

故障事例によるテスト観点知識ベース構築とテスト設計への適用

Construction of Test viewpoint knowledge base by failure examples,
and application to the test design

株式会社日立製作所 情報・通信システム社 IT プラットフォーム事業本部

開発統括本部 ソフトウェア本部 品質保証部

Hitachi, Ltd., Information & Telecommunication Systems Company

IT Platform Division Group, IT Platform R & D Management Division, Quality Assurance Department

○光永 洋 ¹⁾田中 浩和 ¹⁾○Hiroshi Mitsunaga ¹⁾Hirokazu Tanaka ¹⁾**Abstract**

As an practical use of the defect information, we considered using the failure information occurred in the market for a test design. The existing test know-how is used in the present test design. However, the contents with high level of abstraction are not fully be utilized by engineer who is inexperienced in test design. In order to solve this problem, we classified and marshaled the failure information occurred in the market, and generated a "test viewpoint knowledge base by the failure examples" by organizing the contents with lower level of abstraction. The test design was able to be strengthened with applying this knowledge base to a test design process. Furthermore, by having set up the know-how of the low level of abstraction, it leads to a better understanding of the existing test know-how with the high level of abstraction.

1. はじめに

我々の組織では、金融システムや公共システムなどの基盤となる汎用ソフトウェアの品質保証を担当しており、様々な観点で品質を保証することが大きな役割のひとつである。また、開発はウォーターフォールモデルの形態がほとんどであり、長年蓄積したノウハウを活用し、設計・製造の上流工程では不良の作り込み防止、およびテストの下流工程では不良の検出漏れ防止の取り組みを行っている。そのような取り組みによって市場で発生する故障は年々減少しているものの、少なからずいくつかの故障が市場で発生しているのが現状である。

そこで本研究では、過去に市場で発生した故障を分析しそれらの故障の特徴および現状の問題点を明らかにした。そして、故障事例を分類、整理しテスト観点知識ベースの構築を行い、それをテスト分析、テスト設計に活用することで市場での故障の発生の低減を目指した。

本研究は、テスト業務の経験が浅い技術者とテスト技術者全般における問題と対策に着目し、故障事例の分類・整理とその活用に関して取り組んだものである。従来研究^{[1][2]}も同種の取り組みであるが、本研究は、市場発生故障の低減を狙い、テストでの検出難易度に着目した点、テスト観点の抽象度について検討した点が異なっている。

本論文では、2章で市場発生故障を分析し現状の問題点を整理し、3章でその問題点を解決するために「故障事例によるテスト観点知識ベース」の構築、およびテスト分析、テスト設計での適用方法について述べ、4章で適用の評価を行う。

1) 株式会社日立製作所 情報・通信システム社 IT プラットフォーム事業本部開発統括本部 ソフトウェア本部 品質保証部

Hitachi, Ltd., Information & Telecommunication Systems Company IT Platform Division Group, IT Platform R & D Management Division Quality Assurance Department

2. 現状分析

2.1 市場で発生した故障の分析

(1) 故障に対するテストでの検出難易度の定義

今回は、市場で発生した故障が社内テストで検出可能であったかどうかについて調査することとした。そこで、テストでの故障検出難易度を定義し、その定義に従い市場で発生した故障について分析することとした。故障の検出難易度は、表 1 に示すように故障の発生条件に着目し、動的テストでの検出し難さを高、中、低の 3 段階にわけ定義した。

表 1 故障に対するテストでの検出難易度

検出難易度	定義内容
高	マルチプロセサ環境などの条件下で、タイミングにより発生する問題であり、検出が困難であるもの
中	製品や機能の組合せ条件で発生する問題であり、組合せの網羅性を確保すれば検出可能であったもの
低	主に単一条件で発生する問題であり、テスト分析を十分に実施すれば検出可能であったもの

(2) 過去 10 年間の市場で発生した故障の分析

過去 10 年間に市場で発生した弊社製品の故障を表 1 に示したテストでの検出難易度にて分類した結果を図 1 に示す。横軸は過去 10 年間の各年度を示しており、縦軸は表 1 に示したテストでの検出難易度に該当する故障件数の比率を表している。

調査の結果、社内テストでの検出が必ずしも難しくない検出難易度が低から中の故障が、全体の約 90% を占めている状況であった。

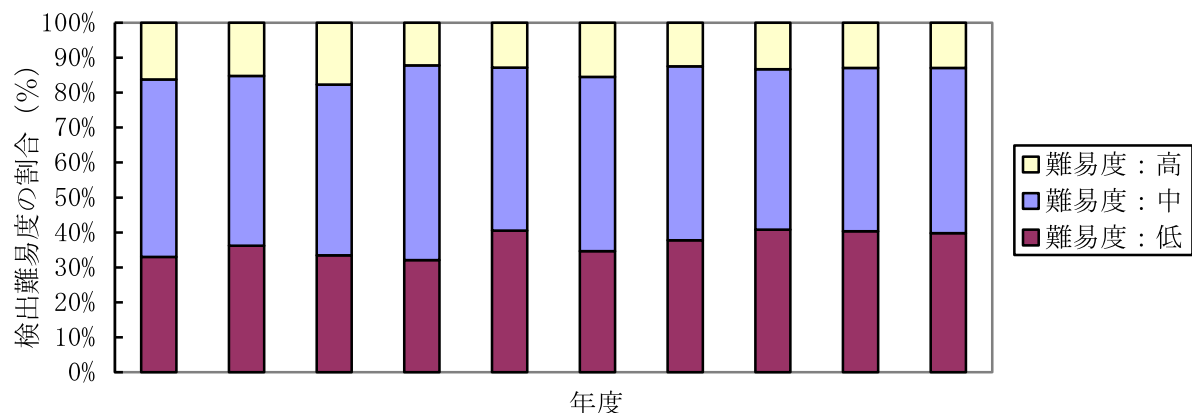


図 1 過去 10 年間の市場で発生した故障のテストでの検出難易度比率

また、故障内容の例としては表 2 に示すものがあつた。項番 1, 2 は製品が一般的に備える機能における故障事例であり、項番 3, 4 は製品が動作する環境条件により発症する故障事例である。項番 5, 6 は一般的な機能や環境条件には該当しない処理方式の誤りによる故障事例である。

表 2 市場で発生した故障事例（抜粋）

項番	故障内容
1	検索機能で“and”を含むと SQL エラーが発生して、検索機能が動作しない。
2	製品を更新インストールすると設定ファイルが初期化されてしまう。
3	”C:\Program” フォルダが存在する環境で製品をインストールすると失敗する。
4	ホスト名が 32 文字の環境で製品を起動すると、メモリアーバーランが発生する。
5	OS 関数呼出しのエラー処理がなく、OS 関数がエラーを返して処理続行不能に陥る。
6	メモリ初期化漏れにより製品が異常終了してしまう。

2.2 問題の整理

2.1 で示した，社内テストでの検出が必ずしも難しくない故障が90%に達している原因について考察した結果，ひとつの要因として，テスト分析，テスト設計が十分になされていない問題があると考えた。また，その原因には，次の2つがあると考えた。

- (1) 既存のノウハウは抽象度が高く、テスト業務の経験の浅い技術者では、十分なテスト分析、テスト設計ができない。
- (2) 故障に対する知識が十分に共有できておらず、テスト業務の熟練者でもテスト分析、テスト設計に反映できない場合がある。

我々のテスト設計プロセスを図 2 に示す。通常のテスト設計プロセス^[3]との相違点は、テスト設計ノウハウを活用すること、およびテスト分析ではテスト観点表として、テスト対象を機能単位に細分化しテスト観点をまとめることである。本研究では文献^[5]を参考にし、テスト観点を「テスト対象に対してテスト目的から設定するテストすべきことの概要」と定めた。文献^[4]に基づく、テスト観点のうちテスト目的に主眼を置いた観点といえる。例えば、GUI のテスト対象に対して、使用性におけるテスト目的からは、「入力値が不正な場合は不正内容を使用者に知らせること」というテスト観点が設定される。

以下に、図 2 のテスト設計プロセスの各手順について説明する。

- ① テスト計画にてテスト計画書を作成する。
- ② テスト分析では、テスト計画書に従い製品仕様書等のテストベースからテスト条件を明確にするために、テスト設計書に相当するテスト観点表を作成する。テスト観点表とは、表 3 に示すように、機能と品質特性をマトリックスにし、各組合せにおいて、テスト観点を抽出したものである。
- ③ テスト設計にてテストケースに相当するテスト項目を作成する。
- ④ 最後にテスト実装にてテスト手順を作成する。

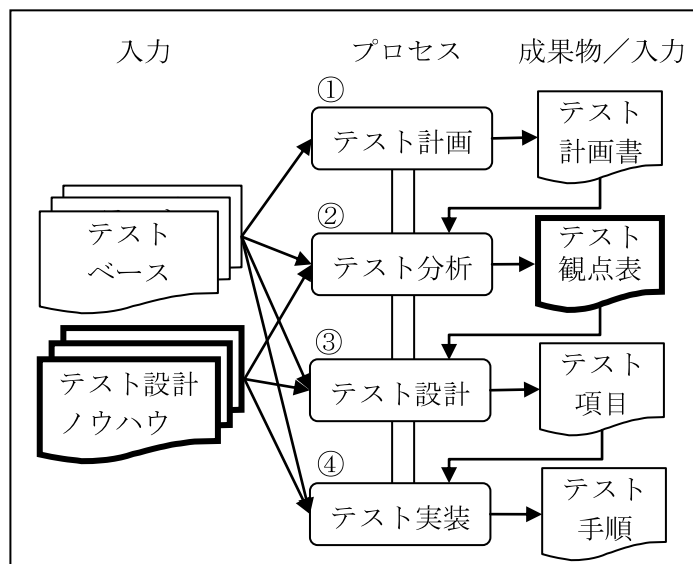


図 2 テスト設計プロセス

表 3 テスト観点表

機能			機能性			…
大項目	中項目	小項目	合目的性	機能性	相互運用性	…
			<div>テスト観点</div>			

とともに洗練され抽象度が高い表現にまとめられている状態である。

表 2 に示した市場で発生した故障事例の項番 1 を例として、社内のテスト設計ノウハウで該当するものを抜粋したものが表 4 である。

表 4 テスト設計に使用する既存のテスト設計ノウハウ（抜粋）

項番	チェックリスト内容
1	特異な入力要因の特徴を考慮した 境界／限界の確認をしたか。
2	異常パラメータの入力による操作に関する項目を設定する。
3	異常なデータのタイプインに関する項目を設定する。

テスト業務の経験を重ねた熟練者は、当該テスト設計ノウハウにまとめられた抽象度の高い内容のみで、テスト対象に対して適切なテスト分析およびテスト設計が可能である。しかし、テスト業務の経験の浅い技術者では、具体化が難しいため適切なテスト設計ができないという問題がある。また、表 2 に示した市場で発生した故障事例の項番 3 などは、故障の知識がなければ、テスト業務の熟練者が既存のノウハウを活用しても適切なテスト設計が難しい例である。そのため、故障に対する経験・知識が必要でその共有が不足している問題がある。

3. 解決策の検討

3.1 故障事例によるテスト観点知識ベースの構築と適用の検討

2.2 で示した問題に対して、市場での故障事例を適切な抽象度によるテスト観点としてまとめた知識ベースを構築した。この知識ベースをテスト分析およびテスト設計工程の入力情報とすることで、次に示すテスト観点の強化を図った。

- (1) テスト業務の経験が浅い技術者に対する既存の抽象度の高いテスト観点の理解促進。
- (2) テスト技術者全般に対する経験のない故障に関するテスト観点の設定強化。

3.2 テスト観点の抽象度の検討

既存のノウハウは抽象度が高く、テスト業務の経験の浅い技術者では十分なテスト設計ができない問題の対策として適切な抽象度を検討した。

テスト観点の抽象度と対応するテスト観点例を、表 2 に示した市場で発生した故障事例の項番 1 に対して、表 4 の既存のテスト設計ノウハウの項番 3 のテスト観点を適用した例を表 5 に示す。

抽象度が高いと製品のどの部位にも適用できる汎用的な観点となるが、適用すべきところの解釈に幅ができてしまう。いっぽう抽象度を低くすると、テスト観点例は具体的になるものの、他の部位へ転用が難しくな

表 5 テスト観点の抽象度でテスト観点例

る。今回はテスト業務の経験の浅い技術者を対象としたため、抽象度の低いテストケースに近い具体的なテスト観点にすることで理解を促進し、その積み上げで抽象度が高いテスト観点でも早期に理解できるようにした。

テスト観点の抽象度	テスト観点例
高	異常なデータのタイプインに関する項目を設定する
中	外部入力データを SQL に渡す場合は特異なデータの入力に関する項目を設定する
低	検索機能では、特異なデータである“and”や“or”を含む文字列を入力して、SQL エラーとならず正しく検索できる項目を設定する

3.3 経験のない故障に対するテスト観点強化の検討

テスト業務の熟練者を含め、テスト技術者全般において、経験のない故障に対してテスト観点を強化する検討をした。

3.2 に示した抽象度の低いテスト観点に加えて、表 6 に示すようにテスト観点とその根拠となった故障事例を対の情報としてくり付けすることで、経験のない故障に関する知識の共有を図ることとした。

また、合わせて知識の共有に加えテスト観点の更なる理解促進にも寄与できると考えた。

表 6 テスト観点と故障事例（抜粋）

テスト観点（テストすべきこと）	故障事例
検索機能では、“and”や“or”を含む文字列を入力して、SQL エラーとならず正しく検索できること。	検索機能で“and”を含むと SQL エラーが発生して、検索機能が動作しない。
製品を上書きや更新インストールした場合に、デフォルト以外に設定した内容が引き継がれること。	製品を更新インストールすると設定ファイルが初期化されてしまう。
特定 OS 環境に対応する製品は、“C:\Program” フォルダが存在する環境でインストール確認すること	“C:\Program” フォルダが存在する環境で製品をインストールすると失敗する。
ホスト名が 32 文字の環境で製品の動作確認をすること。	ホスト名が 32 文字の環境で製品を起動すると、メモリアーバーランが発生する。
各種 OS 関数がエラーを返した場合の動作確認をすること。	OS 関数呼出しのエラー処理がなく、OS 関数がエラーを返して処理続行不能に陥る。
メモリを意図的に汚した状態で、製品を動作させ、誤動作しないことを確認すること。	メモリ初期化漏れにより製品が異常終了してしまう。

3.4 テスト観点の分類・整理

3.2, 3.3 で設定したテスト観点と故障事例を活用するためには、テスト対象となる製品に対してテスト観点が適切に検索できる必要がある。そのため、テスト観点と故障事例を分類・整理方法について検討した。

今回は、表 7 に示すように機能、環境、処理にて分類した。これは、図 2 ②で実施するテスト対象のテスト分析結果から抽出されたテスト観点到直接関連付けられるようにするためである。また、それぞれの分類を副分類として細分化した。副分類は、機能、環境、処理のそれぞれで、テスト観点と故障事例から抽出したものである。

表 6 に示すテスト観点と故障事例を表 7 による分類で整理し、表 8 に示す「故障事例によるテスト観点知識ベース」とした。

表 7 テスト観点の分類

分類	副分類	内容
機能	検索機能 インストール機能 :	製品が備える一般的な機能にて発生する故障で、機能単位の分類
	特定 OS 環境 ホスト名 :	
環境	OS 関数 メモリ初期化 :	機能や環境に依存せず製品が有する一般的な処理にて発生する故障で、処理内容により分類

表 8 故障事例によるテスト観点知識ベースの記載内容（抜粋）

分類	副分類	テスト観点（テストすべきこと）	故障事例
機能	検索機能	検索機能では、“and”や“or”を含む文字列を入力して、SQL エラーとならず正しく検索できること。	検索機能で“and”を含むと SQL エラーが発生して、検索機能が動作しない。
	インストール機能	製品を上書きや更新インストールした場合に、デフォルト以外に設定した内容が引き継がれること。	製品を更新インストールすると設定ファイルが初期化されてしまう。
環境	特定 OS 環境	特定 OS 環境に対応する製品は、“C:\Program” フォルダが存在する環境でインストール確認すること	“C:\Program” フォルダが存在する環境で製品をインストールすると失敗する。
	ホスト名	ホスト名が 32 文字の環境で製品の動作確認をすること。	ホスト名が 32 文字の環境で製品を起動すると、メモリアーバーランが発生する。

処理	OS 関数	各種 OS 関数がエラーを返した場合の確認をすること。	OS 関数呼出しのエラー処理がなく、OS 関数がエラーを返して処理続行不能に陥る。
	メモリ初期化	メモリを意図的に汚した状態で、製品を動作させ、誤動作しないことを確認すること。	メモリ初期化漏れにより製品が異常終了してしまう。

3.5 故障事例によるテスト観点知識ベースのテスト分析、テスト設計への適用

構築したテスト観点知識ベースを含め、各種ドキュメントの位置づけは、表 9 の通りとなった。

表 9 テスト分析・テスト設計に使用するドキュメント

対象ドキュメント	内容
テスト設計ノウハウ	過去の開発時の経験や故障事例等のノウハウが蓄積されたチェックリスト形式のテスト観点集であり、テスト観点の最も抽象化したものが記述されている。
テスト観点知識ベース	故障事例に基づいて最も抽象度の高いテスト観点を、より詳細化・具体化した様々なレベルのテスト観点からなるデータベース。
テスト観点表	テスト分析者がテスト対象製品の各機能に対してテスト分析を実施し、作成するテスト観点表。テスト設計ノウハウ、テスト観点知識ベースをそのまま、あるいはより詳細化・具体化して記載する。

テスト観点知識ベースをテスト分析・テスト設計に適用した。具体的には図 3 に示すように、既存のテスト設計ノウハウとともに、テスト観点知識ベースをテスト分析、テスト設計における入力情報とした。

テスト分析のテスト観点表作成においては、テスト対象の持つ機能や、動作環境をテスト観点知識ベースの副分類と照らし合わせ該当するテスト観点を抽出する。そのテスト観点をテスト対象のテスト目的（品質特性）におけるテスト観点のひとつとして加える。

また、テスト設計のテスト項目作成においては、設定したテスト観点の適用部分の分析において、漏れがないかテスト観点知識ベースにて確認を行う。これらにより、テスト観点およびテスト項目の強化を行う。

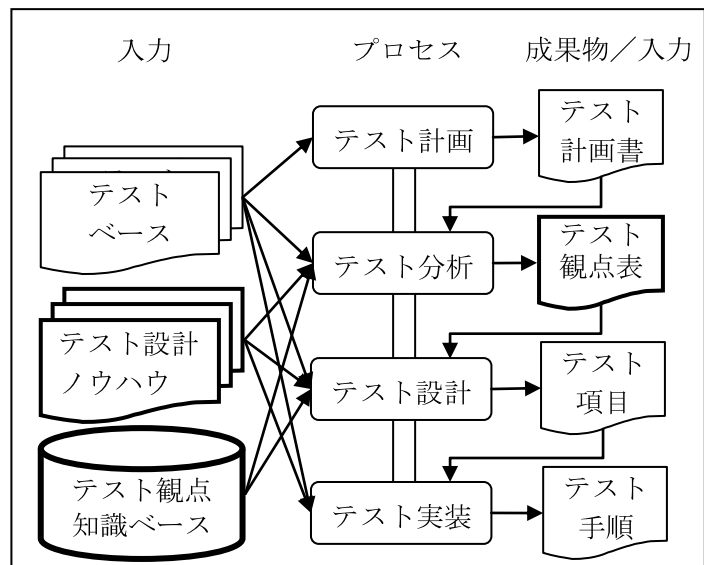


図 3 テスト観点知識ベースを適用したテスト設計プロセス

4. 適用および評価

4.1 定量的評価

(1) 開発プロジェクトへの適用

複数の開発プロジェクトにて、「故障事例によるテスト観点知識ベース」の適用を図った。適用前後における、テスト観点数やテスト項目数の比較を表 10 に示す。プロジェクトの規模や成熟度、所有するノウハウの多寡に依存して追加件数には差が見られたが、多くのプロジェクトにてテスト観点・テスト項目の強化が実現できた。

表 10 故障事例によるテスト観点知識ベース適用によるテスト設計の強化結果

適用プロジェクト	追加テスト観点数・ 追加テスト項目数	総テスト観点数・ 総テスト項目数	追加観点・ 項目割合
プロジェクト A	テスト観点数 13 観点	テスト観点数 106 観点	3%
プロジェクト B	テスト観点数 8 観点	テスト観点数 103 観点	7%
プロジェクト C	テスト項目数 431 項目	テスト項目数 2321 項目	19%
プロジェクト D	テスト項目数 136 項目	テスト項目数 1396 項目	10%

(2) テスト実施済み製品への適用

弊社製品のある製品グループに属するテスト実施済み製品の 18 製品を対象に、「故障事例によるテスト観点知識ベース」にてテスト観点を見直しテスト設計・テスト実装・テスト実施を行った結果 6 件の不具合を検出した。

4.2 定性的評価

「故障事例によるテスト観点知識ベース」を適用したプロジェクトへのヒアリングを実施した。その結果、次のような感想を得ることができ、効果を実感できるとともに問題点もあった。

- (1) テスト観点と故障事例を合わせて見ることでテスト観点の理解が深まり、既存観点の有効性検証に利用できた。
- (2) 市場で発生した故障事例のためテスト観点に説得力があり、品質保証部だけでなく設計部門でも適用の必然性を理解した。
- (3) テスト観点知識ベースに収録したテスト観点が多いため、テスト分析、テスト設計に工数を要することになった。そのため、テスト観点の集約や優先度付けによるテスト観点知識ベースの整理が必要である。

4.3 市場で発生した故障の推移

2.1 で示した、過去 10 年間の市場で発生した故障の分析と同様の分析を、施策適用開始後 3 年間の市場で発生した故障に対して実施した。結果は、図 4 に示すように、適用前の過去 10 年間に比べ、適用 3 年目では、テストでの検出難易度が低から中の故障が占める割合は、約 90% から約 80% となった。ただし、本施策が市場における故障発生件数に影響を及ぼすためには多くの時間を必要するため、今後、本施策の適用プロジェクト・製品を拡大することにより、本施策の有用性を継続して確認していく予定である。

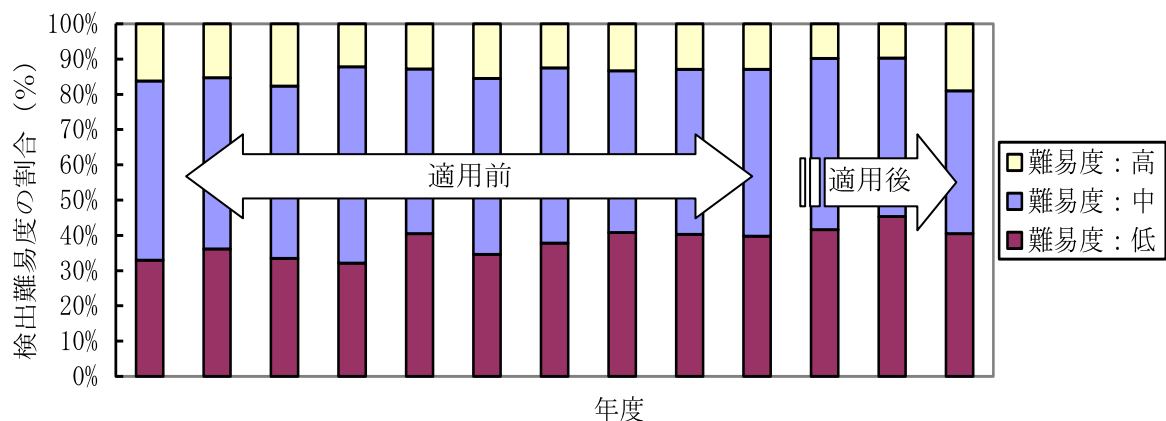


図 4 過去 13 年間の市場で発生した故障のテストでの検出難易度比率

5. まとめ

今回、市場で発生した故障情報を社内でのテスト分析、テスト設計に活用する施策を検討、適用した。テスト分析、テスト設計に適用するために、故障事例を「故障によるテスト観点知識ベース」として、テスト観点の形式でまとめた。テスト観点は、テスト分析、テスト設計の経験が浅い技術者でも理解、活用できるように抽象度の低い具体的な内容にするとともに、テスト分析結果と直接対応付けられる分類とした。これらの施策適用によって、次に示すようなテスト分析およびテスト設計における強化が図れたと考える。

- (1) テスト業務の経験が浅い技術者に対する既存の抽象度の高いテスト観点の理解促進。
- (2) テスト技術者全般に対する経験のない故障に関するテスト観点の設定強化。

「故障事例によるテスト観点知識ベース」の構築、および適用により、テスト観点強化からテスト実施での故障検出増加を期待でき、市場での発生故障件数を低減につながると推察される。しかし、市場での故障件数の低減を確認し、それが本施策の効果であることを分析・確認するにはまだ多くの時間を必要とする。

今後は、市場で発生した故障事例のテスト観点へのタイムリーなフィードバック、設計・製造工程での適用推進を進めていく予定である。

参考文献

- [1] 細川宣啓 ほか, Project Fabre presents 欠陥マスター情報構築ワークショップー「悪さの知識」伝承によるバグ予防の為のバグ情報マスター化実践ー, JaSST' 12 Tokyo
- [2] 大谷和夫 ほか, 上流設計工程における未然防止プロセスの提案 -未然防止リストの活用と欠陥の発想-, JaSST' 12 Tokyo
- [3] ISTQB, Certified Tester Advanced Level Syllabus, 2007
- [4] 智美塾塾長+塾生一同, テストアーキテクチャ解説 ～テストアーキテクチャ設計を実践するには～(秋山浩一, テストアーキテクチャ設計の概念), JaSST' 12 Tokyo
- [5] 西 康晴, テスト設計におけるモデリングのための記法の提案, JaSST' 06 Tokyo
- [6] 杉田正実 ほか, テスト設計書を書こう！ -抜け漏れのないテスト設計書テンプレートの提案-, JaSST' 09 Tokyo