

1/ DOM működési elve

Első lépésben átolvassa a megadott bemeneti XML dokumentumot, majd a memóriában felépíti a hozzá kapcsolódó objektum hierarchiáját. Ez nagyobb végrehajtási költséggel jár, mint a SAX modell. A fa felépítéséhez a DOM a SAX API-t használja fel.

DOM API interface elemei

Document: A dokumentum csomópont megtestesítője

Adattagok:

- documentElement: a fa gyökér eleme, Element típus
- doctype: séma leíró, Doctype típusú

Fő módszerei:

- Element createElement (String elemnév)
- Text createTextNode (String szöveg)
- Attr createAttribute (String jellemző neve)
- NodeList getElementsByTagName (String elemnév)

Node: Általános fa csomópont.

- nodeType
- nodeName
- nodeValue
- parentNode
- childNode
- nextSibling
- attributes
- NamedNodeMap
- ownerDocument

Módszerek

- Node removeChild (Node elem)
- Node appendChild (Node elem)
- Node replaceChild (Node elemúj, Node elemregi)
- Node insertBefore (Node elemúj, Node elemregi)

feldolgozás fontosabb lépései

- Dokumentáció létrehozása
- Objektumok létrehozása
- Dokumentum metódusainak meghívása
- Dokumentum serializálása

+ CharacterData

általános szöveges Node interface -ből származtatott adattagja:

- data
- length

Metódusok

- String substringData(long kezdet, long hossz)
- deleteData(long kezdet, long hossz)
- InsertData(long kezdet, long hossz)

Attr:

elemjellemző leírása, a Node interface -ből származtatott adattagok:

- name
- specified
- value

Element

történet elem csomópontja, a Node interface-ből származtatott.

spec. módszerek:

- String getAttribute (String név)
- setAttribute (String String név, String érték)
- removeAttribute (String név)
- Attr getAttributeNode (String név)
- NodeList getElementByTagName (String név)

NodeList

csomópontok halmaza.

adatai:

- length

módszerek:

- Node item (long index)

DomException

Dom kezelőfelület által generált hiba
Attributumai

- code

3/ XML Schema alapjai

Füredi Gábor
YRNU/BP

- Az XML szintaktikát követi és egy helyesen formált

XML dokumentum

- A DTD-vel ellentétben az XML Schema sémaleírás mindig külön fájlban helyezkedik el.
- Használhatók benne gyári és egyedileg definiált típusok
- Nevesített típusokat és a közöttük lévő specializációs kapcsolatokat, illetve absztrakt típusokat is képes kezelni
- Szigorúbb kulcs, idegenkulcs meghatározások
- Gazdag struktúra elemek
- névtérak használata
- Szerkesztési, integritási elemeket tartalmaz

Megadózis

<?xml version="1.0"?>

<xs:schema
xmlns:xs="http://www.w3.org/2001/XMLSchema" >
sémaleírás

</xs:schema>

Elemi jelölő elemek

Attr.:

name: elem azon. neve

type: elem típusa

minOccurs: min. előfordulása

maxOccurs: max. előfordulása

pl.:

<xs:element jellemző> tartalom

</xs:element>

Összetett jelölő elemek:

- szekvencia $\langle xs: sequence \rangle \dots \langle /xs: sequence \rangle$
- opció lista $\langle xs: choice \rangle \dots \langle /xs: choice \rangle$
- üres elem megadása $\langle xs: element \dots \rangle \langle xs: complexType \rangle \langle /xs: element \rangle$
- végleges elem $\langle xs: element \rangle \langle xs: complexType mixed = "true" \rangle \langle /xs: element \rangle$

Névtér:

$\langle ht: könyvek xmlns:ht="URL"$
 $xmlns:vir="URL"$
 $xmlns="URL" \rangle$

$\langle ht: könyv id="1" \rangle$

$\langle ht: ISBN \rangle \dots \langle /ht: ISBN \rangle$

$\langle vir: olvasó \rangle \dots \langle uid \rangle \dots \langle /uid \rangle \langle /vir: olvasó \rangle$

$\langle vir: nhod xmlns:vir="URL" \rangle$

$\langle /vir: nhod \rangle$

$\langle /ht: könyv \rangle$

$\langle /ht: könyvek \rangle$

- Szerepe az elemek elemnevek / attribútumnevek egyértelműsítése
- elemnév önmagában nem elegendő egyértelmű azonosításra
- Névtér: a feldolgozó program azonosítója
- Megoldás: a feladat név kibővítése a feldolgozó azonosításával

tengelytípusok:

- self: maga a kontextus csomópont
- child: gyerek csomópontok, melybe nem tartoznak bele az elemjellemző és névtér csomópontok
- descendant: befoglalt csomópontok, gyerekek és azok gyerekei, az összes tetszőleges mélységben befoglalt csomópontot beleértve.
- descendant-or-self: a befoglalt csomópontok és maga a kontextus csomópontot.
- parent: tartalmazó csomópont
- ancestor: azon csomópontok, melyek befoglalják magukba a kontextus csomópontot.
- ancestor-or-self
- preceding: azon csomópontok, melyek a dokumentum sorrendben megelőzik a kontextus csomópontot és nem foglalják magukba a kontextus elemet, valamint típusuk nem elemjellemző és nem névtér
- preceding-sibling
- following
- following-sibling
- attribute
- namespace

Elemzés

Füredi Gábor
YRWBP

- *: bármely elem vagy elemjellemző a megadott tengelyre vonatkoztatva.
- comment(): megjegyzési csomópontok
- text(): szöveg csomópontok
- processing-instruction(): feldolgozási direktívák
- node(): normál csomópontok
- szim = pozíció

kifejezések és rövidítések

- new:
- *
- text()
- @new:
- @*
- new[1]
- new[last()]
- */new
- new1||new2
- || new
- ||new1/new2
- .
- ./ new
- ..
- ../@new