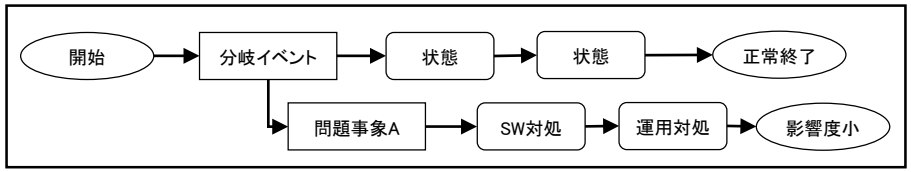


【工夫点2】
発生条件の組み合わせを
コンテキストで表現

【工夫点3】
発生条件の
網羅性向上のため
ガイドワードで補完

③リスクシナリオの作成

組み合わせた発生条件をシナリオとして表現する



④テスト条件の詳細化

【工夫点3】
テスト条件の網羅性向上のため
ガイドワードで補完

分類	属性	ガイドワード
入力	入力値	異常値、特殊値、なし(NULL)
	入力値域	最大、最小、範囲外、範囲内
	初期値	ゼロ、デフォルト値、未更新
アルゴリズム	時刻変化	変化なし、急激な変化
	積算	飛び値、未積算、一部積算

適用事例

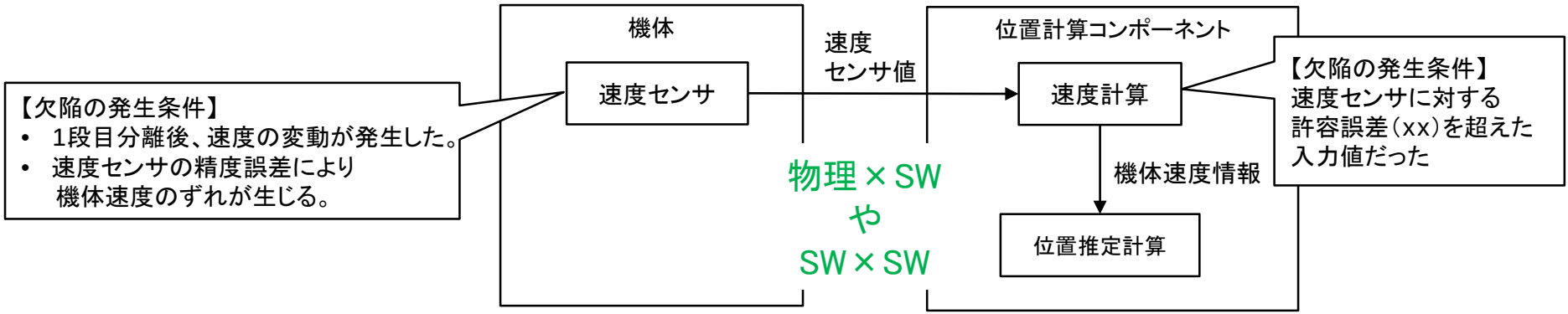
【従来の課題】
2要素間/時系列の発生条件が識別できない

工程①-1：欠陥情報から発生条件の抽出

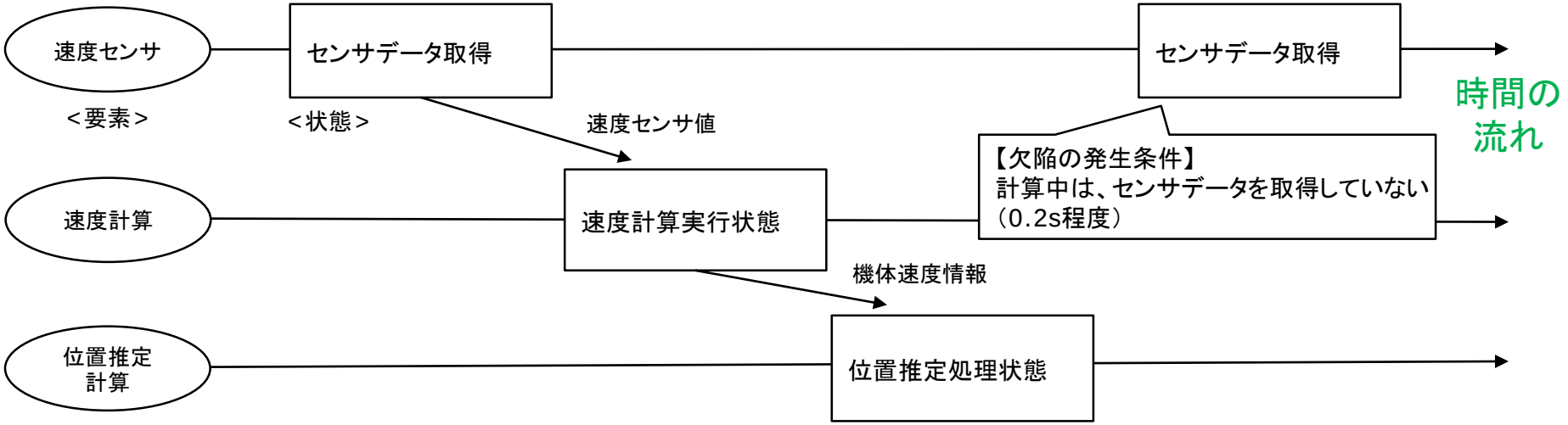
【効果】
・他要素間や時系列の発生条件を識別できる
・シナリオテストの条件として不要な条件は表現されない
  - SW単体の原因：設定値誤り など
  - プロセスの原因：実装ミス/レビュー漏れ など

【工夫点】
データ入出力関係と時系列の視点（ビュー）を使い、
欠陥の発生条件を識別する

■構造ビュー：複数の要素が関わる発生条件の識別



■時系列ビュー：時系列に関する発生条件の識別



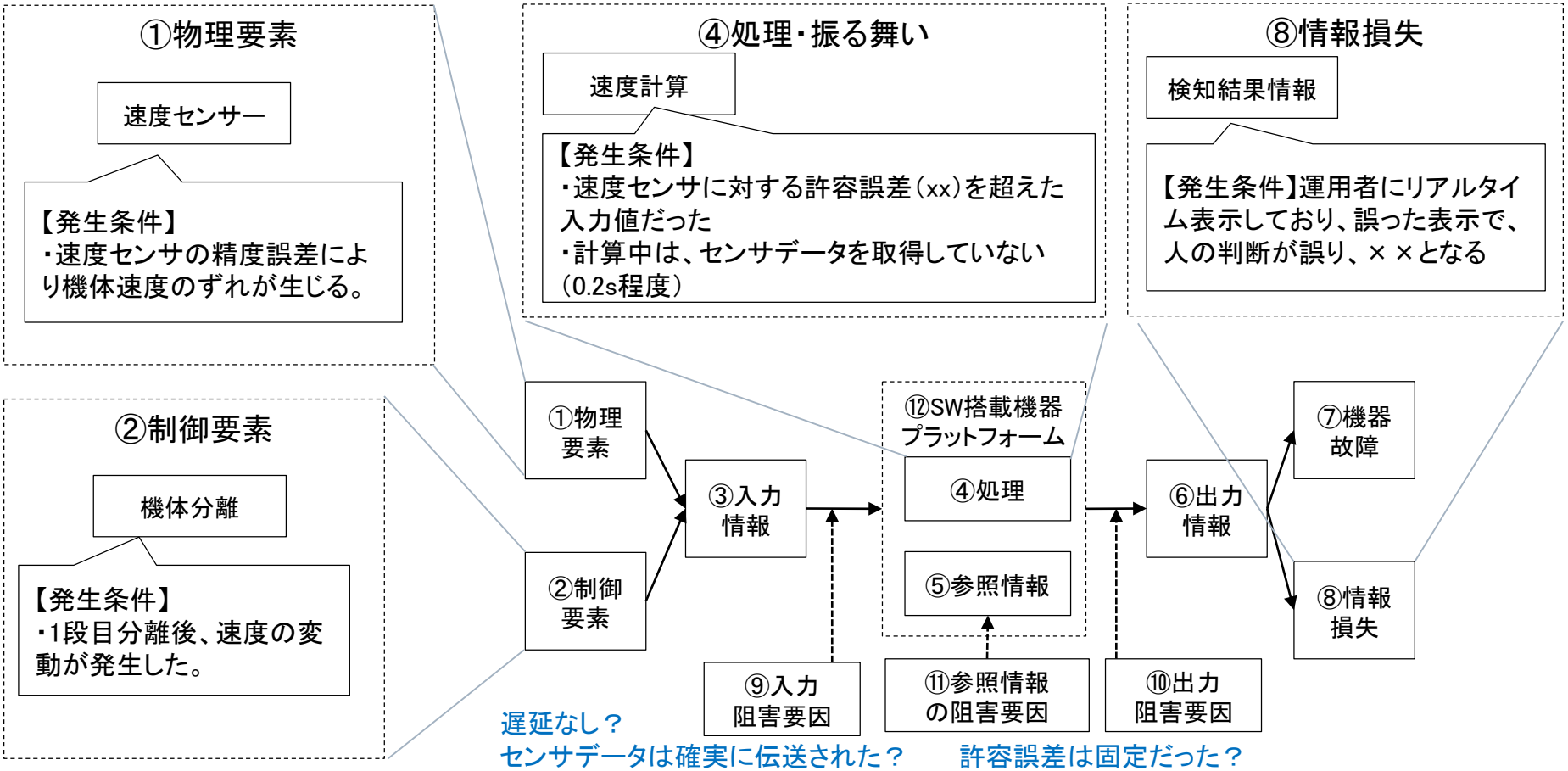
適用事例

工程①-2：欠陥情報から発生条件の抽出

【工夫点】
「構造ビュー」＋「要素の位置づけのガイドワード」で発生条件の漏れがないか確認する

【従来の課題】
シナリオテスト設計に必要な情報が欠落している

【効果】
外部環境や運用、SW処理などに関わる条件が抜けていないかを検証できる

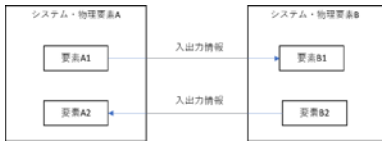


提案手法の概要

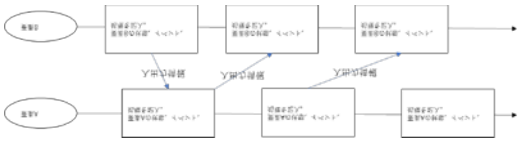
①欠陥情報から発生条件の抽出

【工夫点1】
入出力関係と時系列の視点で
欠陥の発生条件を抽出

構造ビュー

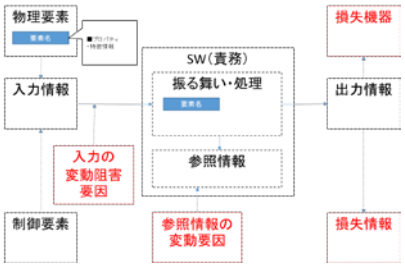


時系列ビュー



各位置づけで特徴情報を抽出

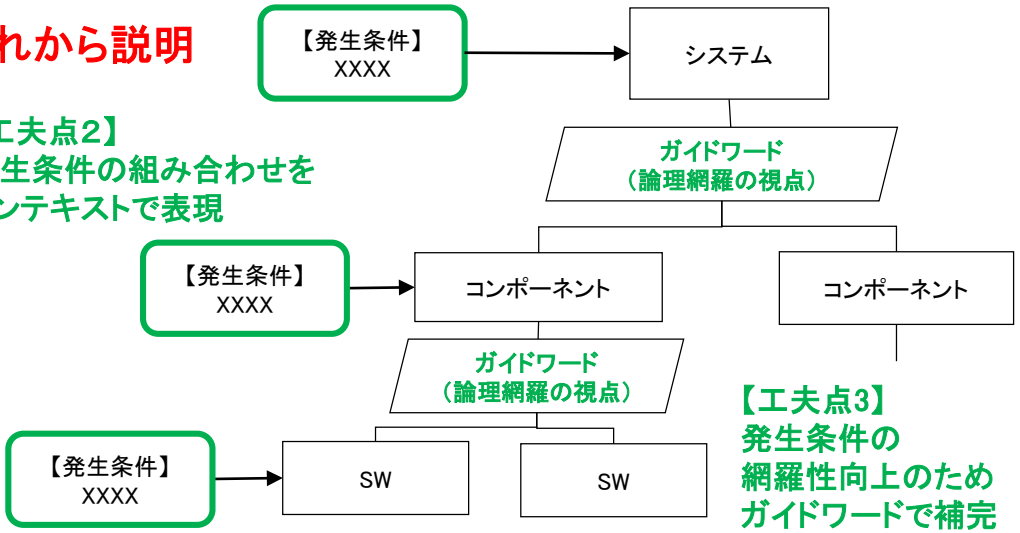
構造ビュー



②リスクシナリオ発生条件の導出

☑これから説明

GSN



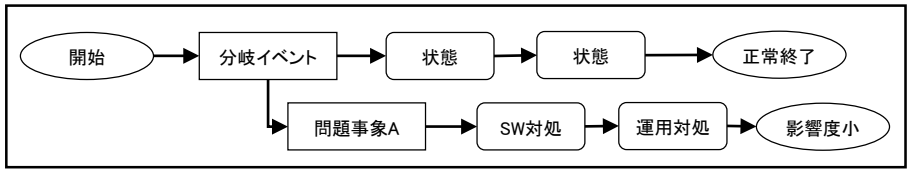
【工夫点2】
発生条件の組み合わせを
コンテキストで表現

【工夫点3】
発生条件の
網羅性向上のため
ガイドワードで補完

③リスクシナリオの作成

組み合わせた発生条件をシナリオとして表現する

☑これから説明



④テスト条件の詳細化

【工夫点3】
テスト条件の網羅性向上のため
ガイドワードで補完

分類	属性	ガイドワード
入力	入力値	異常値、特殊値、なし(NULL)
	入力値域	最大、最小、範囲外、範囲内
	初期値	ゼロ、デフォルト値、未更新
アルゴリズム	時刻変化	変化なし、急激な変化
	積算	飛び値、未積算、一部積算

【従来の課題】
・3つ以上の条件組み合わせが困難
・関係有無の推定が難しい

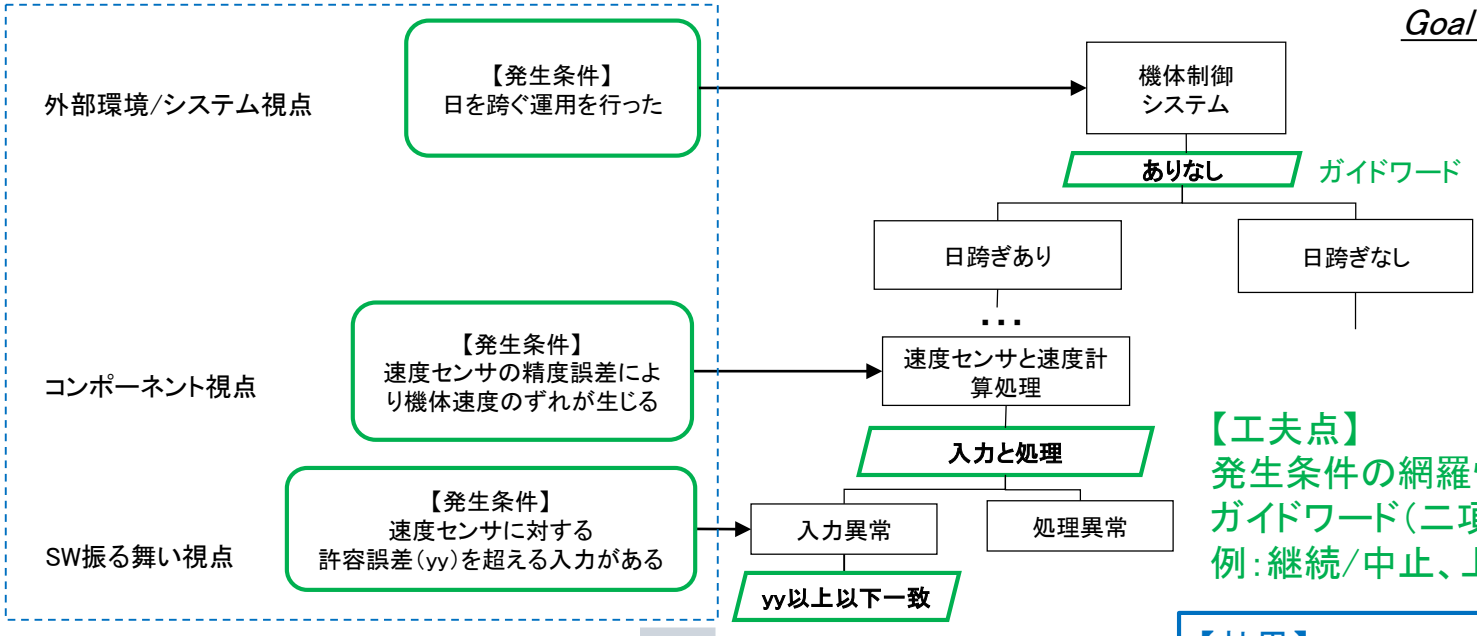
適用事例

【従来の課題】
今回のテスト対象に適用したときにリスクシナリオテストが漏れる

工程②:リスクシナリオの発生条件の導出

【工夫点】ツリービューを使い、多数の欠陥情報から個別に抽出した発生条件を集約し、
今回のテスト対象の仕様を考慮/変更し整理する

Goal Structuring Notation (GSN)

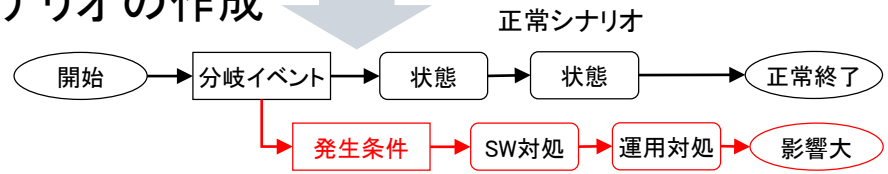


【工夫点】
発生条件の網羅性向上のため
ガイドワード(二項対立/MECE)で補完
例:継続/中止、上位/一致/下位

【効果】
・多数の組み合わせをコンテキストの組み合わせで表現できる
・上位下位の関係有無を検証できる

【効果】
過去に起きた発生条件の組み合わせを
起点として、新たな条件を発想し
網羅性を向上できる

工程③:リスクシナリオの作成

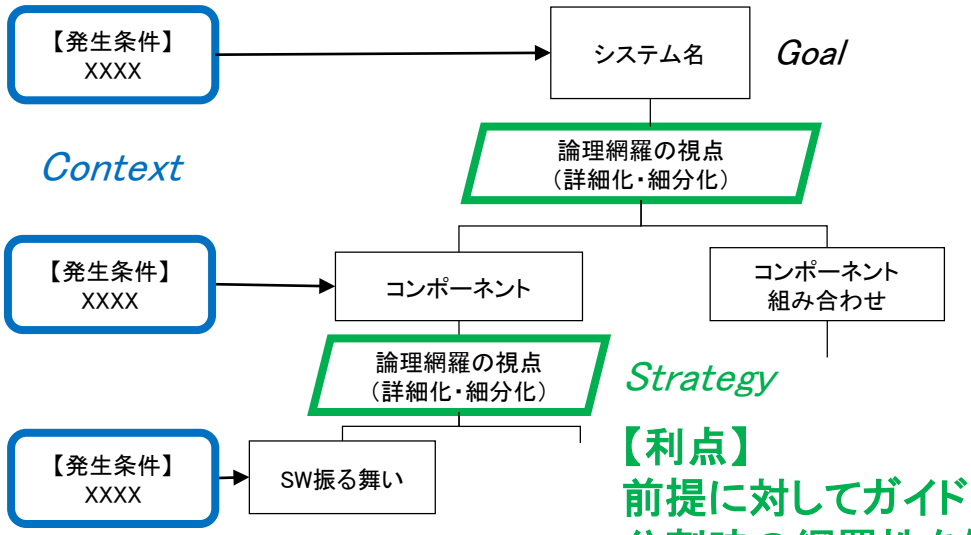


・条件組み合わせの成立を確認する
・テストの必要性有無を検討する

GSNの利点/採用理由

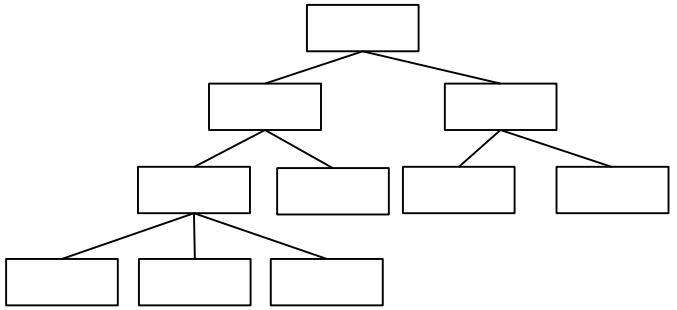
Goal Structuring Notation（GSN）

【利点】
- **テスト実施の目的や保証したいことの可視化（前提/意図の表現）**
- **条件絞り込みができる（発散の防止）**



【利点】
前提に対してガイドワードを設定できるため
分割時の網羅性を保証
（漏れ防止）

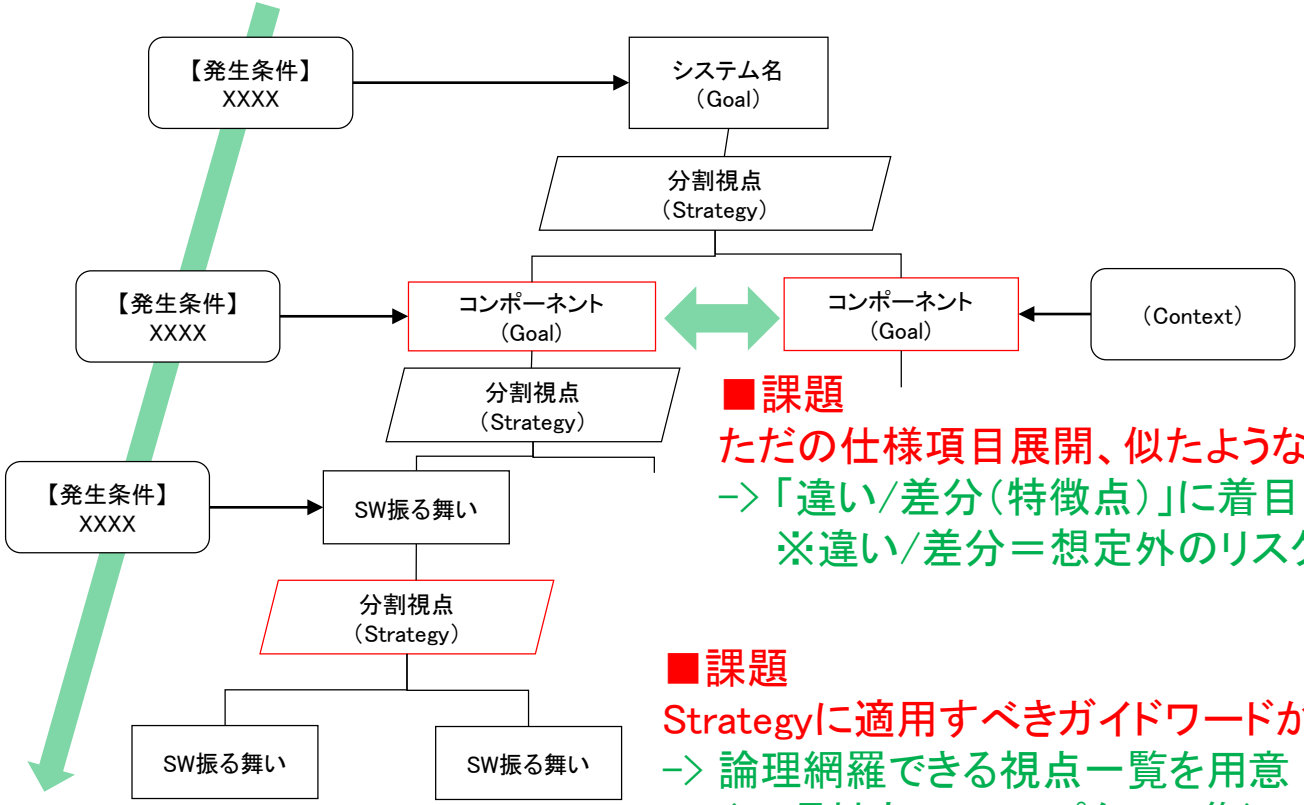
一般的なロジックツリー（例：マインドマップ、FTA）



【課題】
- **なぜこの分解をしたか読み取れない**
- **分解した視点が不明（MECEか確認不可）**

■その他の期待される効果：
テスト条件導出までの思考経緯の可視化により、**技術継承やレビューの促進**

参考：GSN作成のよくある課題と支援



■課題
ただの仕様項目展開、似たような展開が多い
–>「違い/差分（特徴点）」に着目し、Contextに明文化
※違い/差分＝想定外のリスクが潜む

■課題
Strategyに適用すべきガイドワードがわからない
–> 論理網羅できる視点一覧を用意
（二項対立、MECEパターン集）

■課題
展開が発散し、収集がつかない
–>上位下位関係ごとにContextのみを抽出し、Context同士が継承できているかを確認する
（同位包含関係が達成できているか）
※発散＝論理飛び/ずれ

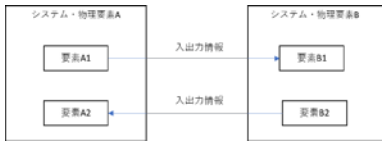
【GSN作成方針】 Context/Strategy中心のGSN作成
※特にContextは他のビューでの識別を誘導し、GSN自体の作成コストを削減する

提案手法の概要

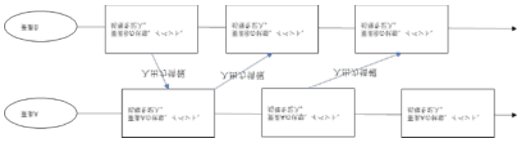
①欠陥情報から発生条件の抽出

【工夫点1】
入出力関係と時系列の視点で
欠陥の発生条件を抽出

構造ビュー

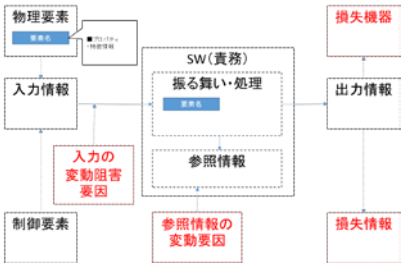


時系列ビュー



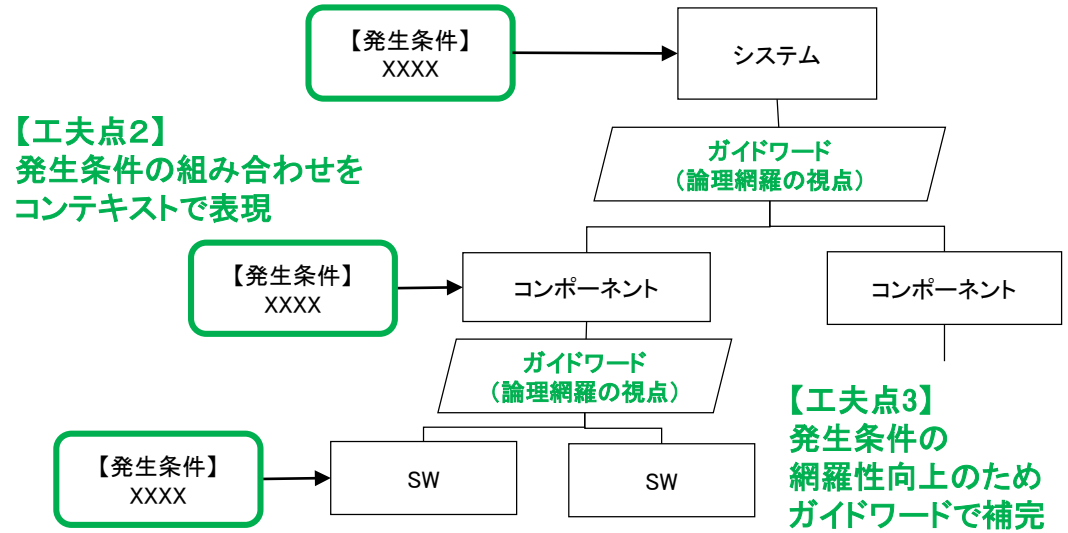
各位置づけで特徴情報を抽出

構造ビュー



②リスクシナリオ発生条件の導出

GSN

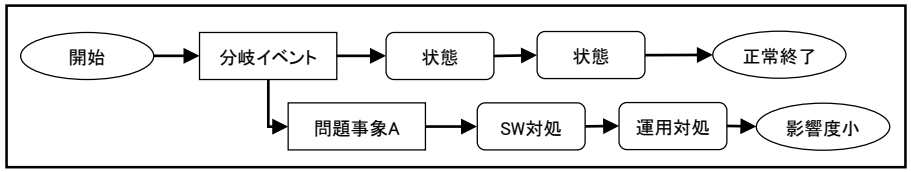


【工夫点2】
発生条件の組み合わせを
コンテキストで表現

【工夫点3】
発生条件の
網羅性向上のため
ガイドワードで補完

③リスクシナリオの作成

組み合わせた発生条件をシナリオとして表現する



④テスト条件の詳細化

☑これから説明

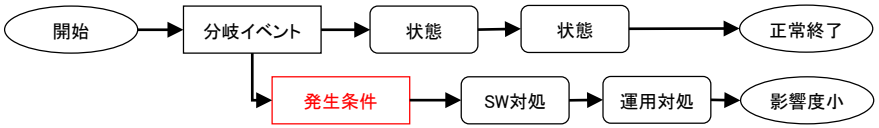
【工夫点3】
テスト条件の網羅性向上のため
ガイドワードで補完

分類	属性	ガイドワード
入力	入力値	異常値、特殊値、なし(NULL)
	入力値域	最大、最小、範囲外、範囲内
	初期値	ゼロ、デフォルト値、未更新
アルゴリズム	時刻変化	変化なし、急激な変化
	積算	飛び値、未積算、一部積算

適用事例

工程④:テスト条件の詳細化

【工夫点】
テスト条件の網羅性向上のため、異常パターンのガイドワードで補完



※ここまでに導出された発生条件(≒テスト観点)

分類	属性	ガイドワード
入力	入力値	異常値、特殊値、なし(NULL)
	入力値域	最大、最小、範囲外、範囲内
	初期値	ゼロ、デフォルト値、未更新
アルゴリズム	時刻変化	変化なし、急激な変化
	積算	飛び値、未積算、一部積算

【従来の課題】
・今回のテスト対象に適用したときにリスクシナリオテストが漏れる

【効果】
・テスト実施可能な条件設定になる
・テストとして実施可能な条件の網羅性を向上できる

■最終的に導出されるリスクシナリオテストの例

★シナリオテストの要件達成!★
・時系列視点の振る舞い
・動作環境～SWまでの条件の組み合わせ

“運用中に日付変更が起きた”場合に“内部時刻が0時に戻り”、“同一の時刻データ処理で処理を行う”

提案手法のまとめ

※特許出願済み（特願2018-121252）
・本手法を実現するツールを提供
・本手法に関するノウハウ情報を提供

問い合わせ先：
umeda.hiroki@jaxa.jp

シナリオテスト設計工程

テスト前提を分析する

↓

リスクシナリオの発生条件を識別する

↓

リスクシナリオを作成する

↓

リスクシナリオテストを作成する

提案手法の作業工程

テスト対象の分析

正常イベント	アイテム	影響の定義

不具合発生条件の分析

特徴情報	懸念情報	特徴抽出図の位置づけ

テスト対象の特性と不具合発生条件の組合せ

No.	特徴情報	懸念情報	アイテム名
1	SW製品は、〇〇という特徴がある。	システムが、××という懸念がある。	具体アイテム名

最上位の要素名	1層目視点	1層目の要素名	2層目視点	2層目の要素名	3層目視点
抽象名詞	視点1	名詞3	視点11	具体名詞1	
				具体名詞2	

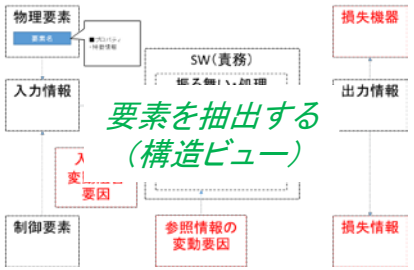
不具合発生前後の仕様を分析

システム状態	発生条件	SW対処	運用システム対処

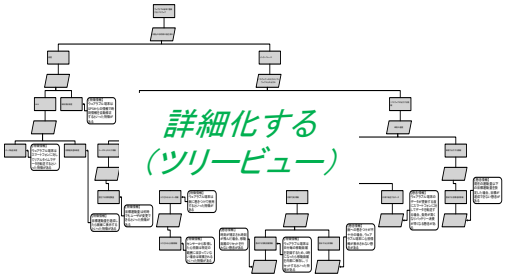
シナリオに対してテスト条件を細分化

分類	属性	ガイドワード
入力	入力値	異常値、特殊値、なし（NULL）
	入力値域	最大、最小、範囲外、範囲内
	初期値	ゼロ、デフォルト値、未更新
アルゴリズム	時刻変化	変化なし、急激な変化
	積算	飛び値、未積算、一部積算

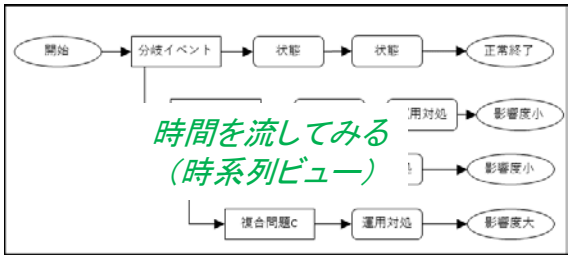
自作ツール（EXCELアドイン）による自動変換



発生条件の識別



リスクシナリオ発生条件の確認



リスクシナリオの確認

まとめ

- 結論
  - 提案手法は、過去欠陥/不具合の発生条件の識別、異常時のシナリオテスト設計を支援する
  - 提案手法は、GSNを活用することで、動作環境やSW処理などの複数の条件組み合わせを検討できる
  - 提案手法は、ガイドワード適用を促進することで網羅性を向上できる

- 今後の展開
  - 提案手法の適用拡大、費用対効果の計測
    → ご協力いただける方募集中！
  - 思考経緯が各ビューとして可視化されるため、技術継承への活用
  - シナリオテスト設計からみた、欠陥情報の記載方法の改善や教育支援

ご清聴　ありがとうございました