Chapter 11

Relational Techniques for Storing and Querying RDF Data: An Overview

Sherif Sakr
University of New South Wales, Australia

Ghazi Al-Naymat
University of New South Wales, Australia

ABSTRACT

The Resource Description Framework (RDF) is a flexible model for representing information about resources in the Web. With the increasing amount of RDF data which is becoming available, efficient and scalable management of RDF data has become a fundamental challenge to achieve the Semantic Web vision. The RDF model has attracted attentions in the database community and many researchers have proposed different solutions to store and query RDF data efficiently. This chapter focuses on using relational query processors to store and query RDF data. It gives an overview of the different approaches and classifies them according to their storage and query evaluation strategies.

INTRODUCTION

The Semantic Web term is coined by W3C founder Tim Berners-Lee in a Scientific American article describing the future of the Web (Berners-Lee et al., 2001). The main purpose of the Semantic Web vision is to provide a common framework for data-sharing across applications, enterprises, and communities. By giving data semantic meaning (through metadata), this framework allows machines to consume, understand, and reason about the structure and purpose of the data. The core of the Semantic Web is built on the Resource Description Framework (RDF) data model (Manola & Miller, 2004).

The RDF model is designed to have a simple data model, with a formal semantics and provable inference, with an extensible URI-based vocabulary that allows anyone to make statements about any

DOI: 10.4018/978-1-60960-475-2.ch011
Relational Techniques for Storing and Querying RDF Data

resource. Hence, in the RDF model, the universe is modeled as a set of resources where a resource is defined as anything that can have a universal resource identifier (URI). RDF describes a particular resource using a set of RDF statements of the form (subject, predicate, object) triples, also known as (subject, property, value). The subject is the resource, the predicate is the characteristic being described, and the object is the value for that characteristic.

Efficient and scalable management of RDF data is a fundamental challenge at the core of the Semantic Web. Several research efforts have been proposed to address these challenges (Abadi et al., 2009; Alexaki et al., 2001; Broekstra et al., 2002; Harth & Decker, 2005; Ma et al., 2004; Weiss et al., 2008). Relational database management systems (RDBMSs) have repeatedly shown that they are very efficient, scalable and effective in hosting types of data which have formerly not been anticipated to be stored inside relational databases such complex objects (TaÆurker & Gertz, 2001), spatio-temporal data (Botea et al., 2008) and XML data (Grust et al., 2004). RDBMSs derive much of their performance from sophisticated optimizer components which make use of physical properties that are specific to the relational model such as: sortedness, proper join ordering and powerful indexing mechanisms.

This chapter focuses on using relational query processors to store and query RDF data. We give an overview of the different approaches and classify them according to their storage and indexing strategy. The rest of the chapter is organized as follows. Section (RDF-SPARQL Preliminaries) introduces preliminaries of RDF data model and the W3C standard RDF query language, SPARQL. It also introduces the main alternative relational approaches for storing and querying RDF. Sections (Vertical (Triple) Stores, Property Table Stores, and Horizontal Stores) provide the details of the different techniques in each of the alternative relational approaches. Section (Experimental Evaluation) presents an experimental comparison between representatives of the different approaches. Finally, Section (Concluding Remarks) concludes the chapter and provides some suggestions for possible future research directions on the subject.

RDF-SPARQL PRELIMINARIES

The Resource Description Framework (RDF) is a W3C recommendation that has rapidly gained popularity a means of expressing and exchanging semantic metadata, i.e., data that specifies semantic information about data. RDF was originally designed for the representation and processing of metadata about remote information sources and defines a model for describing relationships among resources in terms of uniquely identified attributes and values. The basic building block in RDF is a simple tuple model, (subject, predicate, object), to express different types of knowledge in the form of fact statements. The interpretation of each statement is that subject S has property P with value O, where S and P are resource URIs and O is either a URI or a literal value. Thus, any object from one triple can play the role of a subject in another triple which amounts to chaining two labeled edges in a graph-based structure. Thus, RDF allows a form of reification in which any RDF statement itself can be the subject or object of a triple. One of the clear advantages of the RDF data model is its schema-free structure in comparison to the entity-relationship model where the entities, their attributes and relationships to other entities are strictly defined. RDF is not a syntax (i.e. data format). There exist various RDF syntaxes (e.g. Notation 3 (N3) language, Turtle, XML) and depending on the application space one syntax may be more appropriate than another. In RDF, the schema may evolve over the time which fits well with the modern notion of data management, dataspaces, and its pay-as-you-go philosophy (Jeffery et al., 2008). Figure 1 illustrates a sample RDF graph.