

## DTD

Nguyễn Hồng Phương  
 Email: [phuong.nguyenhong@hust.vn](mailto:phuong.nguyenhong@hust.vn)  
 Site: <http://is.hut.edu.vn/~phuongnh>  
 Bộ môn Hệ thống thông tin  
 Viện Công nghệ thông tin và Truyền thông  
 Đại học Bách Khoa Hà Nội

1

## Nội dung

- DTD là gì?
- Khai báo DTD nội
- Khai báo DTD ngoại
- Khai báo phần tử
- Khai báo thuộc tính
- Phần tử vs. Thuộc tính
- Thực thể

2

## DTD là gì?

- Document Type Definition
- Là tài liệu dùng để định nghĩa kiểu dữ liệu cho các phần tử của tài liệu XML.
- Khi đọc tài liệu XML, chỉ cần đọc phần DTD là sẽ biết được cấu trúc XML

3

## Tại sao sử dụng DTD?

- Với một DTD, mỗi file XML có thể mang một mô tả định dạng của nó.
- Các nhóm độc lập có thể chấp nhận một chuẩn DTD để trao đổi dữ liệu
- Một ứng dụng có thể sử dụng một DTD chuẩn để kiểm tra dữ liệu nhận về từ bên ngoài có hợp lệ hay không.

4

## Khai báo DTD nội

- Nếu DTD được khai báo bên trong file XML, nó nằm trong định nghĩa DOCTYPE
- ```
<!DOCTYPE root-element [element-declarations]>
```

5

## Ví dụ: Khai báo DTD nội

```
<?xml version="1.0"?>
<!DOCTYPE note [
<!ELEMENT note (to,from,heading,body)>
<!ELEMENT to (#PCDATA)>
<!ELEMENT from (#PCDATA)>
<!ELEMENT heading (#PCDATA)>
<!ELEMENT body (#PCDATA)>
]>
<note>
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget me this weekend</body>
</note>
```

6

## Giải thích

- ❑ !DOCTYPE note định nghĩa phần tử gốc của tài liệu là note
- ❑ !ELEMENT note định nghĩa phần tử note chứa 4 phần tử: to, from, heading, body
- ❑ !ELEMENT to định nghĩa phần tử to thuộc kiểu #PCDATA
- ❑ !ELEMENT from định nghĩa phần tử from thuộc kiểu #PCDATA
- ❑ !ELEMENT heading định nghĩa phần tử heading thuộc kiểu #PCDATA
- ❑ !ELEMENT body định nghĩa phần tử body thuộc kiểu #PCDATA

7

## Khai báo DTD ngoại

- ❑ Nếu DTD được khai báo bên trong file bên ngoài, nó nằm trong định nghĩa DOCTYPE:

```
<!DOCTYPE root-element SYSTEM "filename">
```

8

## Ví dụ khai báo DTD ngoại

```
<?xml version="1.0"?>
<!DOCTYPE note SYSTEM "note.dtd">
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

### File note.dtd:

```
<!ELEMENT note (to,from,heading,body)>
<!ELEMENT to (#PCDATA)>
<!ELEMENT from (#PCDATA)>
<!ELEMENT heading (#PCDATA)>
<!ELEMENT body (#PCDATA)>
```

9

## Khai báo phần tử

- ❑ Phần tử <!ELEMENT> dùng để định nghĩa kiểu dữ liệu cho một phần tử của một tài liệu XML.

- ❑ Cú pháp:

```
<!ELEMENT element-name category>
hoặc
<!ELEMENT element-name (element-content)>
```

- ❑ Phần tử với kiểu rỗng

```
<!ELEMENT element-name EMPTY>
Ví dụ:
<!ELEMENT br EMPTY>
Ví dụ XML:
<br />
```

10

## Khai báo phần tử (tiếp)

- ❑ Phần tử có kiểu văn bản

```
<!ELEMENT element-name (#PCDATA)>
Ví dụ:
<!ELEMENT from (#PCDATA)>
```

- ❑ Phần tử với kiểu dữ liệu bất kỳ

```
<!ELEMENT element-name ANY>
Ví dụ:
<!ELEMENT note ANY>
```

11

## Khai báo phần tử (tiếp)

- ❑ Định nghĩa phần tử có chứa một phần tử con

```
<!ELEMENT element-name (child-name)>
Ví dụ:
<!ELEMENT note (message)>
```

- ❑ Định nghĩa một phần tử chứa nhiều phần tử con

- Cách 1:
 

```
<!ELEMENT element-name (child1,child2,...)>
      Ví dụ:
      <!ELEMENT note (to,from,heading,body)>
```

12

## Khai báo phần tử (tiếp)

```
<!ELEMENT note (to,from,heading,body)>
<!ELEMENT to (#PCDATA)>
<!ELEMENT from (#PCDATA)>
<!ELEMENT heading (#PCDATA)>
<!ELEMENT body (#PCDATA)>
```

- Chú ý, các phần tử con trong tài liệu có trình tự giống trình tự khai báo các phần tử con ở đây.
- Các phần tử con cũng có thể có các phần tử con mức dưới nữa.

13

## Khai báo phần tử (tiếp)

### □ Cách 2: sử dụng kí tự đại diện

- Phần tử có 1 hoặc nhiều phần tử con

```
<!ELEMENT element-name (child-name+)>
```

Ví dụ:

```
<!ELEMENT note (message+)>
```

- Phần tử có 0 hoặc nhiều phần tử con

```
<!ELEMENT element-name (child-name*)>
```

Ví dụ:

```
<!ELEMENT note (message*)>
```

14

## Khai báo phần tử (tiếp)

- Phần tử có 0 hoặc 1 con

```
<!ELEMENT element-name (child-name?)>
```

Ví dụ:

```
<!ELEMENT note (message?)>
```

- Phần tử có một trong các phần tử con

```
<!ELEMENT note (to,from,header,(message|body))>
```

- phần tử note phải chứa phần tử to, from, header và {message hoặc body}

- Phần tử chứa phần tử con hoặc dữ liệu

```
<!ELEMENT note (#PCDATA|to|from|header|message)*>
```

phần tử note chứa 0 hoặc nhiều kiểu văn bản, phần tử to, from, header hoặc message.

15

## Khai báo thuộc tính với <!ATTLIST>

- Phần tử <!ATTLIST> dùng để định nghĩa kiểu dữ liệu của các thuộc tính cho một phần tử trong tài liệu XML

```
<!ATTLIST element-name attribute-name attribute-type default-value>
```

Ví dụ DTD:

```
<!ATTLIST payment type CDATA "check">
```

Ví dụ XML:

```
<payment type="check" />
```

16

## Khai báo thuộc tính với <!ATTLIST>

- Các kiểu thuộc tính có thể là:

Kiểu	Mô tả
CDATA	Thuộc tính này chỉ có thể chứa kiểu dữ liệu kí tự
(en1 en2 ..)	Thuộc tính chỉ nhận giá trị từ danh sách này
ID	Giá trị của thuộc tính này không được trùng nhau và bắt đầu bởi 1 chữ cái
IDREF	Giá trị của thuộc tính này phải là một trong các giá trị của thuộc tính ID của các phần tử khác
IDREFS	Giá trị của thuộc tính này phải là các giá trị của các thuộc tính có kiểu ID

17

## Khai báo thuộc tính với <!ATTLIST>

Kiểu	Mô tả
NMTOKEN	Giá trị là một tên XML hợp lệ
NMTOKENS	Giá trị là một danh sách các tên XML hợp lệ
ENTITY	Giá trị là một thực thể
ENTITIES	Giá trị là một danh sách các thực thể
NOTATION	Giá trị là tên của một kí hiệu
xml:	Giá trị là một giá trị xml đã định sẵn

18

## Khai báo thuộc tính với <!ATTLIST>

- Giá trị mặc định: có thể nhận một trong các giá trị sau:

Giá trị	Giải thích
value	value là một giá trị mặc định nào đó, ví dụ "SoICT"
#REQUIRED	Thuộc tính này phải có giá trị
#IMPLIED	Thuộc tính này không nhất thiết phải có giá trị
#FIXED value	Giá trị thuộc tính được gán cố định

19

## Ví dụ: thuộc tính mặc định

DTD:  
 <!ELEMENT square EMPTY>  
 <!ATTLIST square width CDATA "0">

XML hợp lệ:  
 <square width="100" />

- Phần tử square là một phần tử rỗng, thuộc tính width có kiểu CDATA. Nếu không được xác định, giá trị của nó sẽ là mặc định 0 (ở đây đã xác định là 100)

20

## #REQUIRED

### Cú pháp:

```
<!ATTLIST element-name attribute-name attribute-type #REQUIRED>
```

### Ví dụ:

#### DTD:

```
<!ATTLIST person number CDATA #REQUIRED>
```

#### XML hợp lệ:

```
<person number="5677" />
```

#### XML không hợp lệ:

```
<person />
```

21

## #IMPLIED

### Cú pháp:

```
<!ATTLIST element-name attribute-name attribute-type #IMPLIED>
```

### Ví dụ:

#### DTD:

```
<!ATTLIST contact fax CDATA #IMPLIED>
```

#### XML hợp lệ:

```
<contact fax="555-667788" />
```

#### XML không hợp lệ:

```
<contact />
```

22

## #FIXED

### Cú pháp:

```
<!ATTLIST element-name attribute-name attribute-type #FIXED "value">
```

### Ví dụ:

#### DTD:

```
<!ATTLIST sender company CDATA #FIXED "Microsoft">
```

#### XML hợp lệ:

```
<sender company="Microsoft" />
```

#### XML không hợp lệ:

```
<sender company="W3Schools" />
```

23

## Thuộc tính nhận một trong các giá trị liệt kê

### Cú pháp:

```
<!ATTLIST element-name attribute-name (en1|en2|..) default-value>
```

### Ví dụ:

#### DTD:

```
<!ATTLIST payment type (check|cash) "cash">
```

#### Ví dụ XML:

```
<payment type="check" />
```

hoặc

```
<payment type="cash" />
```

24

## Phần tử vs. Thuộc tính, khi nào?

- Dữ liệu có thể lưu trữ trong các phần tử con hoặc các thuộc tính.

Ví dụ:

```
<person sex="female">
  <firstname>Anna</firstname>
  <lastname>Smith</lastname>
</person>
```

```
<person>
  <sex>female</sex>
  <firstname>Anna</firstname>
  <lastname>Smith</lastname>
</person>
```

25

## Phần tử vs. Thuộc tính, khi nào?

- Nhìn chung, không có quy tắc. Nên sử dụng phần tử con khi thông tin giống như là dữ liệu.

```
<note date="12/11/2002">
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

```
<note>
  <date>12/11/2002</date>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

26

```
<note>
  <date>
    <day>12</day>
    <month>11</month>
    <year>2002</year>
  </date>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

27

## Tránh sử dụng thuộc tính?

- Một số vấn đề với thuộc tính
  - Không thể chứa nhiều giá trị (trong khi phần tử con có thể!)
  - Không dễ dàng mở rộng trong tương lai
  - Không mô tả các cấu trúc
  - Khó để thao tác bởi đoạn mã lập trình
  - Các giá trị thuộc tính là không dễ dàng để kiểm tra một DTD

Tránh kiểu này!

```
<note day="12" month="11" year="2002"
  to="Tove" from="Jani" heading="Reminder"
  body="Don't forget me this weekend!">
</note>
```

28

## Tránh sử dụng thuộc tính?

- Ngoại lệ:

```
<messages>
  <note id="p501">
    <to>Tove</to>
    <from>Jani</from>
    <heading>Reminder</heading>
    <body>Don't forget me this weekend!</body>
  </note>
```

```
<note id="p502">
  <to>Jani</to>
  <from>Tove</from>
  <heading>Re: Reminder</heading>
  <body>I will not!</body>
</note>
</messages>
```

id ở đây là bộ đếm, bộ định danh duy nhất, không phải là một phần của dữ liệu note

29

## Thực thể

- Các thực thể là các biến được sử dụng để định nghĩa shortcuts tới đoạn text chuẩn hoặc các kí tự đặc biệt.
  - Các tham chiếu thực thể là các tham chiếu tới các thực thể
  - Các thực thể có thể được khai báo bên trong hoặc bên ngoài

30

## Khai báo thực thể bên trong

### □ Cú pháp

- `<!ENTITY entity-name "entity-value">`

### □ Ví dụ

#### Ví dụ DTD:

```
<!ENTITY writer "Donald Duck.">
<!ENTITY copyright "Copyright W3Schools.">
```

#### Ví dụ XML:

```
<author>&writer;&copyright;</author>
```

- Một thực thể có 3 phần: kí hiệu &, tên thực thể và dấu ;

31

## Khai báo thực thể ngoài

### □ Cú pháp

- `<!ENTITY entity-name SYSTEM "URI/URL">`

### □ Ví dụ

#### Ví dụ DTD:

```
<!ENTITY writer SYSTEM
"http://www.w3schools.com/entities.dtd">
<!ENTITY copyright SYSTEM
"http://www.w3schools.com/entities.dtd">
```

#### Ví dụ XML:

```
<author>&writer;&copyright;</author>
```

32

## Một số ví dụ trên internet

### □ TV Schedule DTD

- By David Moisan. Copied from <http://www.davidmoisan.org/>

### □ Newspaper Article DTD

- Copied from <http://www.vervet.com/>

33

## TV Schedule DTD

```
<!DOCTYPE TVSCHEDULE [
<!ELEMENT TVSCHEDULE (CHANNEL+)>
<!ELEMENT CHANNEL (BANNER,DAY+)>
<!ELEMENT BANNER (#PCDATA)>
<!ELEMENT DAY (DATE,(HOLIDAY|PROGRAMSLOT+)+)>
<!ELEMENT HOLIDAY (#PCDATA)>
<!ELEMENT DATE (#PCDATA)>
<!ELEMENT PROGRAMSLOT (TIME,TITLE,DESCRIPTION?)>
<!ELEMENT TIME (#PCDATA)>
<!ELEMENT TITLE (#PCDATA)>
<!ELEMENT DESCRIPTION (#PCDATA)>
<!ATTLIST TVSCHEDULE NAME CDATA #REQUIRED>
<!ATTLIST CHANNEL CHAN CDATA #REQUIRED>
<!ATTLIST PROGRAMSLOT VTR CDATA #IMPLIED>
<!ATTLIST TITLE RATING CDATA #IMPLIED>
<!ATTLIST TITLE LANGUAGE CDATA #IMPLIED>
]>
```

34

## Newspaper Article DTD

```
<!DOCTYPE NEWSPAPER [
<!ELEMENT NEWSPAPER (ARTICLE+)>
<!ELEMENT ARTICLE (HEADLINE,BYLINE,LEAD,BODY,NOTES)>
<!ELEMENT HEADLINE (#PCDATA)>
<!ELEMENT BYLINE (#PCDATA)>
<!ELEMENT LEAD (#PCDATA)>
<!ELEMENT BODY (#PCDATA)>
<!ELEMENT NOTES (#PCDATA)>
<!ATTLIST ARTICLE AUTHOR CDATA #REQUIRED>
<!ATTLIST ARTICLE EDITOR CDATA #IMPLIED>
<!ATTLIST ARTICLE DATE CDATA #IMPLIED>
<!ATTLIST ARTICLE EDITION CDATA #IMPLIED>
<!ENTITY NEWSPAPER "Vervet Logic Times">
<!ENTITY PUBLISHER "Vervet Logic Press">
<!ENTITY COPYRIGHT "Copyright 1998 Vervet Logic Press">
]>
```

35

## Hỏi - đáp



36

## Lời hay ý đẹp

---

"Thiếu tự tin là nguyên nhân của phần lớn mọi sự thất bại"

Bowee