Abstract

In recent years, the popularity of XML has increased significantly. XML is the extensible markup language of the World Wide Web Consortium (W3C). XML is used to represent data in many areas, such as traditional database management systems, e-business environments, and the World Wide Web. XML data, unlike relational and object-oriented data, has no *fixed schema known in advance* and is stored separately from the data. XML data is self-describing and can model heterogeneity more naturally than relational or object-oriented data models. Moreover, XML data usually has XLinks or XPointers to data in other documents (e.g., global-links). In addition to XLink or XPointer links, the XML standard allows to add internal-links between different elements in the same XML document using the ID/IDREF attributes.

The rise in popularity of XML has generated much interest in query processing over graph-structured data. In order to facilitate efficient evaluation of path expressions, structured indexes have been proposed. However, most variants of structured indexes ignore global- or interior-document references. They assume a tree-like structure of XML-documents, which do not contain such global-and internal-links. Extending these indexes to work with large XML graphs considering of global- or internal-document links, firstly requires a lot of computing power for the creation process. Secondly, this would also require a great deal of space in which to store the indexes. As a latter demonstrates, the efficient evaluation of ancestors-descendants queries over arbitrary graphs with long paths is indeed a complex issue.

This thesis proposes the HID index (2-Hop cover path Index based on DAG) is based on the concept of a two-hop cover for a directed graph. The algorithms proposed for the HID index creation, in effect, scales down the original graph size substantially. As a result, a directed acyclic graph (DAG) with a smaller number of nodes and edges will emerge. This reduces the number of computing steps required for building the index. In addition to this, computing time and space will be reduced as well. The index also permits to efficiently evaluate ancestors-descendants relationships. Moreover, the proposed index has an advantage over other comparable indexes: it is optimized for descendants- or-self queries on arbitrary graphs with link relationship, a task that would stress any index structures. Our experiments with real life XML data show that, the HID index provides better performance than other indexes.