

IT 検証技術者 教育研修シラバス

High-Level Syllabus (Level 5)

Ver. 1.0.0 (2011/07)

著作に関する注意点

本書の著作権は、IVIA(IT検証産業協会)が管理、所有します。本書の全部ないし一部を承諾なしに複製し利用することは、著作権法の制限事項に基づき、禁じられています。本書で利用している Test.SSF は、特定非営利活動法人 ソフトウェアテスト技術振興協会(ASTER:Association of Software Test Engineering:<http://aster.or.jp/>)とIT 検証産業協会(略称:IVIA:<http://www.ivia.or.jp/>)が、共同でテストエンジニアのスキル標準のフレームワークを定義したものです。

改定

バージョン	改定内容	改定日	備考
0.1.0	RC 版	2011/7/6	初版 (IVIA 会員内)
0.2.0	RC-1 版	2011/07/20	指摘事項の反映
1.0.0	リリース版	2011/7/28	

目次

著作に関する注意点	1
改定	2
目次	3
0. はじめに	4
■IT検証技術者教育研修シラバスについて.....	4
■IT 検証技術者認定試験について.....	5
■本書の作成にご協力頂いた方々	6
1. IT 検証産業協会 教育シラバスの体系	7
■テスト活動の構造.....	7
■Test.SSF とは	9
2. High-Level Syllabus(Level5).....	10

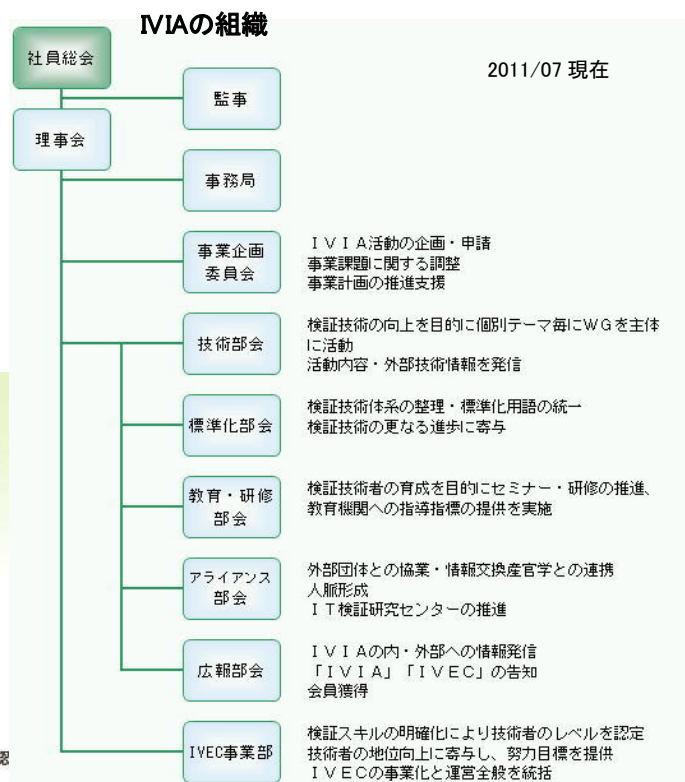
0. はじめに

■IT検証技術者教育研修シラバスについて

IT製品のソフトウェアは、製品の競争力を高めるため高機能／多機能でありながら高品質／高信頼性を維持することを求められています。一方では、これらの製品に対応できる専門性の高いテストを実施できる技術者の不足が、業界全体の問題となっています。

これらの課題に対応すべく 2005 年の 10 月に設立された「IT検証産業協会(以後、IVIAと表記する場合もあり)」(IT Verification Industry Association : 略称IVIA アイビア <http://www.ivia.or.jp/>)では、人材育成やIT検証技術者の地位向上を目指し、テスト技術の向上やテストの標準化の部会活動を行なっています。

IVIAの教育・研修部会では、IT検証技術者の育成支援に取り組んでいます。その中で本書は、技術者の育成に留まらず、教育機関への支援や技術提供も目的としています。高品質な製品の開発を支援する「IT検証技術者」の教育体系として、本書を広く公開して行きます。



■IT 検証技術者認定試験について

コンピュータは様々なものに形を変えて私たちの生活にはいりこんでおり、現代社会において必要不可欠なものとなっています。新製品や新システムに搭載されるソフトウェアの規模は増大しており、企業はその製品・システム開発に追われています。その一方、製品・システムの不具合等による事故や不具合が企業の存在を危うくする問題も頻発しており、品質管理は今や重要な経営問題であります。しかしながら、このような重要な問題にも関わらず、ソフトウェアテストに関する専門家が圧倒的に不足しているのが現状です。

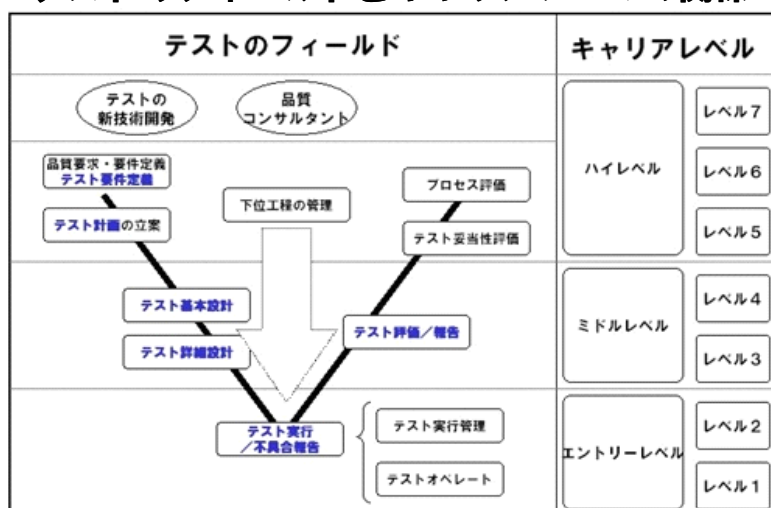
このような拡大している検証ニーズに対応していくためには、検証業務を安心して任せいただける専門家がが必要です。一方、「テストエンジニア」とひと口に言っても、企業や個人によって判断基準が曖昧で明確に定まっておらず、技術スキルを推し量る尺度が求められています。また、技術者が検証スキルを研鑽する際の目標を示すことも大切です。

MIAのテスト標準工程



このような考えのもと当協会ではソフトウェアテストに関わられている方を中心に、検証スキルの明確化、スキルアップ、認知度の向上を目指した認定試験を実施いたします。この認定試験を通じ、検証に必要なスキルを明確にし、だれもが同じ水準で判断できて安心して業務を任せいただける指標となり、かつ検証技術者の方々がご自身の技術スキルのアピールや今後のスキルアップの目標に活用いただけるようになることで、IT製品やシステムの品質管理に貢献できるプロフェッショナルなエンジニアの育成を支援し、IT検証サービス業界の健全な発展を目指してまいります。

テストのフィールドとキャリアレベルの関係



■本書の作成にご協力頂いた方々(敬称略)

株式会社エイネット	若林 幸治
テンプスタッフ・クロス株式会社	瀬間 武
東芝情報システム株式会社	
株式会社西本情報工学研究所	西本 安弘
日本ナレッジ株式会社	石川 俊一
株式会社富士通コンピュータテクノロジーズ	尾屋 正博
株式会社ヴェス	片川 貴博
株式会社ベリサーブ	佐々木 方規
株式会社ベリサーブ	檜村 紘久
株式会社ベリサーブ	冬川 健一
	大山 相達
IT検証産業協会 標準化部会	
IT検証産業協会 教育研修部会	

1. IT 検証産業協会 教育シラバスの体系

■テスト活動の構造

テスト活動が行われる領域は複雑で、開発担当者が作業の一部として実施するテストから、品質保証の立場として専任のテスト担当者が実施するテストまで広範囲です。テスト活動もテスト目的によって、実施するテストのタイミングや種類、状況、方法もさまざまに変化します。これらに対して IVIA では、IEEE829-2008^{注1}の規格で定義されているアクティビティをテスト活動の範囲としました。

テストエンジニアが活動する領域は、「図 1 テストフレームの構造」に示すように、開発軸と深く関係します。開発工程に対してテストは、IEEE829-2008 でテストレベルとしてコンポーネントテストから受入れテストまでが定義されています。

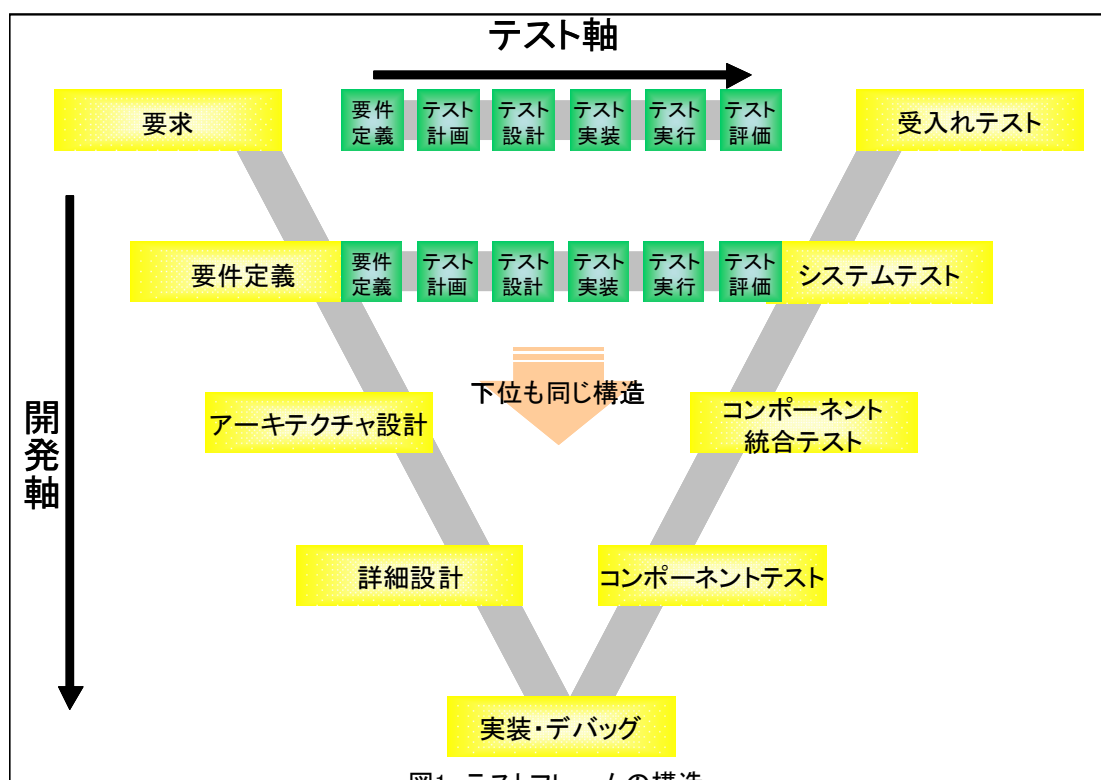
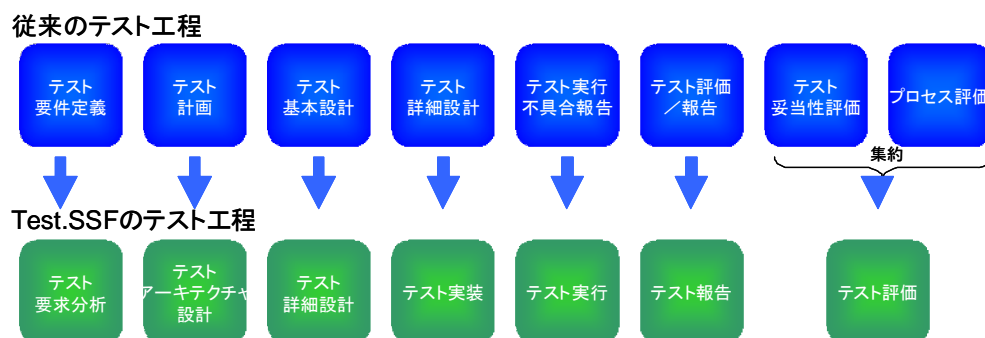


図1 テストフレームの構造

これらの関係は、Test.SSF(次ページ説明)で定義されています。本シラバスは、テストエンジニアのスキル標準フレームワークである Test.SSF を利用しています。Test.SSF を利用することで、Test.SSF を利用する他の団体との連携も図ることができると期待しています。

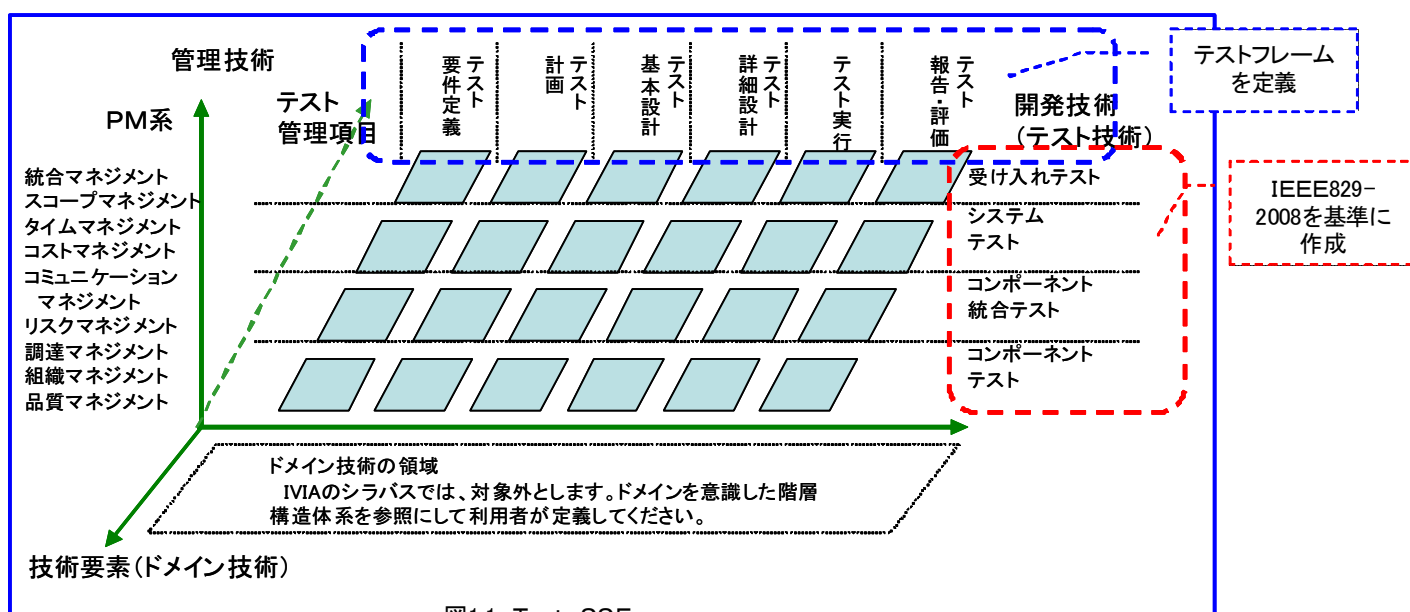
注1 IEEE829-2008 「IEEE Standard for Software and System Test Documentation」

Test.SSF を利用することにより従来のテストフィールド(p.5 テストのフィールドとキャリアレベルの関係を参照)と名称が異なります。したがって、本シラバスを参照する場合は、テスト工程の名称を読みかえる必要があります。従来のテスト工程との関係は下記の通りとなります。



■Test.SSF とは

Test.SSF は、ソフトウェアテストシンポジウム (JaSST) や JSTQB テスト技術者資格認定 (JSTQB) を運営する特定非営利活動法人ソフトウェアテスト技術振興協会 (ASTER: Association of Software Test Engineering: <http://aster.or.jp/>) と IT 検証技術者認定試験を運営する IT 検証産業協会が、IPA/SEC のアドバイスをを受け、共同でテストエンジニアのスキル標準のフレームワークを定義したものです。Test.SSF の特徴は、ETSS のフレームワークを利用しながら、スキルを組み込み領域に限定しないでテスト活動のスキル基準を定義したものです。また、Test.SSF のフレームは、ドメインによる構成することを可能^{注1}としています。



注1 Test.SSFおよびSSFとは

SSFとは
 正式名称
 SSFに基づくテスト技術スキルフレームワーク
 SSF: Skill Standard Framework / スキル標準フレームワーク

Test.SSFとは
 Test Technology Skill Framework based on SSF
 略称 Test.SSF

ドメインを意識した階層構造体系
 例) Automobile.Test.SSF
 mobile.Test.SSF 階層構造の体系化が可能

2. High-Level Syllabus(Level5)

第一階層	
第二階層	
第三階層	
<p>階層の概念モデルについて</p> <p>本シラバスは、Test.SSF の開発技術(この場合は、テスト技術)を用いています。(Test.SSF に関しては、IVIA および ASTER が公開している資料を参照して下さい。)</p> <p>第一階層は、IEEE829-2008 で定義されているレベルテストで構成されています。本シラバスを構成するテストの体系は IEEE829-2008 に定義されている通り、コンポーネントテスト/コンポーネント統合テスト/システムテスト/受入テストの4つのレベルテストを使用しています。</p> <p>第二階層は、テスト工程を示しています。このテスト工程は、全てのテストに対して同じ構造をもちます。したがって、各テストレベルおよび各テストレベルを構成するさまざまなテストに対して適用されます。これらは、テストが開発工程内で回帰構造をもつことを示しています。全てのテストの工程は、テスト要求分析/テストアーキテクチャ設計/テスト詳細設計/テスト実装/テスト実行/テスト報告/テスト評価の7つの作業フレームで構成することができます。</p> <p>第三階層は、テスト工程の活動(ACTIVITY)が定義されています。テストの実務を担当する者のそれぞれの活動に対して、TASK(処理)/INPUT(タスクへの入力)/OUTPUT(タスクからの出力)の形式で定義しています。</p> <p>Test.SSF は、テスト技術者のスキル標準を階層構造で定義しており、本シラバスも同じ構造で用います。</p> <p>本シラバスでは、同じ作業フレームで回帰的な構造をもつレベルテストをジェネリックなテスト活動として定義しています。したがって、第一階層ではレベルテストごとの表記をしないで、コンポーネント統合テストの階層化に集約して記載してあります。また、第三階層についても、各レベルテストの個別の特徴は記載されません。本シラバスを学習する場合は、対象となるレベルテストを想定して取り組んでください。</p> <p>各々のレベルテストの特徴でテスト活動に必要なものは、注記として記載するものとします。</p>	
1	<p>コンポーネントテスト</p> <p>コンポーネント統合テストの階層化にジェネリックなテスト活動として統合して記載してあります。</p>
2	<p>コンポーネント統合テスト</p>
1	<p>テスト要求分析</p> <p>テスト要求分析では、テストプロジェクトへの要求そのものの他に、テスト対象およびテスト対象外の範囲や制約事項などの情報を入手して分析し、テスト要件として定義します。テスト要求分析は、テスト活動において重要なプロセスとなります。ソフトウェアテストは、テスト活動単独では品質改善活動にならない特徴を持っています。テスト対象の品質を見極めるテストと品質を作りこむ開発作業が協調して、品質の改善を行うことができます。したがってテスト活動に対する要求を正しくとらえて実施する必要があります。また、テスト活動のゴール(終了基準やカットオーバークライテリア、リリース基準など)をテスト要求に基づいて定義されないと、延々とテストを続けることになり収束に向かわせることが困難になります。</p>

1		テスト要求分析		
1		テスト要求分析の準備をする。		
		<p>テスト要求分析には、開発の設計書などのドキュメントベースだけでなく、インタビューなどから要求を引き出します。インタビューなどは、テスト要求に対するステークホルダを調査しアサインします。(例: ユーザ要求の場合は利用者、保守要求などは管理者等) また準備フェーズでは、入手すべき情報の整理をすることと入手の手段および入手の経路が正当であることを示しておく必要があります。</p>		
		INPUT	TASK(PROCESS)	OUTPUT
		マスターテスト計画 レベルテストの依頼(テストレベル要求)	テスト要求に必要な入力を整理します。テスト要求は、テストレベル毎に必要な情報が異なります。マスターテスト計画から、該当するテストレベルを確認して、取得すべき情報を整理します。	テスト要求への取得概要(一覧)
		テスト要求への取得概要(一覧) ドキュメントベースのテスト要求	整理したテスト要求に取得すべき概要から、適切な(開発設計書などの)ドキュメントベースを入手する。入手したドキュメントの概要を作成し、各々のバージョンと入手(作成または発行)元を記録しておきます。	(入手するドキュメントの例) マスターテスト設計書 開発設計書 要求仕様書 過去テスト計画書/報告書
		テスト要求への取得概要(一覧) インタビューベースのテスト要求	テスト要求に関係するステークホルダ(要求元、利用者、開発者 等)に対してインタビュー/ヒアリングのアサインを実施します。	インタビュー報告書 議事録等
		テスト要求への取得概要(一覧)	開発中や要件の未決定など、テスト要求への入力となる必要なドキュメントやヒアリング結果が得られない場合は、不足情報として一覧化し、今後の入手の可否や時期、誰が担当するかを設定し、管理しておきます。	不足情報の管理表

1	テスト要求分析			
2	テスト要求を獲得する。			
		<p>テスト要求の獲得では、準備で入手したドキュメントやインタビュー／ヒアリング情報からテスト要求にインプットする情報を抽出します。テスト要求を抽出するとともに、入手した情報が要求なのか阻害要因なのか、さまざまな分類と関連付けを実施しておきます。また、テスト要求の獲得では、一時的に情報を収集するのではなく、収集した情報を整備して妥当な情報であることを要求元と確認を行うことが重要です。テスト要求へのインプットとなる情報は、テストレベルによって異なります。</p>		
		INPUT	TASK(PROCESS)	OUTPUT
		品質計画 マネジメントシステム (品質／環境／セキュリティ 等)	品質目標の基準となるテスト対象を展開する企業の品質方針／コンセプトを確認します。	品質方針 品質特性 テスト対象のコンセプト
		マスターテスト計画	マスターテスト計画から、担当するテストレベルの範囲とテスト目的を獲得します。テストレベルから、大まかなテストの範囲を決定します。	テスト目的 テストレベルの対象範囲
		開発仕様書 要件定義書 アーキテクチャ設計書	開発の要求仕様書または要件定義書より、テスト対象に求められる品質要求を想定します。	テストの目的 品質要求
		開発仕様書 要件定義書 コンセプト一覧	開発の要求仕様書または要件定義書より、テスト対象のコンセプトを抽出します。また、要求されている品質を確認し、品質目標を定義します。	品質目標 関連するメトリクス
		開発仕様書 アーキテクチャ設計書 その他、テストベースになる情報 (打ち合わせ議事録など)	テストベースになる情報を入手し、テストベースの一覧としてまとめます。	テストベース一覧
		マスターテスト計画書 テスト目的 テストレベルの対象範囲 プロジェクト計画書	マスターテスト計画書および獲得したテスト目的やテストレベルの対象範囲から、テスト対象範囲やコストやスケジュールに対する要求を獲得します。	テスト対象／テスト対象外の範囲の一覧 概算コスト 大まかなスケジュール
		プロジェクト計画書 開発の進捗情報 開発の組織情報	プロジェクト計画書から現在の開発の進捗状況や組織の情報を収集します。開発の進捗情報から、テストベースとなる入手できる開発成果物の状況やインタビューやヒアリングなどの窓口情報も獲得しておきます。	開発進捗状況レポート 開発の組織図(OBS)などの情報

1		テスト要求分析		
2		テスト要求を獲得する。(続き)		
		INPUT	TASK(PROCESS)	OUTPUT
		マスターテスト計画 プロジェクト計画 テスト対象の利用環境に関する情報	テストを実現するためのテスト環境情報を入手します。テスト環境情報には、テストベンチや実環境でのテスト要求(フィールドテスト)の有無などを獲得します。 テスト対象の利用環境に関する情報から、準備すべきテストベンチの情報などを収集しておきます。	テストベンチの情報 テスト環境に関する要求
		テスト対象の利用環境に関する情報	テスト対象が法令や規格、標準に対しての要求が存在するか、コンプライアンス情報を収集します。また、オープンソースなどの利用有無などの情報も収集しておきます。	コンプライアンスに関する情報
3		テスト要求を分析する。		
		獲得したテスト要求の情報を分析し、必要な関連付けを行い、さらにテスト要求を体系的に整理します。分析・整理されたテスト要求から、テスト目的(スコープ)を定義します。 テスト要求に対する制約事項の有無などを整理しておきます。		
		INPUT	TASK(PROCESS)	OUTPUT
		品質方針 品質特性 テスト対象のコンセプト	プロダクトおよびプロジェクトの品質方針を確認します。品質方針により、企業理念やプロジェクトを統括する組織の品質計画が定義されている場合は、品質ポリシーとして設定します。 次にテスト対象の品質特性を分析します。テスト対象の品質特性により、テストで実施すべき項目が変わります。 テスト目的を品質特性やテスト対象のコンセプトから詳細化・構造化を行い、精度を高めます。	テスト目的の詳細項目と、それらの構造を可視化したもの
		品質要求 品質目標	テスト目標を品質要求や品質目標から詳細化・構造化を行い、精度を高めます。	テスト目標の詳細項目と、それらの構造を可視化したもの
		テストベース一覧 品質特性 要件定義書 マスターテスト計画書	テストベース一覧と品質特性、要件定義書、マスターテスト計画書との整合性を確認し、過不足を明確にします。不足していると判断した場合は、そのドキュメントがあるかを関係部署に問い合わせを行います。	整合性確認済みのテストベース一覧

1		テスト要求分析		
3		テスト要求を分析する。(続き)		
		INPUT	TASK (PROCESS)	OUTPUT
		テストレベルの対象範囲 要件定義書 品質特性	テストレベルの対象範囲を基に、要件定義書や品質特性から範囲の深さを策定し、分析します。	テスト範囲の分析結果レポート
		マスターテスト計画書 開発進捗状況レポート	マスターテスト計画書を基に、開発進捗状況レポートとの差異を抽出し、テスト計画に影響がある箇所を分析します。	テスト対象が持つ問題・課題一覧
		マスターテスト計画書	マスターテスト計画書を基に、テストプロジェクトへの要求との整合性を抽出し、テストの目的や目標を達成しているか分析します。	テストの目的や目標との整合性一覧
		コンプライアンスに関する情報	コンプライアンスに関する情報を基に、テスト対象のコンプライアンスに対する影響を分析します。	コンプライアンスの対象の有無
		テスト環境に関する要求 テストベンチ情報	テスト環境に関する要求とテストベンチの情報を分析、整理します。	テスト環境要求一覧
4		テスト要求分析成果物を作成する。		
		テスト要求分析結果を整理して、テスト要求分析成果物を作成します。テスト要求へのインプットの項目とテスト要求結果を可視化し関係付けを行なうことで、テスト要求のカバレッジとトレーサビリティを確保します。		
		INPUT	TASK (PROCESS)	OUTPUT
		(テスト要求の分析結果)	下記項目を網羅する成果物の作成を行います。 テスト目的 品質目標 テストベース テスト対象範囲 テスト対象外範囲 テスト対象の開発状況 テスト環境(テストベンチ)	テスト要求分析成果物

1			テスト要求分析		
5			テスト要求分析成果物を検証する。		
			テスト要求分析成果物を要求元と公式レビューを行ない、内容の過不足や整合性の確認を行います。		
			INPUT	TASK(PROCESS)	OUTPUT
			テスト要求分析成果物	テスト要求分析成果物を要求元および関係者と公式レビューを行ない、要求に漏れがないことや要求との整合性の確認を行います。 指摘事項があれば、レビュー記録に基づいてレビューのフォローアップを行ないます。 全てのレビューフォローアップが完了し、要求元の承認を得ます。	レビュー記録 レビューフォローアップ 承認されたテスト要求分析成果物
2			テストアーキテクチャ設計		
			テストアーキテクチャ設計では、テスト要求分析の結果からテストの実現方式の立案を行います。テストを実現するためには、テスト対象の品質特性や環境を考慮したテストツール、テスト技法やテストマネジメント方法、テストベンチ(環境)などの選定や定義を行います。		
1			テスト要求分析成果物を準備する。		
			テストアーキテクチャ設計の入力に必要なテスト要求分析の成果物を用意します。テスト活動への要求の漏れを防止するために、テスト要求分析の成果物の一覧で準備する成果物を確認します。		
			INPUT	TASK(PROCESS)	OUTPUT
			テスト要求分析の成果物一覧 (OUTPUT の一覧を含む)	一覧をもとに実物を確認する。不足している場合に、入手時期を明らかにします。	テスト要求分析成果物入手済一覧表 テスト要求分析成果物
2			テストベースを準備する。		
			テスト対象の開発成果物などテストベースを準備します。入手したテストベースは、入手元や履歴の記録や管理方法を定義して管理します。		
			INPUT	TASK(PROCESS)	OUTPUT
			テストベース一覧	必要なテストベースを入手します。入手したテストベースは、履歴／管理場所／管理方法／管理者などを定めて管理を行います。 テストベースは、開発の状況によって改変される可能性があることを考慮しておきます。	テストベース テストベースの管理

2		テストアーキテクチャ設計		
3		アーキテクチャスタイルに関する要求を獲得する。		
		アーキテクチャスタイルを抽出するために、テスト要求を獲得します。獲得したテスト要求から、選択できるアーキテクチャスタイルを抽出しておきます。アーキテクチャスタイルはテスト要求だけでなく、テストプロジェクトの習熟度などにより影響を受けます。		
		INPUT	TASK(PROCESS)	OUTPUT
		テスト要求分析成果物 テストベース	テスト要求分析の成果物から、テストポリシーを定義します。 (アーキテクチャスタイル検討の材料を抽出)	テストポリシー
		アーキテクチャスタイル テストプロジェクトの習熟度	各種アーキテクチャスタイルから、適切なものを抽出します。また、アーキテクチャスタイルで包括されるテストの種類を一覧にしておきます。	抽出したアーキテクチャスタイル テストの種類の一覧
		過去のテスト実績情報	前バージョンのテスト計画書から、前バージョンで実施したテスト種類を一覧にしておきます。	過去に実施したテスト種類の一覧
4		アーキテクチャスタイルの要求を分析する。		
		抽出したアーキテクチャスタイルへの要求を分析します。アーキテクチャスタイルは、基本的にレベルテストやテスト目的に合わせて選択しますが、テストプロジェクトが得意とするスタイルや習熟度によるアプローチの変更なども考慮します。		
		INPUT	TASK(PROCESS)	OUTPUT
		テスト要求分析成果物 テストポリシー テストベース 抽出したアーキテクチャスタイル 過去に実施したテスト種類の一覧	テスト要求分析結果に沿うテスト種類をリストアップします。 次に、テストアーキテクチャスタイルごとのリスク・コスト・メリット等をリストアップする アーキテクチャスタイルの要求から次の分析を行います。 機能特性／非機能特性のテスト観点の理解 テスト対象の品質に状態に合わせて、テストエリアやテストの深度の定義 テスト対象の特性や開発状況を考慮して、テストを実施する優先順位を設定 過去の実績や、開発量／コード変更量および難易度などを考慮して、不具合発生率の予測 テスト終了時の条件として、不具合の改修条件を設定	(アーキテクチャスタイルの要求分析結果) テストの種類一覧 テストアーキテクチャごとのメリット・デメリット テスト手法の選択 テストサイクル テストの優先度 不具合発生率予想 不具合改修率(不具合改修残存許容数) テスト対象機能 範囲内／範囲外設定

2		テストアーキテクチャ設計		
4		アーキテクチャスタイルの要求を分析する。		
		INPUT	TASK (PROCESS)	OUTPUT
		アーキテクチャスタイル要求分析の比較検討結果成果物	アーキテクチャスタイル要求分析から考えられるプロダクトリスクを抽出します	プロダクトリスク一覧表
5		アーキテクチャスタイルを選択する。		
		アーキテクチャスタイルの要求を分析結果に基づいて、アーキテクチャスタイルを選択します。		
		INPUT	TASK (PROCESS)	OUTPUT
		(アーキテクチャスタイルの要求分析結果) プロダクトリスク一覧表	QCDDMM(Q:品質、C:コスト、D:納期、M:マネジメント、M:マーケット(この場合は、ターゲット))の分析を行い、アーキテクチャスタイルを選択します。	以下の選択結果成果物(選択一覧、 選択理由、リスク) アーキテクチャスタイル選択要素 テスト範囲 テスト手法 テストサイクル テストスケジュール QCD 分析結果
6		テスト全体の構造を設計する。		
		選択したアーキテクチャスタイルに基づいて、各種テストを配置していきます。アーキテクチャスタイルによって実施されるテストの種類は一種類とは限らず、適切なテストをテスト要求に基づいて構成していきます。		
		INPUT	TASK (PROCESS)	OUTPUT
		アーキテクチャスタイル要求分析の選択結果	テストアーキテクチャスタイルに合わせて、テスト対象のテストすべき要素(機能、ユースケース等)を配置します。	テストアーキテクチャスタイル別に分類した以下の要素 ・機能ブロック ・ユースケース 等
		テストアーキテクチャスタイル別の分類要素(1のアウトプット)	テストアーキテクチャ別に共通項、固有項目を分類します	テストアーキテクチャ毎の ・共通要素グループ ・固有要素グループ
		テストアーキテクチャ毎の ・共通要素グループ ・固有要素グループ	テストアーキテクチャ毎のグループ間のテストの関連性を検討し、設計します。	テストアーキテクチャグループ間のテスト関連表

2		テストアーキテクチャ設計		
7		テスト全体のバランスを調整する。		
		INPUT	TASK (PROCESS)	OUTPUT
		<ul style="list-style-type: none"> ・テストベース ・適用可能なアーキテクチャスタイル素材 ・過去に実施したテスト種類の一覧 ・テスト対象の特性要因 	選択されなかったテストアーキテクチャの確認をします。 選択したテストアーキテクチャスタイルの実現可能性確認します。	<ul style="list-style-type: none"> ・非採用のテストアーキテクチャの影響 ・テストアーキテクチャの実現可能性
		<ul style="list-style-type: none"> ・非採用のテストアーキテクチャの影響 ・テストアーキテクチャの実現可能性 	テストアーキテクチャの選択漏れ、または選択したテストアーキテクチャの実現に矛盾がないことを検証・レビューします。	テストアーキテクチャレビュー結果
		テストアーキテクチャグループ間のテスト関連表	テストアーキテクチャスタイルに基づきテスト詳細設計の実施概要を検討します。	テスト詳細設計概要
8		テスト環境の構築方針・方法を検討する。		
		INPUT	TASK (PROCESS)	OUTPUT
		<ul style="list-style-type: none"> ・テストベース ・テストアーキテクチャグループ間のテスト関連表 	テストベースやアーキテクチャスタイル、およびテスト手法等から、テスト環境の構築方針を検討します	テストアーキテクチャグループ別のテスト環境の構築方針
		<ul style="list-style-type: none"> ・テストベース ・テストアーキテクチャグループ別のテスト環境の構築方針 	テスト環境構築方針に基づき、テスト環境の構築方法を検討します	テストアーキテクチャグループ別のテスト環境の構築方法
		<ul style="list-style-type: none"> ・テストアーキテクチャグループ別のテスト環境の構築方針 ・テストアーキテクチャグループ別のテスト環境の構築方法 	テスト環境構築方針、構築方法から、必要とされるファシリティを検討します	テスト環境ファシリティ構成

2		テストアーキテクチャ設計		
9		テスト詳細設計の指針・原則を検討する。		
		INPUT	TASK (PROCESS)	OUTPUT
		テストベース テスト要求分析による品質目標	テスト対象物の品質目標を設定します(テスト実施による不備指摘数)。 テスト仕様書、テストデータの品質目標を設定します(レビューによる不備指摘数)。	品質目標の定義
		プロジェクトが用意したものおよび市販のテスト技法が記載されているドキュメント	テスト詳細設計で使用するテスト技法を選択します。	テスト手法(技法)の定義
10		テストアーキテクチャ設計成果物を作成する。		
テストアーキテクチャ設計結果を整理して、テストアーキテクチャ設計成果物を作成します。				
		INPUT	TASK (PROCESS)	OUTPUT
		※上記5項,7項のアウトプットすべて	上記の工程で検討し設定した内容を用いて、テスト計画書を作成する	以下の項目が含まれたテスト計画書 テスト範囲の定義 自動化・再利用化の定義 テスト目的(目標値を含むテストスコープ) テストスケジュール テストタスク テスト要員体制 テスト範囲(テスト実施範囲と対象テスト工程) リスク一覧表 調達基準と技術要素トランスファー計画
		※上記5項,8項のアウトプットすべて	上記の工程で検討し設定した内容を用いて、テスト環境関連をテスト計画書に記載する	以下の項目が含まれたテスト計画書 テスト環境定義(必要機材、リソース、作業場所)
		※上記9項のアウトプット	上記9項で検討し設定した内容を用いて、テスト詳細設計の指針・原則をテスト計画書に記載する	以下の項目が含まれたテスト計画書 品質目標の定義 テスト手法(技法)の定義

2		テストアーキテクチャ設計		
11		テストアーキテクチャ設計成果物を検証する。		
		テストアーキテクチャ設計成果物を要求元と公式レビューを行ない、内容の過不足や整合性の確認を行います。		
		INPUT	TASK(PROCESS)	OUTPUT
		テストアーキテクチャ設計成果物	テストアーキテクチャ設計成果物を要求元および関係者と公式レビューを行ない、要求に漏れがないことや要求との整合性の確認を行います。 指摘事項があれば、レビュー記録に基づいてレビューのフォローアップを行ないます。 全てのレビューフォローアップが完了し、要求元の承認を得ます。	レビュー記録 レビューフォローアップ 承認されたテストアーキテクチャ設計成果物

3	テスト詳細設計
	本章は、ミドルレベル シラバスに統合されます。
1	テストアーキテクチャ設計成果物を準備する。
2	テストベースを準備する
3	テスト対象の仕様を獲得する。
4	テストアーキテクチャに基づきテスト対象の仕様を分析する。
5	テスト実行条件を定義する。
6	テストカバレッジを設計する。
7	テスト条件(確認項目)を設計する。
8	テストデータを設計する。
9	テスト環境を設計する。
10	テストハーネスを設計する。
11	テスト詳細設計成果物を作成する。
12	テスト詳細設計成果物の検証をする。
4	テスト実装
	本章は、ミドルレベル シラバスに統合されます。
1	テスト詳細設計成果物を準備する。
2	テストベースを準備する
3	テスト環境を調達する。
4	テストデータを準備する。
5	テストデータを作成する。
6	テストケースを作成する。
7	テスト手順仕様を作成する。
8	テスト環境を構築する。
9	テストハーネスを作成する。
10	テストスイートを作成する。
11	テスト実装成果物を検証する。

5	テスト実行
	本章は、エントリレベル シラバスに統合されます。
1	テスト実装成果物を準備する。
2	テスト対象を準備する。
3	テスト環境を準備する。
4	テストハーネスを準備する。
5	テスト計画書を準備する。
6	リリース計画を準備する。
7	テスト実行を計画する。
8	テスト実行環境を設定する。
9	テストを実行する。
10	テスト結果を記録する。
11	実行結果と期待結果を比較する。
12	インシデントまたは不具合を報告する。
13	テスト終了判定情報を作成する。
6	テスト報告
	本章は、ミドルレベル シラバスに統合されます。
1	テスト結果を収集する。
2	インシデントおよび不具合傾向を分析する。
3	テスト実行結果を分析する。
4	不具合原因を分析する。
5	テストレポートを作成する。
6	テスト終了基準との達成差異を検証する。
7	テスト報告書に基づき報告する。

7	テスト評価		
	<p>テスト評価は、テスト報告書を含むテスト活動の全ての情報や結果を用いて、テスト活動の達成度合いを分析および評価します。また、テストプロジェクトの活動に対してポストモーテムを行い、プロジェクト成果および改善点の展開を行います。</p> <p><small>注)テスト報告の内容は、ミドルレベル シラバスの統合で作成されます。</small></p>		
1	テスト報告書を準備する。		
	テスト報告で作成したテスト報告書を準備します。テスト報告書に対するレビュー結果などの情報も用意します。		
	INPUT	TASK(PROCESS)	OUTPUT
	<p>テスト報告に関する入手物一覧表 または、チェックシート</p>	<p>該当のテスト報告書を入手します。テスト報告書の入手時には、テストの計画と実績、テスト対象のリリース実績や仕様変更などの変更履歴、テスト報告のレビュー結果など、テスト報告に関する入手物一覧または、チェックシートに基づいて必要な情報の有無も確認しておきます。</p> <p>入手した情報は、分析し易いようにテスト活動の時系列やイベントごとなどに整理しておきます。</p> <p>参考にてできる過去の類似したプロジェクトのデータや前回からの引継ぎ事項(前回の改善事項)なども用意します。</p>	<p>テスト報告書一式 (報告書事例)</p> <p>スケジュール(工程)の予定/実績</p> <p>工数の予定/実績データ</p> <p>各アウトプットのレビュー結果</p> <p>参考にてできる他プロジェクトの情報</p> <p>前回プロジェクトの情報</p>

7	テスト評価		
2	テスト活動全般の情報を収集する。		
	<p>テスト活動に関するプロジェクトの情報を収集します。収集するプロジェクト情報の事例は次のようなものがあります。</p> <p>テストスコープ</p> <p>テストタスク</p> <p>コスト計画と実績</p> <p>品質計画と実績</p> <p>組織計画と実績(要員の調達と確保、およびトレーニング計画と実績)</p> <p>コミュニケーション計画</p> <p>リスク管理計画と実績(リスク定義書、対策計画書)</p> <p>調達計画と実績</p> <p>プロジェクト改善計画(目標)</p> <p>上記の他に、テスト活動に関する計画や実績(エビデンス)、テスト活動に対する変更要求などの情報も収集しておきます。</p>		
	INPUT	TASK (PROCESS)	OUTPUT
	<p>テスト活動に関する入手物一覧表 または、チェックシート</p>	<p>該当のテスト活動の情報を入手します。テスト活動の情報の入手時には、テスト活動の情報に関する入手物一覧または、チェックシートに基づいて必要な情報の有無も確認しておきます。</p> <p>入手した情報は、分析し易いようにテスト活動の時系列やイベントごとなどに整理しておきます。</p>	<p>テスト活動情報一式 (活動情報事例)</p> <p>マスターテスト計画書</p> <p>プロジェクト計画書</p> <p>テスト要求分析書(スコープ)</p> <p>レベルテスト計画書</p> <p>各レビュー結果</p>

7 テスト評価													
3	<p>テスト活動を分析する。</p> <p>時系列およびイベントごとに整理された情報を用いて、テスト活動を分析します。テスト活動の分析には、テスト活動を評価するための分析と、テスト活動を通じて得られる技術資産を蓄積するための分析の目的に大別できます。テスト活動の評価分析では、テスト計画と実績の対比やプロジェクトへの変更要求とその対応、テスト活動中に発生したトラブルとそのリカバリ(テスト活動に影響する欠陥の発生やテスト対象物などのリリースの遅れなども含む)など、テスト活動の結果に対して分析を行います。技術資産の蓄積のための分析では、テスト活動の成果物(テストアーキテクチャ設計やテスト詳細設計、テスト実装、テスト実行、テスト報告)に対して、実績データなどの分析を行います。</p>												
	<table border="1"> <thead> <tr> <th>INPUT</th> <th>TASK(PROCESS)</th> <th>OUTPUT</th> </tr> </thead> <tbody> <tr> <td> テスト活動への改善施策(改善計画書 等) テスト報告書一式 (報告書事例) スケジュール(工程)の予定/実績 </td> <td> テスト活動を時系列に整理して、テスト活動に関する経験(暗黙知・ノウハウ)を定量的および定性的な情報として記録します。テストリポジトリを構築している場合は、これらのデータを登録します。 </td> <td> テスト設計情報 テスト実績データ </td> </tr> <tr> <td> 工数の予定/実績データ 各アウトプットのレビュー結果 参考にできる他プロジェクトの情報 前回プロジェクトの情報 </td> <td> テスト活動の定量的データ(実績データ)と基準となる指標のデータを比較して、テスト活動におけるボトルネックを抽出します。 </td> <td> 比較データ </td> </tr> <tr> <td> テスト活動情報一式 (活動情報事例) マスターテスト計画書 プロジェクト計画書 テスト要求分析書(スコープ) レベルテスト計画書 各レビュー結果 </td> <td> 標準テストプロセスとの比較やTMapなどを利用して、プロセスの成熟度を計測します。 </td> <td> プロセス成熟度判定結果 </td> </tr> </tbody> </table>	INPUT	TASK(PROCESS)	OUTPUT	テスト活動への改善施策(改善計画書 等) テスト報告書一式 (報告書事例) スケジュール(工程)の予定/実績	テスト活動を時系列に整理して、テスト活動に関する経験(暗黙知・ノウハウ)を定量的および定性的な情報として記録します。テストリポジトリを構築している場合は、これらのデータを登録します。	テスト設計情報 テスト実績データ	工数の予定/実績データ 各アウトプットのレビュー結果 参考にできる他プロジェクトの情報 前回プロジェクトの情報	テスト活動の定量的データ(実績データ)と基準となる指標のデータを比較して、テスト活動におけるボトルネックを抽出します。	比較データ	テスト活動情報一式 (活動情報事例) マスターテスト計画書 プロジェクト計画書 テスト要求分析書(スコープ) レベルテスト計画書 各レビュー結果	標準テストプロセスとの比較やTMapなどを利用して、プロセスの成熟度を計測します。	プロセス成熟度判定結果
INPUT	TASK(PROCESS)	OUTPUT											
テスト活動への改善施策(改善計画書 等) テスト報告書一式 (報告書事例) スケジュール(工程)の予定/実績	テスト活動を時系列に整理して、テスト活動に関する経験(暗黙知・ノウハウ)を定量的および定性的な情報として記録します。テストリポジトリを構築している場合は、これらのデータを登録します。	テスト設計情報 テスト実績データ											
工数の予定/実績データ 各アウトプットのレビュー結果 参考にできる他プロジェクトの情報 前回プロジェクトの情報	テスト活動の定量的データ(実績データ)と基準となる指標のデータを比較して、テスト活動におけるボトルネックを抽出します。	比較データ											
テスト活動情報一式 (活動情報事例) マスターテスト計画書 プロジェクト計画書 テスト要求分析書(スコープ) レベルテスト計画書 各レビュー結果	標準テストプロセスとの比較やTMapなどを利用して、プロセスの成熟度を計測します。	プロセス成熟度判定結果											

7 テスト評価			
4 前回の改善項目の達成率を評価する。			
テスト活動に対する改善計画に基づいて、改善項目の達成度を数値的な分析を行います。			
	INPUT	TASK(PROCESS)	OUTPUT
	プロジェクト改善計画書 改善のねらい(効果) 改善項目(目標) 改善のタスク	プロジェクト改善計画で立案した改善項目とテスト活動の実績データを比較し、達成率を算出します。 また、改善のねらい(効果)に対して改善のタスクの実施程度などを考慮して、定性的な目的の達成度合いを判定します。 改善が不十分な項目やねらい、また新たな改善すべき項目を抽出します。	改善目標の達成率 改善のねらいの達成度合い 改善のタスクの実施程度 改善未達成、新たな改善項目
5 分析結果と評価指標を比較する。			
改善の達成率、改善の達成度合いからプロジェクト改善の目的との差異を比較します。			
	INPUT	TASK(PROCESS)	OUTPUT
	改善目標の達成率 改善のねらいの達成度合い	プロジェクトの改善の目的が、改善目標の達成率や改善のねらいの達成度合いと比較しての満足度(充足度)を判定します。 プロジェクトの改善の満足度から、評価指標の見直しを行います。	プロジェクト改善の全体達成率 評価指標の見直し項目
6 改善点(良かった点・悪かった点)を抽出する。			
プロジェクト関係者から、成果および反省点を広く収集します。			
	INPUT	TASK(PROCESS)	OUTPUT
	プロジェクト成果および反省点のアンケート	プロジェクトの関係者に広く成果および反省点を収集します。 役割の異なる関係者で、プロジェクトに対する成果および反省点のギャップがないか分析します。分析した結果を一覧化して、詳細を記録します。	改善点(良かった点・悪かった点) プロジェクト成果一覧／詳細 プロジェクト反省点一覧／詳細
7 改善策を検討する。			
プロジェクトの改善点の影響度、効果などを分析し、優先度や対策を検討して次期プロジェクトへ展開を行います。			
	INPUT	TASK(PROCESS)	OUTPUT
	改善点(良かった点・悪かった点) プロジェクト成果一覧／詳細 プロジェクト反省点一覧／詳細	プロジェクトの反省点および改善点の影響度合いを分析します。影響度合いが高く、発生する確率／回数が多項目の優先度を高く設定し、根治対策または回避方策を検討します。また、改善点に対して再発防止策を検討します。	改善処置対策書

7	テスト評価		
8	テスト評価報告書を作成する。		
	テスト活動の分析結果や評価結果および改善点などを広く水平展開が実施できるように体系的に整理して、テスト評価報告書を作成する。		
	INPUT	TASK (PROCESS)	OUTPUT
	テスト評価の分析、改善評価結果 テスト報告書 テスト活動の分析結果 改善計画と達成率 改善分析結果	該当するテスト活動の改善施策の視点で、テスト活動の分析結果や評価結果を整理します。各評価結果は、関連性が明確になるような書式で文書化する。	テスト評価報告書 テスト活動改善計画 テスト活動計画対比分析 達成指標(達成項目/未達成項目) 次期改善点一覧 技術蓄積情報一覧(保管情報)
9	テスト評価報告書を開示する。		
	テスト評価報告会などを通じて、テスト評価報告書を関係者に開示します。テスト活動の技術情報および改善成果や失敗事例などは、他のテスト活動のプロジェクトにも広く水平展開します。		
	INPUT	TASK (PROCESS)	OUTPUT
	テスト評価報告書	テスト評価報告会を実施して、広くテスト評価報告書を開示します。報告会を通じて、テスト評価に関するコメントや指摘事項を収集し、次回プロジェクトへの引継ぎ資料として報告会の内容(指摘された点など)を議事録にまとめます。	開示先一覧 テスト評価報告書へのコメント、指摘事項 報告会(実施した場合)の議事録 次回プロジェクトへの引継ぎ資料
3	システテスト		
	コンポーネント統合テストの階層化にジェネリックなテスト活動として統合して記載してあります。		
4	受入テスト		
	コンポーネント統合テストの階層化にジェネリックなテスト活動として統合して記載してあります。		