

AWS における電子帳票の互換性決定の実装検討

安齋 太一朗[†] 平本 真道[‡] 大谷 真[‡]湘南工科大学[†]

1. はじめに

AWS ミドルウェア [1] では、動的モデル協調層において、異なる 2 つのシステム間で使用される電子帳票の互換性を動的に決定するメカニズムが必要である。本研究では、この電子帳票間の関係が RDF/OWL 言語の等価(equivalentClass)、包含(subClassof)、排他(disjointWith)によるクラス間関係によって定義されているものとし、Web 上に点在するそれらに対して Web をトラバースする形で収集し、繋ぎ合わせるメカニズムを提案する。

2. 自律型 Web サービス (AWS)

現在の Web サービスを用いた電子商取引は、そのシステムを構築する際に取引に参加する企業の間で全体のビジネスプロセスモデル(一連の取引手順や、メッセージの送受信手段やフォーマットの定義)を詳細に決定しなければならない。従って、この事前決定に従ったビジネスプロセスモデルを持たない企業はこの取引に参加することができず、また、中小企業間や個人間で、逐一事前に取引モデルの詳細な決定を行うことは現実的に不可能である。

AWS ミドルウェアは、企業間を横断したビジネスプロセスモデルの取り決めを前提とせず、システムがインターネット内で遭遇した際に動的にビジネスプロセスモデルを変更(動的モデル協調 [2])することで異なるシステム間でも Web サービスを用いた商取引を可能にし、企業の自由なシステム設計と多様な商取引メッセージの交換を可能にすることを狙いとしている [1]。

3. 電子帳票の互換性決定

動的モデル協調ではビジネスプロセスモデルを取引における処理の実行順序を定義したもの [2] であるとしている。電子帳票は、この中でオペレーションから発せられる相手企業への定型

メッセージフォーマットのことである。Web サービスにおいて、電子帳票は企業間で統一のフォーマットを使用する必要があり、記述内容の意味が共通でも表現や書式が異なればコンピュータには異なるフォーマットとして扱われてしまう。電子帳票の互換性決定は、電子帳票に対して RDF/OWL 言語によるメタデータを付与、インターネット上に存在する電子帳票間の関係を探索・収集し推論を行うことで Web サービスにおける企業間の電子帳票が互換か否かを決定する。

3.1. 推論の前提

本研究では、電子帳票の互換性決定を実装するにあたって以下のような前提を用意する。

- ▶ 電子帳票は nameSpace と name によって表わされる。
- ▶ nameSpace と name が与えられると、それらに対する URI (nameSpace, name) がただ 1 つ定まるものと仮定する。この URI をフォーマット URI と呼ぶ。
- ▶ 関連情報とは主語と目的語にフォーマット URI が含まれる RDF トリプルである。
- ▶ 関連情報の述語は、本研究では equivalentClass, subClassof, disjointWith のいずれかに限る。
- ▶ 関連情報は URL を持つ。この URL を関連情報 URL と呼ぶ。関連情報 URL を HTTP/GET するとその関連情報の内容にアクセスできる。1 つの関連情報 URL には 2 つ以上の関連情報が含まれてもよい。ただし、本研究では、当面 1 つの URL には 1 つの関連情報が含まれている、と制約を付与する。
- ▶ あるフォーマット URI (f) が与えられたとき、f を主語(目的語)に持つような関連情報を含む関連情報 URL の集まりを検索することができる。これを関連情報検索エンジンと呼び、その実装方法は別途検討する。現時点では、関連情報検索エンジンに対して主語と述語、もしくは述語と目的語をインプットすることで、その二語を満たす関連情報を持つ関連情報 URL のリストをアウトプットものとする。

Automatic Compatibility Determination of Electronic Formats for AWS

[†]Taichiro Anzai, [‡]Masamichi Hiramoto, [‡]Makoto Oya
Shonan Institute of Technology

- 推論を行う対象は企業の保持する電子帳票を表すフォーマット URI の包含関係のみとする。
- 2 種類の電子帳票(out(送信), in(受信))間の互換性とは out 側のフォーマット URI (f_out) が in 側のフォーマット URI (f_in) に含まれるか否かによって決定する。これを out 側から in 側への互換性と表現しそれぞれを集合関係と対応させると、f_out は f_in に互換性有り (True) f_out = f_in、及び f_out ⊆ f_in。f_out は f_in への互換性無し (False) f_out ≠ f_in。判定不能 (Unknown) f_out ⊇ f_in となる。
- また、共通のフォーマット URI (f_v) を介した以下の関連情報 R1, R2 の結合によっても互換性の有無を判断する。結合した関連情報による決定の一覧を表 1 に示す。

なお、各記号は以下の略記で示した。

{R1 | S(R1)=f_out, P(R1)=relation1, O(R1)=f_v}
 {R2 | S(R2)=f_v, P(R2)=relation2, O(R2)=f_in}
 compatible = R1(f_out, relation1, f_v)
 and R2(f_v, relation2, f_in)
 主語:S(R) 述語:P(R) 目的語:O(R)
 =:equivalentClass, ≠:disjointWith
 ⊆, ⊇:subClassof

表 1 互換性決定一覧

S(R1)	P(R1)	O(R1),S(R2)	P(R2)	O(R2)	compatible
f_out	=	f_v	=	f_in	True
f_out	=	f_v	⊆	f_in	True
f_out	⊆	f_v	=	f_in	True
f_out	⊆	f_v	⊆	f_in	True
f_out	=	f_v	≠	f_in	False
f_out	⊆	f_v	≠	f_in	False
f_out	⊇	f_v	≠	f_in	False
f_out	≠	f_v	=	f_in	False
f_out	≠	f_v	⊇	f_in	False
f_out	=	f_v	⊇	f_in	Unknown
f_out	⊆	f_v	⊇	f_in	Unknown
f_out	⊇	f_v	=	f_in	Unknown
f_out	⊇	f_v	⊆	f_in	Unknown
f_out	⊇	f_v	⊇	f_in	Unknown
f_out	≠	f_v	⊆	f_in	Unknown
f_out	≠	f_v	≠	f_in	Unknown

3.2. 推論

前述の関連情報検索エンジンへのインプットは search(S(R), P(R), O(R)) で行い、フォーマット URI のインプットを”f”、空欄のインプットを”none”とする。一度の検索に対する検索エンジンへのインプットは以下の 6 通りである。

search(f, equivalentClass, none)

search(f, subClassof, none)
 search(none, subClassof, f)
 search(f, disjointWith, none)
 search(none, disjointWith, f)

これらを 1 つの検索セットとして getGraph(f) という 1 つの処理とみなして図 1 のフローチャートで探索を繰り返す。

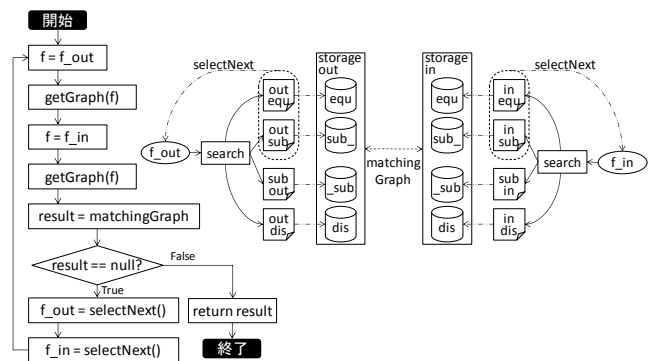


図 1 推論フローチャート

右図の out storage, in storage はそれぞれ search() によって得られた関連情報 URL から関連情報を取得し、その中の空欄部分に相当するフォーマット URI を格納する。左図の matching-Graph は out storage と in storage を比較し、共通のフォーマット URI を発見、表 1 に対応する判定を戻り値として result に渡す処理である。左図の selectNext は推移律により、関係が元のフォーマット URI と保つことのできるフォーマット URI を一覧の中から選択し、次のインプットとするための処理である。

4. まとめ

本論では 2 社間の電子帳票が互換か否かを判定するための前提を定義し、推論を実装する検討を行った。今後の課題として、帳票の記述内容へのメタデータ付与方法の定義。推論エンジンの作成と実装、実装に伴った推論を停止するための機構の検討、及び、推論を行う対象の場の定義が挙げられる。

本研究は科研費(21500110)の助成を受けたものである。

5. 参考文献

- [1]Oya, M., Ito, M, and Kimura, T. :”Middleware for the Autonomous Web Service(AWS)”, IFIP I3E, Software Services for e-World, Springer, pp. 5-16, November, 2010
- [2]大友, 大谷, 自律型 Web サービスにおける複数システム間での動的モデル強調, FIT2011 第 10 回情報科学技術フォーラム, R0-012