Preface

For the past fifteen years, data warehousing and on line analysis (OLAP) technologies have shown their effectiveness in managing and analyzing a large amount of disparate data. Today, these technologies are confronted with new applications (mobile environments, data integration, data flow, personalization, stream data, sensor network applications, etc.) and new challenges as shown by the increasing success of the EII technologies (Enterprise Information Integration). These technologies are attracting a lot of attention from the database, data warehousing and data mining research communities.

Two main particularities of this book are: (i) it covers the whole process of designing and using a data warehouse, which includes requirement specification, conceptual design, logical design, physical design, tuning and evolution management and (ii) it shows the contribution of ontologies in designing and exploiting data warehouses.

The primary objective of this book is to give readers in-depth knowledge of data warehouses and their applications. The book is designed to cover a comprehensive range of steps for designing efficient and self-administrable data warehouses and managing their evolution. The fact that most of these steps are supported by some commercial DBMS motivates the need for such a book.

There was great response to the call for proposals, but due to the limited space only 15 chapters were accepted and selected. These chapters are authored by an outstanding roster of experts in their respective fields, and tackle various issues from different angles, requirements and interests. Their topics include user requirement specification, conceptual design, ontology driven ETL process, physical design and self tuning (materialized view selection, parallel processing, etc.), evolution and maintenance management, and security. The conceptual design concerns both classical and complex data sources (XML and spatial data).

The fifteen selected chapters cover the majority of steps executed within data warehouse projects. They can be classified into four sections: Conceptual Design and Ontology-based Integration, Physical Design and Self Tuning, Evolution and Maintenance Management, and Exploitation of Data Warehouse.

The four sections are summarized as follows.

Section 1: Conceptual Design and Ontology-Based Integration

This part contains six chapters summarized as follows:

From User Requirements to Conceptual Design in Data Warehouse Design – a Survey, by Matteo Golfarelli, gives a nice survey on conceptual design and user requirement analysis in the context of data warehouse environment and shows its importance in guaranteeing the success of business intelligence projects. It points out pros and cons of the different conceptual design techniques. Precise and simple examples are given to facilitate the understanding of existing conceptual models. This criticism may help
readers and designers to identify crucial choices and possible solutions more consciously when designing data warehouse projects. A particular attention is devoted to emphasizing the relationships between user requirement analysis and conceptual design and showing how they can be jointly and fruitfully used.

Data Extraction, Transformation and Integration guided by an Ontology, by Chantal Reynaud, Nathalie Pernelle, Marie-Christine Rousset, Brigitte Safar and Fatiha Saïs, focuses on the problem of integrating XML heterogeneous information sources into a data warehouse. This integration is guided by an ontology. It presents an approach supporting the acquisition of data from a set of external sources available for an application of interest including data extraction, data transformation and data integration or reconciliation. The proposed integration middleware extracts data from external XML sources which are relevant according to an RDFS+ ontology, transforms returned XML data into RDF facts conformed to the ontology and reconciles RDF data in order to resolve possible redundancies. This chapter is a great exercise for readers and designers to understand the whole steps of the ETL process: extraction, transformation and loading.

X-WACoDa: An XML-based approach for Warehousing and Analyzing Complex Data, by Hadj Mahboubi, Jean-Christian Ralaivao, Sabine Loudcher, Omar Boussaïd, Fadila Bentayeb and Jérôme Darmont, proposes a unified XML warehouse reference model that synthesizes and enhances related work, and fits into a global XML warehousing. This chapter is validated by a software platform that is based on this model, as well as a case study that illustrates its usage.

Designing Data Marts from XML and Relational Data Sources, by Yasser Hachaichi, Jamel Feki and Hanene Ben-Abdallah, presents a bottom-up/data-driven method for designing data marts from two types of sources: relational database and XML documents compliant to a given DTD. This method has three automatic steps: Data source pretreatment, relation classification, and data mart schema construction and one manual step for DM schema adaptation. The different steps of this method are illustrated using an e-ticket DTD used by an online broker and a relational database describing a hotel room booking system.

Ontology-based integration of heterogeneous, incomplete and imprecise data dedicated to a decision support system for food safety, by Patrice Buche, Sandrine Contenot, Lydie Soler, Juliette Dibie-Barthélemy, David Doussot, Liliana Ibanescu and Gaëlle Hignette, presents an application in the field of food safety using an ontology-based data integration approach. This chapter is a real application of the conceptual design part. This chapter motivates the use of ontology to resolve different conflicts found when integrating heterogeneous sources (structural and semantic). This chapter explores three ways to integrate data according to a domain ontology: (1) a semantic annotation process to extend local data with Web data which have been semantically annotated according to a domain ontology, (2) a flexible querying system to query uniformly both local data and Web data and (3) an ontology alignment process to find correspondences between data from two sources indexed by distinct ontologies.

On Modeling and Analysis of Multidimensional Geographic Databases, by Sandro Bimonte, presents a panorama of spatial OLAP (SOLAP) models and an analytical review of SOLAP tools. Spatial data warehouse does not get the same amount of attention by the community as the classical data warehouse. This chapter describes a Web-based system: GeWOlap. GeWOlap is an OLAP-GIS integrated solution implementing drill and cut spatio-multidimensional operators, and it supports some new spatio-multidimensional operators which change dynamically the structure of the spatial hypercube thanks to spatial analysis operators.
Section 2: Physical Design and Self Tuning

This section contains three chapters summarized as follows:

View Selection and Materialization, by Zohra Bellahsene, presents the problem of selecting materialized views to speed up decision support queries in a dynamic environment. It proposes a view selection method for deciding which views to materialize according to statistic metadata. Polynomial algorithms selecting views to materialize are given. This work is validated by a tool, called MATUN, build from the ground up to facilitate different view materialization strategies using the proposed algorithms. This tool can be used by users and data warehouse administrators to select materialized views.

ChunkSim: A Tool and Analysis of Performance and Availability Balancing, by Pedro Furtado, proposes ChunkSim, an event-based simulator for analysis of load and availability balancing in chunk-wise parallel data warehouses. This chapter discusses first how a shared nothing machine can store and process a data warehouse chunk-wise, and uses an efficient on demand processing approach. Then, it presents different parameters used by ChunkSim. Finally, it presents data allocation and replication alternatives that ChunkSim implements and the analysis that ChunkSim is currently able to run on performance and availability features. Intensive experimentations are presented and show the interest of the author’s work. This tool mainly contributes in self-tuning of physical parallel data warehouse. The main particularity of this chapter is that it issues a new research direction in data warehouse which is the development of simulators in order to facilitate the deployment of applications.

QoS-Oriented Grid-Enabled Data Warehouses, by Rogério Luís de Carvalho Costa and Pedro Furtado, presents QoS-oriented scheduling and distributed data placement strategies for a Grid-based warehouse. It discusses the use of a physically distributed database, in which tables are both partitioned and replicated across sites. The use of facts’ table partitioning and replication is particularly relevant as Grid users’ queries may follow geographical related access patterns. Inter-site dimension tables fragmentation and replication are done in order to achieve good performance in query execution but also to reduce data movement across sites, which is a costly operation in Grids.

Section 3: Evolution and Maintenance Management

This section contains two chapters summarized as follows:

Data Warehouse Maintenance, Evolution and Versioning, by Johann Eder and Karl Wiggisser, focuses on the problem of maintenance in the data warehouse domain, since, data warehouse systems are used in a changing environment thus the need for evolving systems is inevitable. This chapter provides illustrating examples that motivate the need for data warehouse maintenance. It also distinguishes between evolution and versioning problems. It presents some basic terms and definitions for the common understanding and introduces the different aspects of data warehouse maintenance. Several approaches addressing the problem are presented and classified by their capabilities.

Construction and Maintenance of Heterogeneous Data Warehouses, by Mohamed Badri, Faouzi Boufarès, Sana Hamdoun, Véronique Heiwy and Kazem Lellahi proposes a formal framework which deals with the problem of integrating of heterogeneous data sources from various categories: the structured data, the semi-structured data and unstructured data. This approach is based on the definition of an integration environment that is seen as a set of data sources associated with a set of “integration relationships” between the sources components. The originality of this work, contrary to various works
on integration, lies in the fact of covering the integration of all the categories of data considered at the same time and in the proposition of a theoretical approach of the data integration. The proposed approach is general and is applicable to any type of integration.

Section 4: Exploitation of Data Warehouse

This section contains four chapters summarized as follows:

*On Querying Data and Metadata in Multiversion Data Warehouse*, by Wojciech Leja, Robert Wrembel and Robert Ziembicki, presents the MVDWQL query language, for the multiversion data warehouse that allows: (1) to query multiple data warehouse versions that differ with respect to their schemas, (2) to augment query results with metadata describing changes made to the queried data warehouse versions, and (3) to explicitly query metadata on the history of data warehouse changes and visualize their results. Two types of queries on metadata are supported, namely: (1) queries searching for data warehouse versions that include an indicated data warehouse object and (2) queries retrieving the history of the evolution of an indicated data warehouse object. The MVDWQL have been successfully implemented in a multiversion data warehouse prototype system.

*Ontology Query Languages for Ontology-Based Databases: A Survey*, by Stéphane Jean, Yamine Aït Ameur and Guy Pierra presents a nice survey on ontology query languages developed for ontology based database (OBDB). First of all, this chapter gives a definition and several criteria of ontology and shows that all existing ontologies are not similar. Therefore, it gives three categories of ontologies: conceptual canonical ontologies, non conceptual canonical ontologies and linguistic ontologies that can be combined into a layered model, called the onion model. Based on this model, a general OBDB architecture that extends the traditional ANSI/SPARC database architecture is defined with a set of language requirements for its exploitation. Different ontology query languages are then analysed by studying their capabilities to fulfil this requirements.

*Ontology-Based Database Approach for Handling Preferences*, by Dilek Tapucu, Gayo Diallo, Yamine Aït Ameur and Murat Osman Ünalir proposes a solution handling preferences in ontology-based databases. It is an extension of the previous chapter. First, a formal and generic model to handle users’ preferences is defined. This proposed model is composed of several types of preferences usually addressed in the literature in a separate way. These preferences are independent of any logical model of data. This model is generic thanks to its ability to define a relationship with any ontology model. Then this chapter shows how this model can be integrated into the OntoDB OBDB architecture.

*Security in Data Warehouses*, by Edgar R. Weippl, presents an important issue in data warehouse which is security. It describes the traditional security models: mandatory access control (MAC), driven mainly by military requirements and role-based access control (RBAC) that is the commonly used access control model in commercial databases. Some issues on statistical databases are also given.