IV

Database Design

29 Extending Existing Dependency Theory to Temporal Databases
30 Temporally Enhanced Database Design
31 Temporal Entity-Relationship Models—a Survey
32 Conceptual Modeling of Time-Varying Information
33 From Temporal ER Models to Relations

Database design is an important topic when attempting to make effective use of database technology. Without a well-designed database, the database system is unlikely to adequately support the applications that depend on its services. As a testiment to its importance, database design is covered prominently in most general database textbooks, and a number of books exist that are dedicated entirely to this topic. Database design is also an important research topic in temporal databases.

It is useful to distinguish between two quite different objectives of the research in temporal database design. First, several dozen temporal data models exist, many of which introduce temporal relation structures that deviate quite significantly from those of the relational model. Similar to when using the relational model, guidelines are needed for the design of databases using any of these models; and the existing normalization theory is generally not readily applicable to these models and falls short providing adequate guidelines. This creates a need for transitioning existing normalization theory to temporal data models and for additional design guidelines that address the temporal semantics of data.

Second, conventional relational databases generally record temporal data. The insights obtained through temporal database research into the temporal aspects of data may be exploited to devise guidelines that inform the design of conventional relational databases containing temporal data.

Database design in practice proceeds at separate logical and conceptual levels, and so does the research in temporal database design. At the logical level, normal forms, which are measures of the quality of relation schemas, play a prominent role in guiding database design; as do other guidelines for designing relation schemas. At the conceptual level, the research leads to the design of temporally enhanced conceptual models, most prominently temporal Entity-Relationship models, to guidelines for their use, and to rules for the transformation of temporal ER diagrams to relational database schemas, expressed in the conventional relational model or in a temporal relational model.

The chapters in this part cover most of this spectrum. Specifically, Chapter 29 faces the challenge of transitioning the existing normalization concepts to the numerous existing temporal data models, thus providing essential guidelines for database design in the context of these models. Rather than generalizing existing normalization concepts—dependencies, keys, and normal forms—separately for each of the existing temporal data models, the chapter adopts a wholesale approach. Based on the notion of snapshot equivalence, it translates conventional normalization theory to one temporal data model and then exploits the framework of mappings between models put forward in Chapter 6 to relate the resulting temporal normalization concepts to a wide range of temporal data models.

The contributions of the previous chapter are extended in Chapter 30, which also points out the relevance of (temporally enhanced) database design to the conventional relational model. The chapter's contributions are two-fold. First, it introduces new concepts for capturing the temporal aspects of relation schemas, resulting in a comprehensive set of such concepts. Time patterns capture when the values of attributes change in the modeled reality and in the database. Lifespans describe when attributes have values. The temporal support and precision of attributes indicate which temporal aspects are relevant for the attributes and with which time granularity the aspects are to be recorded. And derivation functions describe how the values of an attribute for all times within its lifespan are computed from stored values. Second, the use of these concepts for database design is explored, resulting in new guidelines for the design of temporal relational and relational databases.

The ER model is seeing widespread use in practice for the task of conceptual database design. In brief, database schemas are initially designed using the ER model and are then translated to schemas in a data model, typically the relational model, supported by a database management system. Because the temporal aspects of database schemas are prevalent and difficult to model using the ER model, almost a dozen temporally enhanced ER models have been proposed. The remaining three chapters concern temporal ER modeling.

Chapter 31 offers a comprehensive survey of temporally enhanced ER models. It is examined how time-varying information is captured in these model, and the new concepts and modeling constructs introduced by the models are described. In addition, each model is characterized according to a total of 19 different design properties for temporally enhanced ER models.

The survey reveals that a new temporal ER model is needed that better satisfies the requirements of conceptual modeling involving temporal data, the topic of Chapter 32. With the existing ER models, an ontological foundation, and novel requirements as its basis, this chapter formally defines a graphical, temporally extended ER model. The ontological foundation serves to aid in ensuring a maximally orthogonal design, and the requirements aim, in part, at ensuring a design that naturally extends the syntax and semantics of the regular ER model. The result is a novel model that satisfies an array of properties not satisfied by any other single model.

Commercial database management systems support neither the ER model nor any temporal ER model as a model for data manipulation, but rather support various versions of the relational model for this purpose. Chapter 33 provides a two-step transformation of temporal ER diagrams to relational schemas. In the first step a temporal ER diagram is translated into relations in a surrogate-based relational model; in the second step, this database schema is translated into a conventional, lexically-based relational database schema.