

履修管理システムにおける オントロジを用いたアカウントビリティ設計手法

早坂 良† 落水 浩一郎†

† 北陸先端科学技術大学院大学 情報科学研究科
{ryoh, ochimizu}@jaist.ac.jp

概要

本論文では、自然言語で記述された規則に従って実現される情報システムである、Law-Defined システムにおけるアカウントビリティの設計手法を提案する。アカウントビリティとはシステムが次の機能を持つことである。(1) システムが提供する機能に関する利用者からの質問に対し、規則を根拠として説明する機能。(2) 利用者が希望する処理結果を得るための入力パラメータの組合せの提案機能。本手法では、まず規則からオントロジを作成し、概念を整理する。次に、原因結果グラフを用いて規則の形式化を行い、これとユースケースモデルからアカウントビリティを設計する。大学の履修管理システムを例にケーススタディを行い、大学が定める履修規則を基にしたアカウントビリティの設計を具体的に示す。

A Method for Designing Accountability By Using Ontology in University Registration System

Ryo HAYASAKA † Koichiro OCHIMIZU †

† School of Information Science, Japan Advanced Institute of Science and Technology

Abstract

In this paper, we propose a method for designing accountability in law-defined systems, which are the information systems that must obey the rules written in natural languages. The method roughly consists of the following steps: Firstly, an ontology is constructed by analyzing the rules so that the notion used in the rules can be reused. Next, the rules are formalized by using a cause-effect graph. Finally, the accountability of the system is designed by using the use case model of the system and the cause-effect graph. We introduce a case study of a university registration system which must obey the registration rules, and illustrate applying the method.

1 はじめに

近年、電子政府・電子自治体への取り組みをはじめとして、社会システムの電子化が急速に押し進められている。政治、行政、司法、金融、医療、交通、教育、企業などが電子システム化され、それらがネットワークを介して相互に接続され巨大な電子社会システムが構築されつつある。われわれの日常生活は、社会基盤としてのこのような電子社会システムの上に成り立っている。したがって、電子社会システムは、機能を単に提供するだけではなく、われわれが安心して生活できることを保証できるように設計・構築され、かつ運用・保守されて

いる必要がある。

本学 21 世紀 COE プログラム「検証進化可能電子社会」(拠点リーダー: 片山卓也) [1] では、安心できる電子社会システムが持っているべき要件として、正当性、アカウントビリティ、セキュリティ、耐故障性、進化性の 5 つからなる安心性要件を提案しており、本研究ではそのうち **アカウントビリティ (Accountability)** および **進化容易性 (Ease-of-Evolution)** *1 を対象としている *2。

*1 ソフトウェアの進化性に加えて、われわれは進化のための変更コストを低減することも特に重要視しており、これを進化容易性と呼ぶ。

*2 アカウントビリティの実現のため、必然的に正当性も一

アカウントビリティとは、情報システムが提供する機能に関する利用者からの質問に対して規則を根拠として説明する機能、および利用者が希望する処理結果を得るための入力パラメータの組合せの提案機能の2つの機能を持つことである。

われわれは、これまでソフトウェアアーキテクチャによるアカウントビリティおよび進化容易性の実現に向けた研究に取り組んでおり [2] [3]、既存の情報システムに対して変更を加えずに両性質を付加する実現アプローチによって構成された参照アーキテクチャを提案し、広く一般の電子社会システムが採用している3層モデルとの対応を示した [4]。

アカウントビリティは、情報システムが従っている規則、および情報システムに関わる複数のステークホルダの関心から設計されなければならない。ステークホルダは、一般利用者、業務担当者、社会規則整備担当者、システム開発・保守担当者の4種類を考えているが、本稿では、一般利用者に対するアカウントビリティの設計手法、すなわち、自然言語で記述された規則の概念をオントロジおよび原因結果グラフを用いて整理し、一般利用者の関心事をモデル化したユースケースモデルを用いることによりアカウントビリティを設計する手法を提案する。

論文の構成は以下のとおりである。まず、2節で本研究で対象としている Law-Defined システムおよびアカウントビリティについての定義とアカウントビリティの設計に関する課題を述べる。3節で課題に対するアプローチと提案手法の詳細を説明し、4節で履修管理システムを対象としたケーススタディを行い提案手法の適用結果を示す。5節では、いくつかの観点から本手法の考察を行う。最後に6節でまとめと今後の課題を述べる。

2 Law-Defined システムにおけるアカウントビリティ

電子社会システムの規模や特徴はさまざまであり、社会における情報システムの位置付けによって決まってくる部分が大きいため、どのような情報システムが電子社会システムであるかを一般的に定義するのは難しい。そこで、電子社会システムを抽象したシステムである Law-Defined システムを導入し、それにアカウントビリティを定義する。アカウントビリティを設計する上での課題を整理する。

部研究対象に含まれる。

2.1 社会規則と Law-Defined システム

社会には、国や地方自治体が定める法令や、会社などの各組織が定める各種規則があり、これらを総称して**社会規則**と呼ぶことにする。社会規則は人や電子社会システムの振る舞いを制約するものであり、電子社会システムは社会規則を完全に満たすように構築されていなければならない。さらに、社会規則に従ってシステムが正しく構築されていることを保証できること、または社会規則に従ってシステムが正しく構築されていることを確認する手段を与える必要がある。一方、社会は常に変化しており、社会規則もそれに応じて適宜改定されていく。電子社会システムは、社会規則の改定に応じてシステムを迅速に進化させていかななければならない。そのためには低いコストでシステムを変更できる必要がある。社会規則の代表例は国や地方自治体が定める法令である。社会規則は自然言語で記述され、条・項または章・節などの単位によって構成される。こういった特徴を持つシステムを片山は **Law-Defined システム**と呼んでいる。電子社会システムは Law-Defined システムの一種と考えることができ、本研究では Law-Defined システムを対象として議論を行う。

社会規則は、階層的に体系化されているものとする。すなわち、上位の層の規則の効力は下位の層の規則の効力に優先する。たとえば、日本国においては最上位の層に日本国憲法があり、下位の層にあるすべての規則（法律など）に優先する。下位の層の規則は上位の層の規則に反する内容であってはならない。個々の社会規則には社会における規則の適用範囲が決まっており、規則はその範囲内において効力を持つ。社会規則は、その適用範囲内において人と Law-Defined システムの両方を支配する。

法令は基本的に、ある物事や事項が一定の条件に該当する場合に、それに対して一定の効果を与えることを内容とする。これを法的要件と法的効果という。法的効果には、権利、義務、禁止などがある [5]。社会規則においても同様に考えることができ、Law-Defined システムは、ある条件を満たしたときにそれに対応する効果（機能）を正しく実現していなければならない。この他、社会規則には、各種制約および条件の定義、事実記述、目的・趣旨、用語・概念の定義、手続き・取扱い・ワークフロー記述、計算式の定義などが含まれる。Law-Defined システムを構築する際、社会規則に規定されている

上記の内容すべてをシステム化する必要はなく、システムの提供する機能に関する規則のみを選び出し、それらをシステムの仕様に取り込んで開発を行う必要がある。

2.2 アカウンタビリティ

一般にアカウンタビリティは説明責任と訳され、出資者（行政は住民，企業は株主）に対し，判断できる材料を示して組織の活動や経営の内容を説明する責任を果たすという意味である [6]。本研究では，この概念を Law-Defined システムに対して適用し，以下のように定義する。

アカウンタビリティとは Law-Defined システムが次の機能を持つことである：

1. システムが提供する機能に関する利用者からの質問に対し，社会規則を根拠として説明する機能
2. 利用者が希望する処理結果を得るための入力パラメータの組合せの提案機能

従来の電子社会システムにおいては，利用者から投入された処理要求は人間の介入しないシステム内部で，さまざまな条件を判断しつつ処理され，結果のみが利用者に提示されるのが普通である。利用者が処理結果の正当性に関する根拠を知りたいとき，多くの場合において処理過程に人間が介入しないことに加え，処理過程のすべてを把握している管理者も存在しないため，人間がその質問に答えることは困難となる。一方，システム自身が利用者からの質問に答えるには，システムに実装されたさまざまな条件や処理内容，およびその根拠となる社会規則を示す必要があると考えられるが，従来の情報システムでは考慮されていない機能である。

また，電子社会システムの利用者は，さまざまなデータを入力して，システムが処理を行い，その結果を受け取るという流れでシステムを利用することが多いと考えられる。そのとき，入力データには任意性がある場合が多く，その入力値の違いによって処理結果が変わってしまうことがある。従来の電子社会システムでは，入力データと処理結果との関係が明示的になっていないため，利用者の得たい処理結果を得にくいという問題が起きる場合がある。利用者の要求に応じて，入力パラメータと処理結果との関連をシステムが説明できれば，この問題は解決できる。

2.3 アカウンタビリティ設計における課題

前節で定義したアカウンタビリティにより，電子社会システムは社会規則を根拠として提示して利用者からの質問に対して説明でき，その結果として利用者はシステムを信頼でき，安心して生活できるようになると期待できる。しかしながら，アカウンタビリティを設計する上で，以下の課題があると考えられる。

- 自然言語で記述された文書である社会規則からアカウンタビリティの抽出および形式化：利用者からの質問に対する説明の根拠としてシステムが従っている社会規則を用いるため，システムが扱えるように自然言語で記述された社会規則を形式化する必要がある。

ここで社会規則の形式化とは，2.1 節で述べたように社会規則は法的要件と法的効果が記述されているものであるため，ある一定の条件を満たした場合に対応する機能が提供されるという関係を社会規則に記述されているすべての事柄について冗長性のない形式言語で記述することである。

形式化するには，規則の意味を一意に決めなければならない。しかし，実際には自然言語で書かれた規則は，複数の解釈が可能であることが多い。本研究では規則の解釈問題には立ち入らず，形式化すべき一意の意味があらかじめ与えられていることとする。

- 個々の Law-Defined システムに必要となるアカウンタビリティの抽出：

Law-Defined システムは，その実現および機能について社会規則から制約を受けるが，社会規則の適用範囲には人間系およびシステム系の両方を含んでいることが普通なので，社会規則を解析することにより，システム化する部分に対して制約を与える規則のみを選別しなければならない。このためには，システムの内部と外部を明確に分離してモデル化する必要がある。

- 社会規則における概念および概念間の依存関係の再利用：

1 節で述べたように，本研究ではアカウンタビリティおよび進化容易性の実現を目的としている。Law-Defined システムにおける進化容易性とは，システムを取り巻く社会や環境の変化に応じて社会規則が改定されていくと

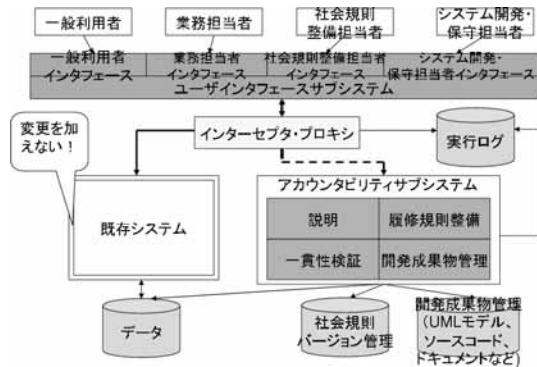


図 1: アカウントビリティを実現する参照アーキテクチャ

き、社会規則の改定に応じてシステムを容易に変更でき、外部の社会環境に迅速に適応できることである。

ある社会規則における概念は、その社会規則の適用範囲内のさまざまなシステムにおいても使用され得る。たとえば、A 大学の履修規則に出てくる概念（修了要件、取得単位、再履修など）は、B 大学の履修規則においてもほとんどの部分が同様に使用可能である。アカウントビリティは社会規則における概念から抽出されるものであり、したがって、社会規則における概念を再利用可能なかたちに抽出できることが望まれる。

さらに、社会規則における概念を抽出し、概念間の関係（特に依存関係）を定義しておけば、規則の改正の際に概念間の関係を元にアカウントビリティの再設計を支援できると考えられる。

3 アプローチと提案手法

2.3 節で述べたアカウントビリティ設計に関する課題について、本研究でとるアプローチを基本となる考え方と共に示し、それに基づいた手法を提案する。

図 1 にアカウントビリティを実現する参照アーキテクチャ [4] を示す。既存システムとユーザインタフェースとの間にラッパーの一種インターセプタ・プロキシを配置し、通常の機能の呼び出しは既存システムで処理され、アカウントビリティ機能の呼び出しはアカウントビリティサブシステムで処理される構造になっている。

本稿で対象としている一般利用者に対するアカウントビリティは、アカウントビリティサブシステム内の説明モジュールで実現される。説明モジュールは、2.2 節で定義したアカウントビリティ機能を提供するために、既存システムが提供する「機能」およびその機能が実行される「条件」との関係を実装していなければならない。したがって、「条件」と「機能」との関係を社会規則から抽出する必要がある。

3.1 アプローチ

以下、各課題に対するアプローチを順に説明する。

- 原因結果グラフを用いた社会規則の形式化：自然言語で記述された文書である社会規則からアカウントビリティの抽出および形式化のために、原因結果グラフを用いるアプローチをとる。

原因結果グラフ [7] は、システムの外部機能に関するテストケースの設計手法のひとつとして提案されており、自然言語で記述された仕様をより形式的な仕様に変換するための厳密な方法である [8]。この方法は、外部仕様の内容の意味を解析し、それを入力（原因）、出力および変換（結果）の間の論理関係として、論理値グラフのかたちに表現するものである。

社会規則には、「一定の条件を満たした場合に、対応する機能が提供される」という条件（法的要件）と機能（法的効果）との関係が書かれているものと考えることができ、この論理関係に基づいてアカウントビリティを設計する。

原因結果グラフの手法を仕様書ではなく社会規則に適用すると、自然言語で記述された社会規則の中から原因（条件）と結果（機能）を識別し、それらを論理的に結合したグラフを得ることができる。

社会規則を形式化した原因結果グラフを利用することにより、Law-Defined システムが提供する機能に対応するすべての条件を容易に得ることができる。その条件の真偽およびその条件が記述されている社会規則の条項等を根拠として説明することにより、アカウントビリティ (1) が実現できる。

また、原因結果グラフにおいて、複数の条件の真偽値の組合せによって異なる結果になる場合、条件の真偽値の組合せと結果を利

用者に提示することにより、アカウントビリティ (2) を実現する。

- Law-Defined システムのユースケースモデルを用いたアカウントビリティのモデル化：
利用者が、システムの提供するありとあらゆる機能について質問できるようにする必要はない。なぜならば、利用者の主な関心は、システムのユースケースとしてとらえられるようなビジネスロジックにあるからである。たとえば、大学の履修管理システムの場合、学生は現在の習得単位数で修了要件を満たしているかどうかや、科目の履修登録のための要件（より入門的な科目の単位を取得済みであることなど）を満たしているかどうかといったことに関心があるのであり、その他の事柄には関心がないのが普通である。

システム内のビジネスロジックにより実現される機能と、システム外の利用者との関係をモデル化するためにユースケースモデルを用いる。アカウントビリティはシステムが利用者に提供する機能であるので、当然ユースケースとしてモデル化できる。

- 社会規則における諸概念に関するオントロジを作成：

社会規則にはその適用範囲に関するさまざまな概念（ドメイン知識）が含まれている。概念の間にはさまざまな関係がある。社会規則に含まれる概念に関するオントロジを作成することにより、概念の再利用および概念間の関係の定義を行うことができる。これは、進化容易性の実現のための一助になると考えている。

社会規則に含まれる概念の抽出および関係の定義を行うオントロジ作成作業は、原因結果グラフを作成する上でも有効であると考えられる。文献 [7] では、「文書を読みながら、出会った原因と結果のそれぞれに下線を引いていく」という方法により原因と結果を抽出する方法を説明している。この方法による原因結果グラフの作成は、作業が容易ではあるものの、少々アドホックであるという欠点がある。

一方、社会規則に関するオントロジを作成するアプローチをとると、オントロジ作成の際に概念間の関係（対照的な概念、推移的な概

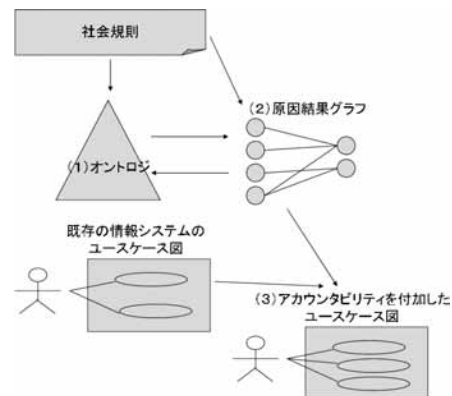


図2: アカウントビリティ設計の流れ

念、必要条件となる概念など）の分析も合わせて行うことになるため、この分析結果を用いることにより、システムティックに原因結果グラフを作成でき、さらに、オントロジに記述された論理式を利用して、原因結果グラフの整合性チェックを行うことができるという利点がある。

3.2 提案手法

前節で述べたアプローチに基づいた提案手法の概略を図2に示す。

社会規則は、条・項または章・節などの各規則単位で一意に識別可能な規則番号が与えられているものとする。これは機械的に付与できる。

本研究では、導入部でも述べたとおり、スクラッチからアカウントビリティを持つ情報システムを構築するのではなく、既存の情報システムにアカウントビリティを付加する方針で研究を進めている。これは、現在既に電子社会システムが数多く存在する中で、もし前者の方針をとる場合、アカウントビリティ実現するためには現存するすべてのシステムを作りなおさなければならないという強い実現上の制約が起きてしまうのを避けるねらいがある。したがって、既存の情報システムのユースケース図は与えられているものとして差し支えない。

以下、各ステップを順に説明する。

[1] オントロジの作成：

社会規則を解析して、社会規則に含まれる概念と概念間の関係を抽出し、オントロジとして整理する。社会規則の適用範囲における、社会規則には表れない背景知識を含めてオン

トログを作成すればより完全なドメイン知識の体系が得られるが、アカウントビリティの設計のみを対象とした場合、あえて背景知識を含める必要はない。

[2] 原因結果グラフの作成：

社会規則とオントログを参考にして、原因結果グラフを作成する。つまり、ある結果を引き起こすすべての条件を洗い出す。単に社会規則を読んで条件と結果を取り出すだけでなく、オントログを用いることによって条件にあたる概念と他の概念との関係が明確になるので、より完全な原因結果グラフが作成できることが期待できる。

また、[1] と [2] の間のステップを繰り返すことで、それぞれ別の観点から社会規則を解析することになるため、両者を洗練させることができる。

[3] アカウントビリティを追加したユースケース図の作成：

既存の情報システムのユースケース図を参照して、システムが提供する機能（ユースケース）と、原因結果グラフの中の結果との対応をとる。対応がとれるということは、アカウントビリティを提供できることを意味している。そのユースケースに対応する、アカウントビリティ・ユースケースをユースケース図にモデル化する。

上記のステップを踏むことで、アカウントビリティを追加した情報システムのユースケース図が得られる。この後、アカウントビリティユースケースに対応する原因結果グラフ中の条件を基に、アカウントビリティを実装していけばよい。

4 ケーススタディ

大学の履修管理システムを例に取り上げ、本提案手法を適用したアカウントビリティの設計について述べる。大学の履修管理システムは、大学が定める履修規則に基づいて実現されていなければならないので Law-Defined システムである。なお、履修規則は本学のもの [9] を使用した。

4.1 履修管理システムの概要

履修管理システムは、学生の履修登録データを管理しているシステムであり、学生からの履修登録要求の処理や修了要件を満たしているかどうかの確認

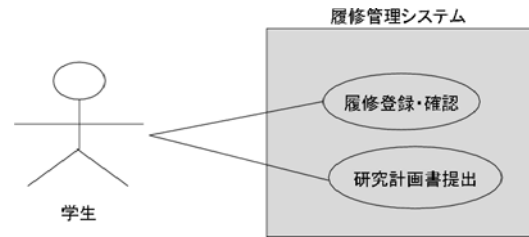


図 3: 履修管理システムのユースケース図

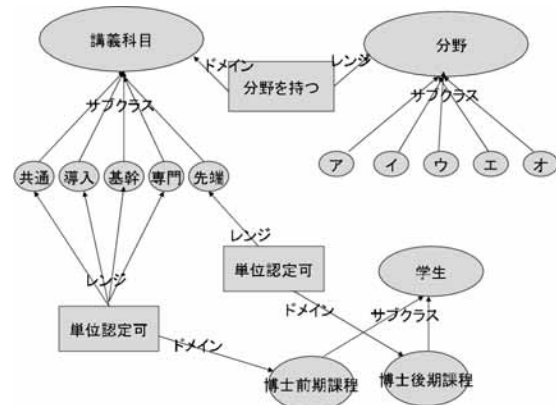


図 4: 履修規則のオントログ (抜粋)

などが行える。

図 3 に本例題で扱う簡略化した履修管理システムのユースケース図を示す。「履修登録・確認」ユースケースは、学生がシステムを使用して履修登録申請を行う機能、および現在履修済みの科目一覧を確認する機能である。「研究計画書提出」ユースケースは、学生がシステムを使用して研究計画書を提出する機能である。

4.2 提案手法の適用

以下、図 2 の設計手順に沿って各ステップの適用結果について説明する。

[1] オントログの作成：

履修規則を基に作成したオントログの抜粋を図 4 に示す。履修規則の解析は、履修規則をまず始めから終わりまで通読し、何度も出現するような重要と思われる概念を取り出してオントログを構成していく方法をとった。

図 4 中の楕円が概念クラスを表し、四角がプロパティを表している。共通、導入、基幹、専門、先端の各科目は講義科目のサブクラスであり、かつ、講義科目は、プロパティ「分

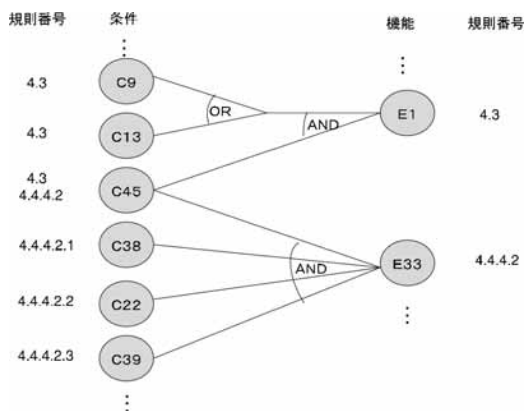


図 5: 履修規則の原因結果グラフ (抜粋)

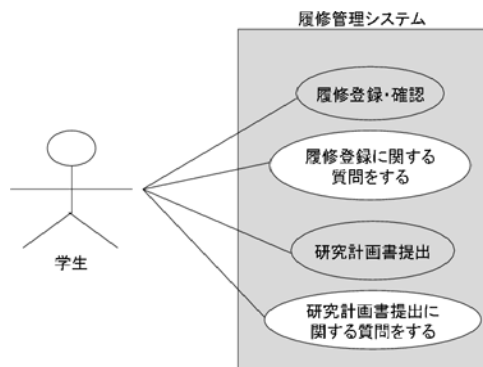


図 6: アカウンタビリティを追加した履修管理システムのユースケース図

野を持つ」により、ア～オのいずれかの分野を最低ひとつ持つことが表現されている。

[2] 原因結果グラフの作成：

履修規則とオントロジを基に作成した原因結果グラフの抜粋を図 5 に示す。履修規則の解析は、オントロジの概念構造を参考にしつつ、各規則ごとに条件と結果を抜き出していく方法で行った。

図 5 中の条件および機能は、以下である。条件には C、機能には E の記号と識別番号を与える。

- C9 : 「I216 の講義の単位を取得済み」
- C13: 「I222 の講義の単位を取得済み」
- C45: 「博士前期課程の学生である」
- C38: 「副テーマの研究が終了している」
- C22: 「基幹・専門講義科目から 5 科目以上、導入・基幹・専門講義科目から 4 分野 8 科目 16 単位以上取得している」
- C39: 「研究計画の内容が十分である」

- E1 : 「I431 を受講できる」
- E33: 「研究計画書が提出可能である」

E1 は、専門科目（比較的高度な内容の科目）を履修するためには、一定の受講条件（入門的な科目が履修済みであること）を満たしている必要があることを表している。E33 は、研究計画書を提出するためには、一定の条件を満たしている必要があることを表している。

また、条件および機能を履修規則から抽出

する際、規則番号と対応付けておくことでトレーサビリティを確保する。C45 は、規則番号 4.3 および 4.4.4.2 に両方で使用される条件である。

[3] アカウンタビリティを追加したユースケース図の作成：

まず、既存の情報システムのユースケース図（図 3）のユースケースと原因結果グラフ（図 5）の「機能」との対応をとる。「履修登録・確認」ユースケースにおいては、科目の履修登録をするためには受講条件を満たしているかどうかのチェックを行う必要があるということが、原因結果グラフからわかる。同様に、「研究計画書提出」ユースケースにおいても、研究計画書の提出条件を満たしているかどうかをチェックする必要があることがわかる。よって、これらのユースケースに対応するアカウンタビリティ・ユースケースをモデル化する。アカウンタビリティを追加したユースケース図を図 6 に示す。

アカウンタビリティ・ユースケース「研究計画書提出に関する質問をする」は、利用者がユースケース「研究計画書提出」を実行した後で、なぜその処理結果になったのかについて質問があった場合に、原因結果グラフの条件の論理値および関連する履修規則を示して説明する機能である。

アカウンタビリティ・ユースケース「履修登録に関する質問をする」では、利用者がある科目を履修したいときに、どの科目を事前に履修すべきかの選択肢を原因結果グラフの条

件を用いて提示する機能である。

5 考察

大学の履修規則と履修管理システムを対象として、部分的ではあるがアカウントビリティの設計手法の適用事例を示した。

アカウントビリティの設計において、自然言語で記述された社会規則からどのようにアカウントビリティの候補を抽出するかが重要なポイントとなる。まずオントロジを用いて社会規則全般の概念構造を整理し、次に条件-結果という因果関係を原因結果グラフで表現する本アプローチは、社会規則からのアカウントビリティの抽出に有効であると考えられる。

オントロジと原因結果グラフとでは作成する際の観点が異なるため、同じ社会規則の形式化を行っているにもかかわらず、表現している内容が異なる。オントロジは、社会規則中の単語やフレーズ単位という細かい粒度で構造化が行われる。原因結果グラフは、社会規則のひとつの規則番号について、条件と帰結を抽出するため、比較的粗い粒度になる。互いの良い点を利用できるように、アカウントビリティの設計手法にさらに工夫を加えていく予定である。

オントロジの使用法については課題が残っている。本稿では、主にオントロジをスケッチとしての概念の構造化を行うために使用している。そのため、論理による概念構造の表現が弱い。RDFスキーマやOWLによる記述も検討していきたい。

6 おわりに

本論文では、自然言語で記述された社会規則から、オントロジ作成および原因結果グラフ作成を経て、既存の Law-Defined システムのユースケース図にアカウントビリティユースケースを追加するアプローチによるアカウントビリティの設計手法を提案した。また、大学の履修管理システムを例に、簡単なケーススタディを行った。

本手法の事例への適用は試行的なものであり、他のシステムに対しても適用実験を行い、手法の有効性を確認していく予定である。

謝辞

本学知識科学研究科 池田 満 教授には、オントロジに関してご助言をいただきました。インテック・

ウェブ・アンド・ゲノム・インフォマティクス株式会社 堀 雅和 氏ならびに川添 恭平 氏には、議論において有益なコメントをいただきました。感謝致します。

本研究は、文部科学省科学研究費補助金(21世紀COE特別研究員奨励費)研究課題「自己説明性および進化容易性を有する電子自治体シミュレータに関する研究」(採用年度・受付番号:16・54141)の交付を受けて行われた。

参考文献

- [1] 片山卓也: 特集 21世紀卓越した情報研究拠点プログラムの目指す研究(後編) 検証進化可能電子社会 — 情報科学による安心な電子社会の実現 —, 情報処理, Vol. 46, No. 5, pp. 515-521 (2005).
- [2] 早坂良, 藤枝和宏, 落水浩一郎: 履修管理システムにおけるアカウントビリティおよび進化容易性を実現するソフトウェアアーキテクチャ, 信学技報 SS2005-32, pp. 49-54 電子情報通信学会ソフトウェアサイエンス研究会 (2005).
- [3] 早坂良, 藤枝和宏, 落水浩一郎: アカウントビリティおよび進化容易性を持つ履修管理システムの設計, 日本ソフトウェア科学会 第22回大会, CD-ROM (2005).
- [4] 早坂良, 堀雅和, 藤枝和宏, 落水浩一郎: アカウントビリティおよび進化容易性を持つソフトウェアアーキテクチャと3層モデルとの対応, 情報研報 2005-SE-150, pp. 1-8 情報処理学会ソフトウェア工学研究会 (2005).
- [5] 大島稔彦(編): 法令起案マニュアル, 株式会社ぎょうせい (2004).
- [6] アカウ ン タ ビ リ テ ィ (accountability) <http://www.pref.hiroshima.jp/soumu/kikou/jouhou/keyword/keyword4.html>.
- [7] Elmendorf, W. R.: Functional Analysis Using Cause-Effect Graphs, in *SHARE XLIII*, pp. 567-577, New York: SHARE (1974).
- [8] Myers, G. J.: ソフトウェアの信頼性 —ソフトウェア・エンジニアリング概説—, 近代科学社 (1977), 有澤 誠 訳.
- [9] 北陸先端科学技術大学院大学: 平成17年度履修案内, 学生課教務係 編集 (2005).