DEOS-FY2013-IS-01J

JST-CREST

研究領域

「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」

DEOS プロジェクト

DEOS 実用化のための オープンシステム・ディペンダビリティ 国際標準化戦略

2013/5/7

DEOS 研究開発センター編

科学技術振興機構
Japan Science and Technology Agency

DEOS 実用化のための オープンシステム・ディペンダビリティ 国際標準化戦略

2013/5/7

木下佳樹・武山誠（神奈川大学理学部情報科学科）

DEOS プロジェクト文書番号 DEOS-FY2013-IS-01J

(Research funded by JST CREST DEOS project)

概要

一つのシステムのステークホルダ間の努力だけでは、高いオープンシステム・ディペンダビリティを達成することは難しい。広く産業界が問題意識・目的、実現の指針、評価法、自動化・ツール化のための技術標準等を共有し協調する環境づくりは、DEOS 実用化の要のひとつである。これに向けて DEOS が国際標準化団体で進めている標準化の戦略、作業中の規格案、関連規格の動向、問題点等について説明し、協調にむけた議論を促す。[1]

オープンシステム・ディペンダビリティ(OSD)標準化の必要性

標準化は、オープンシステム・ディペンダビリティ実現の本質的な一要素です。各社がばらばらに OSD 導入をめざしても、標準がなければ困ってしまう局面をいくつか見てみましょう。

一番わかりやすいのは、ツールに互換性がなかったり、他システムとの情報交換のインターフェースがばらばらだったりして困るという問題です。

- 「納入されたコンポーネントのアシュランスケース[2]がうちのツールでは読めないぞ」
- 「接続先の A 社、B 社、C 社、…のサービス変更を自動検知したいけれども、相手先ごとにプログラムを組むのは大変すぎるなあ」

ここで必要になるのは、アシュランスケースの記述形式の規格や、OSD 情報交換用の標準 API です。これらは、DVD の記録方式の規格と同じ位置づけで、昔からあるような規格の使われ方です。

---

[1]本稿は、**ET2012 DEOS** スペシャルセッションにおける講演[**12**]の草稿に基づくものです。

[2] アシュランスケースは、システムが特定の要求を満たすという論点を一連の主張、証拠、その間を結ぶ議論によって監査可能な形で明示的に示す文書一式です。DEOS プロジェクトでは、これを拡張し電子化した「D-Case」を開発・運用の要となるデータとして用いますが、ここでは一般的な呼称として「アシュランスケース」を用います。

つぎに、大プロジェクトでの仕事の受発注に関連して、

- 「他社担当のシステムの障害や変更を、どこまで想定して頑張れば十分と認められるのか？うちだけがコストを負担するのは避けたいなあ」
- 「開発側のうちとしても運用側と一体的にやりたいけど、何しろ別会社でやり方が全然違うからなあ。変に介入されても困るなあ」

といった不安が OSD 導入の障壁になります。これは、標準がない中での、受注側と発注側の間の意思疎通の難しさともいえます。ここでは、仕事についての概念を規格化する必要があります。これは単に用語の規格化にとどまりません。用語が表しているプロセスや成果物についての概念も、受注側と発注側で一致させる必要があります。これについては、わが国は世界的に見ても先進的な「共通フレーム」が IPA で開発されています。ISO/IEC 12207 Systems and software engineering -- Software life cycle processes などは、共通フレームからアイディアを取り入れている状態です。このような働きをする規格で、オープンシステム・ディペンダビリティ実現に関連するものが必要です。もちろん共通フレームのなかにオープンシステム・ディペンダビリティの考えが取り入れられればそれでもいいわけです。

意思疎通の難しさは、トラブルが発生したときの責任問題にもつながります。開発者の側では、いろんな「まさか」が叫ばれます。

- 「まさかうちのシステムがこんな使い方をされるとは。これで障害をうちのせいにされても困るなあ。」
- 「まさかあそこのシステムはこんな細かいところまで指定通りに使わないとこんなにおかしくなるとは。これで障害をうちのせいにされても困るなあ。」
- 「まさかあそこのシステムがうちのシステムのこんな仕様外の動作に依存していたとは。うちのシステムを少し改善したつもりだったのにひどいことになってしまった。」
- 一方で、利用者側では
- 「こんなにひどい目にあったのに、どこの会社も“他社システムのせいではないか”とかはっきりしないことを言って責任をとってくれない。どうしてくれる。」

などと怒ったりします。利用者としては、開発者が想定していなかったような「まさか」が起こっても、何とか事態を解決する、あるいは解決してもらうことが必要です。しかし、開発者側では、あまりいろいろな問題を持ち込まれると、対応しきれなくなりますし、そもそもあらゆる事態を完全に解決することなど不可能です。

ここでは、利用者側の都合と開発者側の都合に矛盾がありますから、真っ向から解決しようとしても無理です。そこで、ひどいことが起こった場合に、関係者がお互いにどの程度妥協するか、について、リスクコミュニケーションを予めとっておくことが望ましいでしょう。そのために必要な手続きや手段についての規格を制定しておくことが、リスクコミュニケーションを円滑に進める為に

も望ましいと思われます。

アシュランスケースはまさに、リスクコミュニケーションの手段となる文書です。アシュランスケースに書かれるべきことを構文、内容の両面から規定することは、極めて有効だと考えられ、実際、後述する ISO/IEC 15026 Systems and software engineering -- Systems and software assurance や OMG SACM（Structured Assurance Case Metamodel）等は、アシュランスケースに関する、構文と内容の規格です。

ここでの規格の役割は、ツールの入出力様式を決める規格の役割とは随分異なります。入出力様式の規格は、まずツールの技術開発が行われ、それに目処が立ってから入出力様式を規格化する、と順番に行われるものです。伝統的な規格はそういう場合が多いのですが、アシュランスケース規格制定の場合、規格制定と技術確立が、絡まり合って進んでいくことになります。

リスクコミュニケーションの相場作りは、説明責任と技術上の秘密保持の兼ね合いを考える上でも必要です。

- 開発者は「どこまで障害発生の事情を説明したら被害者は理解してくれるのだろう。必要以上に他社に秘密をさらすのはいやだなあ」
- 利用者は「もしこんな障害が起きたらどうなるか聞きたいのに、”大丈夫です“とか”秘密です“とかしか言ってくれない。代わりのないシステムなのに全く信用できなくて不安だなあ」

とここでも両者の言い分は矛盾するのですから、お互いにどう妥協するかについてのコミュニケーションが必要です。

リスクコミュニケーションの品質の評価にも一定の基準が必要になります。

- 「うちで使った市販コンポーネントについてきたアシュランスケースは、全部うちの基準では OK だった。でも発注元は別の基準でダメだといってきた。買ってきたコンポーネントのアシュランスケースまで、相手に合わせてうちが書き直すのは大変すぎる」

さらに言えば、そもそもリスクコミュニケーションの相手がいるか、という問題もあります。

- 「もちろんうちのシステムにもあちらのシステムにも障害は起こりうるのだから、あらかじめ対処を話し合っておきたいのに、“そんなことにならないように、お互いちゃんとやりましょう”とか、“それは想定外ですから、その場で頑張るしかないでしょう”とかばかりで全然話が通じない。障害が起きてから裁判所に行くしかないのかなあ」

毎回、新しい考え方を説明するところから始めなければならないようでは、OSD を導入しようという気も失せるでしょう。OSD を規格として明文化することは、皆が「ああ、あれのことね」と了解して概念や目的意識を共有するためにも必要です。

標準がオープンシステム・ディペンダビリティ実現に果たす役割

このように、いろいろな局面で標準が必要になりますが、OSD の実現にあたって標準が果たす役

割は、普通の技術標準よりも能動的、積極的なものです。標準化は、OSD 実現の本質的な一要素で、OSD の技術開発と一体となって進められるものです。

通常の技術標準の場合、実験室なり研究所なりでまず技術が開発、実用化されて、そのあとで標準化作業が始まる、という順序になります。しかしこれは、研究所でしっかりと想定した条件のもとで技術を磨けば、十分皆の役にたつものになるはず、というクローズドな立場での順序であると言えます。開発者が、自分のシステムだけを見てディペンダビリティを向上させるための技術であれば、これも適当です。

しかし、オープンシステム・ディペンダビリティは、まさにこのようなクローズドな考え方では避けられてきた問題を扱うものです。例えば、多くの COTS つまり Commercial Off The Shelf コンポーネントを用いながら構成していくシステムインテグレーションを考えましょう。COTS の詳細は公開されずにブラックボックス化されざるを得ず、そのディペンダビリティをインテグレーターの努力によってコントロールするわけにはいきません。また、COTS は供給者の都合でアップグレードされたりして勝手に変わってしまうので、インテグレータは自分のシステムのディペンダビリティのコントロールにも苦労します。

クローズドな立場では、インテグレータは全ての COTS の詳細を公開させ、把握し、すべてをコントロールするための前提条件を整えよ、となるかもしれません。しかし、莫大なコストをかけてそんなことをするのであれば COTS を採用した意味がありません。

COTS は、一例にすぎません。詳細を把握しきれないという不完全さ、何がどう変わっていくか予測しがたいという不確実さ、関連システムを全てを管理する主体はいない、といった問題はネットワークを通じたサービスのやり取りでも、レガシーコードの再利用でも、環境や市場の変化でも、常に起こっていることです。

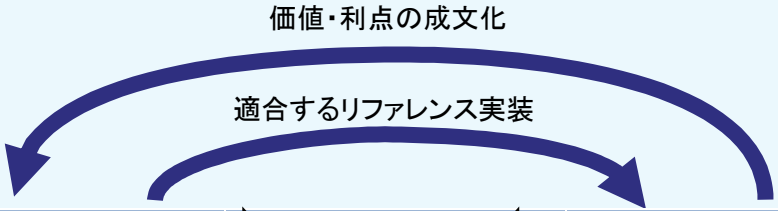
オープンシステムの見方からの解決法は、これら関連システムの動作の詳細を把握するかわりに、それらが一定のディペンダビリティに到達していることを要求することができるようにしよう、というものです。つまり、自分だけがディペンダビリティを実現しようとするだけでは、いろいろなシステムに繋がった体系のディペンダビリティの実現は不可能であり、皆が一緒にディペンダビリティを向上させていかなければならない、とするのがオープンシステム・ディペンダビリティです。

このことは、社会の安全に例えることができます。一人だけで行儀よくしていても、隣に乱暴な人がいれば、危険がやってきそうなわけで、安全が保障されるわけではありません。しかし、社会の人が皆行儀よくなれば、一人一人の安全も保障されることになるでしょう。これと同じで、ブラックボックスを持つシステムのディペンダビリティは、ブラックボックスを使う側の努力だけでは達成されません。ブラックボックスを作ったのは誰だか分からない場合もありますが、それも含めて、関連するシステムが皆、ディペンダビリティを達成している必要があります。

さて、自分のシステムだけでなく、関連システムがみなディペンダブルであることを要求しなければならないのはわかりましたが、それではつぎに、そのような状況を達成しよう、ということになります。その為に必要なのが、「ディペンダブルであるということはこういうことだよ」と、ディペンダビリティ達成の条件を明確に示すことです。しかし、そのような条件を記したものは、ディペンダビリティの要件を示す規格に他なりません。つまり、COTS やネットワーク化、レガシーコード利用など、ブラックボックス化が必然的におこるようなオープンシステムでは、ディペンダビリティ達成のための道具の一つに、ディペンダビリティ要件規格がある、というわけです。

本稿では、このような観点から、DEOS プロジェクトでの標準化戦略のお話をいたします。

標準化戦略：要件規格とツール規格



	規格	規定	内容	実現	DEOS成果
ディペンダビリティ要件規格	IEC NP Open Sys. Dep. IEC 60300-1		OSDを持つライフサイクル		DEOSプロセス
	IEC NP Open Sys. Dep. ISO/IEC 15026, IEC 62741		ライフサイクルの持つOSDのアシュランス		D-Case 手法
ツール規格	OMG Spec. SACM		電子化アシュランスケース		D-Case Editor
	OMG RFI MACL		機械的に検査可能なアシュランスケース記述言語		D-Case in Agda
			OSDを持つライフサイクル遂行の技術的支援		DEOS Arch. D-Script, D-ADD, D-RE, DS-Bench, …

規格化の目的：

・制定する規格への適合物が DEOS 成果を含むようにする

・ツールの入出力データの規格化

DEOS プロジェクトの成果は、大きく分けて二通りあります。DEOS プロセス、D-Case 記述などの、システムライフサイクルプロセスを新しく規定するものと、これらのプロセスを実現するための D-ADD, D-RE, D-Script, D-Case editor, D-Case in Agda などのツール群です。これら二通りについて、それぞれ標準規格を定めていこう、というのが DEOS の標準戦略です。

我々の標準化活動の場は第一に、ISO や IEC などのいわゆる de jure 規格を制定する団体です。ISO/IEC JTC1 SC7 Software Engineering や IEC TC56 Dependability でシステムライフサイクルプロセスのための要件規格の制定に携わっています。それだけではなく、OMG や The Open Group などの de facto 標準ないしは forum 標準を制定する団体でも活動しています。OMG SysA (System Assurance) TF や The Open Group においてツール規格制定の活動を行っています。

要件規格

まず、オープンシステム・ディペンダビリティ達成のための要件規格制定活動を紹介します。つまり、システムがオープンシステム・ディペンダビリティを達成する為に必要な条件を規定する規格です。

現在存在する要件規格として第一にあげるべきは、ISO/IEC 15026 Systems and software assurance です。この規格は、ISO、International Organization for Standardization と、IEC、International Electrotechnical Commission が共同で設置している JTC1, Joint Technical Committee 1, Information Technology の Subcommittee SC7 Software Engineering のそのまた下の Working Group WG7 Life Cycle Management において開発され、保守されています。米国と日本からエディタが派遣されて開発されました。

はじめに、system assurance という言葉について、誤解を解いておく必要がありそうです。assurance を保証と翻訳するのは、間違いとまでは言いませんが、適切ではないようです。というのは、逆に保証という日本語からは、英語の guarantee あるいは warranty というような語をまず思い浮かべてしまうのではないでしょうか。しかし system assurance という場合には、guarantee や warranty という言葉が持つ、うまく行かない場合に賠償するといった意味はなく、むしろ、自己の能力に対する自信、confidence あるいは確信、確実性といった意味が強いのです。保証というよりは、よく言われる安全・安心のうちの安心の部分を意味すると考えることもできます。うまく行かない場合の賠償、という意味は、少なくとも中心にはありません。

さて、ISO/IEC 15026 は四つのパートから構成されていますが、本年 9 月に第四部がでて、すべてのパートが出そろったばかりです。Part 1 は Concepts and Vocabulary という題で、2010 年に発行されました[3]。ありていにいって用語集です。また、これは International Standard ではなく、Technical Report として発行されており、国際規格の中では、正式の規格ではなく、いわば参考情報程度の扱いと考えればよいかと存じます。ただ、2010 年発行のこの Technical Report を 改訂（改正？）して International Standard にする作業が現在進行中です。特に支障がない限り、一、二年後には改訂されると思われます。

Part 2 は Assurance case という題で、アシュランスケースの形式と内容を定義しています[4]。自動車の機能安全の規格 ISO 26262 が、安全ケース safety case に言及しているのはご存知の方

も多いと思います。また、Common Criteria では Security Target 文書がシステムのセキュリティ認証の上で重要な役割を果たしていますが、このような文書はセキュリティケースと呼ばれています。このように、安全性やセキュリティなど、システムの重要な性質について、それが達成されていることを客観的な証憑をもとに議論する文書を、一般にアシュランスケースとよんでいます。安全ケースやセキュリティケースは、アシュランスケースの特別な場合と考えてよいでしょう。ISO/IEC 15026 Part 2 は、このアシュランスケースがどのような様式を持つべきか、またその内容がどうあるべきかについての規定です。

Part 3 は System Integrity Levels という題で、2011 年に発行されました[5]。ISO/IEC 15026 という規格は、初版が 1998 年に発行されました[2]が、そのときの題は System Integrity Levels という題で、一つのパートのみからなるものでした。機能安全に関連して IEC 61508 規格で定められている safety integrity level という言葉をご存知の方は多いと思います。また、common criteria で用いられている、セキュリティに関連する同様のものが evaluation assurance level (EAL)です。ちょうど、safety case や security case を一般化したものが assurance case であったように、safety integrity level や EAL を一般化したものを system integrity level と呼んでいます。実は ISO/IEC 15026:1998 は翻訳されて JIS になっており、そこでは、system integrity level はリスク抑制の完全性水準と和訳されています。英語にはリスクという言葉は出ていないのですが、これは内容をよく表した翻訳といえるでしょう。ISO/IEC 15026 Part 3 は、安全性やセキュリティだけでなく、信頼性、可用性、インテグリティといった、システムの様々な属性に対して、リスク抑制の完全性水準の物差しを決める方法を規定するものです。なお、この規格は 2011 年に発行されたものの、既に改訂の動きが出ており、2012 年 11 月現在、規格の保守を担当する ISO/IEC JTC1 SC7 WG7 には、ISO/IEC 15026 Part 3 の改訂の必要の有無を検討する Study Group が設置されて活動しています。

Part 4 は、システムやソフトウェアのライフサイクルの各プロセスにおいて、アシュランスを達成するために必要な事柄を規定するガイドラインで、assurance in the life cycle という題です。ISO には、システムやソフトウェアのライフサイクルを規定した規格があります。それぞれ ISO/IEC 15288 System Life Cycle Processes と ISO/IEC 12207 Software Life Cycle Processes で、どちらも、要求抽出からアーキテクチャデザイン、実現、運用、保守、廃棄の、せまい意味でのライフサイクルに限らず、調達や供給、プロジェクト管理などを含めた広い意味でのライフサイクルのプロセスがどのようなものであるかを規定しています。

ISO/IEC 15026 制定活動には、日本から執筆者、editor を派遣したおかげで、オープンシステム・ディペンダビリティの考えを企画の中に随分取り入れることができました。しかし、執筆者を派遣したのは、規格制定プロジェクトが始まって一、二年経ってからでしたので、オープンシステム・ディペンダビリティの考えを根本から反映させるには至りませんでした。

一方で、変化と曖昧さを許容するオープンシステム・ディペンダビリティの達成のための要件を規定する規格制定の New Work Item Proposal (NP)を現在、IEC TC56 日本委員会から国際委員会に提出しています。NP には、制定する規格の草稿もつけられています。そこでは、オープンシステム・ディペンダビリティの達成は、管理（マネージメント）と、明示されたアシュランスからなりたち、そのための四つの手段が合意形成、説明責任、故障応答、変化対応であるとされています。そして、これら四つの手段への要件があげられます。さらに、アシュランスを明示するアシュランスケース自体への要件が四つ挙げられています。それらは内部整合性、外部との整合性、主張の妥当性、アシュランスケース作成者の確信度の記述の四つです。

本年８月に提出したこの提案は、成立を認めるか否かについて、現在各国委員会からの投票が行われています。12 月にはその投票が終了し、承認されれば来年度には制定活動が開始されます。

ツール規格

つぎに、オープンシステム・ディペンダビリティ実現のためのツールに関する規格についてお話いたします。ここでツールというのは、いわゆるソフトウェア・ツールで、システム開発のいろいろなプロセスにおいて、道具として用いられることを想定したソフトウェアのことをさします。ツールに関する規格には、ツール自体の仕様に関する規格の他に、ツールへの入出力データに関する規格（様式等）も含みます。

D-Case editor が扱う D-Case を含む assurance case を格納するデータ記述形式を規定するものとして OMG SACM があります。アシュランスケースの記述ファイルをこの規格に適合するよう作成することによって、ツール間のデータ交換が容易になります。SACM は本年 5 月に制定されましたが、早くも改訂作業が始まっています。

OMG Object Management Group は、UML や COBRA の規格を提供しているので有名ですからご存知の方も多いと思いますが、米国をベースにする標準化のフォーラムの一つです。その Task Force TF のうちの一つ SysA System Assurance TF ではアシュランスケースを初めとするアシュランス技術に関する技術仕様を開発、規定しています。

もう一つのツール規格として、アシュランスケースの整合性についての機械的検査、つまり自動検査を可能にするようなアシュランスケース記述言語を作ろうという動きがあります。実際に、今回、DEOS の展示ブースで発表しておりますように、D-Case in Agda というツールでは、D-Case の検査を構文的なものだけでなく意味的なものまで含めて行うことを可能にしています。これは、Agda という函数型記述言語によって D-Case 記述を行うことによって可能になった技術ですが、Agda に限らず、もっと広範囲の記述言語で同様の検査ができることがわかっています。そこで、アシュランスケース記述言語で機械的検査が可能になるようなものを、その細部は規定せず、

本質的な抽象構文だけを規定していこうというのが、やはり OMG に提出されている MACL の提案です。現在 Request For Information RFI が OMG から提示されており、誰でも意見や情報を提供することができます。

ISO や IEC などの de jure 規格団体は、「ハウ」を規定することを嫌う傾向があるので、ツール規格の制定の場所には向きません。DEOS では、de jure 規格では要件規格などのような、概念的で上流におかれるべきものを提案し、ツール規格は OMG などのようなフォーラム規格で提案していく方針です。もちろん、フォーラム規格では OMG 以外のものもあり、現在例えば The Open Group への働きかけも行っています。

波及効果

これらオープンシステム・ディペンダビリティ関係の規格をはじめとする、アシュランス、ディペンダビリティ関連の規格はどこで用いられるでしょうか？

アシュランスケースが自動車産業で用いられつつあることは、既にご存知の方も多いでしょう。アシュランスケースの「安全」版である safety case が、ISO 26262 Road vehicles – Functional safety で、適合性検査への提出物の一つとして上げられているため、自動車業界ではにわかに safety case への興味が高まっています。

また、医療システムへの影響も無視できません。医療機器のみならず、病床での操作プロセスまで含んだ医療システムの安全性についての評価に、safety case を取り入れる動きが米国 FDA(Food and Drug Administration) で出ています。FDA は、assurance case の枠組みを医療機器の市販前届出制度 501(k)に取り入れる改革方針を 2010 年に打ち出しました。現に、輸液ポンプの認可でのパイロットスタディを実施中で、その結果を受けて規制を改定する計画を持っています。

ISO/IEC 15026 の米国政府における用いられ方について、米国政府関連団体の担当者に問い合わせたところ、DOD(Department of Defence), DHS(Department of Homeland Security), DOC(Department of Commerce) などで取り上げているようで、まもなくわが国にも波及するものと思われます。DOD の調達部門は ISO/IEC 15026 Part 4 を有益なものとしてとりいれつつあり、調達基準のために、必須ではありませんが、ISO/IEC 15026 Part 4 を用いてもよいことになっています。Department of Homeland Security のソフトウェアアシュランスグループも ISO/IEC 15026 利用を促進しており、あちこちの会議で言及しています。また、Department of Commerce、とくに NIST はサプライチェーンのセキュリティに関連して ISO/IEC 15026 に言及しており、NIST 800 シリーズの一つとなる新しい規格ができるものと思われるとのことです。

なお、旧版の ISO/IEC 15026:1998 は既に和訳されて JIS X0134:1999 として発行されていますが、今回改訂された四部構成の版を和訳して JIS にする作業が現在提案されています。

おわりに

オープンシステム・ディペンダビリティ関連の規格を、要件規格とツール規格の二つの面から制定する活動を行っています。要件規格は ISO/IEC JTC1 SC7 Software Engineering や IEC TC56 Dependability などでの de jure 規格として、またツール規格は OMG や The Open Group などのフォーラム規格ないし技術仕様として、制定する方針で進めています。

参考文献

[1] Mario Tokoro (ed.). Open Systems Dependability – Dependability Engineering for Ever-Changing Systems, CRC Press, ISDN 978-1-4665-7751-0, 2012.

[2] ISO/IEC 15026:1998 IS Information Technology – Software Engineering – System Integrity Level (superseded by [5]).

[3] ISO/IEC 15026-1:2010 TR Information Technology – Software Engineering – Systems and software assurance – Concepts and vocabulary.

[4] ISO/IEC 15026-2:2011 IS Information Technology – Software Engineering – Systems and software assurance – Assurance case.

[5] ISO/IEC 15026-3:2011 IS Information Technology – Software Engineering – Systems and software assurance – System Integrity Level.

[6] ISO/IEC 15026-4:2011 IS Information Technology – Software Engineering – Systems and software assurance – Assurance in the life cycle.

[7] ISO/IEC 12207:2008 IS Information Technology – Software Engineering – Software life cycle processes.

[8] ISO/IEC 15288:2008 IS Information Technology – Software Engineering – System life cycle processes.

[9] 情報処理振興機構ソフトウェアエンジニアリングセンター（編）．共通フレーム 2007 – 経営者、業務部門が参画するシステム開発および取り引きのために, SEC BOOK, オーム社, 第二版, ISBN-13 978-4274502477, 2009.

[10] OMG. Structured Assurance Case Metamodel (SACM), http://www.omg.org/spec/SACM/, 1.0 Beta 2, September 2012.

[11] OMG. Machine-checkable Assurance Case Language (MACL) RFI, document sysa/2012-09-04, http://www.omg.org/public_schedule/, September 2012.

[12] 木下佳樹、武山誠．DEOS 実用化のためのオープンシステム・ディペンダビリティ国際標準化戦略．Embedded Technology 2012 スペシャルセッション C-8、パシフィコ横浜、http://www.dependable-os.net/osddeos/event/201211/et2012.html、2012 年 11 月 16 日