

I-6 データマイニングのためのセンサと構造物情報を包含する

A&A データモデルの開発

Development of A&A Data Model Incorporating Sensor and Structural Information for Data Mining

吉田善博¹・矢吹信喜²

Yoshihiro YOSHIDA, Nobuyoshi YABUKI

抄録: 将来的に、センサは様々な施設に大量に設置されるようになると予想される。本研究では、センサが設置されたオブジェクトや、センサ周囲の環境等の、コンテキストに関する情報を貯蔵するための、柔軟性に優れたデータモデルとして、A&A (Addition & Absorption) データモデルを開発した。さらに、コンテキストに関する情報を、A&A データモデルを適用したリレーショナルデータベース (A&A データベース) に格納し、本データモデルの有用性の検証を行った。

Abstract: More and more sensors will be installed in various facilities for their monitoring in the future. In this research, the Addition & Absorption (A&A) data model was developed to store contextual information regarding the object and the environment where sensors are installed. Then, contextual data was stored in a relational database based on the A&A data model and the feasibility and practicability of A&A data model was verified.

キーワード: データベース, プロダクトモデル, センサ, データマイニング, 関係データベース, IFC

Keywords : Database, product model, sensor, data mining, relational database, IFC

1. はじめに

センサの無線化, 小型化, 低価格化により, あらゆる物, 人, 環境の様々な情報を取得・利用可能なユビキタスな社会が形成されつつある。このような社会が実現されると, 膨大な量のデータがコンピュータ内に蓄えられるようになるが, データ量が多すぎるため, データ内に紛れ込んだ重要な情報を, みすみす見逃してしまうといったことが起こると懸念される。

膨大な量のデータから, 効率的に有用な知識を発見するための手法としてデータマイニングがあるが, センサデータのほとんどは数値データであるため, 単にセンサデータのみを用いてデータマイニングを行っただけでは, 有意な知識を発見できる可能性は低いものと考えられる。また, もし知識が得られたとしても, 単なる数値情報でしかない為, 人間にとって理解し難い結果しか得られないであろう。従って, センサデータから有意な知識を発見するためには, センサデータのみならず, センサが設置された構造物等のオブジェクトに関する情報や, センサ周囲の環境に関する情報,

すなわちコンテキストのデータを加味してデータマイニングを行う必要があるものと考えられる。

構造物の情報に関するデータモデルすなわちプロダクトモデルとしては, 国際標準 ISO10303 の STEP (STandard for the Exchange of Product model data)¹⁾がある。また, 建築の業界標準としては, IAI (International Alliance for Interoperability) の IFC²⁾ (Industry Foundation Classes) 等がある。センサに関するデータモデルとしては, Open Geospatial Consortium³⁾によって策定が行われている Sensor Web Enablement (SWE) standards の SensorML⁴⁾や Observation and Measurements⁵⁾ 等がある。また, 米国において Peng, Law ら^{6), 7)}によって NEES (George E. Brown, Jr. Network for Earthquake Engineering Simulation) のために開発された NEES Reference Data Model がある。さらに, 我が国では, 矢吹ら^{8), 9)}によって, 独立行政法人防災科学技術研究所の E-Defense のために, イベントに基づくスタースキーマ¹⁰⁾を利用して EDgrid データモデル^{8), 9)}が開発された。

しかしながら, センサデータを管理するためのデー

1 : 学生会員 工修 大阪大学大学院工学研究科 環境・エネルギー工学専攻 博士後期課程 環境設計情報学領域

2 : 正会員 Ph.D. 大阪大学大学院工学研究科 環境・エネルギー工学専攻 教授

(〒565-0871 大阪府吹田市山田丘 2-1, Tel :06-6879-7660, E-mail : yabuki@see.eng.osaka-u.ac.jp)

タモデルの開発と、構造物の形状データや属性情報等を管理するためのデータモデルの開発は別々に行われているのが現状である。また、データマイニングの多くのアルゴリズムは、表形式のデータフォーマットを対象としているが、センサやコンテキストのデータの形式は様々である。さらに、センサの適用範囲は、今後さらに拡大すると予想されることから、データマイニングのためのセンサデータモデルには、あらゆるコンテキストのデータに対応可能な、高い柔軟性が求められる。

そこで本研究では、データマイニングと親和性が高いと考えられるリレーショナルデータモデルの概念に基づき、センサとコンテキストに関する情報を広く貯蔵可能なデータモデルとして、A&A (Addition & Absorption) データモデルを開発した。A&A データモデルの特徴は、従来の RDBMS (Relational DataBase Management System) で、スキーマ情報として管理されていたテーブルあるいは属性の情報を、インスタンスとして貯蔵し、テーブル及び属性をインスタンスの組み合わせによって仮想的に生成することである。この方法により、貯蔵しているインスタンスに変更が必要な場合でも、スキーマ情報を変更することなく、テーブルあるいは属性の情報を追加 (addition) しても、データベースのスキーマ情報を変更せずに、うまく吸収 (absorption) するという意味で、A&A (Addition & Absorption) データモデルと命名した。A&A データモデル開発後は、A&A データモデルを適用したリレーショナルデータベース (A&A データベース) 内に、センサのコンテキストに関する情報を格納し、本データモデルの有用性の検証を行った。

2. データウェアハウス

データマイニングは、大きく分けて、データウェアハウス構築、前処理、パターン発見、知識獲得の4つのプロセスから構成される。データウェアハウス構築のプロセスでは、データマイニングに必要なデータを収集し、一箇所に集めて管理する。このことをデータハウジングと呼び、集められたデータ集合をデータウェアハウスという。データウェアハウジングにおいては、収集元のデータが、複数のデータベースに分かれて貯蔵されていることや、データによって形式が異なることがある。従って、データウェアハウスを構築するためには、データを収集するだけでなく、データ形式の統一も必要になる場合がある。

データウェアハウスのための代表的なデータモデルとしてスタースキーマ¹⁰⁾がある。このデータモデルは、1つの事実テーブルと複数の次元テーブルから構成さ

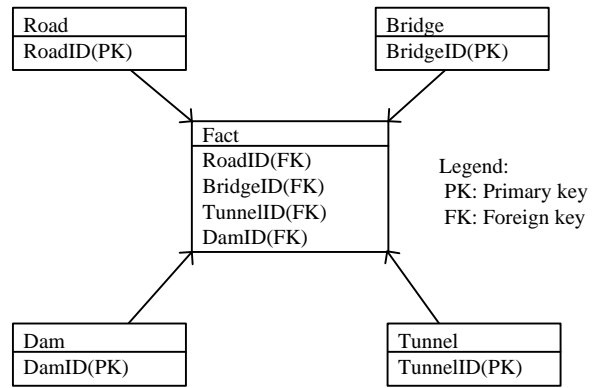


図-1 スタースキーマの例

れ、全ての次元テーブルは事実テーブルとのみ関係している。そのため事実テーブルを中心とする星形となる。図-1のFactテーブルは、事実テーブルの例であり、Bridge、Tunnel、Dam、Road テーブルは、次元テーブルの例である。スタースキーマの特徴は、テーブル間の関係を最小限に抑えることができ、柔軟性・拡張性に優れたデータモデルを作成できることである。また、テーブル間の関係が単純なため、ユーザーが容易にデータの構造を理解でき、使い易い。EDgrid データモデル^{8), 9)}は、このスタースキーマを利用して開発された。

3. A&A データモデル

(1) センサデータとコンテキストデータのためのデータモデル

将来的に、センサは様々な施設に大量に設置されるようになると予想される。従って、データマイニングのデータ基盤として、センサ及びコンテキストのデータを貯蔵するデータウェアハウスを構築するためには、新しいタイプのインスタンスの追加が必要になった場合でも柔軟に対応可能なデータモデルが必要になるものと考えられる。また、データマイニングの多くのアルゴリズムは、表形式のデータフォーマットを対象としていることから、データウェアハウスのデータモデルは、リレーショナルデータモデルが適切であると考えられる。

一般に、リレーショナルデータモデルは、初期の段階で作成した冗長性のあるテーブル群に対して正規化を行い、データの整合性を保った形で冗長性を排除し、データを効率的に管理できるように設計される。しかしながら、センサのコンテキストの場合、必要なテーブルをデータモデル設計時に全て定義するのは難しい。必要なテーブルを必要に応じて追加していく方法も考えられるが、既存のテーブルの大幅な修正が必要になる可能性もあり、効果的な方法とは言えない。

データウェアハウスのための代表的なデータモデル

としてスタースキーマがあるが、コンテキストのデータの種別は多種多様なため、事実テーブルの周りには、多くの次元テーブルが配置されるようになるものと考えられる。その結果、次元テーブルと事実テーブルの間の関係の数が膨大になり、スタースキーマの特徴である、データの構造の分かり易さや、ユーザーにとっての使い易さが、著しく低下することが懸念される。

(2) A&A データモデルの開発

従来の RDBMS では、まず、テーブル、属性、テーブル同士の関係等に関する情報をスキーマ情報として定義し、その後、スキーマ情報に基づいて、インスタンスがデータベース内に格納される。RDBMS は、厳密なスキーマ情報に基づいてインスタンスがデータベース内に貯蔵されているため、堅牢性が高い。しかしながら、貯蔵しているインスタンスに変更があった場合は、まず、スキーマの情報を変更し、その後、インスタンスを再登録しなければならないため、煩雑な作業が必要となる。センサの設置環境の拡大に伴って、センサのコンテキストの情報は増加していくものと考えられることから、従来の RDBMS の堅牢性を多少緩和し、インスタンスの変更に対する柔軟性を向上させる必要があるものと考えられる。

そこで本研究では、センサとコンテキストに関する情報を広く貯蔵可能なリレーショナルデータモデルとして、A&A データモデルを開発した。図-2 に、A&A データモデルの基盤となるテーブル構成を示す。A&A データモデルでは、スキーマ情報を、Element テーブルと Attribute テーブルのインスタンスとして格納し、これら2つのテーブルのキーの値を Value テーブル内で組み合わせによって、テーブルを仮想的に生成する(図-3)。

(3) ユーザーインターフェースの開発

A&A データモデルの実用性を検討する為に、A&A データベースと、従来のリレーショナルデータベース(従来型データベース)の比較を行った。その結果、A&A データベースについて以下の3つの問題が明らかになった。

- 従来型データベースでは、まず、テーブル及び属性の指定し、その後、インスタンスをデータベース内に格納する。一方、A&A データベースでは、インスタンスの格納時に、ElementID と AttributeID を同時に関連付けなければならないため、入力ミスが発生する可能性が高い。
- 従来型データベースでは、表形式でインスタンスを格納できるが、A&A データベースでは出来ない。
- 従来型データベースでは、主キーと外部キーの関係により、複数のテーブルを一つのテーブルとして結合出来るが、A&A データベースでは出来ない。上述の問題点を解決するために、本研究では、A&A

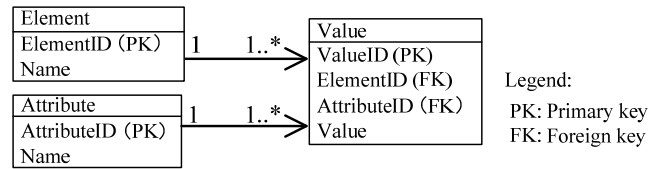


図-2 A&A データモデルの基盤となるテーブル構成

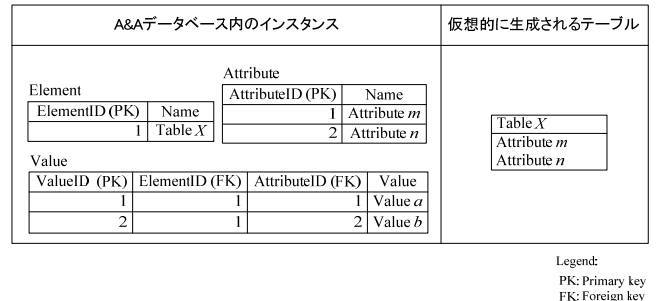


図-3 仮想的に生成されるテーブルの例

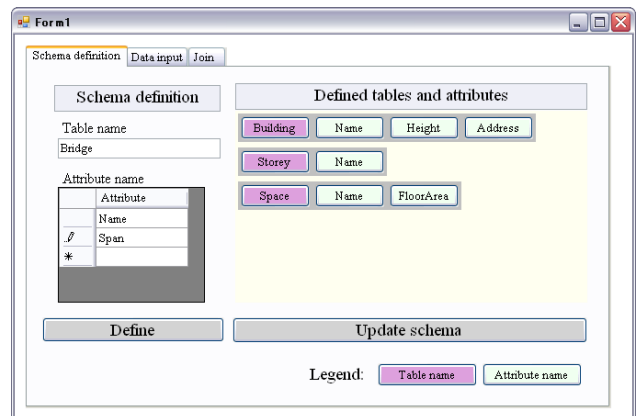


図-4 テーブル構造の確認及びテーブル・属性定義用フォーム

データベース用のユーザーインターフェース (UI) を開発した。UI の開発には、Microsoft Visual Basic 2008 を用いた。

a) スキーマの定義と参照

A&A データベースでスキーマ情報を参照する場合、テーブル名と属性名が列挙されるため、テーブルの構造を把握し難い。そこで、テーブル構造を直感的に理解できるフォームを UI 内に作成した(図-4)。また、テーブルと属性定義用のテキストボックスを設け、スキーマ情報を確認しながら、テーブルや属性を追加できるようにした。

b) インスタンスの格納

A&A データベースでは、インスタンスの格納時に、ElementID と AttributeID を同時に関連付けなければならないため、入力ミスが発生する可能性が高い。そこで、表形式でインスタンスを格納するためのフォーム

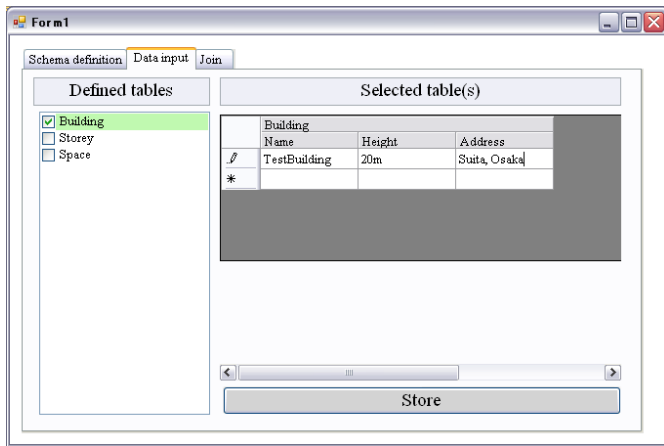


図-5 インスタンスの格納を表形式で行うためのフォーム

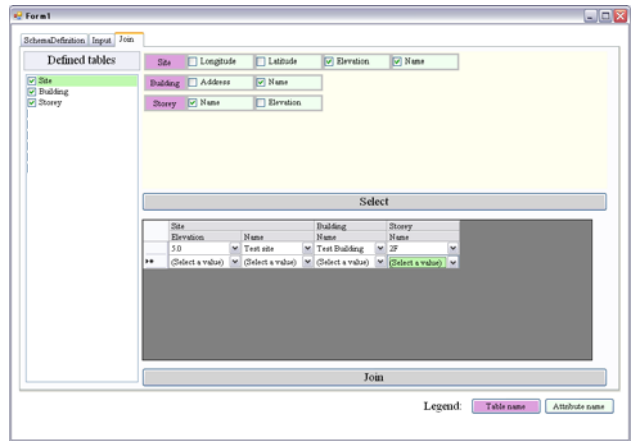


図-7 テーブルの結合用フォーム

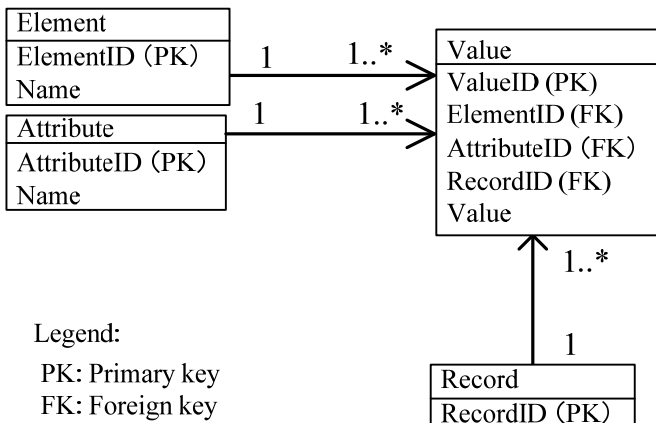


図-6 A&A データモデルのテーブル構成

を UI 内に作成した (図-5)。図-5のフォームを用いることで、インスタンスの格納時に、フォーム内で選択したテーブルに対応した ElementID と AttributeID が、バックグラウンドで自動的に関連付けられる。従って、ユーザーはインスタンスのみを入力するだけでよい。

c) レコードの結合

テーブルの結合を、A&A データベースで実現するために、図-2に Record テーブルを追加し、A&A データモデルのテーブル構成を図-6に変更した。その結果、Value テーブル内で、異なる ElementID に同じ RecordID を保持させることによって、レコードの結合が可能となった。また、図-7のフォームを作成し、ユーザーがスキーマ情報を確認しながらレコードの結合を簡便に行えるようにした。レコードの結合は、ユーザーがリストから値を選択する操作のみで行えるようにした。

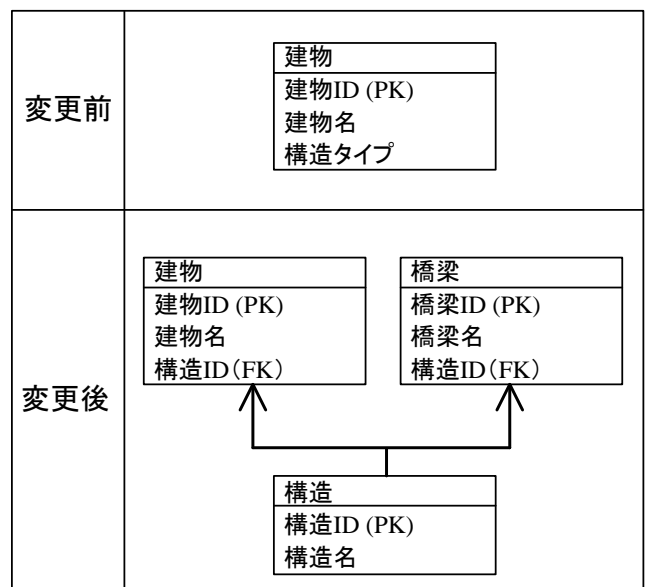


図-8 変更前と変更後のデータモデル (従来型データベース)

4. A&A データモデルの有用性の検証

(1) 検証方法

A&A データモデルの有用性を検証するために、データモデルの変更が必要な事例を用いて、A&A データベースと従来型データベースの比較を行った。以下に事例の内容を示す。

- ① データベース内には、「建物」テーブルが既に定義されている。また、「建物」テーブル内の属性として、「建物ID」、「建物名」、「構造タイプ」が定義されている。
- ② ①と同じデータベース内に、新しく「橋梁」テー

	A&Aデータベース内のインスタンス	テーブルの生成方法	仮想的に生成されるテーブル																																																													
変更前	Element <table border="1"> <tr><th>ElementID (PK)</th><th>Name</th></tr> <tr><td>1</td><td>建物</td></tr> </table> Attribute <table border="1"> <tr><th>AttributeID (PK)</th><th>Name</th></tr> <tr><td>1</td><td>建物ID</td></tr> <tr><td>2</td><td>建物名</td></tr> <tr><td>3</td><td>構造タイプ</td></tr> </table>	ElementID (PK)	Name	1	建物	AttributeID (PK)	Name	1	建物ID	2	建物名	3	構造タイプ	Value <table border="1"> <tr><th>ValueID (PK)</th><th>ElementID (FK)</th><th>AttributeID (FK)</th><th>RecordID (FK)</th><th>Value</th></tr> <tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>2</td><td>1</td><td>2</td><td>1</td><td>ビルA</td></tr> <tr><td>3</td><td>1</td><td>3</td><td>1</td><td>鉄筋コンクリート</td></tr> </table>	ValueID (PK)	ElementID (FK)	AttributeID (FK)	RecordID (FK)	Value	1	1	1	1	1	2	1	2	1	ビルA	3	1	3	1	鉄筋コンクリート	<table border="1"> <tr><td>建物</td></tr> <tr><td>建物ID (PK)</td></tr> <tr><td>建物名</td></tr> <tr><td>構造タイプ</td></tr> </table>	建物	建物ID (PK)	建物名	構造タイプ																									
ElementID (PK)	Name																																																															
1	建物																																																															
AttributeID (PK)	Name																																																															
1	建物ID																																																															
2	建物名																																																															
3	構造タイプ																																																															
ValueID (PK)	ElementID (FK)	AttributeID (FK)	RecordID (FK)	Value																																																												
1	1	1	1	1																																																												
2	1	2	1	ビルA																																																												
3	1	3	1	鉄筋コンクリート																																																												
建物																																																																
建物ID (PK)																																																																
建物名																																																																
構造タイプ																																																																
変更後	Element <table border="1"> <tr><th>ElementID (PK)</th><th>Name</th></tr> <tr><td>1</td><td>建物</td></tr> <tr><td>2</td><td>橋梁</td></tr> </table> Attribute <table border="1"> <tr><th>AttributeID (PK)</th><th>Name</th></tr> <tr><td>1</td><td>建物ID</td></tr> <tr><td>2</td><td>建物名</td></tr> <tr><td>3</td><td>構造タイプ</td></tr> <tr><td>4</td><td>橋梁ID</td></tr> <tr><td>5</td><td>橋梁名</td></tr> </table>	ElementID (PK)	Name	1	建物	2	橋梁	AttributeID (PK)	Name	1	建物ID	2	建物名	3	構造タイプ	4	橋梁ID	5	橋梁名	Value <table border="1"> <tr><th>ValueID (PK)</th><th>ElementID (FK)</th><th>AttributeID (FK)</th><th>RecordID (FK)</th><th>Value</th></tr> <tr><td>1</td><td>1</td><td>1</td><td>1</td><td>101</td></tr> <tr><td>2</td><td>1</td><td>2</td><td>1</td><td>ビルA</td></tr> <tr><td>3</td><td>1</td><td>3</td><td>1</td><td>鉄筋コンクリート</td></tr> <tr><td>4</td><td>1</td><td>3</td><td>1</td><td>201</td></tr> <tr><td>5</td><td>2</td><td>4</td><td>1</td><td>橋梁B</td></tr> <tr><td>6</td><td>2</td><td>5</td><td>1</td><td>トラス</td></tr> </table>	ValueID (PK)	ElementID (FK)	AttributeID (FK)	RecordID (FK)	Value	1	1	1	1	101	2	1	2	1	ビルA	3	1	3	1	鉄筋コンクリート	4	1	3	1	201	5	2	4	1	橋梁B	6	2	5	1	トラス	<table border="1"> <tr><td>建物</td></tr> <tr><td>建物ID (PK)</td></tr> <tr><td>建物名</td></tr> <tr><td>構造タイプ</td></tr> </table> <table border="1"> <tr><td>橋梁</td></tr> <tr><td>橋梁ID (PK)</td></tr> <tr><td>橋梁名</td></tr> <tr><td>構造タイプ</td></tr> </table>	建物	建物ID (PK)	建物名	構造タイプ	橋梁	橋梁ID (PK)	橋梁名	構造タイプ
ElementID (PK)	Name																																																															
1	建物																																																															
2	橋梁																																																															
AttributeID (PK)	Name																																																															
1	建物ID																																																															
2	建物名																																																															
3	構造タイプ																																																															
4	橋梁ID																																																															
5	橋梁名																																																															
ValueID (PK)	ElementID (FK)	AttributeID (FK)	RecordID (FK)	Value																																																												
1	1	1	1	101																																																												
2	1	2	1	ビルA																																																												
3	1	3	1	鉄筋コンクリート																																																												
4	1	3	1	201																																																												
5	2	4	1	橋梁B																																																												
6	2	5	1	トラス																																																												
建物																																																																
建物ID (PK)																																																																
建物名																																																																
構造タイプ																																																																
橋梁																																																																
橋梁ID (PK)																																																																
橋梁名																																																																
構造タイプ																																																																

Legend:
 PK: Primary key
 FK: Foreign key

図-9 変更前と変更後のデータモデル (A&A データベース)

ブルを定義する。また、「橋梁」テーブル内の属性として、「橋梁 ID」、「橋梁名」、「構造タイプ」を定義する。

(2) 検証結果

従来型データベースの、変更前と変更後のデータモデルを図-8に示す。まず、「建物」テーブルと「橋梁」テーブルには、重複した「構造タイプ」属性があるため、スキーマ情報を変更し、「構造タイプ」属性を、「構造」テーブル内の属性として定義する。次に、「建物」テーブルと「構造」テーブルを関係付けるために、スキーマ情報を変更して、「建物」テーブル内の「構造タイプ」属性を「構造 ID」属性とする。従って、「建物」テーブル内の「構造タイプ」属性に格納されていたインスタンスを、「構造 ID」属性用のインスタンスとして再登録し直す必要がある。最後に、「橋梁」テーブル及び「橋梁 ID」、「橋梁名」、「構造 ID」属性を定義するためにスキーマ情報を変更する。

一方、A&A データベースでは、Element テーブルの「Name」属性に、「橋梁」を格納し、Attribute テーブルの「Name」属性に、「橋梁 ID」と「橋梁名」を格納することによって、仮想的に「橋梁」テーブルを生成する(図-9)。尚、「橋梁」テーブルの「構造タイプ」属性には、「建物」テーブルを定義する際に、Attribute テーブルの「Name」属性に格納された「構造タイプ」を用いる。

(3) 考察

既存のテーブルと同じ属性を持つ、新しいテーブルの追加が必要な場合、従来型のデータベースでは、関連付けを行うためのテーブルを追加する必要がある。また、既存のテーブルのスキーマ情報を一部変更する必要があるため、インスタンスの再登録が必要になる。一方、A&A データベースでは、Value テーブル内で、異なる ElementID に同じ AttributeID を関係付けることで、異なるテーブルが同じ属性を保持していることを表現できる。従って、関連付けを行うためのテーブルを追加する必要がない。また、既存のテーブルのスキーマ情報を変更する必要がない。さらに、インスタンスを再登録する必要がない。従って、A&A データベースは、従来のデータベースよりも、インスタンスの変更に対して柔軟に対応可能である。

5. まとめ

本研究では、データマイニングと親和性が高いと考えられるリレーショナルデータモデルの概念に基づき、センサとコンテキストに関するあらゆる情報を貯蔵可能なデータモデルとして、A&A データモデルを開発した。また、A&A データベースと従来型のリレーショナルデータベースの比較を行い、A&A データベースの問題点を解決するためのユーザーインタフェースを開発した。さらに、A&A データベースに、センサのコンテ

クストのデータを格納し、本データモデルの有用性の検証を行った。A&A データベースは、従来のデータベースよりも、インスタンスの変更に対して柔軟に対応可能であることから、センサとコンテキストに関する情報を広く貯蔵可能であると考えられる。

今後の課題として、本モデルの有用性の更なる追加検証を実施していくことが挙げられる。また、実データを用いて A&A データベースを基盤とするデータウェアハウスを構築し、データマイニングを実施することが挙げられる。

参考文献

- 1) ISO10303, Industrial Automation Systems and Integration – Product Data Representation and Exchange, 1994.
- 2) IFC: <http://www.iai-international.org/index.html>
- 3) Open Geospatial Consortium:
<http://www.opengeospatial.org/>
- 4) SensorML: <http://vast.uah.edu/SensorML>
- 5) Observation and Measurements:
<http://www.opengeospatial.org/standards/om>
- 6) Peng, J. and Law, K.H.: A Brief Review of Data Models for NEESgrid, NEESgrid TR-2004-01, 2004.
- 7) Peng, J. and Law, K.H.: Validity and Usability of the NEESgrid Reference Data Model, NEESgrid TR-2004-44, 2004.
- 8) 矢吹信喜, 吉田善博 : EDgrid (E-Defense Grid) のデータモデルの提案, 第 61 回年次学術講演会講演概要集, 土木学会, pp.549-550, 2006. 矢吹信喜, 吉田善博 : EDgrid (E-Defense Grid) におけるデータモデルの開発, 土木情報利用技術論文集, Vol.15, pp.111-118, 2006.
- 9) T. J. ティオリー (原田勝訳) : データベースの設計, 勁草書房, 2000.

(2010. 5. 28 受付)