人工知能学会
第13回知識流通ネットワーク研究会
SIG-KSN-013-04

保証ケース手法に基づくテスト十分性に関する合意形成手法の提案

山本 修一郎

名古屋大学
愛知県名古屋市千種区不老町

A Proposal on Consensus Building Method to assure the Sufficiency of Testing with Assurance case

Shuichiro YAMAMOTO

Nagoya University
Furo-cho, Chikusa-ku, Nagoya Aichi Japan

概要

テストの十分性を確認するためには，システムを必要とする発注者，システムを実装する開発者，実装されたシステムが要求を満たすことを確認する検証者とのテスト項目についての合意が必要である．本稿では，保証ケースを用いてテスト項目が必要な確認内容を含むという主張を証拠によって立証することにより，テスト項目の十分性を保証する方法を提案する．

Abstract

To assure the sufficiency of testing, it is necessary to build consensus among stakeholders such as system acquirers, developers, and test engineers who validate the implementation of the system satisfies requirements. In this paper, an approach is proposed to assure the sufficiency of testing by developing claims and evidences that test items include necessary validation activities.

1 はじめに

テストの十分性を確認するためには，システムを必要とする発注者，システムを実装する開発者，実装されたシステムが要求を満たすことを確認する検証者とのテスト項目についての合意が必要である．

保証ケース(Assurance case)では，立証したい主張を証拠に基づいてステークホルダ間で議論することによって，主張についての合意を形成できる[1]-[6]．保証ケースは，これまで，テスト結果を証拠としてシステムの安全性を確認するために，用いられてきた．最近では，システムのディペンダビリティを確認するために，ディペンダビリティケース(Dependability case)[7]と呼ばれることもある．筆者らもディペンダビリティケース（D-Case）と称している[8]-[28]．しかし，テストの十分性自体を保証ケースによって確認する方法については研究されていない．このため，本稿では，保証ケースを用いてテスト項目が必要な確認内容を含むという主張を証拠によって立証することにより，テスト項目の十分性を保証する方法を提案する

以下では，2 節で関連研究について述べる．3 節で保証ケースを用いたテスト工程の十分性についての合意形成手法を提案する．4 節で提案手法の具体例を説明する．5 節で考察を述べ，最後に 6 節でまとめと今後の課題を明らかにする．

2 関連研究

以下では，保証ケースとテスト工程の完了基準についての研究動向を紹介する．

2.1 保証ケース

保証ケースの構成要素は，主張（claim），説明（strategy），コンテクスト(context)，証拠（evidence），未展開である．構成要素間の関係は，主張，説明，証拠を関連付ける矢線と，主張と説明をコンテクストと関連付ける白抜きの矢線の 2 種類である．

【主張】システムが達成すべき性質を示す矩形．下位主張や戦略に分解される

【説明】主張の達成を導くために必要となる論証を示す平行四辺形．下位主張や下位説明に分解される．

【コンテクスト】主張や説明が必要となる理由としての外部情報を示す楕円．

【証拠】主張や説明が達成できることを示す証拠

【未展開】まだ具体化できていない主張や説明を示すひし形

保証ケースの例を図 1 に示す．この図では「システムがディペンダブルである」という主張 G1 を前提条件に基づいて S1 で 3 つの下位の主張 G2「前提条件の下でシステムを正常に運用可能」,G3「発生した想定内の逸脱に対してシステムを継続運用可

能」,G4「発生した想定外の逸脱に対してシステムを復旧可能」に分解することよって説明している．G2,G3,G4 の主張が成立する根拠として，それぞれ 3 つの証拠 E1「前提条件に基づくシステム開発完了報告」,E2「前提条件に基づく想定リスク対策実施完了報告」,E3「想定外の逸脱に対する運用プロセス定義完了報告」が提示されている．

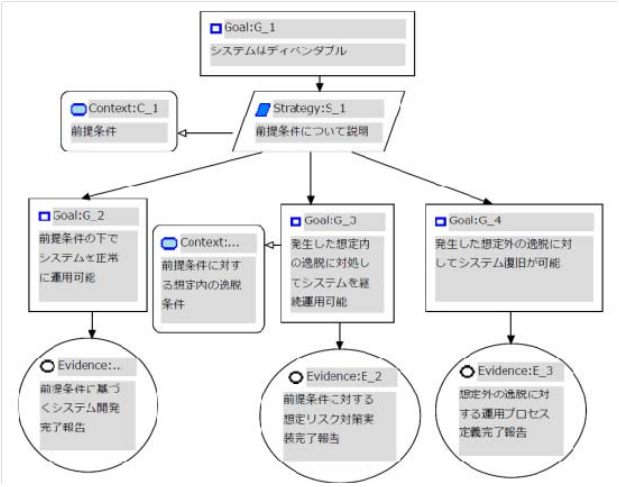


図 1　保証ケースの例

ISO/IEC 15026 では，保証ケースの構造と内容に対する最低限の要求を規定している．保証ケースには，システムや製品の性質に対する主張（claim），主張に対する系統的な議論(argumentation)，この議論を裏付ける証跡(evidence)と明示的な前提(explicit assumption)が含まれる．ISO/IEC 15026 では，保証ケースが持つべき構造と内容を規定対象としており，保証ケースの品質については規定していない．また保証ケース（Assurance case）については，ISO/IEC 15026[12]や OMG の ARM(ARgument Metamodel)[13]と SAEM(Software Assurance Evidence Metamodel)[14]などで標準化がすすめられている．ISO/IEC 15026 では，対象範囲，適合性，利用法，保証ケースの構造と内容，適用成果物などについて規定している．

最近では ISO26262 で，自動車分野のおける機能安全を保証するために，開発対象だけでなく開発プロセスの安全性についても保証ケースの適用が推奨されている．しかし，テスト工程に対して保証ケースをどのように適用すべきかについての具体的な言及はない．

2.2 テストの目的

IEEE Std. 829-2008[29]によれば，システムとその関連生産物に対して、以下の条件を満たすように，具体的な証拠を提供することがテストの目的である。

a) 要求をシステムが満足すること
b) 物理法則、ビジネス規則、前提条件に基づいて適切に問題を解決すること
c) 意図された利用とユーザニーズをシステムが満足すること

3 保証ケースを用いたテスト十分性の確認

以下では，保証ケースとテストとの関係ならびに，保証ケースを用いたテスト十分性を保証する手順について述べる．

3.1 保証ケースとテストの関係

保証ケースとテストの関係には、保証ケースの証拠としてテスト結果を用いる方向と、保証ケースを用いてテストの十分性を確認する方向がある。

2.2 節で述べたテストの目的に対する保証ケースの例を示すと図 2 のようになる．

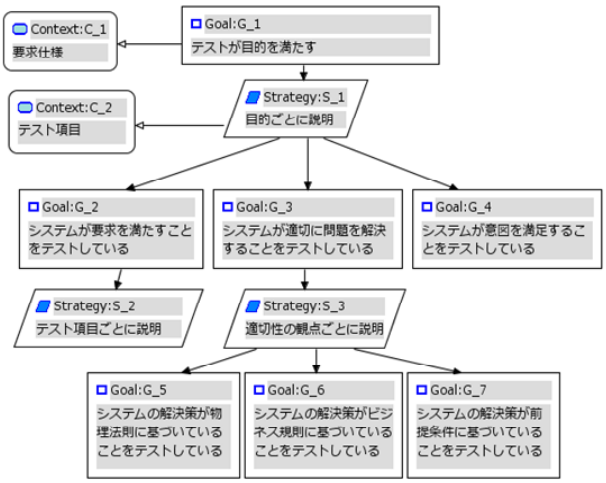


図 2 テスト目的に対する保証ケースの例

以下では、テストの十分性に対する保証ケースの作成手順を提案する。

3.2 テスト十分性に対する保証ケース作成手順

テストが十分であることを説明するためには、①システムが前提条件の下で要求仕様を満足することと、②前提条件からの逸脱に対してシステムが不具合を生じないことを確認するために、十分なテスト項目が用意されていることを示す必要がある。

最初の条件①が正常系テスト項目に、次の条件②が例外系テスト項目に対応する。例外系テスト項目を抽出するためには要求仕様の逸脱（要求逸脱）を分析する必要がある。ユースケースなどによる仕様記述では自然言語で記述されているため、十分に要求逸脱を抽出することが困難である。このため、ISO/IEC/IEEE std.29148[30]に基づいた要求記述表を利用することにより要求仕様の逸脱を識別する必要がある。

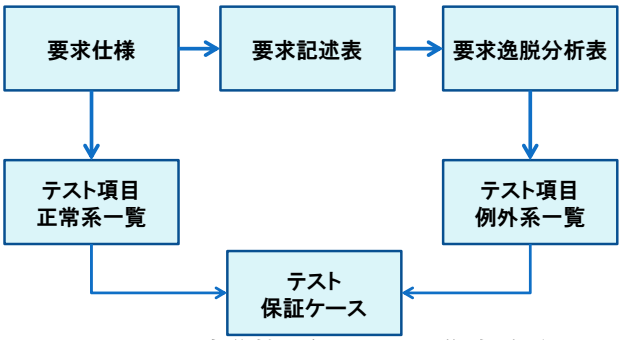


図 3 テスト十分性の保証ケース作成手順

このような準備に基づいてテスト十分性を確認するための保証ケースの作成手順をまとめると、図 3 のようになる。

要求記述表では、表 1 に示すように，主体、対象、条件、機能、制約を記述する．主体として、要求をみたそうとするシステムを記述する。対象として、システムが要求に従って操作する資源（データ，被制御装置など）を記述する。

条件部では、主体状態、対象状態、イベント条件、入力条件を記述する。

イベント条件では、機能を実行する契機となるイベントの内容とタイミングについての条件を記述する。

機能部では、入力に基づいて出力を生成する機能の内容を記述する。

制約部では、機能を実行する上での条件として、応答制約、出力制約、値制約を記述する。イベント事象に対応する応答事象の内容とタイミングについての制約条件が応答制約である。

要求逸脱分析表では、表 2 に示すように、要求仕様ごとに、パラメータ、ガイドワード、逸脱とその重大性、識別 ID を表形式で系統的に記述する。

パラメータには、要求記述項目に基づいて、システム、対象、イベント事象、入力、出力、応答事象を記述する。パラメータの逸脱を分類するガイドワードとして、なし、以外、部分、冗長、遅い、早いなどを用いる。ここで、パラメータの逸脱に起因する影響の重大性を評価することによって、テスト項目を選択することができる点に留意してほしい。これによって重要性の高いテスト項目を優先して選択できる。

表 1　要求記述表の構成

要求仕様	主体が対象に対して，条件のとき，制約を満たすように機能を実行する		
主体	条件	機能	制約
前提条件：	主体状態： 対象状態： イベント条件：		応答制約： 出力制約： 値制約：
対象			
前提条件：	入力条件：		

表 2　要求逸脱分析表

要求仕様	主体が対象に対して，条件のとき，制約を満たすように機能を実行する		
パラメータ	ガイドワード	逸脱／重大性	ID
システム 対象 イベント事象 入力 出力 応答事象	なし／以外／部分／冗長 遅れ／早い		

テストの十分性を確認するための保証ケースの構成例を図 4 に示す。

この図では、まずシステムが要求を満足することを要求ごとに説明している。次に、各要求をシステムが満たしていることを示すために、要求ごとに正常系と異常系テストに分けて説明している。この図では要求 R-1 についてだけ分解して示している。異常系については、逸脱種別として、条件逸脱、機能逸脱、制約逸脱に分けて、テストの十分性を説明するようにしている。

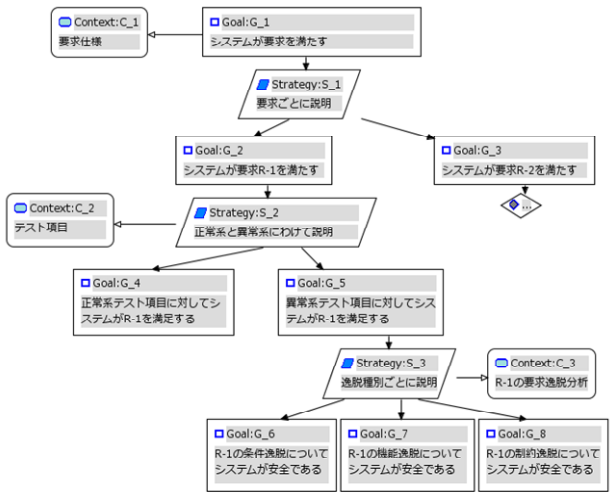


図 4 テストの十分性を確認する保証ケースの概要

4 具体例

上述したテスト保証ケースの作成手順を付表 1 に示した「ビデオ貸出機能」に対する要求仕様の記述例[31]を用いて説明する。

4.1 要求記述表

ビデオ貸出機能に対する要求記述表の例を表 3 に示す。

表 3　要求記述表の例

要求仕様	主体が対象に対して，条件のとき，制約を満たすように機能を実行する 会員がビデオを借りる[システムがビデオ貸し出し画面を表示する]		
主体	条件	機能	制約
前提条件： ・メニュー画面に[ビデオ貸し出し]リンクが表示されている ・システム環境？	主体状態： 会員情報が識別されている？ メニュー画面を表示している？ 対象状態： 会員情報？ イベント条件： メニュー画面の[ビデオ貸し出し]リンク押下 入力条件： 会員情報？	会員がメニュー画面の[ビデオ貸し出し]リンクを押したときに開始する システムがビデオ貸し出し画面を表示する	応答制約： 出力タイミング？ 出力制約： 画面表示制約？ ビデオ本数サイズ？ 値制約：
対象			
前提条件： ・会員情報がある？ ・会員番号？ ・配送システムの状態？			

表 3 では、付表 1 の仕様 1−1−1 と 1−1−2 に対して示している。同様にして、仕様 1−1−3 と仕様 1−1−4、ならびに、仕様 1−1−5、仕様 1−1−6、仕様 1−1−7 に対して要求記述表を作成できる。ここで、利用者からのイベントに対してシステムが応答することを 1 つのまとまりとして要求記述表を作成していることを注意しておく。

表 3 の要求仕様の内容として、仕様 1−1−2 の応答に基づいて、「会員がビデオを借りる[システムがビデオ貸し出し画面を表示する]」とした。

表 3 から、仕様上の不明点として以下の項目があることが分かる。

主体：システム環境？

対象：

・会員情報がある？

・会員番号？

・配送システムの状態？

主体状態：
・会員情報が識別されている？
・メニュ画面を表示している？
対象状態：
・会員情報？
イベント条件：
・メニュ画面の[ビデオ貸し出し]リンク押下についての回数？、未検出の扱い？、待ち時間の上限？
入力条件：
・会員情報？
応答制約：
・出力タイミング？
出力制約：
・画面表示制約？
・ビデオ本数の上限？

4.2 要求逸脱分析表

表 3 の結果から要求逸脱を抽出すると表 4 のようになる．同表では、主体と対象について示した。同様にして、イベント、入力、機能、出力、応答についても要求逸脱を抽出できる。この逸脱項目が異常系テスト項目の候補になる．この表の「逸脱/障害（重大性）」の項目で、障害の重大性を評価することにより、テスト項目の中から重要な項目を選択する理由を説明できる。

表 4　異常系テスト項目の候補例

パラメータ	ガイドワード	逸脱/障害(重大性)	異常系確認項目
主体（システム）	なし	起動不能/重大	システム起動確認
	以外	メニュ以外の画面が提示/表示異常	画面表示確認
	部分	メニュ画面の一部だけ提示/表示異常	画面表示確認
対象	なし	会員情報がない/重大	アクセス情報確認
	以外	他人の会員情報が提示される/重大	アクセス情報確認
	部分	ビデオ貸出画面の一部だけ提示/表示異常	画面表示確認
	冗長	余分な情報が提示/表示異常	画面表示確認

4.3 テスト十分性に関する保証ケース

要求逸脱分析に基づくテスト十分性を確認するための保証ケースを付図 1 に示した。この図では、付表 1 の要求仕様 1−1 について説明している。同様にして他の要求仕様についても分解して、テストの十分性を確認できる。

5 考察

以下では，上述したテスト十分性を確認するための保証ケース作成手順について考察する．

5.1　要求記述表

要求記述表を用いることにより、テストの対象とするシステムの要求仕様の曖昧な点を、主体、対象、イベント、入力、機能、出力、応答の観点から網羅的に抽出できる．これによって、要求仕様の逸脱分析の準備ができる．

5.2 要求逸脱分析表

要求逸脱分析表では、ガイドワードを用いて要求仕様の逸脱を網羅的に分析できる。これにより，従来のテキストでは触れられていなかった異常系テスト項目を系統的に抽出できることを明らかにした．一方、提案した手法に従うと，大量の逸脱項目を抽出することになり、重要なテスト項目を適切に選択するという課題がある。このため、逸脱分析表では、逸脱項目の重大性を識別するための項目を定義している。

5.3　保証ケースに基づくテスト十分性の確認

前節で示したように、本提案によって、自然言語で記述された要求仕様に基づいて、正常系と異常系のテスト項目を網羅的に抽出できること、ならびに、付図 1 に示した保証ケースからテスト項目の十分性について、ステークホルダ間で合意形成できることは明らかである。

本手法は要求仕様に基づいて段階的にテスト項目を作成できるだけでなく、抽出したテスト項目が必要な理由を明確に説明できる。したがって、選択したテスト項目が十分であるという説明を客観的に実施できる。

また，自然言語による要求仕様記述から段階的にテスト項目を抽出できるので、実際のプロジェクトに適応することは容易であると考えられる．したがって，今後，本手法を実際のプロジェクトに適用する研究を進める予定である．

5.4 有効性

本稿では，提案手法を例題の一部に適用しただけであり、有効性について定量的に評価していない．このため，本手法の適用について定量的な効果を明らかにしていく必要がある．

5.5 適用範囲

本稿で対象とした事例は教科書の例題であり、適用範囲についてはまだ明確とはいえない．したがって、本手法の適用範囲についても、今後，明らかにする必要がある．

6 まとめと今後の課題

本稿では，ソフトウェアテストの十分性を対象として保証ケースによる確認手法を構築する取組みについて紹介した．ただし，適用対象とした事例は 1 件だけであり，一般化するためには，他の事例についても評価する必要がある．

また，本稿の内容は手法の実行可能性を評価するための思考評価の段階にとどまっているため，定量評価までは至っていない．今後，実際のソフトウェアを対象とする評価実験を進める予定である．

謝辞

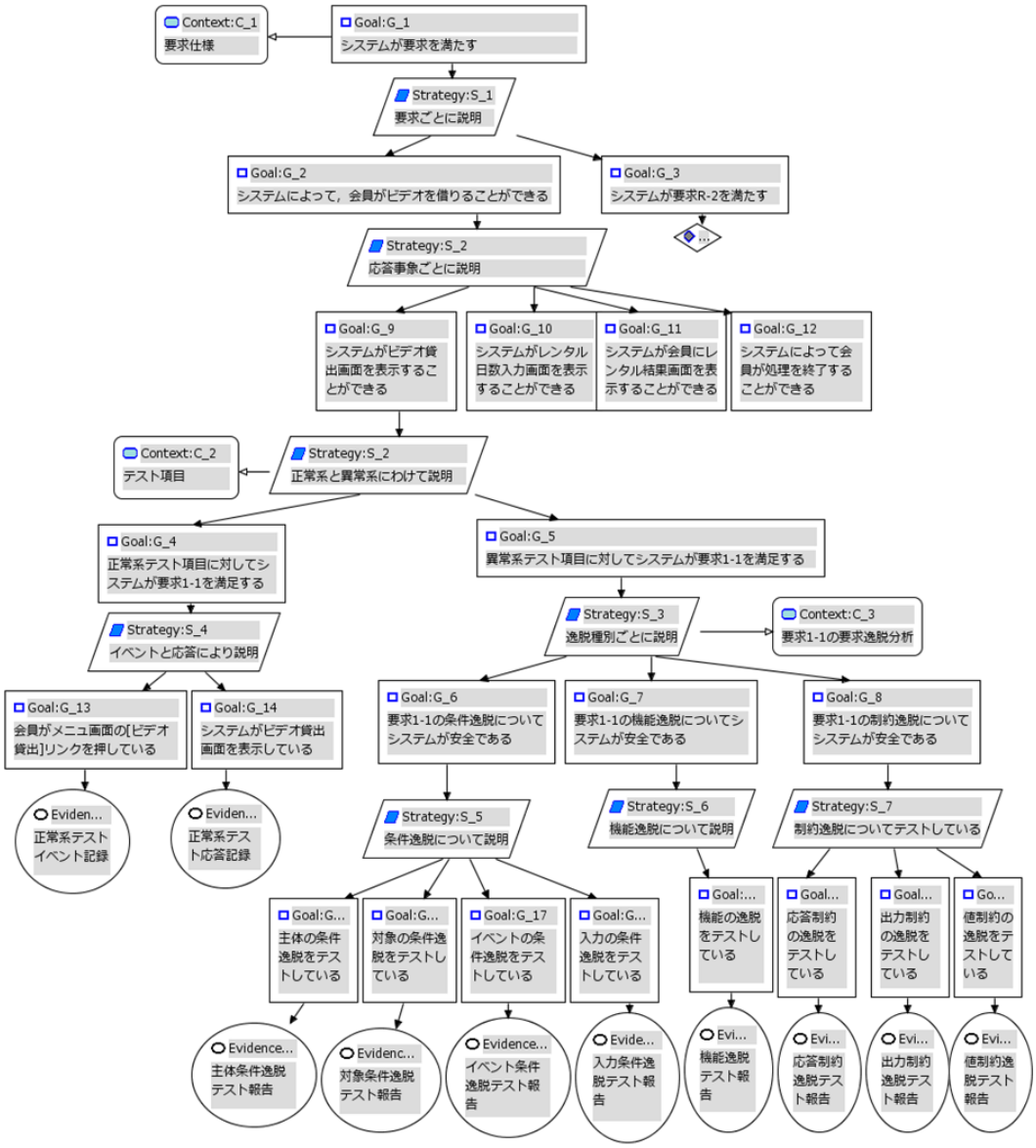
本研究は CREST「実用化を目指した組込みシステム用ディペンダブル・オペレーティングシステム」研究領域（DEOS プロジェクト）の支援を受けたものである[32][33][34].

参考文献

[1] Kelly, T. P, A Six-Step Method for the Development of Goal Structures, York Software Engineering, 1997
[2] T. Kelly. “Arguing Safety, a Systematic Approach to Managing Safety Cases”. PhD Thesis, Department of Computer Science, University of York, 1998
[3] J. A. McDermid. Software safety: where's the evidence? In SCS '01: Proceedings of the Sixth Australian workshop on Safety critical systems and software, pages 1-6, Darlinghurst, Australia, Australia, 2001. Australian Computer Society, Inc.
[4] I. Bate, T. Kelly, Architectural considerations in the certification of modular systems, Reliability Engineering and System Safety 81, pp.303–324,2003
[5] Tim Kelly and Rob Weaver, The Goal Structuring Notation – A Safety Argument Notation, Proceedings of the Dependable Systems and Networks 2004 Workshop on Assurance Cases, July 2004
[6] Despotou G., Kelly T., Extending the Concept of Safety Cases to Address Dependability. In proceedings of the 22nd International System Safety Conference (ISSC), Providence, RI USA, proceedings by System Safety Society 2004.
[7] Jackson, D. et al, Software for dependable systems–sufficient evidence?, NATIONAL RESEARCH COUNCIL, 2008
[8] 山本修一郎, 松野裕, ディペンダビリティケース作成法に関する一考察, KBSE研究会, IEICE-112, vol. IEICE-SS-164, No. IEICE-KBSE-165, pp.61-66, 2012
[9] 松野 裕, 高井利憲, ヴァイセ パテゥー, 山本修一郎, アシュアランスケース構築法の提案, KBSE 研究会, 2012
[10] 松野裕, 山本修一郎,ユースケース分析に基づくディペンダビリティケース作成法の提案, KBSE研究会, IEICE-112, vol. IEICE-KBSE-419, KBSE2012-61, pp.19-24
[11] Vaise Patu, Yutaka Matsuno, Shuichiro Yamamoto, Application of D-Case to the usage flow diagram scenario of the Distributed E-Learning System called KISSEL in Asian Pacific Universities, KBSE 研究会, 2012
[12] 高間翔太, 松野裕, 山本修一郎, スーパーコンピュータ運用手順に対するディペンダビリティの確認手法の提案, 信学技報, vol. 112, no. 165, KBSE2012-18, pp. 37-42 2012
[13] 高間翔太, 松野裕, 山本修一郎, ディペンダビリティ・コンテクストの推定手法の提案, KBSE 研究会, 信学技報, vol. 112, no. 314, KBSE2012-42, pp. 25-30, 2012
[14] 徳野達也, 松野裕, 山本修一郎, エンタープライズアーキテクチャ開発プロセスに対するディペンダビリティケース作成法の提案, 信学技報, vol. 112, no. 165, KBSE2012-36, pp. 145-150 2012
[15] 徳野達也, 松野裕, 山本修一郎, TOGAF NEXT に対する ADM プロセステンプレートの提案, KBSE 研究会, 信学技報, vol. 112, no. 314, KBSE2012-55, pp. 103-108, 2012
[16] 山本修一郎, 松野裕, ディペンダビリティケースへの責任属性導入法の検討, KBSE 研究会, 信学技報, vol. 112, no. 314, KBSE2012-52, pp. 85-90, 2012
[17] 松野裕, ヴァイセ バトゥ, 山本修一郎, アシュアランスケースへの構造化文書の適用に関する調査, 信学技報, vol. 112, no. 165, KBSE2012-20, pp. 49-54, 2012
[18] Vaise Patu, Yutaka Matsuno, Shuichiro Yamamoto, Research framework for dependability science based on assurance cases, IEICE Tech. Rep., vol. 112, no. 165, KBSE2012-21, pp. 55-59, July 2012
[19] 猿渡卓也, 松野裕, 星野隆, 山本修一郎, Modular GSNの定式化, KBSE研究会, 信学技報 vol.112, No.165, pp.151-156, 2012
[20] Shuichiro Yamamoto, Yutaka Matsuno, d* framework: Inter-Dependency Model for Dependability, DSN 2012
[21] Robin Bloomfield and Peter Bishop, Safety and Assurance Cases: Past, Present and Possible Future – an Adelard Perspective, 2010 [4] 山本修一郎, 松野裕, ディペンダビリティケース分解パターンについての考察, KBSE 研究会, 2013.3.15
[23] 山本修一郎, 松野裕, ディペンダビリティケース分解パターンについての考察, KBSE 研究会, 信学技報, vol. 112, no. 496, KBSE2012-80, pp. 67-72, 2013
[24] 松野裕, 高井利憲, 山本修一郎, D-Case 入門, ～ディペンダビリティ・ケースを書いてみよう！～, ダイテックホールディング, 2012, ISBN 978-4-86293-079-8
[25] 松野裕, 山本修一郎, 実践 D-Case,～ディペンダビリティ・ケースを活用しよう！～, ダイテックホールディング, 2013, ISBN 978-4-86293-091-0
[26] D-Case 実証評価研究会, http://***dcase***.jp/
[27] D-Case エディタ, http://www.dependable-os.net/tech/D-CaseEditor/
[28] 松野裕, 山本修一郎, アシュアランスケースツールへのプログラミング言語技術の適用, KBSE 研究会, 信学技報, vol. 112, no. 496, KBSE2012-81, pp. 73-78, 2013
[29] IEEE Std. 829-2008 Standard for Software and System Test Documentation
[30] ISO/IEC/IEEE 29148:2011 Systems and software engineering —Life cycle processes — Requirements engineering
[31] NTTデータソフトウェア工学推進センタ, 実例で学ぶソフトウェア開発, オーム社, 2008
[32] DEOSプロジェクト, http://www.crest-os.jst.go.jp
[33]DEOS プロジェクト, 2011 科学技術振興機構 White Paper DEOS-FY2011-WP-03J, *www.dependable-os.net/ja/topics/file/**White_Paper**_V3.0J.pdf*
[34] Mario Tokoro eds., Open Systems Dependability, Dependability Engineering for Ever-Changing Systems, CRC Press, 2012

付表 1　ビデオ貸出機能

仕様 1-1	会員がビデオを借りる	
	仕様 1-1-1	会員がメニュ画面の[ビデオ貸出]リンクを押したときに開始する
	仕様 1-1-2	システムが[ビデオ貸出]画面を表示する
	仕様 1-1-3	会員がレンタルしたいビデオを選択し、[次へ]ボタンを押下する
	仕様 1-1-4	システムが選択されたビデオの在庫の数を確認し、レンタル日数の入力画面を表示する
	仕様 1-1-5	会員がレンタル日数を入力し、[レンタル]ボタンを押下する
	仕様 1-1-6	システムが配送システムに会員情報を送り、ビデオの配送を依頼し、会員に[レンタル結果]画面を表示する
	仕様 1-1-7	会員が[レンタル結果]画面を確認し、処理を終了する



付図 1　ビデオ貸出システムのテスト十分性についての保証ケース