XQuery Language

Erick Gonzalez Parada ID: 178145
Emiliano Ruiz Plancarte ID: 177478
Andre Francois Duhamel Gutierrez ID: 177315
Antonio Gutiérrez Blanco ID: 177442

March 15, 2025

Abstract

This document explores the fundamentals of the XQuery language, showcasing query examples, methodologies, and conclusions drawn from practical applications.

Keywords: XML, DTD, XPath, XQuery, BaseX.

1 Theoretical Framework

XQuery is a powerful and flexible query language designed for querying and transforming XML data. It is often referred to as "SQL for XML" due to its ability to extract and manipulate data stored in XML documents. XQuery is built on XPath expressions and provides additional features such as FLWOR (For, Let, Where, Order by, Return) expressions, which allow for complex data retrieval and transformation tasks [**?**].

One of the key strengths of XQuery is its ability to handle hierarchical and nested data structures, which are common in XML documents. This makes it particularly useful for applications such as web services, data integration, and content management systems. For example, XQuery can be used to extract specific elements from an XML document, transform the data into a different format, or generate reports based on the data [**?**].

Another important feature of XQuery is its support for strong typing and schema validation. This ensures that the data being queried adheres to a predefined structure, reducing the risk of errors and improving the reliability of the queries. Additionally, XQuery supports modularity, allowing developers to create reusable modules and functions that can be shared across different projects [**?**].

Goals

The goal of this lab is to have a first encounter with the XQuery language to examine XML documents.

Materials

- *BaseX*

2 Methodology

The methodology involves writing and executing XQuery scripts to perform the operations on the XML documents. Each query is designed to demonstrate a specific feature or use case of XQuery, such as filtering, sorting, grouping, and transforming data. The results of each query are analyzed to understand the underlying principles and techniques.

3 Query Results

Below are the XQuery scripts and their corresponding results:

1. List books published by ”Addison-Wesley” after 1991:

```
    let $books := doc("bookshop.xml")//book
    where $books/publisher = "Addison-Wesley" and $books/@year > 1991
    return
        <book><year>{$books/@year}</year><title>{$books/title/text()}</title></book>
```

2. Create a flattened list of ”result” items with title and author:

```
    for $book in doc("bookshop.xml")//book,
        $author in $book/author
    return
        <result><title>{$book/title/text()}</title><author>{$author/name/text()}
        {$author/lastname/text()}</author></result>
```

3. List titles with grouped authors:

```
    for $book in doc("bookshop.xml")//book
    return <result>
        <title>{$book/title/text()}</title>
        <authors>{
            for $author in $book/author
            return <author>{$author/name/text()}
                {$author/lastname/text()}</author>
        }</authors>
    </result>
```

4. List authors with their books:

```
    for $author in distinct-values(doc("bookshop.xml")//author)
    let $books := doc("bookshop.xml")//book[author/name = $author/name and
        author/lastname = $author/lastname]
    return <result>
        <author>{$author/name/text()} {$author/lastname/text()}</author>
        <titles>{
            for $book in $books
            return <title>{$book/title/text()}</title>
        }</titles>
    </result>
```

5. Books with number of authors:

```
    for $book in doc("bookshop.xml")//book
    return <book>
        <title>{$book/title/text()}</title>
        {if ($book/author) then
            <number-of-authors>{count($book/author)}</number-of-authors> else
            ()}
    </book>
```

6. Find minimum price for each book in prices.xml:

```
    for $title in distinct-values(doc("prices.xml")//book/title)
    let $minPrice := min(doc("prices.xml")//book[title = $title]/price)
    return <minimum-price title="{$title}">{$minPrice}</minimum-price>
```

7. Alphabetically list Addison-Wesley books after 1991:

```
    for $book in doc("bookshop.xml")//book
    where $book/publisher = "Addison-Wesley" and $book/@year > 1991
    order by $book/title
    return <book>
        <title>{$book/title/text()}</title>
        <year>{$book/@year}</year>
    </book>
```

8. Return book element for authored books, reference for published ones:

```
    for $book in doc("bookshop.xml")//book
    return
        if ($book/author)
        then <book>
            <title>{$book/title/text()}</title>
            <authors>{
                for $author in $book/author
                return <author>{$author/name/text()}
                    {$author/lastname/text()}</author>
            }</authors>
        </book>
        else if ($book/editor)
        then <reference>
            <title>{$book/title/text()}</title>
            <affiliation>{$book/editor/affiliation/text()}</affiliation>
        </reference>
        else ()
```

9. List book titles with their prices at each bookshop:

```
    for $book in distinct-values(doc("prices.xml")//book/title)
    return <book>
        <title>{$book/text()}</title>
        {
            for $store in doc("prices.xml")//book[title = $book]
            return <price
                source="{$store/source/text()}">{$store/price/text()}</price>
        }
    </book>
```

4 Conclusions

Through this exercise, we gained a deeper understanding of the XQuery language and its capabilities in querying and transforming XML data. The practical examples demonstrated the flexibility and power of XQuery in handling complex data retrieval and manipulation tasks. Future work could explore advanced features such as XQuery functions, modules, and integration with other technologies, will ORACLE save us from a painfull dev experiencie with XML?

References

[1] XQuery 3.1: An XML query language. (n.d.). Www.w3.org. Retrieved March 15, 2025, from https://www.w3.org/TR/xquery-31/

[2] XQuery language reference (SQL Server). (n.d.). Microsoft.com. Retrieved March 15, 2025, from https://learn.microsoft.com/en-us/sql/xquery/xquery-language-reference-sql-server?view=sql-server-ver16

[3] XQuery tutorial. (n.d.). W3schools.com. Retrieved March 15, 2025, from https://www.w3schools.com/xml/xquery_intro.asp