Quality-driven Integration of Heterogeneous Information Sources

J.-C. Freytag\textsuperscript{1}
F. Naumann\textsuperscript{1}, U. Leser\textsuperscript{2}
\textsuperscript{1} Humboldt University Berlin
\textsuperscript{2} Technical University Berlin
\{naumann,freytag\}@dbis.informatik.hu-berlin.de
leser@cs.tu-berlin.de

---

Introduction & Motivation

- DB
  - distributed
  - controlled
  - QL
  - query
  - correct
  - replication

- WEB-IS
  - distributed
  - autonomous
  - search
  - browse
  - fuzzy
  - overlap

Freytag, Naumann, Leser: "Quality-driven Integration ...", December 2000
WEB-based Querying

- DB-Querying
  - cost-based
  - each plan produces same result
  - optimize operators

- WWW-Querying
  - cost based (?)
  - each plan produces different results
  - optimize sources

IQ is main discriminator of WWW planning

Introduction & Motivation

Query: Give me all genes on the human X chromosome with pattern ‘…” in their sequence

- Many alternative ways of obtaining answers
- Varying information quality

⇒ Find plans with highest information quality

Similar problems in:
- Travel planning systems
- Stock information services
- etc.
Framework

Plan: combination of “wrapper” queries

Our solution: 1. Preparation

Annotate correspondences with information quality:
2. Source selection

Use IQ scores to remove the worst sources:

3. Query planning

Compute all plans:

$p_1 := q_1 q_2 q_3 q_4 q_5 q_6 q_7 q_8 q_9 q_{10} q_{11} q_{12} q_{13} q_{14} q_{15} q_{16} q_{17} q_{18} q_{19} q_{20}$

$p_2 := q_1 q_2 q_3 q_4 q_5 q_6 q_7 q_8 q_9 q_{10} q_{11} q_{12} q_{13} q_{14} q_{15} q_{16} q_{17} q_{18} q_{19} q_{20}$

$p_3 := q_1 q_2 q_3 q_4 q_5 q_6 q_7 q_8 q_9 q_{10} q_{11} q_{12} q_{13} q_{14} q_{15} q_{16} q_{17} q_{18} q_{19} q_{20}$

etc.
4. Plan selection

Use IQ scores to find the best plans:

\[ \rho_1 := q_1 q_2 q_3 q_4 \ldots \]
\[ \rho_2 := q_5 q_6 q_7 q_8 \ldots \]
\[ \rho_3 := q_9 q_{10} q_{11} q_{12} \ldots \]

etc.

\[ \text{IQ}(\rho_1) = (60,0,3\ldots) \bowtie (46,0,2\ldots) \bowtie \ldots \]

Overview

- Introduction
- Describing Information Sources
- Information Quality: Criteria & Classification
- Finding High-Quality Plans
- Discussion
Query Correspondence Assertions

- Connect two conjunctive queries:
  - one executable query against the export schema
  - one intentionally equivalent query against the mediator schema
  - head is "shared"

Examples

The global schema:

Some exemplary QCAs:

QCA1: `sequence(Gn,Se,Or,An) ← s_j(v_j(Gn,Se,Or,An) ← seq(Gn,Se,Or,An))`

QCA2: `sequence(Gn,Se,Or,An) ← s_j(v_j(Gn,Se,Or,An) ← seq(Gn,Se,Or,An))`

QCA3: `gene(Gn),cDNACluster(Gn,Dn),cDNA(Dn,Ch,P,P) ← s_j(v_j(Gn,Di,Ch,P,P) ← clustering(Gn,Dn,Di,Ch,P,P),primers(Dn,P,P))`
Overview

• Introduction
• Describing Information Sources
• Information Quality: Criteria & Classification
• Finding High-Quality Plans
• Discussion

Information quality

• “Fitness for use”
• IQ criteria
  - objective/subjective
  - application domain specific
  - measurement methods
• Empirical study found
  15 criteria [Wang, Strong 96]
IQ criteria can be...

- **Source-specific**
  - Understandability
  - Reputation
  - Reliability
  - Timeliness

- **QCA-specific**
  - Availability
  - Price
  - Repr. consistency
  - Response time
  - Accuracy
  - Relevancy

- **Attribute-specific**
  - Completeness
  - Amount

---

**Phase 1**
Source selection

**Phase 2**
Query planning

**Phase 3**
Plan selection

---

**IQ scores (example)**

\[ IQ(QCA) = (Un, Rep, Rel, Ti, Av, Pr, RC, RT, Ac, Re); \]

<table>
<thead>
<tr>
<th></th>
<th>S1</th>
<th>S2</th>
<th>S3</th>
<th>S4</th>
<th>QCA1</th>
<th>QCA2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Understandability</td>
<td>5</td>
<td>7</td>
<td>7</td>
<td>8</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Reputation</td>
<td>5</td>
<td>7</td>
<td>7</td>
<td>8</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>Reliability</td>
<td>2</td>
<td>6</td>
<td>4</td>
<td>6</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Timeliness</td>
<td>30</td>
<td>30</td>
<td>2</td>
<td>1</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>Availability</td>
<td>99</td>
<td>99</td>
<td>60</td>
<td>80</td>
<td>99</td>
<td>95</td>
</tr>
<tr>
<td>Price</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Resp. Consistency</td>
<td>1</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.7</td>
<td>0.7</td>
</tr>
<tr>
<td>Resp. Time</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td>3</td>
<td>0.1</td>
<td>1</td>
</tr>
<tr>
<td>Accuracy</td>
<td>99.9</td>
<td>99.9</td>
<td>99.8</td>
<td>99.95</td>
<td>99.95</td>
<td>99.95</td>
</tr>
<tr>
<td>Reliability</td>
<td>60</td>
<td>80</td>
<td>90</td>
<td>80</td>
<td>80</td>
<td>60</td>
</tr>
</tbody>
</table>

Preising, Neumann, Leser: "Quality-driven Integration ...", December 2000
Overview

- Introduction
- Describing Information Sources
- Information Quality: Criteria & Classification
- Finding High-Quality Plans
- Discussion

Phase 1: Source selection (sketch)

Input
- User query, sources with QCAs, and IQ scores

Phase 1 Source Selection

- Data Envelopment Analysis (DEA)
- only source specific IQ criteria
Phase 2: Query Planning (sketch)

- "Answering queries using views" [Levy et al. 95]
- User query q
- Search plans p: \( p \subseteq q \)
  \[ \Rightarrow \text{All correct plans:} \]
  - \( p_1 := QCA_1 \sqcap QCA_2 \sqcap QCA_5 \)
  - \( p_2 := QCA_3 \sqcap QCA_4 \sqcap QCA_6 \sqcap QCA_7 \)
  - ...

Phase 1: Source Selection  \rightarrow  Phase 2: Query Planning

Phase 3: Plan selection

- Goal: Find best \( N \) plans
- Three steps for each plan
  1. Determine attribute-specific IQ scores
     \( \Rightarrow \) Complete IQ vector per QCA
  2. Aggregate the IQ scores along the plan
     \( \Rightarrow \) Complete IQ vector per plan
  3. Find overall IQ score
     \( \Rightarrow \) IQ scalar per plan

Output

Quality-ranked plans

Query planning  \rightarrow  Plan selection
Step 1: Determine IQ scores

IQ vector for each QCA in a plan

\[ \text{IQ}(\text{QCA}_2) = (99, 0, 1, 0.2, 99.9, 80, \ldots) \]

- **Amount**
- **Completeness** considers:
  - Completeness of each exported attribute
  - Importance of this attribute in user query
  - Average weighted QCA - completeness

\[ \text{IQ}(\text{QCA}_2) = (99, 0, 1, 0.2, 99.9, 80, 52.8, 0) \]

---

Step 2: Aggregate IQ scores

**Merge functions:**
- Availability: *
- Price: +
- Response time: max
- Completeness: Sylvester
- ...
Step 3: Rank plans

Simple Additive Weighting (SAW)

\[ P_3: (89.35, 0, 1, 1, 99.8, 28.8, 76.06, 3) \]

\[ v_{ij} = \frac{d_{ij} - d_{ij}^{\text{max}}}{d_{ij}^{\text{max}} - d_{ij}^{\text{min}}} \]

\[ P_3: (1, 1, 0, 1, 1, 0.62, 0.51, 1) \]

\[ \text{IQ}(P_i) = \sum_{j=1}^{n} w_j v_{ij} \]

\[ \text{IQ}(P_3) = 0.77 \]

Results

Order of plans depends on:

- IQ scores of sources,
- IQ scores of QCAs and attributes
- User weighting of criteria
- User weighting of attribute importance
- [length]

Order of plans does not depend on:

- Join order
Discussion

- Fine-grained IQ score assignments:
  - per source, per query and per attribute
  - selection and classification of criteria

- Quality as “cost”:
  - different levels
  - merge functions

- Outlook:
  - branch & bound algorithm
  - IQ of union operations

Quality-driven Integration of Heterogeneous Information Sources

Questions?
Using QCAs

- Mediator queries are used to rewrite a user query into a plan of wrapper queries.

QCAs allow...

**Interface-dependent IQ scores:**
- Web interface to $S_3$ is incomplete, but free
- CORBA interface to $S_4$ is more complete, but costly
Examples

The global schema:

4 sources, 7 QCAs:

QCA_1: $sequence(Gn,Se,Or,An) \leftarrow S_1 \cdot V_1(Gn,Se,Or,An) \leftarrow seq(Gn,Se,Or,An)$
QCA_2: $sequence(Gn,Se,Or,An) \leftarrow S_2 \cdot V_1(Gn,Se,Or,An) \leftarrow seq(Gn,Se,Or,An)$
QCA_3: $sequence(Gn,Se,Or,An) \leftarrow S_3 \cdot V_1(Gn,Se,Or,An) \leftarrow seq(Gn,Se,Or,An)$
QCA_4: $cDNA(Dn,Ch,Po,P_3) \leftarrow S_4 \cdot V_1(Dn,Ch) \leftarrow www(Dn,Ch)$
QCA_5: $cDNA(Dn,Ch,Po,P_3) \leftarrow S_5 \cdot V_2(Dn,Ch,Po,P_3) \leftarrow direct(Dn,Ch,Po,P_3)$
QCA_6: $gene(Gn,Di) \cdot sequence(Gn,Se,-An) \leftarrow S_6 \cdot V_1(Gn,Di,Se,An) \leftarrow genes(GID,Gn,Di,...)$
QCA_7: $gene(Gn,Di) \cdot cDNACluster(Gn,Dn) \cdot cDNA(Dn,Ch,Po,P_3) \leftarrow S_7 \cdot V_2(Gn,Di,Ch,Po,P_3) \leftarrow ...

Step 1: Determine IQ scores

IQ vector for each QCA in a plan

IQ(QCA_2) = (99, 0, 1, 0.2, 99.9, 80, 2.2, 7)

- Amount
- Completeness

$\rightarrow q(Gn \ 100,Di \ 100,Se \ 100,Or \ 30,An \ 70)$
$\rightarrow QCA_2: S_2 \cdot V_1(Gn \ 100,Se \ 80,Or \ 55,An \ 21)$

IQ(QCA_2) = (99, 0, 1, 0.2, 99.9, 80, 52.8, 0)
Example cont’d

User query:

q(Gn, Dn, Se, Or, An) ← gene(Gn, Dn), sequence(Gn, Se, Or, An),
cDNACluster(Gn, Dn), cDNA(Dn, Ch, ?, ?), Ch = 'x';

Correct plans:

\[ P_1 = S_2 \cup G_2 (Gn, Dn, ?, ?), S_2 \cup G_2 (Gn, Se, Or, An), S_2 \cup G_2 (Gn, Dn, ?, ?), \]
\[ S_2 \cup G_2 (Dn, Ch), Ch = 'x'; \]

\[ P_2 = S_2 \cup G_2 (Gn, Dn, ?, ?), S_2 \cup G_2 (Gn, Se, Or, An), S_2 \cup G_2 (Gn, Dn, ?, ?), \]
\[ S_2 \cup G_2 (Dn, Ch, ?, ?), Ch = 'x'; \]

\[ \Rightarrow \text{Too many plans (6)} \]

\[ \Rightarrow \text{Different quality: cost ? completeness ?} \]

Example cont’d

• Indifferent user weighting:
  1. P_3: <QCA_6, QCA_2, QCA_7>
  2. P_6: <QCA_6, QCA_3, QCA_7>
  3. P_3: <QCA_6, QCA_3, QCA_7, QCA_4>
  4. P_3: <QCA_6, QCA_3, QCA_7, QCA_8>
  5. ...

• Response time over price:
  1. P_3: <QCA_6, QCA_2, QCA_7>
  2. P_6: <QCA_6, QCA_3, QCA_7>
  3. P_3: <QCA_6, QCA_3, QCA_7, QCA_8>
  4. P_3: <QCA_6, QCA_3, QCA_7, QCA_8>
  5. ...