XML Consortium

内部統制勉強会

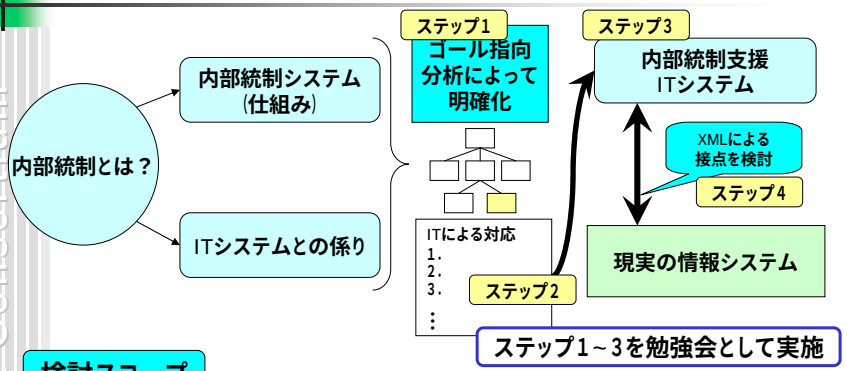
～内部統制対応を効率化するリスクパターンの活用～

2008.6.3

内部統制勉強会

XML Consortium

勉強会の進め方の考え方



検討スコープ

- J-SOX対応そのものはスコープとはせず、あるべき内部統制への対応をスコープとする。
- BPMやシステム間連携など業務の効率化やコストの削減につながる方法をスコープの中心とする。

COSOモデルを用いた内部統制のゴールモデル分析

＜凡例＞
COSOモデルがカバーする範囲

【目的】内部統制を行う

業務を効率的に行う(want) | 財務報告に誤りをなくす(want) | 法令に遵守する(want) | その他(want)

AND AND AND AND

【方法】内部統制システムを整備する

【統制環境】	【リスク評価】	【統制活動】	【情報と伝達】	【モニタリング】
事業体に属する人々の誠実性･倫理的価値観･能力、経営者の哲学･行動様式などが統制されている	目的の達成に影響を与えるリスクが識別･分析されている	リスク対処に必要な経営者の指示が実行されている	必要な情報が組織や関係者相互間に、適切に伝わっている	内部統制の有効性を継続的に監視、評価するプロセスが整備されている

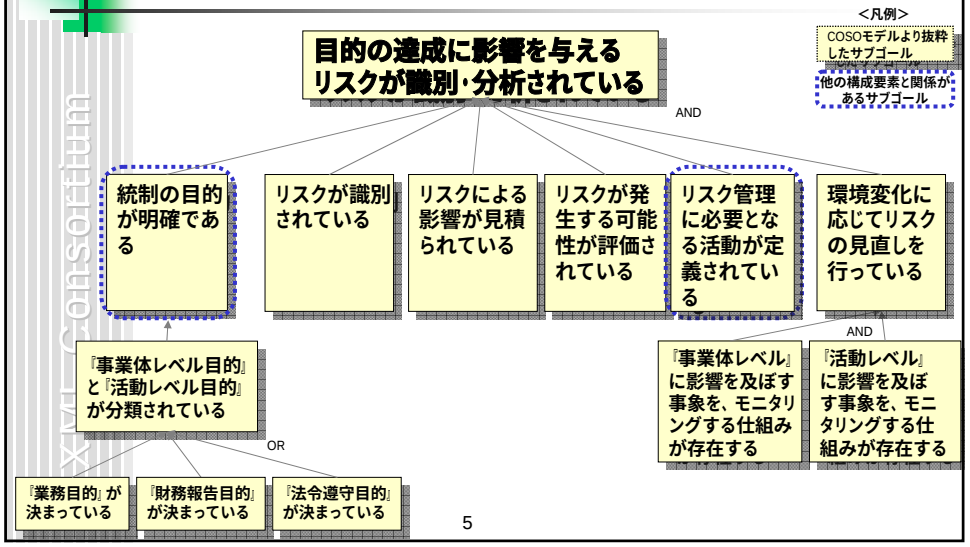
「統制環境」のゴールモデル分析

事業体に属する人々の誠実性･倫理的価値観･能力、経営者の哲学･行動様式などが統制されている

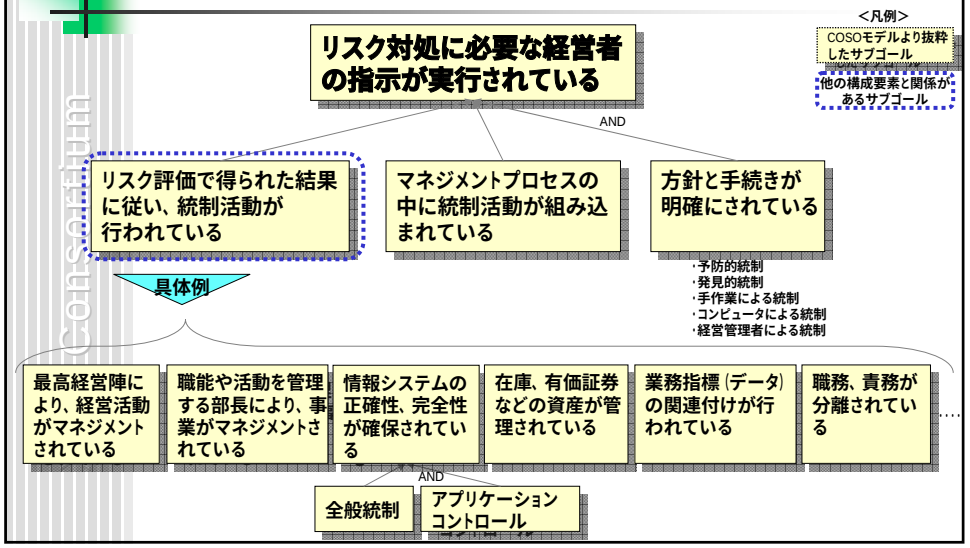
要素

経営者の哲学、行動様式	誠実性と倫理的価値観	不正に繋がる誘因	道徳的指針の提供、伝達	取締役会、監査委員会	能力に対する取組み･人的資産	権限と責任
	･企業に誠実性がある ･倫理的価値観が定められている	･『誘因』が除去、低減されている ･『誘惑』が除去、低減されている ･十分に統制されている報告システムが存在する	･行動の善悪について、明示的な指針がある（無知の除去）	･構成員の経験と能力が十分である ･経営者から独立して活動している ･関与、調査する活動の範囲が適切である	･研修が日常的に行われている ･定期的な職務評価を通じて、人事異動や昇進が行われている ･「業務体が従業員に期待している水準」を伝えている	･目的の達成に必要な範囲内で権限が委譲されている ･適切な報告ラインが確立されている

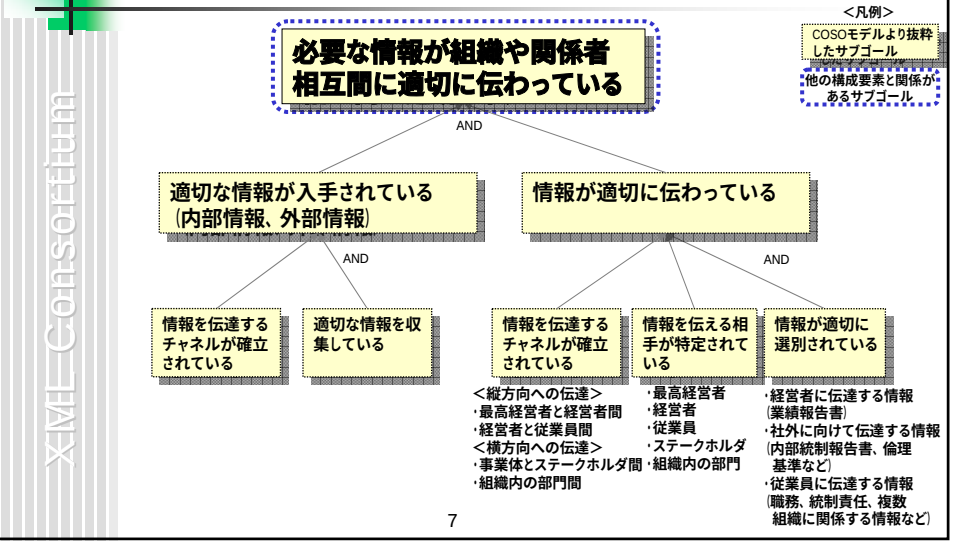
「リスク評価」のゴールモデル分析



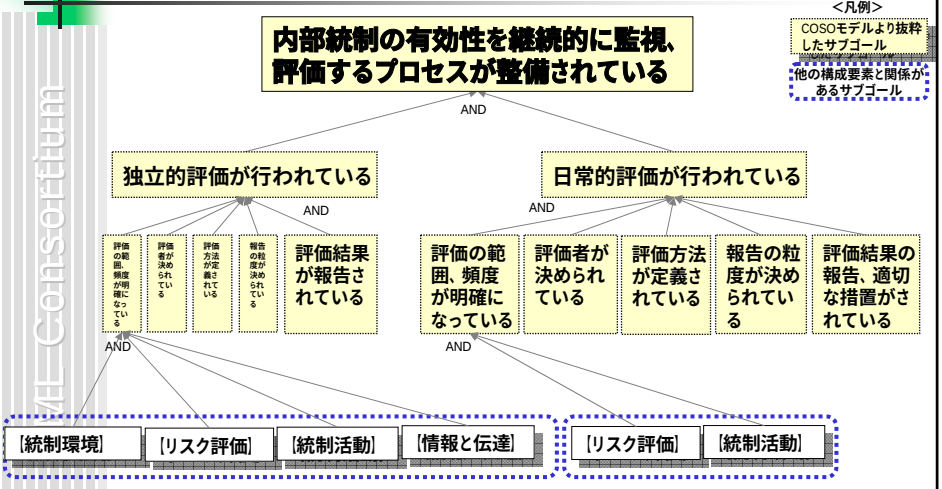
「統制活動」のゴールモデル分析



「情報と伝達」のゴールモデル分析



「モニタリング」のゴールモデル分析



次のステップに向けて

- 課題
  - ゴールモデル分析の内容は抽象的
  - ゴールモデル分析（特にリスク評価、統制活動）を詳細化していくために、具体的な材料が必要

- 進め方
  - 事故、失敗の事例集である「失敗知識データベース」が活用できないか検討する
  - 失敗知識データベースの事例の再発防止は内部統制が肝。COSOとの関係を整理したい

失敗知識データベース

- 科学技術振興機構（JST）
- 機械、建設、化学などのものづくりに関係する事故や失敗の事例を分析し、得られる教訓とともにデータベース化したもので、ものづくり現場の事故や失敗の未然防止への活用を目指している
- 平成17年3月23日（水）一般公開　http://shippai.jst.go.jp/
- 事業統括：畑村 洋太郎　工学院大学教授
- 2008年5月現在、1136事例
  - 原子力発電所配管破裂事故、銀行システム統合不具合、トラック車輪脱落（リコール隠し）　等々
- 分野
  - 機械、化学、石油、石油化学、建設、電気・電子・情報、電力・ガス、原子力、航空・宇宙、自動車、鉄道、船舶・海洋、金属、食品、自然災害、その他

失敗知識データベースWebサイト

失敗事例掲載例

原因まんだら

JST畑村委員会作成 2002

行動まんだら

JST畑村委員会作成 2002

結果まんだら

JIST標村委員会作成 2002

リスク発生要因の分類（運用時）

運用手順に着目してリスク発生要因を以下のように分類を実施

リスクが発生しないように運用手順が、

- 決まっている
  - 正しい
    - 無断変更
      - 上層部の圧力
      - 経営重視の対応（利益重視･安全軽視）
    - 意識的な手順違反
      - 過信
      - 期限やコストなどのプレッシャー
    - 確認不足
      - 技術不足
      - 注意･用心不足
  - 間違っている
    - 未知
    - 想定外
    - 情報伝達不足
  - 更新されていない
    - 環境変化未対応
- 無いまたは不足している
  - 考慮漏れ
  - 未知
  - 想定外

リスク発生要因の分類（設計時）

同様に、リスクが発生しないように設計時の検討手順が、

- 決まっている
  - 正しい
    - 無断変更
      - 上層部の圧力
      - 経営重視の対応（利益重視・安全軽視）
    - 意識的な手順違反
      - 過信
      - 技術的こだわり
      - 外部条件からの制約
      - 期限やコストなどのプレッシャー
    - 確認不足
      - 技術不足
      - 注意・用心不足
  - 間違っている
    - 未知
    - 想定外
    - 情報伝達不足
    - 技術的こだわり
    - 外部条件からの制約
  - 更新されていない
    - 環境変化未対応
- 無いまたは不足している
  - 考慮漏れ
  - 未知
  - 想定外

対応策の整理（1）

リスク発生原因	主な要因	対応策		
		内部統制の入門と実践より*	失敗百選より（設計）	失敗百選より（運用）
手順が無いまたは不足している	・考慮漏れ ・未知 ・想定外	・規程、マニュアルの整備	・設計基準の整備 ・過酷な条件、最悪シナリオの想定 ・関連情報、類似事例、過去の欠陥や問題点の収集・解析と手順への反映 ・仮想演習によるチェックの実施 ・不確定要素があれば余裕を持たせる ・技術者の専門領域間の情報伝達	・過酷な条件、最悪シナリオの想定 ・関連情報、類似事例、過去の欠陥や問題点の収集・解析と手順への反映 ・仮想演習によるチェックの実施 ・人事異動時の情報共有 ・設計基準を超えた場合の対処の想定
手順が間違っている	・未知 ・想定外 ・情報伝達不足 ・技術的こだわり ・外部条件からの制約	・照合による妥当性の検証 ・証憑による事実の検証 ・業績などの指標の分析 ・異常値把握	・過酷な条件、最悪シナリオの想定 ・関連情報、類似事例、過去の欠陥や問題点の収集・解析と手順への反映 ・仮想演習によるチェックの実施 ・専門家の専門領域間の情報伝達 ・複合因子での発生を考慮し総合的に余裕度を設定 ・外国輸入技術の未消化適用をしない	・過酷な条件、最悪シナリオの想定 ・関連情報、類似事例、過去の欠陥や問題点の収集・解析と手順への反映 ・仮想演習によるチェックの実施 ・専門家の専門領域間の情報伝達
手順が更新されていない	・環境変化未対応	・照合による妥当性の検証 ・証憑による事実の検証 ・業績などの指標の分析 ・異常値把握	・最新技術の設計基準への反映 ・製品変更の際の発生しうる問題点の明確化と対応検討 ・不適切な手順の放置の禁止 ・過去の実績に準拠した基準の最新情報の確認	・不適切な手順の放置の禁止 ・過去の実績に準拠した基準の最新情報の確認

*内部統制の入門と実践 佐々野未知 中央経済社 2006 年1 月 ISBN978-4-502-27550-0

対応策の整理（2）

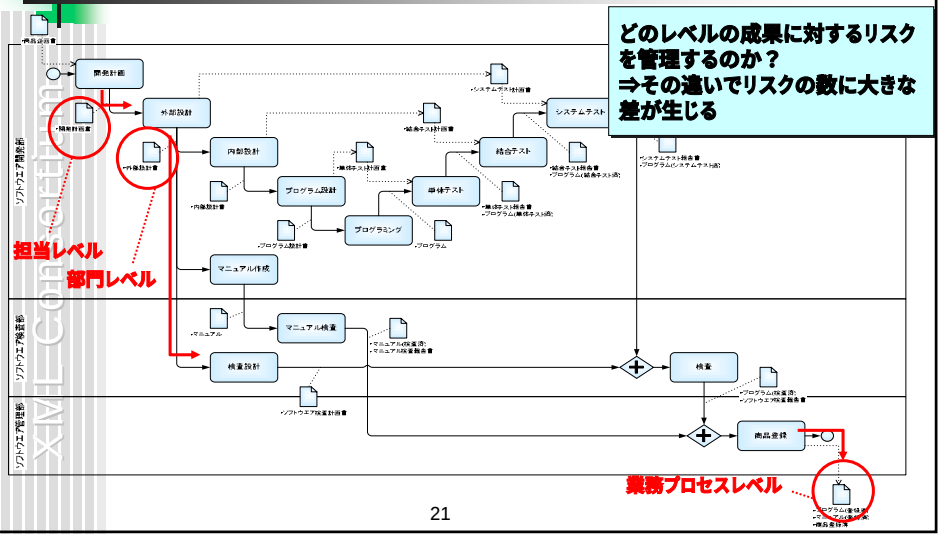
リスク発生原因	主な要因	対応策		
		内部統制の入門と実践より	失敗百選より（設計）	失敗百選より（運用）
手順書の無断変更	・利益重視 ・安全軽視 ・上層部の圧力	・職務の分離・分掌 ・権限の移譲 ・定期的な配置転換	・安全対策には経費削減をしない ・コスト低減できる部分とできない部分を明確に区別する	・資金不足、人員不足、タイトなスケジュールの回避 ・関係者間の情報共有と責任体制の明確化 ・検査基準の整備
意識的な手順違反	・過信 ・技術的こだわり ・外部条件から・の制約 ・期限やコストなどのプレッシャー	・上長による承認 ・照合による妥当性の検証 ・証憑による事実の検証 ・職務の分離・分掌 ・権限の移譲 ・定期的な配置転換 ・業績などの指標の分析・異常値把握 ・ITを利用した自動化 ・セキュリティ管理	・資金不足、人員不足、タイトなスケジュールの回避 ・関係者間の情報共有と責任体制の明確化 ・検査基準の整備	・運用状態を適切に確認できるようにする ・不具合発生を検出する機構の組込み ・検査基準の整備
手順の確認不足	・技術不足 ・注意、用心不足	・上長による承認 ・照合による妥当性の検証 ・証憑による事実の検証 ・業績などの指標の分析・異常値把握 ・ITを利用した自動化 ・セキュリティ管理	・関係者の技術水準の確認 ・検査基準の整備	・分かりやすく誤判断の起こしにくい手順、構造 ・誤操作や誤判断に対する安全対策の組み込み ・不具合発生を検出する機構の組込み ・検査基準の整備

以降、P.18,19の表をリスクパターンと呼ぶ

内部統制文書化における リスクパターンの活用

- 内部統制文書化における問題点
  - リスクの識別において、大きな個人差が生じてしまう
    - 同じような業務でも、ある人は100のリスクを、他の人は10のリスクを識別するということが起こり得る
      - その主な要因は、リスクをどのようなレベルで管理するのかという点で統一的な考え方がないこと
  - リスクやコントロールの網羅性をどのようにチェックすれば良いかがわからない
    - リスクを識別する手順、コントロールを定義する手順が確立されていないため、チェックの視点が曖昧になる
- リスクパターンを活用したRCM作成手順の確立

リスク管理レベルの違い



提案：リスク識別

- どのレベルの成果に対してリスクの管理をするのかを明確にする
  - 担当レベル、部門レベル、ビジネスプロセスレベル
  - リスク管理を行うレベルで業務フローを記述する詳細度（業務フローに記述すべきスイムレーンなど）が決まる
- リスク識別の網羅性を管理目標でチェックする
  - 「成果×目標」　※各成果に対して各目標が達成できる状況を列挙する
    - 「業務の有効性及び効率性」の視点
    - 「財務報告の信頼性」の視点
    - 「事業活動に関わる法令等の遵守」の視点
    - 「資産の保全」の視点
  - 「成果×目標」が達成できない状況＝リスク　と考える

リスク識別の例

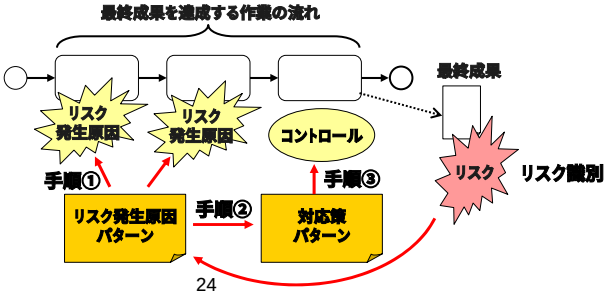
ソフトウェア開発部のリスク識別の実施例を示す

成果	観点	目標	リスク内容
設計書(外部,内部,ﾌﾟﾛｸﾞﾗﾑ)	品質	プロジェクトの情報の共有	プロジェクト情報(ｽｹｼﾞｭｰﾙ等)が伝わらない
設計書(外部,内部,ﾌﾟﾛｸﾞﾗﾑ)	品質	製品(ﾊｰﾄﾞ)仕様情報の共有	仕様変更の情報が伝わらない
設計書(外部,内部,ﾌﾟﾛｸﾞﾗﾑ)	品質	開発プロセス遵守	開発手順が守られない(省略される等)
テスト計画書(外部,内部,ﾌﾟﾛｸﾞﾗﾑ	品質	テスト項目の十分性	テスト項目が十分でない
テスト計画書(外部,内部,ﾌﾟﾛｸﾞﾗﾑ	品質	導入IPの品質確保	導入IPの評価やテストが十分でない
テスト報告書(外部,内部,ﾌﾟﾛｸﾞﾗﾑ	品質	開発プロセス遵守	開発手順が守られない(省略される等)
プログラム(ｼｽﾃﾑﾃｽﾄ済)	品質	入試レベル(ｼｽﾃﾑﾃｽﾄ完)に到達	テストでバグが十分に修正されていない
プログラム(ｼｽﾃﾑﾃｽﾄ済)	期間/納期	生産性(+)≧基準値	納期に間に合わない
プログラム(ｼｽﾃﾑﾃｽﾄ済)	コスト	実績開発工数－予定開発工数≦基準値	予定工数を超過する
プログラム(ｼｽﾃﾑﾃｽﾄ済)	品質	開発プロセス遵守	開発手順が守られない(省略される等)

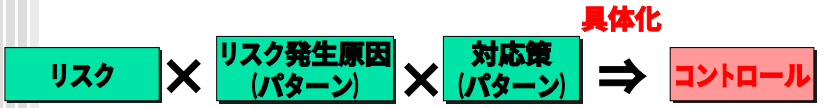
成果は成果物でみる

提案：コントロール検討

- リスクパターン（リスク発生原因、対応策）を活用してコントロールを検討する
  - 手順①：リスクに関連するリスク発生原因をリスクパターンの中から特定する
  - 手順②：リスク発生原因の対応策をリスクパターンの中から特定する
  - 手順③：対応策（パターン）を業務に即した具体的な表現にすることでコントロールを定義する



コントロール検討の例



リスク内容	リスク発生原因	対応策	コントロール
プロジェクト情報(ｽｹｼﾞｭｰﾙ等)が伝わらない	運用：手順の確認不足	情報伝達手順の規約化	工程会議での連絡・確認
仕様変更の情報が伝わらない	運用：手順の確認不足	情報伝達手順の規約化	工程会議での連絡・確認
開発手順が守られない(省略される等)	作成：意識的な手順違反	各プロセスの責任体制の明確化	チェックリストによる手順確認
テスト項目が十分でない	作成：意識的な手順違反	最新技術の作業基準へのフィードバ	チェックリストによる項目確認
導入IPの評価やテストが十分でない	作成：手順がない/不足	最新技術の作業基準へのフィードバ	チェックリストによる項目確認
開発手順が守られない(省略される等)	作成：意識的な手順違反	各プロセスの責任体制の明確化	チェックリストによる項目確認

コントロール検討における注意点

- 業務フローを利用してコントロールの記述レベルを合わせる
  - 1つ下のレベル（部門レベルのリスクを管理する場合には担当レベル）の作業をコントロールとする
    - ○○レビュー、○○テスト など
    - レビュー、テスト などの作業内で実施される様々なコントロールを個々に列挙しないこと
      - ○○レビューチェックリスト、○○テストチェックリスト などとして別途文書化
      - コントロールの記述：「○○レビューチェックリストにもとづきレビューする」
- コントロールに対するリスクを識別しない
  - 無限ループに陥る（コントロール⇒リスク⇒コントロール⇒リスク）

ソフトウェア開発プロセスでの検証

- P.21のソフトウェア開発フローをサンプルとして利用し、前頁までの考え方に基づいてRCM（リスクコントロールマトリクス）を作成

- 失敗百選を利用して導き出した、リスク分類と対応策の考え方により、抜け、漏れのチェックが可能なことを確認

さらに、

- 一般的な意味でのソフト検査業務では、検査設計書に対するコントロールがないことを発見

[参考]作成したRCM

No.	開発プロセス	組織単位	アウトプット	観点	目標	リスク内容	リスク分類	コントロール	対応策
1	ソフトウェア開発	ソフトウェア開発部	設計書(外部,内部,プログラム)	品質	プロジェクトの情報の共有	プロジェクト情報(スケジュール等)が伝わらない	運用：手順の確認不足	工程会議での連絡・確認	情報伝達手順の規約化
2	ソフトウェア開発	ソフトウェア開発部	設計書(外部,内部,プログラム)	品質	製品ハード仕様情報の共有	仕様変更の情報が伝わらない	運用：手順の確認不足	工程会議での連絡・確認	情報伝達手順の規約化
3	ソフトウェア開発	ソフトウェア開発部	設計書(外部,内部,プログラム)	品質	開発プロセス遵守	開発手順が守られない(省略される等)	作成：意識的な手順違反	チェックリストによる手順確認	各プロセスの責任体制の明確化
4	ソフトウェア開発	ソフトウェア開発部	テスト計画書(外部,内部,プログラム	品質	テスト項目の十分性	テスト項目が十分でない	作成：意識的な手順違反	チェックリストによる項目確認	最新技術の作業基準へのフィードバック
5	ソフトウェア開発	ソフトウェア開発部	テスト計画書(外部,内部,プログラム	品質	導入PIの品質確保	導入PIの評価やテストが十分でない	作成：手順がない/不足	チェックリストによる項目確認	最新技術の作業基準へのフィードバック
6	ソフトウェア開発	ソフトウェア開発部	テスト報告書(外部,内部,プログラム	品質	開発プロセス遵守	開発手順が守られない(省略される等)	作成：意識的な手順違反	チェックリストによる項目確認	各プロセスの責任体制の明確化
7	ソフトウェア開発	ソフトウェア開発部	プログラム(システムテスト済)	品質	入試レベルのシステムテスト完に到達	テストでバグが十分に修正されていない	作成：意識的な手順違反	チェックリストによる成果物確認	各プロセスの責任体制の明確化
8	ソフトウェア開発	ソフトウェア開発部	プログラム(システムテスト済)	期間/納期	生産性：≧基準値	納期に間に合わない	作成：手順がない/不足	チェックポイント設定による中間チェッ	最新技術の作業基準へのフィードバック
9	ソフトウェア開発	ソフトウェア開発部	プログラム(システムテスト済)	コスト	実績開発工数－予定開発工数≦基準値	予定工数を超過する	作成：手順がない/不足	チェックポイント設定による中間チェッ	最新技術の作業基準へのフィードバック
10	ソフトウェア開発	ソフトウェア開発部	プログラム(システムテスト済)	品質	開発プロセス遵守	開発手順が守られない(省略される等)	作成：意識的な手順違反	チェックリストによる手順確認	各プロセスの責任体制の明確化
11	ソフトウェア開発	ソフトウェア開発部	マニュアル設計書	品質	使いやすいマニュアル	ソフトの使われ方とマニュアルの説明に乖離がある	作成：手順がない/不足	チェックリストによる成果物確認	最新技術の作業基準へのフィードバック
12	ソフトウェア開発	ソフトウェア開発部	マニュアル	品質	設計書に基づく記述	マニュアル設計書とマニュアル記述に乖離がある	作成：手順の確認不足	設計書との照合	各プロセスの責任体制の明確化
13	ソフトウェア開発	ソフトウェア開発部	マニュアル	品質	記述ミス≦基準値	ソフトの仕様を正確に記述していない	作成：手順の確認不足	仕様書との照合	各プロセスの責任体制の明確化
14	ソフトウェア検査	ソフトウェア検査部	プログラム(検査済)	品質	検査検出バグ数≦基準値	検査でバグが多数検出される	作成：手順がない/不足	チェックリストによる成果物確認	最新技術の作業基準へのフィードバック
15	ソフトウェア検査	ソフトウェア検査部	プログラム(検査済)	品質	障害発生～初期対応までの時間≦基準値	障害発生からの対応が遅い	運用：意識的な手順違反	定期的な工程進捗会議の開催	各プロセスの責任体制の明確化
16	ソフトウェア検査	ソフトウェア検査部	ソフトウェア検査報告書						
17	ソフトウェア検査	ソフトウェア検査部	マニュアル(検査済)						
18	ソフトウェア検査	ソフトウェア検査部	マニュアル検査報告書						
19	ソフトウェア登録	ソフトウェア管理部	プログラム(登録済)	品質	出荷製品の登録	未登録出荷製品がある	運用：意識的な手順違反	チェックリストによる手順確認	各プロセスの責任体制の明確化
20	ソフトウェア登録	ソフトウェア管理部	マニュアル(登録済)	品質	出荷製品の登録	未登録出荷マニュアルがある	運用：意識的な手順違反	チェックリストによる手順確認	各プロセスの責任体制の明確化
21	ソフトウェア登録	ソフトウェア管理部	商品登録簿	品質	出荷製品と登録内容の一致	登録簿に一致しない製品が出荷されている	運用：手順の確認不足	チェックリストによる手順確認	各プロセスの責任体制の明確化

注）T：生産性＝ステップ数/(外部設計～システムテストにかかった工数)

No.	開発プロセス	組織単位	アウトプット	観点	目標	リスク内容	リスク発生原因	コントロール	対応策
1		ソフトウェア開発統括部	商品企画書						
2		ソフトウェア開発統括部	プログラム(システムテスト済)	品質	入試レベルのシステムテスト完了到達率≧基準値	テスト不十分のまま検査工程に入る	作成：手順の確認不足	チェックリストによる成果物確認	
3		ソフトウェア開発統括部	プログラム(登録済)	品質	出荷製品登録率≧基準値	未登録出荷製品がある	運用：手順の確認不足	チェックリストによる手順確認	各プロセスの責任体制の明確化
4		ソフトウェア開発統括部	マニュアル(登録済)	品質	出荷製品登録率≧基準値	未登録出荷マニュアルがある	運用：手順の確認不足	チェックリストによる手順確認	各プロセスの責任体制の明確化
5		ソフトウェア開発統括部	商品登録簿	品質	出荷製品登録簿一致率≧基準値	登録簿に一致しない製品が出荷されている	運用：手順の確認不足	チェックリストによる手順確認	各プロセスの責任体制の明確化

リスク管理構造を整理してXMLで表現

リスク X リスク発生原因 X 対応策 → コントロール

```
<リスク一覧>
<リスク>
  <リスク内容>
    テスト項目が十分でない
  </リスク内容>
  <達成できない目標>
    テスト項目の十分性
  </ 達成できない目標>
  <目標の観点>
    品質
  </ 目標の観点>
  <成果物>
    テスト計画書
  </成果物>
  <プロセス>
    ソフトウェア開発
  </プロセス>
  <部署>
    開発部
  </部署>
</リスク>
<リスク>
  ・・・・・・・・・・・・・・・・
</リスク>
</リスク一覧>
```

```
<リスク発生原因一覧>
<リスク>
  <リスク内容>
    テスト項目が十分でない
  </リスク内容>
  <発生原因 分類＝'作成'>
    意識的な手順違反
  </発生原因>
  <発生原因 分類＝'運用'>
    意識的な手順違反
  </発生原因>
  <発生原因 分類＝'・・・'>
    ・・・・・・・・・・・・・・・・
  </発生原因>
</リスク>
<リスク>
  ・・・・・・・・・・・・・・・・
</リスク>
</リスク発生原因一覧>
```

```
<対応策一覧>
 <リスク発生原因 分類＝'作成'
  原因＝'意識的な手順違反'>
  <対応策>
    最新技術の作業基準のFDBK
  </対応策>
  <対応策>
    ・・・・・・・・・・・・・・・・
  </対応策>
 </リスク発生原因>
 <リスク発生原因 分類＝'運用'
  原因＝'意識的な手順違反'>
  <対応策>
    ・・・・・・・・・・・・・・・・
  </対応策>
 </リスク発生原因>
 <リスク発生原因 分類＝'・・・'>
    ・・・・・・・・・・・・・・・・
 </リスク発生原因>

  ・・・・・・・・・・・・・・・・

</対応策一覧>
```

```
<コントロール一覧>
 <コントロール>
  <コントロール名称>
    チェックリストによる項目確認
  </コントロール名称>
  <リスク内容>
    テスト項目が十分でない
  </リスク内容>
  <発生原因 分類＝'作成'>
    意識的な手順違反
  </発生原因>
  <対応策>
    最新技術の作業基準のFDBK
  </対応策>
 </コントロール>

  ・・・・・・・・・・・・・・・・

</コントロール一覧>
```

XMLで表現する利点

リスク X リスク発生原因 X 対応策 → コントロール


```
<リスク一覧>
<リスク>
  <リスク内容>
    テスト項目が十分でない
  </リスク内容>
  <達成できない目標>
    テスト項目の十分性
  </ 達成できない目標>
  <目標の観点>
  </プロセス>
  <部署>
    開発部
  </部署>
</リスク>
<リスク>
  ・・・・・・・・・・・・・・・・
</リスク>
</リスク一覧>
```

```
<リスク発生原因一覧>
<リスク>
  <リスク内容>
    テスト項目が十分でない
  </リスク内容>
  <発生原因 分類＝'作成'>
    意識的な手順違反
  </発生原因>
  <発生原因 分類＝'運用'>
    意識的な手順違反
  </発生原因>
  <発生原因 分類＝'・・・'>
    ・・・・・・・・・・・・・・・・
  </発生原因>
</リスク>
<リスク>
```

```
<対応策一覧>
 <リスク発生原因 分類＝'作成'
  原因＝'意識的な手順違反'>
  <対応策>
    最新技術の作業基準のFDBK
  </対応策>
  <対応策>
    ・・・・・・・・・・・・・・・・
  </対応策>
 </リスク発生原因>
 <リスク発生原因 分類＝'運用'
  原因＝'意識的な手順違反'>
  <対応策>
    ・・・・・・・・・・・・・・・・
  </対応策>
```

```
<コントロール一覧>
 <コントロール>
  <コントロール名称>
    チェックリストによる項目確認
  </コントロール名称>
  <リスク内容>
    テスト項目が十分でない
  </リスク内容>
  <発生原因 分類＝'作成'>
    意識的な手順違反
  </発生原因>
  <対応策>
    最新技術の作業基準のFDBK
  </対応策>
 </コントロール>
```


部署別検索をすることで、整合性の確認に使用できる

容易にリスク発生原因一覧XML文書を概要と詳細に分離したり、統合したりして可読性を向上できる

容易に対応策一覧XML文書から必要部分のリストを生成し、Word等の対応策指示書や対応結果報告書とかに貼り付けられる

容易にコントロール一覧定義XML文書から必要部分の**チェックリストを生成**し、Excel等に貼り付けられる

最後に、
内部統制IT支援システムへの発展の考え方