Flexible and Scalable Query Planning in Mediators (Research Statement) *

José Luis Ambite
Information Sciences Institute and Department of Computer Science
University of Southern California
ambite@isi.edu

My research interests are on flexible and efficient query processing in mediator systems. In the context of the SIMS (Arens *et al.* 1993; Arens, Knoblock, & Shen 1996) and Ariadne (Knoblock *et al.* 1998) projects, we have applied a general framework for efficient high-quality planning, called Planning by Rewriting (Ambite & Knoblock 1997), to the problem of generating query plans in distributed and heterogeneous environments (Ambite & Knoblock 1998).

SIMS and Ariadne are mediator systems that provide integrated access to heterogeneous sources in an application domain by building a model for the domain and mapping the contents of the sources to this domain model. The domain model is expressed in the Loom description logic (MacGregor 1988). The user poses queries in terms of the domain model and the system generates a plan that answers the query by combining information from the available relevant sources. SIMS has focussed more on the integration of databases and structured sources, while Ariadne addresses the issues arising in Web and other semi-structured sources, such as the need for wrappers and limited source capabilities (e.g., binding pattern constraints).

Planning by Rewriting (PbR) follows the iterative improvement style of many optimization algorithms. The framework works in two phases:

1. Efficiently generate an initial solution plan (possibly suboptimal).
2. Iteratively rewrite the current solution plan in order to improve its quality using a set of plan rewriting rules until either an acceptable solution is found or a resource limit is reached.

We have developed PbR-based query planners for the SIMS and Ariadne systems. The query planner accepts a set of declarative plan rewriting rules and uses local search methods to efficiently generate a high-quality plan. The initial query plan is trivially generated by a random parse of the query (or a greedy one). The rewriting rules for query planning in mediators (Ambite & Knoblock 1998) arise from three sources:

Relational algebra: These rules are the traditional query optimization rewrites based on the properties of the relational algebra, such as pushing selections and exploring different join orders. Some of the transformations are:

- **Join-Swap:** $Q1 \bowtie (Q2 \bowtie Q3) \Leftrightarrow Q2 \bowtie (Q1 \bowtie Q3) \Leftrightarrow Q3 \bowtie (Q2 \bowtie Q1)$
- **Selection-Swap:** $\sigma_A(Q1 \bowtie Q2) \Leftrightarrow \sigma_A Q1 \bowtie Q2$
- **Join-Union-Distribution:** $Q1 \bowtie (Q2 \cup Q3) \Leftrightarrow (Q1 \bowtie Q2) \cup (Q1 \bowtie Q3)$

Distributed environment: These rules capture the characteristics of the distributed environment. For example, the fact that, whenever possible, it is generally more efficient to execute a set of operations remotely (if the source has the appropriate query processing capabilities) than to transmit the data over the network and execute the operations locally at the mediator. Such a rewriting rule in the syntax accepted by our planner is shown in Figure 1.

```
(define-rule :name remote-join-eval
  :if (:operators
       ((?n1 (retrieve ?query1 ?source))
        (?n2 (retrieve ?query2 ?source))
        (?n3 (join ?query ?jc ?query1 ?query2)))
       :constraints ((capability ?source 'join)))
  :replace (:operators (?n1 ?n2 ?n3))
  :with (:operators
         ((?n4 (retrieve ?query ?source))))
```

Figure 1: Remote-Join-Eval Rewriting Rule

Semantic heterogeneity in the domain: These rules are derived from pre-compiled axioms that describe the alternative ways of combining sources to obtain a particular class of information in the domain. These axioms facilitate the exploration of alternative sources for a query. For example, assuming there is one source that provides `airport(geoloc-code port-name)` and another for `location(geoloc-code country-code)`, the fact that the system can obtain `airport(country-code port-name)` by joining these two sources on the key `geoloc-code` is recorded in the axiom:

*AI and Information Integration Workshop (AAAI 98)

```
airport(country-code port-name) ⇔
    airport(geoloc-code port-name) ∧
    location(geoloc-code country-code)
```

The system automatically compiles these axioms given a domain model and source descriptions (Ambite *et al.* 1998). Also, the query planner automatically generates the rewriting rules from the relevant axioms for a given user query. The rewriting rule corresponding to the axiom above is shown in Figure 2. This rule introduces this axiom into the plan, possibly replacing an alternative axiom implementation for `airport(country-code port-name)`. See (Ambite & Knoblock 1998) for details.

```
(define-rule :name
  (<=> (airport country-code port-name)
       (:and (airport geoloc-code port-name)
             (location geoloc-code country-code)))
 :if (:constraints
      ((identify-axiom-steps
        (airport country-code port-name) ?nodes)))
 :replace (:operators ?nodes)
 :with
 (:operators
   ((?n1 (retrieve port@local
           (airport geoloc-code port-name)))
    (?n2 (retrieve geoh@higgledy.isi.edu
           (location geoloc-code country-code)))
    (?n3 (join (airport country-code port-name)
           ((= geoloc-code.1 geoloc-code.2))
           (airport geoloc-code.1 port-name)
           (location geoloc-code.2 country-code))
))))
```

Figure 2: Rewriting Rule for Integration Axiom

These three aspects of mediator systems, namely, the need for traditional query optimization, the distributed environment, and the semantic heterogeneity, are also the sources of complexity that result in the highly combinatorial nature query planning in mediators. Instead of exhaustively searching the space of query plans or optimize each aspect independently, PbR addresses the complexity in the three fronts simultaneously by using local search techniques.

PbR has several characteristics that make it especially well-suited for query planning. First, PbR is a *declarative domain-independent* framework which implies that the planner is easier to understand, maintain and refine than traditional query optimizers. Different domains can be easily specified, for example, for different data models such as relational and object-oriented. The uniform specification of the planner facilitates its *extension* with new capabilities, such as learning mechanisms or interleaving planning and execution. Moreover, a general planning architecture fosters *reuse* in the domain specifications, the search methods and the search control techniques. Second, PbR scales better than other domain-independent planning algorithms. *Scalability* is critical because of the complexity of query planning in mediators. Third, an important advantage of PbR is its anytime nature, which allows it to trade off planning effort and plan quality. For example, a typical quality metric in query planning is the plan execution time. It may not make sense to keep planning if the cost of the current plan is small enough, even if a cheaper one could be found. Finally, the generality of the PbR framework has allowed the design of a novel combination of traditional query optimization and source selection.

I am also interested in the scalability of systems with multiple mediators (Knoblock & Ambite 1997), maintaining accurate source descriptions in a mediator (Ambite & Knoblock 1995), and general issues of knowledge representation and reasoning for information integration.

References

Ambite, J. L., and Knoblock, C. A. 1995. Reconciling distributed information sources. In *Working Notes of the AAAI Spring Symposium on Information Gathering in Heterogeneous, Distributed Environments.*

Ambite, J. L., and Knoblock, C. A. 1997. Planning by rewriting: Efficiently generating high-quality plans. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence.*

Ambite, J. L., and Knoblock, C. A. 1998. Flexible and scalable query planning in distributed and heterogeneous environments. In *Proceedings of the Fourth International Conference on Artificial Intelligence Planning Systems.*

Ambite, J. L.; Knoblock, C. A.; Muslea, I.; and Philpot, A. 1998. Compiling source descriptions for efficient and flexible information integration. Submitted.

Arens, Y.; Chee, C. Y.; Hsu, C.-N.; and Knoblock, C. A. 1993. Retrieving and integrating data from multiple information sources. *International Journal on Intelligent and Cooperative Information Systems* 2(2):127–158.

Arens, Y.; Knoblock, C. A.; and Shen, W.-M. 1996. Query reformulation for dynamic information integration. *Journal of Intelligent Information Systems, Special Issue on Intelligent Information Integration* 6(2/3):99–130.

Knoblock, C. A., and Ambite, J. L. 1997. Agents for information gathering. In Bradshaw, J., ed., *Software Agents.* Menlo Park, CA: AAAI/MIT Press.

Knoblock, C. A.; Minton, S.; Ambite, J. L.; Ashish, N.; Modi, P. J.; Muslea, I.; Philpot, A. G.; and Tejada, S. 1998. Modeling web sources for information integration.

MacGregor, R. 1988. A deductive pattern matcher. In *Proceedings of the Seventh National Conference on Artificial Intelligence.*