

# オントロジー言語 CQL と工業技術データの標準モデルへの適用

細川 晃<sup>†</sup> 村山 廣<sup>†</sup>

<sup>†</sup> (株) 東芝 研究開発センター 知識メディアラボラトリー  
〒212-8582 神奈川県川崎市幸区小向東芝町 1

E-mail: <sup>†</sup> {akira.hosokawa, hiroschi.murayama}@toshiba.co.jp

**あらまし** 我々は、これまでに SQL を公理的集合論におけるクラス論理に拡張したオントロジー言語である CQL(Class Query Language)を提案し、アサイクリックなグラフ構造を持つクラス階層に対する、より中立的で強力な言語へと拡張してきた。他方で、近年、工業技術データに関するいくつかの標準データモデルが ISO や IEC などの国際標準化機構を通して提案・発行されており、次世代の国際的な電子商取引の基盤技術として、これらの標準データモデルに従ったデータが工業界で広く利用されることが期待されている。本研究は、CQL の特徴・利点を詳説するとともに、工業データの代表的なモデルへの CQL の適用に言及し、CQL の有効性を証明することを目的とする。

**キーワード** オントロジー、公理的集合論、国際標準、工業技術データ

## Class Query Language and its application to International Standard data models for industrial data

Akira HOSOKAWA<sup>†</sup> and Hiroshi MURAYAMA<sup>†</sup>

<sup>†</sup> Knowledge Media Laboratory, Corporate Research and Development Center, Toshiba Corporation

1 Komukai-Toshiba-cho, Saiwai-ku, Kawasaki 212-8582 Japan

E-mail: <sup>†</sup> {akira.hosokawa, hiroschi.murayama}@toshiba.co.jp

**Abstract** Since the first release of Class Query Language(CQL), we have extended the CQL to make a neutral and set-theoretic ontology language that is as an extension of SQL, but at the same time, deeply rooted in the class concept in mathematical set theory. CQL provides a set of powerful functions for both manipulating and retrieving information in classes which form an acyclic graph structure. In the meantime, several International Standards that propose data models for industrial data have been developed by International Standard organizations such as ISO and IEC, and are expected to be used widely in e-business fields for the processing and transport of industrial data. This paper illustrates some typical features of CQL and explains the advantage of applying it to the data models described by the International Standards.

**Keyword** Ontology, Set theory, International Standard, Industrial data

### 1. はじめに

近年、電子商取引の拡大に伴い、企業で扱う部品・製品のデータが電子化されてきた。このような工業技術データは、企画から廃棄・再利用に至るまでの過程で様々な形で利用されており、現在の工業界における製品ライフサイクル管理 (PLM) を支える重要な要素の一つとなっている。

PLM ツールの導入などにより、企業単位では工業技術データの一元的な管理が容易になっている。その一方で、現在、企業が持つ工業技術データの多くは紙ベースで管理していたものをそのまま電子データに置き換えたものが多く、これらは企業独自に定められた仕様項目 (プロパティ) に従って記述・蓄積されている。そのため、異なる企業間でデータ交換を行う際には、

二社間のデータ構造の解釈、すなわちプロパティの対応付けや型変換、単位の換算などを行うことが必要となるが、多くの場合、これらは完全に自動化することができず、解決のために人の言語による意思の疎通や解釈を必要とする場合が多い。このように、現在の工業技術データの交換の現場では、交換対象とする企業のデータ構造の確認や変換後のデータの検証などに要する人的・時間的費用が顕著な問題となっている。

そこで、このような問題を解決するために、工業技術データの標準辞書を記述するために用いる標準データモデルが ISO (International Organization for Standardization | 国際標準化機構) や IEC (International Electrotechnical Commission | 国際電気標準会議) などの国際標準化機関を通して規格化されてきた。これら

のデータモデルは、所謂オブジェクト指向的な観点から、モノの分類概念をクラス、モノの特性記述をプロパティとして体系的に表現するための情報構造を提供している。

これまでに、これらのデータモデルを用いて、企業内で管理するデータや企業間で交換されるデータのための標準辞書（オントロジー）を作成して利用する取り組みが成されてきた。例えば国内団体では、社団法人 電子情報技術産業協会（JEITA）の EC センターが中心となって、「ECALS 辞書」と呼ばれるデータ辞書を構築・公開しており、その普及を推進する活動を行っている[1]。また既に複数企業において、実際の電子部品の電子的取引の現場で ECALS 辞書を一部または全面的に利用しており、今後多くの企業に同様の動きが広がることが期待されている。

その一方、これまではそれぞれ異なるデータモデルを用いて構築されたデータ辞書のクラスやプロパティについて、下敷きとするデータ構造に依存することなく、横断的に参照したり検索したりする手段が提供されていなかった。これに対して、現在、ISO TC184/SC4 では、それぞれのデータモデルに従って作成されたデータ辞書に記載されたクラスやプロパティの持つ意味の解決、およびそれらのデータ辞書に従って記述されたデータの発見や再利用を実現するための WEB サービスのフレームワークとして、ISO29002 の規格化が検討されている。これを利用することで、これまでに個々のデータモデルを用いて構築・活用されてきたデータ辞書についても、柔軟な検索や応用が可能になると考えられている。我々は、このような目的に対して、これまでに我々が提案してきたオントロジー言語 CQL（Class Query Language）を基本に、その一部または全部を簡素化したものの提案を考えている。

本稿では、ISO29002 のスコープと対象となる関連規格を紹介し、ISO29002 に紐付けされる辞書情報の検索手段として、CQL の適正を検討し、その機能に関する検証を通して CQL の有効性の証明と適用の是非を検討する。

## 2. ISO/IEC 標準と Web サービス

### 2.1. データモデルの国際標準

#### 2.1.1. ISO13584/IEC61360

ISO13584/IEC61360（PLIB）[2]は、製品のライフサイクルにわたる技術データを表現するための国際規格であり、データ辞書の構造を規定する Part42 に従って、製品のクラス体系や個々のクラスの実製品が持つ共通のプロパティ、およびそれらのクラスやプロパティの ID や名称、定義などのデータ辞書の情報が記述される。ここで、クラスやプロパティには ISO6523 の ICD

（International Code Designator）に基づいた BSU（Basic Semantic Unit）コードと呼ばれる ID が割り当てられ、他のものと完全に識別される。あるクラスのデータは、そのクラスが持つプロパティの ID とその値を組として、そのクラスのデータが記述される。

図 1 のように、PLIB では、上位クラスから下位クラスへのプロパティの継承に加えて、他のクラスで定義されたプロパティを再利用する Case of と呼ばれる機能が提供され、これらによって単純継承を基本とした部分多重継承的なクラス体系が形成される。また、下位クラスで共通して利用するプロパティを予め上位クラスで定義し（Visible な状態と呼ばれる）、下位クラスで必要になった時点でこのプロパティの利用を開始する（Applicable な状態と呼ばれる）ことができる。

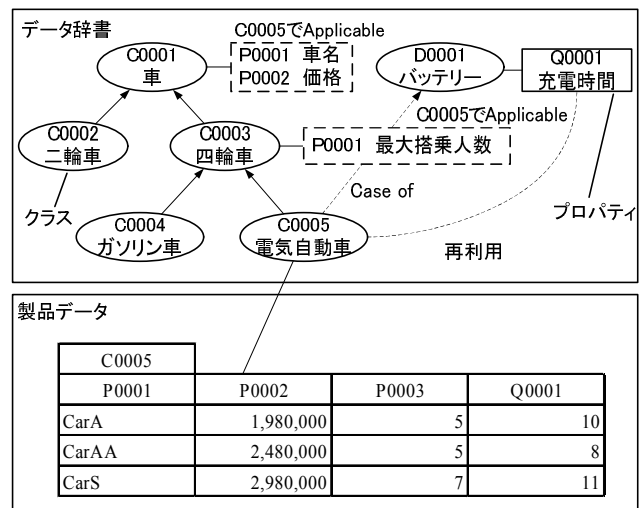


図1 PLIB のデータ辞書とそのデータの例

PLIB のデータを包括的に操作するための言語を考える場合、部分多重継承的なクラス体系を表現できる必要があるが、単純継承のみサポートした SQL や純粋な単純継承/多重継承のみサポートした ODL では、表現が容易ではない。一方、汎用製品の検索など、クラス階層を柔軟に参照できることが望ましいが、SQL や ODMG の OQL などの既存の問い合わせ言語はクラス階層の下位方向への検索しかサポートしておらず、その実現が難しい[3]。

#### 2.1.2. ISO15926

ISO15926（Oil&Gas）[4]は、石油やガス採掘プラントのあらゆる構成要素のライフサイクルデータを記述するための国際規格であり、Part2 でデータ辞書の構造が規定される。PLIB と同様に、Oil&Gas のデータ辞書でも、オブジェクト指向的に上位クラスから下位クラスへプロパティが継承されるが、部分多重継承ではなく完全な多重継承となる点が PLIB と異なる。

Oil&Gas では、Part4 で Reference Data Library (RDL) と呼ばれるプロセスプラントのための抽象的なクラスとプロパティのセットを提供しており、新たに独自のクラスを生成する際にこれを参照して再利用することができる。

Oil&Gas のデータを包括的に操作する言語を考える場合、多重継承の表現や検索が可能である必要がある。

### 2.1.3. ISO22745

ISO22745 (OTD) は、人や組織、場所など様々なものの概念を記述するための手段を規定する国際規格である。OTD は、図 2 に示すように、主に Part10 で規定されているデータ辞書と、データのスキーマを表現するための仕組みとして Part30 で規定されている Identification Guide (IG) と、Part40 で規定されているカタログとから構成される。

PLIB や Oil&Gas と異なり、OTD のデータ辞書はクラス階層を持たない。そのため、OTD のデータを包括的に操作する言語は、クラスやプロパティを簡潔に記述し、単純な構文でこれらのデータを検索できることが望ましい。

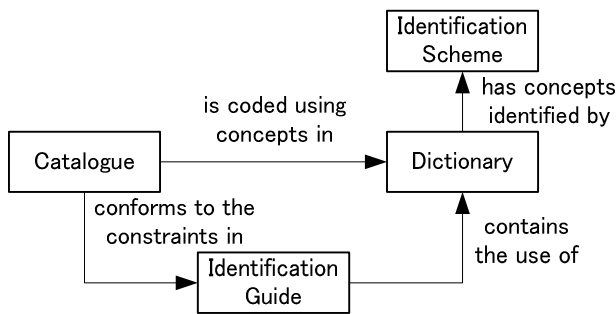


図2 OTD の基本データ構造

## 2.2. Web サービス

ISO29002 は、データ辞書を構成する概念やプロパティの解決のための Web サービス技術の仕様として構想され、現在 ISO TC184/SC4 で仕様の詳細についての議論・検討が行われている。その目的は、重厚になりがちなデータ辞書全体を交換したり参照したりしなくても、クラスやプロパティの必要な情報を取得することである。これにより、データ辞書利用者が必要とする情報のみを容易に取得したり、既存のデータ辞書のクラスやプロパティの再利用を促進したりすることが期待される。

ISO29002 の構造は、機能別に 3 段で構成することが検討されている。まず、第 1 段では、辞書の提供者等の管理情報を管理する。次に、第 2 段では、用語とその ID と定義とを基本的な意味要素として蓄積し、各

辞書の記述に用いられる基本的な意味要素の検索を可能にし、またそれらの要素を記述する辞書への経路 (path) を示す機能を提供する。最後に、第 3 段では、必用に応じて、特定の辞書モデルに従って構築された辞書情報を実際に取得して活用するために必要な機能や情報 (データ型、ファイル形式) を提供する。これら 3 段の機能を組み合わせることで、データ辞書要素の意味の解決のための Web サービスの仕組みが提供される予定である。

ISO29002 では、PLIB や RDL, OTD などの様々なデータモデルで記述されたデータ辞書の情報を対象としている。このとき、前述のようにデータモデル毎の構造の違いにより、SQL や OQL など既存の問い合わせ言語ではこれらの情報を十分に検索することができない。従って、これらのデータモデルの違いを吸収した、中立的な問い合わせ言語が必要となる。

そこで、このような ISO29002 の要求を満たす問い合わせ言語として、これまでに我々が提案してきたオントロジー言語 CQL の適用を検討する。

## 3. CQL

CQL[6] は、SQL を公理的集合論におけるクラス論理に拡張した宣言的で中立的なオントロジー言語である。その機能により、PLIB や RDL, OTD のいずれのデータモデルについても、データ辞書やそのデータを表現し、検索することができる。本章では、CQL の理論の基礎である公理的集合論におけるクラス論理を紹介し、これに基づいた CQL の概要および特徴を詳説する。

### 3.1. 公理的集合論におけるクラス論理

公理的集合論によると、クラス  $A$  は(3.1)式で定義される。

$$A = \{x \mid \phi(x)\} \quad (3.1)$$

ここで、クラスは複数のプロパティで記述され得る。このとき、例えば  $n$  種類 ( $1 < n$ ) のプロパティを持つクラスの場合、このクラスを構成する個々のプロパティを  $\phi_i(x)$  ( $1 \leq i \leq n$ ) とし、(3.1)式の  $\phi(x)$  は(3.2)式に展開される。

$$\phi(x) = \phi_1(x) \wedge \cdots \wedge \phi_n(x) \quad (3.2)$$

次に、クラスとサブクラスとの間におけるプロパティの継承の観点から、(3.1)式および(3.2)式の定義を用いて、クラス  $A$  のサブクラス  $A'$  は(3.3)式で定義される。

$$A' \subseteq A \Leftrightarrow \phi(x) = \phi'(x) \wedge \cdots \wedge \phi_n(x) \quad (3.3)$$

ここで、クラス  $A$  と  $A'$  およびそれらのプロパティは、 $m < n$  とし(3.4)式に従うものとする。

$$A = \{x \mid \phi(x)\}, \phi(x) = \phi_1 \wedge \dots \wedge \phi_n, \quad (3.4)$$

$$A' = \{x \mid \phi'(x)\}, \phi'(x) = \phi'_1 \wedge \dots \wedge \phi'_m$$

以上の定義に従い、数学的なクラス階層が形成される。ここで、公理的集合論では、全てのクラスは **Universal class** と呼ばれるクラスのサブクラスとして定義される。従って、あらゆるクラス階層は **Universal class** を根とした木構造になる。

一方、(3.5)式のように、2つのクラス A, B の間で積 (intersection), 和 (union), 差 (subtraction) のブール演算および直積 (product) が定義されている。

$$A \cap B = \{x \mid x \in A \wedge x \in B\}$$

$$A \cup B = \{x \mid x \in A \vee x \in B\}$$

$$A - B = \{x \mid x \in A \wedge x \notin B\}$$

$$A \times B = \{(x, y) \mid x \in A \wedge y \in B\}$$
(3.5)

なお、公理的集合論の定義により、これらの演算結果もクラスである。

### 3.2. CQL の構成

CQL のデータ構造は、図 3 に示すようにメタ辞書層、辞書層、データ層の 3 層で構成される。

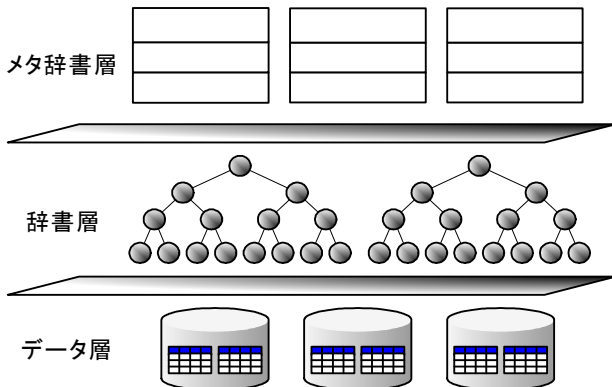


図3 CQL の構成

- メタ辞書層

データモデルの構造およびデータモデル間の概念の対応等を記述する。メタ辞書層では、自由にデータ辞書のデータモデルを定義できるが、後述の言語学的な観点から、名称や定義などの基本概念は予め共通して提供される。

- 辞書層

メタ辞書層で定義されたデータモデルに従って、クラスやプロパティなどの概念を記述する。そのため、これらの概念の構造は基本的にそれらのデータモデルに従うが、名称や定義などの情報は、いずれのデータモデルのものでも共通して参照することができる。

- データ層

辞書層で定義されたクラスに対するデータを、そのクラスが持つプロパティとその値の組み合わせで記述する。

CQL では、辞書層で記述される全ての概念は、ID によって完全に一意に識別される。ここで、言語学におけるシニフィエ (signifié: 記号内容) とシニフィアン (signifiant: 記号表現) [7]あるいはその発展形と捉えられる意味の三角形 (semantic triangle) [8]の視点から、概念は計算機メモリ空間と人間の思考空間を合わせた空間上に存在する言語に依存しないものであり、これに与えられる ID は「概念そのもの」を直接指し示すもの、あるいは抽象化された絶対アドレスと捉えられる。また、同様の視点から、概念には、これを人間が認識する際に用いる各言語に依存する「名称」、および各言語における概念を解釈するための「定義」が与えられる。すなわち、これら 3つを最小構成要素として各概念が表現される。

ここで、図 4 のように、ものの概念 (クラス) は、言語学的な識別だけではなく、クラスに結び付けられたプロパティの ID により構造的に識別されることに注意されたい。すなわち、プロパティは自身を定義したクラスをネームスコープとして持ち、一方でクラスはこれらのプロパティから自身の特徴を記述する。ここでネームスコープとは、XML のそのの如く単に名称を区別するための名前空間ではなく、ネームスコープで指し示されたクラス以下のクラス空間で意味を持つという意味の階層的な概念空間である。

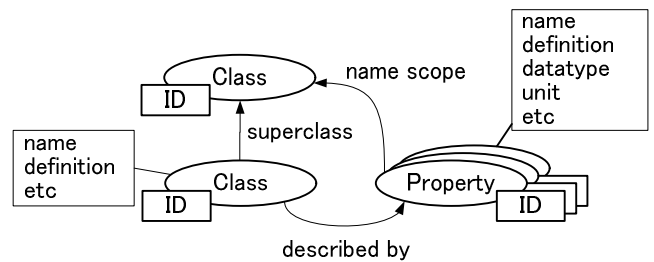


図4 クラスとプロパティの基本構造

図 5 は図 4 のクラスとプロパティの関係に従って形成されるクラス階層の例を示す。公理的集合論に従い、クラス階層は **Universal class** (universe) を根とし、プロパティの継承関係から形成されるアサイクリックな木構造となる (図 5 の各実線矢印)。

ここで、各プロパティは ID により他のプロパティと厳密に区別されるため、継承の利用だけではクラス階層上で継承関係のないクラスで定義されたプロパティを利用することができない。そこで、継承関係のないクラスで定義されたプロパティを再利用する手段と

して、プロパティの輸入機能を提供している。具体的な方法として、輸入先のクラスを指定し、更にそのクラスが持つプロパティの中から輸入するものを選択する(図5のクラス C<sub>31</sub> からクラス C<sub>23</sub> に向かう破線矢印)。なお、輸入先のクラスおよび輸入するプロパティを複数選択することができる。

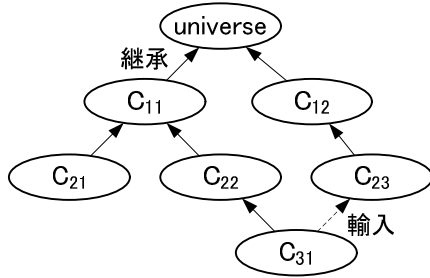


図5 クラス階層の例

### 3.3. 検索機能

CQL の検索構文は、SQL など他の多くの問い合わせ言語と同様に、Select-From-Where で構成される。このうちの最小構成は Select 句と From 句であり、これらで指定されるものおよびこれらの中での演算により導出されるものは必ず集合である。

図6は CQL の検索構文のBNF(Backus Naur Form)を表す。このうち、太字で記述されたものは、予約語を表しており、このうち <meta\_class\_signifier> はデータ辞書のメタ情報を構成する概念を示す予約語である。また、<class> および <property> はそれぞれクラスおよびプロパティの ID を、<suffix> は、多国語での表現を可能とするための言語コードなどの付加情報を表す。また、<comp\_op> は比較演算子を、<value> はプロパティの値を、<number> は整数を表す。

```

SELECT [(<meta_class_signifier> | <class>).]*
  | <property_signifier> {, <property_signifier>}
FROM [^]<class>[<scope>]
  {<class_op> [^]<class>[<scope>]
[WHERE <property_signifier> <comp_op> <value>
  {AND|OR
    <property_signifier> <comp_op> <value>
  }
[ORDER BY <property>[<suffix>] [ASC | DESC]
  {,<property>[<suffix>] [ASC | DESC]}
[RESULT_NUM (<|=>) <number>]

<meta_class_signifier> := CLASS | PROPERTY
  | ENUMERATION | SUPPLIER
  | DICTIONARY | DATATYPE
  | DOCUMENT | TABLE | UoM | *
<property_signifier> :=
[<meta_class_signifier>.,]<property>[<suffix>]
  
```

図6 CQL の検索構文

ここで、CQL の検索の特徴的な機能を実現するための <scope>, <class\_op>, 「^」については、それぞれ以降で詳説する。

#### 3.3.1. 検索範囲の指定

CQL は、他のオブジェクト指向的な問い合わせ言語と同様に、子孫クラスを含めた検索のための機能を提供する。これに加えて、継承関係における先祖クラスの検索機能を提供する。更に、プロパティの輸入関係を含めた上位クラスおよび下位クラスの検索機能も提供しており、「ある特徴を持ったクラス」の検索も容易に行うことができる。

表1は検索範囲の指定のために図6の<scope>に記述する4種類の接尾辞とそれぞれの機能を示したものであり、必要に応じてクラスのIDに続けて記述される。これらの接尾辞を利用することで、上位クラスの検索、下位クラスの検索だけでなく、それらにプロパティの輸入関係を含めるか否かを簡潔に指定することができる。

表1 検索範囲の指定に用いられる接尾辞

<scope>	機能説明
\$	指定されたクラスとその子孫クラスのみを検索
!	指定されたクラスとその先祖クラスのみを検索
*	指定されたクラスとプロパティの輸入関係(輸入元)を含めた子孫クラスを検索
%	指定されたクラスとプロパティの輸入関係(輸入先)を含めた先祖クラスを検索

表2は、図5のクラス階層に対して、これらの接尾辞を用いてインスタンスを検索した場合の、検索文とその結果指定されるクラスの例である。接尾辞を用いない場合には、指定されたクラスのみ検索対象になる。一方、接尾辞に「\$」および「!」を用いた場合には、指定されたクラスからそれぞれ下位、上位の方向に実線矢印を辿った際の経路上のクラスが選択される。一方、接尾辞に「\*」および「%」を用いた場合は、破線矢印も含めた経路上のクラスが選択される。

なお、universe はプロパティを持たないクラスであるため、通常は検索の対象とならないことに注意されたい。

表2 検索範囲を指定した検索文の例

検索文	指定されるクラス
SELECT * FROM C <sub>12</sub>	C <sub>12</sub>
SELECT * FROM C <sub>12</sub> \$	C <sub>12</sub> C <sub>23</sub>
SELECT * FROM C <sub>31</sub> !	C <sub>11</sub> C <sub>22</sub> C <sub>31</sub>
SELECT * FROM C <sub>12</sub> *	C <sub>12</sub> C <sub>23</sub> C <sub>31</sub>
SELECT * FROM C <sub>31</sub> %	C <sub>11</sub> C <sub>22</sub> C <sub>23</sub> C <sub>31</sub>

### 3.3.2. クラス間のブール演算を用いた検索

クラス階層の検索において、3.1項で述べた公理的集合論の集合演算を提供する機能として、FROM 句内での複数のクラス間の積、和、差の3種類のブール演算機能を提供する。ここで、これらの演算機能は3.3.1項の検索範囲を指定する各演算子と同時に使用することができる。これにより、複雑な検索範囲の指定を集合論的な観点から容易に行うことができる。

表3は、図6の<class\_op>に記述するクラス間の集合演算のための演算子を示したものである。

表3 クラス間の演算に用いられる演算子

演算の種類	<class_op>
積	AND または &
和	OR または +
差	-

表4は、図5のクラス階層に対して、クラス間のブール演算機能および3.3.1項で説明した検索範囲を指定する機能を利用してインスタンスを検索した場合の、検索文とその結果指定されるクラスの例である。この例では、3.3.1項で説明した検索範囲を指定するための接尾辞の中から「\*」を利用し、クラス C<sub>22</sub> とその下位クラスおよび C<sub>23</sub> とその下位クラスとの間で3つのブール演算を行っている。ここで、C<sub>22</sub>\*={C<sub>22</sub>, C<sub>31</sub>}、C<sub>23</sub>\*={C<sub>23</sub>, C<sub>31</sub>}であるから、それぞれのブール演算により、指定されるクラスは表4ようになる。

表4 クラス間のブール演算の例

検索文	指定されるクラス
SELECT * FROM C <sub>22</sub> * & C <sub>23</sub> *;	C <sub>31</sub>
SELECT * FROM C <sub>22</sub> * + C <sub>23</sub> *;	C <sub>22</sub> C <sub>23</sub> C <sub>31</sub>
SELECT * FROM C <sub>22</sub> * - C <sub>23</sub> *;	C <sub>22</sub>

なお、リレーショナルモデルではテーブル間の演算が定義されているが、これは直積演算のみであり、ブール演算については基本的に対象となるテーブル間でカラムの数や順序が等しい場合に限られる。これに対し、CQLでは演算を行う各クラスが異なるプロパティを持っていても良い。これは、CQLで行う演算が純粋に集合論に基づいていることを示している。また、CQLでは、SQLと同様の構文を用いて次の式のようにクラス間の直積を実現している。

```
SELECT * FROM C22, C23;
```

### 3.3.3. メタデータ検索

FROM 句でクラスのIDを記述する際に、その接頭辞に「^ (caret)」演算子を付け加えることにより、指定されるクラスのメタデータを検索する機能を提供する。ここで、この演算子は3.3.1項のクラス階層での検索範囲の指定機能、および3.3.2項のクラス間のブール演算機能と併用することができる。従って、これらの機能によって、クラス階層においてデータおよびそのメタデータを柔軟に検索することができる。

式は、メタデータ検索の検索文の例である。この例では、検索範囲の指定とクラス間のブール演算によって選択されたクラス C<sub>31</sub> のメタデータを結果として返す。

```
SELECT * FROM ^C22* & ^C23*
```

### 3.4. 検索の例

クラス階層上での検索の例として、クラスおよびプロパティの情報の典型的な例とそれに対するCQLの検索文を紹介する。

例1) ミニノート PC という名のクラスのIDを取得する

```
SELECT CLASS.IDENTIFIER
FROM ^ROOT*
WHERE CLASS.NAME.JA = 'ミニノート PC';
```

ここで、ROOTはUniverseより下のデータ辞書の根となるクラスの総称である。この検索文を実行することで、名称にミニノート PCを持つクラスのIDENTIFIERが返される。ここでは、この検索により、AAA000とBBB010という2種類のIDが返されたとする。

例2) 例1で調べたミニノート PCに類されるクラスが持つプロパティの情報を検索する

```
SELECT PROPERTY.*
FROM ^AAA000* OR ^BBB010*
```

例1で取得した、ミニノート PCに関する2つのク

ラスを対象に、これらの持つプロパティを検索する。このクエリでは、ミニノート PC が更に細かく分類されている場合を想定して、ミニノート PC 以下のクラスについても検索の対象に含めるように設定している。

例 3) 例 2 で発見されたミニノート PC のプロパティ P0001 について、ミニノート PC よりも汎用的な製品分類でこのプロパティを持つクラスの情報を検索する

```
SELECT CLASS.*  
FROM ^AAA000!  
WHERE PROPERTY.IDENTIFIER = 'P0001';
```

この例は、ある特徴を有する汎用クラスの検索を表す。ここで、SQL や OQL では、プロパティのネームスコープのために、あるプロパティについて上位クラスから下位クラスを検索する場合、その上位クラスがそのプロパティを必ず有する必要があるため、予めこのプロパティを持つクラスを知る必要がある。一方、CQL ではそのプロパティを持つ上位クラスまでを下位クラスから検索できるため、このような検索を容易に実現できる。

このように、CQL では階層構造を持つデータ辞書の情報に対する強力な検索機能を有する。特に、検索する対象のクラス階層構造を知らなくても、いくつかの簡単な検索文を利用することで、容易にデータ辞書の情報を検索することができる。

### 3.5. CQL の対象範囲

前述のように、CQL はオブジェクト指向的に上位クラスから下位クラスへとプロパティが継承されるデータモデルや、それに基づくデータ辞書およびそのデータを包括的に記述し、検索する機能を提供する。これらの機能は、SQL や OQL などの既存の問い合わせ言語と比べて、柔軟かつ強力なものである。

図 7 は、CQL で表現できるデータモデルを概念的に示したものである。まず、階層構造を持つデータモデルの中で、部分継承を持つ PLIB は、CQL のプロパティの輸入関係を利用することで表現できる。また、プロパティの輸入関係において、輸入先のクラスが持つ全てのプロパティを輸入することで、多重継承と同様の構造を実現することができる。これにより、多重継承を持つ RDL のデータモデルやオブジェクトデータモデルを記述することが可能である。また、CQL は SQL を拡張した言語であるため、階層構造を持たない OTD のデータモデルやリレーショナルデータモデルを記述することが可能である。

一方、検索範囲の指定やクラス間のブール演算を行うことで、PLIB や RDL の階層的クラス構造を持つモデルについても柔軟な検索を行うことができる。さら

に、データモデルに依らず、データだけでなくメタデータもデータとして検索することができるため、様々なデータモデルで記述されたデータ辞書を横断的に検索することができる。

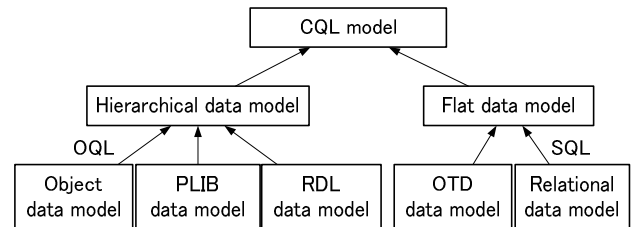


図 7 CQL の対象範囲

## 4. ISO29002 への CQL の適用に向けて

CQL の構文仕様はシンプルではあるが、ISO29002 の範囲から外れる機能も多い。そこで、CQL の機能を限定したものを策定し、提案することを検討している。前述のように、現在検討されている ISO29002 の仕様では、クラスやプロパティの検索に重点が置かれている。そこで、まず検索に機能を絞って提案を行うことを検討する。

Web サービスを利用して検索を行う場合、検索メッセージの送信 (request) および検索結果の受信 (response) を XML フォーマットで行うことが想定される。ここで、現在の CQL は検索結果 (ResultSet) のフォーマットを策定していない。従って、ResultSet のフォーマットを含めて、CQL の request/response メッセージのフォーマットを検討する必要がある。

一方、これまでの CQL は、特定のサーバで管理されているリソースを検索することを想定していたため、サーバのネットワーク上の位置情報を指定する手段を構文仕様の中に含んでいない。しかし、ISO29002 で検討されている Web サービスの仕様では、Web 上に分散した複数のサーバの持つオントロジー情報を検索することが想定されている。従って、XML フォーマットの仕様策定と併せて、サーバの位置情報の表現方法を検討する必要がある。

## 5. まとめ

本稿では、現在策定中の ISO29002 に紐付けされる辞書情報の検索手段としての CQL の適正を検討することを目的に、ISO29002 の対象となる 3 つのデータモデルで記述された辞書情報の検索に必要な機能を説明し、CQL の特徴を詳説することで、CQL がこれらの要件を満たすことを示した。また、Web サービスで CQL を利用するに当たって検討が必要となる項目を紹介した。今後は、これらを踏まえた上で、ISO29002 の検索

言語として CQL の仕様を固め，提案していく．

## 文 献

- [1] JEITA EC センター， <http://ec.jeita.or.jp/>
- [2] 村山廣，“国際規格 PLIB (ISO13584)，” 東芝レビュー, Vol.56, No.11, pp76-77, 2001.
- [3] 細川晃，溝口祐美子，村山廣，“インスタンス検索における CQL の記述性に関する一考察，” デジタルドキュメント研究会, Vol.57, No.6, pp.37-44, Sept.2006.
- [4] D. Leal, “ISO 15926 Life Cycle Data for Process Plant: An Overview,” Oil & Gas Science and Technology – Rev.IFP, Vol.60, No.4, pp.629-637, 2005.
- [5] ISO Standard : ISO22745 Open Technical Dictionary
- [6] Y. Mizoguchi-Shimogori, H. Murayama, N. Minamino, “Class Query Language and its application to ISO13584 Parts Library Standard,” ECEC2002 Concurrent Engineering: System integration for profit, pp128-135, Modena, Italy, Apr.2002.
- [7] 丸山圭三郎，ソシユールを読む，岩波書店，東京，1983.
- [8] C. Ogden, I. Richards, 意味の意味，石橋幸太郎(訳)，新泉社，東京，2001.