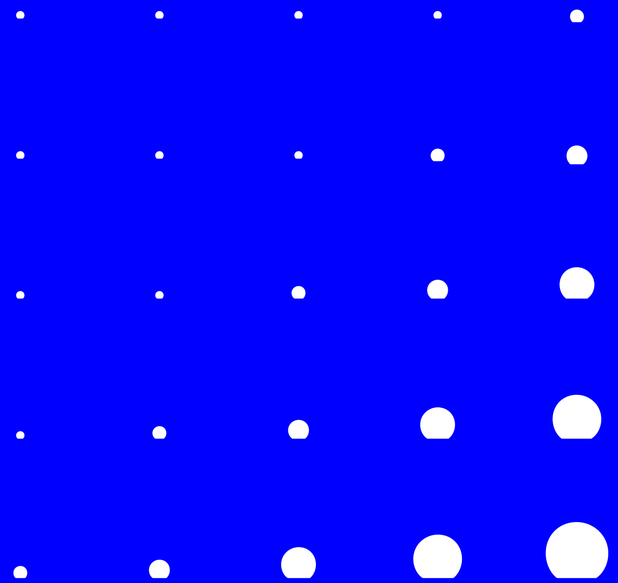
Technische Hochschule
mannheim

XML-Parser: DOM & SAX

Technische Hochschule Mannheim | Prof. Dr. Frank Dopatka



Was ist ein XML-Parser?

Technische Hochschule mannheim

- XML-Parser können anhand zweier Kriterien unterschieden werden:
- Sie sind validierend oder nichtvalidierend.
  - Nichtvalidierende Parser kontrollieren lediglich, ob das Dokument wohlgeformt ist, also ob es den Spezifikationen des W3C entspricht.
  - Validierende Parser prüfen zusätzlich die Konformität gegenüber einer DTD oder einem XML Schema.
- Sie greifen auf den XML-Baum zu, also auf dessen Document Object Model (DOM) oder
  sequentiell auf einen XML-Datenstrom durch Verwendung der Simple API for XML (SAX).

Was ist ein DOM-Parser?

Technische Hochschule mannheim

- DOM ist ein Standard des W3C.
- Beim DOM-Parsen wird das XML-Dokument zunächst in einer Baum-Struktur angelegt, dem DOM-Baum.
- Das Root-Element des XML-Dokuments fungiert als Wurzel für den DOM-Baum.
- Dieser Baum steht dann als Abbild des XML-Dokuments im Speicher zur Verfügung und kann beliebig navigiert werden.
- Jede HTML5-Datei ist auch ein DOM-Baum.

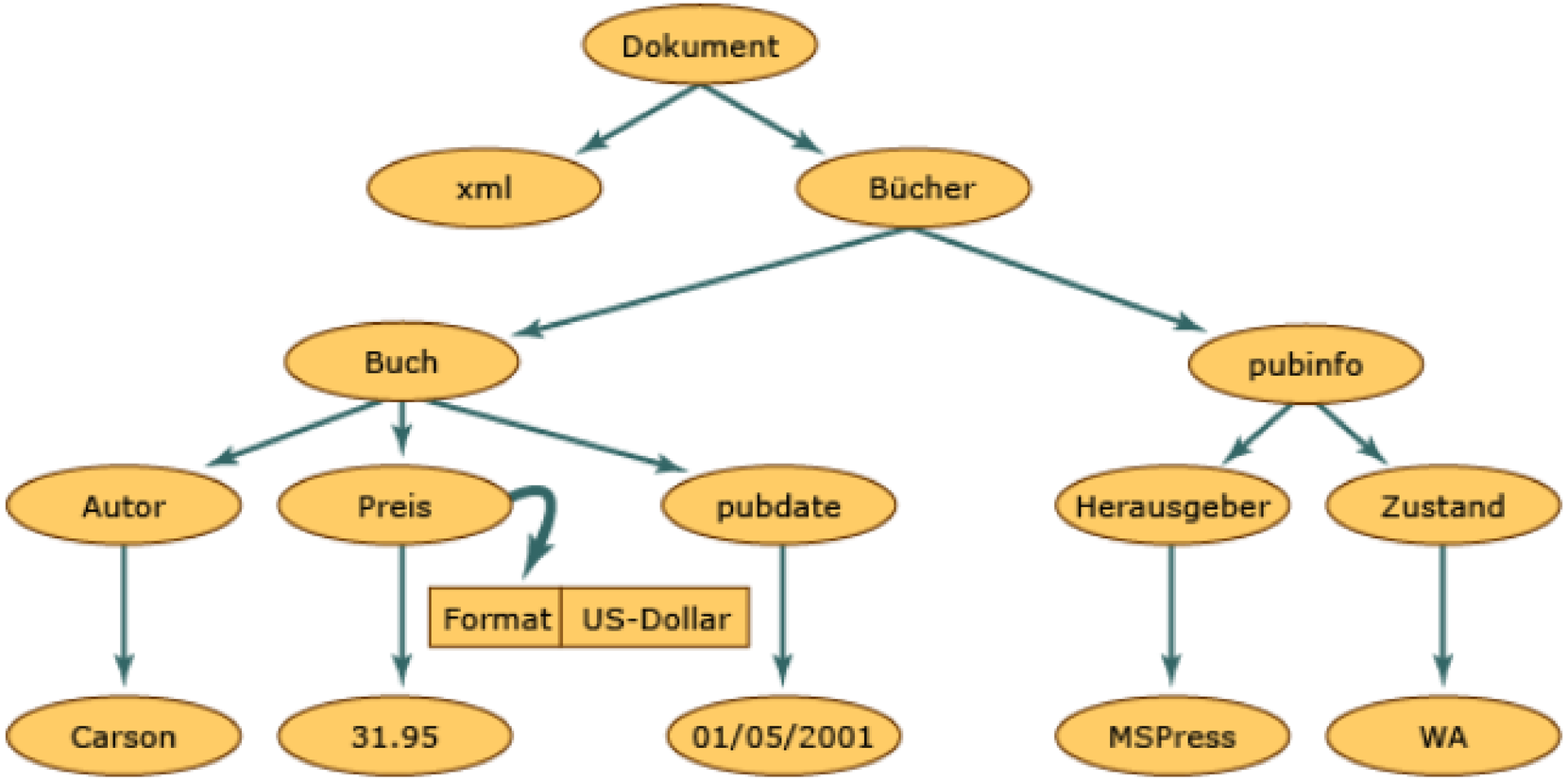
Ein kleines XML-Dokument...

Technische Hochschule mannheim

```xml
<?xml version="1.0"?>
  <books>
    <book>
        <author>Carson</author>
        <price format="dollar">31.95</price>
        <pubdate>05/01/2001</pubdate>
    </book>
    <pubinfo>
        <publisher>MSPress</publisher>
        <state>WA</state>
    </pubinfo>
  </books>
```

Das XML-Dokument und dessen DOM

Technische Hochschule mannheim



Was ist ein DOM-Parser?

Technische Hochschule mannheim

- Eine DOM-API bietet Werkzeuge zum Zugriff auf den Baum,
  und zum Verändern des Baums, und ist deshalb ideal für interaktive Anwendungen geeignet.
- Das gesamte Object Model steht die ganze Zeit im Speicher zur Verfügung.
- Diese DOM-Struktur ist in der Regel objektorientiert und damit sehr speicherlastig.

Ein weiteres XML-Dokument...

Technische Hochschule mannheim

```xml
<?xml version="1.0"?>
<company>
    <staff id="1001">
        <firstname>yong</firstname>
        <lastname>mook kim</lastname>
        <nickname>mkyong</nickname>
        <salary>100000</salary>
    </staff>
    <staff id="2001">
        <firstname>low</firstname>
        <lastname>yin fong</lastname>
        <nickname>fong fong</nickname>
        <salary>200000</salary>
    </staff>
</company>
```

Ein weiteres XML-Dokument und dessen Java-Parser…

Technische Hochschule mannheim

```
import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.parsers.DocumentBuilder;
import org.w3c.dom.Document;
import org.w3c.dom.NodeList;
import org.w3c.dom.Node;
import org.w3c.dom.Element;
import java.io.File;

public class ReadXMLFile {
  public static void main(String argv[]) throws Exception {
    File fXmlFile = new File("/Users/mkyong/staff.xml");
    DocumentBuilderFactory dbFactory = DocumentBuilderFactory.newInstance();
    DocumentBuilder dBuilder = dbFactory.newDocumentBuilder();
    Document doc = dBuilder.parse(fXmlFile);
    doc.getDocumentElement().normalize();
```

Ein weiteres XML-Dokument und dessen Java-Parser…

Technische Hochschule mannheim

```
System.out.println("Root element:" + doc.getDocumentElement().getNodeName());
NodeList nList = doc.getElementsByTagName("staff");
System.out.println("----------------------------");
for (int temp = 0; temp < nList.getLength(); temp++) {
  Node nNode = nList.item(temp);
  System.out.println("\nCurrent Element :" + nNode.getNodeName());
  if (nNode.getNodeType() == Node.ELEMENT_NODE) {
    Element ele = (Element) nNode;
    System.out.println("Staff id : "
            + ele.getAttribute("id"));
    System.out.println("First Name : "
            + ele.getElementsByTagName("firstname").item(0).getTextContent());
    System.out.println("Last Name : "
            + ele.getElementsByTagName("lastname").item(0).getTextContent());
    System.out.println("Nick Name : "
            + ele.getElementsByTagName("nickname").item(0).getTextContent());
    System.out.println("Salary : " +
            ele.getElementsByTagName("salary").item(0).getTextContent());
  }
}
```

Ausgabe des DOM-Parsers

Technische Hochschule mannheim

```
Root element :company
----------------------------

Current Element :staff
Staff id : 1001
First Name : yong
Last Name : mook kim
Nick Name : mkyong
Salary : 100000

Current Element :staff
Staff id : 2001
First Name : low
Last Name : yin fong
Nick Name : fong fong
Salary : 200000
```

Was ist SAX?

Technische Hochschule mannheim

- SAX wurde ursprünglich in Java entwickelt und besteht aus einer Anzahl von Java-Interfaces; heute finden sich jedoch Implementierungen für nahezu alle gängigen Programmiersprachen.
- SAX unterliegt keinem formalen Konsortium, dennoch ist es ein De-facto-Standard.
- Es spezifiziert eine Menge von Methoden für den Zugriff auf XML-Dokumente mittels eines SAX-Parsers.
  - SAX arbeitet, anders als DOM, ereignisorientiert.
  - Das Verarbeitungsprinzip entspricht dem Konzept einer Pipeline.
- SAX definiert eine Menge von Ereignissen, die beim sequentiellen Lesen eines XML-Dokuments vorkommen können.
- Diese Ereignisse sind zustandslos, sie referenzieren nicht auf andere, vorhergegangene Ereignisse und stehen auch sonst in keinem Verhältnis zu anderen Ereignissen.

Was ist SAX?

Technische Hochschule mannheim

- Beim Erkennen einer syntaktischen Struktur startet der SAX-Parser eine Behandlungsroutine, welche gegebenenfalls eine individuelle Behandlungsroutine des Ereignisses ausführt.
- Hierdurch kann mit dem Einlesen der ersten Zeichen bereits mit der Auswertung des Dokuments begonnen werden.
- Dies verkürzt insbesondere in interaktiven Systemen die subjektiv gefühlte Zugriffszeit.
- Gleichzeitig minimiert der SAX-Parser den Speicherbedarf, da neben dem jeweils eingelesenen Element nur solche Daten im Speicher stehen, die mittels einer Behandlungsroutine explizit ausgewählt wurden.

Ein beispielhaftes XML-Dokument...

Technische Hochschule mannheim

```xml
<?xml version="1.0"?>
<seminararbeit>
 <titel>DOM, SAX und SOAP</titel>
 <inhalt>
  <kapitel value="1">Einleitung</kapitel>
  <kapitel value="2">Hauptteil</kapitel>
  <kapitel value="3">Fazit</kapitel>
 </inhalt>
</seminararbeit>
```

Ein beispielhaftes XML-Dokument... löst folgende SAX-Ereignisse aus

Technische Hochschule mannheim

- startDocument()
  Parser ist auf Anfang des XML-Dokuments gestoßen.
- startElement("seminararbeit",[])
  Ein Element mit dem Namen "seminararbeit" wurde gefunden;
  der 2. Parameter ist eine Liste aller zum Element gehörenden Attribute.
  Da dieses Element allerdings keine Attribute hat, ist diese Liste in diesem Fall leer.
- startElement("titel",[])
- characters("DOM, SAX und SOAP")
  der Inhalt des Elements "titel“
- endElement("titel")
  Kennzeichnet, dass das Ende des zuvor gefundenen Elements erreicht wurde.
- ...