

グラフに基づくデータベースに対するグラフィカル問合せ言語の実現

宮崎 則雄 † 寶珍 輝尚 ‡
都司 達夫 ‡ 樋口 健 ‡

† 福井大学大学院工学研究科 ‡ 福井大学工学部情報・メディア工学科
{miyazaki, hochin, tsuji, higuchi}@pear.fuis.fukui-u.ac.jp

本論文では、ラベル付き有向グラフで表現されたデータベースに対する問合せ言語 DUO のインタプリタの開発について述べる。グラフィカル問合せ言語 DUO について概説した後、DUO 用インタプリタのシステム構成と実現方法について述べ、その動作例を示す。

Implementation of an Interpreter of the Graphical Query Language for the Database Based on Directed Graphs

Norio MIYAZAKI†, Teruhisa HOCHIN‡
Tatsuo TSUJI‡ and Ken HIGUCHI‡

†Graduate School of Engineering, Fukui University
‡Faculty of Engineering, Fukui University
{miyazaki, hochin, tsuji, higuchi}@pear.fuis.fukui-u.ac.jp

This paper describes an implementation of an interpreter for DUO, which is a graphical query language for the database based on labeled directed graphs. First, this paper briefly describes DUO. Next, the implementation of the DUO interpreter is presented. Finally, examples of the execution of the interpreter are shown.

1 はじめに

コンピュータの世界ではソフトウェア構築のために様々なプログラミング言語が開発され、ソフトウェアの発展に貢献してきた。従来のプログラミング言語は、様々な文字列を記述することによりプログラミングを行う。このような言語を文字列言語と呼ぶことにする。文字列言語は我々が普段用いる自然言語とは異なり、プログラミング特有の独特な文字列・記号の列挙であり、使用される文字列も英語やその短縮形であることが多い。また、プログラミング言語特有の独特の構文やアルゴリズムを修得しなければならない。そのためプログラミング初心者などにとっては、修得が困難であるという問題がある。また、文字列を線型に記述しているため、複雑な構造 (3次元配列など次元数の高い配列、線型リストや木) を持つデータの扱いが難しく、読解も非常に困難になっている。

そこで、これらの問題点を解決するものとしてヴィジュアル言語が提案されている。これは視覚的にプログラミングを行うことができるため、修得、読解が比較的容易であり、複雑なデータも容易に記述することができる。

このようなヴィジュアル言語のひとつとして、グラフィカル問い合わせ言語 DUO [1] が提案されている。DUO は、有向グラフとして表現されるデータに対して、グラフィカルな一種の正規表現を用いて問合せを行なうための言語である。

本研究では、グラフィカル問合せ言語 DUO の文法を完全にサポートし、十分な機能を持つインターフェースを備えた DUO インタプリタの実現を目指す。本研究で作成した DUO インタプリタはインターフェースと本体に大きく分けられる。本体は、記号解析部、構文解析部、問合せ変換部、結果変換部から構成されている。構文解析部は多次元言語パーザ自動生成システム NAE [3] を用いて作成されて

いる。実際のデータベース処理には、ICOT(新世代コンピュータ技術開発機構)で研究・開発された演繹オブジェクト指向データベースシステム Quixote を使用している。本論文では、実現したインタプリタを路線データとXMLデータに対して使用し、木構造やDAG構造の問合せが必要であることを示す。

以下、2ではグラフィカル問合せ言語DUOについて概説する。3では多次元言語パーザ自動生成システムNAEについて概説する。4ではDUOインタプリタのシステム構成と実現方法について述べる。5ではDUOインタプリタの動作例を示す。

2 グラフィカル問合せ言語DUO

グラフィカル問い合わせ言語DUOはラベル付き有向グラフで表されるデータに対してグラフィカルな正規表現を用いて問合せを行なうための言語である。DUOは以下に挙げるような特徴を備えている。

- 問合せグラフと問合せ表現
DUOは2段階の問合せの表現方法を持っている。簡単な問合せを宣言的に行う問合せグラフと複雑な問合せを手続き的に行う問合せ表現である。問合せグラフは点、枝、括弧、スコープが連結された物である。点と枝はそれが答を求めたいものである場合には黒い線で、そうでないものはグレイの線で記述する。
- 括弧による正規表現
括弧を使うことにより再帰指定等がグラフィカルに表現できる。
- スコープによる否定の表現
矩形によりスコープの範囲を指定でき、これを利用して否定を表現できる。
- 矩形領域の重なりによる集合演算等の表現
集合演算や集合比較演算を矩形領域の重なりによりグラフィカルに表現できる。

図2は、図1に示したデータに対して、“Fukuoka”という値を持つcityに、パスarc,または、パスarc,city,arcでつながっているcityを求めるための問合せである。図の最左の点は黒線で描かれてい

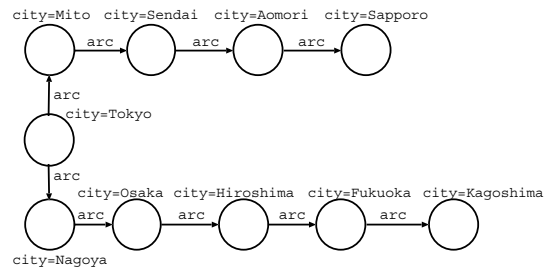


図1: 問合せの対象となるデータの例

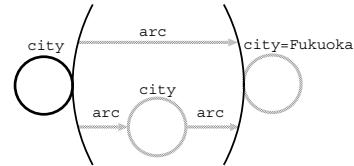


図2: 問合せグラフの例

るので、この点が問合せ結果として利用者に返却される。

また、図3は、nagoyaの上り方面、または、sendaiの上り方面のcityを求める”という問合せ表現である。ここでは、nagoyaの上り方面を求め、次にsendaiの上り方面を求め、最後にそれらのcityの和を求めている。問合せ表現は複数の導出データグラフと3種類のラベルから構成される。3種類のラベル“Query Start”、“Evaluate”、“Display”はそれぞれ、“問合せを開始する”、“問合せグラフを評価する”、“結果を表示する”ことを示す。

3 多次元言語パーザ自動生成システムNAE

3.1 概要

NAE(N-dimension syntax Analysis Environment generator)とは、制約・関係マルチセット文法(Constraint Relation Multiset Grammar: CRMG)を用いた多次元言語の定義から自動的に多次元言語のパーザを生成するシステムである[3]。

制約・関係マルチセット文法(CRMG)[3]とは、Constraint Multiset Grammar[4]に記号間の関係を導入した文法である。これにより、制約を簡潔

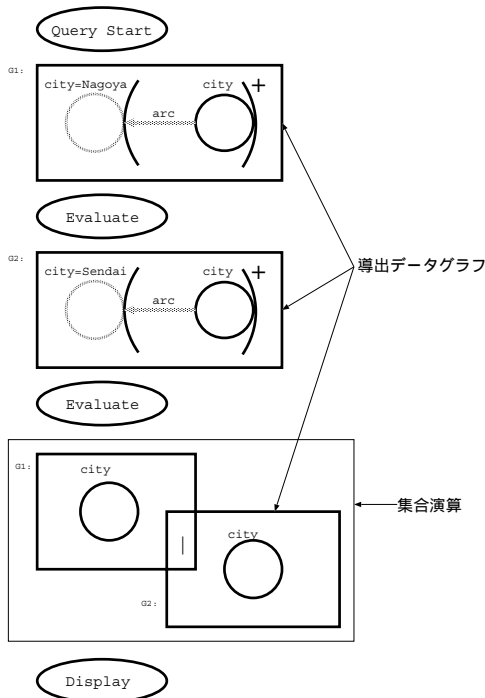


図 3: 問合せ表現の例

に表現することができ、生成規則も簡潔なものとなる。Constraint Multiset Grammar [4] は、ビジュアル言語の構文解析を行うための文法であり、文脈自由文法の拡張である。記号の属性間の制約を用いて記号間の関係を表記するという特徴としている。

3.2 システム構成

NAE は、多次元言語の定義を入力として受け取り、多次元言語パーザの C++ソースプログラムを自動的に生成するシステムである。NAE は、記号の型変換部、関係変換部、生成規則変換部、共通パーザの 4 つの要素から構成される。NAE のシステム構成を図 4 に示す。

- 記号の型変換部
記号の型の定義を C++のクラスへ変換する。
- 関係変換部
記号間の関係の定義を制約を満たすかどうか(真か偽か)を返す C++の関数へと変換する。
- 生成規則変換部

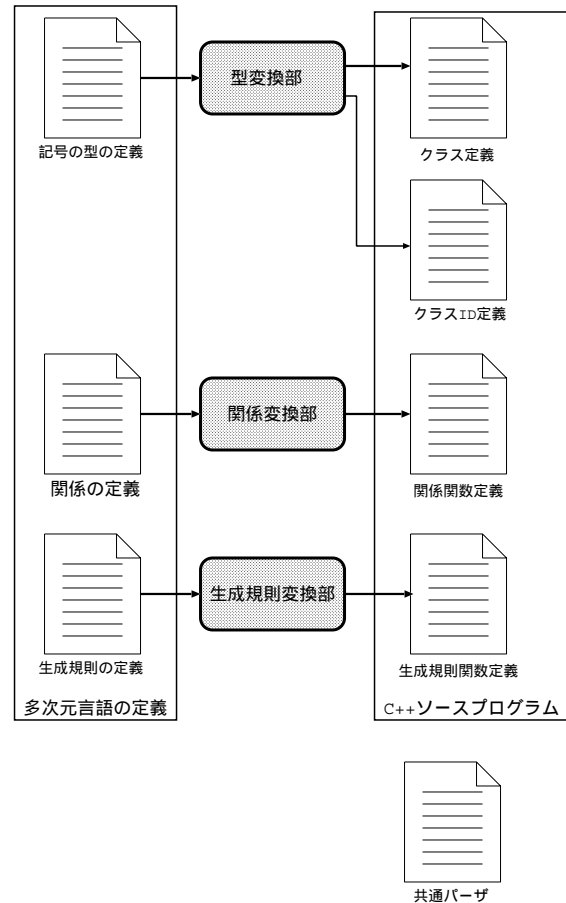


図 4: 多次元言語パーザ自動生成システム NAE

生成規則の定義を生成規則を処理するための C++の関数へ変換する。

- 共通パーザ
多次元言語を構文解析するための構文解析(パーザ)関数である。各変換部で生成されたクラス、関係関数、生成規則関数を用いて構文解析を行う。

4 DUO インタプリタの実現

まず、DUO インタプリタに対する要求を示し、次に、実現した DUO インタプリタのシステム構成について述べる。その後、各構成部の実現について述べる。

4.1 要求

DUO インタプリタとして以下が要求される。

- ・ DUO 言語の完全なサポート
- ・ 使いやすいインターフェース
- ・ 問合せの記述のしやすさ
- ・ グラフィカルな結果の表示

4.2 システム構成

DUO インタプリタのシステム構成を図 5 に示す。

DUO インタプリタはインターフェースと本体の 2 つに大きく分けられる。インターフェースは Java で作成されている。本体は、記号解析部、構文解析部、問合せ変換部の 3 つの部分から構成される。構文解析部は NAE によって製作されている。NAE を用いることによって構文解析部の作成や修正が容易になる。また、実際のデータベース処理には演繹オブジェクト指向データベースシステム Quixote [5] を使用している。

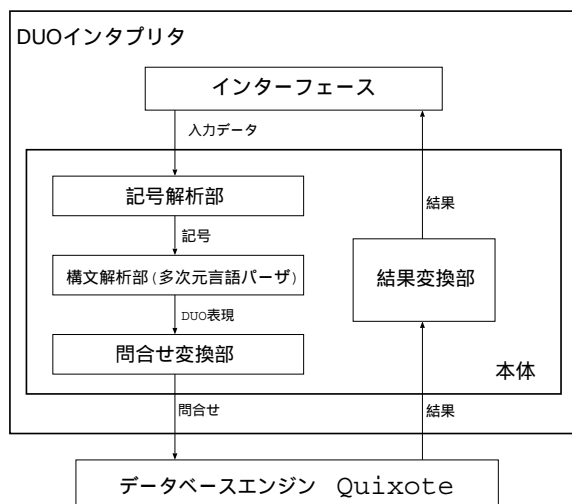


図 5: DUO インタプリタのシステム構成

- インターフェース
問合せの入力情報を文字列として本体に送る。
問合せの結果を受け取って表示する。
- 本体
記号解析部、構文解析部、問合せ変換部、ならびに、結果変換部で構成されている。

- 記号解析部
インターフェースから送られた文字列を記号の集合という形に変換し構文解析部へ送る。
- 構文解析部
記号解析部から送られた記号の集合に対して構文解析を行い、DUO のセマンティックスに沿った表現である DUO 表現へと変換を行う。構文解析部は多次元言語パーザ自動生成システム NAE を用いて作成されている。
- 問合せ変換部
問合せ変換部では、構文解析部から送られた DUO 表現を Quixote の問合せ表現へと変換し、パイプを用いて Quixote へ実際に問合せを行う。
- 結果変換部
Quixote への問合せの結果をインターフェースが受け取る結果として適切な結果に変換して、インターフェースに渡す。

4.3 実現

4.3.1 インターフェース

1. 本体への問合せ情報の送信

インターフェースにおいて問合せが行われると、インターフェースは、一つの部品について一行の情報を記号解析部へと渡す。

図 6 に示す問合せがインターフェースで入力された場合、図 7 に示す文字列が、インターフェースから記号解析部へと渡される。まず、問合せであることを示す文字列 "Query" と、インターフェースで指定された、問合せを行うデータベースファイルの情報 (この例では AREA) が送られる。データベースファイルの指定の行から End のみの行までが、インターフェースに描画された要素のデータである。

行の先頭が要素の種類を表しており、その後に要素の座標や詳細データが続く。要素の種類には、node、edge、par の 3 種類があり、それぞれ、点、枝、括弧を表している。また、それぞれ、要素の座標情報 (x,y 等) と問合せ情報 (es,cs 等) の情報を表している。なお、!は条件が指定されていないことを表している。

2. 問合せ結果の表示方法

本体から検索結果が 4.3.5 で述べるような形式で返却されると、それを解釈して点と枝からなるグラフ構造を作成する。このグラフ構造の描画には、Java2SDK に含まれるグラフ描画のサンプルアプレットの一部を変更して用いている。描画アルゴリズムの概略を以下に示す。

- まず、初めに各点の座標を適当に決める。
- 各点をつなぐ枝の長さが定められた長さに近づくように、また、各点の間の距離がなるべく離れるように、各点を少しずつ動かしていく。
- ある程度収束するまで繰り返す。

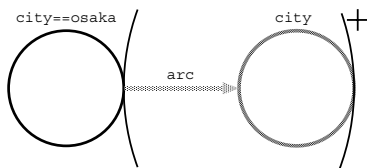


図 6: 問合せグラフの例

4.3.2 記号解析部

記号解析部はインターフェースから受け取った記号の情報を適切なデータ形式に変換し、構文解析部に要素の集合データを送る

4.3.3 構文解析部

NAE によって生成された多次元言語パーザである。DUO 文法の記号として、以下に示す記号の型を定義する。

- Node: 点の情報を示す。
- Edge: 枝の情報を示す。
- Bracket: 括弧の情報を示す。
- FullElms: 完全な要素列を示す。完全な要素列とは、始終点ともに点の要素列のことを指す。
- DestinationElms: 終点がない要素列を示す。
- SourceElms: 始点がない要素列を示す。

Query AREA

```
node x: 0 y: 0 r: 10 margin: 3
```

```
query: NORETURN
```

```
es: city rl: EQUAL rc: osaka V: !
```

```
node x: 50 y: 0 r: 10 margin: 3
```

```
query: RETURN
```

```
es: city rl: NONE rc: ! V: !
```

```
edge fromx: 11 fromy: 0 tox: 39 toy: 0
```

```
margin: 3 query: NORETURN es: arc rl:
```

```
NONE rc: ! V: !
```

```
par x1: 10 y1: 10 x2: 60 y2: 10
```

```
x3: 60 y3: -10 x4: 10 y4: -10 margin: 3
```

```
cs: PLUS V: ! prc: !
```

End

図 7: 記号解析部に渡される情報の例

- ArcElms: 始点、終点ともない要素列を示す。
- QueryGraph: 入力されたグラフ全体を示す。

これらの記号に対して記号間の関係と生成規則を定義し、構文解析部を生成する。

4.3.4 問合せ変換部

構文解析部から受け取った DUO 表現を解析して、Quixote の問合せに変換する。

問合せ変換部に渡される DUO 表現は図 6 に示す問合せの場合、図 8 に示す構造になる。qg_type は問合せグラフ, ELMS_type は要素か要素列, parent_hesis_type は括弧, element_type は要素をそれぞれ表している。

図 8 に示す DUO 表現の場合、図 9 に示す Quixote の問合せに変換される。

パスの和を表す括弧を Quixote の問合せに変換するアルゴリズムを以下に示す。

1. 括弧を表すルール名を適当につける。

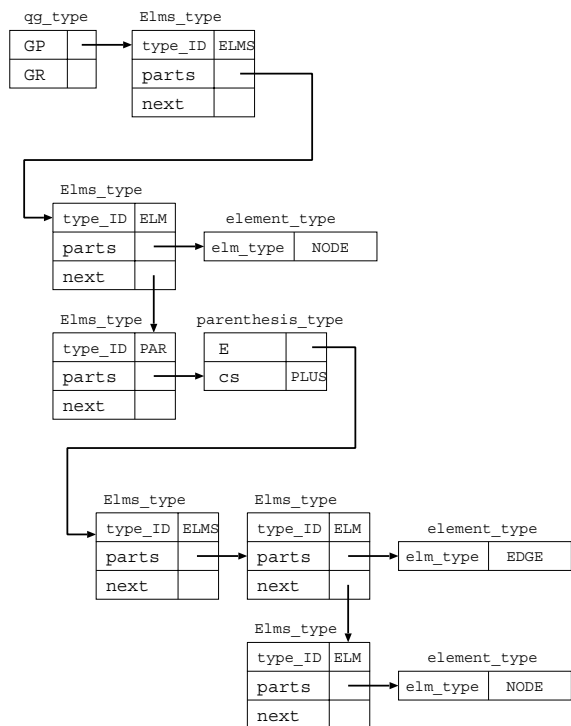


図 8: DUO 表現の例

2. 括弧の内部のパスをそれぞれ、問合せの形に変換する。
3. 括弧の内部のパスが変換された結果を利用して、括弧を表すルールをつくる。
4. 問合せを追加する。
 - 括弧の内部が始終不完全パスのとき括弧の前の要素と括弧の後の要素から変数名を得る。その変数名と括弧を表すルールから問合せを作成し、その問合せを追加する。
 - 括弧の内部が完全パスのとき適当な変数名を2つ作る。その変数名と括弧を表すルールから問合せを作成し、その問合せを追加する。

4.3.5 結果変換部

Quixote からの問合せの結果を適切な形に変換してインターフェースに渡す。

変換のアルゴリズムを以下に示す

1. DUO 表現を参照して、値を返す点と、その点

```

&load "AREA.qxt";

?-city[name=_VAR13] | {_VAR13==osaka},
k1336_003[lb0=_VAR13,lb1=_VAR14];

&program;;&rule;;

k1336_002[lb0=_VAR13,lb1=_VAR14]<=
arc[up=_VAR13,down=_VAR14],
city[name=_VAR14];

k1336_003[lb0=_VAR13,lb1=_VAR14]<=
k1336_002[lb0=_VAR13,lb1=_VAR14];

k1336_003[lb0=_VAR13,lb1=_k1336_004]<=
k1336_002[lb0=_VAR13,lb1=_VAR14]
,k1336_003[lb0=_VAR14,lb1=_k1336_004];

&end.
&quit;;

```

図 9: Quixote 問合せの例

- に割り付けられた変数名の組をリストに加える。値を返す枝と、その両端の点に割り付けられた変数名の組をリストに加える。
2. リストを参照して、Quixote の結果に、値を返す点に割り付けられた変数名があれば点の情報と変数の値をインターフェースに渡す結果の文字列に変換する。このとき、変数名と点の番号の組の情報を保持しておく。
3. リストを参照して、Quixote の結果に、値を返す枝の両端の点に割り付けられた変数名があれば、変数名と点の番号の組の情報を参照して、枝の情報をインターフェースに渡す結果の文字列に変換する。

以下に示す Quixote からの問合せ結果を上述のアルゴリズムに従って変換した結果を図 10 に示す。

```
_VAR1 = nagoya, _VAR2 = osaka
```

図 10 に示した文字列がインターフェースに返却される。

```
node 1 es:city val:nagoya
node 2 es:city val:osaka
edge 3 es:arc from:1 to:2
End
```

図 10: インターフェースへの返却文字列

5 DUO インタプリタの動作例

DUO インタプリタの動作の様子を示す。

図 11 に示すデータに対して、図 12 のような問合せを行う。これは、city,arc,city というパスにマッチするものを全て求めるという問合せである。検索結果を図 13 に示す。

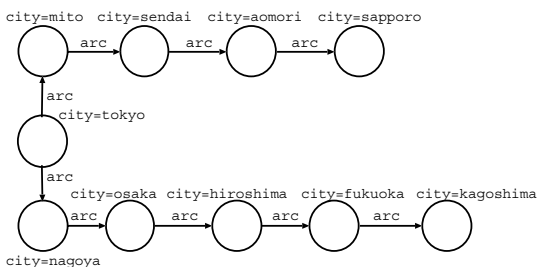


図 11: 問合せの対象となるデータ 1

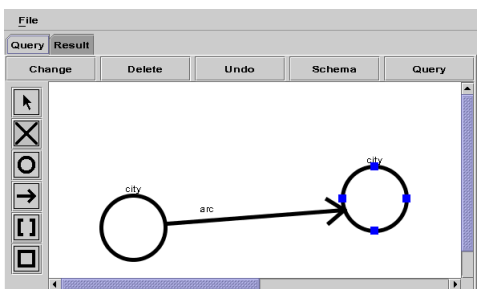


図 12: 問合せ例 1

次に XML で表されたデータに対する問合せ例を示す。図 14 に示す XML のデータを図 15 に示すグラフ構造にして扱う。このデータに対して図 16 に示す問合せを行う。この問合せは子孫に”jane” という名前を持つ人がいる人を求めるという問合せである。検索結果を図 17 に示す。

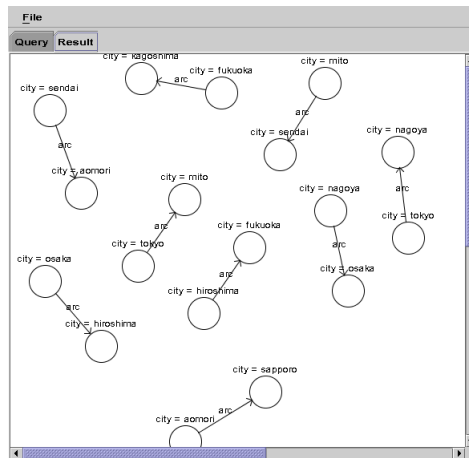


図 13: 検索結果 1

6 おわりに

本研究では、グラフィカル問合せ言語 DUO のインタプリタの開発を行った。DUO 問合せの基本である問合せグラフをサポートしている。また、いくつかのデータに対する検索を行った結果について述べた。

現在のインタプリタでは、DUO 文法を完全にはサポートしていない。具体的には、図 16 に示すような線形構造の問合せしかできない。XML データの問合せには木構造や DAG 構造の問合せも必要と考えられるが、このサポートは今後の課題である。さらに、否定を表すスコープ、問合せ結果の和集合や差集合を求めるための問合せ表現などもサポートしていない。これらの問題を解決し DUO 文法の完全なサポートを行なうことは今後の課題である。

インターフェースの問合せ結果表示では、結果の構造によっては適切な表示がされないことがある。この問合せ結果表示の改良も課題である。

参考文献

- [1] 宝珍輝尚: グラフィカル問い合わせ言語 DUO の問い合わせ能力, 情処論 Vol36, No.4, pp.959-970 (1995).
- [2] 畠山竜輔, 宝珍輝尚, 都司達夫: 多次元言語パーザ自動生成システムを用いたグラフィカ

- [3] 石井義知, 宝珍輝尚, 都司達夫: 多次元言語の構文解析に関する一手法, 信学技法 SS96-64, pp.25-32 (1997).
- [4] Kim Marriot: Constraint Multiset Grammars, 1994 IEEE Symposium on Visual Languages, pp.118-125 (1994).
- [5] 財団法人 新世代コンピュータ技術開発機構: Quixote 言語マニュアル (1995).

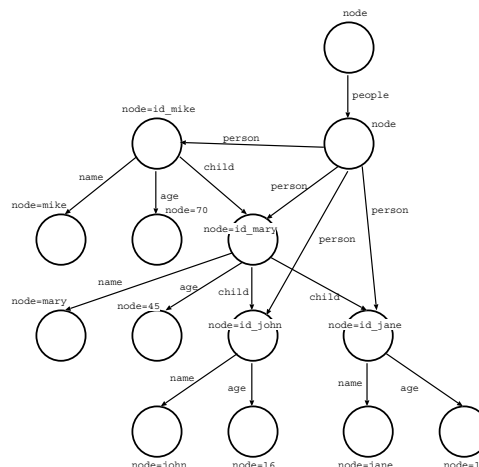


図 15: 問合せの対象となるデータ 2

```

<people>
  <person id="id_mike">
    <name> mike </name>
    <age> 70 </age>
    <child idref="id_mary"/>
  </person>

  <person id="id_mary">
    <name> mary </name>
    <age> 45 </age>
    <child idref="id_john"/>
    <child idref="id_jane"/>
  </person>

  <person id="id_john">
    <name> john </name>
    <age> 16 </age>
  </person>

  <person id="id_jane">
    <name> jane </name>
    <age> 14 </age>
  </person>
</people>

```

図 14: XML で表されたデータ

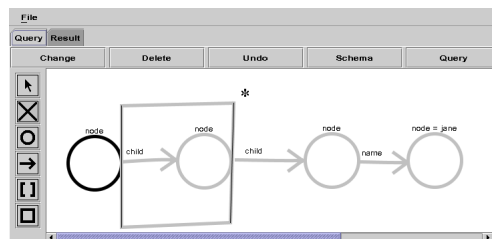


図 16: 問合せ例 2

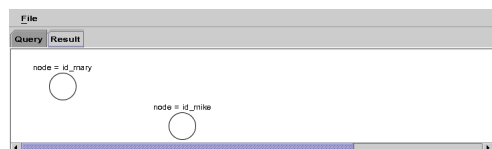


図 17: 検索結果 2