

Mata Kuliah : Sistem Basis Data

Dosen : Karmilasari



LINGKUNGAN BASIS DATA

Outline

- Arsitektur basis data
- Data independence
- Konsep DBMS, Komponen DBMS, Fungsi DBMS dan Bahasa yang digunakan dalam DBMS
- Model Data : berbasis objek, berbasis record, konseptual dan fisik (overview model data berbasis record : model data relasional, jaringan, hirarki)
- Data dictionary
- Arsitektur DBMS multiuser : file server, teleprocessing, client server (contoh : 2 tier, 3 tier)

Model Data

- **Model Data** : suatu konsep untuk menggambarkan struktur basis data dan batasan tertentu yang harus dipatuhi
- **Operasi Model Data** : operasi untuk menentukan temu kenali/retrieval basis data dan update, dengan mengacu pada konsep model data. Operasi pada model data termasuk operasi dasar dan operasi yang ditetapkan oleh pengguna

Kategori Model Data

- **Model data Konseptual (high-level, semantik)** : konsep yang dekat dengan cara pengguna melihat data (disebut juga berbasis entitas atau model data berbasis objek)
- **Model data Fisik (low-level, internal)** : konsep yang menggambarkan detail bagaimana data disimpan dalam komputer
- **Model data Implementasi/representasi** : gabungan konsep di atas yang menyeimbangkan pandangan pengguna dan beberapa rincian penyimpanan komputer

Sejarah Model Data

- **Model Relasional** : diusulkan tahun 1970 oleh EF Codd (IBM), sistem komersial pertama pada tahun 1981-1982. Beberapa produk komersial yang dikenal : DB2, Oracle, SQL Server, Sybase, Informix.
- **Model Jaringan** : pertama kali diperkenalkan oleh Honeywell tahun 1964-1965 (IDS System). Diadopsi penuh karena dukungan CODASYL (CODASYL-Laporan DBTG 1971), kemudian diimplementasikan dalam berbagai sistem – IDMS (Cullinet-sekarang CA), DMS 1100 (Unisys), IMAGE (HP), VAX-DBMS (Digital Equipment Corp)
- **Model Data Hirarkikal** : diimplementasikan dalam upaya bersama IBM dan Rockwell sekitar tahun 1965. Menghasilkan keluarga sistem IMS yang sangat populer. Sistem lain yang berdasarkan model ini adalah System 2k (SAS inc.)

Sejarah Model Data

- **Model Berorientasi Objek** : beberapa model yang diusulkan untuk diterapkan dalam sistem database, seperti model persisten bahasa pemrograman berorientasi objek seperti C++ (di ObjectStore atau VERSANT), Smalltalk (misal : GEMSTONE). Selain itu sistem seperti O2, ORION (pada MCC-selanjutnya ITASCA), IRIS (pada HP digunakan pada Open OODB)
- **Model Relasional Objek** : trend terbaru. Dimulai dengan Informix Universal Server. Contoh versi terbaru dari Oracle-10i, DB2 dan SQL Server

Model Hirarkikal

- **Keuntungan**

- Sederhana dalam hal pembangunan dan pengoperasian
- Sesuai dengan jumlah domain hirarki yang teroganisir, misal perakitan dalam fabrikasi, organisasi dalam perusahaan
- Bahasanya sederhana, dibangun dengan GET, GET UNIQUE, GET NEXT, GET NEXT WITHIN PARENT dsb.

- **Kerugian**

- Navigasional dan prosedural dalam pengolahan
- Database divisualisasikan sebagai susunan linear record
- Hanya memiliki sedikit ruang untuk optimasi query

Model Jaringan

- **Keuntungan**
 - Mampu memodelkan hubungan kompleks dan merepresentasikan penambahan/pengurangan relasi
 - Dapat menangani banyak situasi pemodelan menggunakan tipe record dan relasional
 - Bahasanya navigasional, menggunakan FIND, FIND member, FIND owner, FIND NEXT within set, GET dsn.
 - Programmer dapat melakukan navigasi secara optimal melalui basis data
- **Kerugian**
 - Sifat navigasi dan prosedural pengolahan
 - Database berisi array kompleks dari pointer yang melalui suatu record.
 - Hanya memiliki sedikit ruang untuk optimasi query

Skema vs. Instance

- **Skema Database** : dekskripsi database, termasuk deskripsi dari struktur basis datadan batasan yang harus dipatuhi basis data
- **Diagram Skema** : tampilan diagram dari beberapa aspek skema basis data
- **Konstruk Skema** : sebuah komponene dari skema atau objek dalam skema, misal : MAHASISWA, KURSUS
- **Instance Database** : data aktual yang disimplan dalam basis data pada saat tertentu. Disebut juga state database (atau occurence)

Skema Database vs. State Database

- **State Database** : mengacu pada isi dari basis data pada suatu waktu.
- **Initial Database State** : mengacu pada basis data saat dimuat
- **Valid State** : suatu state yang memenuhi struktur dan batasan dari basis data
- **Perbedaan** :
 - Skema database sangat jarang berubah.
 - State database berubah setiap kali basis data diperbaharui
 - Skema disebut intensi, sedangkan State disebut ekstensi

Tiga Skema Arsitektur

Ditujukan untuk mendukung karakteristik DBMS :

- **Program-data independence**
- Dukungan beberapa sudut pandang data /**multiple view data**

Tiga Skema Arsitektur

Mendefinisikan skema DBMS pada tiga tingkatan :

- **Skema Internal** di tingkat internal, untuk menggambarkan struktur penyimpanan fisik dan jalur akses. Biasanya menggunakan model data fisik
- **Skema Konseptual** pada tingkat konseptual, untuk menggambarkan struktur dan batasan seluruh basis data untuk komunitas pengguna
- Skema **Eksternal** pada tingkat eksternal, untuk menggambarkan berbagai pandangan pengguna. Biasanya menggunakan model data yang sama sebagai tingkat konseptual.

Tiga Skema Arsitektur

- **Pemetaan** antara tingkat skema diperlukan untuk mengubah permintaan transformasi dan data. Program mengacu pada skema eksternal dan dipetakan oleh DBMS untuk skema internal saat eksekusi.

Data Independence

- **Logical Data Independence** : kapasitas untuk mengubah skema konseptual tanpa harus mengubah skema eksternal dan program aplikasi
- **Physical Data Independence** : kapasitas untuk mengubah skema internal tanpa harus mengubah skema konseptual

Data Independence

- Ketika skema ada tingkat yang lebih rendah berubah, hanya pemetaan antara skema ini dan skema di tingkat yang lebih tinggi yang perlu diubah dalam DBMS, dan didukung sepenuhnya dalam hal data independence. Skema yang lebih tinggi sendiri tidak berubah. Oleh karena itu, program aplikasi tidak perlu diubah karena mereka mengacu pada skema eksternal.

Bahasa DBMS

- **Data Definition Language (DDL):** digunakan oleh DBA dan desainer basis data untuk menentukan skema konseptual dari database. Dalam banyak DBMS, DDL juga digunakan untuk mendefinisikan skema internal dan eksternal (tampilan). Dalam beberapa DBMS, DDL terpisah dari Storage Definition Language (SD) dan View Definition Language (VDL) yang digunakan untuk mendefinisikan skema internal dan eksternal.

Bahasa DBMS

- **Data Manipulation Language (DML)** : digunakan untuk melakukan temu kenali /retriev dan update basis data
 - Perintah DML (**sub language**) dapat ditanamkan dalam bahasa pemrograman tujuan (**host language**) seperti : COBOL, C atau Bahasa Assembly
 - Atau perintah DML dapat pula berdiri sendiri dan diterapkan secara langsung (**query language**)

Pengantarmukaan DBMS

- Pengantarmukaan bahasa query, stand alone
- Pengantarmukaan programmer untuk DML tertanam dalam bahasa pemrograman :
 - Pendekatan Pre-Compiler
 - Pendekatan Procedure (Subroutine) Call
- Pengantarmukaan yang user friendly :
 - Berbasis menu, populer untuk browsing di web
 - Berbasis Form, dirancang untuk pengguna naif
 - Berbasis Grafis (Point & Click, Drag & Drop, dst)
 - Bahasa Natural : permintaan dalam bahasa Inggris
 - Kombinasi dari bentuk di atas

Sistem Utilitas Basis Data

Sistem utilitas digunakan untuk menjalankan fungsi :

- Loading data disimpan dalam file ke dalam basis data
- Back up basis data secara berkala pada tape
- Reorganisasi struktur file basis data
- Utilitas monitoring kinerja
- Fungsi lain, seperti pengurutan, monitoring pengguna, kompresi data, dsb.

Perangkat Lain

- **Data dictionary/repository**
 - Digunakan untuk menyimpan deskripsi skema dan informasi lain seperti keputusan desain, deskripsi program aplikasi, informasi pengguna, standar pengguna, dll
 - Active data dictionary diakses oleh perangkat lunak DBMS dan pengguna/DBA
 - Passive data dictionary diakses pengguna/DBA
- Application Development Environment dan CASE (Computer Aided Software Engineering) tools , seperti Power Builder (Sybase), Builder (Borland)

Referensi

- Connolly, Thomas; Beg, Carolyn; Strachan, Anne; **Database Systems : A Practical Approach to Design, Implementation and Management**, 3rd edition, Addison Wesley, 2001.
- Date, C.J; **An Introduction to Database System**, Addison Wesley Publishing Company; Vol.7, New York, 2000
- Elmasri, Ramez; Navathe, Shamkant B; **Fundamental of Database Systems**, The Benjamin/Cummings Publishing Company, Inc., California, 2001
- Korth, H; **Database System Concept**, Mc Graw Hill, 4th edition, 2002
- Mc Fadden, F; Hoffer, Jeffrey A; **Modern Database Management**, 4th edition, Benjamin Publishing Company Inc, 1994

DISKUSI