

# SAGE: 仮想カタログ — データベースのエージェント化 — \*

佐藤 陽      益岡 竜介      丸山 文宏      菅坂 玉美

(株) 富士通研究所

E-mail: {satoyan, masuoka, superb, tamami}@flab.fujitsu.co.jp

## 1 はじめに

最近 WWW の成功などにより、情報の公開や公開された情報の積極的な利用への関心が高まってきている。特にビジネスの分野では、独自にデータベース（以下 DB）を持つ企業も増え、DB の規模も相当の大きさになってきており、DB の情報公開や、近い業種間での情報共有などを実現しようとする意識が高まっている。しかし、各企業で独自に管理運営している DB では、構造や用語について統一する手間や管理が必要となる。

我々は SAGE[1][2][3] において、情報の要求者、提供者を ACL[4][5] で知識をやり取りするエージェントにし、エージェントどうしの仲介を行なう特別なエージェント (facilitator) を含めたエージェント間の情報交換をうまく設計することで上記の問題を解決していこうとするアプローチを行なっている。このアプローチでは、情報を提供するアプリケーション (DB など) を SAGE に取り込むために、ACL を話す仕組みで包み込んでエージェント化する。

本稿では、DB が持つ知識を個々の DB の構造や用語の違いに依存せずに仮想化する方法と、DB をエージェント化したデータベースエージェントの動作、構成について述べる。

## 2 データベースの抽象化 (VKB)

我々は、個々の DB の構造や用語の違いに依存せずに DB の知識を表現するため、仮想的な知識ベース (VKB: Virtual Knowledge Base) を用いてデータベースを抽象化して扱うことにした。以下では SAGE における VKB の定義、VKB のアクセス方法の定義、それらを facilitator に advertise する方法について述べる。

### 2.1 VKB の定義

DB を表現する VKB を以下のように定義する。

- VKB はレコードの集合である
- レコードは指定したフィールドの値を聞くことができるものである

### 2.2 VKB へのアクセス方法

VKB へのアクセスは、VKB に記述されている知識の一部を条件によって切り出すことと、切り出したものに

対する操作 (取り出す (検索する)、消去する、書き換えるなど) を指定することからなる。ACL で VKB にアクセスする場合には、KIF による relation によって対象となる VKB 上の知識を指定し、それに対する操作を KQML による performative で指定する。

今回は、操作に関しては検索のみを考えるものとした。VKB に対して利用できる relation としては以下のものが利用できるものとして定義する。

- VKB のレコードの特定のフィールドとその値とを対応づける relation
- 量を表わす数値同士を比較する算術的な relation
- relation 同士の論理的組み合わせを定義する relation (and, or など)
- 条件に当てはまるものに適用して二次的な結果を得る relation (レコード数, 合計, 平均, 最大, 最小, 先頭, 最後 など)

### 2.3 VKB の advertise

VKB の構造と VKB へのアクセス方法を facilitator に advertise することにより、他のエージェントからアクセスしてもらえるようになる。このとき、エージェントの持つ知識をより過不足なく宣言するためには advertise のしかたが重要となる。

我々は以下のものを advertise ことにした。

- VKB の名前
- VKB で扱うことのできるカテゴリ
- VKB に現れるフィールド
- VKB へのアクセスに利用できる relation

このうち VKB が扱うことのできるカテゴリの宣言は、他のエージェントに自分の扱う内容を示すために必要である。VKB の定義に含まれるデータベースの名前だけではどのような情報を扱うことができるかを規定できないため、扱っている物事を表わすカテゴリを宣言する。カテゴリは複数与えることができるものとする。

また、VKB にアクセスする relation については 2.2 で定義した relation をすべて個別に advertise するのではなく、受け付けるフォーマットを定めたルールをつくり、それを advertise するという方法をとった。

\*SAGE:Smart AGent Environment - Database Agent -, Akira Sato, Ryusuke Masuoka, Fumihito Maruyama, and Tamami Sugasaki; Fujitsu Laboratories Ltd. 2-2-1 Momochihama, Sawaraku, Fukuoka 814, Japan

以下の形式で advertise すると, 2.2で定義した全ての relation を利用できる.

(allows-relational-db-query データベース名)

### 3 データベースエージェント

前節で定義した VKB から実際の DB へのアクセスを行なうデータベースエージェント (以下 DB エージェント) について, 動作と構成を述べる.

#### 3.1 DB エージェントの動作

DB エージェントは, 起動した時点で facilitator に VKB の構造と VKB へのアクセス方法を advertise する.

実際の検索要求が ACL のメッセージで届くと, DB エージェントは以下のような処理を行なう.

- (1) メッセージの取り出しを行なう.
- (2) ACL メッセージを DB への検索要求へと変換する. 今回は DB として SQL を受け付ける RDB を対象としたので, VKB と実際の RDB との項目どうしを変換する規則と KIF の relation で記述された VKB へのアクセスを RDB へのアクセス (SQL) に変換する規則を定義し, これを用いて変換する.
- (3) 変換した結果を用いて DB にアクセスする.
- (4) DB からの回答を ACL メッセージに変換する. 回答の内容を検索要求中で指定された KIF の形式に変換し, それをさらに KQML のメッセージに変換する.
- (5) メッセージを送信する.

#### 3.2 DB エージェントの構成

DB エージェントは, 以下のような構成要素からなる (図 1).

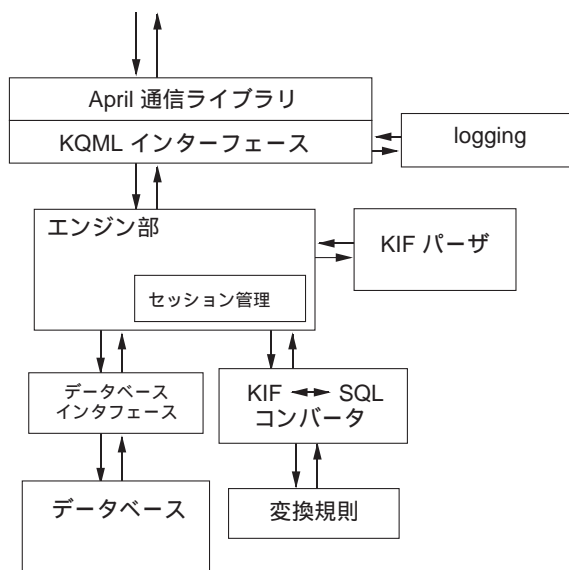


図 1: データベースエージェントの構成

- April 通信ライブラリ・KQML インタフェース  
メッセージの送受信を担当する. また, logging 部においてメッセージの内容を保持し, その情報を読み出すことができるようなサポートを行なう.

- エンジン部  
入力に対しては, KQML メッセージの取り出しを行ない, メッセージを解析してデータベースへのアクセスを行なう. また, データベースから返ってきたメッセージを KQML のメッセージに変換する. advertise を行なうタイミングの制御も行なう.

- KIF パーザ  
メッセージに含まれる KIF 部分の解析を行なう.

- KIF  $\leftrightarrow$  SQL コンバータ  
VKB にアクセスする KIF の relation を, 変換規則を利用してデータベースを扱う SQL コマンドへと変換する. また, データベースから返ってきたメッセージを KIF の形式に変換する.

- データベースインタフェース  
DB へのアクセスを行なう. 対象とする DB をサポートする ODBC や JDBC などのライブラリを用いて実装する.

- データベース  
実際のデータを扱う. 通常のリレーショナルデータベースをそのまま用いる.

## 4 まとめ

本稿では, SAGE:仮想カタログにおける DB のエージェント化について, DB を仮想化する方法と, DB エージェントの動作・構成を述べた.

今後は, より高度な質問に答えられるようにすることと, facilitator と連携してより適切な advertise を行ない, DB エージェント自身の能力をより過不足なく表現する仕組みを目指したい.

## 参考文献

- [1] 丸山文宏他, "SAGE: 仮想カタログ", 情報処理学会第 54 回全国大会, 1997.
- [2] 益岡竜介他, "SAGE: 仮想カタログ-システムデザイン-", 情報処理学会第 54 回全国大会, 1997.
- [3] 菅坂玉美他, "SAGE: 仮想カタログ-ユーザエージェント-", 情報処理学会第 54 回全国大会, 1997.
- [4] Michael R. Genesereth and Richard E. Fikes, "Knowledge Interchange Format Version 3.0 Reference Manual", June 1992.
- [5] The DARPA Knowledge Sharing Initiative External Interfaces Working Group, "Specification of the KQML Agent-Communication Language", February 9 1994.