人工知能学会
第 13 回知識流通ネットワーク研究会
SIG-KSN-013-06

高保証性エンタープライズアーキテクチャ標準 O-DA の概要と課題

徳野　達也†　　山本修一郎††

† 名古屋大学 大学院 情報科学研究科 〒 464-8601 愛知県名古屋市千種区不老町
†† 名古屋大学 情報連携統括本部 情報戦略室 〒 464-8601 愛知県名古屋市千種区不老町
E-mail: †tokuno.tatsuya@b.mbox.nagoya-u.ac.jp, ††yamamotosui@icts.nagoya-u.ac.jp

あらまし　2013 年 7 月のオープングループ会議で TOGAF の新たな標準 Open Dependability Through Assuredness(O-DA) が採択された．O-DA のガイドラインとして TOGAF ADM の工程ごとにアシュアランスケースによるアーキテクチャの高保証性を確認できる AADM(Assured ADM) が用意されている．本稿では，O-DA の概要と AADM を具体的に適用する上での課題について述べる．

Overview and issues of the Open Dependability Through Assuredness Framework

Tatsuya TOKUNO† and Shuichiro YAMAMOTO††

† Graduate School of Information Science Nagoya University
Furo-cho, Chikusa-ku, Nagoya, Aichi, 464-8601 Japan
†† Strategy Office, Information and Communications Headquarters Nagoya University
Furo-cho, Chikusa-ku, Nagoya, Aichi, 464-8601 Japan
E-mail: †tokuno.tatsuya@b.mbox.nagoya-u.ac.jp, ††yamamotosui@icts.nagoya-u.ac.jp

Abstract　Assured Architecture Development Method is provided as a guideline in O-DA to ensure the assuredness of the architecture development steps of the TOGAF ADM with assurance case. In this paper, we describe the overview of O-DA and the application issues of AADM.

1. はじめに

我々の日常生活におけるソフトウェアシステムへの依存性は，利便性，効率性，セキュリティのために日々増大している．ほとんどのシステムは，長時間使用され，また技術の進歩や規制や標準の変更によって常にサービスの目的やユーザの要求が更新されている．そのためこれらのシステムは非常に複雑なものとなっており，これらのソフトウェアのディペンダビリティは単にソフトウェアプロセスや形式手法のような従来の手法を使用するだけでは達成することが難しくなっている．実際に身近となったシステムに障害が発生し，大きな問題となった例も少なくない．このような大規模で複雑なシステムにおけるディペンダビリティを達成するために，高保証性 (Assuredness) という概念を導入した Open Dependability Through Assuredness(O-DA) が 2013 年 7 月に行われたオープングループ会議で The Open Group Architecture Framework(TOGAF) [1] の新たな標準として採択された．

O-DA に記載されている情報は，TOGAF ADM のような包括的なアーキテクチャ開発手法 (ADM) に利用可能である．Assured ADM(AADM) として高保証アーキテクチャを実装するために必要な追加手順について TOGAF ADM の各フェーズごとに述べられている．また近年，システムの安全性を確認する手法としてアシュアランスケースが注目されており，これをアーキテクチャを保証するための議論構造を記録する成果物として用いることで，アーキテクチャが高保証性を持つことを確認することが可能である．

そこで本稿では，O-DA の概要と，O-DA を TOGAF ADM に適用した AADM において高保証に必要な追加のステップを実行する際に発生する具体的な課題について述べる．またその課題のいくつかにおいて課題を解決する方法についても議論する．

以下に本論文の構成を述べる．2 章では，O-DA について説明する．3 章では，アシュアランスケースについて説明する．4 章では，TOGAF とそのアーキテクチャ開発手法 (ADM) について説明する．5 章では，高保証アーキテクチャ開発手法 (Assured ADM) について，また ADM と AADM の関係を説

表 1　O-DA の構成

章	内容	主な項目
1	はじめに	目的，概要，今後の方向性
2	定義	保証，高保証，アシュアランスケース
3	O-DA フレームワーク	非機能要求としてのディペンダビリティ，アシュアランスケース開発，説明責任，障害対応サイクル，変化適応サイクル
4	ガイドライン	アシュアランスケースの構造，証拠と説明責任，アシュアランスケース例，アシュアランスケースの強化
5	形式手法	証拠としての形式手法
付録 A	AADM	背景，TOGAF における高保証性フレームワーク，保証内容メタモデル，アーキテクチャリポジトリ
付録 B	DEOS	DEOS フレームワーク，DEOS サイクル，D-Case，合成ディペンダビリティ，合意形成と説明責任

明し，それらを対応付けて比較する．6 章では，高保証アーキテクチャを実装するために必要な追加ステップを実行する際に発生する課題について述べる．7 章では，いくつかの課題に対し課題を解決する方法について検討し，最後に今後の課題について述べる．

2. O-DA とは

2013 年 7 月，オープングループ会議で TOGAF の新たな標準として採択された．O-DA は JST-CREST の DEOS(Dependability Engineering for Open Systems) プロジェクト [2] の基本概念に基づいた高保証性を持つアーキテクチャを開発するためのフレームワークとガイドラインを定義している．フレームワークはアーキテクトに概念モデルを提供する．O-DA では，アーキテクチャの実装が保証すべき指定された要件を満たしていることを確信するために，満足な水準の証拠が提供されていることについてシステムのステークホルダが合意している状態を，高保証性があると定義されている．各ステークホルダはその合意に責任を持つ必要がある．

2.1 O-DA の構成

表 1 に示すように 1 章はじめに，2 章定義，3 章 O-DA フレームワーク，4 章ガイドライン，5 章形式手法，付録 A. AADM，付録 B. DEOS から構成されている．

3. アシュアランスケース

O-DA でもアーキテクチャが高保証性を持つことを確認するために使用されるアシュアランスケース [3] [4] は，主に欧米で普及しているシステムの安全性などを確認するために用いられる方法である．またディペンダビリティについて議論する場合は，ディペンダビリティケースとも呼ばれる．

アシュアランスケースの記法の一つに Goal Structuring Notation(GSN) [5] がある．これは Tim Kelly らによって提唱された要求を木構造に分解し，議論を容易にする表記方法である．議論すべき主張をトップゴールとし，ゴール分解の理由などが記述されるストラテジに基づいてトップゴールを複数のサブゴールに分解する．またコンテキストとしてゴールを議論する際の条件や前提などの情報が記述される，そして最下層のゴールにエビデンスとして証拠を与えることでそのゴールを保証している．このように抽象的なトップゴールを分解し，各サブゴールを保証することで，トップゴールを保証することが可能である．

本稿では DEOS プロジェクトの一環として開発された D-Case Editor [6] と呼ばれるアシュアランスケース作成を支援するツールを用いて，アシュアランスケースを GSN で記述する．

4. TOGAF

TOGAF(The Open Group Architecture Framework) は，オープングループのアーキテクチャフォーラムが開発してきたエンタープライズの経営意思に立脚した IT システム体系を作成するための手法およびツールであり，エンタープライズ・アーキテクチャの導入，作成，利用，維持を支援するための手法と支援ツールを提供するエンタープライズ・アーキテクチャを開発するためのフレームワークである．

TOGAF は，ベスト・プラクティスおよび既存アーキテクチャ資産の再利用可能なセットによって支えられた，反復型プロセス・モデルに基づいている．また TOGAF の重要な要素は手法であるということである．これはつまりアーキテクチャ開発手法 (ADM) によって経営ニーズに合致したエンタープライズ・アーキテクチャを策定することができるということである．

4.1 ADM

TOGAF の中核には前述したアーキテクチャ開発手法 (ADM) が存在する．ADM はアーキテクチャ開発のための，検証済みの反復可能なプロセスを提供する．ADM は，アーキテクチャ・フレームワークの確立，アーキテクチャ・コンテンツの開発，トランジションの実行，アーキテクチャの実現のガバナンスを含んでいる．これらのアクティビティは全て，継続的なアーキテクチャの定義と実現の反復サイクルの中で実行される．これにより，ビジネス・ゴールと機会に応じて，コントロールされた方法で，組織がエンタープライズを変革することが可能となる．

ADM には以下の複数のフェーズが存在する．

- 初期フェーズ
- フェーズ A：アーキテクチャ・ビジョン
- フェーズ B：ビジネス・アーキテクチャ
- フェーズ C：情報システム・アーキテクチャ
- フェーズ D：テクノロジ・アーキテクチャ
- フェーズ E：機会とソリューション
- フェーズ F：移行プランニング
- フェーズ G：実践ガバナンス
- フェーズ H：アーキテクチャ変更管理

- 要件管理

5. ADM と AADM

O-DA に記載されている情報は，TOGAF ADM のような包括的なアーキテクチャ開発手法 (ADM) に利用可能であり，O-DA の付録 A では TOGAF ADM に適用することで ADM を拡張した高保証アーキテクチャ開発手法 (Assured ADM) が説明されている．

現在の TOGAF ADM の記述だけでは，高保証アーキテクチャを開発するために，各フェーズごとに保証すべき要件を満たしていることを確信するための満足な水準の証拠が定義または作成されているか分からない．

そこで O-DA の付録 A では TOGAF を利用して高保証アーキテクチャを開発する際に影響を受けるステップごとに追加の記述が ADM の各フェーズで述べられており，それぞれの ADM のフェーズごとにアシュアランスケースを用いたアーキテクチャの段階的な保証方法が提示されている．

AADM に拡張する際に影響を受ける ADM のステップと，AADM で追加記述された各ステップで保証すべきものについて比較し，対応づけたものが表 4 となる．

6. 適用上の課題

O-DA を TOGAF ADM に適用した AADM において，高保証アーキテクチャを開発する際に影響を受けるステップの追加記述を実行する際に発生する具体的な課題について述べる．影響を受けるステップの追加記述は前節で述べた表 4 の通りである．フェーズごとにいくつか課題が存在するが，フェーズごとの主な課題を挙げたものが表 2 に記述してある．

7. 解決方法の提案

前節で挙げた課題のうち，本稿ではフェーズ F までについて解決法の提案を行う．

7.1 初期フェーズ

アシュアランスケースを用いて初期フェーズの課題を保証する方法について提案する．まず「ディペンダビリティ委員会が組織されている」と「競合するゴール間の優先順位と比率が決定されている」の 2 つのサブゴールに分けて議論し，それぞれを保証する．このとき問題となるのは後者のゴールの作成方法であるが，これまでに提案されているアシュアランスケースのパターンの一つである均衡分解パターンを利用することでアシュアランスケースが記述可能であると考えられる．

均衡分解パターンとは，複数の競合する特性を満足するアシュアランスケースを作成するためのパターンである．まずそれぞれのゴールが独立するか，あるいは競合するかの場合ごとに，システムがこれらのゴールを満たすという主張に分解する．たとえば，システムゴール P と Q が互いに独立であるか，依存するかについて見解が分かれる可能性があるとする．その場合図 1 のようなアシュアランスケースで均衡パターンを記述できる．

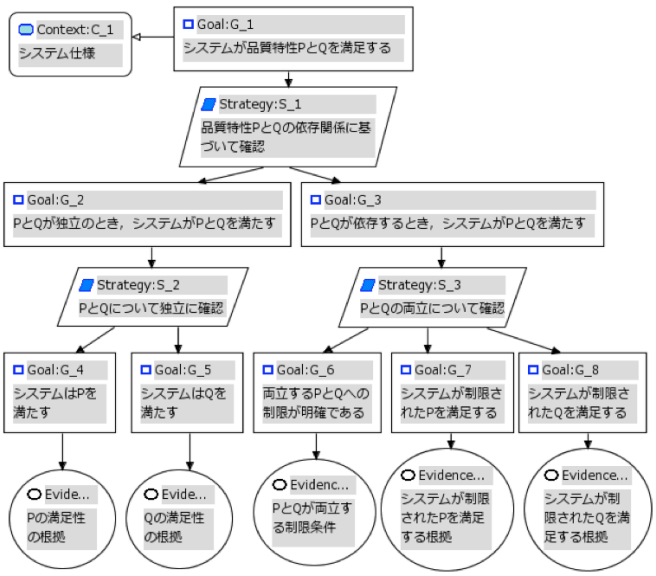

図 1　均衡分解パターン

7.2 フェーズ A

フェーズ A の課題について解決方法を提案する．ディペンダビリティはアーキテクチャビジョンに含まれる必要があり，ディペンダビリティ要素を各ステークホルダから収集し，明示的に文書化する必要がある．そこでディペンダビリティパラメータの候補を決定する方法として IPA が提供する非機能要求グレード [7] の非機能要求グレード活用シートを利用することが考えられる．

非機能要求グレードは，表 3 のように可用性，性能・拡張性，運用・保守性，移行性，セキュリティ，システム環境・エコロジーに分類される小項目からなり，また小項目は非機能要求を定量的に評価するためのメトリクスと，メトリクスに対して具体的な取りうる値を 6 段階で示している．

これを利用することにより，非機能要求であるディペンダビリティ要素の可視化，またメトリクスによってディペンダビリティ要素をどの程度満たすべきかの水準も決定することができ，異なるパラメータ間での優先順位の検討等にも利用可能であると考えられる．

7.3 フェーズ B, C, D

フェーズ B, C, D における各アーキテクチャに対するアシュアランスケース作成について，本稿では代表としてフェーズ B について提案する．

7.3.1 TOGAF 文書の出力を利用

TOGAF ADM にはフェーズごとにフェーズが完了した際の出力が記述されている．そこでフェーズ B の出力にあるアーキテクチャ定義文書の各要素を満たしているかをゴールとして，ステップが行われたかで保証することでアーキテクチャのアシュアランスケースが作成できると考えられる．フェーズ B に関係のあるアーキテクチャ定義文書の各要素は以下の 11 項目である．

- ベースラインアーキテクチャ
- ターゲットアーキテクチャ

表 2　フェーズごとの主な課題

フェーズ	主な課題
初期フェーズ	ディペンダビリティ委員会を組織し，競合するゴール間の優先順位と比率を決定していることをどのように保証するか
フェーズ A	何を用いてディペンダビリティパラメータの定義を行うか
フェーズ B, C, D	それぞれのアーキテクチャのアシュアランスケースをどのように作成するか
フェーズ E	B, C, D で作成したアシュアランスケースをどのように統合するか
フェーズ F	運用マネジメントのアシュアランスケースをどのように作成するか
フェーズ G	これまでに作成したアシュアランスケースの証拠をどのように作成するか
フェーズ H	アシュアランスケースを用いてどのようにリスク管理と障害分析を行うか
要件管理	どのようにアシュアランスケースの主張と要求の追跡管理を行うか

表 3　非機能要求グレードの記述項目

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，性能目標値オンライン，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	13
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，環境マネージメント	17

- 組織構造
- ビジネスゴールと目的
- ビジネス機能
- ビジネスサービス
- ビジネスプロセス
- ビジネスロール
- ビジネスデータモデル
- 組織と機能の相互関係
- ビュー

フェーズ C, D でも同様に出力にはアーキテクチャ定義文書が存在しており，文書で関係する要素が記述されている．そのため同様に適用可能であると考えられる．

7.3.2　コンテンツメタモデルを利用

TOGAF にはコンテンツメタモデルが存在する．図 2 のように共通またはアーキテクチャごとのエンティティが表現されている．そこでアーキテクチャごとに対応するエンティティが存在するかどうかでアシュアランスケースを作成することができるのではないかと言える．

フェーズ B のビジネスアーキテクチャに関わるエンティティは以下の 10 項目である．

- ドライバ
- ゴール
- 目標
- 評価指標
- 組織ユニット
- ロケーション
- アクター，ロール
- プロセス，イベント，コントロール，プロセス生成物
- ビジネスサービス，コントラクト，サービス品質
- 機能

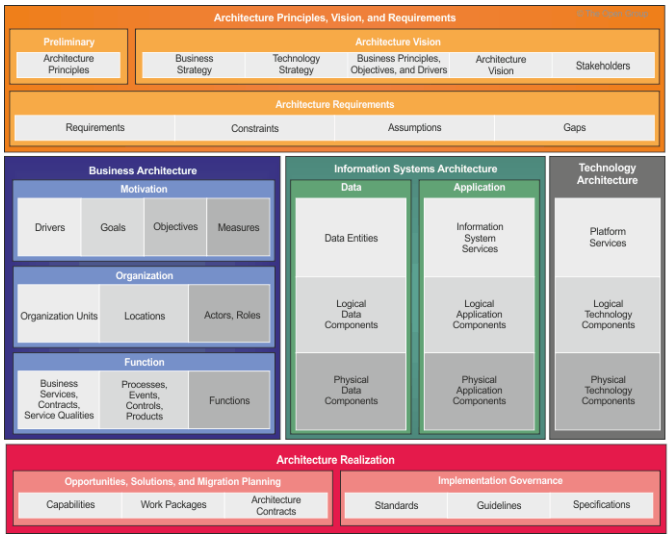

図 2　コンテンツメタモデルの表現

フェーズ C, D も同様にそれぞれのアーキテクチャに対するエンティティが存在しているので，同様に適用可能であると考えられる．

7.4　フェーズ E

フェーズ E の課題について解決方法を提案する．フェーズ E

ではフェーズ B, C, D で作成したアシュアランスケースを統合し，一貫性の確認を行う必要がある．

そこでフェーズ B, C, D で作成した 3 つのアシュアランスケースの主張および証拠の間で，対立があるかないかを判断する．なければそのまま統合し，ある場合は，それぞれのアシュアランスケースのやり取りが保証されているというサブゴールを作成し，そこで対立のある主張を移動させ，均衡分解パターンを利用することで対立を解消していくことが考えられる．

また図 3 のようにコンテンツメタモデルの要素と関係が図示されている．そのためフェーズ B, C, D でコンテンツメタモデルを利用してアシュアランスケースを作成した場合，これらのエンティティ間のやり取りを追加保証することで，アシュアランスケースを統合することができるのではないかと言える．

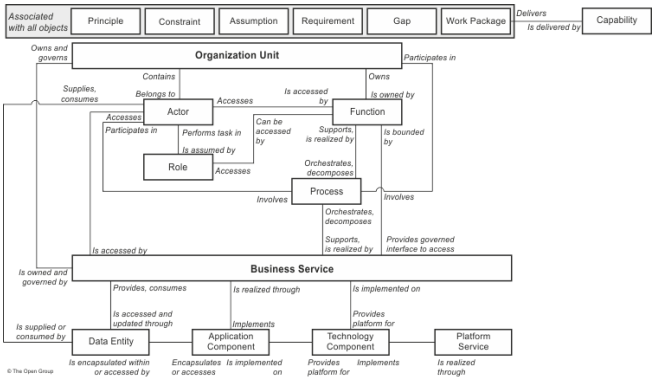

図 3　コアコンテンツメタモデルの要素と関係

7.5　フェーズ F

次にフェーズ F の課題について解決方法を提案する．運用手順に対するアシュアランスケース作成法 [8] については我々も研究を進めている．そのため我々が提案している運用手順に対するアシュアランスケースのパターンを利用することで，比較的容易に運用マネジメントに対するアシュアランスケース作成が行えるのではないかと考える．

8. おわりに

本稿では TOGAF の新しい標準として採択された O-DA の概要と TOGAF ADM を拡張した AADM(Assured ADM) について説明し，また ADM を拡張する上での問題点をフェーズごとに記述した．また挙げた問題点の代表的なものに対し，解決方法を検討し提案した．

今後の課題として，まず今回提案した解決方法が実際にうまくいくかさらに検討を行う必要がある．また解決方法を検討していない他の問題点の解決方法の提案を行う必要がある．

謝　　辞

本研究は CREST「実用化を目指した組み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．

文　　献

[1] TOGAF Version 9.1 http://www.opengroup.or.jp/togaf.html
[2] DEOS プロジェクト http://www.crest-os.jst.go.jp
[3] Peter Bishop, Robin Bloomfield, Sofia Guerra, The future of goal-based assurance cases, DSN, 2004
[4] T. Scott Ankrum, Alfred H. Kromholtz, Structured Assurance Cases: Three Common Standards, IEEE International Symposium on High Assurance Systems Engineering, 2005
[5] Tim Kelly and Rob Weaver. The goal structuring notation - a safety argument notation. In Proc. of the Dependable Systems and Networks 2004, Workshop on Assurance Cases, 2004.
[6] D-Case Editor http://www.dependable-os.net/tech/D-CaseEditor/
[7] 独立行政法人 情報処理推進機構 技術本部 ソフトウェア高信頼化センター. 非機能要求グレード研修教材.2010
[8] 高間翔太，山本修一郎，運用手順に対するアシュアランスケース作成法の比較評価，2013

表 4　TOGAF アーキテクチャ開発手法と高保証アーキテクチャ開発手法

フェーズ	ADM各フェーズにおけるステップ	AADMにおける追加ステップ
初期	6.4.1 Scope the Enterprise Organizations Impacted	アシュアランスケースと証拠がアーキテクチャリポジトリに格納されること
	6.4.2 Confirm Governance and Support Frameworks	ディペンダビリティ委員会で主張間の優先順位と比率が決定されていること
	6.4.3 Define and Establish Enterprise Architecture Team and Organization	
	6.4.4 Identify and Establish Architecture Principles	
	6.4.5 Tailor TOGAF and, if any, Other Selected Architecture Framework(s)	
	6.4.6 Implement Architecture Tools	
A.アーキテクチャビジョン	7.4.2 Identify Stakeholders, Concerns, and Business Requirements	ディペンダビリティスコープの定義
	7.4.3 Confirm and Elaborate Business Goals, Business Drivers, and Constraints	ビジネスゴールの実現のためのディペンダビリティゴールの定量的尺度の定義
	7.4.6.Define Scope	チームの能力，組織の評価
	7.4.7 Confirm and Elaborate Architecture Principles, includeing Business Principles	ディペンダビリティパラメータの定義
	7.4.4 Evaluate Business Capabilities	
	7.4.5 Assess Readiness for Business Transformation	
	7.4.8 Develop Architecture Vision	
	7.4.9 Define the Target Architecture Valuje Propositions and KPIs	
B.ビジネスアーキテクチャ	X.4.1 Select Reference Models, Viewpoints, and Tools	ディペンダビリティプリンシプルの決定
	X.4.2 Develop Baseline X Architecture Description	ビジネスアーキテクチャに対するアシュアランスケースの作成
	X.4.3 Develop Target X Architecture Description	アシュアランスケースのレビュー
	X.4.4 Perform Gap Analysis	
	X.4.7 Conduct Formal Stakeholder Review	
C.情報システムアーキテクチャ	X.4.1 Select Reference Models, Viewpoints, and Tools	情報システムアーキテクチャに対するアシュアランスケースの作成
	X.4.2 Develop Baseline X Architecture Description	アシュアランスケースのレビュー
	X.4.3 Develop Target X Architecture Description	
	X.4.4 Perform Gap Analysis	
	X.4.7 Conduct Formal Stakeholder Review	
D.テクノロジアーキテクチャ	X.4.1 Select Reference Models, Viewpoints, and Tools	テクノロジアーキテクチャのアシュアランスケースの作成
	X.4.2 Develop Baseline X Architecture Description	アシュアランスケースのレビュー
	X.4.3 Develop Target X Architecture Description	
	X.4.4 Perform Gap Analysis	
	X.4.7 Conduct Formal Stakeholder Review	
E.機会とソリューション	13.4.1 Determine/Confirm Key Corporate Change Attributes	B,C,Dそれぞれのアシュアランスケースを統合
	13.4.2 Determine Business Constraints fot Implementation	統合したアシュアランスケースの一貫性の確認
	13.4.7 Confirm Readness and Risk fot Business Transformation	
	13.4.3 Review and Consolidate Gap Analysis Results from Phases B to D	
	13.4.4 Review Consolidated Requirements Across Related Business Functions	
	13.4.9 Identify and Group Major Work Packages	
	13.4.10 Identify Transition Architectures	
	13.4.11 Create the architecture Roadmap & Implementation and Migration Plan	
F.移行プランニング	14.4.1 Confirm Management Framework Interactions for the Implementation and Migration Plan	運用管理のアシュアランスケースの作成
	14.4.2 Assign a Business Value to Each Work Package	アシュアランスケースの証拠を作成
G.実装ガバナンス	15.4.1 Confirm Scope and Priorities fot Deployment with Development Management	プロセスのアシュアランスケースの証拠を作成
	15.4.2 Identify Deployment Reasources and Skills	網羅的に主張と証拠の関係を確認
	15.4.3 Guide Development of Solutions Deployment	運用に対するアシュアランスケースのレビュー
	15.4.4 Perform Enterprise Architecture Compliance Reviews	
	15.4.5 Implement Business and IT Operations	
H.アーキテクチャ変更管理	16.4.2 Deploy Monitoring Tools	運用のアシュアランスケースの証拠の管理
	16.4.4 Provide Analysis for Architecture Change Management	ゴール不成立時の対処方法の確認
	16.4.3 Manage Risks	アシュアランスケースによるリスク管理
	16.4.5 Develop Change Requirements to Meet Performance targets	アシュアランスケースによる障害分析
要件管理		アシュアランスケースの主張と要求の追跡管理