

8-25-1995

# Coupling Enterprise Planning with the Creation of the Conceptual Schema in Database Design

Paul F. Fuhs

Virginia Commonwealth University, pfuhs@cabell.vcu.edu

Follow this and additional works at: <http://aisel.aisnet.org/amcis1995>

---

## Recommended Citation

Fuhs, Paul F., "Coupling Enterprise Planning with the Creation of the Conceptual Schema in Database Design" (1995). *AMCIS 1995 Proceedings*. 61.

<http://aisel.aisnet.org/amcis1995/61>

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 1995 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact [elibrary@aisnet.org](mailto:elibrary@aisnet.org).

# **Coupling Enterprise Planning with the Creation of the Conceptual Schema in Database Design**

**By F. Paul Fuhs, Ph.D.  
Virginia Commonwealth University  
School of Business  
Department of Information Systems  
Box 844000  
Richmond, Va. 23284  
pfuhs@cabell.vcu.edu**

## **Introduction**

Enterprise Planning is a structured approach to help a corporation establish an information system plan at a high enough level of abstraction to model the primary business sub-systems and applications. Whether the planning technique is the popular IBM Business Systems Planning (BSP) (IBM, 1984) or one of its derivatives, such as strategic information systems planning (Lederer and Sethi, 1992) the deliverables include the identification of the major business processes, their associated data classes, and the applications to which the business processes and data classes belong. The association among these three elements in BSP is referred to as the Information Architecture. The assumption is that the planning deliverables will be used throughout the rest of the SDLC, beginning in a top-down fashion in the analysis phase of development.

The first stage of database development following enterprise planning is the identification and specification of subject databases and the development of a logical model to support the conceptual schema and its external schemas. The development of subject databases is frequently performed with a bottom-up approach, using hierarchical clustering in which entities are grouped together into databases according to their common participation in business processes. Tools such as enhanced ER diagrams (Elmasri, 1994) are most commonly used for the logical model.

The development of the conceptual view and its implementation as the conceptual schema is also a bottom-up effort in that it is, at the least, the integration of all of the user views. It is tempting to carry out both of these database development efforts in a purely bottom-up approach, with little reference to previous enterprise planning. The temptation to disregard what has occurred during planning becomes greater with increased emphasis on prototyping and the misuse of rapid application development. The difficulty of applying data modeling to the entire enterprise has been noted (Scheer and Hars, 1992).

The top-down/bottom-up dichotomy creates a coupling mismatch between enterprise planning and logical database development. This has been noted by many authors and with respect to BSP by Barlow, 1990. This mismatch can result in one of two reactions. The first is a reliance on ad hoc methods to couple planning and analysis. The second is to de-emphasize the outcome of the planning process or worse -- to merely give lip service to it.

### **Levels of Abstraction and Mapping Matrices**

Figure 1 shows mapping matrices that can be created to harmonize the top-down/bottom-up mismatch of different levels of abstraction as well as intermediary steps that help reduce the abstraction distance between business systems planning and the conceptual schema.

Within the business systems planning box of figure 1 is depicted the Information Architecture and its relationship to its component business processes and data classes. The term data classes as used in BSP refers to an identifiable grouping of unspecified attributes at a high level of abstraction. For example, "hiring data" and "forecasted sales" are data classes. The term data classes is different from, and only remotely associated with the OOP concept of classes. Having identified the data classes, one can then proceed to identify the business entities to which these data classes belong. Traditionally a mapping matrix associating data classes and entities is then constructed.

### **The Planning/Analysis Transition**

Figure 1 also shows that part of the analysis phase that extends from the creation of business activities through the development of the subject databases and on to the development of the conceptual schema. The transition from planning to analysis under the process-oriented aspect of the SDLC model has traditionally decomposed (for a chosen application) business processes into business activities and then decomposed the latter into business tasks that are at a low enough level of abstraction to enter the programming phase of development.

It is here recommended that, instead of defining the subject databases directly from the intersection of the business processes and high level business entities of the business systems planning phase, the analyst drop down a level of abstraction along both the process and data components of system development. That is, after decomposing business processes into business activities, a mapping matrix of business activities and *their* associated data classes should be constructed. This enhanced set of data classes would include data classes identified during planning as well as new data classes at lower levels of abstraction associated with the business activities. The enhanced set of data classes should then be analyzed to construct an enhanced set of business entities. The latter set includes entities identified during planning as well as new entities at these lower levels of abstraction.

The identified business activities and their associated enhanced set of entities may then be used as input to hierarchical clustering for the identification of the entity clusters that compose the subject databases. This approach would create a finer grained set of subject databases.

Given identified business activities and enhanced sets of data classes and entities one can then create the additional mapping matrices shown in figure 1 as the process/activity matrix, the DC/\*DC matrix, and the \*DC/\*E matrix. These matrices tie together the business processes, activities, data classes, and entities at different levels of abstraction.

### **The Conceptual Schema**

The development of the Conceptual Schema as the integration of individual user views at different levels of abstraction within the business enterprise assumes that the data views that are virtually contained within the conceptual schema can be mapped back to the subject databases. The actual integration of these views has been treated in numerous articles, such as (Batini, et al., 1986). The mapping of these views to the subject databases has been left mostly to ad hoc approaches.

Figure 1 shows a suggested user view/entity mapping matrix (V/E). This matrix represents a many-to-many relationship between user views and business entities. This matrix associates entities and views at different levels of abstraction within the strategic, tactical, and operational levels within the enterprise. The entities within V/E are of three kinds. Firstly, there are those from planning. Secondly, there are those entities (\*E) that participated in subject database determination. Thirdly, there are those that are the result of the decomposition of business activities into tasks (not shown in figure 1).

The construction of the view/subject database matrix (V/SDB) can be performed as follows. For each user view expressed in an enhanced ER model, extract the set of entities. This step is trivial. Then for each of these entities, identify the single subject database to which it belongs. Under the goal of little or no data redundancy among databases each entity should map to only one subject database. This is the entity/subject database (E/SDB) matrix of figure 1.

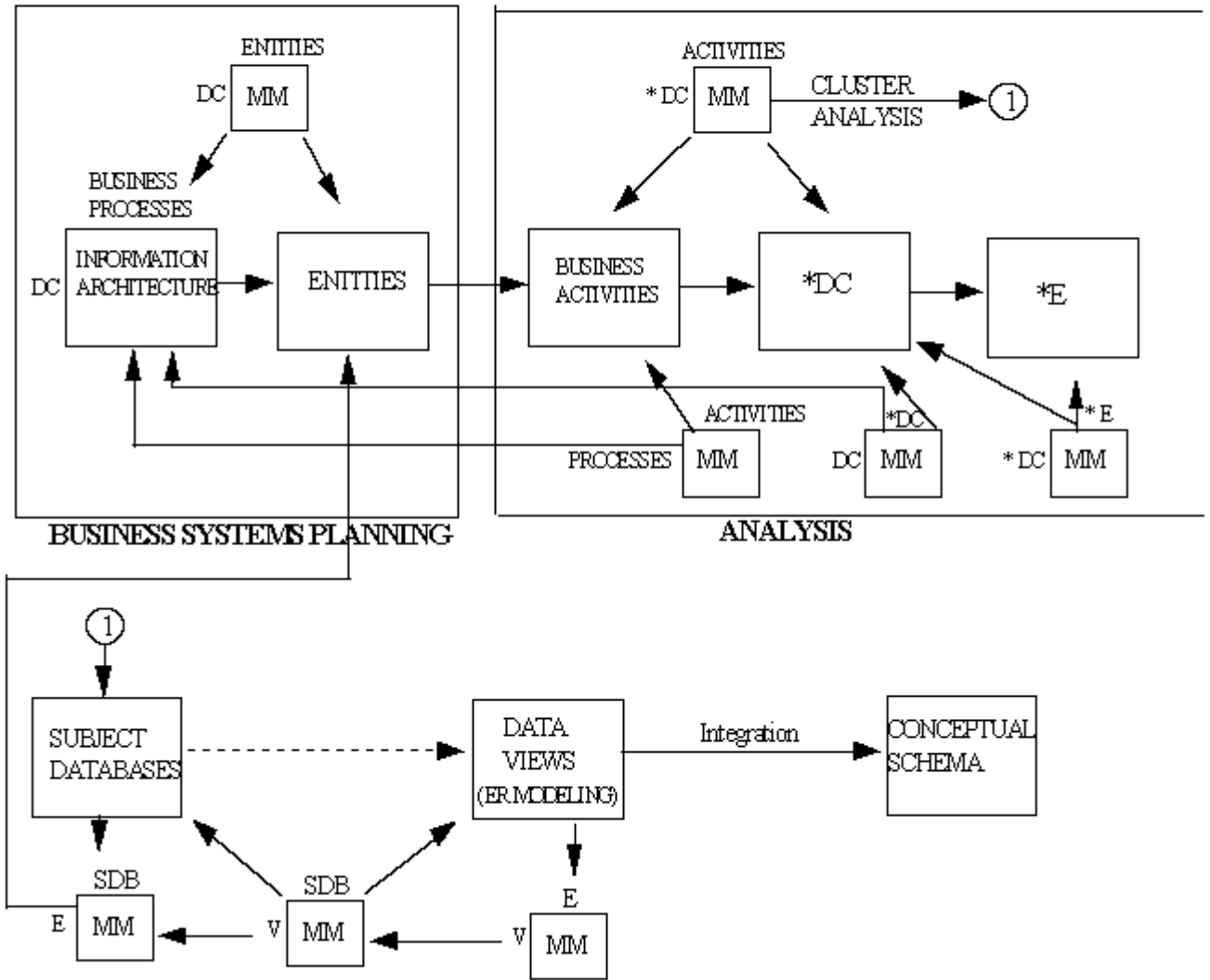
A given entity within a view may be at a level of aggregation or inheritance above or below that specified within the set of subject databases. Therefore, to make the association between an entity within a user view and its associated subject database, one may need to progress through an association trail of mapping matrices. Consequently, the entity/subject database matrix (E/SDB) is in reality a composite, including all entities at all levels of abstraction, all of which map back ultimately to the entities within business planning. The E/SDB associations can be implemented either as a single matrix containing all of the entities or better yet, as a *set* of entity/entity matrices that link together entities existing at different levels of abstraction.

### **Conclusion**

This paper shows how to create and maintain associations between the processing and data content of the Information Architecture and derived subject databases using mapping matrices at different levels of abstraction. Then it demonstrates how to associate the combined content of the Information Architecture and the subject databases with the user views composing the conceptual schema. This is a framework for extending to logical database design the associations among business processes and data developed within enterprise planning. A sequence of mapping matrices (MM) is presented that support the construction of these associations under the preservation of the bottom-up approach for conceptual schema development and a combined top-down/bottom-up approach for the specification of subject databases. With this set of mapping matrices in hand one has multiple Ariadne threads to integrate the decomposed business processes and data through multiple levels of abstraction. The implementation of these matrices can be integrated into already existing data dictionaries that currently store some of this associational information (albeit in haphazard fashion).

**FIGURE 1: The Mapping Matrices (MM) for Logical Database Design**

Legend: MM Mapping Matrix  
 DC Data Classes  
 \*DC Enhanced Set of Data Classes  
 E Entities  
 \*E Enhanced Set of Entities  
 SDB Subject Databases  
 V Views



Complete References will be provided upon request.