Správnost XML dokumentu

Spolufinancováno Evropskou unií

Ministerstvo školství, mládeže a tělovýchovy

Projekt: UčiTUL
CZ.02.02.XX/00/23_019/0008386

Správně strukturovaný dokument

- v originále **well-formed**
- dokument dodržuje syntaktická pravidla XML:
  - má jeden kořenový prvek, prvky jsou správně ukončeny (vnořeny)
  - správný zápis prvků a entit, hodnoty atributů jsou v uvozovkách atd.
  - správně vytvořené jsou i vložené prvky
- nutný předpoklad pro zpracování v XML nástrojích
- týká se formy, nikoli obsahu/struktury

Správný (validní) dokument

- v originále **valid**
- musí platit:
  - dokument je **správně strukturovaný**
  - struktura jeho prvků **odpovídá definici jazyka** (obsahuje přípustné prvky v přípustných vztazích)
- zaručuje korektní zpracování v aplikacích podporujících daný jazyk
- jak definovat jazyk?

Definice jazyka

Document Type Definition (DTD)

- definuje jazyk dokumentu
  - jaké existují prvky
  - co mohou obsahovat (a tedy jaká jsou pravidla pro jejich vzájemné vnořování)
  - jaké mají atributy
- zavádí obecnou strukturu
  - neumí datové typy – nelze např. omezit obsah prvku či hodnotu atributu na celá čísla od 1 do 100

Definice prvku

- <!ELEMENT *jméno obsah*>
- *jméno* určuje jméno prvku, musí být jednoznačné
- *obsah* omezuje, co prvek smí a nesmí obsahovat
- dva typy textových obsahů:
  - PCDATA (Parsed Character Data) – text analyzovaný procesorem, expandují se entity,... – pro prvky
  - CDATA (Character Data) – text není analyzován, bere se jako konstanta – pro atributy

Jednoduché obsahy

- EMPTY – prvek je prázdný
  <!ELEMENT br EMPTY>
- ANY – prvek může mít libovolný analyzovatelný obsah; vzdáváme se přísnější kontroly
- (#PCDATA) – prvek obsahuje text
  <!ELEMENT den (#PCDATA)>

Prvek jako obsah

- (prvek) – daný prvek
- (prvek1,prvek2,...) – prvky v daném pořadí
- (prvek*) – libovolný počet těchto prvků
- (prvek+) – alespoň jeden tento prvek
- (prvek?) – nepovinný výskyt daného prvku
- (prvek1|prvek2) – jeden nebo druhý
- pomocí závorek lze operátory aplikovat na více prvků

Příklad: Datum

```
<!ELEMENT datum (den,mesic,rok)>
<!ELEMENT den (#PCDATA)>
<!ELEMENT mesic (#PCDATA)>
<!ELEMENT rok (#PCDATA)>
```

Odpovídající XML:

```
<datum>
    <den>17</den>
    <mesic>10</mesic>
    <rok>2006</rok>
</datum>
```

Nevalidní (špatné pořadí):

```
<datum>
    <rok>2006</rok>
    <mesic>10</mesic>
    <den>17</den>
</datum>
```

Příklad: Plné jméno

```
<!ELEMENT  plnejmeno  (titul*,krestni,dalsikrestni*,prijmeni+,titul*)>
<!ELEMENT  titul (#PCDATA)>
<!ELEMENT  krestni (#PCDATA)>
<!ELEMENT  dalsikrestni (#PCDATA)>
<!ELEMENT  prijmeni (#PCDATA)>
```

```
<plnejmeno>
    <krestni>Jan</krestni>
    <prijmeni>Nový</prijmeni>
</plnejmeno>
```

```
<plnejmeno>
    <titul>Ing.</titul>
    <krestni>Emanuel</krestni>
    <dalsikrestni>Ivo</dalsikrestni>
    <dalsikrestni>Jan</dalsikrestni>
    <prijmeni>Kyselý</prijmeni>
    <prijmeni>Hrozen</prijmeni>
</plnejmeno>
```

Smíšený obsah

- text i vnořené prvky
  jako jednu z variant v „nebo“ uvést #PCDATA
- např. nadpis připouštějící text a zvýraznění (vnořený prvek em), to může být víceúrovňové
  <!ELEMENT nadpis (#PCDATA|em)*>
  <!ELEMENT em (#PCDATA|em)*>

Definice atributů

- <!ATTLIST *prvek jméno typ implicit_hodnota*>
- závěrečná trojice se opakuje pro každý atribut, nebo lze použít několik <!ATTLIST ...>
- *prvek* je jméno prvku, jehož atributy definujeme
- *jméno* určuje jméno atributu
- *typ* jeho typ (charakter, nikoli datový typ)
- *implicit_hodnota* poskytuje informace o hodnotě

Typy atributů (1)

- CDATA – libovolný (nezpracovávaný) text
- (hod1|hod2|...) – výčet přípustných hodnot
- ID – jednoznačný identifikátor (definice ident.)
  omezení: XML identifikátory nesmí začínat číslicí
- IDREF – identifikátor jiného prvku (odkaz na něj)
- IDREFS – seznam identifikátorů jiných prvků, oddělovány mezerami

Typy atributů (2)

- NMTOKEN – platné XML jméno (písmena, číslice, -, _, ., :)
- NMTOKENS – seznam jmen oddělených mezerami
- ENTITY, ENTITIES – jméno entity, seznam entit
- NOTATION – jméno notace definované pomocí <!NOTATION...>, nepoužívá se

Implicitní hodnota

- "hodnota" – konkrétní hodnota
- #REQUIRED – atribut je povinný
- #IMPLIED – atribut lze vynechat, implicitní hodnota není definována
- #FIXED “hodnota“ – hodnota je neměnná

Příklad: Telefonní seznam

- kořenovým prvkem je seznam
- obsahuje libovolné množství prvků osoba
- osoba obsahuje jmeno a alespoň jedno cislo, má také povinný atribut id obsahující jednoznačný identifikátor
- cislo má nepovinný atribut typ s hodnotami „mobil“, „stabil“ nebo „skype“

DTD pro telefonní seznam

```
<?xml version="1.0"?>

<!ELEMENT seznam (osoba*)>

<!ELEMENT osoba (jmeno,cislo+)>
<!ATTLIST osoba
    id ID #REQUIRED
>

<!ELEMENT jmeno (#PCDATA)>

<!ELEMENT cislo (#PCDATA)>
<!ATTLIST cislo
    typ (mobil|stabil|skype) #IMPLIED
>
```

Telefonní seznam – příklad

```
<?xml  version=“1.0“?>

<seznam>

<osoba id="elib">
    <jmeno>Eleonora Líbezná</jmeno>
    <cislo>123 456 789</cislo>
</osoba>

<osoba id="mojl">
    <jmeno>Mojmír Luzný</jmeno>
    <cislo typ="mobil">606 707 808</cislo>
</osoba>

</seznam>
```

Definice entit

- **interní entity:**
  <!ENTITY *jméno* "*hodnota*">
  <!ENTITY tul "Technická univerzita v Liberci">
  použití v XML: Naší školou je &tul;.
- **externí entity (textové):**
  <!ENTITY *jméno* SYSTEM "*lokátor*">
  <!ENTITY kontakt SYSTEM "doc/kontakt.xml">
  <!ENTITY *jméno* PUBLIC
  "veřejný identifikátor" "*lokátor*">

Binární entity

- <!ENTITY *jméno* SYSTEM "*lokátor*" NDATA *notace*>
- notace určuje obslužný program
  <!NOTATION *notace* SYSTEM "*program*">
- např.:
  <!ENTITY logo SYSTEM "logo.gif" NDATA gif>
  <!NOTATION gif SYSTEM "c:/graphic/irfanview/i_view.exe">
- v podstatě se nepoužívá

Parametrické entity

- zkratky používané přímo v DTD
- <!ENTITY % *jméno* "*hodnota*">
  použití v DTD: *%jméno;*
- příklad – standardní atributy:

```
<!ENTITY % stdattr "id ID #IMPLIED
                    style CDATA #IMPLIED">
<!ELEMENT nadpis (#PCDATA)>
<!ATTLIST nadpis
    %stdattr;
    uroven (kapitola|cast|podcast) #IMPLIED
>
```

Připojení DTD ke XML

- nejčastější – odkaz na externí soubor:

```
<?xml version="1.0"?>
<!DOCTYPE seznam SYSTEM "telsez.dtd">
```

- může být uvedeno i přímo v XML souboru:

```
<?xml version="1.0"?>
<!DOCTYPE seznam [
    <!ELEMENT seznam (osoba*)>
    ...
]>
```

Klady DTD

- **nejstarší definiční jazyk**
  - široce podporován
  - nástroje jsou běžně dostupné
- **jednoduché**
  - v podstatě tři konstrukce:
    ELEMENT, ATTLIST, ENTITY

Nedostatky DTD

- **neumí datové typy**
  - velmi omezené možnosti pro definici obsahu prvků a hodnot atributů
- **nepodporuje jmenné prostory**
  - problém při kombinování několika DTD
- **nezvyklá syntaxe**
  - formálně je XML
  - obsahem je vůbec nepřipomíná

Spolufinancováno Evropskou unií

Ministerstvo školství, mládeže a tělovýchovy

Projekt: UčiTUL
CZ.02.02.XX/00/23_019/0008386