

# Towards Building a MetaQuerier: Extracting and Matching Web Query Interfaces\*

Bin He, Zhen Zhang, Kevin Chen-Chuan Chang  
Computer Science Department  
University of Illinois at Urbana-Champaign  
{binhe, zhang2}@uiuc.edu, kcchang@cs.uiuc.edu

## 1. Introduction

Recently, we witness the rapid growth and thus the prevalence of databases on the Web. Our recent study [1] in April 2004 estimated 450,000 online databases. On this deep Web, myriad databases provide dynamic query-based data access through their *query interfaces*, instead of static URL links. It is thus essential to integrate these query interfaces for integrating the deep Web.

The overall goal of the MetaQuerier project (<http://metaquerier.cs.uiuc.edu>) aims at opening up the deep Web to users, by building a system to help users exploring and integrating deep Web sources. In particular, to start with, we focus on the integration of deep Web sources in the same domain, which is itself an important integration task. The typical scenarios include purchasing a book with lowest price among book sources and a flight ticket with the best trade-off between price and number of connections among airline sources. The deep Web presents challenges for such *large-scale* integration: With plenty of databases in one domain, how can we integrate them to facilitate user queries?

To automate this integration scenario, we need to solve two critical problems: Extracting query interfaces (i.e., extracting the attribute information, given a query interface in HTML format) and matching query interfaces (i.e., discovering the semantic correspondences among attributes). By addressing these two core problems, we are able to develop an application of matching any two query interfaces. We believe such an application is important on its own right because it is the preceding step for enabling automatic query translation between sources. For instance, if a user filled the book search interface in *amazon.com*, to translate the user's query to other book sources such as *bn.com*, it is essential to match the attributes from *amazon.com* to other sources. In particular, the scope of this demo is to automate the two

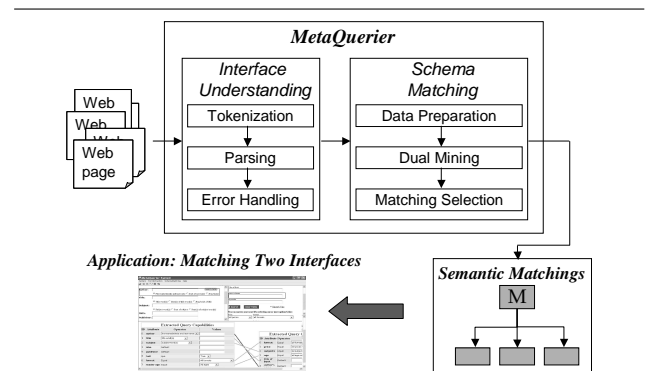


Figure 1. The system framework.

core steps and further illustrate this application of matching any two query interfaces.

As a new attempt, our core techniques leverage the large scale “regularity” of Web query interfaces to explore their hidden “semantics” [3]. Specifically, to solve the interface extraction problem, we introduce a *parsing* paradigm by hypothesizing the existence of *hidden syntax* which describes the layout and semantic of Web interfaces [8]. Also, unlike traditional pairwise schema matching, we propose a holistic matching approach, which matches all schemas at the same time with the hypothesis of a *hidden schema model* [4, 5]. Therefore, our techniques explore, in essence, “data mining for information integration.” That is, we mine the observable information to discover the underlying semantics.

Overall, MetaQuerier is a fully integrated streamlined system to automatically match Web interfaces. In the literature, large-scale integration or *meta-search* of structured information sources has largely been left unexplored. Some recent works such as [7] also aim at integrating Web interfaces. However, they all assume that query interfaces have been perfectly extracted. In contrast, the MetaQuerier system fully integrates the extraction of interfaces from raw HTML pages with subsequent schema matching.

\* This material is based upon work partially supported by NSF Grants IIS-0133199 and IIS-0313260. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the funding agencies.

## 2. The MetaQuerier System

The MetaQuerier system consists of two subsystems in a sequence: interface extraction and schema matching, as Figure 1 illustrates. The input of the MetaQuerier system is a set of Web pages (containing query interfaces) in the same domain. In practice, crawling (e.g., our IP based crawling [1]) and clustering (e.g., our work on clustering Web interfaces [6]) techniques can be used to get such input. In this demo, for the purpose of simplifying the illustration, we show the MetaQuerier system on the TEL-8 dataset of the UIUC Web Integration Repository [2], which contains deep Web sources well-classified into 8 domains.

**Interface Extraction:** Interface extraction extracts the attribute information (e.g., names and domain values) from the Web query interfaces in HTML format. We observe that, in Web interfaces, *conditions patterns* are quite regular in layout. For instance, the most frequently used condition pattern is a text (as the attribute name) followed by an input box (as the attribute value). Our survey [1] shows that Web interfaces, although designed autonomously, often share a small set of condition patterns. We thus hypothesize the existence of the *hidden syntax* of these interfaces. The hidden syntax describes the composition from condition patterns to Web interfaces based on the layout information. With this hypothesis, we pursue a *parsing* approach for understanding Web interfaces [8], considering the hidden syntax as the grammar to parse. As Figure 1 shows, the interface extraction consists of: 1) *tokenization* – accepts Web pages as input and outputs a set of *tokens* as instances of terminal symbols (e.g., text, inputbox) in the grammar, 2) *parsing* – accepts tokens generated by the tokenizer and derives best-effort parse trees based on the grammar, and 3) *error handling* – handling conflicts and errors in the parsing result such as the situation of multiple parse trees.

**Schema Matching:** Schema matching discovers the semantic correspondences (i.e., matchings) among the attributes in Web interfaces. For instance, in Books domain, **author** is the synonym of the grouping of **last name** and **first name**, i.e.,  $\{\text{author}\} = \{\text{first name}, \text{last name}\}$ ; Also, we have  $\{\text{subject}\} = \{\text{category}\}$  and  $\{\text{format}\} = \{\text{binding}\}$ . Traditionally, schema matching relies on matchings between pairwise schemas before integrating multiple ones. In contrast, we propose to exploit statistical analysis to holistically match many schemas at the same time [4]. Specifically, we observe that *grouping attributes* (i.e., attributes in one group of a matching e.g.,  $\{\text{last name}, \text{first name}\}$ ) tend to be co-present and thus positively correlated across sources in the same domain. In contrast, *synonym attributes* (i.e., **subject** and **category**) are negatively correlated because they rarely co-occur in schemas. These observations motivate us to develop a mining approach to match schemas [5], consisting of: 1) *data preparation* – as preprocessing, data preparation accepts extracted attributes as input and outputs cleaned data by merging syntactically similar attributes, 2)

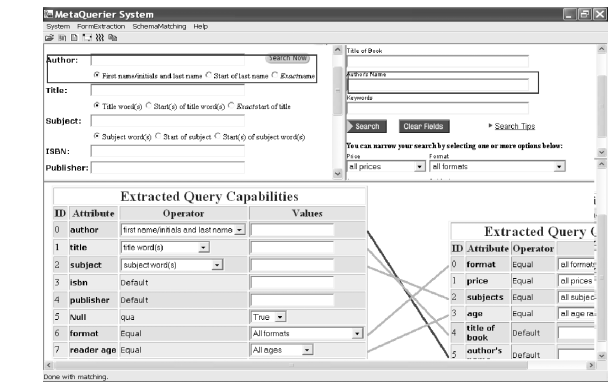


Figure 2. Application: matching two query interfaces.

*dual mining* – discovering matchings with a dual mining of positive correlation (i.e., potential groups) and negative correlation (i.e., potential matchings), 3) *matching selection* – choosing the most convincing and consistent matchings from the mining result.

## 3. Demonstration

The MetaQuerier system fully automates all the tasks in streamline to output semantic matchings. For demonstration purpose, the MetaQuerier system also supports “single-step” operation to explicitly illustrate the function of each task. Also, based on discovered matchings, we further perform the matching of any two query interfaces, which is an important step to enable query translation between sources. In particular, a user can choose any two book sources and the system will show the matchings between their attributes, as Figure 2 illustrates.

## References

- [1] K. C.-C. Chang, B. He, C. Li, M. Patel, and Z. Zhang. Structured databases on the web: Observations and implications. *SIGMOD Record*, 33(3):61–70, September 2004.
- [2] K. C.-C. Chang, B. He, C. Li, and Z. Zhang. The UIUC web integration repository. Computer Science Department, University of Illinois at Urbana-Champaign. <http://metaquerier.cs.uiuc.edu/repository>, 2003.
- [3] K. C.-C. Chang, B. He, and Z. Zhang. Metaquerier over the deep web: Shallow integration across holistic sources. In *Proceedings of the VLDB Workshop on Information Integration on the Web*, 2004.
- [4] B. He and K. C.-C. Chang. Statistical schema matching across web query interfaces. In *SIGMOD Conference*, 2003.
- [5] B. He, K. C.-C. Chang, and J. Han. Discovering complex matchings across web query interfaces: A correlation mining approach. In *SIGKDD Conference*, 2004.
- [6] B. He, T. Tao, and K. C.-C. Chang. Organizing structured web sources by query schemas: A clustering approach. In *CIKM Conference*, 2004.
- [7] H. He, W. Meng, C. Yu, and Z. Wu. Wise-integrator: An automatic integrator of web search interfaces for e-commerce. In *VLDB Conference*, 2003.
- [8] Z. Zheng, B. He, and K. C.-C. Chang. Understanding web query interfaces: Best-effort parsing with hidden syntax. In *SIGMOD Conference*, 2004.