# THE ASSOCIATIVE DATA MODEL VERSUS THE RELATIONAL MODEL: A PEDAGOGICAL COMPARISON

*Richard L. Jenson, School of Accountancy, Utah State University, 3540 Old Main Hill, Logan, UT 84322-3540, (435) 797-2335, richard.jenson@usu.edu*

## ABSTRACT

The associative model of data has been proposed as a "small footprint" alternative to the highly dominant relational database model. A drawback of the relational model is the extensive table proliferation required to support the various processes modeled therein. The associative data structure is vertically defined with meta-data and data contained together. Database attributes are represented as associations, not as fields. This model also minimizes domain redundancy, allowing an entity to be represented only once in the system, yet captures the various roles in which the entity interacts with the system. The author presents a pedagogical example that enables students to contrast the features of the relational and associative data models.

## INTRODUCTION

Despite being the most dominant data model for enterprise applications, the relational database model has been criticized for its extensive table proliferation -- hence it extensive structural footprint. Each additional process that is modeled in the relational database adds a related family of tables to support the concepts and linkages required. In addition, relational tables are heterogeneous in that each table necessarily varies structurally from other tables in the database with a variable number of fields and field types in each table. Linkages are maintained through primary key-foreign key relationships, thus adding to the model's structural complexity. The relational model imposes an entity redundancy that requires an entity to be represented separately in the database for each role it assumes.

The associative model of data has been proposed as a "small footprint" alternative to the highly dominant relational database model. The associative database represents information as items (having independent existence) and linkages among such items. The associative data structure is vertically defined with meta-data (data structures) and data contained together. Database attributes are represented as associations rather than fields. Items are not all required to have the same associations – they may vary by item. Therefore, null values play no role. This allows for a more granular storage of data; entity data are not stored contiguously as in a relational table. In addition, with meta-data embedded with instance data, associative program code and data structures are highly portable across multiple application platforms. This model also minimizes domain redundancy, allowing an entity to be represented only once in the system, yet capturing the various roles in which the entity interacts with the system. The vertical definition of the data structure, and the embedding of such definitions within the data, facilitates the transport of the data in a tagged format such as XML.

The author presents a pedagogical case example used to introduce students to the associative data model and to enable them to contrast the characteristics of the relational and associative data models. Student teams develop a basic sales order-entry prototype using Sentences, an associative database development tool. Data for the prototype are based on the same relational dataset previously used by the students in an MS-Access tutorial, allowing students to see how each database approach modeled the same information. In the course of this project, students gain further experience in data modeling using the associative schema and design methodology. In addition, students receive a hands-on XML experience using Sentences' XML export utility to facilitate the transport of their associative data from their Sentences application to other applications, such as Excel.