XML PARSERI

Novi Sad, 2017

Miroslav Zarić

ŠTA JE XML PARSIRANJE

- Podrazumeva prolazak kroz XML dokument sa ciljem:
  - pristupa određenim (ili svim) podacima u njemu
  - modifikacije podataka (ne dozvoljavaju svi parseri)
- XML parser - program koji putem odgovarajućih (poželjno jednostavnih) funkcija omogućava laku manipulaciju sadržajima u XML dokumentima

VRSTE PARSERA

- **SAX Parser** - obrađuje dokument na osnovu događaja - za određeni tip događaja piše se odgovarajući *handler*. Ne učitava odjednom ceo dokument u memoriju (ne pravi njegovu memorijsku predstavu).
- **DOM Parser** - obrada XML dokumenta se obavlja tako što se ceo dokument učita u memoriju - formiranjem strukture hijerarhijskog stabla. Sva manipulacija se zatim obavlja nad ovom memorijskom reprezentacijom dokumenta.

VRSTE PARSERA

- **StAX Parser** - Koristi slične principe kao SAX, bolje optimizovan.
- **XPath Parser** - Obradu dokumenta vrši na osnovu odgovarajućih izraza - koristi se intenzivno u kombinaciji sa XSLT.

SAX PARSER

- **SAX Parser** - obrađuje dokument na osnovu događaja - za određeni tip događaja piše se odgovarajući *handler*. Ne učitava odjednom ceo dokument u memoriju (ne pravi njegovu memorijsku predstavu).
- **Osnovne karakteristike:**
  - Stream orijentisan interfejs za obradu XML-a. Aplikacija koja koristi ovaj tip parsera dobija notifikaciju svaki put kada se procesira element, atribut…
- Čita dokument počevši od root elementa i generiše događaje svaki put kada prepozna neki "token" u dobro formiranom dokumentu.
  - obrada se vrši isključivo po redosledu pojavljivanja u dokumentu
  - obaveštava aplikaciju o tipu tokena na koji je trenutno naišao
  - da bi obezbedila smislenu obradu aplikacija mora parseru registrovati svoj *event handler*
  - kada se određeni token detektuje, trigeruje se odgovarajući registrovani *event handler* za dati tip tokena

SAX PARSER

- **ContentHandler interfejs** (specificira *callback* metode koje parser koristi da notifikuje aplikaciju o tipu sadržaja na koji je naišao)**:**
  - void **startDocument()**
  - void **endDocument()**
  - void **startElement(String uri, String localName, String qName, Attributes atts)**
  - void **endElement(String uri, String localName,String qName)**
  - void **characters(char[] ch, int start, int length)**
  - void **ignorableWhitespace(char[] ch, int start, int length)**
  - void **processingInstruction(String target, String data)**
  - void **setDocumentLocator(Locator locator))** - (postavlja lokator kojim je moguće pratiti poziciju u dokumentu)
  - void **skippedEntity(String name)**
  - void **startPrefixMapping(String prefix, String uri)**
  - void **endPrefixMapping(String prefix)**

SAX PARSER

- **Kada ga je pogodno koristiti:**
  - Kada je moguće obradu vršiti linearno - redom kojim se elementi pojavljuju od vrha ka dnu dokumenta
  - Dokument nije "duboko ugnježden"
  - Kada se procesira veliki dokument (ili veliki broj dokumenata u paraleli) - u tom slučaju bi formiranje modela dokumenta bilo resusrsno prezahtevno (DOM implementacije koriste i do 10 bajta memorije za 1 bajt XML podataka)
  - Kada problem koji se rešava zahteva korišćenje samo dela dokumenta
  - Podaci su dostupni čim ih parser "prepozna" - idealno za obradu podataka koji se "streamuju"

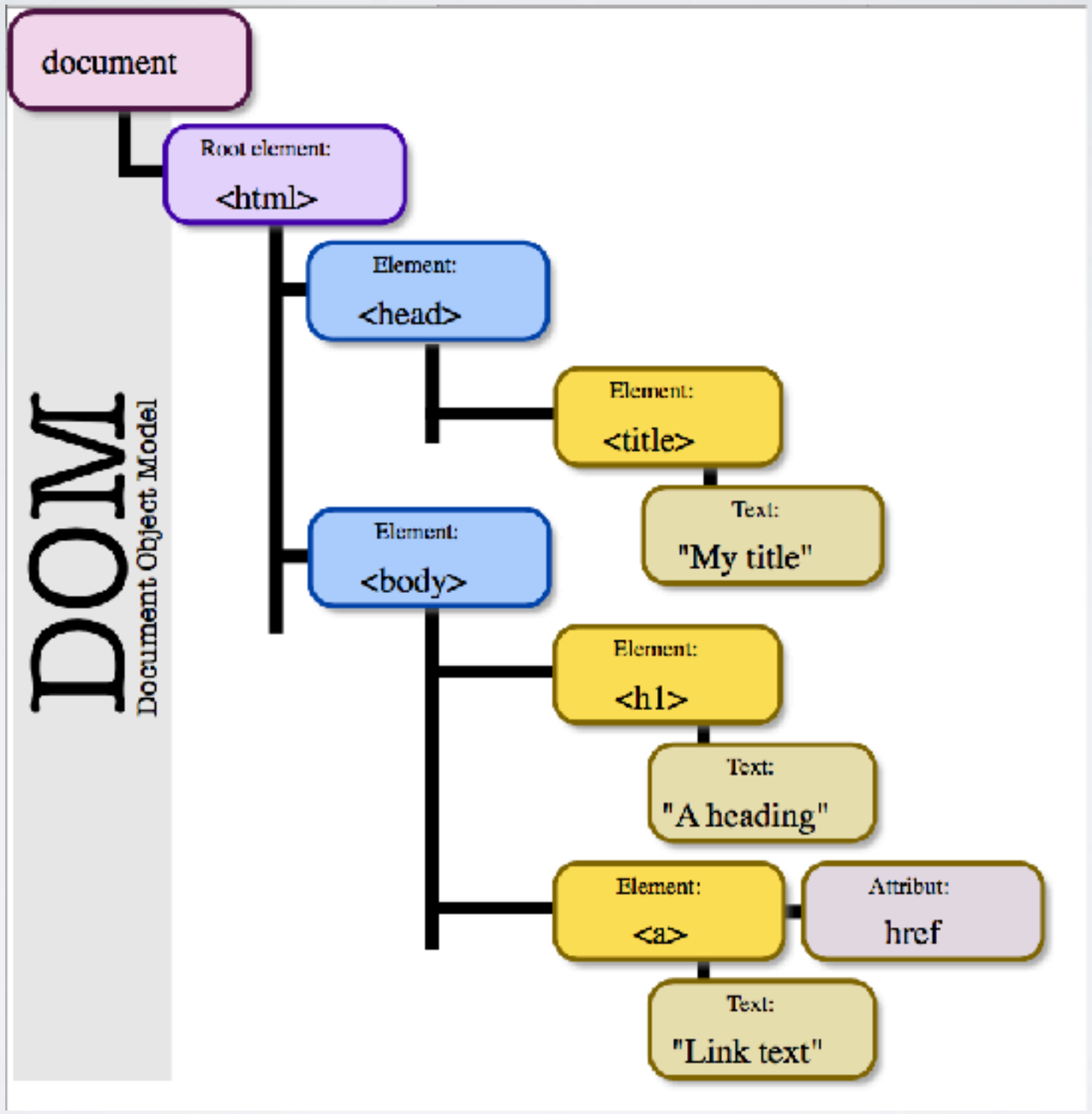
SAX PARSER

- **Nedostaci:**
  - Nije moguć direktni pristup određenom elementu u strukturi (jer se sve procesira linearno)
  - Ukoliko je potrebno imati naknadni pristup podacima koje je parser već "prošao" neophodno je pisati dodatni sopstveni kod koji bi takve podatke negde privremeno čuvao

DOM

- **Document Object Model (DOM)** - oficijelna preporuka modela XML dokumenta World Wide Web Consortium-a (W3C).
  - Definiše interfejse koji aplikacijama omogućavaju pristup i manipulaciju sadržaja XML dokumenata. Parseri koji implementiraju DOM implementiraju sledeće interfejse:
    - **Node** - osnovni tip u DOM.
    - **Element** - predstavlja elemente XML-a.
    - **Attr** - reprezentuje atribute elementa XML-a.
    - **Text** - predstavlja tekstualni sadržaj elementa ili tributa.
    - **Document** - predstavlja ceo dokument.

DOM


document
Root element:
<html>
Element:
<head>
Element:
<title>
Text:
"My title"
Element:
<body>
Element:
<h1>
Text:
"A heading"
Element:
<a>
Attribut:
href
Text:
"Link text"
DOM
Document Object Model

DOM PARSERI

- **Osnovne karakteristike:**
  - Po završenom parsiranju aplikaciji je na raspolaganju objektna reprezentacija sadržaja dokumenta formirana kao hijerarhijsko stablo (DOM).
  - DOM obezbeđuje veliki broj funkcija za pristup i manipulaciju nad DOM strukturom i sadržajem.

DOM PARSERI

- **Osnovna prednost:** DOM je de facto standardni interfejs za manipulaciju strukturom XML dokumenata. Kod napisan da koristi jedan DOM parser trebao bi da radi i ako mu se "podmetne" druga implementacija usklađena sa W3C preporukama.
- **Glavni nedostatak:** nepogodan je za jako velike dokumente jer formiranje *in-memory* strukture može biti resursno prezahtevno

DOM PARSERI

- **Kada ga je pogodno koristiti?**
  - Kada je neophodno dobro sagledati i za obradu dobro poznavati strukturu celog dokumenta
  - Kada je neophodno reorganizovati dokument (premeštati, dodavati, sortirati elemente…)
  - Kada postoji velika verovatnoća da će nam određene informacije iz dokumenta biti potrebne više puta tokom obrade.