

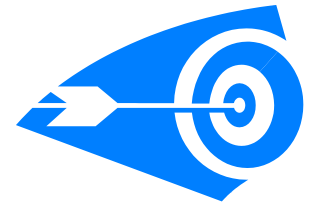
Basis Data 2

12

Database Internet

*Arif Basofi, S.Kom. MT.
Teknik Informatika, PENS*

Tujuan



- 1) Memahami konsep dari pengembangan database berbasis web:
 - ▶ World Wide World
 - ▶ Pendahuluan HTML
 - ▶ Databases dan Web
- 2) Mampu mengembangkan aplikasi basis data berbasis web:
 - ▶ Arsitektur (Application Servers dan Server-Side Java)
 - ▶ Pendahuluan XML
 - ▶ XML DTDs
 - ▶ Domain-Specific DTDs
 - ▶ Querying XML data
 - ▶ Model data semi-terstruktur

World Wide Web (WWW)

- ▶ Web memungkinkan mengakses suatu file dimanapun pada internet.
- ▶ Suatu file diidentifikasi dengan *Universal Resource Locator (URL)*, contohnya : <http://www.eepis-its.edu/~database/index.html>
- ▶ URL diatas mengidentifikasi suatu file bernama **index.html**, disimpan pada direktory database pada mesin **www.eepis-its.edu**.
- ▶ File ini adalah suatu dokumen yang mempunyai format *Hypertext Markup Language (HTML)* dan berisi beberapa link ke file lain (yang diidentifikasi melalui URL-nya).
- ▶ Perintah diterjemahkan oleh *Web browser* seperti Microsoft Internet Explorer, Netscape Navigator atau Opera untuk menampilkan dokumen secara atraktif dan pemakai kemudian dapat menavigasi ke dokumen yang berhubungan dengan memilih link.
- ▶ Suatu koleksi beberapa dokumen disebut *web site* dan diatur menggunakan program yang disebut *web server*, yang menerima URL dan menampilkan dokumen yang diperlukan.
- ▶ **World Wide Web/Web** adalah kumpulan *web site* yang dapat diakses di internet.

World Wide Web (WWW)...

- ▶ **HTML** menghubungkan isi URL, yang mengidentifikasi tempat yang berisi file yang berhubungan.
- ▶ Jika seorang pemakai men-click pada suatu link, web browser menghubungkan ke web server tujuan menggunakan protokol koneksi HTTP dan mengirim link URL.
- ▶ Jika browser menerima file dari web server, akan diperiksa tipe file dengan melihat **extension** dari nama file.
- ▶ Web browser menampilkan file berdasarkan tipe file dan jika perlu memanggil program aplikasi untuk menangani file.
- ▶ Sebagai contoh, sebuah file yang diakhiri **.txt** merupakan file teks tak berformat, web browser menampilkan dengan menterjemahkan karakter ASCII dalam file.
- ▶ Struktur dokumen yang lebih tepat dapat dikodekan dalam HTML, yang menjadi cara standart struktur web page.
- ▶ Contoh lain, file **.doc** merupakan dokumen Microsoft Word dan Web browser menampilkan file dengan menggunakan Microsoft Word.

HyperText Markup Language (HTML)

- ▶ *Hipertext Markup Language* (HTML) merupakan bahasa pemrograman berbentuk skrip yang sederhana.
- ▶ Teks diawali dan diakhiri dengan perintah yang disebut tags, biasanya terdiri dari start tag dan end tag.
- ▶ Contoh skrip HTML untuk melihat daftar buku :

```
<HTML><BODY>
Fiction:
<UL><LI>Author: Milan Kundera</LI>
  <LI>Title: Identity</LI>
  <LI>Published: 1998</LI>
</UL>
Science:
<UL><LI>Author: Richard Feynman</LI>
  <LI>Title: The Character of Physical Law</LI>
  <LI>Hardcover</LI>
</UL>
</BODY></HTML>
```

Database dan WEB

- ▶ Halaman web berisi hasil dari **query** database.
- ▶ Untuk **membangkitkan** halaman web, dilakukan hal-hal sebagai berikut :
 - ▶ Web server membuat proses baru dari program yang berinteraksi dengan basis data.
 - ▶ Web server berkomunikasi dengan program tersebut melalui **CGI** (*Common gateway interface*).
 - ▶ Program membangkitkan halaman hasil dengan isinya berasal dari basis data.
 - ▶ Protokol lain selain **CGI** adalah **ISAPI** (*Microsoft Internet Server API*) atau **NSAPI** (*Netscape Server API*).

ARSITEKTUR APPLICATION SERVER DAN SERVER-SIDE JAVA

- ▶ Pada CGI, setiap halaman merupakan hasil pembuatan sebuah proses baru sehingga sangat tidak efisien.
- ▶ Oleh karena itu dikembangkan arsitektur **Application Server** yaitu suatu perangkat lunak antara **web server** dan **aplikasi**.
- ▶ Fungsi dari **Application Server** adalah:
 - ▶ Mengendalikan thread atau proses pre-fork untuk performansi
 - ▶ Tempat koneksi basis data (dan menggunakan kembali koneksi yang tersedia)
 - ▶ Integrasi antara sumber data yang heterogen
 - ▶ Manajemen transaksi melibatkan beberapa sumber data
 - ▶ Manajemen session

ARSITEKTUR APPLICATION SERVER DAN SERVER-SIDE JAVA...

- ▶ Pemrosesan pada sisi server dapat menggunakan :
 - ▶ **Java Servlet** : yaitu program java yang berjalan pada server dan berinteraksi dengan server melalui API yang didefinisikan dengan baik (*well-defined*).
 - ▶ **JavaBeans** : yaitu komponen perangkat lunak yang dapat digunakan kembali (*reusable*) yang ditulis dalam bahasa pemrograman Java.
 - ▶ **Java Server Page** dan **Active Server Page** : yaitu kode yang berada di dalam halaman web yang diterjemahkan oleh web server.

XML

- ▶ **Extensible HTML (XML)** merupakan bahasa skrip versi lanjutan dari HTML.
 - ▶ XML merupakan penggabungan antara **SGML** (turunan dari *Generalized Markup Language (GML)* milik IBM) dan HTML dimana kekuatan SGML digabungkan dengan kesederhanaan HTML.
 - ▶ XML menggunakan definisi *markup language* baru yang disebut **Document Type Declaration (DTDs)**.
 - ▶ Elemen yang terdapat pada XML meliputi :
 - ▶ blok utama yang dibangun secara terstruktur dengan XML
 - ▶ tag mulai (*start tag*) dan akhir (*end tag*)
 - ▶ harus secara tepat berulang
 - ▶ Elemen pada XML dapat mempunyai atribut yang merupakan informasi tambahan mengenai elemen tersebut.
 - ▶ Entity pada XML serupa dengan makro yang berupa teks biasa.
 - ▶ Pada XML juga tersedia perintah untuk komentar dan elemen yang
- ▶ 9 terpenting adalah **Document Type Declaration (DTDs)**.

XML...

- ▶ Contoh skrip XML untuk melihat daftar buku adalah sebagai berikut:

```
<?XML version="1.0" standalone="yes"?>
<!DOCTYPE BOOKLIST SYSTEM "booklist.dtd">
<BOOKLIST>
<BOOK genre="Fiction">
  <AUTHOR>
    <FIRST>Milan</FIRST><LAST>Kundera</LAST>
  </AUTHOR>
  <TITLE>Identity</TITLE>
  <PUBLISHED>1998</PUBLISHED>
<BOOK genre="Science" format="Hardcover">
  <AUTHOR>
    <FIRST>Richard</FIRST><LAST>Feynman</LAST>
  </AUTHOR>
  <TITLE>The Character of Physical Law</TITLE>
</BOOK></BOOKLIST>
```

XML : DTDs

- ▶ Sebuah *Document Type Declaration* (DTDs) adalah himpunan aturan yang mendefinisikan elemen, atribut dan entity yang diperbolehkan pada dokumen.
- ▶ Dokumen pada XML mempunyai bentuk yang baik jika tidak mempunyai asosiasi DTD tetapi digunakan berulang dengan tepat.
- ▶ Dokumen XML valid jika mempunyai DTD-an dokumen mengikuti aturan dlm DTD.
- ▶ Contoh DTDs untuk melihat daftar buku adalah sebagai berikut:

```
<!DOCTYPE BOOKLIST [  
  <!ELEMENT BOOKLIST (BOOK)*>  
  <!ELEMENT BOOK (AUTHOR, TITLE, PUBLISHED?)>  
  <!ELEMENT AUTHOR (FIRST, LAST)>  
  <!ELEMENT FIRST (#PCDATA)>  
  <!ELEMENT LAST (#PCDATA)>  
  <!ELEMENT TITLE (#PCDATA)>  
  <!ELEMENT PUBLISHED (#PCDATA)>  
  <!ATTLIST BOOK genre (Science|Fiction) #REQUIRED>  
  <!ATTLIST BOOK format (Paperback|Hardcover) "Paperback">  
]>
```

Spesifikasi Domain Pada DTDs

- ▶ Untuk menangani sumber-sumber data yang berlainan, dikembangkan standarisasi DTDs untuk memungkinkan domain dapat menukar data diantara sumber-sumber yang heterogen.
- ▶ Contohnya adalah domain pada DTDs untuk mengkodekan material matematika pada web dengan menggunakan *Mathematical Markup Language (MathML)*.
- ▶ Perbedaan HTML dengan MathML dapat dilihat pada persamaan matematika di bawah ini :

- ▶ Dalam HTML : ``

- ▶ Dalam MathML :

```
<apply> <power/>  
  <apply> <plus/> <ci>x</ci> <ci>y</ci> </apply>  
  <cn>2</cn>  
</apply>
```

Melakukan Query Data XML (XML-QL)

- ▶ Tujuan melakukan query data XML adalah melakukan manipulasi dokumen XML dengan menggunakan bahasa deklaratif tingkat tinggi.
- ▶ Query data XML belum dilakukan standarisasi.
- ▶ Contoh query dengan XML-QL :

```
WHERE
<BOOK>
  <NAME><LAST>$1</LAST></NAME>
</BOOK> in "www.booklist.com/books.xml"
CONSTRUCT <RESULT> $1 </RESULT>
```

- ▶ Contoh query dengan XML-QL yang lebih kompleks:

```
WHERE <BOOK> $b <BOOK> IN "www.booklist.com/books.xml",
<AUTHOR> $n </AUTHOR>
<PUBLISHED> $p </PUBLISHED> in $e
CONSTRUCT
<RESULT>
<PUBLISHED> $p </PUBLISHED>
WHERE <LAST> $1 </LAST> IN $n
CONSTRUCT <LAST> $1 </LAST>
</RESULT>
```

Model Data Semi Terstruktur

- ▶ **Data semi terstruktur** adalah data dengan struktur parsial.
- ▶ Semua model data untuk data semi terstruktur menggunakan beberapa tipe graph berlabel.
- ▶ **OEM (Object Exchange Model)** mempunyai karakteristik sebagai berikut :
 - ▶ Obyek berbentuk triple (label, tipe, nilai).
 - ▶ Obyek yang kompleks di dekomposisi secara hirarki ke dalam obyek yang lebih kecil.
- ▶ Contoh model data daftar buku menggunakan OEM dapat dilihat pada Gambar:

