Schemaless Representation of Semistructured Data and Schema Construction

Dong-Yal Seo¹, Dong-Ha Lee¹, Kang-Sik Moon¹, Jisook Chang¹, Jeon-Young Lee¹, and Chang-Yong Han²

¹ Dept. of Computer Science and Engineering
Pohang University of Science and Technology
Pohang, Kyungbuk, 790-784, KOREA
² Data Warehouse Advanced Technology
Oracle Systems Korea, Ltd.
Youngdeungpo-Gu, Seoul, 150-010, KOREA

Abstract. We should consider semistructured data of which have a weak schema information in networked information world. To manage such semistructured data efficiently, this paper introduces a data model for semistructured data and operations for schema construction. We transform semistructured data into structured one by introducing schema construction methodology, compared to the former studies which are fully dependent on schemaless manipulations. For schema construction, we defined operations for building IS-A/IS-PART-OF relationships, collecting data objects to build a primitive class, and merging two data instances or classes.

1 Introduction

1.1 Motivation

In early stages of data processing such as inventory/account management systems, a centralized large database system was used as an information server. Through the database design (or real-world modeling) phase, DBA(Database Administrator) defines a well-structured schema. We perform data acquisition and creation after schema definition. For end-users, their role is to manipulate data via predefined schema. The schema is firm and end-users are not responsible for schema management.

As data sources and computing environment are distributed, abundant of information is provided by individual users and updated very quickly. Each end-user creates and updates his/her own information, like DBA. The WWW(World-Wide Web) is a typical domain of the examples. Every user creates his/her own documents and submitts in the WWW. How to manage those plenty of HTML documents and other web resources? We should always navigate through hyper-links or search by keywords because there is no absolute schema in the stored information. If we could define a schema on the set of web resources, not only we have a better structure of gathered information but also we can use a structured query language. Schema provides the well-structured view of
stored data. It expresses data location, relationships among data objects, data categories, summarized concepts, and so on.

The data sets, where there is no absolute schema fixed in advance and whose structure may be irregular or incomplete, are known as *semistructured data*. Even though a data is created as a well-structured, i.e., schema-based, set of data, it becomes semistructured when the data comes out from its original structure. For example, a single record from a relational table is semistructured if we have no idea about the overall structure of the table and the relationships with other tables.

### 1.2 Problems and Approaches

Figure 1 shows the approaches of information processing based on the structural nature of data sources. Right side of the figure presents conventional data processing. Information itself has a rigid structure and is represented with a well-structured model, like relational or object-oriented. Users manipulate data with a schema which mainly provides an structural abstraction of stored data.

![Fig. 1. Information Structures and Processings](image)

Left side shows the processing of information with poor schema, i.e. *semistructured information* [12] (or even *unstructured* [3]). Semistructured information is represented with a lightweight model, which permits schemaless creation of data instances, and is stored in a lightweight storage. Stored information is manipulated with a lightweight query language, which can be used with incomplete schema information.

The studies on lightweight approaches much overlooked the importance of database schema. Although schemaless manipulation is convenient for users who want to retrieve data without deep knowledge of underlying structures, schema is indispensable for embedded SQL, API calls, or stored procedures. Schema gives firmness and conceptualized view.

In semistructured data processing, end-users represent data instances without complete knowledge of the predefined schema. (Or even without any schema