

ソフトウェア要求、設計、製造、試験、保守、運用を含むソフトウェア開発・運用プロセスの全ライフサイクルを一貫して支援するディペンダビリティ保証技術の研究に取り組んでいる。また、理論、技術を実際のソフトウェア開発に適用し、その効果を確かめる実証のソフトウェア工学に取り組んでいる。

## A. ディペンダビリティ保証技術

### アシュアランスケース

ディペンダビリティを保証にはアシュアランスケースを用いる。図1はアシュアランスケースの例であり、システムの応答遅延障害を軽減するという要求を構造化し、議論できるようにしている。

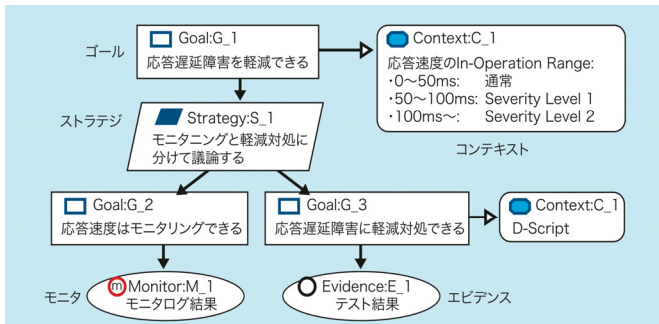


図1 アシュアランスケースの例(応答遅延障害の軽減)

### テーマA-1 アシュアランスケース用語辞書構築方法

アシュアランスケースで記述されている用語や用語間の関係を明確にするために、人、物、活動等を要素として、その要素間の関係をまとめた用語関係図(Word Relationship Diagram)を定義した辞書の構築方法を提案している。アシュアランスケース(図2)と用語関係図(図3)との相互変換可能な生成規則を提案している。

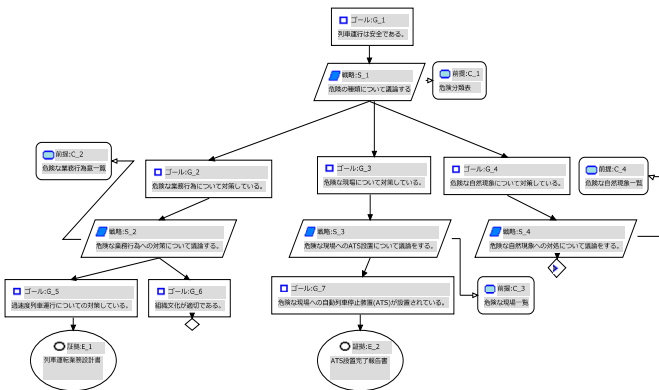


図2 列車運行の安全性のアシュアランスケース

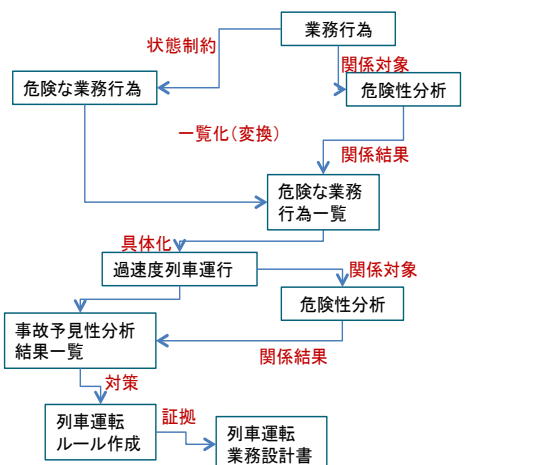


図3 列車運行の安全性のアシュアランスケースの用語関係図

### テーマA-2 非機能要求グレードのシステム適用についての考察

IPAが提案している非機能要求を視覚化して確認する手段である非機能要求グレードを用いて、既存システムの信頼性指標を試行し、システム適用の効果と課題を明らかにする。名古屋大学ポータルサービスを対象とし、システムを開発、管理している機構である情報基盤センターに対して調査し、非機能要求グレードの指標評価の項目で構成される項目を得た(表1)。

表1 信頼性特性と信頼性評価指標

| 信頼性特性       | 特性項目  | 指標数 |
|-------------|---|-----|
| 可用性(8)      | 運用時間(通常), 業務継続性, 目標復旧水準(通常), 目標復旧水準(大規模災害時), 稼働率, 耐障害性, 災害対策, 回復性                               | 24  |
| 性能・拡張性(7)   | 通常時の業務量, 業務量増大度, 保管期間, 性能目標値バッチ, オンラインスループット, バッチスループット   | 26  |
| 運用・保守性(11)  | 計画停止, 運用負荷削減, 運用保守, 復旧作業, 異常検知対応, 運用時間, バックアップ, 運用監視, 交換用部材の確保, 運用環境, 運用管理方針                    | 48  |
| 移行性(4)      | スケジュール, データ, リハーサル, 移行トラブル  | 14S |
| セキュリティ(10)  | コンプライアンス, セキュリティリスク分析, セキュリティ診断, セキュリティリスク管理, アクセス・利用制限, データの秘匿, 不正監視, ネットワーク対策, マルウェア対策, Web対策 | 34  |
| 環境・エコロジー(3) | 制約条件, システム特性, 環境マネジメント  | 17  |
| 合計          | 43項目  | 162 |

## B. 実証的ソフトウェア工学

### テーマB-1 実証的アプローチによるソースコード読解戦略の分析

100名以上のソフトウェア開発の実務者を対象に同一のソースコード差分4件を読解してもらい、コードの読解時間と欠陥指摘の正確さ(指摘レベル)を比較した。読解時間と指摘レベルの間には有意な差がないこと、レビュー経験が指摘レベルに影響を与えていることがわかった。対象のソースコードにおいて時間をかけたからといって正確に読めているわけではないことが示された。

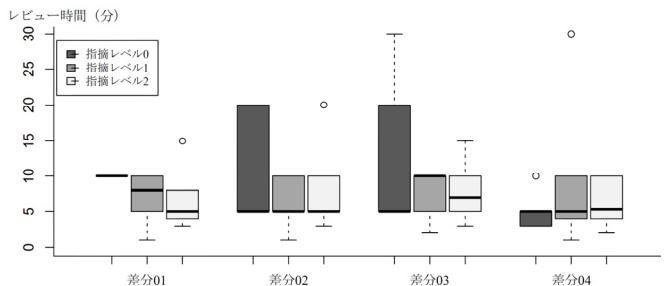


図4 欠陥指摘の正確さ(指摘レベル)とソースコード読解時間

### テーマB-2 欠陥データベースのマイニング

過去のバグレポートからソフトウェアレビューで優先的に検出すべき欠陥を抽出するマイニング手法を提案している。過去のバグレポートにおいて修正時間が他より長いもの、欠陥の深刻度が大きいものを選択し、含まれるキーワードを抽出する。銀行システムの構築時に収集したバグレポートから「日付」「期間」といった銀行システムにおいて特に重要なキーワードを抽出できた。