千葉工業大学

博士学位論文

情報システム開発における
リスクマネジメント方法に関する研究

平成 29 年 3 月

所属専攻：マネジメント工学

学生番号・氏名：1499503　蒋　成荣

指導教員：森　雅俊　教授

目次

第1章 序論

1.1 研究背景

近年，情報化が進むにつれて，企業は情報システムへの依存度が高くなっている．企業の情報化が進む中,1980 年代から日本企業は情報システム開発の人材不足とコスト高騰を抑えるために,海外でのソフト開発を拡大している.現在，日本から海外への発注総額は世界三位になっている[1].自国企業が情報システム開発事業の一部を海外の事業者・子会社に委託することは情報システムのオフショア開発と呼ばれる.情報システムのオフショア開発には，一般的な製品生産と異なって，受発注間のコミュニケーション能力の差などの特有なリスクが存在している.このように，企業が情報システムのオフショア開発におけるリスクを把握するため，リスクマネジメント方法を研究し，リスクマネジメントの適用化を図る必要がある.

また，企業や組織の運営においても，インターネットを基盤として利用しているため，情報セキュリティのリスクが存在している．情報セキュリティリスクが発生すると，企業や組織に大きな被害をもたらす．例えば，2014 年 6 月に，通信教育講座企業ベネッセにおいて個人情報が漏洩するインシデントが発生した．そのため，ベネッセの 2016 年 3 月期連結決算の最終損益は 82 億円の赤字となった [2]．これは，情報セキュリティ対策により約 260 億円の特別損失を計上した 2015 年 3 月期の 107 億円に続くもので，2 期連続の最終赤字を記録した.

経済産業省の平成 26 年度の調査報告によれば，調査を受けた企業の中で，23.2%の企業は情報セキュリティ上のトラブル(重要情報の漏洩，不正アクセス，コンピュータウィルス及びシステムトラブルなど)にあった [3]．上記のように，情報セキュリティマネジメントは多くの企業や組織にとって重要な課題となっている．これに対し，企業では，情報セキュリティマネジメントの重要性を認識し，情報セキュリティマネジメントに力を入れている．例えば，国際標準化機構 ISO の公開情報によると，2014 年までの世界の ISO/IEC 27001 認証取得件数は，2013 年と比べて 7%増加し，2 万 3972 件となっている．日本でも ISMS(Information Security Management System，情報セキュリティマネジメントシステム)を取得する件数は，2015 年 12 月の時点で 4789 件となっている．情報セキュリティマネジメントに対する意識が強くなる一方，多くの企業はセキュリティ対策に困難を感じている [4]．経産省の調査によると，全業種企業の 59.4%，情報サービス業企業の 68.3%が情報セキュリティ対策を実施する時に「手間・コストがかかる」と感じ，主な対策の阻害要因となっている [3].

情報サービス業の中にソフトウェア業を含めて，情報システム開発サービスを提供し，顧客のビジネス支援を行っている．情報システム開発における情報セキュリティ対策がゆるいと顧客に対する大きな損失を招く．前述のベネッセ個人情報流出事件では，データベースに保存している顧客の個人情報を情報開発会社の SE が無断で外部に持ち出し，流出し，これに対応するため，データベースの稼働を停止せざるを得なかった．また，情報システムが運営開始後，システムの脆

弱性が見つけられると，システムの修正に大きなコストが発生する．このため，情報システム開発において生じる情報セキュリティリスクに注目し，対策を実施するマネジメントが必要である．

情報システム開発における従来のリスクマネジメントでは，情報セキュリティリスクの連鎖関係の観点に基づく検討が十分ではない．PMBOK（Project Management Body of Knowledge，プロジェクトマネジメント知識体系ガイド）では，マネジメント対象はプロジェクトに存在する不確実性に起因するものであるが，リスク連鎖に関しては，十分に記されていない．また，現状の情報システム開発会社はシステム開発プロジェクトを評価するため，CMMI（Capability Maturity Model Integration，能力成熟度モデル統合）を導入している．CMMI のリスクマネジメントの対象はシステムまたソフトウェアの品質である．この二つのリスクマネジメントのリスク分析手法としては，マトリックス分析が主なものである．情報セキュリティリスクを分析する場合，分析要因は情報資産，脅威と脆弱性の 3 つであるが，脆弱性と脅威の間には依存関係がある．各脆弱性の間の連鎖関係もリスク値に影響を与えるため，従来のマトリックス分析手法によるもので，情報セキュリティリスクを分析することには活用できない．

そこで本研究では，連鎖的な因果関係を表現できるベイジアンネットワーク技術に注目し，情報システム開発において，新しいリスク分析手法をモデル化し提案することとした．

1.2　本研究の目的と意義

1.2.1　本研究の目的

本研究の第 1 の目的は情報システムのオフショア開発におけるリスクマネジメントの分析手法の提案を行うことである．その方法として，リスク分析とコンジョイント分析に基づいて評価方法を考案し，CMMI のリスクマネジメント・プロセスの改善案を提案する．

第 2 の目的は情報セキュリティリスクのマネジメントモデルを考察し，リスク分析を通して，組織の情報セキュリティ対策の向上を目指す．ベイジアンネットワークを用いて，情報セキュリティリスクの因子を対象として，その分析方法を考案し，情報システム開発における情報セキュリティリスク因子間の連鎖的な因果関係を表現する手法を提案する．

1.2.2　本研究の意義

まず，本研究では情報システムのオフショア開発プロジェクトのリスクを管理するため，本論文で提案するリスクマネジメント・プロセスとそれを用いたオフショア開発プロジェクトのリスク評価は，情報システムのオフショア開発プロジェクトを推進するためのリスクマネジメントに対して有益なマネジメント手法を提供するものと考える．

また，情報システムを開発する場合，これまでの情報セキュリティリスクの分析手法は以前のリスク分析手法と同様で，情報セキュリティ中の脅威や脆弱性間の因果関係を考慮していない．

本論文で提案する情報セキュリティリスク分析手法は，情報システム開発における新たな情報セキュリティのリスクマネジメント手法を提供するものと考える．

すなわち，本研究の意義は，下記のように要約できる．

第一，情報システムのオフショア開発プロジェクトを成功に導くためのリスクマネジメントの分析を適切にする方法の提案をする．

第二，情報システム開発における情報セキュリティをマネジメントする際のリスク分析を適切に行う一つの手法を示す点である．

1.3　研究の前提

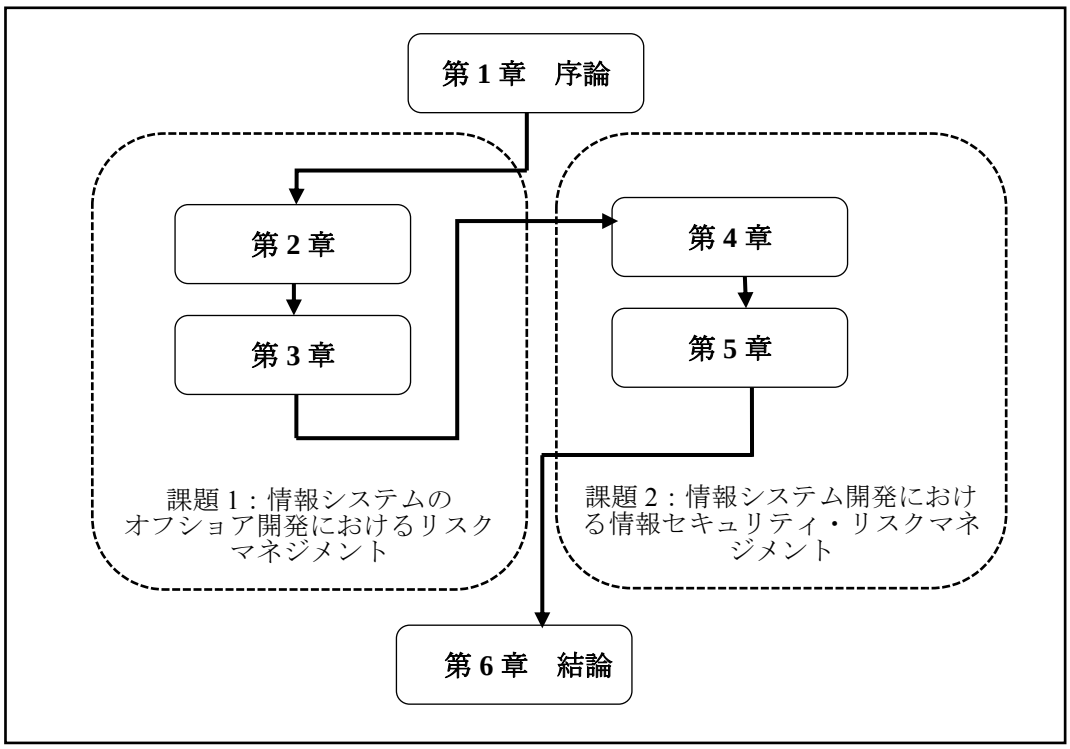
本研究は，情報システム開発におけるオフショア開発を研究対象としている．一般に，オフショア開発は世界の各国で行なわれているが，本研究の対象は，日本における一般企業の経理会計システムの開発を主な対象とする．即ち，日本企業がオフショア開発の発注側となり，受注側は中国やベトナムの IT 開発企業を前提としている．

また，情報システムの開発モデルは，ウォーターフォールモデルを用い，CMMI に基づくマネジメントを参考としている．

1.4　論文の構成

本論文では二つ課題を研究し，解決策を提案した．第 1 の課題は情報システムのオフショア開発におけるリスクマネジメントの問題点である．第 2 の課題は情報システム開発における情報セキュリティ・リスクマネジメントの問題点である．

本論文は 6 章の構成になっている．その中で，第 2 章と第 3 章は，課題 1 を取り上げ研究し，解決策を提案した．第 4 章と第 5 章では，課題 2 を取り上げ研究し，解決策を提案した．



各章の詳説については，次のようになっている．

第 1 章の序論では，研究背景を詳述し，本研究の目的と意義について述べる．その中に，本論文の新規性及び立ち位置について説明した．

第 2 章の情報システムのオフショア開発におけるリスクマネジメントでは，情報システムのオフショア開発におけるリスクとそのリスクマネジメントを考察し，現状と既存課題を分析する．情報システムのオフショア開発をリスクマネジメントする場合に存在している 2 つ問題点を提出した．

第 3 章の情報システムのオフショア開発におけるリスク分析手法では，第 2 章で提出した問題点に対する研究を行い，CMMI リスクマネジメント中の不足点に対して，コンジョイント分析を用いた CMMI のリスクマネジメント・プロセスの解決策を提案した．

第 4 章の情報セキュリティリスクとリスクマネジメントでは，情報セキュリティ及び情報セキュリティリスクのマネジメント手法の基礎知識を論述し，既存のセキュリティリスク分析方法の問題点を述べた．

第 5 章のベイジアンネットワークによる情報セキュリティリスク分析の提案では，ベイジアンネットワークを利用し，リスク分析手法を提案することを解説した．

第 6 章の結論では，本論文では情報システム開発において存在している 2 つの課題に対する解決策を提案した．一つはオフショア開発におけるリスクマネジメントに対する改善策を提案である．もう一つは情報セキュリティのリスクマネジメントに対する提案である．本章で，その 2 つ研究結果と今後の課題をまとめ，結論を述べた．

第2章 情報システムのオフショア開発における リスクマネジメント

2.1 情報システムのオフショア開発

2.1.1 オフショア開発の歴史と特色

ソフトウェアのオフショア開発とは，ソフトウェア開発をする場合に，企業や組織が人件費の安価な海外のソフトウェア会社や海外の子会社に委託することである．オフショア開発が始まった当初は，概念設計，詳細設計の上流工程は日本側が担当し，製造工程以下の下流工程を委託することから始まった [5].

日本においては 1970 年代から，このソフトウェア工場を実現しようと試みられた．海外におけるオフショア開発においても，開発コストを低減したいとの考え方は同じである．

図 1 に示すように，現在，日本の IT 企業のオフショア開発相手国は，アジア諸国が中心であり，中国，ベトナム，インド，フィリピンの 4 カ国が主な選択肢となっている [6].

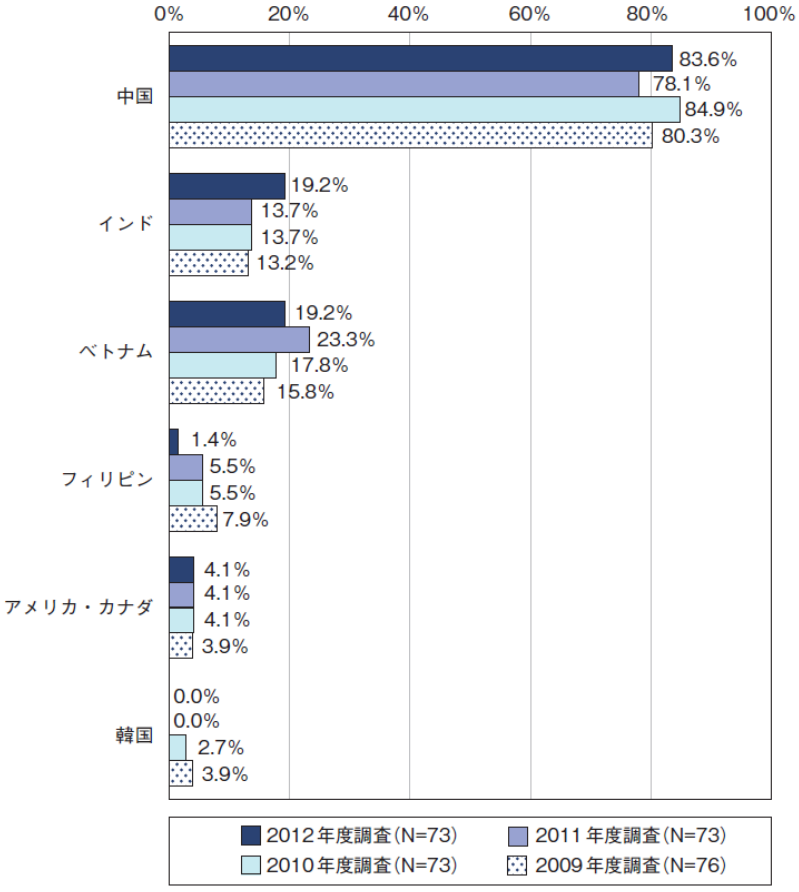


図 1. オフショア開発発注先相手国の実績(引用：IT 人材白書 2013 [6])

その中で，中国は発注額の 8 割を占めており，最大の相手国となっている．日本の中国へのオフショア開発の歴史は，1980 年半ばから始まったとされている．中国は日本と比べて IT 専攻者の人数が圧倒的に多く，1980 年代以来，日本は中国でのオフショア開発が大発展となる．しかし，近年，中国の人力費などのコストが上がっているため，発注割合が下り，ベトナムへの発注が多くなっている．

本論文では，著者が勤務経験を積んだ中国の会社を主な対象としている，日中間のオフショア開発に存在している問題点を研究する．

2.1.2 オフショア開発のモデル

1) 情報システム開発モデル

情報システム開発におけて，さまざまな開発モデルを定義している．ソフトウェア品質知識体系ガイド V2(第 2 版)では，以下の開発モデルを挙げている [7].

①ウォーターフォールモデル

ウォーターフォールモデルは，要求定義に始まり，分析，設計，実装，試験，運用に至る体系的な逐次型のアプローチである．

②反復型開発プロセス

反復型開発プロセスは，小さい機能単位に動作可能なソフトウェアの開発を反復することにより，ソフトウェアを完成させる方法である．

③プロトタイピング

プロトタイピングとは，最終的に求めるソフトウェアのプロトタイプの早期作成を通じて，合目的性の高い形で要求分析や設計上の決定を進める開発方法である．プロトタイプとは，実働するモデルや模型である．

④スパイラルモデル

スパイラルモデルとは，スパイラルに開発を繰り返すことにより，少しずつソフトウェアの定義と実装を拡大，詳細化する開発方法である．

2) 情報システムのオフショア開発モデル

日本から中国へのオフショア開発において用いられる開発方法は，そのほとんどがウォーターフォール型である．この開発方式は，プロジェクト全体をいくつかの工程に分割し，各工程での成果物に基づいて後工程の作業を順次行っていく開発モデルである [8].

日本企業はエンドユーザーと打ち合わせ，情報システムの要求を分析し，基本設計を行い，海外でシステムのプログラミング作業を実施する．日本のオフショア開発の流れでは，上流工程の企業は下流工程の企業に書類を用いて指示を与えることが必要である．従って，このような開発プロセスはプログラミングと主なオフショア開発工程に有効になる．

図 2 はある企業がウォーターフォールモデルに基づくオフショア開発を行う方法および役割分担を示す．

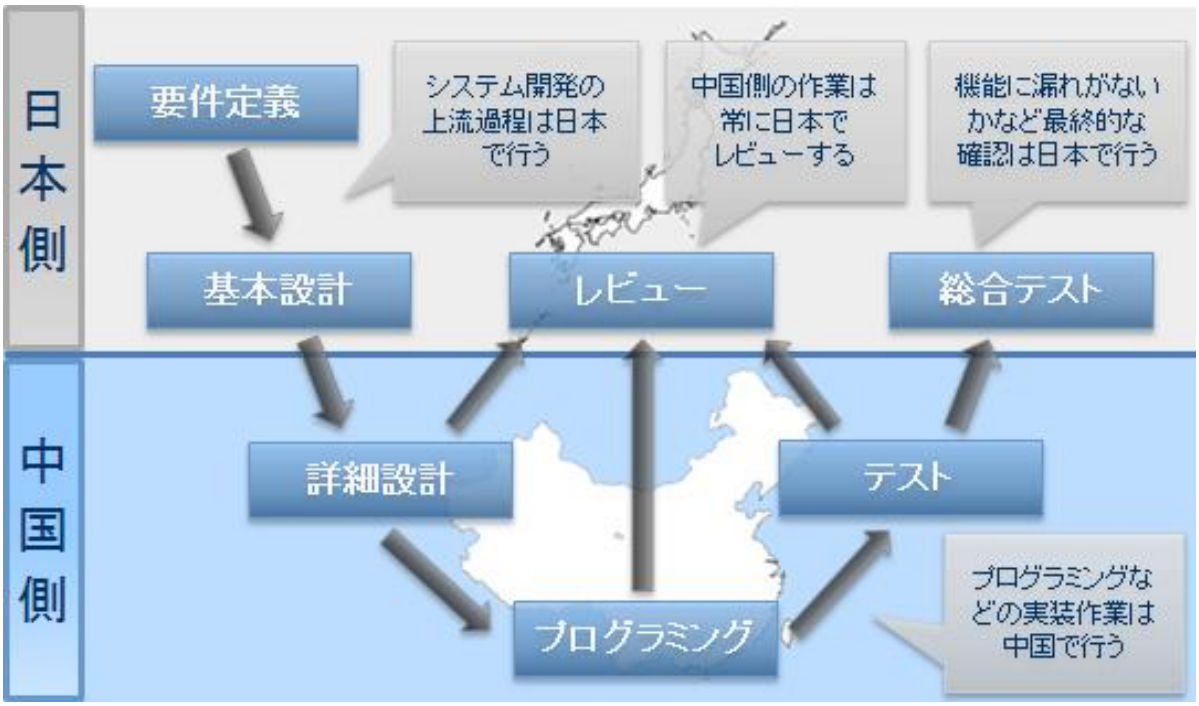


図 2．日本のオフショア開発のプロセス一例(引用：参考文献 [9])

2.1.3　オフショア開発の体制

情報システムのオフショア開発の際には，発注側(日本企業)と受注側(海外ソフトウェア企業)が一つプロジェクトを協同で開発する．図 3 に示すような，一般的なオフショア開発プロジェクト体制では，受注側は発注側からシステムの要求を受けて，情報システムのプログラミングを行っている．この体制で受注側の海外会社は，自分の立場から情報システムの開発プロジェクトにおけるリスクを特定し，マネジメントする．

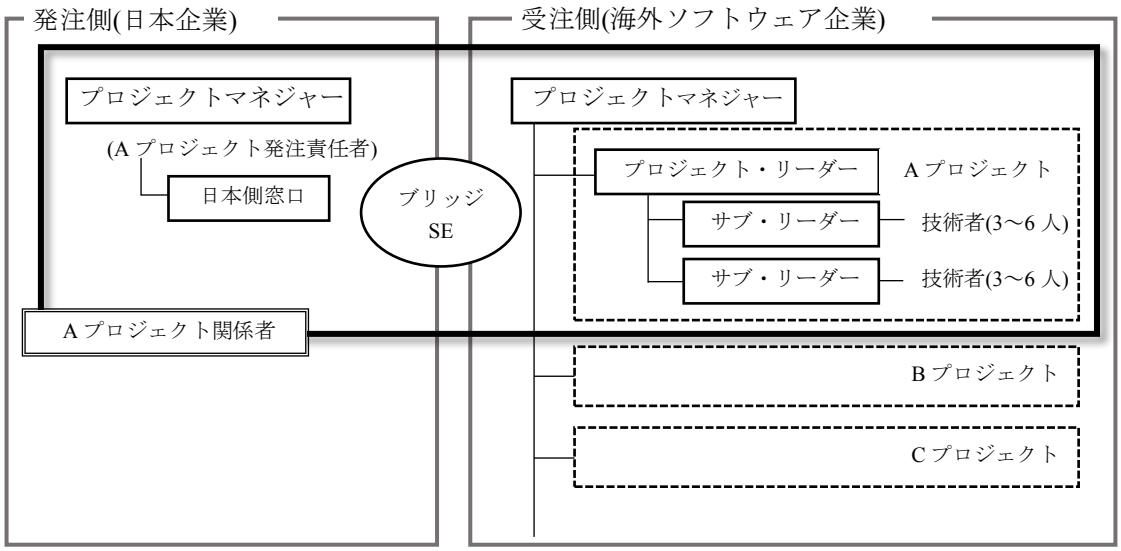


図 3．一般的なオフショア開発のプロジェクト体制

(出典：ソフトウェア開発オフショアリング完全ガイド [5])

情報システムのオフショア開発の体制において，発注側は依頼人として，情報システムの製造工程に参加せずに代理人とする受注側に委任する．Stephan Manning はプリンシパル＝エージェント理論(principal-agent theory)の角度から分析して，多くのオフショア開発におけるリスクがエージェンシー問題(agency problem)により発生したと考えている [10].

2.2 情報システムのオフショア開発におけるリスク

2.2.1 リスクの定義及びその分類

1) リスクの定義

まず，リスクの定義について，情報システム開発に限らずに一般的なリスクについて述べる．

リスクの定義については識者によってさまざまな定義がなされている．リスクの特徴からリスクを分類する方法はその一つである．例えば，純粋リスクと投機リスクを分けている [11].

純粋リスク(pure risk)とは，自然災害，事故などによる損害賠償のように損害や損失しか与えないリスクのことで，意志に関係なく被るリスクである．

投機的なリスク(speculative risk)とは，新たなリターンを目指して，新製品開発や新しい市場の開拓などを行うことで発生するリスクのことであり，成功してリターンを得る可能性と失敗して損害を被る両側面があるリスクである．

実務のプロジェクトをマネジメントするためには，適切な知識，プロセスなどが必要ある．国際標準のガイドブックでは，その実務に必要な知識を特定した．

情報システム開発の場合, PMBOK，SEWBOK(Guide to the Software Engineering Body of Knowledge, ソフトウェアエンジニアリング基礎知識体系)と CMMI は主なマネジメントガイドであるが，リスクに対する認識が異なっている．

PMBOK では，リスクは「もし発生すれば，プロジェクト目標にプラスあるいはマイナスの影響を及ぼす，不確実な事象あるいは状態」と定義される [12]．ここで，プロジェクトのマネジメントで利益も損失も得る可能性があるため，リスクの定義は「投機的リスク」という概念を採用している．

CMMI では，具体的なリスクの定義がないが，リスクマネジメント(RSKM)の目的が説明されている．CMMI の説明によれば，「リスクマネジメントの目的は，潜在的な問題が顕在化する前にその問題を特定することである. これにより, 目標達成の妨げとなるような影響を軽減するために, 成果物またはプロジェクトの全期間にわたって, 必要に応じてリスク取り扱いの活動が計画され, 開始される」 [13]．ここで,「目標達成の妨げるとなるような影響」はプロジェクトのリスクを認識することで,「純粋リスク」という概念を採用している．

本章では情報システムのオフショア開発を行う場合の情報システムの開発プロジェクトを研究対象として，論述する．特に経理・会計などアプリケーション Web システム(Browser/Server)の開発プロジェクトを対象とする．

2)　リスクの分類

情報システムのオフショア開発におけるリスクマネジメントを準備する一環として，リスクを洗い出す作業が必要である.その作業を行う際に，まずリスクを分類する必要がある.

リスクの分類について，一般的な情報システム開発プロジェクトのリスクが存在している一方，オフショア開発における独自のリスクも生じている．その原因は，情報システムのオフショア開発プロジェクトに 2 つ組織(発注側と受注側)が存在していることである．この 2 つ組織は違う国や地域にあるので，言葉，政治及び文化環境に大きな相違点が存在している．プロジェクトを評価する時，組織文化によって，評価結果が違っている状況がある．例えば，日中間の情報システムのオフショア開発の体制では，日本企業が上流工程を実施する一方，中国企業が下流工程の仕事を執り行っている．日本企業は情報システムのユーザーと直接接触し，ユーザーのニーズを満たしていることをプロジェクトの目的とする．中国企業としては，成功したプロジェクトは情報システムの要件を完成し，ソフトウェアのコード品質を満たしていることである [14].

リスクを分類する時，この一般的な情報システムの開発リスクとオフショア開発リスク 2 つ項目を考慮しなければならない.

2.2.2　一般的な情報システム開発プロジェクトのリスク

情報システム開発に存在しているリスクは，企業やプロジェクトによって認識が異なり，リスクを洗い出した結果が違う．田島の研究により，5 カテゴリ 34 種類の典型的なリスクを列挙し表 1 に示す [15].

表 1. 情報システム開発におけるリスク

顧客	1	仕様変更や仕様追加
	2	パッケージを使うといった仕様の確認がない
	3	顧客との仕様検討遅れ
	4	顧客内のコミュニケーション不足
	5	顧客とのコミュニケーション不足
	6	ドキュメントの取り決めが曖昧
	7	顧客予算が確保できなかった(交渉力不足)
	8	保守契約があいまい(ハードウェアやパッケージソフトも含める)
	9	顧客体制の変化
	10	顧客の開発の遅れ
	11	顧客のキーパーソンがいない
	12	協力会社・他社(顧客)のソフト品質不良
	13	検収条件が曖昧
技術	14	開発部門の管理スキル不足
	15	パッケージや既存ソフトの相性・カスタマイズ性
	16	見積りミス
	17	性能条件が不明確
	18	再利用不足
	19	ハード・OS の選定ミス
	20	PC 性能不足
管理	21	リソースの不足(スキル，人数，予算など)
	22	管理能力不足
	23	開発手順の設定ミス
	24	自社要員・協力会社・顧客・PL などのモラールの問題
	25	著作権(他社侵害を含む)
	26	ウイルス
	27	要員のけが(交代要員)
	28	ハードの発注遅れ・納期遅れ
	29	OS・パッケージのバージョンアップ
協力会社 資源購入	30	協力会社のスキル不足
	31	ハード故障・パッケージソフトや OS 故障
	32	協力会社の開発遅れ
	33	協力会社の品質不良
その他	34	天災

2.2.3 オフショア開発のリスク

S-open オフショア開発研究会[注1]はオフショア開発の具体例から情報システムのオフショア開発中の問題点をまとめて，次のように分類した [5].

① 仕様伝達・変更時の問題
② 言語，コミュニケーションの問題
③ 発注案件，発注作業範囲など調達全般に関する問題
④ プロジェクト管理問題
⑤ 品質確保に関する問題
⑥ 体制，人材に関する問題
⑦ インフラ，開発環境に関する問題

S-open オフショア開発研究会では，以上の 7 つの問題が存在していることから，オフショア開発での失敗が起きると論じている．しかし，オフショア開発の場合，二つ以上の組織が協同でプロジェクトを進めることから，企業は問題点の重要性に対する感じ方が違う．日本のオフショア開発の場合，日本企業と海外企業では感じる問題点が違う．その問題点を次の表 2 と表 3 に示す [5].

表 2. 日本企業が感じるオフショア開発の問題点

順位	問題点	比率(%)
1	コミュニケーション	18%
2	仕様伝達・変更	13%
2	海外発注のオーバーヘッド	13%
4	品質	10%
5	開発プロセスの差異	9%
6	受注側の技術力	7%
6	海外発注の仕組み	7%
8	異文化理解	6%
9	受注側の経営安定性	4%
9	受注側のエンジニアの離職率	4%
11	受注側のインフラ	3%
12	機密漏洩	1%
12	日本行政の仕組み	1%
	そのほか	4%

注 1 S-open:ソフトウェア技術者ネットワーク ; Software Professional Engineer’s Network.
「オフショア開発研究会」は S-open の擁するユニークな研究会の一つである.

表 3. 海外企業が感じるオフショア開発の問題点

順位	問題点	比率(%)
1	仕様確定・変更	29%
2	コミュニケーション	17%
3	日本独特の要求	11%
3	技術力	11%
5	計画合意	8%
6	スケジュール	6%
6	日本側の体制	6%
8	異文化理解	4%
8	品質	4%
8	開発中の問題対応	4%

2.2.4 情報システムのオフショア開発におけるリスク

田島とS-openオフショア開発研究会は各自の研究分野で情報システム開発におけるリスクをまとめた(2.2.2と2.2.3)．田島の研究では，情報システム開発を対象として，オフショア開発する場合のリスクが含まれていない(2.2.2)．S-open オフショア開発研究会はオフショア開発に注目しリスクをまとめたが，情報システム開発における本来のリスクに対する整理が行なわれていない(2.2.3)．それでも，両方の研究では接点や共通点があると考えられる．

まず，コミュニケーションの問題が一番重要な問題である．田島の研究では，「顧客との仕様検討遅れ」，「顧客内のコミュニケーション不足」及び「顧客とのコミュニケーション不足」が全て「コミュニケーション能力」という問題点と示している．同時に，S-open オフショア開発研究会の研究では，日本企業が「コミュニケーション」という問題点を感じるが18%になっており，17%の海外企業も「コミュニケーション」に問題を感じている．

次は，田島とS-openオフショア開発研究会の研究では「仕様変更」と「仕様追加」の問題点が存在し，S-openオフショア開発研究会の研究で高い順位になっている．日本企業から発注の場合，仕様が明確でなく開発することが多い，海外企業ではその問題がよく指摘されている．この原因は日本の独特の商習慣で要求定義が遅れ，仕様変更の多発に起因する [5].

更に，ソフトウェアの品質，開発コスト，プロジェクト管理の能力及び開発要員管理の問題点は田島とS-openオフショア開発研究会の研究で明らかになった．例えば，田島の研究では「要員のけが(交代要員)」リスクを挙げ，S-openの研究では「受注側のエンジニアの離職率」を挙げている．開発要員の交代の原因が多くて，けがだけではなく，転職の可能性を日本以外の国では軽視できない原因であると考えられる．特に中国では社員の転職率が日本より高く，開発要員の定着度は共通のリスクである．

2012年9月，筆者は中国のオフショア企業を訪問し，実務の実態や課題調査を実施した．その中，LIANDI社(下記はL社)は中国オフショア成長型企業TOP100強に入選し，代表的なオフショア開発会社である．もう一社でハイロン社(下記はH社)は新創立会社であり，急発展のオフショア開発会社である．

L社は「国家重点ソフトウェア企業」に認定されている．江蘇省最大のオフショア開発企業であり，CMMIレベル4のソフトウェアプロセス管理及び規範を実施している．現在，会社の従業員は1150人以上である．L社で第三システム事業部部長と面談し，開発現場を見学した．

H社は，近年に某大手サービス・アウトソーシング企業から投資し，江蘇省で新創立したオフショア開発会社である．創立から3年間経て，従業員は1000人に達し，年間売上も連続増加している．H社で社長および主な開発責任者と面談し，社内見学をした．

本研究では，L 社と H 社に対するヒアリング結果により，以上の 2 つ研究を参考し，情報システムのオフショア開発における主な 12 件のプロジェクト・リスクを以下の三種類に分類して表 4 に示し，研究を行う．

① 技術類
② プロジェクト類
③ 受注側類

表 4．情報システムオフショア開発リスク

分類	No.	リスク項目
技術類	1	システムの複雑さ
	2	性能条件の明確
	3	仕様確定と仕様変更
	4	開発部門の管理スキル
プロジェクト類	5	スケジュールの厳しさ
	6	コストの妥当性
	7	開発中の問題対応
	8	受注側の技術力
受注側類	9	受注側のコミュニケーション能力
	10	開発環境の整備
	11	受注側開発要員の定着度
	12	経営安定性

2.3　情報システムのオフショア開発におけるリスクマネジメント

2.3.1　主なリスクマネジメント手法

国際標準また日本標準となる著作ではリスクマネジメント手法がプロジェクトマネジメントの全体のプロセスをガイドしている．主な 3 つのリスクマネジメントが重要な管理手法として利用されている．

1)　PMBOK(Project Management Body of Knowledge，プロジェクトマネジメント知識体系ガイド)のリスクマネジメント

PMBOK ガイドはプロジェクトマネジメントの国際標準として，広い範囲のプロジェクトで利用されており，プロジェクトマネジメントの専門家に最もよく使われる文書の 1 つである．PMBOK ガイドはプロセスベース体系として，第 5 版では 47 個のプロセスを，幅広いプロジェクトに適用可能な 5 個の基本的なプロセス群と 10 個の知識エリアとに分類する．その中の 10 個の知識エリアとは次の通り．

① プロジェクト統合マネジメント
② プロジェクト・スコープ・マネジメント
③ プロジェクト・タイム・マネジメント
④ プロジェクト・コスト・マネジメント
⑤ プロジェクト品質マネジメント
⑥ プロジェクト人的資源マネジメント
⑦ プロジェクト・コミュニケーション・マネジメント
⑧ プロジェクト・リスクマネジメント
⑨ プロジェクト調達マネジメント
⑩ プロジェクト・ステークホルダー・マネジメント

PMBOK でのリスク定義により，マネジメント次第でプラスの影響にもマイナスの影響にもなり得る．よって，プロジェクトにプラスの影響を及ぼす可能性があるものは，できるだけ利益になるように努め，マイナスの影響を及ぼす可能性があるものはうまく対応して影響を軽減することをリスク・マネジメントのテーマとして取り組んでいる．PMBOK のリスクマネジメントは下記のような手順で示す.

- リスクマネジメント計画
- リスク識別
- 定性的リスク分析
- 定量的リスク分析
- リスク対応計画
- リスクの監視・コントロール

2) SWEBOK(Guide to the Software Engineering Body of Knowledge，ソフトウェアエンジニアリング基礎知識体系)のリスクマネジメント

ソフトウェアエンジニアリングはソフトウェアの開発，運用，および保守に関する，体系化され，実践規律化され，かつ定量可能化された手法である．すなわち，エンジニアリングのソフトウェアに対する適用である [16].

SWEBOK の目的は，一般的な認められている知識体の各部を組織化して示し，それに含まれるトピックを利用するための手段，すなわちアクセス(access)を提供することにある [16]. 2013 年の SWEBOK V3.0 版では，知識領域を追加，変更し，次の 15 個領域になった.

① ソフトウェア要求
② ソフトウェア設計
③ ソフトウェア構築
④ ソフトウェアテスティング
⑤ ソフトウェア保守
⑥ ソフトウェア構成管理
⑦ ソフトウェアエンジニアリング・マネジメント
⑧ ソフトウェアエンジニアリングプロセス
⑨ ソフトウェアエンジニアリングモデルおよび方法
⑩ ソフトウェア品質
⑪ ソフトウェアエンジニアリング専門技術者実践規律
⑫ ソフトウェアエンジニアリング経済学
⑬ 計算基礎
⑭ 数学基礎
⑮ エンジニアリング基礎

領域のソフトウェアエンジニアリング・マネジメントの中に，リスクマネジメントは下記の手順によるに示す.

- リスク因子の同定
- 各リスク因子の確率とそれがもたらし得る影響の分析
- リスク因子の優先度をつける
- リスク因子が問題となった場合の否定的影響の確率を低減，それを最小化するためのリスク回避を，戦略として開発するという負担を課す.

3) CMMI(Capability Maturity Model Integration，能力成熟度モデル統合)のリスクマネジメント

品質，生産性の向上，工期短縮などのニーズを満たすためにソフトウェアプロジェクトの活動をより高度なレベルに高めるため，米国国防総省(DoD)の出資のもとに，カーネギーメロン大学の

ソフトウェアエンジニアリング研究所(SEI)が能力成熟度モデル(CMM: Capability Maturity Model)を開発した.

CMM は当初ソフトウェアの開発を対象に開発され, これをソフトウェア CMM と呼んでいる. CMM にはこれ以外に，システムエンジニアリグ CMM，システム調達 CMM など，いろいろなモデルが開発された．これらのモデルはそれぞれ似て非なるところがあり，使いづらいことからこれらを統合したモデルの開発が求められた．それが CMM 統合モデル，すなわち CMMI である.

CMMI では，組織のプロセスの発展段階を 5 段階の成熟度レベルでモデル化している．成熟度レベルは，最初にプロジェクトレベルでプロジェクト管理の基礎を達成することからはじまり，定性的データ，定量的データの両方を使用して意思決定を行い，最終的には組織全体にわたる継続的な改善へと進む段階的な改善経路を提供している．表 5 にレベルと特性を示す.

表 5．成熟度と組織の特性

成熟度レベル		特性
レベル１	「初期」	ソフトウェアプロセスは場当たり的で無秩序である．通常，組織はプロセスを支援するための安定した環境を提供しない. 成功は組織に属する人員の力量や英雄的行為に依存する.
レベル２	「管理された」	プロセスは，方針に従って計画され実施され，制御された出力を作成するためにプロジェクトが必要十分な資源を持つ熟練した人員を利用し，直接の利害関係者を関与させ，監視され制御されかつレビューされ，そしてプロセス記述に対する忠実さが評価される.
レベル３	「定義された」	プロセスは，特性が十分に明確化され理解され，そして標準，手順，ツール，および手法の中で記述される．成熟度レベル 3 の基盤となる「組織の標準プロセス群の集合」が確立され，時間の経過とともに改善される.
レベル４	「定量的に管理された」	ソフトウェアプロセスおよび成果物品質に関する詳細な計測結果が収集されている．ソフトウェアプロセスも成果物も，定量的に理解され制御される.
レベル５	「最適化している」	革新的なアイディアや技術の試行, およびプロセスからの定量的フィードバックによって，継続的なプロセス改善が可能になっている.

2010 年に公開された CMMI 1.3 版では，3 つの関連要素に分類している．それぞれは調達ための CMMI(CMMI-ACQ), 開発ための CMMI(CMMI-DEV)とサービスための CMMI(CMMI-SVC)である.

CMMI-ACQ は IT 調達プロセスにおいて IT 調達を決定するために必要な情報を提供する.

CMMI-DEV は開発プロセスにおける管理，評価，および監視するためのガイドを与える.

CMMI-SVC は外部顧客や組織内にサービスを供給するためのガイドを提供する.

情報システムのオフショア開発の際に，CMMI は発注側と受注側が図 4 のような共通のプロセスモデルを提供している.

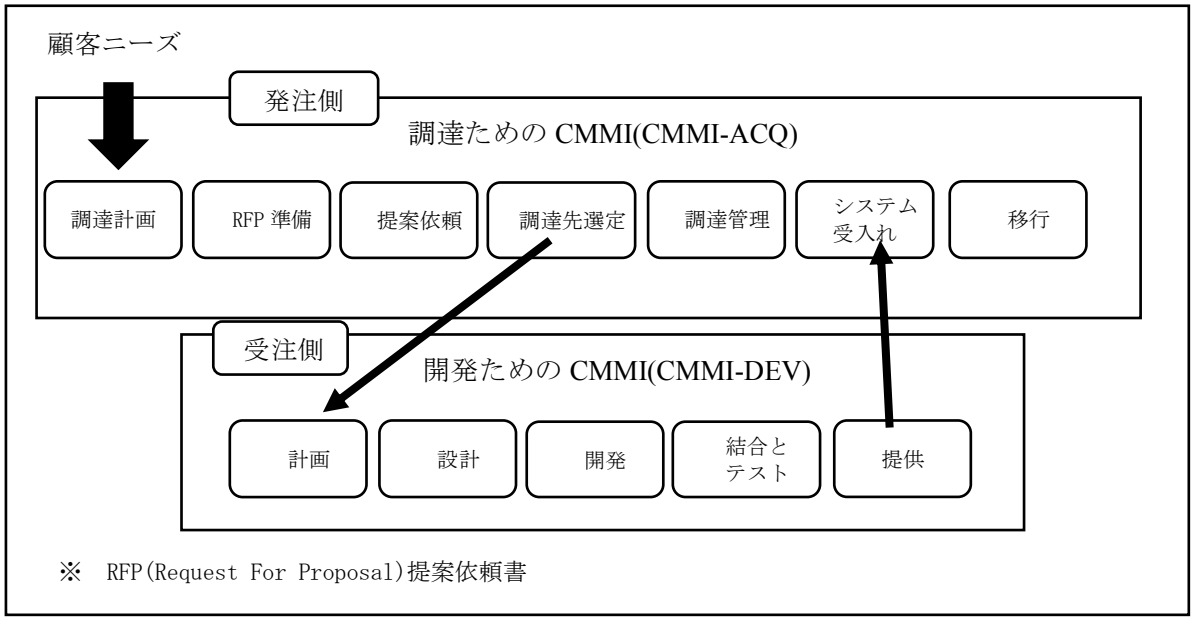


図 4. 調達と開発の CMMI モデル

多くのオフショア開発を行っている受注側の海外情報システム開発企業が CMMI-DEV の認証を取得している．また，開発中のリスクをマネジメントする際に，マネジメントの実施者は受注側であり，発注側がコスト上の原因で実施しないので，本研究で，CMMI-DEV のリスク概念とリスクマネジメント手法に基づき，マネジメント手法を研究した.

2.3.2 CMMI-DEV リスクマネジメントのプロセス

CMMI-DEV のリスクマネジメント・プロセス領域ではリスクマネジメントが 3 つの部分に分けられている.それぞれ下記の固有ゴールで規制された [13].

① リスクマネジメントの準備をする
② リスクを特定し分析
③ リスクを軽減する

固有ゴールを達成するため，CMMI-DEV モデルは固有プラクティスを記述する.

CMMI-DEV の用語説明 [13]によると，固有ゴールは必要とされる CMMI-DEV モデル構成要素であり，プロセス領域を満たすために示されねばならない独特な特性を記述したものである．固有プラクティスは期待されるモデル構成要素であり，関連づけられた固有ゴールを達成するために重要であると見なされるものある.

固有プラクティスは，プロセス領域の固有ゴールの達成につながることが期待される活動を記述している．図 5 はリスクマネジメント領域の固有ゴールと固有プラクティスの関係及びリスクマネジメントのプロセスを示す. 図 5 の「SG」は固有ゴール(Specific Goals)で，「SP」は固有プラクティス(Specific Practices)である．

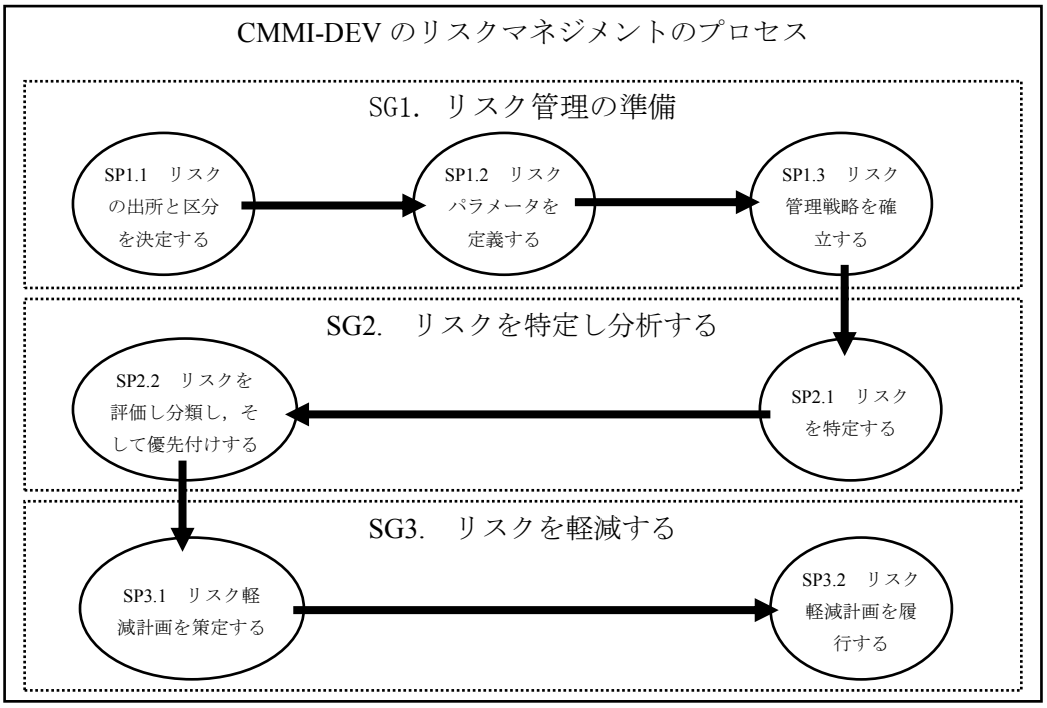


図 5. CMMI-DEV のリスクマネジメント・プロセス (参考文献 [13]による作成)

2.3.3 リスク値の算出方法

プロジェクトへの有害な影響を低減するため，リスクが取り扱われ軽減されている．定義されたリスク値を超えた場合のリスク取り扱いの活動を実施する．

情報システム開発プロジェクトにおけて，複数のリスク項目が存在している．本論文では，複数のリスク項目がプロジェクトのリスク集合 $R=\{x_i \mid i=1,2,3\ldots n\}$を定義する．$R$ はプロジェクトのリスク集合で，x は各リスク項目である．

あるリスク項目 x_i に対して，リスク値は $R(x_i)$で表記する．また，リスク発生確率を $P(x_i)$,リスク影響度を $E(x_i)$と定義する．

リスク値を算出する際にはリスクの発生確率とリスクの影響度を考慮し確定する. Boehm(1991) [17]により，リスクの発生確率 P とリスクの影響度 L からリスク値をかけ算で確定する．本論文では，リスク項目のリスク値は下記の式で算出する．

$$R(x_i)=E(x_i)\times P(x_i) \qquad (1)$$

2.4 情報システムのオフショア開発におけるリスクマネジメント上の問題点

情報システムのオフショア開発を実施する際に，発注側は受注側企業のソフトウェアの開発能力を重視している. CMMI はソフトウェア開発プロセス改善ための評価基準となっており, CMMI の認証を持っている会社はソフトウェア開発上の品質などが信用できると言われる．よって，日本の発注企業は海外の受注会社を選ぶ際に，CMMI 認証を持っていることが重要な評価基準となっている．つまり，CMMI の認証は情報システムオフショア開発の受注側が取る必要資格の一種である.

しかし，CMMI 認証を持っていても，オフショア開発において，リスクマネジメントをする際に，解決しにくい問題点が存在している．それは海外の受注側がどんなプロジェクトを提供したら，発注側がこのプロジェクトに高い評価をするかである.

2.4.1 CMMI-DEV のリスク影響度の評価

リスクが顕在化した時の影響の大きさをリスクの影響度という. CMMI−DEV により, 影響度は, 一般に，費用，スケジュール，環境の影響，または人間的な尺度 (例えば，損失労働時間，怪我の重大度) に関連する [13]. CMMI−DEV では，評価の結果は，多くの場合，三つから五つの値を持つ基準を使用しており，影響度の尺度により決めた区分が「低，中，高」や「無視できる，ささいな，重要な，危機的な，破滅的な」の例を示している.

影響度を評価するために，事前に影響度の尺度を定義する必要がある．CMMI−DEV では具体的な技法を提供していないので，PMBOK での尺度定義方法を下記の表 6 で示す.

表 6. リスク影響度定義

主要なプロジェクト目標に対するリスクの影響度の尺度とその条件の定義					
プロジェクト目標	相対的な尺度または数値尺度				
	非常に低い/0.05	低い/0.10	普通/0.20	高い/0.40	非常に高い/0.80
コスト	軽微なコスト増大	10%未満のコスト増大	10−20%のコスト増大	20-40%のコスト増大	40%超のコスト増大
タイム	軽微な期間延長	5%未満の期間延長	5-10%未満の期間延長	10-20%未満の期間延長	20%超の期間延長
スコープ	軽微なスコープ縮小	スコープへの影響は限定	スコープへの影響は広範	スポンサーが許容しないスコープ縮小	プロジェクトの最終成果物は実用に耐えない
品質	軽微な品質低下	非常に高い要求事項にのみ影響	スポンサーの承認を必要とする品質低下	スポンサーが許容しない品質低下	プロジェクトの最終成果物は実用に耐えない
4 つの異なるプロジェクト目標に対するリスク影響度の定義の例. それらの値は，リスク・マネジメント計画プロセスにおいて，個々のプロジェクト及び組織のリスク限界値に合わせて調整する必要がある．好機に対する影響度の定義も同様の方法で作成される.					

個々のプロジェクト目標に対する影響度のレベルは，インタビューや会議において評価する.

2.4.2 オフショア開発におけるリスク評価の問題点

CMMI-DEV のリスク影響度の評価方法から見ると，CMMI の評価方法は完全にシステム開発プロジェクトの目標を予想通り完成するため，リスク影響度尺度を定義し，影響度を評価し，そしてリスクの閾値を決めてリスクの優先順位をつける．このリスク分析方法には開発を実施する受注側がプロジェクトの具体的な状況によりリスク影響度の区分を定量的な定義する．例のプロジェクトのコスト目標が 2000 万円であれば，10%未満(200 万円以下)のコストを増大すると「低い」のリスクを定義し，数値 0.10 で表示できる．

しかし，既存のリスク影響度尺度を定義する方法によりリスクマネジメントにおける問題点がある．Eisenhardt はプリンシパル＝エージェント理論に基づき研究し，受注側(エージェント)が発注側(プリンシパル)の利益ために労務を実施すべきであるが，情報システムのオフショア開発におけて，発注側と受注側が各自の利益を持っているので，プロジェクトの目標に対する認識が異なっている [18]． 従って，情報システムのオフショア開発におけて，発注側と受注側は各自の利益や立場から考えると，リスクに対する分析の結果が異なる可能性が高い．この相違点の存在はリスク影響度の尺度を定義する際に，発注側と受注側の両方は尺度に対する争議が起こりやすい．

また，発注側は実際の情報システムのオフショア開発プロジェクトに対する評価において，単一なリスクではなく，同時に複数リスクを考慮しなければならない．発注側はプロジェクトにおけるあるリスクが低いことを前提とする場合，ほかのリスクがある程度高くなっても許容することがある．表 6 の例として，「10%未満のコスト増大」というリスクは「低い」また「0.10」の尺度を定義している．但し，別のリスク「軽微な期間延長」と確定したら，発注側は「10%未満のコスト増大」のリスクは「非常に低い/0.05」と認める可能性がある．従って，オフショア開発プロジェクトに対する評価結果は，各リスクの総合的な影響を受けることである．

即ち，問題点としては，オフショア開発プロジェクトのリスクを評価する際に，リスクの大きさは，リスク影響度とリスク発生確率により求めるが，オフショア開発プロジェクトにおける各リスク項目の影響度を定量的な数値で表すことが難しい．

CMMI-DEV のリスクマネジメント手法プロセスと評価方法を使用しても，海外の受注側は日本の発注側がオフショア開発プロジェクトに対する高い評価を取るという目標を妨害するリスクの影響度尺度を定量化し確定することが難しいと思われる．

2.5 まとめ

本章では，情報システムのオフショア開発において存在しているリスクの定義や種類を考察した．これらのリスクの解釈は曖昧で，ガイドによって，採用されている概念が異なっている．研究のテーマと解決したい問題点により，本章で取り扱うリスクを明確にした．

情報システムの開発において，オフショア開発プロジェクトのリスクマネジメントに存在している問題点がある．

オフショア開発プロジェクトにおけるリスクを評価する際に，単一のリスクではなく，複数のリスクを総合的に考慮しにくいため，各リスクはプロジェクトの全体にどんな程度の影響を及ぼしているか数値化することが難しい．

問題点として，オフショア開発プロジェクトにおける複数のリスク影響度を定量的な数値で表すことが難しい．そのため，海外の受注側は日本の発注側からオフショア開発プロジェクトに対する高い評価を取りたいが，CMMI でリスクマネジメントを行う際に，正しいリスクをコントロールできない可能性がある．

従って，この問題点を解決するために，次章(第 3 章)でコンジョイント分析方法を CMMI リスクマネジメントのプロセスに追加する提案を行う．

第3章 情報システムのオフショア開発プロジェクト・リスク分析手法

本論文の第 2 章で述べたオフショア開発の問題点として，オフショア開発プロジェクトのリスクを評価する際に，リスクの大きさは，リスク影響度とリスク発生確率により求めるが，オフショア開発プロジェクトにおける複数のリスク影響度を定量的な数値で表すことが難しい.

本章では上記のオフショア開発の問題点を解決するために，コンジョイント分析を用いたプロジェクト・リスク分析を研究する.

3.1 コンジョイント分析の考察

3.1.1 コンジョイント分析とは

コンジョイント分析とは，多変量解析の一つ手法である.商品やサービスの持つ複数の特徴について，ユーザーはどの点を重視しているかを探すことができる.市場調査の研究では，各製品やサービスを構成するさまざまな属性(例の製品の特徴，機能，価格など)を決める方法として，コンジョイント分析がよく使用されている．現在，コンジョイント分析はマーケティング，製品管理およびオペレーションズリサーチの研究を含む社会科学分野で適用されている.

製品の研究開発での利用において，消費者の製品全体に対する評価から製品の構成要素の好ましさ(選好度)を推定でき，企業は消費者の選好度が高い製品を生産することができる．例えば，消費者はあるコーヒーメーカーを購入するときに「少し価格が高いけれども加熱が速いので購入しよう」とか,「容量が少ないけれども価格が安いので購入しよう」という意思決定をしたことがある．コンジョイント分析を用いて，消費者に具体的な製品サンプルを複数提示し，その価格や性能などを示した製品サンプルを評価してもらうことで，製品の構成要素，例えばコーヒーメーカーの価格や容量の好ましさを推定する.

3.1.2 コンジョイント分析の流れ

製品に対するコンジョイント分析を行う際には，次のような流れで行う.

まず，分析の対象としたい価格・機能・性能・ブランド・デザインといった商品の構成要素(これを属性と呼ぶ)とその属性の具体的なレベル(具体的な価格やデザイン案など．これを水準と呼ぶ)を特定化しなければならない.

次に，属性・水準表をもとに複数の製品プロファイルを作成し，調査対象者にこれらの製品プロファイルを評価してもらう．コンジョイント分析では，商品やサービスは「プロフィール(Profiles)」を呼ぶ.

最後の分析段階で，消費者の製品の各属性に対する好ましさが分析出来る.

3.2　コンジョイント分析によるリスク分析

コンジョイント分析は製品やサービスに対する分析の場合によく使うが，この節でコンジョイント分析によりリスク分析方法を考察する．

3.2.1　コンジョイント分析によりリスク分析のメリット

コンジョイント分析を利用して，リスクを分析することにより，下記のメリットがあると思っている．

まず，既存のリスク影響度尺度の定義方法と違って，コンジョイント分析方法は受注側が自己の利益や立場から考えなく，発注側がプロジェクトの目標に対する評価から分析し，定量的な尺度を定義できるため，尺度の説得力が高くなると考えている．

次に，コンジョイント分析により得たリスク影響度はプロジェクト全体の目標への影響を表れる．既存のリスク影響度尺度の定義方法はプロジェクト全体の目標ではなく，プロジェクトのコストや納期など目標への影響を分けて定義する方法である．この方法は各リスクの総合的な影響を受けることを考えずに，複数のリスク影響度を定量的な数値で表すことができない．コンジョイント分析では，効用値(utility)がプロジェクト全体の目標への影響度を表れる．

つまり，コンジョイント分析を分析ツールとする用いて，発注側がプロジェクト全体の目標に対する評価から各リスクがプロジェクト全体の目標への影響度を分析でき，問題点とするプロジェクトにおける複数のリスク影響度を定量的な数値で表すことを解決することが可能である．

3.2.2　コンジョイント分析によるリスク分析の流れ

本研究では，情報システムのオフショア開発における各リスク項目を分析する際に，コンジョイント分析の流れは下記の通りである．

① プロジェクトのリスク要因とリスク項目を特定する．
② 複数の評価用プロジェクトのプロファイル(Profiles)を作成する．
③ 調査対象者にこれらのプロジェクトのプロファイルを評価してもらう．
④ それぞれのリスク項目が回答者に与える効用を数値化したもの(効用値と呼ばれる)を表する．
⑤ その効用値を元に各リスク要因の組み合わせプロジェクトの選好程度を予測できる．

コンジョイント分析では，効用値などの得点を偏りなく測定するために，現実にないような組合せ要素(一般製品として,価格や機能などである.本研究では各リスク要因である)を評価させる．上記のプロフィールとは評価させるため，各リスク要因から組合せる現実にないような情報システムのオフショア開発プロジェクトである．

次の項から上記の流れによって，コンジョイント分析によりリスク分析方法を説明する．

3.2.3 プロジェクトのリスク要因とリスク項目を特定する

説明が便利にするため，オフショア開発リスクの分類により，リスク要因は①技術類②プロジェクト類③受注側類 3 つの区分に分けられる.

それぞれの情報システムのオフショア開発プロジェクトの各リスク要因はどんな具体的なリスク項目を持っているかを特定する必要である.

そのリスク分類ごとにリスク要因とリスク項目を表 7～表 9 で示す.

表 7. 技術類

	リスク要因	リスク項目 (L	M	H)
1	システムの複雑さ	単純	普通	複雑
2	性能条件の明確	明確	主な機能が明確	明確できない
3	仕様確定と仕様変更	変更･追加が少ない	変更・追加がやや多い	変更・追加が多い
4	開発部門の管理スキル	スキルが高い	平均水準スキル	スキルが低い

表 8. プロジェクト類

	リスク要因	リスク項目 (L	M	H)
5	スケジュールの厳しさ	厳しくない	軽微な期間延長	厳しい
6	コストの妥当性	予算内	軽微なコスト増大	40%超コスト
7	開発中の問題対応	柔軟である	対応できる	柔軟でない
8	受注側の技術力	技術力高い	平均水準技術力	技術力低い

表 9. 受注側類

	リスク要因	リスク項目 (L	M	H)
9	受注側のコミュニケーション能力	能力が高い	教育で満足	能力が低い
10	開発環境の整備	整備済み	必要程度の整備済み	整備不完備
11	受注側開発要員の定着度	定着している	交代要員がいる	離職率が高い
12	経営安定性	安定	予期内安定	不安定

3.2.4　プロジェクトのプロフィール作成

各リスク項目はプロジェクトにどの程度の影響を与えるかを調べるためにアンケート調査が必要である.しかし，本研究でプロジェクトは 3 種類属性を持つ，12 項目のリスク項目を設定する.全面的な調査をすれば，負担になる.コンジョイント分析は全ての組合せを調査しなくても，表 10 のような直交配列表を利用し組合せを作ることで実現する.

1)　直交配列実験とは

通常，多くの因子を同時に取り上げて実験する必要がある．このような場合すべての組み合わせで実験を行うため，実験回数が非常に多くなる．これに対して直交配列表による実験では少ない回数で多くの要因効果を検定できるように工夫されている.

直交配列表には，主な 2 水準の因子を取り扱う 2 水準系直交配列表，主な 3 水準の因子を取り扱う 3 水準系直交配列表がある.

2)　直交配列表による組合せ表の作成

本研究でまとめたリスク要因は 12 個で，それぞれ要因は 3 項目(高中低 3 つ水準のリスク項目)の値を与える．全部の組み合わせは 3^{12}＝531441 である．つまり，全てのプロフィールは 531441 であるから，全部評価すれば，大変な作業である．直交配列実験法で，直交配列表を作り，一部の組み合わせ，あるいは一部の代表的なプロフィールは全ての実験を代わりに評価する.

本論文のリスクは 3 リスク項目 12 リスク要因で，直交配列表 $L_{27}(3^{12})$を選択する．その中，L は直交配列表という意味である．27 はこの表に 27 行があり，27 個組み合わせを示す．3 は 3 リスク項目であり，12 はリスク要因数である．この表は 3 リスク項目 15 リスク要因の実験を決める.

本論文では，IBM SPSS Statistics 22(以下 SPSS と略記)でプロフィールを作って，表 10 で表している(ID の P は Project の略称である)．SPSS で直交設計のプログラムが下記に示す.

```
ORTHOPLAN
  /FACTORS=  システムの複雑さ 'システムの複雑さ' (1 'L' 2 'M' 3'H')
             性能条件の明確 '性能条件の明確' (1 'L' 2 'M' 3'H')
             仕様確定と仕様変更 '仕様確定と仕様変更' (1 'L' 2 'M' 3'H')
             開発部門の管理スキル '開発部門の管理スキル' (1 'L' 2 'M' 3'H')
             スケジュールの厳しさ 'スケジュールの厳しさ' (1 'L' 2 'M' 3'H')
             コストの妥当性 'コストの妥当性' (1 'L' 2 'M' 3'H')
             開発中の問題対応 '開発中の問題対応' (1 'L' 2 'M' 3'H')
             発注先の技術力 '発注先の技術力' (1 'L' 2 'M' 3'H')
             発注先のコミュニケーション能力
                  '発注先のコミュニケーション能力' (1 'L' 2 'M' 3'H')
             開発環境の整備 '開発環境の整備' (1 'L' 2 'M' 3'H')
             発注先開発要員の定着度 '発注先開発要員の定着度' (1 'L' 2 'M' 3'H')
             経営安定性 '経営安定性' (1 'L' 2 'M' 3'H')
/OUTFILE='D:¥SPSS¥paper¥project_plan.sav'.
```

SPSS は，初心者からプロまでが利用する，統計解析のスタンダード・ツールである．社会調査の統計解析ツールとして 1968 年にスタンフォード大学で生まれた SPSS は，その後 40 年以上，さらなる進化を遂げてきた [19]．SPSS は重回帰分析，ロジスティック回帰分析，生存分析，共分散構造分析，因子分析，主成分分析，一般線型モデル，ノンパラメトリック検定，信頼性分析，決定木分析など 30 種類以上の豊富な分析メニューを持っている [20].

表 10．プロジェクトのリスク項目の組合せ

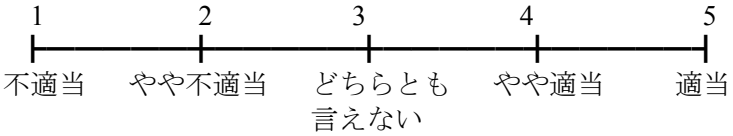
ID	システムの複雑さ	性能条件の明確	仕様確定と仕様変更	開発部門の管理スキル	スケジュールの厳しさ	コストの妥当性	開発中の問題対応	受注側の技術力	受注側のコミュニケーション能力	開発環境の整備	受注側開発要員定着度	経営安定性
P1	L	L	L	L	L	L	L	L	L	L	L	L
P2	M	M	L	L	M	M	M	L	L	H	H	H
P3	H	H	H	L	L	L	M	H	H	M	H	L
P4	M	M	M	L	L	L	H	M	M	H	M	L
P5	M	H	M	H	M	L	L	L	H	L	M	M
P6	L	M	M	H	L	H	H	L	H	M	H	H
P7	L	H	H	M	L	M	M	L	M	H	M	M
P8	H	M	L	M	M	H	M	M	H	L	M	L
P9	L	M	L	H	M	L	M	H	M	M	L	M
P10	L	L	M	L	H	H	M	M	M	L	H	M
P11	L	L	H	L	M	M	H	H	H	L	M	H
P12	H	M	M	M	L	M	H	H	L	L	L	M
P13	H	M	H	M	H	L	L	L	M	L	H	H
P14	M	H	H	H	L	H	M	M	L	L	L	H
P15	H	L	L	H	L	H	L	H	M	H	M	H
P16	M	L	M	M	H	L	M	H	L	M	M	H
P17	L	H	L	M	H	L	H	M	H	H	L	H
P18	L	H	M	M	M	H	L	H	L	H	H	L
P19	H	L	M	H	H	M	M	L	H	H	L	L
P20	L	M	H	H	H	M	L	M	L	M	M	L
P21	M	H	L	H	H	M	H	H	M	L	H	L
P22	H	H	L	L	H	H	H	L	L	M	M	M
P23	M	L	H	M	M	H	H	L	M	M	L	L
P24	H	H	M	L	M	M	L	M	M	M	L	H
P25	M	L	L	M	L	M	L	M	H	M	H	M
P26	M	M	H	L	H	H	L	H	H	H	L	M
P27	H	L	H	H	M	L	H	M	L	H	H	M

表 10 の L(Low)は，表 7～表 9 の低水準のリスク項目を示し，M(Middle)は，表 7～表 9 の中水準のリスク項目を示し，H(Height)は表 7～表 9 の高水準のリスク項目を示す.

3.2.5 プロファイルを評価してもらう

作成したオフショア開発プロジェクトのプロフィールは計 27 件である.各プロフィールに対して適当かどうかの調査を行う.

発注側は評価を行う際に，各評価用プロジェクトの評価を 5 段階とした.発注側はプロジェクトのプロフィールに対する評価は，1～5(1 は不適当，5 は適当)まで採点する.具体的な尺度は下記に示す.



発注側がプロフィール P1 に対する評価の調査票(コンジョイント分析でカードと呼び)の例は次の図 6 に示す．すべての調査票は付録に記載する.

プロジェクト P1 の評価

P1 の各項目の状況は次のように示している，P1 の全体に対して，評価してください

No.	リスク項目	リスク項目
1	システムの複雑さ	単純
2	性能条件の明確	明確
3	仕様確定と仕様変更	変更・追加が少ない
4	開発部門の管理スキル	スキルが高い
5	スケジュールの厳しさ	厳しくない
6	コストの妥当性	予算内
7	開発中の問題対応	柔軟である
8	受注側の技術力	技術力高い
9	受注側のコミュニケーション能力	能力が高い
10	開発環境の整備	整備済み
11	受注側開発要員の定着度	定着している
12	経営安定性	安定

P1 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

図 6. プロフィール P1 に対する評価調査仕様

3.2.6　リスク項目に対する効用値の導出

このステップでは，各リスク項目の効用値がオフショア開発プロジェクト全体の目標への影響，つまり，リスク影響度を表せる．

コンジョイント分析では全体評価が各要素と関係がある．この関係は重回帰モデルで表す．本論文では，情報システムのオフショア開発プロジェクト全体の目標への評価が各リスク項目との関係が，下記の重回帰モデルで表す．

$$U = \mathrm{C} + \beta_1 * X_1 + \beta_2 * X_2 + \ldots\ldots + \beta_{12} * X_{12} \quad (2)$$

ここで，Uはプロジェクトのプロフィールに対する評価の値である．この値は前項(3.2.5)で評価から得た数値であり，発注側がプロジェクト全体の目標に対する評価である．

X_i(i=1,2,3…12)は重回帰モデルでダミー変数(dummy variable)と呼び，本論文でまとめた情報システムのオフショア開発におけるリスクである．例えば，表 10 のプロフィールにより X_1 はリスク項目「普通のシステムの複雑さ」と表す．組み合わせの表では，リスク要因が「L」,「M」,「H」3 つのリスク項目を設定したが，コンジョイント分析をするため，「L」「M」「H」のリスク項目を数値化し，それぞれ「1」「2」「3」になる．つまり，例の「普通のシステムの複雑さ」が「H」のリスク項目「複雑」である場合，X_1 の値は 3 になる(X_1＝3).

β_i(i=1,2,3...12)は重回帰モデルで回帰係数(regression coefficient)を呼び，各ダミー変数が全体評価への影響を示すため，コンジョイント分析での効用値である．本論文のリスク分析方法では，この効用値は各リスク項目が情報システムのオフショア開発プロジェクト全体の目標への評価に対して，どの程度の影響があるかを示す．

つまり，効用値は各リスク項目がプロジェクト全体への影響が数値で示すことできる．本論文では，効用値を用いて，リスク値を確定するため，CMMI−DEV のリスクマネジメントのプロセスにコンジョイント分析をツールとして導入することを考慮し，既存プロセスを改善する．

本論文で，E 社[注 2]からのアンケートに基づきシミュレーションデータを自分で作成した．下記の表 11 で示している評価点数のシミュレーションデータを利用し，SPSS でコンジョイント分析を実施する方法を説明する．

[注 2] E 社は日中間情報システムのオフショア開発会社(本社・東京)である．ISMS 認証取得.

表 11. プロフィールに対する評価の点数(ID1~5)

ID	システムの複雑さ	性能条件の明確	仕様確定と仕様変更	開発部門の管理スキル	スケジュールの厳しさ	コストの妥当性	開発中の問題対応	発注先の技術力	発注先のコミュニケーション能力	開発環境の整備	発注先開発要員の定着度	経営安定性	評価1	評価2	評価3	評価4	評価5
1	単純	明確	変更・追加が少ない	スキルが高い	厳しくない	予算内	柔軟である	技術力高い	能力が高い	整備済み	定着している	安定	5	4	5	4	4
2	普通	主な機能が明確	変更・追加が少ない	スキルが高い	軽微な期間延長	軽微なコスト増大	対応できる	技術力高い	能力が高い	整備不完備	離職率が高い	不安定	3	3	4	3	4
3	複雑	明確できない	変更・追加が多い	スキルが高い	厳しくない	予算内	対応できる	技術力低い	能力が低い	必要程度の整備済み	離職率が高い	安定	2	1	2	2	1
4	普通	主な機能が明確	変更・追加がやや多い	スキルが高い	厳しくない	予算内	柔軟でない	平均水準技術力	教育で満足	整備不完備	交代要員がいる	安定	3	2	3	2	2
5	普通	明確できない	変更・追加がやや多い	スキルが低い	軽微な期間延長	予算内	柔軟である	技術力高い	能力が低い	整備済み	交代要員がいる	予期内安定	3	3	4	3	4

表 11 プロフィールに対する評価の点数(ID6~11)

6	単純	主な機能が明確	変更・追加がやや多い	スキルが低い	厳しくない	40%超コスト	柔軟でない	技術力高い	能力が低い	必要程度の整備済み	離職率が高い	不安定	2	2	3	3	2
7	単純	明確できない	変更・追加が多い	平均水準スキル	厳しくない	軽微なコスト増大	対応できる	技術力高い	教育で満足	整備不完備	交代要員がいる	予期内安定	4	5	4	4	4
8	複雑	主な機能が明確	変更・追加が少ない	平均水準スキル	軽微な期間延長	40%超コスト	対応できる	平均水準技術力	能力が低い	整備済み	交代要員がいる	安定	4	3	3	3	4
9	単純	主な機能が明確	変更・追加が少ない	スキルが低い	軽微な期間延長	予算内	対応できる	技術力低い	教育で満足	必要程度の整備済み	定着している	予期内安定	5	4	5	5	5
10	単純	明確	変更・追加がやや多い	スキルが高い	厳しい	40%超コスト	対応できる	平均水準技術力	教育で満足	整備済み	離職率が高い	予期内安定	2	1	2	1	2
11	単純	明確	変更・追加が多い	スキルが高い	軽微な期間延長	軽微なコスト増大	柔軟でない	技術力低い	能力が低い	整備済み	交代要員がいる	不安定	4	2	4	3	4

表 11 プロフィールに対する評価の点数(ID12~17)

12	複雑	主な機能が明確	変更・追加がやや多い	平均水準スキル	厳しくない	軽微なコスト増大	柔軟でない	技術力低い	能力が高い	整備済み	定着している	予期内安定	2	1	2	2	2
13	複雑	主な機能が明確	変更・追加が多い	平均水準スキル	厳しい	予算内	柔軟である	技術力高い	教育で満足	整備済み	離職率が高い	不安定	2	1	1	2	2
14	普通	明確できない	変更・追加が多い	スキルが低い	厳しくない	40%超コスト	対応できる	平均水準技術力	能力が高い	整備済み	定着している	不安定	1	2	2	2	1
15	複雑	明確	変更・追加が少ない	スキルが低い	厳しくない	40%超コスト	柔軟である	技術力低い	教育で満足	整備不完備	交代要員がいる	不安定	3	2	3	3	4
16	普通	明確	変更・追加がやや多い	平均水準スキル	厳しい	予算内	対応できる	技術力低い	能力が高い	必要程度の整備済み	交代要員がいる	不安定	3	3	4	3	4
17	単純	明確できない	変更・追加が少ない	平均水準スキル	厳しい	予算内	柔軟でない	平均水準技術力	能力が低い	整備不完備	定着している	不安定	4	4	5	4	4

表 11 プロフィールに対する評価の点数(ID18~23)

18	単純	明確できない	変更・追加がやや多い	平均水準スキル	軽微な期間延長	40%超コスト	柔軟である	技術力低い	能力が高い	整備不完備	離職率が高い	安定	1	1	2	2	2
19	複雑	明確	変更・追加がやや多い	スキルが低い	厳しい	軽微なコスト増大	対応できる	技術力高い	能力が低い	整備不完備	定着している	安定	4	3	4	3	3
20	単純	主な機能が明確	変更・追加が多い	スキルが低い	厳しい	軽微なコスト増大	柔軟である	平均水準技術力	能力が高い	必要程度の整備済み	交代要員がいる	安定	4	3	4	4	3
21	普通	明確できない	変更・追加が少ない	スキルが低い	厳しい	軽微なコスト増大	柔軟でない	技術力低い	教育で満足	整備済み	離職率が高い	安定	3	3	4	3	4
22	複雑	明確できない	変更・追加が少ない	スキルが高い	厳しい	40%超コスト	柔軟でない	技術力高い	能力が高い	必要程度の整備済み	交代要員がいる	予期内安定	3	2	2	3	3
23	普通	明確	変更・追加が多い	平均水準スキル	軽微な期間延長	40%超コスト	柔軟でない	技術力高い	教育で満足	必要程度の整備済み	定着している	安定	3	2	2	2	3

表 11 プロフィールに対する評価の点数(ID24~27)

24	複雑	明確できない	変更・追加がやや多い	スキルが高い	軽微な期間延長	軽微なコスト増大	柔軟である	平均水準技術力	教育で満足	必要程度の整備済み	定着している	不安定	1	2	2	1	2
25	普通	明確	変更・追加が少ない	平均水準スキル	厳しくない	軽微なコスト増大	柔軟である	平均水準技術力	能力が低い	必要程度の整備済み	離職率が高い	予期内安定	2	1	1	2	2
26	普通	主な機能が明確	変更・追加が多い	スキルが高い	厳しい	40%超コスト	柔軟である	技術力低い	能力が低い	整備不完備	定着している	予期内安定	2	1	2	1	1
27	複雑	明確	変更・追加が多い	スキルが低い	軽微な期間延長	予算内	柔軟でない	平均水準技術力	能力が高い	整備不完備	離職率が高い	予期内安定	4	5	4	4	5

コンジョイント分析により，評価点数から各リスク項目がプロジェクトに対する影響を算出できる．この影響はコンジョイント分析で効用値として表示する．ここの効用値は上記の重回帰分析式(2)によって計算することができる．ただし，効用値の計算がかなり複雑であるので，SPSS を利用して計算することが実際の分析で一般な方法である．本論文では，SPSS の Category モデルを利用し，下記のシンタックスを実行した．

まず，集まった点数を入力する．入力するシンタックスは次に示す．

```
DATA LIST FREE /ID score1 to score27.
BEGIN DATA
  1   5 3 2 3 3 2 4 4 5 2 4 2 2 1 3 3 4 1 4 4 3 3 3 1 2 2 4
  2   4 3 1 2 3 2 5 3 4 1 2 1 1 2 2 3 4 1 3 3 3 2 2 2 1 1 5
  3   5 4 2 3 4 3 4 3 5 2 4 2 1 2 3 4 5 2 4 4 4 2 2 2 1 2 4
  4   4 3 2 2 3 3 4 3 5 1 3 2 2 2 3 3 4 2 3 4 3 3 2 1 2 1 4
  5   4 4 1 2 4 2 4 4 5 2 4 2 2 1 4 4 4 2 3 3 4 3 3 2 2 1 5
END DATA.
SAVE OUTFILE='D:¥SPSS¥ paper ¥project_data.sav'.
```

入力データは'D:¥SPSS¥paper¥project_data.sav'に保存した. そして, このデータを使ってコンジョイント分析をする．コンジョイント分析を実施するシンタックスは次に示す

```
CONJOINT PLAN='D:¥SPSS¥paper¥project_plan.sav'
/DATA='D:¥SPSS¥paper¥project_data.sav'
/SCORE=score1 TO score27
/SUBJECT=ID
/FACTORS=システムの複雑さ
        性能条件の明確
        仕様確定と仕様変更
        開発部門の管理スキル
        スケジュールの厳しさ
        コストの妥当性
        開発中の問題対応
        発注先の技術力
        発注先のコミュニケーション能力
        開発環境の整備
        発注先開発要員の定着度
        経営安定性
/PRINT=SUMMARYONLY.
```

PLAN サブコマンドには，プロジェクトの直交計画が入っている project_plan.sav というファイルを指定する.

DATA サブコマンドには，評価のデータが入っている project_data.sav というファイルを指定する.

SPSS でコンジョイント分析を実施し，各リスク項目の効用値を得た.下記の表 12 で示す.

表 12. 全体統計量の効用値

		効用値
システムの複雑さ	L	0.519
	M	-0.281
	H	-0.237
性能条件の明確	L	0.230
	M	-0.081
	H	-0.148
仕様確定と仕様変更	L	0.585
	M	-0.437
	H	-0.148
開発部門の管理スキル	L	-0.348
	M	-0.148
	H	0.496
スケジュールの厳しさ	L	-0.281
	M	0.319
	H	-0.037
コストの妥当性	L	0.519
	M	0.096
	H	-0.615
開発中の問題対応	L	-0.370
	M	0.207
	H	0.163
発注先の技術力	L	0.252
	M	-0.126
	H	-0.126
発注先のコミュニケーション能力	L	0.141
	M	-0.059
	H	-0.081
開発環境の整備	L	-0.104
	M	-0.170
	H	0.274
発注先開発要員の定着度	L	0.007
	M	0.430
	H	-0.437
経営安定性	L	0.074
	M	0.030
	H	-0.104
(定数)		2.837

そのたの重要度値と相関分析の結果は下記の図 7 に示す.

重要度値

システムの複雑さ	10.868
性能条件の明確	5.392
仕様確定と仕様変更	13.042
開発部門の管理スキル	11.161
スケジュールの厳しさ	7.277
コストの妥当性	14.062
開発中の問題対応	7.404
発注先の技術力	6.070
発注先のコミュニケーション能力	4.196
開発環境の整備	6.016
発注先開発要員の定着度	10.713
経営安定性	3.799

平均化された重要度得点

相関分析[a]

	値	有意確率
Pearson の R	.969	.000
Kendall のタウ	.893	.000

a. 観測嗜好値と予測嗜好値の相関

図 7 コンジョイント分析の重要度値と相関分析の結果

表 12 で示している効用値により，各リスク項目はプロジェクト全体への効用値を下記の表 13～表 24 で示す.

表 13. リスク要因「システムの複雑さ」に関するリスク項目の効用値

リスク要因	リスク項目	効用値
システムの複雑さ	L：単純	0.519
	M：普通	-0.281
	H：複雑	-0.237

表 14. リスク要因「性能条件の明確」に関するリスク項目の効用値

リスク要因	リスク項目	効用値
性能条件の明確	L：明確	0.23
	M：主な機能が明確	-0.081
	H：明確できない	-0.148

表 15. リスク要因「仕様確定と仕様変更」に関するリスク項目の効用値

リスク要因	リスク項目	効用値
仕様確定と仕様変更	L：変更・追加が少ない	0.585
	M：変更・追加がやや多い	-0.437
	H：変更・追加が多い	-0.148

表 16. リスク要因「開発部門の管理スキル」に関するリスク項目の効用値

リスク要因	リスク項目	効用値
開発部門の管理スキル	L：スキルが高い	-0.348
	M：平均水準スキル	-0.148
	H：スキルが低い	0.496

表 17. リスク要因「スケジュールの厳しさ」に関するリスク項目の効用値

リスク要因	リスク項目	効用値
スケジュールの厳しさ	L：厳しくない	-0.281
	M：軽微な期間延長	0.319
	H：厳しい	-0.037

表 18. リスク要因「コストの妥当性」に関するリスク項目の効用値

リスク要因	リスク項目	効用値
コストの妥当性	L：予算内	0.519
	M：軽微なコスト増大	0.096
	H：40%超コスト	-0.615

表 19. リスク要因「開発中の問題対応」に関するリスク項目の効用値

リスク要因	リスク項目	効用値
開発中の問題対応	L：柔軟である	-0.370
	M：対応できる	0.207
	H：柔軟でない	0.163

表 20. リスク要因「受注側の技術力」に関するリスク項目の効用値

リスク要因	リスク項目	効用値
受注側の技術力	L：技術力高い	0.252
	M：平均水準技術力	-0.126
	H：技術力低い	-0.126

表 21. リスク要因「受注側のコミュニケーション能力」に関するリスク項目の効用値

リスク要因	リスク項目	効用値
受注側のコミュニケーション能力	L：能力が高い	0.141
	M：教育で満足	-0.059
	H：能力が低い	-0.081

表 22. リスク要因「開発環境の整備」に関するリスク項目の効用値

リスク要因	リスク項目	効用値
開発環境の整備	L：整備済み	-0.104
	M：必要程度の整備済み	-0.170
	H：整備不完備	0.274

表 23. リスク要因「受注側開発要員の定着度」に関するリスク項目の効用値

リスク要因	リスク項目	効用値
受注側開発要員の定着度	L：定着している	0.007
	M：交代要員がいる	0.430
	H：離職率が高い	-0.437

表 24. リスク要因「経営安定性」に関するリスク項目の効用値

リスク要因	リスク項目	効用値
経営安定性	L：安定	0.074
	M：予期内安定	0.030
	H：不安定	-0.104

3.2.7　効用値の特徴およびリスク値の確立

リスク値を算出するにあたり，CMMI や PMBOK により，リスク値 $R(x_i)$は式(1)を使用して算出する．このとき，各リスク項目 x_i に対して，リスク影響度 $E(x_i)$とリスク発生確率 $P(x_i)$を決める必要がある．影響度と発生確率はインタビューまた会議によって決める．インタビューや会議の参加者として，プロジェクト・チーム・メンバーのほか，可能であればプロジェクト外の専門家を含める．

本論文では, 式(1)によるリスク値の導出のためのリスク発生確率の決定方法を論述しないので, 計算例の提示のために「0.3」「0.5」「0.8」の場合を仮定する．

各リスク項目がプロジェクト全体に与える影響を前項のコンジョイント分析により数値化し，得られた効用値を各リスク項目の影響度として用いる．なお，効用値がマイナスの値を示す場合は，プロジェクトに対して悪影響を与えることを意味する．

コンジョイント分析の効用値をリスク影響度として使用する方法を表 18 リスク要因「コストの妥当性」についてのリスク値導出を説明する．「コストの妥当性」のリスクとは，対象とする情報システムのオフショア開発プロジェクトにおけるコスト増大評価のリスクを意味する．オフショア開発の目的の 1 つはコストの節約であるので，コストを節約できるかどうかはプロジェクトの評価に影響する．計算例における「コストの妥当性」の発生確率は，M（0.5）を使用する．影響度は,表 18 の通り，コンジョイント分析により，「予算内」である場合の「コスト妥当性」の効用値は，プロジェクトに対して 0.519 と求められた.次に「軽微なコスト増大」である場合の効用値は, 0.096 であった.つまり, リスク値 $R(x_i)$は, 式(1)$R(x_i)=E(x_i)\times P(x_i)$ より「軽微なコスト増大」のリスク値は,0.096×0.5 で表される.この場合のリスクはプロジェクトに対してわずかな影響であることを意味する．さらに「40%超コスト」である場合の「コスト妥当性」の効用値は -0.615 であり, プロジェクトに対して大きく悪影響を与えることを意味する.このとき,「コストの妥当性」に関するリスク影響度・発生確率マトリックスは，リスク影響度に効用値を適用することで，式(1)から表 25 のように求められる．

表 25．コストの妥当性に関するリスク影響度・発生確率マトリックス

発生確率 / リスク影響度	0.3	0.5	0.8
予算内：0.519	0.1557	0.2595	0.4152
軽微なコスト増大：0.096	0.0288	0.0480	0.0768
40%超コスト：-0.615	-0.1845	-0.3075	-0.492

3.2.8　リスク対応の優先順位の決定

リスク値を確立して，リスクの受容可能性あるいは非受容可能性，リスクの優先付け，または管理上の処置のきっかけを判断することができる [13].

CMMI−DEV のリスクマネジメント手法により，リスクの優先順位の決定には，明確な基準を使用する．リスクの優先付けは，プロジェクトに最大限の良い影響を与えるように，リスク軽減のための資源を適用する対象として最も効果的な領域を決定するのに役立つ [13].

コンジョイント分析による決めたリスク影響度とリスク確率から確立するリスク値は，ただのマイナスの影響を示すことではなく，リスク要因はプロジェクト全体への良い影響も表す．そのリスク値を用いて，CMMI−DEV のリスクマネジメントプロセス中の SP2.6 でリスク優先順位を付ける．

3.3　情報システムのオフショア開発におけるリスクマネジメントの提案

3.3.1　リスクマネジメントのプロセスに対する改善

本論文では，コンジョイント分析と CMMI-DEV の組み合わせ，情報システムのオフショア開発でのリスクマネジメントのプロセスを改善について提案する．

本論文では，リスクマネジメント SG2(リスクの特定と分析)は，提案モデルの図 8 で示したように，改善することを提案する．モデルは，また CMMI-DEV のリスクマネジメント・プロセスを持っているが, 情報システムのオフショア開発における特定の作業を持っており, SG と SP で CMMI の要件を満たしている．これは，情報システムのオフショア開発のリスクマネジメントための参考価値を持っている．

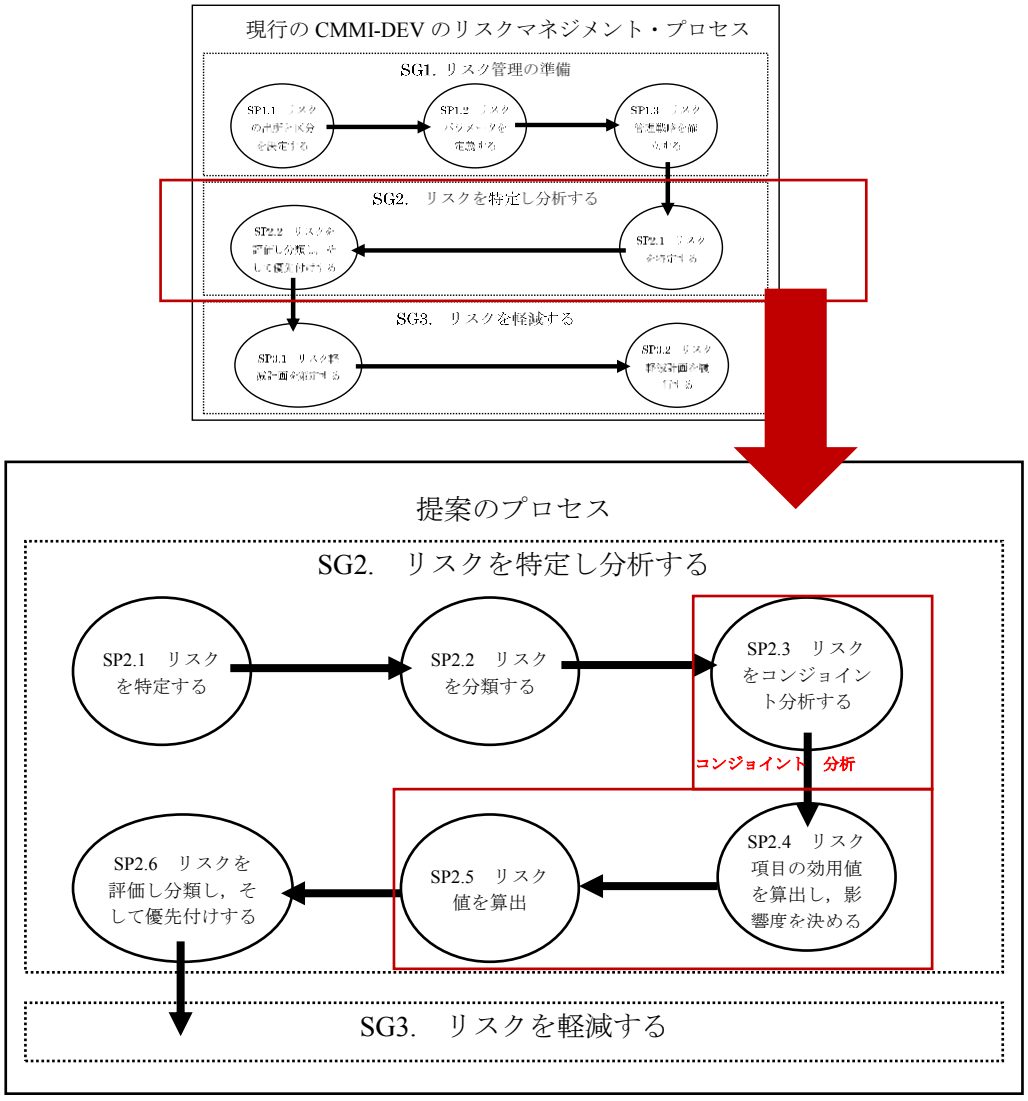


図 8. 提案プロセスモデル

3.3.2　リスクのコンジョイント分析

既存の CMMI−DEV リスクマネジメントのプロセスの SG2(リスクを特定し分析する)においては，コンジョイント分析方法を SP2.3 とする導入した．

コンジョイント分析を実施する前，評価用プロジェクトに対する評価を経て，オフショア開発プロジェクトのリスクに対する評価者は，現在の受注側だけから発注側と受注側の両方になる．

追加したプラクティス SP2.3 では，発注側がプロジェクト全体に対する要望を把握する．情報システムのオフショア開発の場合には，受注側は情報システムを作るので，発注側の製品提供者を見なす．その場合，発注側はプロジェクトの全体に対する評価を通じ，受注側は発注側がプロジェクトの要望をよく理解できる．

提案の実施流れについて，まず，受注側はリスクを特定し次第，リスク項目をコンジョイント分析するために，評価用プロジェクトを組み合わせる．

次に発注側はそれらの評価用プロジェクトを全体評価して，結果を受注側に渡す．受注側は発注側のプロジェクト評価結果に対するコンジョイント分析を実施し，発注側が今回のプロジェクトに対して，どんな要望を持っているかを理解できる．

提案の実施流れは，図 9 で示す．

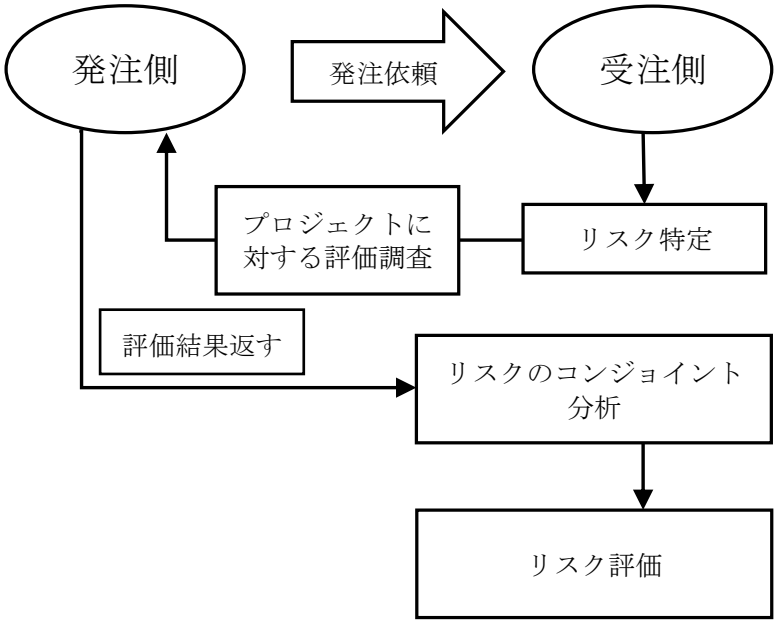


図 9．提案の実施流れ

3.3.3　効用値によりリスク項目の影響度の決める

既存プロセス中には，各リスク項目がプロジェクトのコストや納期など目標への影響に基づき影響度を決める．

問題点としては，各リスクの影響が単独で存在せず，相互に依存するリスク影響が存在しており，プロジェクトの全体を考慮していないため，相対的な影響度が変動する可能性がある．コンジョイント分析方法では，事前にプロジェクトの各リスクを組み合わせて，各評価用プロジェクトは全体として発注側に評価させる．発注側は評価する際に，各リスクの相互影響を考慮し評価する．

受注側は発注側のプロジェクトに対する全体評価をコンジョイント分析し，発注側が各リスクに対する重視度，あるいはリスクの影響度を現れる．コンジョイント分析は全体から各リスクの影響度を分析できたから，問題点とする各リスクの相互影響が考慮している．よって，既存の個別に評価する方法により，信用性が高い．

また，コンジョイント分析の結果で効用値は数値で表す．効用値は良い影響がプラスの数値で表示し，悪い影響がマイナスの数値で表示する．受注側はこのような数値により，各リスク項目がプロジェクト全体への影響を判断できる．

既存の影響度の確立手法と比べて，効用値の利用がプロジェクト全体への影響及び良い影響と悪い影響が表示できるので，効用値により影響度を決めることを SG2 のプラクティス SP2.4 として，プロセスに追加する．

3.3.4　リスク値の算出

追加したプラクティス SP2.3 と SP2.4 により，各リスク項目がプロジェクト全体への影響度を決めた．プラクティス SP2.5 で影響度と発生確率によりリスク値を算出し，各リスク項目のリスクマトリックスを作成する．

プラクティス SP2.5 では，リスク値は良い影響と悪い影響の両方が表示する．従って，情報システムのオフショア開発を実施する際に，良い影響(また好機と呼ぶ)に対する「活用」「共有」「強化」「受容」戦略を考えて，悪い影響に対する「回避」「転嫁」「軽減」「受容」戦略を考えられる．次の既存ステップ「リスクを評価し分類し，そして優先付けする」では，算出したリスク値に基づいて，リスクの優先順位を付ける．

本研究の提案として，情報システムのオフショア開発におけるリスクに対する適切な影響度評価の問題に対して，SP2.3～SP2.5 のプロセスを追加することにより解決できることを提案する．SP2.3 の内容は前述の 3.2 に記述した．

3.4 まとめ

本章で，オフショア開発の問題点として，第 1 は海外の受注側は日本の発注企業がプロジェクトに対する要望に対する理解が困難で，リスクを分析・評価する際に発注側の要望からずれることである．第 2 は既存のリスク評価はリスク項目を単独で評価するため，リスク影響度の相互関係を考慮せず，全体のプロジェクトリスク評価が正しく行えないという 2 つの問題点に対し，コンジョイント分析を説明して，CMMI のリスクマネジメント・プロセスに SP2.3～SP2.5 のプロセスを追加し，情報システムのオフショア開発におけるリスクマネジメントのプロセス改善案及びリスク値の定量化評価方法を提案した．

まず，コンジョイント分析の方法は商品生産業界で開発したい製品に対する市場分析の常用の一つ手法である．顧客の製品に対する選好程度を分析し，開発の参考とするための手法であり．オフショア開発の場合，受注側は発注側がプロジェクトに対する要望を分析するため，コンジョイント分析が有効と考えた．

つぎにコンジョイント分析の流れにより，情報システムのオフショア開発プロジェクトのリスクを分析した．

① プロジェクトの属性を考慮し，オフショア開発におけるリスクを分類し，特定した．
② 各リスクの水準を決めて，評価ための評価用プロジェクトの組み合わせを作成した．
③ 各評価用プロジェクトに対して，プロジェクトの適切を評価した．
④ 評価結果により，コンジョイント分析で各リスクの影響度を明確にした．

従って，コンジョイント分析を用いて，リスクの影響度を基づきリスク値評価方法を定量化し，この定量的なリスク影響度を算出するため，コンジョイント分析を CMMI-DEV のリスクマネジメント・プロセスに追加し，組み合わせたリスクマネジメント・プロセス改善案を提案した．

第4章 情報セキュリティリスクとリスクマネジメント

本章では，情報システム開発におけるプロジェクト上の情報セキュリティ管理について研究する．最近，情報セキュリティマネジメントの重要性に対する認識が高くなっており，情報セキュリティに関する研究も増加している．本論文では，情報システム開発におけるプロジェクト上の情報セキュリティに注目し，研究を行う．

4.1 情報セキュリティ及び情報セキュリティリスク

この項では，情報セキュリティ及び情報セキュリティリスクについて，基本概念を述べる．

4.1.1 情報セキュリティの基本

国際標準 ISO/IEC 27000 により，情報セキュリティ(Information security，略称 InfoSec)とは，情報の機密性(Confidentiality)，完全性(Integrity)，及び可用性(Availability)を維持することである．加えて，真正性 (Authenticity)，責任追跡性 (Accountability)，否認防止 (non-Repudiation)，信頼性(Reliability)などのほかの特性にも関与する可能性がある．

情報の機密性，完全性と可用性は，情報セキュリティの 3 要素と呼ばれ，この 3 要素の頭文字を取って「CIA」と言われている. 情報攻撃者は攻撃の手法にかかわらず, 目的を達成するために, この 3 要素を攻撃する．

1) 機密性

機密性とは情報の機密性は許可された者(個人また組織)だけが情報にアクセスできるようにすることである．例えば，ID やパスワードの設定などによって，組織外の者が組織内の情報へアクセスできないようにすることである．攻撃者が情報の機密性を攻撃するためにフィッシング詐欺やトロイを使うなどの手法で ID やパスワードを手に入れることは情報の機密性に対する攻撃である．

2) 完全性

完全性とは, 情報や情報の処理方法が, 正確で完全であるようにすることである. 具体的には, 情報を保存また転送する際に，情報が改竄，損害，遅延及び紛失されないことである．

3) 可用性

可用性とは許可された者が，必要な時に情報や情報資産にアクセスできることを確実にすることである．

可用性に対する攻撃は情報の可用性を妨げることである．例えば，DoS 攻撃(ドスこうげき)(Denial of Service attack)は稼働しているサーバやネットワークなどのリソース(資源)に意図的に

過剰な負荷をかけたり脆弱性をついたりする事でサービスの正常な動作を侵害する攻撃を指し，可用性を侵害する攻撃手法である．

4.1.2　情報資産とリスク，インシデント

1)　情報資産

資産には，不動産や商品など，目に見える資産もあれば，財務情報，人事情報，顧客情報，戦略情報，技術情報などの目に見えない資産もある．これらを情報資産という [21].

情報資産はさまざまな形態で蓄えられている．例えば，ハードウェア，ソフトウェア，ネットワーク，データ，ノウハウなどの形態で個人や組織が持ち続けている．

2)　リスク

プロジェクトをマネジメントする場合のリスクとは，プロジェクトの目的達成に妨害を与えることであるが，情報セキュリティ上のリスクというのは，別の定義がある．

情報セキュリティのリスクとは，情報資産を脅かす内外の要因(情報セキュリティでは脅威という)によって情報資産が損なわれる可能性をいう [21].

3)　インシデント

情報セキュリティのリスクが起こり，実際に情報資産が損なわれてしまった事態をインシデントと呼ぶ．情報セキュリティのリスクの概念は次の図 10 で表現できる．

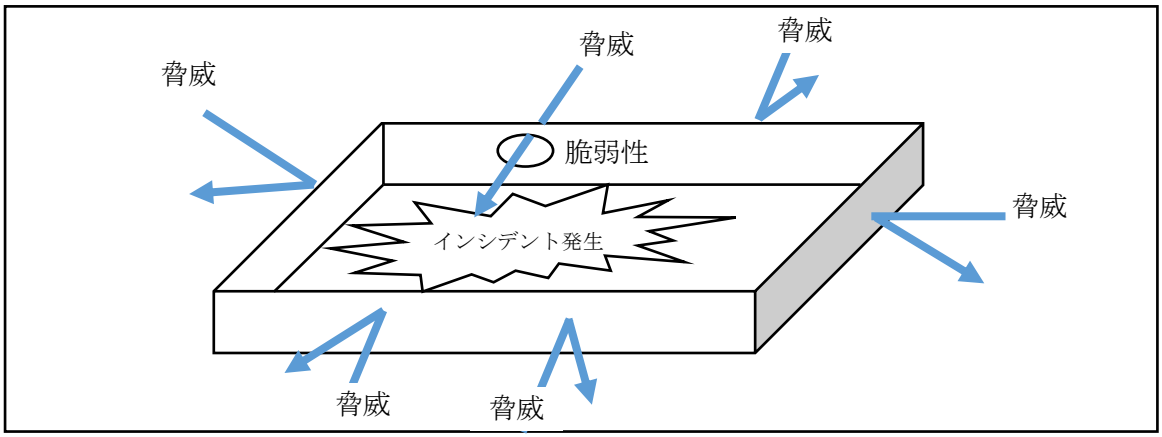


図 10．情報セキュリティのリスク概念　(参考文献 [21])

情報インシデントの発生には，2 つ要因が存在している．一つは脅威であり，もう一つは脆弱性である．脅威とは情報資産に危害を与える原因となるものである．脅威は客観的に存在しているものが，組織や情報システムの内部に，利用できる弱点が存在していなければ，インシデントも発生しない．この内部に存在している弱点を脆弱性と呼ぶ．

4.2 情報セキュリティ・リスクマネジメントの現状

4.2.1 情報セキュリティマネジメントの標準

情報セキュリティマネジメントの重要性への認識が高まり，国際標準化機構 (ISO) と国際電気標準会議 (IEC) が共同で ISO/IEC 27000 シリーズを策定した．一般的な ISMS を導入する組織は 27000～27005 を利用している．27006 は ISMS 認証機関にためのガイドであり，27007 は監査員(組織内外両方)のためのガイドラインである．

ISO/IEC 27000： ISMS 規格についての概要と基本用語集
ISO/IEC 27001： 組織の ISMS を認証するための要求事項
ISO/IEC 27002： ISMS 実践のための規範
ISO/IEC 27003： ISMS 実装ガイド
ISO/IEC 27004： 情報セキュリティの測定
ISO/IEC 27005： 情報セキュリティのリスクマネジメント
ISO/IEC 27007： ISMS 監査の指針(主にマネジメントシステム)

ISO/IEC27001 はその管理策を適切に運用していることを認証するための規格である．これらは，もともとは BS7799 というひとつの規格であった．BS7799 は 1995 年に BSI(British Standards Institution:英国規格協会)により制定された情報セキュリティマネジメントシステムの英国規格で，情報セキュリティ対策を行う際の管理ガイドライン及びガイドラインへの準拠性を認証する仕組みを規定していた [22].

2000 年には BS7799 Part1 が ISO/IEC17799:2000 として採用され，それに伴い英国規格も BS7799-1:2000 として改正された．Part2 はその後，プロセスアプローチ，PDCA サイクル，継続的改善等の考えを盛り込み，BS7799-2:2002 となった．2005 年に BS7799-2:2002 が ISO/IEC27001:2005 として国際標準化された [22].

ISO/IEC 27000 シリーズの各ガイドから ISMS(Information Security Management System，情報セキュリティマネジメントシステム)は以下の図 11 のような構成となる．

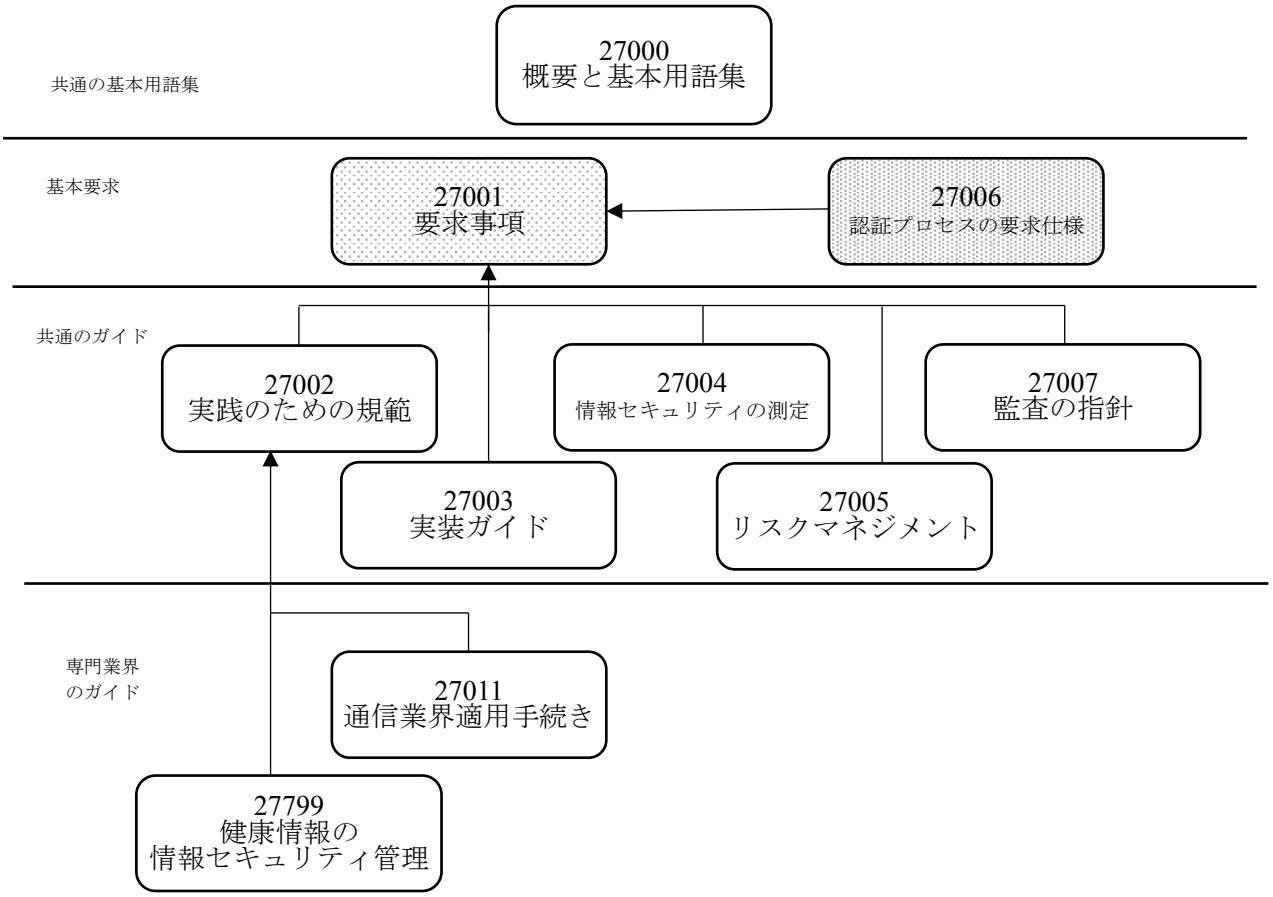


図 11．ISMS 規格群の関係

4.2.2 研究対象

本論文では，情報システム開発プロジェクトにおける情報セキュリティマネジメントを対象とする．その具体的な事例として，E 社の情報システム開発の事業を考察した．E 社は日中間オフショア開発を行って，システム提案から設計,製造,テスト,納品までの総合的なシステム開発サービスを提供している．

E 社のオフショア開発の特徴は一般企業の経理・会計など Browser/Server(B/S)型ウェブアプリケーションシステム開発を行っている．この B/S 型システムの構成は，Web サーバ，データベースサーバ及びそれらをアクセスするためのクライアント(PC やスマホなど)となる．クライアントはウェブブラザーを利用するので，クライアントへのソフトのインストール必要がない．

E 社は ISO27001(ISMS)認証を持っているので，情報セキュリティ対策として，情報システム開発する際に，外部ネットワークを接続する場合には，ブロードバンドルータなどを経由させ，許可する通信だけに限定している．

4.2.3　情報セキュリティマネジメントシステム

情報セキュリティマネジメントシステム(ISMS)とは，組織の情報セキュリティマネジメントの状況を審査し，それを正しく運用している組織を認証する制度である．ISMS を構築する際に，適切な手順によって，確立する必要がある．日本では，一般財団法人日本情報経済社会推進協会(略称 JIPDEC)が組織の ISMS 適合性評価を行っている．次の図 12 で構築手順を示している．

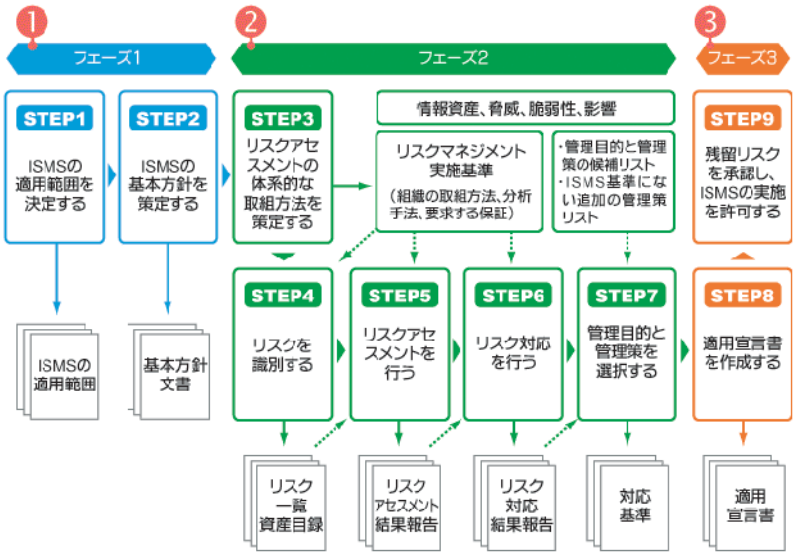


図 12.　ISMS 構築の手順(JIPDEC 資料より引用)

4.2.4　本研究の注目点

本論文の研究では，情報システム開発におけるプロジェクト上の情報セキュリティマネジメントを対象とする．即ち，一般的な組織の情報セキュリティマネジメントの対象と違い，ISMS の適用範囲は情報システム開発プロジェクトである．

次に，本論文では，リスクマネジメントのフェーズでリスクマネジメント手法に関する研究を行う．特徴は次の 3 点にある．

①既存情報システムのセキュリティ評価ではなく，情報システムを新たに開発するプロジェクトにおけるセキュリティ評価である．

②既存情報システムの運用/管理者へのリスク分析ではなく，情報システム開発者/設計者のためのリスク分析である．

③情報セキュリティ・リスクマネジメントの国際標準である ISO27005 : 2011 のプロセスに基づく分析手法である．

4.2.5 情報システム開発におけるセキュリティ

ISO/IEC 27001 : 2013 附属書 A「A.14.2 開発及びサポートプロセスにおけるセキュリティ」により，9 つの管理策が策定された [23]. この管理策は表 26 に示す.

表 26. 開発及びサポートプロセスにおけるセキュリティの管理策

A.14.2 開発及びサポートプロセスにおけるセキュリティ	
目的：情報システムの開発サイクルの中で情報セキュリティを設計し，実施することを確実にするため	
A.14.2.1	セキュリティに配慮した開発ための方針
A.14.2.2	システムの変更管理手順
A.14.2.3	オペレーティングプラットフォーム変更後のアプリケーションの技術的レビュー
A.14.2.4	パッケージソフトウェアの変更に対する制限
A.14.2.5	セキュリティに配慮したシステム構築の原則
A.14.2.6	セキュリティに配慮した開発環境
A.14.2.7	外部委託による開発
A.14.2.8	システムセキュリティ試験
A.14.2.9	システムの受入れ試験

4.2.6　情報セキュリティ・リスクマネジメントのプロセス

ISO27005 により，情報セキュリティ・リスクマネジメントのプロセスは図 13 のように表すことができる [24].

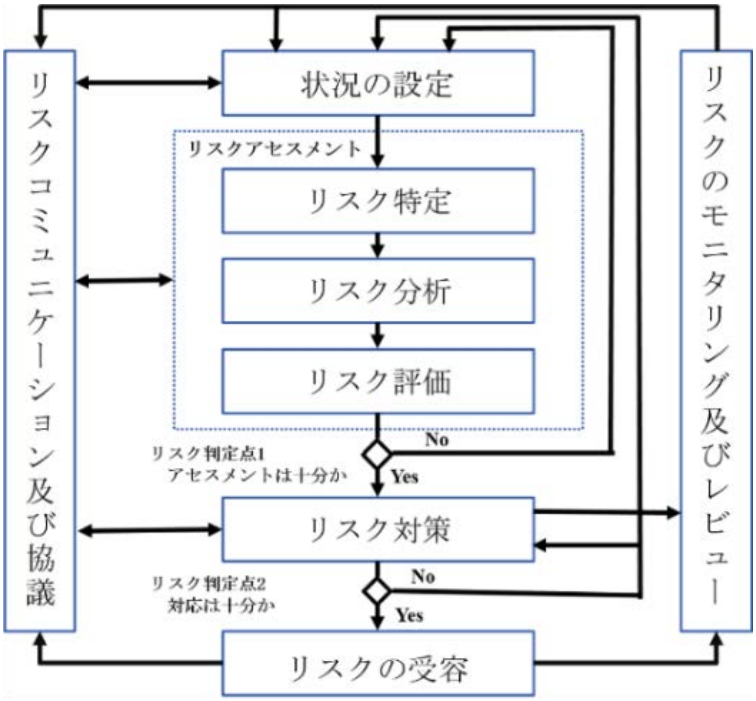


図 13.　リスクマネジメント・プロセス

その中のリスクアセスメントの内容は，図 14 に示すようにリスク特定，分析と評価がある．リスクの特定について，情報セキュリティリスクの特定は他のリスク特定と比べて，独自の要素をもっている．具体的には情報セキュリティの要素で資産，脅威，脆弱性などを特定する必要がある．

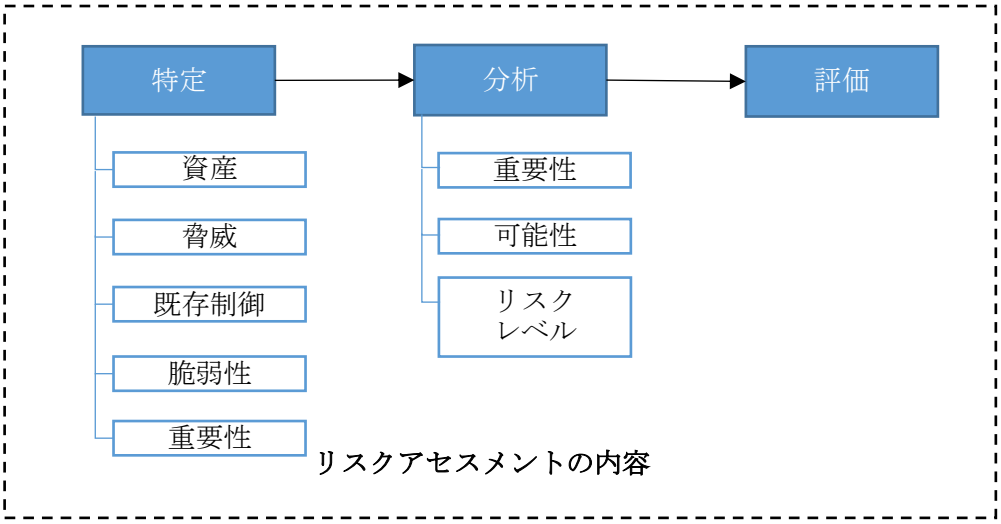


図 14.　リスクアセスメント内容(ISO/IEC　27005 から筆者作成)

4.3　情報セキュリティ・リスクマネジメントの課題

4.3.1　情報システム開発における情報セキュリティリスク

PMBOK などマネジメントガイドでは，リスク分析の主な手法で確率影響マトリックスを記載している．この方法はリスク発生確率と評価したリスク影響値を乗じて，リスク値を決める．しかし，情報セキュリティリスクを分析する場合，分析要因は情報資産，脅威と脆弱性の 3 つであるが，脆弱性と脅威の間に存在している依頼関係や各脆弱性の間の連鎖関係がリスク値に影響を与えるため，既存のマトリックス手法ではリスク分析する際に不十分である．既存リスクアセスメントのプロセスはリスク特定，リスク分析とリスク評価の 3 つ段階である．

1)　段階 1 ：リスク特定

情報セキュリティのリスクマネジメントにおいて，情報資産の安全性，可用性と機密性の維持はマネジメントの中心であるから，脅威と脆弱性の分析は情報資産を保護するために必要である．特定手順は「情報資産を洗い出す」➡「資産に対する脅威を特定する」➡「脅威が利用できる脆弱性を特定する」ということである．

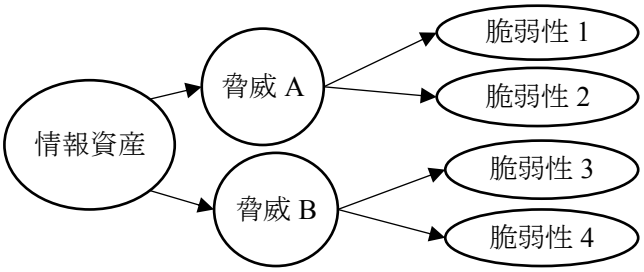


図 15．資産，脅威及び脆弱性の特定手順

2)　段階 2 ：リスク分析

既存のリスク分析手法は情報資産から出発し，ある脆弱性までのルートのリスク値を 1 つずつ計算する．例えば，情報資産から脅威 A を経って，脆弱性 2 までのルートのリスク値の計算は：リスク値＝情報資産値×脅威 A の値×脅威 A の発生頻度値×脆弱性 2 の値という方法である．算出の方法は表 27 の見積もり表で表現する．

表 27．リスク値の見積り表

No	情報資産名	脅威	値	脆弱性	値	発生頻度	値	脅威×脆弱性×発生頻度	リスク値（情報資産値×）
1	機密性 値=3	ハッカー	3	パスワード保護のみ	3	1 回/1 月	4	36	3×36=108
		社員による漏洩	3	個人の自主性	2	1 回/5 年	3	18	3×18=54
		外来者による漏洩	3	協力会社との信頼関係のみ	3	1 回/2 年	3	27	3×27=81

引用：「情報セキュリティのためのリスク分析・評価」 [25]

3)　段階 3：リスク評価

リスク評価はリスク及びその大きさが受容可能か，または許容可能かを決定するために，リスク分析の結果をリスク基準と比較するプロセスである．このプロセスでは，前段階のリスク分析の結果から大きな影響を受ける．

4.3.2　既存の情報セキュリティのリスクマネジメント研究

情報セキュリティのリスクマネジメントに対する研究は，これまで二つ方面に注目している．

一つは企業などの組織の運営・管理をするときのマネジメントに対する研究である．

もう一つは，既存情報システムの脆弱性に対する分析研究である．

例えば，磯部はベイジアンネットワークを利用して，セキュリティリスク評価システムという方法を提案した．この評価システムは，既存情報システムのソフトウェア脆弱性を対象としたリスク評価を行ったものであり，情報システムの運用/管理者の立場からのリスク分析である [26].

この分析方法では，既存情報システムを対象としているため，システムを構成している PC やルータなど各機器に対する脆弱性や情報資産の保有状況，機器間のネットワーク持続性などから脅威と脆弱性の関係をグラフ化している．

本論文では，ベイジアンネットワークのノードは具体的な機器ではなく，情報システム開発の際に，情報セキュリティの脅威・脆弱性および情報資産である．

4.3.3　問題点

上記の分析手法には以下の 2 つ問題点が存在している．

1)　従来手法では各脆弱性間の関係が考えられていない

脅威が複数の脆弱性を原因として，リスクを招いていることが多いが，脆弱性も互いに影響を及ぼす．従来手法で作成された表 27 の見積り表でリスクを分析する方法では各脆弱性間の関係が考えられてないため，相互の影響が表示できない．

2)　脆弱性の相互影響は従来手法で表すことができない

脆弱性は相互の影響が存在しているので，一つ脆弱性に対して対策をすれば，他の関連ある脆弱性の発生確率は変わる．表 27 ではこの変化を表すことができない．

よって，本研究では，連鎖的な因果関係を表現することにすぐれた，ベイジアンネットワーク (Bayesian networks, BN)を利用して，リスク分析における発生確率を決める手法を提案する．

4.4 まとめ

本章では，最初，情報セキュリティの基本概念と情報セキュリティマネジメントに関する標準や管理体制を考察し，説明した．

そして，情報システムの開発において，情報セキュリティのリスクマネジメントに存在している問題点を論述した．

第 1 の問題点として，情報セキュリティの脆弱性を確定する場合，現在は各脆弱性間の関係が考えられてない．

第 2 の問題点として，脆弱性は相互の影響が存在しているので，一つ脆弱性に対して対策をすれば，他の関連ある脆弱性の発生確率は変わる．

従って，次章ではこの二つ問題点を解決するために，ベイジアンネットワークを用いて，現在のリスクマネジメント手法を改善する．

第5章 ベイジアンネットワークによるリスク分析手法の提案

本章では，4 章で明らかになった問題点を解決するためにベイジアンネットワークを利用した情報セキュリティのリスクの分析方法を述べる．ここでベイジアンネットワークについて解説し，情報セキュリティマネジメント中のリスク分析手法での適用をする．適用の研究について，ベイジアンネットワークによりリスク分析モデルの構築を行う．まず，本研究での情報セキュリティリスクは情報システム開発におけるリスクを対象とする．次にモデルの構築手順を論述し，新たなリスク分析手法を提案する．最後に具体的な例を通して，提案の方法を詳説する．

5.1 ベイジアンネットワークについて

5.1.1 ベイジアンネットワークの概念

ベイジアンネットワーク(Bayesian networks，略称 BN)とは，信念ネットワーク(Belief Network)または因果ネットワーク(Causal Networks)として知られている．確率変数間の依存性と独立性の関係を表すことができる知識表現ツールである．

ベイジアンネットワークはアメリカ人の計算機科学者 Judea Pearl が 1985 年に提出した．それは人間が推理する際に不確実の因果関係を処理することを模擬するモデルである．ベイジアンネットワークで用いるグラフは，非循環有向グラフ(Directed Acyclic Graph，略称 DAG)と呼ばれる．

5.1.2 ベイジアンネットワークのノード

ベイジアンネットワークの非循環有向グラフ中の各ノードは確率変数{X_1, X_2, X_3,…, X_n}で表す．因果関係があると見なす確率変数のノード間に矢印で繋がり，一つのノードは因(親ノード，parents)であり，もう一つは果(子ノード，children)である．

例えば，図 16 のようなベイジアンネットワーク中では，X_6 は X_3 と X_4 に依存している．つまり，これらの 2 つの要因から決定されると思える．X_3 と X_4 は X_6 の親ノード，X_6 は X_3 と X_4 の子ノードと呼ばれる．例としては，X_3 と X_4 が既知情報になると，X_6 がこのくらいの確率になるかということを示している．

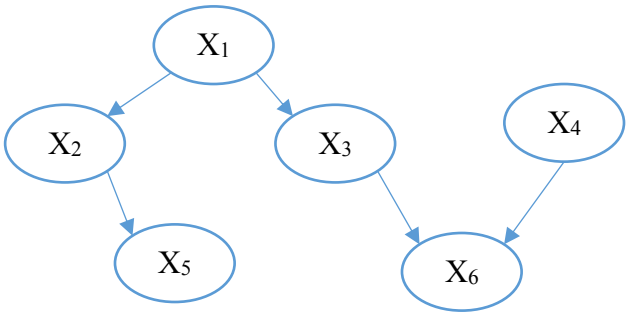


図 16．ベイジアンネットワークの例

5.1.3　ベイジアンネットワークの CPT

各因果を示しているノードの確率は，ベイジアンネットワークで，条件付き確率分布を定義する必要がある．この分布を表現するために，CPT(条件付き確率表，conditional probability table)を使用する．

CPT で示している確率分布による発生確率を計算するため，事前に CPT が与えられる必要があり，これが無いと計算出来ない．親ノードが一つの場合は，親ノードの真偽による発生確率，親ノードが複数の場合は，各親ノードの真偽の全組合せについての発生確率を決める．

本研究の提案では，各ノードの空間状態とは各情報資産の脆弱性に対する対策でありし，対策があるかどうかの 2 値(ある＝Y，なし＝N)となるように定義する．

CPT の各確率値は，主観的に決められた主観確率である．中身の確率値は最初に専門家の評価データまた観察した歴史的なデータから決め，「初期確率値」と言われる．

例えば，図 16 のノード X_3 の CPT は表 28 のように設定する．

表 28．X_3 の CPT の例

X_1	Y	N
Y	0.3	0.7
N	0.6	0.4

この CPT で示している 0.3 という値の意味は，ノード X_3 は親ノード X_1 が発生した(V_1=Y)場合に発生する可能性が 0.3 であるということである．数式で表示すると，$P(X_3=Y \mid X_1=Y) = 0.3$ になる．条件付確率表の各行の和が 1 になることがわかる．

5.1.4　ベイジアンネットワークの応用

ベイジアンネットワークは広い分野で活用されるようになっている．医療診断，データマイニング，イメージと言語認識などが主な活用分野である．予測はベイジアンネットワーク活用の主な役目である．現在，ベイジアンネットワークを利用して，リスク分析手法とする研究も進んでいる．

5.2　提案手法の目的と概要

5.2.1　提案手法の目的

本研究で提案するリスク分析手法は，情報システム開発におけるセキュリティ・リスクマネジメントのリスクを分析することを目的としている．ここで，セキュリティリスクは情報システムに存在している脆弱性により情報資産の損失可能性ではなく，情報システムを開発する際に情報システム製品に関連するリスクである．例えば，情報システム開発に必要な書類，テスト用の顧客情報などである．

情報セキュリティの脆弱性の間には相互影響が存在している．本研究での提案では，その相互影響及び因果関係をグラフで示し，リスク評価する際に分かりやすくすることが目的の一つである．さらに，相互影響でリスクの発生確率の変化を算出し，リスクへの対応を早める可能である．

5.2.2　提案手法の概要

本研究の提案では，次の要素からなる．

① 情報資産リスト
② 脅威リスト
③ 脆弱性リスト
④ 条件付き確率表

①は，情報システム開発組織において想定される情報資産のリストである．本研究では，情報資産を洗い出し，リストを作ることを述べる．

②は，想定した情報資産に対する潜在的な破壊要因である．

③は，情報資産を保護する際に，脅威に対する対策をしない，または不足のリストである．

④は，各脆弱性により異常が発生する確率を評価し，作成される確率表である．この確率表によって，リスクインシデントの発生確率を計算できる．

つまり，①～④の各要素は因果関係が存在し，情報セキュリティのリスク発生確率を決める．本研究での提案は，その要素を分析し，適切なリスク分析する手法である．

5.3　提案手法の手順

提案手法では,必要なリスクデータを洗い出すのが最初の作業である.ここのリスクデータは,情報セキュリティの 3 要素で情報資産，脅威及び脆弱性である．リスクデータに基づきベイジアンネットワークを作成できる．具体的な手順は次のようになっている.

① リスクデータを洗い出す
② ベイジアンネットワークのノードを特定する
③ ベイジアンネットワークの構造を作成する
(ア) 因果関係を示す
(イ) 各ノードの CPT を特定する
④ ベイジアンネットワークによるリスク発生確率の算出

5.3.1　リスクデータの洗い出し

最初に，情報システム開発における資産・脅威・脆弱性を洗い出す．一般的な洗い出しの順番は，まず，情報資産を洗い出し，次に資産に対する脅威を特定し，最後に脅威が利用できる脆弱性を特定する.

1)　情報資産の洗い出し

表 29 で例を示すように，システム開発を「委託側(一般企業)の問題」と「受託側(IT 企業)の問題」の２種類に大別できる．情報資産を洗い出す際に，その両方の情報資産を考慮し，ドキュメントを作成する.

表 29．システム開発における情報資産の例

A	システム開発情報	1	開発中システム	A11	要求分析書類
				A12	データベース設計仕様
				A13	詳細設計仕様
				A14	ソースコード
		2	オペレーティングプラットフォーム	A21	OS
				A22	Office ソフト
		3	開発環境	A31	Database
				A32	IDE
				A33	Packaged Software
B	発注側企業情報	1	テストデータ	B11	取引先データ
				B12	製品データ
				B13	顧客データ
				B14	社員情報
				B15	財務情報

2) 脅威と脆弱性を洗い出す

次に脅威と脆弱性を特定するために，洗い出した情報資産により，脅威のリストとして表 30 と脆弱性のリストとして表 31 を作成する．

表 30．脅威リスト

分類	脅威
物理的損傷	火災
	水害
許可されていない行為	認可されていない機器の使用
	データの破壊
重要なサービスの喪失	空調・給水システムの故障
	電源供給の停止
情報を危うくすること	遠隔スパイ行為
	盗聴
技術的な故障	機械の故障
	機械の誤動作

表 31．脆弱性リスト

分類	脆弱性の例	脅威の例
ハードウェア	記憶媒体の不十分な保守/不適当な設置	情報システムの保守に関する違反
	定期交換計画の欠如	機器や媒体の破壊
ソフトウェア	ソフトウェアのテストをしない，又は不十分なソフトウェアのテスト	権限の濫用
	ソフトウェアの公知の欠陥	権限の濫用
	SQL 文のレビューが不足	データの破壊
ネットワーク	保護されていない通信回線	盗聴
	安全性が確保されていないネットワークアーキテクチャ	遠隔スパイ行為
要員	セキュリティ意識の欠如	使用時のミス
	ソフトウェア及びハードウェアの正しくない使い方	使用時のミス
組織	ユーザーの登録及び登録取消に関する正規手続きの欠如	権限の濫用
	定期的監査(監督)の欠如	権限の濫用

5.3.2 ベイジアンネットワークのノード特定

ベイジアンネットワーク中のノードを特定し，ノードのリストを作る．

各脆弱性と脅威はノードで表し，その間の因果関係を確定する．「原因」とするノードは親ノードであり，「結果」ノードは子ノードである．

前の段階で洗い出した脅威と脆弱性に基づき，ノードリストを作る．表 32 と表 33 ではノードの例を挙げて，ノードの依存関係を表示した．

表 32．脅威ノード

ID	脅威	親ノード	子ノード
T_1	遠隔スパイ行為	V_6, V_7, V_8	製品データ A_1
T_2	認可されていない機器の使用	V_6, V_7, V_8	製品データ A_1

表 33．脆弱性ノード

種類	ID	脆弱性	親 ノード	子 ノード
要員	V_1	セキュリティ意識の欠如	/	V_3,V_4
物理	V_2	安全性が確保されていないネットワークアーキテクチャ	/	V_4,V_5
	V_3	入退室管理システムの導入なし	V_1	V_6,V_7, V_8
技術	V_4	保護なし公開ネットワーク接続	V_1, V_2	V_6, V_7
	V_5	暗号化	V_2	V_6, V_7, V_8
	V_6	侵入検知・防御対策不足	V_3, V_4, V_5	T_1, T_2
	V_7	認証	V_3, V_5	T_1, T_2
	V_8	コンピュータウィルス対策	V_3, V_4, V_5	T_1, T_2

5.3.3　ベイジアンネットワークの構造を作成する

1)　脆弱性間の連鎖的な因果関係を示す

資産，脅威と脆弱性はそれぞれ相互関係を矢印で連結する．その際に，V＝脆弱性，T=脅威，A=情報資産で示す．図 17 に示すように前段階で整理したノードリストの例に従って，ベイジアンネットワークを作成でき，各脆弱性間の連鎖的因果関係を可視化する．

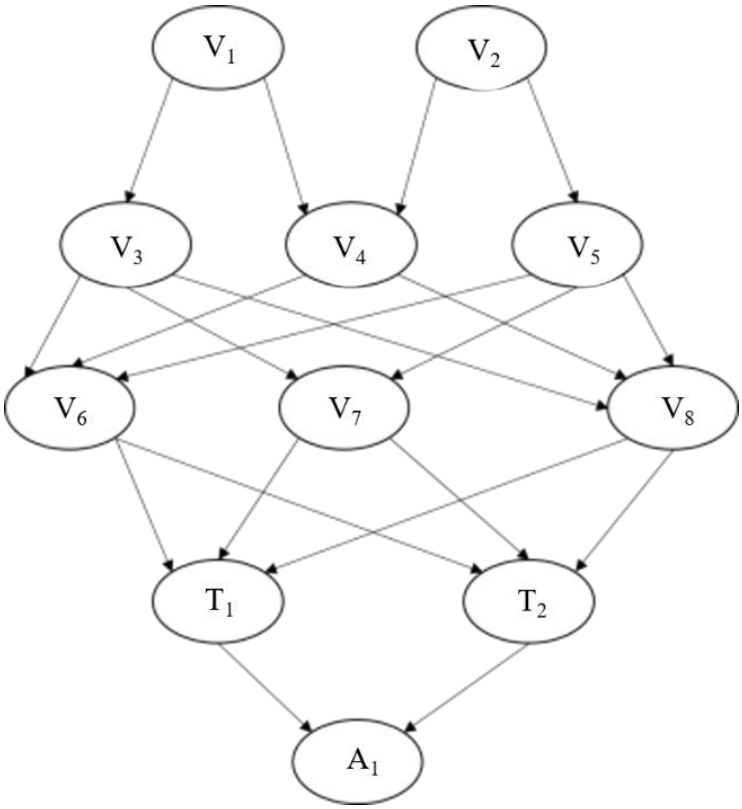


図 17．情報セキュリティのベイジアンネットワーク

2)　CPT によるノード別の条件付き発生確率を示す

図 17 のようなベイジアンネットワークのノードの CPT 値は専門家の評価から決めるべきであるが，次の表 34 で想定データ例を示す．

表 34. 各ノードの CPT の例

ノード	CPT
V_1	$P(V_1=Y)$: 0.13 / $P(V_1=N)$: 0.87
V_2	$P(V_2=Y)$: 0.09 / $P(V_2=N)$: 0.91
V_3	V_1: Y → $P(V_3=Y)$: 0.89, $P(V_3=N)$: 0.11 V_1: N → $P(V_3=Y)$: 0.35, $P(V_3=N)$: 0.65
V_4	V_1: Y, V_2: Y → $P(V_4=Y)$: 0.95, $P(V_4=N)$: 0.05 V_1: Y, V_2: N → $P(V_4=Y)$: 0.90, $P(V_4=N)$: 0.10 V_1: N, V_2: Y → $P(V_4=Y)$: 0.48, $P(V_4=N)$: 0.52 V_1: N, V_2: N → $P(V_4=Y)$: 0.12, $P(V_4=N)$: 0.88
V_5	V_2: Y → $P(V_5=Y)$: 0.90, $P(V_5=N)$: 0.10 V_2: N → $P(V_5=Y)$: 0.17, $P(V_5=N)$: 0.83
V_6	V_3: Y, V_4: Y, V_5: Y → $P(V_6=Y)$: 0.87, $P(V_6=N)$: 0.23 V_3: Y, V_4: N, V_5: Y → $P(V_6=Y)$: 0.74, $P(V_6=N)$: 0.26 V_3: N, V_4: Y, V_5: Y → $P(V_6=Y)$: 0.55, $P(V_6=N)$: 0.45 V_3: N, V_4: N, V_5: Y → $P(V_6=Y)$: 0.13, $P(V_6=N)$: 0.87 V_3: Y, V_4: Y, V_5: N → $P(V_6=Y)$: 0.85, $P(V_6=N)$: 0.15 V_3: Y, V_4: N, V_5: N → $P(V_6=Y)$: 0.71, $P(V_6=N)$: 0.29 V_3: N, V_4: Y, V_5: N → $P(V_6=Y)$: 0.54, $P(V_6=N)$: 0.46 V_3: N, V_4: N, V_5: N → $P(V_6=Y)$: 0.18, $P(V_6=N)$: 0.82
V_7	V_3: Y, V_5: Y → $P(V_7=Y)$: 0.65, $P(V_7=N)$: 0.35 V_3: Y, V_5: N → $P(V_7=Y)$: 0.54, $P(V_7=N)$: 0.46 V_3: N, V_5: Y → $P(V_7=Y)$: 0.20, $P(V_7=N)$: 0.80 V_3: N, V_5: N → $P(V_7=Y)$: 0.05, $P(V_7=N)$: 0.95

表 34 各ノードの CPT の例(続き)

ノード	CPT				
V_8	V_3	V_4	V_5	$P(V_8=Y)$	$P(V_8=N)$
	Y	Y	Y	0.93	0.07
	Y	N	Y	0.64	0.36
	N	Y	Y	0.58	0.42
	N	N	Y	0.33	0.67
	Y	Y	N	0.44	0.56
	Y	N	N	0.37	0.63
	N	Y	N	0.32	0.68
	N	N	N	0.20	0.80
T_1	V_6	V_7	V_8	$P(T_1=Y)$	$P(T_1=N)$
	Y	Y	Y	0.98	0.02
	Y	N	Y	0.95	0.05
	N	Y	Y	0.87	0.13
	N	N	Y	0.65	0.35
	Y	Y	N	0.85	0.15
	Y	N	N	0.83	0.17
	N	Y	N	0.30	0.70
	N	N	N	0.11	0.89
T_2	V_6	V_7	V_8	$P(T_2=Y)$	$P(T_2=N)$
	Y	Y	Y	0.99	0.01
	Y	N	Y	0.89	0.11
	N	Y	Y	0.51	0.49
	N	N	Y	0.34	0.66
	Y	Y	N	0.86	0.16
	Y	N	N	0.73	0.27
	N	Y	N	0.25	0.75
	N	N	N	0.09	0.91

5.3.4　ベイジアンネットワークによるリスク発生確率の算出

1)　CPT による確率の計算方法

ベイジアンネットワークを利用し，任意の脆弱性ノードから周辺のノードの確率を算出できる．つまり，ベイジアンネットワーク推論アルゴリズムを使用して，CPT に従って，ノードの発生確率を計算する．ベイジアンネットワーク推論は，因果推論や障害診断などの方法があり，本論で因果推論を使って，結合確率法で確率を計算する．

例の図 18 により，P(B)は下記の式で算出する(A，B は事象, P(B)は B 事象の発生確率)：

$$\mathrm{P(B)} = \sum_{A} P(AB)$$

$$= P(B|A) \times \mathrm{P(A)} + \mathrm{P(B|\overline{A})} \times \mathrm{P(\overline{A})} \qquad (1)$$

A	P(A)
Y	0.9
N	0.1

A	P(B=Y)	P(B=N)
Y	0.9	0.1
N	0.2	0.8

図 18．結合確率計算の例

リスク発生確率の算出

前表 34 の例を使って，V_3 の確率 $\mathrm{P(V_3)}$を算出する．V_3 の親ノードは V_1 である．V_1 の確率は $\mathrm{P(V_1=Y)}$の確率である．つまり，$\mathrm{P(V_1)=0.13}$ である．数式(1)により，V_3 の確率を算出する．この結果はノード V_3 で入退室管理システム上のリスク発生確率である．

他のノードの確率の計算方法は V_3 の計算方法と同様で，表 34 のデータを例として，全てのノードの確率を算出できる．

$$\mathrm{P(V_3)} = \sum_{V1} P(\mathrm{V_1V_3})$$

$$= P(\mathrm{V_3|V_1}) \times \mathrm{P(V_1)} + \mathrm{P(V_3|\overline{V_1})} \times \mathrm{P(\overline{V_1})}$$

$$= 0.89 \times 0.13 + 0.35 \times 0.87$$

$$= 0.42$$

表 35. ノード別リスクの確率表

リスクノード	確率
V_1	0.13
V_2	0.09
V_3	0.42
V_4	0.25
V_5	0.24
V_6	0.47
V_7	0.29
V_8	0.35
T_1	0.60
T_2	0.50

リスク対策を作成する際に，ベイジアンネットワークを使用し，脆弱性に対する対策を策定すると他の脆弱性や脅威の発生確率の変動が推測できる．例えば，図 17 の中の V_3 について，この脆弱性に対する対策を実施すると，リスク発生確率は現在の 0.42 から 0.32 に減らすことができる．この場合，V_7 の発生率は式(１)により計算して，0.29 から 0.24 になる．

5.3.5　リスク分析結果とリスク対策の作成

リスク対策の作成と実施には，以下を考慮する．

1)　対策優先度を決める

ベイジアンネットワークを利用し，脆弱性ごとに脅威まで到着のルートを判明し，脅威の発生確率が明らかになると，情報資産に存在しているリスクの値が判断できる．これにより，脆弱性に対する対処の優先度の判明がリスクの減少が支援できるものと考える．

2)　最適な対策の策定

脆弱性の間の連鎖関係があるため，対策を策定する際に，各脆弱性の対策を実施するコストが判明できる．ベイジアンネットワークを利用し，個別脆弱性に対する対策を実施したら，周辺の脆弱性ノードの確率も変化する．この関連性を考慮し，対策を実施する際に，かかる資金や人力が少ない対策を選定することも考えられる．

5.4 まとめ

本章では提案分析手法を説明し，その構成方法を論述した．

まず，ベイジアンネットワークについて考察し，その基礎を説明した．ベイジアンネットワークは相互影響が存在している各リスクの間に因果関係を図で表示できるために，情報セキュリティの情報資産・脅威・脆弱性を示すことにより，便利性があることが明らかになった．また，ベイジアンネットワークの CPT で確率を算出する方法があることから，リスクを分析する際に，適切な方法であると考えた．

次にベイジアンネットワークを用いた情報システム開発における情報セキュリティのリスク分析手法を述べた．

この手法は情報システム開発における情報セキュリティリスクを対象とするため，最初はシステム開発中に存在しているセキュリティリスクを洗い出す方法を考察し，研究を行った．そして，洗い出したリスクにより，ベイジアンネットワークのノードを特定し，ノード間の因果関係を整理した．その因果関係により，ベイジアンネットワークのグラフを作成した．最後に各ノードの確率表を決めた．

本章では，第 4 章で述べた問題点に対する解決策として，ベイジアンネットワークを用いた情報システムの開発における情報セキュリティリスクを分析する手法を提案した．

第 1 の問題点である情報セキュリティの脆弱性を確定する場合，現在は各脆弱性間の関係が考えられてない．この既存のリスク見積り表で情報セキュリティの各脆弱性間に因果関係を示せないのに対して，前述の 5.3.3 項でベイジアンネットワークを利用して，その因果関係を考慮し，グラフで直観的に表示できた．

第 2 の問題点としては，脆弱性の相互影響は従来手法で表することができない．脆弱性に対応策を実施するとその脆弱性により発生するリスクの確率が変化する可能性が高いが，既存の評価手法によって，その変化が対応できない．5.3.4 項で述べたベイジアンネットワーク中の CPT を利用して，その確率の変化を算出できた．

第6章 結論

6.1 研究成果のまとめ

本論での提案は下記の具体的な課題を解決した.

課題一：情報システムのオフショア開発プロジェクトのリスクマネジメント方法の改善

課題二：情報システム開発における情報セキュリティリスクの分析手法の提案

6.1.1 課題一の成果

本論文は 2 章と 3 章で課題一に対する研究をまとめた．情報システムのオフショア開発の現状を分析し，オフショア開発プロジェクトにおけるリスクをマネジメントすることに存在している問題点を得た.

まず，情報システムのオフショア開発におけて，発注側と受注側は各自の利益や立場から考えると，プロジェクトの目標に対する認識が異なって，リスクに対する分析の結果が異なる可能性が高く，リスク影響度の尺度を定義することが難しい．CMMI-DEV でリスクマネジメントを行う際に，正しいリスク影響度を確定することできない可能性がある.

また，各具体的なプロジェクトの中で各リスクの影響が単独で存在せず，相互に依存するリスク影響度評価方法が存在し，各リスクの総合的な影響を受けることである．オフショア開発プロジェクトのリスクを評価する際に，プロジェクトのコスト，納期など目標を分けて評価したら，各リスク項目がプロジェクト全体の目標への影響を定量的な数値で表すことが難しい.

従って，上記の 2 つの既存の現状が存在しているため，情報システムのオフショア開発プロジェクトにおける複数のリスク影響度を定量的な数値で表すことが難しいという問題点を導く.

この問題点を解決するため，既存のリスクマネジメントのリスク分析及び評価する際に，コンジョイント分析手法を導入し，マネジメントプロセスを改善した.

これにより，この改善した方法は情報システムのオフショア開発におけるリスクを分析及び評価する際のマネジメント方法として有用の可能性があるとわかった.

6.1.2 課題二の成果

本論文の 4 章と 5 章では，課題二を解決するため，ベイジアンネットワークにより情報セキュリティ・リスクマネジメントに対する研究をまとめた．現在，既存の情報システムのセキュリティに対するリスク評価する方法があるが，情報システム開発における情報セキュリティリスクが存在しているから，情報システム開発部門またシステム開発会社はシステム開発中のセキュリティリスクをマネジメントする方法が不十分であると考え，2 つの問題点を明らかにした.

第 1 の問題点として，情報セキュリティの脆弱性を確定する場合，現在は各脆弱性間の関係が考えられてない.

第 2 の問題点として，脆弱性は相互の影響が存在しているので，一つ脆弱性に対して対策をすれば，他の関連ある脆弱性の発生確率は変わる.

本論文では，その問題を検討し，リスク分析手法を提案した．この提案では，国際標準 ISO27000 シリーズに基づき開発中情報資産・脅威・脆弱性を洗い出して，その 3 要素間に因果関係をベイジアンネットワークで示す方法を述べた．さらに，この手法で各要素の確率により最後のリスクインシデントの発生確率を算出できることを得た.

これにより，第 5 章での提案手法を用いて，情報システム開発中におけるリスクをマネジメントする際のリスク分析を出来て，開発前にリスク対策を決めることに参考できる価値があると思われる.

6.2　本論文の結論

6.2.1　情報システムオフショア開発プロジェクト・リスク分析と評価提案結論

本論文では，調査企業の主な事業が日中間経理・会計などアプリケーション Web システム開発であるので，このような情報システムのオフショア開発におけるプロジェクトのリスクをマネジメントするときに存在している問題点を考察した.

コンジョイント分析を用いて，複数のリスク項目はプロジェクトの全体に対する影響することが出来た．本研究ではコンジョイント分析を利用し出来た効用値を考察し，効用値は各リスク項目がプロジェクト全体への影響を適切表すため，リスクマネジメントにおけるリスク軽減計画を策定する場合，リスク評価のマトリックス表を作成した.

提案としては，CMMI－DEV のリスクマネジメントのプロセスにコンジョイント分析手法を導入した．コンジョイント分析の結果とする効用値をリスク影響度の値とし，リスク値を確定し，優先順位を付ける．現在のプロセスでは，従来のリスク項目の影響度が単一プロジェクト目標(例えばコストの目標，納期目標など)に対する影響度を確定する．効用値はプロジェクト全体への影響を表せるため，既存プロセスに導入し，リスクマネジメントする際に，リスク項目がプロジェクト全体への影響度を明確できる.

6.2.2　情報システム開発における情報セキュリティリスク分析手法提案結論

情報システムの開発における情報セキュリティの問題を解決するため，脆弱性など間に連鎖的な因果関係とリスクの発生確率を計算することが必要である．その結果はリスク対策決定の基本であると考える.

本研究は情報システム開発における情報セキュリティ・リスクマネジメントの立場から考えることで，リスクアセスメント改善案を提出した．

この方法は情報セキュリティリスクを分析する際に，脆弱性の間に相互関係を考慮し，その関係を表示するため，ベイジアンネットワークを作成し，リスク発生確率を算出した．この方法は従来手法と比べて，ベイジアンネットワークの導入により，新たな特徴が明らかになった．

第１に，ベイジアンネットワークはグラフの一種類であるから，脆弱性，脅威と情報資産間に連鎖的な因果関係を可視化できる．

第２に，この方法により，個別ノード(脆弱性)の対策を実施したら，このノードとその周りノードの発生確率が変わるが，この変化も算出できる．

第３に, ベイジアンネットワークを利用することにより, 確率で正規化されたリスク値として，算出することが可能である．従来手法では，リスク値は完全に専門家に評価された定性的な数値である．提案手法は定性的なノードの条件付き確率から，定量化された発生確率を算出する．

これにより以下の成果が期待できる．

① 情報システムを開発する時点から，情報セキュリティリスクを把握できる．

② 情報セキュリティにおけるリスク発生確率を推測し，情報資産に対する発生可能性が高いリスクインシデントを確定できる．

③ 脆弱性間に連鎖的な因果関係を考慮してリスクの最優対策を作成できる．

6.3 今後の課題

情報システム開発におけるリスクのマネジメントは重要である．しかし，情報システム開発のリスクは，さまざまな分野に存在していることから，幅広い検討が必要である．グローバル化が進む現状では，国際間の協同開発の事例が多く，今回，対象とした経理システム以外にも多くのシステムが存在する．今後の課題としては，このような様々な分野におけるオフショア開発と情報セキュリティの両方を検討する必要がある．

また，今回は，マネジメント（CMMI）の観点に主眼を置いているのに対し，より一般化するためには，プロダクトに主眼を置くコモンクライテリア（ISO/IEC 15408）の適用についても検討する必要がある．

謝辞

本研究を行うにあたり, 主査として, 種々ご指導いただきました森雅俊教授に心より深謝申し上げます.

また，副査の秋葉先生,谷本先生，鴻巣先生，滝先生にも，大変多くの良き指導を頂き，博士論文を完成させることができました．心より感謝を申し上げます.

付録

本論文では，コンジョイント分析を実施するため作成するプロジェクトのプロフィールがすべて 27 件である．この 27 件のプロフィールに対する評価の調査票(カード)は下記に示す.

プロジェクト P1 の評価

P1 の各項目の状況は次のように示している，P1 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	単純
2	性能条件の明確	明確
3	仕様確定と仕様変更	変更・追加が少ない
4	開発部門の管理スキル	スキルが高い
5	スケジュールの厳しさ	厳しくない
6	コストの妥当性	予算内
7	開発中の問題対応	柔軟である
8	受注側の技術力	技術力高い
9	受注側のコミュニケーション能力	能力が高い
10	開発環境の整備	整備済み
11	受注側開発要員の定着度	定着している
12	経営安定性	安定

P1 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P2 の評価

P2 の各項目の状況は次のように示している，P2 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	普通
2	性能条件の明確	主な機能が明確
3	仕様確定と仕様変更	変更・追加が少ない
4	開発部門の管理スキル	スキルが高い
5	スケジュールの厳しさ	軽微な期間延長
6	コストの妥当性	軽微なコスト増大
7	開発中の問題対応	対応できる
8	受注側の技術力	技術力高い
9	受注側のコミュニケーション能力	能力が高い
10	開発環境の整備	整備不完備
11	受注側開発要員の定着度	離職率が高い
12	経営安定性	不安定

P2 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P3 の評価

P3 の各項目の状況は次のように示している，P3 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	複雑
2	性能条件の明確	明確できない
3	仕様確定と仕様変更	変更・追加が多い
4	開発部門の管理スキル	スキルが高い
5	スケジュールの厳しさ	厳しくない
6	コストの妥当性	予算内
7	開発中の問題対応	対応できる
8	受注側の技術力	技術力低い
9	受注側のコミュニケーション能力	能力が低い
10	開発環境の整備	必要程度の整備済み
11	受注側開発要員の定着度	離職率が高い
12	経営安定性	安定

P3 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P4 の評価

P4 の各項目の状況は次のように示している，P4 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	普通
2	性能条件の明確	主な機能が明確
3	仕様確定と仕様変更	変更・追加がやや多い
4	開発部門の管理スキル	スキルが高い
5	スケジュールの厳しさ	厳しくない
6	コストの妥当性	予算内
7	開発中の問題対応	柔軟でない
8	受注側の技術力	平均水準技術力
9	受注側のコミュニケーション能力	教育で満足
10	開発環境の整備	整備不完備
11	受注側開発要員の定着度	交代要員がいる
12	経営安定性	安定

P4 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P5 の評価

P5 の各項目の状況は次のように示している，P5 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	普通
2	性能条件の明確	明確できない
3	仕様確定と仕様変更	変更・追加がやや多い
4	開発部門の管理スキル	スキルが低い
5	スケジュールの厳しさ	軽微な期間延長
6	コストの妥当性	予算内
7	開発中の問題対応	柔軟である
8	受注側の技術力	技術力高い
9	受注側のコミュニケーション能力	能力が低い
10	開発環境の整備	整備済み
11	受注側開発要員の定着度	交代要員がいる
12	経営安定性	予期内安定

P5 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P6 の評価

P6 の各項目の状況は次のように示している，P6 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	単純
2	性能条件の明確	主な機能が明確
3	仕様確定と仕様変更	変更・追加がやや多い
4	開発部門の管理スキル	スキルが低い
5	スケジュールの厳しさ	厳しくない
6	コストの妥当性	40%超コスト
7	開発中の問題対応	柔軟でない
8	受注側の技術力	技術力高い
9	受注側のコミュニケーション能力	能力が低い
10	開発環境の整備	必要程度の整備済み
11	受注側開発要員の定着度	離職率が高い
12	経営安定性	不安定

P6 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P7 の評価

P7 の各項目の状況は次のように示している，P7 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	単純
2	性能条件の明確	明確できない
3	仕様確定と仕様変更	変更・追加が多い
4	開発部門の管理スキル	平均水準スキル
5	スケジュールの厳しさ	厳しくない
6	コストの妥当性	軽微なコスト増大
7	開発中の問題対応	対応できる
8	受注側の技術力	技術力高い
9	受注側のコミュニケーション能力	教育で満足
10	開発環境の整備	整備不完備
11	受注側開発要員の定着度	交代要員がいる
12	経営安定性	予期内安定

P7 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P8 の評価

P8 の各項目の状況は次のように示している，P8 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	複雑
2	性能条件の明確	主な機能が明確
3	仕様確定と仕様変更	変更・追加が少ない
4	開発部門の管理スキル	平均水準スキル
5	スケジュールの厳しさ	軽微な期間延長
6	コストの妥当性	40%超コスト
7	開発中の問題対応	対応できる
8	受注側の技術力	平均水準技術力
9	受注側のコミュニケーション能力	能力が低い
10	開発環境の整備	整備済み
11	受注側開発要員の定着度	交代要員がいる
12	経営安定性	安定

P8 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P9 の評価

P9 の各項目の状況は次のように示している，P9 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	単純
2	性能条件の明確	主な機能が明確
3	仕様確定と仕様変更	変更・追加が少ない
4	開発部門の管理スキル	スキルが低い
5	スケジュールの厳しさ	軽微な期間延長
6	コストの妥当性	予算内
7	開発中の問題対応	対応できる
8	受注側の技術力	技術力低い
9	受注側のコミュニケーション能力	教育で満足
10	開発環境の整備	必要程度の整備済み
11	受注側開発要員の定着度	定着している
12	経営安定性	予期内安定

P9 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P10 の評価

P10 の各項目の状況は次のように示している，P10 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	単純
2	性能条件の明確	明確
3	仕様確定と仕様変更	変更・追加がやや多い
4	開発部門の管理スキル	スキルが高い
5	スケジュールの厳しさ	厳しい
6	コストの妥当性	40%超コスト
7	開発中の問題対応	対応できる
8	受注側の技術力	平均水準技術力
9	受注側のコミュニケーション能力	教育で満足
10	開発環境の整備	整備済み
11	受注側開発要員の定着度	離職率が高い
12	経営安定性	予期内安定

P10 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P11 の評価

P11 の各項目の状況は次のように示している，P11 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	単純
2	性能条件の明確	明確
3	仕様確定と仕様変更	変更・追加が多い
4	開発部門の管理スキル	スキルが高い
5	スケジュールの厳しさ	軽微な期間延長
6	コストの妥当性	軽微なコスト増大
7	開発中の問題対応	柔軟でない
8	受注側の技術力	技術力低い
9	受注側のコミュニケーション能力	能力が低い
10	開発環境の整備	整備済み
11	受注側開発要員の定着度	交代要員がいる
12	経営安定性	不安定

P11 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P12 の評価

P12 の各項目の状況は次のように示している，P12 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	複雑
2	性能条件の明確	主な機能が明確
3	仕様確定と仕様変更	変更・追加がやや多い
4	開発部門の管理スキル	平均水準スキル
5	スケジュールの厳しさ	厳しくない
6	コストの妥当性	軽微なコスト増大
7	開発中の問題対応	柔軟でない
8	受注側の技術力	技術力低い
9	受注側のコミュニケーション能力	能力が高い
10	開発環境の整備	整備済み
11	受注側開発要員の定着度	定着している
12	経営安定性	予期内安定

P12 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P13 の評価

P13 の各項目の状況は次のように示している，P13 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	複雑
2	性能条件の明確	主な機能が明確
3	仕様確定と仕様変更	変更・追加が多い
4	開発部門の管理スキル	平均水準スキル
5	スケジュールの厳しさ	厳しい
6	コストの妥当性	予算内
7	開発中の問題対応	柔軟である
8	受注側の技術力	技術力高い
9	受注側のコミュニケーション能力	教育で満足
10	開発環境の整備	整備済み
11	受注側開発要員の定着度	離職率が高い
12	経営安定性	不安定

P13 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P14 の評価

P14 の各項目の状況は次のように示している，P14 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	普通
2	性能条件の明確	明確できない
3	仕様確定と仕様変更	変更・追加が多い
4	開発部門の管理スキル	スキルが低い
5	スケジュールの厳しさ	厳しくない
6	コストの妥当性	40%超コスト
7	開発中の問題対応	対応できる
8	受注側の技術力	平均水準技術力
9	受注側のコミュニケーション能力	能力が高い
10	開発環境の整備	整備済み
11	受注側開発要員の定着度	定着している
12	経営安定性	不安定

P14 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P15 の評価

P15 の各項目の状況は次のように示している，P15 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	複雑
2	性能条件の明確	明確
3	仕様確定と仕様変更	変更・追加が少ない
4	開発部門の管理スキル	スキルが低い
5	スケジュールの厳しさ	厳しくない
6	コストの妥当性	40%超コスト
7	開発中の問題対応	柔軟である
8	受注側の技術力	技術力低い
9	受注側のコミュニケーション能力	教育で満足
10	開発環境の整備	整備不完備
11	受注側開発要員の定着度	交代要員がいる
12	経営安定性	不安定

P15 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P16 の評価

P16 の各項目の状況は次のように示している，P16 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	普通
2	性能条件の明確	明確
3	仕様確定と仕様変更	変更・追加がやや多い
4	開発部門の管理スキル	平均水準スキル
5	スケジュールの厳しさ	厳しい
6	コストの妥当性	予算内
7	開発中の問題対応	対応できる
8	受注側の技術力	技術力低い
9	受注側のコミュニケーション能力	能力が高い
10	開発環境の整備	必要程度の整備済み
11	受注側開発要員の定着度	交代要員がいる
12	経営安定性	不安定

P16 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P17 の評価

P17 の各項目の状況は次のように示している，P17 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	単純
2	性能条件の明確	明確できない
3	仕様確定と仕様変更	変更・追加が少ない
4	開発部門の管理スキル	平均水準スキル
5	スケジュールの厳しさ	厳しい
6	コストの妥当性	予算内
7	開発中の問題対応	柔軟でない
8	受注側の技術力	平均水準技術力
9	受注側のコミュニケーション能力	能力が低い
10	開発環境の整備	整備不完備
11	受注側開発要員の定着度	定着している
12	経営安定性	不安定

P17 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P18 の評価

P18 の各項目の状況は次のように示している，P18 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	単純
2	性能条件の明確	明確できない
3	仕様確定と仕様変更	変更・追加がやや多い
4	開発部門の管理スキル	平均水準スキル
5	スケジュールの厳しさ	軽微な期間延長
6	コストの妥当性	40%超コスト
7	開発中の問題対応	柔軟である
8	受注側の技術力	技術力低い
9	受注側のコミュニケーション能力	能力が高い
10	開発環境の整備	整備不完備
11	受注側開発要員の定着度	離職率が高い
12	経営安定性	安定

P18 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P19 の評価

P19 の各項目の状況は次のように示している，P19 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	複雑
2	性能条件の明確	明確
3	仕様確定と仕様変更	変更・追加がやや多い
4	開発部門の管理スキル	スキルが低い
5	スケジュールの厳しさ	厳しい
6	コストの妥当性	軽微なコスト増大
7	開発中の問題対応	対応できる
8	受注側の技術力	技術力高い
9	受注側のコミュニケーション能力	能力が低い
10	開発環境の整備	整備不完備
11	受注側開発要員の定着度	定着している
12	経営安定性	安定

P19 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P20 の評価

P20 の各項目の状況は次のように示している，P20 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	単純
2	性能条件の明確	主な機能が明確
3	仕様確定と仕様変更	変更・追加が多い
4	開発部門の管理スキル	スキルが低い
5	スケジュールの厳しさ	厳しい
6	コストの妥当性	軽微なコスト増大
7	開発中の問題対応	柔軟である
8	受注側の技術力	平均水準技術力
9	受注側のコミュニケーション能力	能力が高い
10	開発環境の整備	必要程度の整備済み
11	受注側開発要員の定着度	交代要員がいる
12	経営安定性	安定

P20 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P21 の評価

P21 の各項目の状況は次のように示している，P21 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	普通
2	性能条件の明確	明確できない
3	仕様確定と仕様変更	変更・追加が少ない
4	開発部門の管理スキル	スキルが低い
5	スケジュールの厳しさ	厳しい
6	コストの妥当性	軽微なコスト増大
7	開発中の問題対応	柔軟でない
8	受注側の技術力	技術力低い
9	受注側のコミュニケーション能力	教育で満足
10	開発環境の整備	整備済み
11	受注側開発要員の定着度	離職率が高い
12	経営安定性	安定

P21 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P22 の評価

P22 の各項目の状況は次のように示している，P20 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	複雑
2	性能条件の明確	明確できない
3	仕様確定と仕様変更	変更・追加が少ない
4	開発部門の管理スキル	スキルが高い
5	スケジュールの厳しさ	厳しい
6	コストの妥当性	40%超コスト
7	開発中の問題対応	柔軟でない
8	受注側の技術力	技術力高い
9	受注側のコミュニケーション能力	能力が高い
10	開発環境の整備	必要程度の整備済み
11	受注側開発要員の定着度	交代要員がいる
12	経営安定性	予期内安定

P22 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P23 の評価

P23 の各項目の状況は次のように示している，P23 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	普通
2	性能条件の明確	明確
3	仕様確定と仕様変更	変更・追加が多い
4	開発部門の管理スキル	平均水準スキル
5	スケジュールの厳しさ	軽微な期間延長
6	コストの妥当性	40%超コスト
7	開発中の問題対応	柔軟でない
8	受注側の技術力	技術力高い
9	受注側のコミュニケーション能力	教育で満足
10	開発環境の整備	必要程度の整備済み
11	受注側開発要員の定着度	定着している
12	経営安定性	安定

P23 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P24 の評価

P24 の各項目の状況は次のように示している，P24 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	複雑
2	性能条件の明確	明確できない
3	仕様確定と仕様変更	変更・追加がやや多い
4	開発部門の管理スキル	スキルが高い
5	スケジュールの厳しさ	軽微な期間延長
6	コストの妥当性	軽微なコスト増大
7	開発中の問題対応	柔軟である
8	受注側の技術力	平均水準技術力
9	受注側のコミュニケーション能力	教育で満足
10	開発環境の整備	必要程度の整備済み
11	受注側開発要員の定着度	定着している
12	経営安定性	不安定

P24 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P25 の評価

P25 の各項目の状況は次のように示している，P25 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	普通
2	性能条件の明確	明確
3	仕様確定と仕様変更	変更・追加が少ない
4	開発部門の管理スキル	平均水準スキル
5	スケジュールの厳しさ	厳しくない
6	コストの妥当性	軽微なコスト増大
7	開発中の問題対応	柔軟である
8	受注側の技術力	平均水準技術力
9	受注側のコミュニケーション能力	能力が低い
10	開発環境の整備	必要程度の整備済み
11	受注側開発要員の定着度	離職率が高い
12	経営安定性	予期内安定

P25 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P26 の評価

P26 の各項目の状況は次のように示している，P26 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	普通
2	性能条件の明確	主な機能が明確
3	仕様確定と仕様変更	変更・追加が多い
4	開発部門の管理スキル	スキルが高い
5	スケジュールの厳しさ	厳しい
6	コストの妥当性	40%超コスト
7	開発中の問題対応	柔軟である
8	受注側の技術力	技術力低い
9	受注側のコミュニケーション能力	能力が低い
10	開発環境の整備	整備不完備
11	受注側開発要員の定着度	定着している
12	経営安定性	予期内安定

P26 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

プロジェクト P27 の評価

P27 の各項目の状況は次のように示している，P27 の全体に対して，評価してください

No.	リスク要因	リスク項目
1	システムの複雑さ	複雑
2	性能条件の明確	明確
3	仕様確定と仕様変更	変更・追加が多い
4	開発部門の管理スキル	スキルが低い
5	スケジュールの厳しさ	軽微な期間延長
6	コストの妥当性	予算内
7	開発中の問題対応	柔軟でない
8	受注側の技術力	平均水準技術力
9	受注側のコミュニケーション能力	能力が高い
10	開発環境の整備	整備不完備
11	受注側開発要員の定着度	離職率が高い
12	経営安定性	予期内安定

P27 に対する評価： 1 2 3 4 5

(不適当 ⟷ 適当)

図の目次

表の目次

参考文献

[1] 中国サービス･アウトソーシング研究センター, 中国サービス･アウトソーシング発展報告(日本語簡版), 中国サービス･アウトソーシング研究センター, 2013.

[2] 株式会社ベネッセホールディングス.(2016). 財務・業績情報. 参照先: 株式会社ベネッセホールディングス: http://www.benesse-hd.co.jp/ja/ir/doc/library/annual/databook2016j.pdf.

[3] 経済産業省, “平成２６年　情報処理実態調査,” 2016.

[4] 独立行政法人情報処理推進機構, 情報セキュリティ白書 2016, 独立行政法人情報処理推進機構(IPA), 2016.

[5] S-open オフショア開発研究会, ソフトウェア開発オフショアリング完全ガイド, 日経 BP 社, 2004.

[6] 情報処理推進機構 IT 人材育成本部, IT 人材白書 2013, 情報処理推進機構, 2013.

[7] SQuBOK 策定部会, ソフトウェア品質知識体系ガイド V2(第 2 版), オーム社, 2014.

[8] 高橋美多. (2009). 中国ソフトウェア産業の技術発展ー日中企業間の分業形態の変化に即して ー. アジア研究, 55(1), 40-53.

[9] 株式会社オプター. (2016). オフショア開発.com. 参照先: http://www.offshore-kaihatsu.com/mypage/opter.php

[10] S. Manning, “The Stability of Offshore Outsourcing Relationships,” Management International Review, 第 巻 51, 第 3, p. 381‑406, 2011.

[11] 亀井利明, 亀井克之. (2009). リスクマネジメント総論(増補版). 同文館.

[12] プロジェクトマネジメント協会(PMI), プロジェクトマネジメント知識体系ガイド(PMBOK ガイド)第 4 版, プロジェクトマネジメント協会(PMI).

[13] Software Engineering Institute. (2016). 開発のための CMMI　1.3 版(日本語訳). 参照先: http://www.sei.cmu.edu/library/assets/whitepapers/CMMI-DEV-V1.3-Japanese.pdf

[14] S. JIANG , M. MORI, “A STUDY ON THE COMPARISON OF ORGANIZATION IN THE IT OFFSHORE DEVELOPMENT BETWEEN JAPAN AND CHINA”, *Proceedings of The Twelfth International Conference On Industrial Management*, 2014, pp.414~418.

[15] 田島理史. (2002). ソフトウェア開発プロジェクトにおける実践的初期リスク管理手法の提案. プロジェクトマネジメント学会誌 Vol.4,No.5.

[16] 松本吉弘. (2014 年). ソフトウェアエンジニアリング基礎知識体系-SWEBOK V3.0-. 株式会社オーム社.

[17] B. W. Boehm, “Software risk management: principles and practices,” IEEE Software Volume: 8, Issue: 1, Jan. 1991, 1991 .

[18] K. M. Eisenhardt, “Agency Theory: An Assessment and Review,” Academy of Management Review, 第 巻 14, 第 1, pp. 57-74, 1989.

[19] IBM. (2016 年 12 月). SPSS Statistics. 参照先 : IBM: http://www-01.ibm.com/software/jp/marketplace/spss/

[20] IBM. (2016 年 12 月). 分析手法別 SPSS 対応製品一覧. 参照先: IBM: http://www-01.ibm.com/software/jp/marketplace/spss/methodology.html

[21] 独立行政法人情報処理推進機構, 情報セキュリティ読本, 実教出版株式会社, 2013.

[22] IPA 独立行政法人 情報処理推進機構. (2016 年 11 月). 情報セキュリティマネジメントと PDCA サイクル. 参照先: https://www.ipa.go.jp/security/manager/protect/pdca/standard.html

[23] 国際標準化機構 (ISO) と国際電気標準会議 (IEC) , ISO/IEC 27001 Information security management systems(2013), 2013.

[24] 国際標準化機構 (ISO) と国際電気標準会議 (IEC) , ISO/IEC 27005:2011 Information security risk management, 2011.

[25] 畠中伸敏, 羽生田和正, 折原秀博, 伊藤重隆, 相沢健実. (2008). 情報セキュリティのためのリスク分析・評価: 官公庁・金融機関・一般企業におけるリスク分析・評価の実践. 日科技連出版社.

[26] 磯部義明. (2015). ベイジアンネットワークを利用した動的モデリングによるセキュリティリスク評価システムの開発.