Prof. Dr. Stefan Brass
Institut für Informatik
MLU Halle-Wittenberg

29. November 2022

XML und Datenbanken
— 7. Übungsblatt: XDM, DOM, SAX —

Allgemeine Aufgabe

Notieren Sie sich eventuelle Verständnisfragen, so dass wir diese im nächsten Online-Treffen klären können. Da es erfahrungsgemäß sehr sill ist: Nehmen Sie sich bitte die Zeit und denken Sie bewusst über mögliche Fragen nach (wenigstens eine). Ich kann auch Studierende drannehmen, die sich nicht gemeldet haben. Ein Mal würde ich wohl verstehen, dass der Stoff so einfach war, dass Sie einfach keine Fragen haben. Wenn sich das wiederholt, müsste ich dann aber umgekehrt prüfungsähnliche Fragen stellen.

Hausaufgabe

Geben Sie die folgenden Aufgaben bis Montag, 05.12.2022, 16^{00}, über die Übungsplattform in StudIP ab. Schreiben Sie die Lösung für a) in eine `.txt`-Datei und die für b) in eine `.java`- oder `.cpp`/`.cxx` oder `.py`-Datei. Es gibt 2 Punkte für Teil a), und 8 Punkte für Teil b).

Die Abgaben nur stichprobenartig kontrolliert. Wenn Ihre Abgabe nicht kontrolliert wurde, bekommen Sie die volle Punktzahl. Wenn Sie später wegen Plagiaten auffallen, oder bei einer Stichprobe eine fast gar nicht gelöste Aufgabe entdeckt wird, können auch alte Abgaben kontrolliert werden. Dann können auch rückwirkend Punkte abgezogen werden.

Sie benötigen 67% der Hausaufgabenpunkte und eine aktive Mitarbeit in den Übungen für die Studienleistung.

Die „Wiederholungsaufgaben“, also Teil c) und d), sind nicht abzugeben. Beschäftigen Sie sich aber bitte auch mit diesen Aufgaben. Sie müssen damit rechnen, dass Sie beim Online-Treffen gebeten werden, einen Teil des Vorlesungs-Stoffes zu wiederholen und insbesondere eine der Fragen aus c) zu beantworten.

Zur Bearbeitung der Hausaufgaben müssen Sie sich selbständig über die SAX- und DOM-Schnittstellen informieren. Über diese Schnittstellen kann man XML-Daten in Programmen verarbeiten (falls entsprechende Bibliotheken zur Verfügung stehen, in Java gibt es das jedenfalls). Informieren Sie sich bitte selbständig über dieses Thema. Z.B. könnten Sie das folgende Tutorial zu DOM lesen:

[https://docs.oracle.com/javase/tutorial/jaxp/dom/index.html]

Weitere Tutorials sind:

- [https://www.tutorialspoint.com/java_xml/java_dom_parser.htm]
- [https://www.mkyong.com/java/how-to-read-xml-file-in-java-dom-parser/]

Die DOM-Spezifikation wurde vom W3C entwickelt:

- [https://www.w3.org/TR/DOM-Level-3-Core/]
- [https://www.w3.org/DOM/]

DOM wird auch benutzt, um HTML in JavaScript-Programmen zu manipulieren. Für XML ist aber nur der "DOM Core" relevant.

Eine andere Schnittstelle zur Verarbeitung von XML in Java ist die SAX-Schnittstelle. Sie ist event-basiert, bei ihr wird die Baumstruktur nicht im Speicher materialisiert, sondern es werden Methoden z.B. beim Lesen von Start- und End-Tags aufgerufen. Tutorials sind:

- [https://docs.oracle.com/javase/tutorial/jaxp/sax/parsing.html]
- [https://mkyong.com/java/how-to-read-xml-file-in-java-sax-parser/]

a) Schauen Sie sich folgendes Beispiel-Programm für die SAX-Schnittstelle an:

   [http://users.informatik.uni-halle.de/~brass/xml22/SaxExample.java]

   Compilieren Sie das Programm mit „`javac SaxExample.java`“ und probieren Sie es dann mit verschiedenen Eingabedateien aus:

   `java SaxExample ⟨Datei⟩.xml`

   Beschreiben Sie kurz mit eigenen Worten (ggf. anhand einer Beispiel-Ausgabe), was das Programm tut.

b) Schreiben Sie ein kleines Java-Programm, das die SIDs aller Studenten druckt, die 10 Punkte in Hausaufgabe 1 haben. Diese Aufgabe könnte natürlich auch mit einer ganz kurzen XPath-Anfrage gelöst werden, aber es ist nur ein Beispiel für die Verarbeitung von XML-formatierten Daten in Java (oder bei Bedarf anderen Programmiersprachen).

   Sie dürfen wählen, ob Sie die DOM- oder die SAX-Schnittstelle verwenden. Für die SAX-Schnittstelle dürfen Sie das Beispielprogramm passend modifizieren.

   Sie dürfen auch wählen, welche Codierung der Daten in XML Sie bevorzugen:

   - Die XML-Datei mit den Daten in Attributen:

     [http://users.informatik.uni-halle.de/~brass/xml22/examples/ex1.xml]

```
<?xml version='1.0' encoding='ISO-8859-1'?>
<!-- Grades Database with Data in Attributes -->
<GRADES-DB>
<STUDENT SID='101' FIRST='Ann' LAST="Smith"
        EMAIL='smith@acm.org'/>
    <STUDENT SID='102' FIRST='Michael' LAST='Jones'/>
    ...
    <EXERCISE CAT='H' ENO='1' TOPIC='Relational Algebra' MAXPT='10'/>
    ...
    <RESULT SID='101' CAT='H' ENO='1' POINTS='10'/>
    <RESULT SID='101' CAT='H' ENO='2' POINTS= '8'/>
    ...
</GRADES-DB>
```

- Oder die Version mit den Daten in geschachtelten Elementen:

  [http://users.informatik.uni-halle.de/~brass/xml22/examples/ex2.xml]

```
<?xml version='1.0' encoding='ISO-8859-1'?>
<!-- Grades Database with Data in Elements -->
<GRADES-DB>
    <STUDENTS>
        <STUDENT>
            <SID>101</SID>
            <FIRST>Ann</FIRST>
            <LAST>Smith</LAST>
            <EMAIL>smith@acm.org</EMAIL>
        </STUDENT>
        ...
    </STUDENTS>
    <EXERCISES>
        <EXERCISE>
            <CAT>H</CAT>
            ...
        </EXERCISE>
        ...
    </EXERCISES>
    <RESULTS>
        <RESULT>
            <SID>101</SID>
            <CAT>H</CAT>
            <ENO>1</ENO>
            <POINTS>10</POINTS>
        </RESULT>
        ...
    </RESULTS>
</GRADES-DB>
```

Wiederholungsaufgaben

Beschäftigen Sie sich mit diesen Aufgaben. Sie brauchen aber nichts abzugeben.

c) Was würden Sie in einer mündlichen Prüfung auf folgende Fragen zum XPath/XQuery Data Model (XDM) antworten?

- Die wesentliche Datenstruktur, mit der XPath und XQuery arbeiten, ist eine Sequenz. Wie ist das definiert?
- Was sind die sieben Arten von Knoten? Welche Knoten können Kind-Knoten sein, welche Eltern-Knoten?
- Warum braucht man einen extra Dokument-Knoten und verwendet nicht einfach einen Wurzel-Element-Knoten?
- Nennen Sie einige syntaktische Details, die beim Einlesen/Parsen einer XML Datei in die interne XDM-Repräsentation verloren gehen.
- Warum gibt es noch den XML InfoSet Standard, und nicht nur den XDM Standard?
- Welche Unterschiede bestehen zwischen einer XDM-Baumstruktur, die ohne Validierung aufgebaut wird, und einer, die aus einem "Post-Validation InfoSet" aufgebaut wird?
- Beschreiben Sie die Behandlung von Leerplatz zwischen Elementknoten.
- Warum sollen Namenspace-Knoten nicht mehr verwendet werden? Was ist das Problem?
- Was ist die "Document Order"?
- Welche Beziehung besteht zwischen
  - "`anyType`",
  - "`anySimpleType`",
  - "`anyAtomicType`" und
  - "`untypedAtomic`"?

  Erklären Sie die Typ-Hierarchie und nennen Sie jeweils ein Beispiel für einen Typ, der noch zum jeweiligen Obertyp gehört, aber nicht zum Untertyp.
- Was ist der String-Value von Element-Knoten? Und der typed value, wenn das Dokument validiert wurde? Hier müssen Sie Elemente mit einfachem Inhalt, Elemente mit reinem Element-Inhalt und Elemente mit "mixed content model" unterscheiden.

d) Schauen Sie sich die folgenden Dateien im Browser an (aufgrund von Sicherheitseinschränkungen ist die Ausgable leer, wenn sie lokal gespeicherte Kopien anschauen):

- [http://users.informatik.uni-halle.de/~brass/xml22/xsl/ex1_query.xml]
- [http://users.informatik.uni-halle.de/~brass/xml22/xsl/ex2_query.xml]

Das verwendete Stylesheet finden Sie hier:

- [http://users.informatik.uni-halle.de/~brass/xml22/xsl/query.xsl]

Das Stylesheet zeigt informationen über die XDM Knoten an, die beim Parsen der XML Dateien erzeugt werden. Wenn Sie die XML-Dateien in einem Browser öffnen, sollte Ihnen eine HTML Liste der Knoten angezeigt werden, und zu jedem Knoten der Typ, der Name und der "String Value". Vergleichen Sie dies mit dem XML Code in den Dateien. Die originalen XML-Dateien (vor Anwendung des Stylesheets) können Sie sich mit der Funktion „More Tools"/„Page Source" (`Ctrl+U`) anschauen. Alternativ gibt es die XML-Dateien auch ohne Link zum Stylesheet:

- [http://users.informatik.uni-halle.de/~brass/xml22/examples/ex1.xml]
- [http://users.informatik.uni-halle.de/~brass/xml22/examples/ex2.xml]

Diese Dateien sind oben auch abgedruckt.

Für Interessierte

e) Sie können die Daten der DBLP Bibliographie

[https://dblp.org/]

als XML-Datei herunterladen:

[https://dblp.org/xml/]

Diese Seite enthält eine ziemlich vollständige Liste der meisten anständig veröffentlichten Artikel über Informatik. Ein Artikel dazu finden Sie hier:

[https://dblp.org/xml/docu/dblpxml.pdf]

Ich habe die Daten für einen Benchmark für deduktive Datenbanken verwendet (eine neue Version einer Aufgabe aus dem OpenRuleBench). Die benötigten Daten habe ich mit einem Java-Programm mit SAX-Schnittstelle in die Eingabesprache der deduktiven Datenbank übersetzt.