

Integrating and Querying Taxonomies with QUEST in the Presence of Conflicts *

Yan Qi
Comp. Sci. and Eng
Arizona State Univ.
yan.qi@asu.edu

Maria Luisa Sapino^{*}
Dip. di Informatica
Universita' di Torino
mlsapino@di.unito.it

K. Selçuk Candan
Comp. Sci. and Eng
Arizona State Univ.
candan@asu.edu

Keith W. Kintigh
Sch. of Human Evol. &
Social Change
Arizona State Univ.
kintigh@asu.edu

ABSTRACT

We present the *Q*Uery-driven *E*xploration of *S*emistructured data and *meta*-data with *conflic*Ts and *partial* knowledge (QUEST) system for supporting the integration of scientific data and taxonomies in the presence of misalignments and conflicts. QUEST relies on a novel constraint-based data model that captures both *value* and *structural* conflicts and enables researchers to observe and resolve such misalignments in the integrated data by considering the context provided by the data requirements of given research questions.

Categories and Subject Descriptors: H.2.5 [Heterogeneous Databases]; H.3.3 [Information Search and Retrieval]; Relevance feedback; H.5.2 [User Interfaces]

General Terms: Algorithms, experimentation

Keywords: Reasoning with misaligned data, conflicts, taxonomy, relevance feedback, query processing

1. MOTIVATION

Large-scale archaeological data are never collected by a single research team, but must be compiled from many independent projects. QUEST (*Q*Uery-driven *E*xploration of *S*emistructured data and *meta*-data with *conflic*Ts and *partial* knowledge [14]) is part of a joint effort of archaeologists and computer scientists to develop a framework of collaborative tools that will enable a shared information infrastructure for archaeology [3]. In archaeology, while there is widespread agreement about the importance of specific classes of observations, there are relatively few completely established conventions. The meaning of an archaeological observation is rarely self-evident. Researchers reach conflicting conclusions, not just because their primary data differ, but because they operationalize interpretive concepts differently. For example, when researchers are unable to precisely identify the species of a bone collected at a given site, they may use less specific terms (or references to a hierarchically higher concept in the biological taxonomy) that lack a

*Supported by NSF Grants “Enabling the Study of Long-Term Human and Social Dynamics: A Cyberinfrastructure for Archaeology (0433959)” and “Archaeological Data Integration for the Study of Long-Term Human and Social Dynamics (0624341)”

*This work was done while the author was visiting ASU

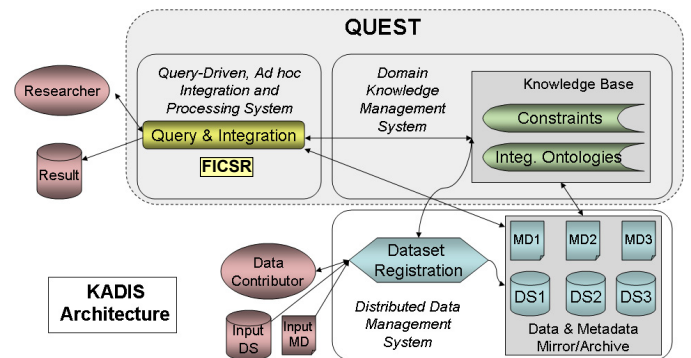


Figure 1: QUEST architectural overview

generally accepted interpretation. Thus it is necessary to maintain the semantic content of original observations while allowing the extraction of the maximum meaning possible from existing datasets. An ultimate integrated view of knowledge is often not possible or not needed and overly-eager conflict resolution may be detrimental to the effective use of the available knowledge. Based on these observations, we demonstrate the QUEST system (Figure 1) that supports exploratory research on the incomplete and conflicting data and taxonomies, based on the query driven data integration and exploration paradigm. QUEST relies on the FICSR query processor for feedback-driven inconsistency resolution and reasoning on misaligned data sources [15].

1.1 Related Work

Integration of data from different sources starts with a *matching/alignment* phase. Work on automated matching systems, which take data or schemas as input and produces a mapping or concept similarity, include [2,5,7,9–13]. Despite such advances in mapping technologies, alignments across the data sources are rarely perfect. Uncertain information at the data level has been modeled by means of different types of *null* values in databases [6]. [8] presents an attempt at modeling partial data in semistructured domains.

TRIO [1] is a system in which relations are extended to store a measure of the uncertainty (probability) associated to the data. Coherence in the use of alternative data values in the relational tuples is maintained through lineage. *Orchestra* [16] is specifically conceived for dealing with multiple cooperating, and possibly mutually inconsistent, data sources. *Momis* [4] implements a semiautomatic integration methodology that follows the global-as-view approach,

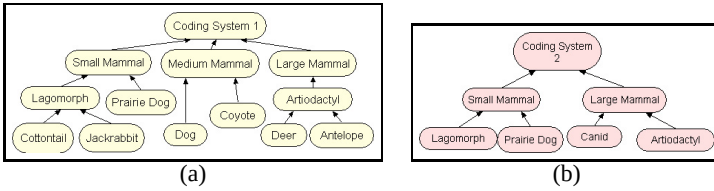


Figure 2: Fragments of two different taxonomies, which represent different views on the mammal categorization

in which a view over the data sources is associated with each element of the global schema.

In [14] we introduced a constraint-based model, which enables representation of not only value-conflicts, but also *structural* conflicts and null-values. We also discussed the use of this model for Query-driven Exploration of Semistructured data with conflict s and partial knowledge (i.e., QUEST). Our approach to integration lies in enabling researchers to *observe the conflicts in the data* and *provide feedback to resolve them* by considering the context provided by the data requirements of a given research question. In this demo, we illustrate the use of QUEST for archaeological data integration and reasoning under conflicts.

1.2 Running Example

Figure 2 is an example of two similar, but different taxonomy fragments (or coding systems). In this example, both systems classify concepts such as *Small Mammal*, *Large Mammal*, *Lagomorph* and *Artiodactyl*, but the names of the first two concepts have inconsistent semantics. For instance, the concept ‘Large Mammal’ in the second coding system is more general than in the first one. There is no a priori rule for preferring one coding system to the other, and their integration might need the domain expert’s (maybe context-dependent) knowledge interpretation. Depending on the query context, for example, the concept *Large Mammal* in the second coding system could (or could not) be seen as a superclass of both of *Medium Mammal* and *Large Mammal* in the first taxonomy.

2. QUEST USE FLOW

Figure 3 illustrates the overall use flow for QUEST. The system combines two main phases: one devoted to (offline) data integration and analysis and the second one on (online) query processing and feedback-based conflict resolution. In the next section we briefly describe the roles of the individual steps in these two phases.

2.1 Offline Phase: Data Integration and Analysis

The data integration phase combines the given (possibly misaligned/conflicting) taxonomies into a single graph-based data structure, in which the information captured by the individual taxonomies (as well as the additional domain knowledge available to the domain expert users in charge of the integration) is coded. In addition, whenever the graph structure is not sufficient to describe the alignment semantics, additional constraints are recorded.

Step 1: Concept matching. QUEST provides semi-automated alignment between taxonomies. An initial automatic mapping is suggested by the system [5, 7]. The domain expert is then given the opportunity to tune the default mapping: she can require some concepts to be kept as distinct or she can add mapping specifications on (sets of) nodes in the integrated taxonomies. A graphical interface allows the domain expert to define her mappings among the concepts appearing in the taxonomies.

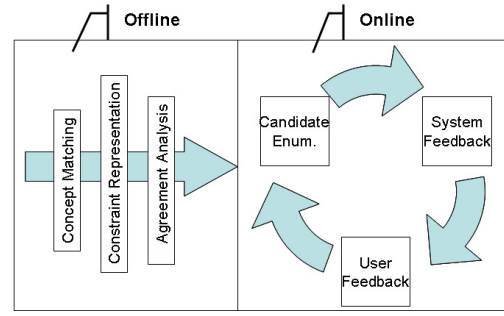


Figure 3: Overview of the QUEST use flow

Referring to the examples in Figure 2, a possible matching is $[SmallMammal]_1 \equiv [SmallMammal]_2$ (asserting the identity of the concepts denoted as *SmallMammal* in the two given taxonomies),

$[Lagomorph]_1 \equiv [Lagomorph]_2$,
 $[PrairieDog]_1 \equiv [PrairieDog]_2$,
 $[Artiodactyl]_1 \equiv [Artiodactyl]_2$,
 $[LargeMammal]_1 \equiv [LargeMammal, MediumMammal]_2$,
mapping the concept *LargeMammal* in the first taxonomy on the set of concepts $\{LargeMammal, MediumMammal\}$ in the second.

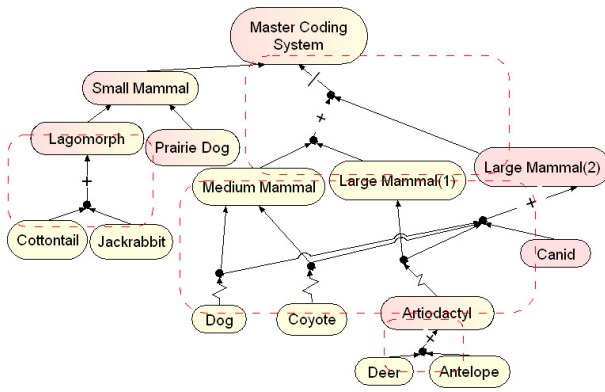
In addition to the arity constraints implicitly defined by the hierarchical structure of the taxonomies, the user can introduce coordination/synchronization/mutual exclusion constraints, as well as different kinds of arity constraints, limiting for example the branching factors of the nodes.

Step 2: Preliminary Integration and Constraint representation.

The previous step specifies the concept matching requirements for all the concepts mentioned in the given taxonomies. Step 2 integrates these into an intermediary graph-based data structure and additional data integrity constraints (whenever the graph is not able to capture the requirements) [14]. Figure 4 shows the constraint graph resulting from the integration of the fragments in the running example. In this graph, dashed rectangles identify the *zones* corresponding to localized alignment and choice constraints within the graph. For example, the leftmost zone includes a *positive coordination constraint*: in any interpretation of the integrated data, the class *Logomorph* should include the concept *Cottontail* iff it also includes *Jackrabbit*. Analogously, the top zone asserts a *negative coordination constraint* between the descendants of the root: any interpretation of the integrated data can either contain the concept *Large Mammal* corresponding to the coding system 2, or the couple of concepts *Medium Mammal* and *Large Mammal* corresponding to the coding system 1. Other coordination specifications in the graph includes *mutual exclusion constraints*, which for instance, specify that the immediate ancestor of *Dog* is either *Medium Mammal* or *Large Mammal* depending upon which coding system was used.

Note that a user can provide additional integrity constraints, such as “*Coyote* is not a *Large Mammal*”. Other examples of such constraints (to be satisfied by any interpretation of the integrated data) include acyclicity requirements, constraints on single paths (e.g., constraints on the allowed distance between two given nodes along any path in the integrated taxonomy, or constraints on the order according to which some nodes can occur in the returned paths), and constraints on multiple paths (for example, paths disjointness requirements).

Step 3: Preprocessing for agreement analysis. This step does not require the direct participation of the expert user. The FICSR



Additional integrity constraint: "Coyote is not a Large Mammal"

Figure 4: Integrated taxonomy fragments

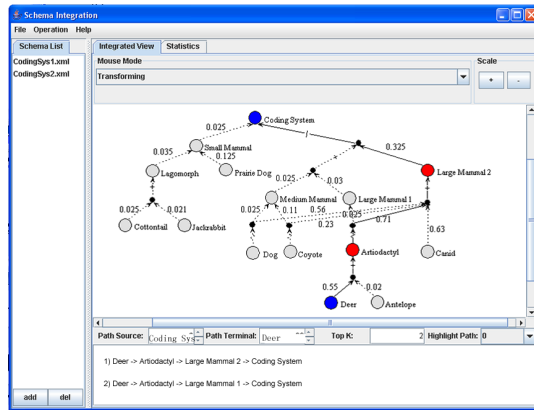


Figure 5: Use of the QUEST interface for locating "deer" in the integrated taxonomy

pre-processing module extracts alignment measures from the integrated data sources (i.e., from the graph) and edges in the graph are associated with values representing the degree of alignment among the given taxonomies [15]. Coherent and consistent parts of the taxonomies get highest (close to 1) agreements, while values are close to 0 at parts that are highly misaligned.

2.2 Online Phase: Querying and System/User Feedback

QUEST relies on the graph that integrates the given taxonomies and on the *objective* measure (computed by FICSR [15]) of the source agreements for enumerating matches. In response to a query, QUEST might present conflicting alternatives (depending on the degree of misalignment of the original taxonomies). To help her choose among alternatives, QUEST assists the user through a system/user feedback process.

Step 4: Querying. In this step, QUEST provides a query interface to the user (Figure 5). At the later stages, this interface also visualizes the system feedback to the user and enables her to provide feedback.

Step 5: Candidate enumeration and ranking. Given the query, the FICSR processor identifies and ranks an initial set of candidate matches, using the agreement values computed in the offline phase. The candidate results can be mutually inconsistent with respect to the existing constraints.

Step 6: Constraint analysis and system feedback. To measure

an *objective* degree of conflict associated with integrity constraints and provide this information to the user as system feedback, the FICSR engine *fuzzifies* the available constraints and formulates an optimization problem whose *objective* is to identify a maximal solution in which validities of the constraints are maximized. This process may negatively affect the rankings of the candidate paths that conflict with the integrity constraints.

Step 7: User's feedback At this point the user is presented through QUEST with ranked path results as well as with a corresponding objective measures of conflicts with respect to the constraints. The user can provide her *subjective* feedback and the underlying FICSR engine incorporates this user feedback into the data. User's validity feedback is primarily in the form of preferred validity rankings and assessments for constraints, conflicts, and candidate paths. Based on her knowledge about the domain, the user can choose to assess a lower validity on certain candidate or give lower importance to some constraints.

The user's feedback is incorporated as additional constraints into the optimization problem and she is presented with a new set of candidate results.

3. DEMO SCENARIO

The demo involves archaeological taxonomy alignment, integration, and query processing on imperfectly aligned data sources. We demonstrate both off-line and on-line phases of the system and how system/user feedback enables the user to resolve conflicts, in archaeological and other related data sources, within the context of user's queries.

4. REFERENCES

- [1] O. Banjelloun, A. D. Sarma, A. Halevy, J. Widom. ULDBs: Databases with uncertainty and lineage. *VLDB*, 2006.
- [2] A. Doan, P. Domingos, A. Y. Levy. Learning source description for data integration. *WebDB*, 81-86, 2000.
- [3] Kintigh, Keith W. (Ed.) The Promise and Challenge of Archaeological Data Integration. In *American Antiquity* 71(3): 567-578, 2006
- [4] D. Beneventano, S. Bergamaschi, F. Guerra, M. Vincini. Synthesizing an integrated ontology. In *IEEE Internet Computing Magazine*, 2003.
- [5] K. S. Candan, J.W. Kim, H. Liu, R. Suvama. Discovering mappings in hierarchical data from multiple sources using the inherent structure. *J. of KAIS*, 2006.
- [6] T. Imielinski and W. Lipski. Incomplete information in relational databases. *JACM*, 31(4):761-791, 1984.
- [7] J.W. Kim and K.S. Candan. Concept similarity mining without frequency information from domain describing taxonomies, *CIKM*, 2006.
- [8] M. Liu, T. W. Ling. A data model for semistructured data with partial and inconsistent information. *LNCS 1777*, 2000.
- [9] J. Madhavan, P. A. Bernstein, and E. Rahm. Generic schema matching with cupid. *VLDB*, 2001.
- [10] R. Miller, L. Haas, M. Hernandez. Schema mapping as query discovery. *VLDB*, 2000.
- [11] T. Milo, S. Zohar. Using schema matching to simplify heterogeneous data translation. *VLDB*, 1998.
- [12] L. Palopoli, D. Sacca, D. Ursino. An automatic technique for detecting type conflicts in database schemes. *CIKM*, 1998.
- [13] P.Mitra, G.Wiederhold, M.Kersten. A graph oriented model for articulation of ontology interdependencies. *EDBT*, 2000.
- [14] Y. Qi, K.S. Candan, M.L. Sapino, K. Kintigh. Quest: Query-driven exploration of semistructured data with conflicts and partial knowledge. *CleanDB*, 2006.
- [15] Y. Qi, K.S. Candan, M.L. Sapino. FICSR: Feedback-based InConsistency Resolution and query processing on misaligned data sources. *SIGMOD*, 2007.
- [16] N. E. Taylor, Z. G. Ives. Reconciling while tolerating disagreement in collaborative data sharing. *SIGMOD*, 2006.