BAB II

LANDASAN TEORI

2.1. Web Server

Dengan terus melajunya roda perkembangan teknologi, web berkembang menjadi alat bantu yang tidak hanya mampu menyediakan informasi, namun juga mampu untuk mengolah informasi. Proses pengolahan informasi dengan memanfaatkan teknologi web menyebabkan web menjadi media informasi yang dinamis. Hal ini membutuhkan penggunaan sarana teknis yang menentukan. Sarana tersebut banyak bergantung pada penggunaan perangkat lunak yang kuat, aman, terpercaya, dan cepat. Perangkat lunak penentu yang dibutuhkan antara lain adalah sebuah *sever-side scripting language*, atau juga disebut sebagai CGI (Common Gateway Interface), yang banyak tersedia di pasaran seperti: Perl, ASP (Active Server Page), Cold Fusion, Python, dan PHP.

2.1.1 Sejarah PHP (PHP Hypertext Preprocessor)

PHP adalah suatu bahasa *server-side* yang didesain khusus untuk aplikasi web. PHP dapat disisipkan diantara HTML, dan karena bahasa *server-side*, maka bahasa PHP akan dieksekusi di server, yang dikirimkan ke browser adalah "hasil jadi" bentuk HTML, dan kode PHP tidak akan terlihat.

PHP dahulu merupakan proyek pribadi dari Rasmus Lerdorf tahun 1994 (dengan dikeluarkannya PHP versi 1 yang digunakan untuk membuat home page pribadi yang dipublikasikan 8 juni 1995). Versi pertama ini berupa kumpulan script

PERL. Untuk versi kedua Ramus menulis ulang script-script PERL tersebut menggunakan bahasa C, kemudian menambahkan fasilitas untuk form HTML dan koneksi MYSQL. Adapun PHP didapat dari singkatan *PHP Hypertext Preprocessor*.

Setelah mengalami perkembangan, Zeev Suraski dan Andi Gutmans (termasuk Rasmus) yang tergabung dalam suatu kelompok open source, maka mereka memulai versi 3 PHP nya, PHP telah menampakkan keunggulan sebagai salah satu bahasa *server scripting* yang handal. Melalui perkembangan yang pesat ini banyak fasilitas yang ditambahkan oleh kelompok ini PHP disebut sebgai "PHP: Hypertext Preprocessor". Pada bulan Mei 2000 dirilis PHP versi 4 yang dikeluarkan oleh Zend Engine yaitu sebuah produk yang diciptakan oleh Suraski dan Gutmans. Kehandalan PHP diperlengkap dengan dirilisnya PHP versi 5 pada 13 Juli 2004 yang mendukung OOP serta berbagai manajemen database yang luas dan penanganan error yang mudah. Sintak yang digunakan berasal dari bahsa C, Java maupun PERL. Sampai tulisan ini dibuat versi PHP yang terbaru adalah versi 5.2.0 yang dirilis pada November 2006.

PHP merupakan bahasa script yang digunakan untuk membuat halaman web yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh client. Mekanisme ini menyebabkan informasi yang diterima client selalu yang terbaru. Semua script PHP dieksekusi pada server dimana script tersebut dijalankan. Oleh karena itu, spesifikasi server lebih berpengaruh pada eksekusi dari script php daripada spesifikasi client. Namun tetap diperhatikan bahwa web yang dihasilkan tentunya harus dapat dibuka oleh browser pada client. Dalam hal ini versi dari html yang digunakan harus didukung oleh browser client.

PHP termasuk dalam *Open Source Product*. Jadi dapat menambah *source code* dan dapat mendistribusikannya secara bebas. PHP juga diedarkan secara gratis. Kita bisa mendapatkannya secara gratis. PHP juga dapat berjalan diberbagai web server semisal IIS, Apache, PWS, dll.

2.1.2 PHP 5

PHP 5 dirilis tanggal 13 Juli 2004. PHP 5 dapat digunakan pada hampir semua sistem operasi utama, seperi Linux, varian Unix (mencakup HP-UX, Solaris, dan Openbsd), Microsoft Windows, Mac OS X, RISC o, dan mungkin juga yanga lain. PHP juga mendukung hampir semua *web server*, seperti Apache, Microsoft Internet Information Server, Personal Web Server, Netscape, dan iPlanet Servers, Oreilley Website Pro Server, Caudium, Xitami, Omni HTTPd, dan lain-lain. Sebagian besar server PHP mempunyai modul dan untuk yang mendukung standar CGI maka PHP bekerja sebagai pemroses CGI.

PHP memberi kebebasan untuk memilih sistem operasi dan server web. PHP juga mmberi pilihan untuk menggunakans pemrograman biasa atau program berorientasi objek *(Object Oriented Programming*/OOP*)* atau gabungan keduanya. PHP tidak membtasi jumlah HTML yang dihasilkan. Kemampuan PHP meliputi dukungan terhadap gambar, file PDF, dan bahkan *movie flash* (menggunakan *libswf* dan *ming*) secara *on the fly*. PHP juga mendukung teks, seperti file XHTML dan XML.

Salah satu fitur paling penting pada PHP adalah PHP mendukung banyak basis data. Membuat web yang menampilkan basis data bukan hal sulit untuk PHP.

Sistem atabase yang didukung oleh PHP 5 adalah: Adabas D, dBase, Empress, File Pro, Hyperware, IBM DB2, Informix, Ingres, Inter Base, Front Base Direct MS-SQL, MySQL, ODBC, Oracle (OC17 dan OC18), Ovrimos, PostgreSQL, SQLite, Solid, Sybase, Velocis, Unix dbm. PHP mempunyai ekstensi DBX yang memungkinkan penggunaan basis data dengan ekstensi tersebut. Apalagi PHP mendukung ODBC *(Open Database Connection Standard)* sehingga dapat menghubungkan PHP ke basis data mana pun yang mendukung standard ini.

PHP mendukung komunikasi dengan layanan lain menggunakan protokol seprti LDAP,IMAP, SNMP, NNTP, POP3, HTTP, dan COM (pada Windows). User juga dapat membuka soket jaringan dan berinteraksi dengan protokol lain. PHP medukung pertukaran data kompleks WDDX secaa virtual pada semua bahasa pemrograman web. Untuk interkoneksi, PHP mendukung perubahan objek Java dan menggunkannya sebgai objek PHP. Penggunaan ekstensi CORBA dapat digunakan untuk mengakses objek jarak jauh.

PHP mempunyai fitur *text processing* yang sangat bermanfaat, dari *POSIX Extended* atau *Perl Regular Expression* untuk mem-*parsing* dokumen . Untuk mem-*parsing* dan mengakses dokumen XML, PHP 4 mendukung standard SAXOFON dan DOM. Untuk mengubah dokumen XSLT. PHP 5 menstandarisasi semua ekstensi XML menjadi libxml2 yang tangguh dan memperluas fitur dengan menambahkan SimpleXML dan mendukung XMLREADER. Penggunaan PHP dalam bidang *e-commerce* meliputi pembayaran *Cybercash, Cybermut, VeriSign Payflow Pro,* dan fungsi MCVE yang bermanfaat untuk program pembayaran secara online. Fitur-fitur

lain yang mendukung PHP 5 adalah fungsi mesin pencari *mnoGoSearch*, fungsi *Getewy IRC*, utilitas pemadat (gzip, bz2), konversi penanggalan, terjemahan, dan lain-lain.

PHP 5 dan Zend Engine 2 telah memperbaiki tampilan dan kemampuan PHP. Hanya ada sedikit perubahan kode sehingga perubahan kode dari PHP 4 menjadi PHP 5 sangat mudah, Sebagian besar kode PHP 4 masih dapat berjalan tanpa mengalami perubahan, tetapi tetap harus teliti terhadap sedikit perubahan yang ada. Beberapa perbaikan yang dilakukan pada PHP 5 yaitu:

1) Kinerja

   PHP 5 akan lebih cepat dibandingkan PHP 4 karena model eksekusi telah diubah. PHP 4 masih menggunakan banyak statemen *switch* ketika mengkesekusi operasi dari *aray-array*. PHP 5 menggunakan fungsi pointer sehingga pemanggilan fungsi akan menjadi 40% lebih cepat. Algoritma program telah diperbaiki agar lebih efisien.

2) Konstanta

   Pada versi sebelumnya digunakan definisi () untuk mendeklarasikan sebuah konstanta. Hal ini agar pemanggilan fungsi untuk proses ekseskusi menjadi lebih lambat. Mungkin tidak menjadi masalah jika pemakainya idak banyak. Pemakaian konstanta akan menghemat waktu kompilasi. Seperti

pada bahasa pemrograman yang lain, konstanta diletakkan dengan nilai sebelum eksekusi.

3) XML manjadi lebih handal

XML yang didukung pada PHP 4 masih mengalami banyak masalah. Ekstensi DOM dan XSLT masih berupa eksperimen. SAX bisa bekerja dengan baik, tetapi banyak mengalami kekacauan di dalamnya. Pada PHP 4, proses parsing pada XML memerlukan kode yang panjang. Semua ekstensi XML baru menggunakan *libxml* dan *libxslt*. Libxml adalah salah satu implementasi yang cepat. Ekstensi XML di PHP 5 antara lain: xml, dom, simplexml, xsl, dan soap. Ekstensi xml, dom, dan simplexml akan dikompilasi secara *default* dan memperbaiki dukungan XML pada *hosting*.

4) CGI

Pada PHP 5 ada beberapa perubahan dalam penamaan file CLI dan CGI. Di PHP 5, versi CGI kembali diberi nama php-cgi.exe (sebelumnya php.exe) dan versi CLI sekarang berada pada direktori utama (sebelumnya cli/php.-exe). Pada PHP 5 juga diperkenalkan suatau model baru: php-win.exe. Versi ini sama dengan versi CLI, tetapi php-win tidak memberi keluaran apapun dan juga tidak menyediakan konsol. Meskipun register_argc_argv dibuat off, hal ini tidak akan menimbulkan pengaruh apapun pada CLI. Perubahan nama pada modul ISAPI dari PHP 4 xxx menjadi PHP 5 xxx memerlukan beberapa perubahan pada konfigurasi file. Nama file CLI dan

CGI juga mengalami perubahan. Apabila *wwb server* menjalankan PHP pada module CGI maka user harus ingat bahwa versi CGI telah berubah nama dari php.exe menjadi php-cgi.exe.

5) Pelaporan Kesalahan

Mulai dari PHP 5, pelaporan kesalahan mempunyai konstanta baru yaitu konstanta E_STRICT dengan nilai 2048. Dengan pelaporan kesalahan, setiap menulis kode dan mengalami keslahan maka akan ada saran perbaikan untuk kesalahan tersebut

6) Adanya fungsi-fungsi baru Array

a) Array_combine() – membuat array dengan satu array sebgaia kunci *(key)* dan yang lain sebagai nilainya.

Array_diff_uassoc() – menghitung perbedaaan array dengan pengecekan indeks dengan menjalankan fungsi *Callback Supplied User*.

b) Array_udiff() – menghitung perbedaan array dengan menggunakan fungsi *Callback* untuk pebandingan data.

c) Array_udiff_assoc() – menghitung perbedaan array dengan pengecekan indeks tambahan. Data dibandingkan dengan menggunakan fungsi *Callback*

d) Array_udiff_uassoc() – menghitung perbedaan array dengan pengecekan indeks tambahan. Data dibandingkan dengan menggunakan fungsi *Callback.* Indeks juga dicek menggunakan funsi *Callback*.

e) Array_walk_rcursive() – memberikan fungsi yang digunakan secara berulang-ulang kepada tiap-tiap anggota dari suatu array.

f) Array_uintersect_assoc() – menghitung penyimpangan antar-array dengan pengecekan indeks tambahan. Data diperbandingkan menggunakan fungsi *Callback*.

g) Array_uintersect_uassoc() – menghitung penyimpangan antar-array dengan pengecekan indeks tambahan. Antara data dan indeks diper-bandingkan dengan menggunakan suatu fungsi *Callback*.

h) Array_uintersect() – menghitung penimpangan antar-larik. Data diperbandingkan memakai fungsi *Callback*

i) Interbase:

ibase_affected_rows() – mengembalikan jumlah baris yang disebabkan oleh *query* sebelumnya.

j) Stream

Stream_copy_to_stream() – mengopi data dari suatu steram ke stream yang lain.

Stream_get_line() – mengambil sutau baris dari stream sumber dan memberikannya kepada delimiter.

k) Korelasi tanggal dan waktu

Idate() – memformat tanggal dan waktu lokal sebagai bilangan bulat/integer.

Date_sunset() – waktu matahari terbenam unutk hari dan lokasi/daerah yang ditentukan.

Date_sunrise() – waktu matahari terbit untuk hari dan lokasi/daerah yang ditentukan.

Time_nanosleep() – penundaan untuk beberapa detik atau nanodetik

l) String

Str_split() – mengonversi suatu string menjadi sebuah array

7) Direktif baru

a) Mail.force_extra_parameters – memaksa beberapa parameter spesifik tambahan menjadi parameter ekstra untuk pengiriman surat (mail) secara biner. Parameter ini akan selalu mengganti parameter ke-5 pada sebuah mail, bahkan pada *safe mode*.

b) Register_long_arrays – mengijinkan atau tidak mengijinkan PHP untuk meregistrasi $HTTP_*_VARS yang panjang.

c) Session.hash_function – memilih fungsi *ash* (MD5 atau SHA-1).

d) Session.hash_bits_per_character – mendefinisikan berapa banyak bit yang disimpan pada setiap karakterketika terjadi konversi data hash biner menjadi sesuatu yang dapat dibaca (dari 4 ke 6).

8) Zend.ze1_compatibility-mode – membuat suatu model yang cocok dengan Zend Engine 1.

9) Perbaikan pada orientasi objek

a) Mendukung *private, public, dan protected.* User harus mendeklarasikannya untuk masing-masing *property* dan *method*. Hal ini berbeda dari bahasa lin (seperti C++). Jika *keyword* diabaikan mka default-nya adalah *public*.

b) Objek yang direferensi oleh *identifier* atau *resource handler* merupakan bilangan integer.

2.1.3 XAMPP

XAMPP merupakan sebuah aplikasi yang didalamnya terdapat *web server*, *script language*, dan *database*. Ketiga variabel tadi terintegrasi dalam sebuah *executable* file, ini amat memudahkan karena ketiga variabel tersebut tidak perlu dikonfigurasi secara manual, sebelum XAMPP dirilis tahun 2003 sempat ada aplikasi serupa yaitu phptriad hanya saja tidak digunakan dengan alasan sudah terlalu tua dan hanya terbatas untuk *web server* dan basis data. XAMPP mempunyai banyak sekali fitur-fitur unggulan selain *web server*, yaitu:

1) phpMyAdmin

   Adalah sebuah tools kecil berfungsi untuk menjembatani anatara PHP denagan MySQL sehingga mudah untuk dikelola, mulai dari membuat, menghapus, mengatur *account* pengguna *databse* MySQL sampai dengan mengeksekusi file sql. Selain gratis tools kecil ini juga mendukung 52 bahasa di dunia

2) Webalizer

   Sebuah tools untuk memantau atau menganalisis pergerakan file yang telah diaplikasikan sebagai *file server*. *Webalizer* ditampilkan dengan bentuk diagram batang dan statistik pergerakan file di bawahnya, webalizer juga mampu menganalisis perubahan berdasarkan hari, bulan dan menjumlahkan berapa orang yang meng akses file serta berapa kali web dikunjungi.

3) The Mercury Mail Transport System

Adalah *mail server* yang berguna untuk mengelola e-mail yang masuk ke jaringan komputer baik itu komuter *standalone* maupun jaringan komputer yang terhubung LAN. Mercury Mail juga bisa mengirim e-mail secara otomatis dan *autoforwarding*.

4) FileZilla FTP Server

Sebuah tolls yang mengatur file-file yang akan di sebar luaskan melalui internet, filezilla juga mampu membatasi bandwidth untuk tiap-tiap file yang akan diakses, file-file yang diakses pun bisa dienkripsi untuk kemanan datanya sampai dengan mengetahui siapa saja yang telah mengkases file tersebut.

2.2. Basisdata

Menurut McLeod, Basisdata adalah kumpulan seluruh sumber daya berbasis komputer milik organisasi [SP06]. Sistem manajemen basisdata adalah aplikasi perangkat lunak yang menyimpan struktur basisdata, hubungan antardata dalam basisdata, serta berbagai formulir dan laporan yang berkaitan dengan basisdata. Basisdata yang dikendalikan oleh sistem manajemen basisdata adalah satu set catatan data yang berhubungan dan saling menjelaskan.

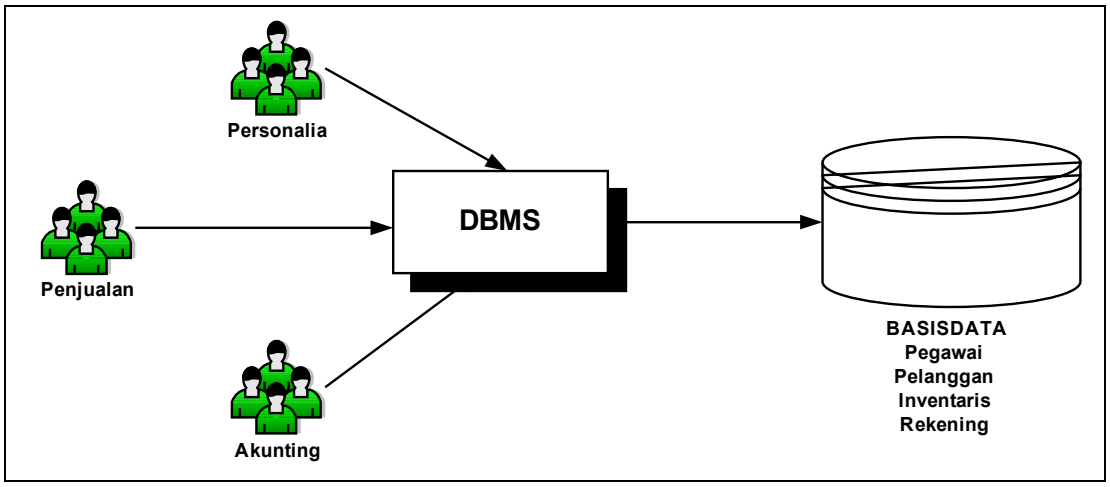
2.2.1. Sistem Basisdata dan Sistem File

Salah satu cara menyimpan informasi pada komputer adalah menyimpannya pada file-file sistem operasi. Agar pengguna dapat memanipulasi informasi, sistem operasi memiliki sejumlah progrm aplikasi untuk memanipulasi file. Sebagai contoh:

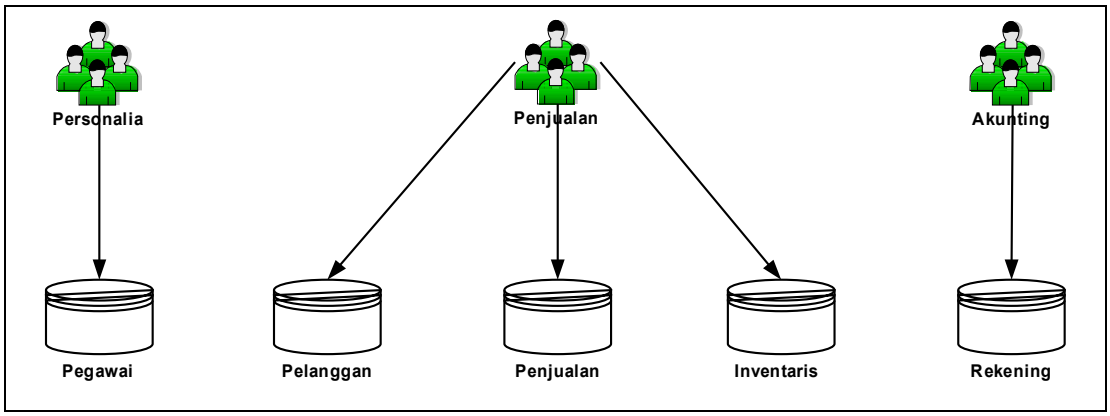
- Program untuk menambah atau mengurangi jumlah uang dalam rekening.
- Program untuk membuat rekening baru.
- Program untuk menghitung saldo rekening.
- Program untuk membuat laporan bulanan.

Program aplikasi baru ditambahkan ke dalam sistem jika dibutuhkan. Misalnya bank memutuskan untuk menawarkan pengecekan rekening. Sebagai hasilnya, bank membuat file-file baru berisi informasi semua rekening yang disimpan. Oleh karena itu, dengan berjalannya waktu, sistem membutuhkan lebih banyak file dan lebih banyak program aplikasi.

DBMS adalah bagian perangkat lunak yang dirancang untuk membuat tugas sebelumnya menjadi lebih mudah. Dengan menyimpan data dalam DBMS daripada sebagai kumpulan file sistem operasi, user dapat menggunakan fitur DBMS untuk mengelola data secara efisien. Karena volume data dan pengguna terus bertambah, ratusan *gigabyte* data dan ribuan pengguna merupakan hal biasa dalam basisdata perusahaan saat ini sehingga dukungan DBMS sangat diperlukan [SP06].



Gambar 2.1 Sistem Basisdata [Sumber:Janner Simarmata, Iman Prayudhi,"Basis Data"]



Gambar 2.2 Sistem File [Sumber:Janner Simarmata, Iman Prayudhi,"Basis Data"]

Sistem pemrosesan file didukung oleh sistem operasi konvensional. Sistem file menyimpan record-record bermacam-macam file, membutuhkan program aplikasi yang berbeda untuk memilah record suatu file, dan menambahkan rcord ke dalam suatu file. Sebelum sistem manajemen basisdata muncul, perusahaan-perusahaan menyimpan informasi dalam sistem file memiliki sejumlah kelemahan [SP06], yaitu:

- Pengulangan dan ketidakkonsistenan data

- Kesulitan mengkases data
- Masalah integritas
- Masalah atomik
- Anomali akses bersamaan
- Masalah keamanan

2.2.2. Sistem Manajemen Basisdata (DBMS)

Sistem Manajemen Basisdata (DBMS) adalah pernagkat lunak yang didesain untuk membantu memelihara dan memanfaatkan kumpulan data yang besar [SP06]. Kebutuhan aka sistem termasuk pula penggunaannyayang berkembang pesat. Alternatif penggunaan DBMS adalah menyimpan data dalam file dan menulis kode aplikasi tertentu untuk mengaturnya. Penggunaan DBMS memiliki beberapa manfaat penting.

2.2.2.1. Keuntungan DBMS

DBMS memungkinkan perusahaan maupun pengguna individu untuk:

- Mengurangi pengulangan data
- Mencapai independensi data
- Mengintegrasikan data beberapa file
- Mengambil data dan informasi dengan cepat
- Meningkatkan keamanan

2.2.2.2. Kerugian DBMS

Keputusan menggunakan DBMS mengikat perusahaan atau pengguna untuk:

- Memperoleh perangkat lunak yang mahal
- Memperoleh konfigurasi perangkat keras yang besar
- Memperkerjakan dan mempertahankan staf DBA

Baik basisdata terkomputerisasi maupun DBS bukanlah prasyarat untuk memecahkan masalah. Namun, keduanya memberikan dasar-dasar menggunakan komputer sebagai suatu sistem informasi bagi para spesialis informasi dan pengguna.

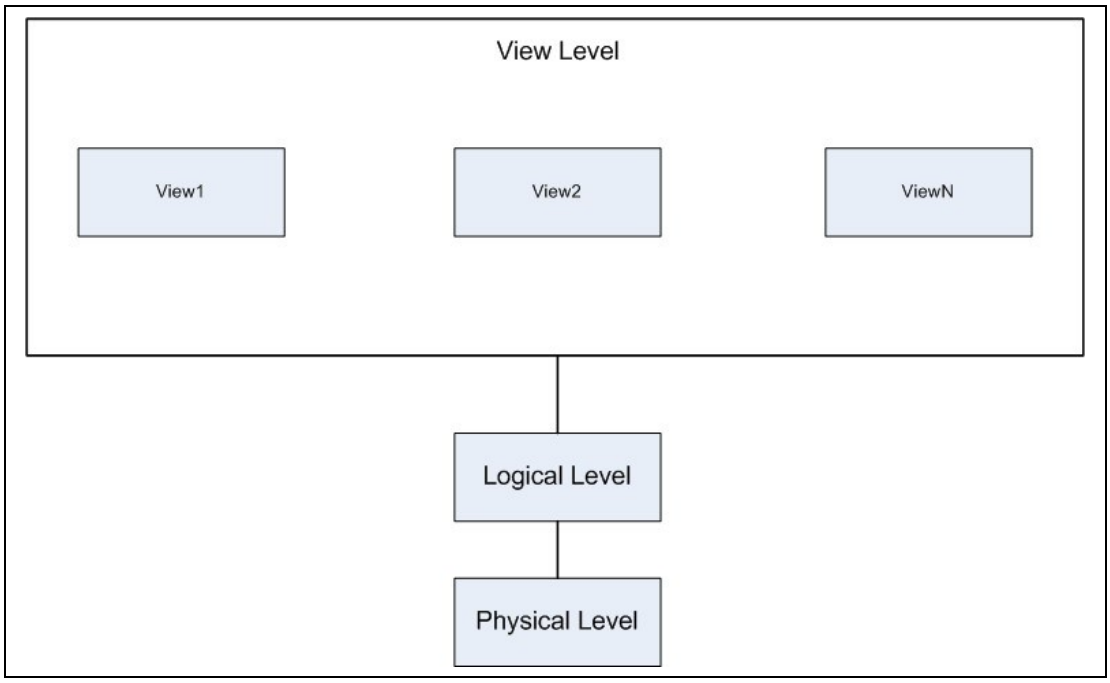
2.2.3. Pandangan Data

Sistem basisdata adalah kumpulan file yang saling berhubungan dan kumpulan program yang memungkinkan pengguna mengakses dan mengubah file-file. Tujuan utama sistem basisdata adalah memberi pengguna pandangan abstrak suatu data. Pandangan abstrak berarti sistem menyembunyikan cara data disimpan dan dirawat.

2.2.3.1. Abstraksi Data

Supaya suatu sistem berguna, maka sistem itu harus bisa mengambil data secara efisien. Kebutuhan akan efisiensi ini mendorong programmer untuk menggunakan struktur dan kompleks untuk menyajikan data dalam basisdata. Karena banyak pengguna sistem basisdata tidak mahir menggunakan komputer, pengembang *(developer)* menyembunyikan kompleksistas sistemnya dari pengguna melalui beberapa tingkatan abstraksi. Tingkatan abstraksi berguna untuk menyederhanakan interaksi pengguna dengan sistem.

1) Tingakatan Fisik, merupakan tingkatan abstraksi paling rendah dan menggambarkan bagaimana data sebernar nya disimpan. Tingkatan fisik menggambarkan secara rinci struktur data tingkat rendah yang kompleks.
2) Tingkatan Logis, merupakan tingkatan yang setingkat lebih tinggi dari tingkatan fisik dang menggambarkan data yang disimpan dalam basisdata serta hubungan antardata. Tingkatan logis menggmbarkan basisdata secara keseluruhan melalui sejumlah struktur sederhana. Meskipun penerapan struktur pada tingakatan logis bisa mengikutsertakan struktur tingkatan fisik yang kompleks, pengguna pengguna tingakatan logis tidak harus memperhatikan kompleksitas tingkatan fisik. Administrator basisdata yang harus memutuskan informasi yang disimpan dalam basisdata, menggunakan abstraksi tingkatan yang logis.
3) Tingkatan View, merupakan tingkatan abstraksi paling tinggi dan hanya menggambarkan bagian tertentu suatau basisdata. Meskipun tingkatan logis menggunakan struktur lebih sederhana, tetapi masih kompleks karena beragamnya informasi yang disimpan dalam basisdata. Banyak pengguna sistem basisdata tidak memerlukan informasi tersebut hanya karena hanya perlu mengakses bagian dari basusdata. Kegunaan abstraksi tingkatan view adalah menyederhanakan interaksi dengan sistem. Sistem bisa meyediakan banyak view untuk basisdata yang sama.



Gambar 2.3 Hubungan Abstraksi Data [Sumber:Janner Simarmata, Iman Prayudhi,"Basis Data"]

Analogi dengan konsep tipe data pada bahasa pemrograman bisa memperjelas perbedaaan ketiga tingkatan abstraksi. Sebagian besar bahasa pemrograman tingkat tinggi mendukung gagasna tipe. Misalnya, pada bahasa seperti pascal, bisa mendeklarasikan sebuah record sebagai berikut:

```
Type mahasiswa = record
                    nim:stiring;
                    nama:string;
                    alamat:string;
                    kota:string;
                    end;
```

Program di atas mendefinisikan record baru yang diberi nama mahasiswa dengan empat field. Masing-masing field memiliki sebuah nama dan sebuah tipe yang berhubungan dengannya. Sebuah universitas bisa memiliki beberapa tipe record seperti itu, antara lain:

- *Mata kuliah*, dengan field KodeMK, NamaMK, SKS.
- *Dosen*, dengan field NIP, Nama, Alamat, Kota

Pada tingkatan fisik, sebuah record pada mahasiswa, mata kuliah, dan dosen dapat digambarkan sebagai sebuah blok lokasi penyimpanan yang berurutan misalnya word atau byte. Kompiler menyembunyikan rincian tingkatan fisik dari programmer. Demikian pula, sistem basisdata menyembunyikan banyak rincian penyimpanan tingkatan paling rendah dari programmer basisdata. Di lain pihak, administrator basisdata mungkin mengetahui rincian tertentu pengaturan data.

Pada tingkatan logis, masing-masing record digambarkan oleh sebuah definisi tipe (seperti dituliskan pada kode program di atas) dan hubungan tipe-tipe record didefinisikan pula. Programmer menggunakan bahasa pemrograman pada tingkatan abstrkasi logis. Kemudian , administrator basisdata sering pula bekerja pada tingkatan abstraksi ini.

Pada tingkatan view, pengguna komputer melihat kumpulan program aplikasi yang menyembunyikan rincian tipe-tipe data. Lebih lanjut, pada tingkatan ini, beberapa view basisdata didefinisikan pula dan pengguna basisdata melihatnya. Selain menyembunyikan rincian tingkatan logis suatu basisdata , tingakatan view menyediakan mekanisme keamanan untuk mencegah pengguna mengakses bagian trtentu basisdata. Misalnya pegawai administrasi tidak boleh melihat rincian dan mahasiswa dan hanya boleh melihat data pegawai.

2.2.3.2. Instansi dan Skema

Basisdata berubah dari waktu ke waktu ketika informasi dimasukkan dan dihapus. Kumpulan informasi yang disimpan dalam basisdata pada suatu waktu disebut instansi *(instance)* suatu basisdata. Rancangan basisdata secara keseluruhan disebut skema basisdata.

Konsep instansi dan skema basisdata dapat dimengerti dengan menganalogikan dengan sebuah program yang ditulis dalam suatu bahasa pemrograman. Skema basisdata berhubungan dengan deklarasi variabel maupun definisi tipe sebuah program. Masing-masing variabel memilikinilai tertentu pada suatu saat. Nilai variabel dalam prgram pada suatu saat berhubungan dengan instansi skema basisdata.

Sistem basisdata memiliki beberapa akema yang dibagi menurut tingkatan abstarksi. Skema fisik menggambarkan rancangan basisdata pada tingkatan fisik, sedangkan skema logis menggambarkan rancangan basisdata pada tingkatan logis. Sebuah basisdata bisa memiliki beberapa skema pada tingkatan view, yang kadang-kadang disebut subskema dan menggambarkan view yang berbeda dalam basisdata.

2.2.4. Struktur Sistem Basisdata

Sistem basisdata dibagi menjadi modul-modul yang berhubungan dengan masing-masing tanggungjawab sistem secara keseluruhan. Komponen fungsional sistem basisdata dapat dibagi menjadi komponen pengatur penyimpanan dan komponen pemroses query.

2.2.4.1. Pengatur Penyimpanan

Pengatur penyimpanan adalah modul program yang menyediakan anatrmuka antar dua tingkat randah yang disimpan pada basisdata dengn program aplikasi dan query yang di kirimkan ke sistem. Pengatur penyimpanan bertanggungjawab terhadap interaksi dengan pengatur file. Data mentah disimpan pada disk menggunkan sistem file yang disediakan oleh sistem operasi konvensional. Pengatur penyimpanan menerjemahkan berbagai pernyataan DML menjadi perintah sistem file tingkat rendah. Oleh karena itu, pengatur penyimpanan bertanggungjawab terhadap penyimpanan, pengambilan dan perubahan data dalam basisdata. Komponen pengatur penyimpanan sebagai berikut:

1) Pengatur otorisasi dan integritas, menguji batasan integritas dan mengecek otoritas pengguna untuk mengakases data.
2) Pengatur transaksi, memastikan bahwa basisdata tetap konsisten meskipun ada kegagalan pada sistem dan eksekusi transaksi yang terjadi bersamaan berlangsung tanpa terjadi konflik.
3) Pengatur file, mengatur alokasi ruang pada disk dan struktur data yang digunakan untuk mewakili informasi yang disimpan pada disk.
4) Pengatur penyimpanan sementara, bertanggungjawab mengambil data dari disk, memasukannya ke memori utama, dan memutuskan data yang akan disimpan sementara dalam memori utama. Pengatur penyimpanan sementara adalah bagian penting sistem basisdata karena ini memungkinkan basisdata menangani data berukuran jauh lebih besar dari ukuran memori utama

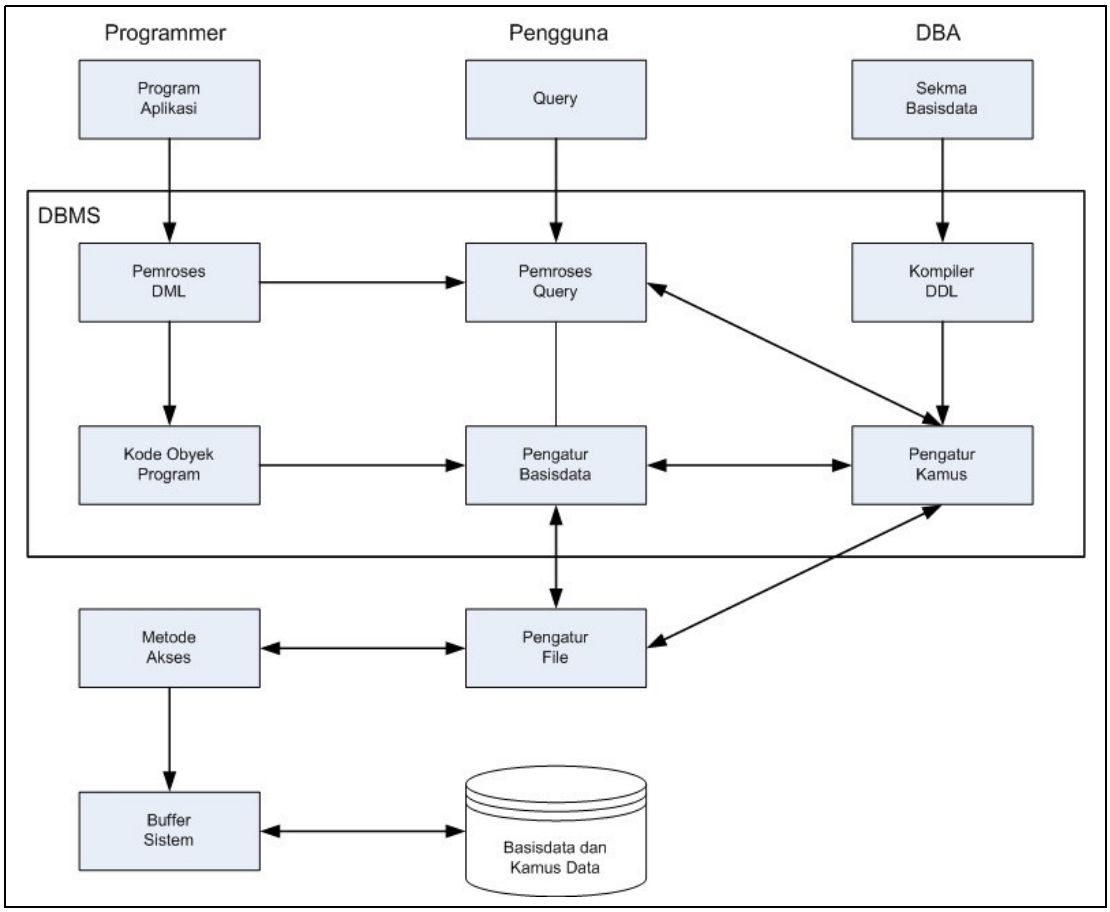
Pengatur penyimpanan menerapkan beberapa struktur data sebgai bagian penerapan sistem fisik, yaitu:

- File data, menyimpan basisdata sendiri
- Kamus data, menyimpan metadata tentang struktur
- Indeks, menyediakan akses cepat ke data yang memiliki nilai tertentu.

2.2.4.2. Pemrosesan Query

Komponen pemroses query meliputi:

1) Intepreter DDL, menterjemahkan perintah DDL dan mencatat definisi-definisi dalam kamus data.
2) Compiler DML, menterjemahkan perintah DML dalam bahasa query menjadi rencana evaluasi yang terdiri atas instruksi tingkat rendah yang dimengerti oleh mesin evaluasi query.
3) Mesin evalusi query, mengeksekusi instruksi tingkat rendah yang dihasilkan oleh kompiler DML



Gambar 2.4 Struktur sistem [Sumber:Janner Simarmata, Iman Prayudhi,"Basis Data"]

2.2.5. Bahasa Basisdata

Sistem basisdata menyediakan bahasa pendefinisian data (*Data Definition Language*-DDL) unutk menentukan skema basisdata dan bahasa manipulasi data (*Data Manipulation Language*-DML) untuk menyatakan query dan update basisdata. Pada praktiknya, DDL dan DML bukan merupakan dua bahasa yang terpisah melainkan membentuk bagian bahasa basisdata, seperti yang umum digunakan pada bahasa SQL.

2.2.5.1. Data Definition Language (DDL)

Skema basisdata ditentukan sekumpulan definisi yang dinyatakan dengn bahasa tertentu yang disebut Data Definition Language (DDL). Sebagai contoh, pernyataan berikut dalam bahasa QL mendefinisikan tabel rekening.

```
create table rekening
        (no_rekening char(10),
        saldo integer)
```

Eksekusi pernyataan DDL di atas akan membuat tabel rekening. Sebagai tambahan, pernyataan itu menambah kumpulan tabel yang disebut kamus data atau direktori data. Kamus data adalah istilah basisdata yang mengacu pada definsi data yang disimpan dalam basisdata dan dikendalikan oleh sistem manajemen basisdata [JI06].

Kamus data mengandung metadata, yaitu data tentang data. Skema sebuah tabel adalah contoh sebuah metadata. Sistem basisdata akan mengecek kamus data sebelum membaca atau memodifikasi data sebenarnya. Struktur penyimpanan dan metode akses yang digunkan oleh sistem basisdata dengan menggunkan sekumpulan pernyataan dalam tipe DDL tertentu disebut bahasa penyimpanan dan definisi data. Pernyataan ini menetukan penerapan detail skema basisdata, yang sering disembunyikan dari pengguna.

2.2.5.2 Data Manipulation Language

Bahasa manipulasi data (DML) adalah bahasa yang memungkinkan pengguna mengakses atau memanipulasi data seperti yang diatur oleh model data. Ada dua tipe DML, yaitu:

- DML Prosedural, menharuskan pengguna untuk menentukan data yang dibutuhkan dan bagaimana mendapatkannya.
- DML Deklaratif (DML nonprosedural), mengharuskan pengguna menentukan data yang dibutuhkan tanpa menentukan bagaimana mendapatkannya.

DML deklaratif lebih mudah dipelajari dan digunakan daripada DML prosedural. Namun karena pengguna tidak harus menentukan cara mendapatkan data, sistem basisdata harus mencari alat yang efektif untuk mengakses data. Komponen DML dari bahasa SQL adalah nonprosedural.

Query adalah pernyataan yang meminta pengguna mengambil informasi. Bagian DML yang terlibat dalam pengambilan informasi disebut bahasa query. Istilah bahasa query sering disamakan dengan istilah bahasa manipulasi data.

2.2.6. Tipe-tipe Basisdata

Langkah-langkah dasar merancang basisdata adalah keputusan perusahaan mengalokasikan sumberdaya untuk mempelajari basiadata, merancang struktur basisdata, dan menerapkan serta mengatur basisdata. Setelah basisdata dirancang dan aplikasi telah dibuat atau dibeli, manajeman basisdata secara keseluruhan menjadi sederhana untuk basisdata administrator basisdata yang berpengalaman. Setelah perusahaan memutuskan unutk merancang sistem basisdata, model basisdata yang akan digunakan harus dibuat. Ada beberapa tipe basisdata yaitu:

1. Model basisdata file (*Flat file database model*)
2. Model basisdata hierarki (*Hierarchial database model*)
3. Model basisdata jaraingan (*Network database model*)

4. Model basisdata relasional (*Relational database model*)
5. Model basisdata berorientasi objek (*Object oriented database model*)
6. Model basisdata relasi objek (*Object relational database model*)

2.2.6.1 Model Basisdata File

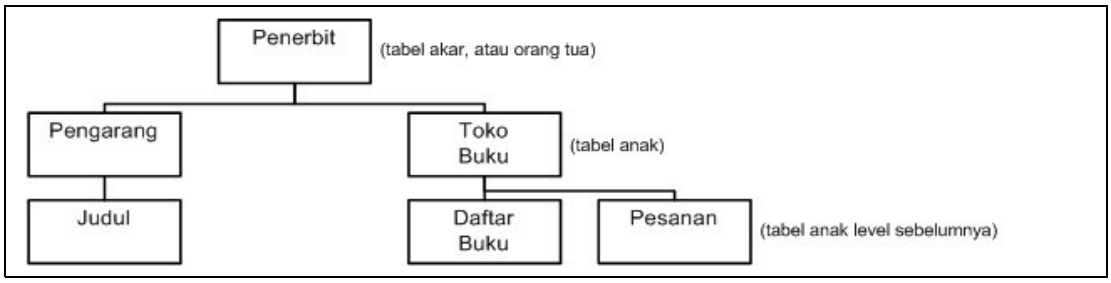
Sebelum Oracle dan Microsoft mengembangkan sistem manajemen basisdata, banyak perusahaan yang menggunakan komputer untuk menyimpan datanya dalam bentuk file. Penggunaan file untuk menyimpan data sangat umum pada masa mainframe. Basisdata file tersusun atas satu atau lebih file dan disimpan dalam format teks. Informasi dalam file-file teks disimpan sebgai field. Field-field bisa memiliki panjang yang tetap atau berubah-ubah dan dipisahkan oleh beberpa pemisah

Masalah pengaksesan data membutuhkan sekumpulan program yang dapat mengakses informasi yang disimpan dalam file. Salah satu masalah utama menggunakan sistem basisdata file bukan hanya memerlukan pengetahuna tentang struktur file, malainkan harus mengetahui pasti di mana data disimpan. Berikut adalah kekuranagn basisdata file:

- File tidak menyediakan struktur di mana data mudah dihubungkan
- Mengatur data secara efektif dan memastikan akurasi merupakan hal sulit
- Pengguna perlu menyimpan data secara berulang sehingga menyebabkan lebih banyak pekerjaan untuk merawat data secara akurat.
- Lokasi fisik field data pada file harus diketahui.
- Program harus dibuat untuk mengatur data.

2.2.6.2. Model Basisdata Hierarki

Model basisdata hierarki setingkat di atas basisdata file terutama karena kemampuan membuat dan merawat hubungan antar kelompok data. Arsitektur basisdata hierarki berdasarkan pada konsep hubungan orang tua dan anak. Pada basisdata hierarki, tabel akar atau tabel orang tua pada puncak struktur dan menunjuk pada tabel anak yang mengandung data yang berhubungan.



Gambar 2.5 Model Basisdata Hierarki [Sumber:Janner Simarmata, Iman Prayudhi,"Basis Data"]

Pada gambar diatas, penerbit adalah tabel akar. Penerbit memiliki dua tabel anak, yaitu Pengarang dan Toko Buku. Sebuah penerbit memiliki banyak Pengarang yang dikontraknya serta banyak Toko Buku di mana dia menyuplai buku. Pengarang adalah tabel orantua untuk Tabel Judul, seperti Toko Buku untuk Tabel Daftar Buku. Judul adalah tabel anak dari Pengarang, seperti Daftar Buku dan Pesanan yang merupakan tabel anak dari Toko Buku. Salah satu masalah menggunkan hierarki adalah informasi judul buku yang berulang karena harus disimpan pada Tabel Daftar Buku. Hal ini dikarenakan tidak ada hubungan langsung antara Pengarang dan Toko Buku.

Tabel orangtua dapat memiliki banyak tabel anak, tetapi tabel anak hanya dapat memiliki satu tabel orangtua. Untuk mengakses tabel anak, harus mengakses tabel orangtua terlebih dahulu. Tabel-tabel yang berhubungan dalam tabel hierarki dihubungkan dengan pointer, yang menunjuk ke lokasi fisik record anak.

Keuntungan model basisdata hierarki dibanding model basisdata file adalah:

- Data dapat diambil secara cepat.
- Integritas data lebih mudah diatur

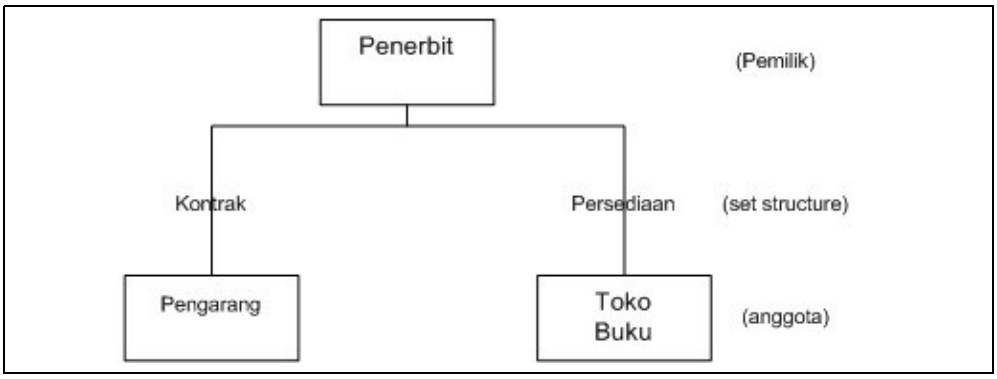
Kekurangan model basisdata hierarki adalah:

- Pengguna harus memahami struktur basisdatanya
- Terjadi perulangan data

2.2.6.3. Model Basisdata Jaringan

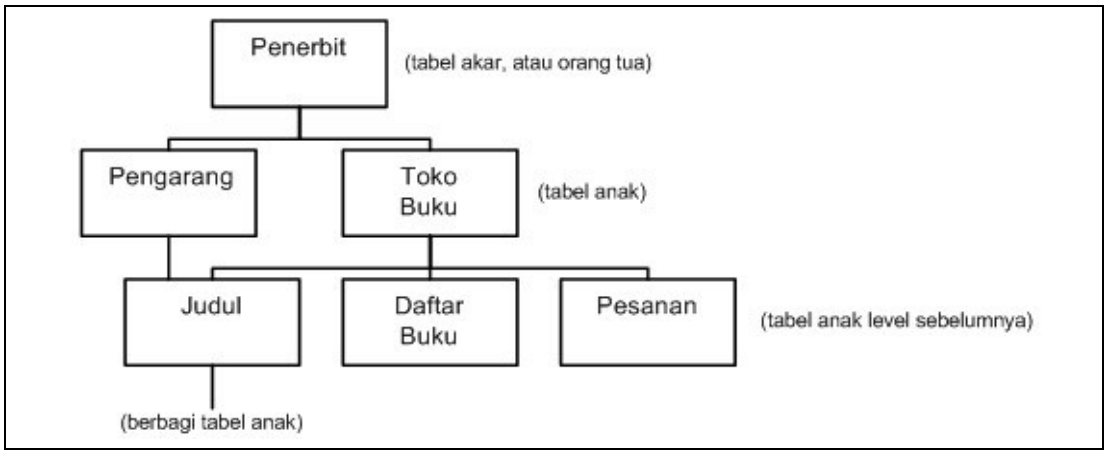
Seperti pada basisdata hierarki, tabel-tabel basisdata jaringan berhubungan satu sama lain. Satu keuntungan utama basisdata jaringan adalah kemampuan tabel orangtua untuk berbagi hubungan dengan tabel anak.

Hubungan antar tabel pada basisdata jaringan disebut *set structure*, di mana satu tabel adalah pemilik dari tabel lainnya adalah anggota. *Set Structure* dapat mewakili hubungan one-to-many antara tabel-tabel. Program aplikasi yang mengakses basisdata jaringan menggunakan *set structure* untuk mengarahkan tabel ke bagian lain basisdata. Oleh karena itu, jika *set structure* diubah, program aplikasi yang mengakses basisdata harus diubah pula.



Gambar 2.6 Model Basisdata Jaringan [Sumber:Janner Simarmata, Iman Prayudhi,"Basis Data"]

Pada gambar diatas tabel penerbit memiliki dua table, yaitu pengarang dan toko buku. Pengarang dan toko buku masing-masing adalah anggota tabel penerbit. Penerbit mengontrak pengarang dan menyuplai buku ke toko buku. Set structure antara pengarang dan penerbit disebut kontrak.



Gambar 2.7 Berbagi Tabel Anak [Sumber:Janner Simarmata, Iman Prayudhi,"Basis Data"]

Pada gambar diatas menunjukkan bagaimana tabel anak atau anggota dapat dimiliki lebih dari satu tabel orangtua. Tabel judul dimiliki oleh tabel pengarang dan toko buku, baik tabel pengarang maupun toko buku membutuhkan hubungan dengan

tabel judul. Meskipun kedua set structure dapat digunakan untuk mengakases tabel judul, informasi judul buku hanya disimpan pada satu tabel sehingga mengurangi perulangan data.

Keuntungan model basisdata jaringan adalah:

- Data dapat diakses dengan cepat
- Pengguna dapat mengakses data mulai dari sembarang tabel
- Pengguna lebih mudah membuat model basisdata yang lebih kompleks.
- Pengguna lebih mudah membuat query yang lebih kompleks untuk mengambil data.

Kekurangan model basisdata jaringan adalah:

- Struktur basisdata tidak mudah diubah
- Perubahan struktur basisdata mempengaruhi program aplikasi yang mengakses basisdata.
- Pengguna harus memahami struktur basisdata.

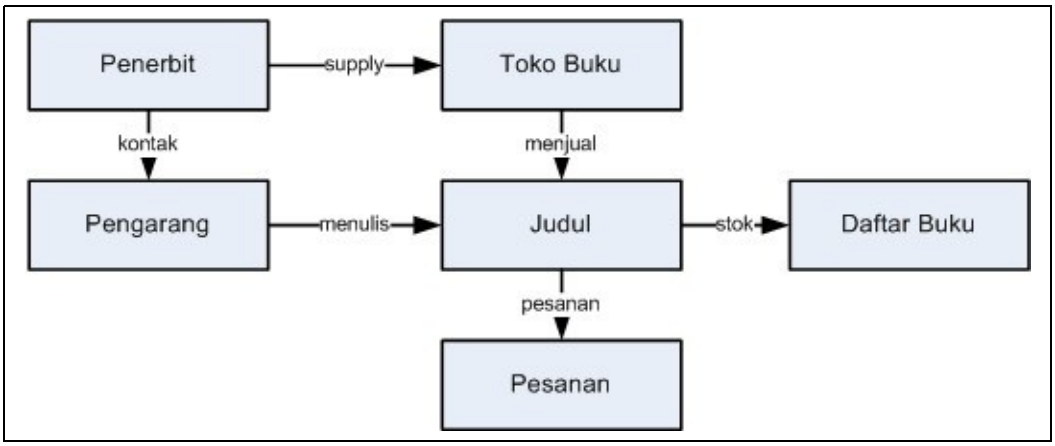
2.2.6.4. Model Basisdata Relasional

Basisdata relasional adalha tipe basisdata yang paling populer digunakan saat ini. Banyak perkembangan yang telah dibuat dari model baisdta sbelumnya menyederhanakan manajemen data dan pengambilan data. Data menjadi lebih mudah diatur, terutama melalui penggunaan batasan integritas. Pengambilan data merupakan proses penyaringan yang memungkinkan pengguna memvisualisasikan basisdata

melalui struktur tabel relasional dan meminta data tertentu tanpa perlu memahami struktur basisdata.

Unit penyimpanan utama dalam basisdata adalah tabel atau kelompok data yang saling berhubungan. Sebuah tabel terdiri atas baris dan kolom. Baris berhubungan dengan record dalam tabel dan kolom mengandung nilai semua baris yang berhubungan dengan field tetentu. Tabel dapat dihubungkan satu sama lain melalui nilai kolom yang disebut kunci (*key*).

Ada tiga tipe hubungkan antar tabel, yaitu one-to-one, one-to-many, dan many-to-many. Suatu basisdata harus memiliki lebih dari satu hubungan berbeda. Hubungan antar tabel ditentukan oleh integritas referensial (*referentila integrity*) yang memerlukan penggunaan batasan kunci utama (*primary key*) dan kunci tamu (*foreign key*). Integritas referensial adalah penggunaan batasan di atas untuk membuat valid data yang dimasukkan ke dalam tabel dan mengatur hubungan antara tabel orang tua dan anak. Tipe batasan lain dapat pula dibuat untuk mengontrol data yang bisa dimasukkan dalam kolom tertentu dan membuat hubungan antar tabel.



Gambar 2.8 Model Basisdata Relasional [Sumber:Janner Simarmata, Iman Prayudhi,"Basis Data"]

Pada gambar diatas model relasional tidak memiliki tabel akar meskipun hubungan antara tabel anak dan orang tua masih dimungkinkan. Tabel orang tua dapat memiliki banyak tabel anak, demikian pula, tabel anak dapat memilih banyak tabel orang tua (hubungan dua arah).

Keuntungan model basisdata relasional adalah:

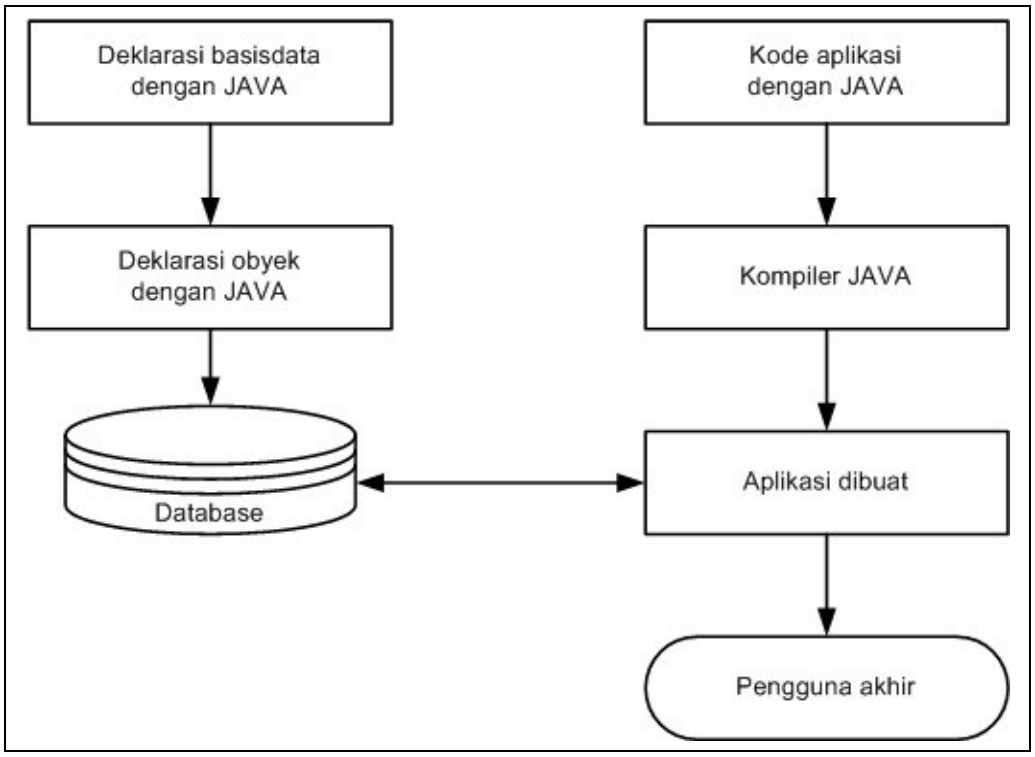
- Data dapat diakses secara cepat
- Struktur basisdata mudah diubah.
- Data disajikan secara logis sehingga pengguna tidak perlu mengetahui bagaimana data tersimpan
- Pengguna mudah membuat dan memodifikasi program aplikasi.
- Bahasa standar SQL sudah dibuat

Kekurangan model basisdata relasional adalah:

- Kelompok informasi atau tabel yang berbeda harus dihubungkan untuk mengambil data.
- Pengguna harus memahami hubungan antar tabel.
- Pengguna harus belajar SQL.

2.2.6.5. Model Basisdata Berorientasi Objek

Basisdata berorientasi objek adalah basisdata di mana data dapat ditentuan, disimpan dan diakses menggunakan pendekatan pemrograman berorientasi objek. Basisdata berorientasi objek menggunakan bahasa pemrograman berorientasi objek untuk menentukan struktur basisdata dan membuat aplikasi yang digunakan untuk berinteraksi dengan basisdata.



Gambar 2.9 Model Basisdata Berorientasi Objek dan Penerapannya [Sumber:Janner Simarmata, Iman Prayudhi,"Basis Data"]

Gambar diatas menunjukkan bahwa basisdata berorientasi objek ditentukan menggunakan bahasa pemrograman berorientasi objek seperti Java. Aplikasi untuk pengguna akhir (end user) dibuat pula dengan menggunakan bahasa pemrograman berorientasi objek. Sistem manajemen basisdata objek digunakan untuk menghubungkan basisdata dengan aplikasi.

Keuntungan model basisdata berorientasi objek adalah:

- Programmer hanya perlu mengerti konsep berorientasi objek.
- Objek dapat mewariskan properti ke objek lainnya.
- Banyak proses dalam program aplikasi terjadi secara otomatis.
- Secara teoritis, pengguna lebih mudah menangani objek.
- Model data berorientasi objek lebih kompatibel dengan bahasa pemrograman berorientasi objek.

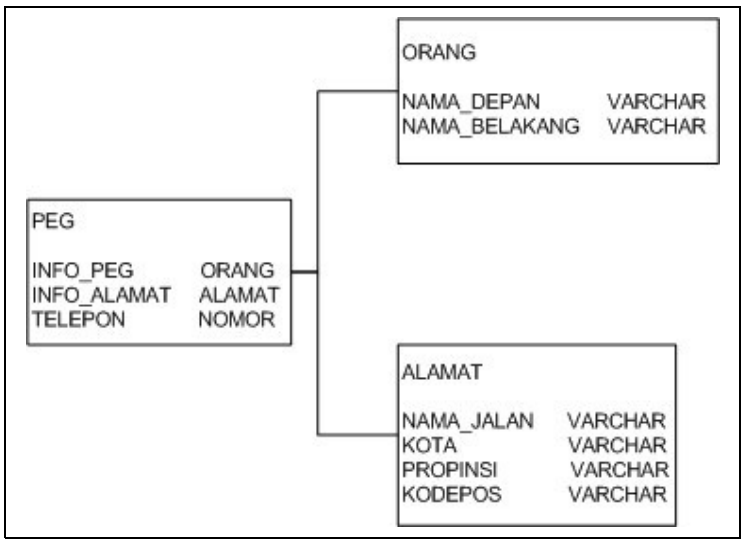
Kekurangan model basisdata berorientasi objek adalah:

- Pengguna harus belajar konsep berorientasi objek karena basisdata berorientasi objek tidak bekerja dengan metode pemrograman tradisional.
- Model belum memiliki standar.
- Stabilitas model basisdata berorientasi objek belum terjamin karena model masih baru.

2.2.6.6. Model Basisdata Relasional Objek

Model basisdata relasional objek dibuat dengan tujuan menggabungkan konsep model basisdata relasional dengan pemrograman berorientasi objek. Model

relasional objek dianggap mewakili yang terbaik dari kedua pihak (*relasional* dan berorientasi objek) meskipun model relasional objek masih dalam tahap awal perkembangan.



Gambar 2.10 Data dalam Basisdata Berorientasi Objek dan Penerapannya
[Sumber:Janner Simarmata, Iman Prayudhi,"Basis Data"]

Pada gambar diatas menggambarkan contoh penerapan relasional objek dalam sistem manajemen basisdata relasional (RDBMS) Oracle 8. Dua tipe yang ditentukan oleh pengguna (*user defined*) telah dibuat yaitu ORANG dan ALAMAT. Masing-masing tipe memiliki kolom yang menentukan data tertentu untuk sebuah kolom dalam tabel dasar sehingga menyebabkan terjadinya efek 3D pada data. Sebagai contoh, kolom INFO_PEG pada tabel PEG memiliki sebuah tipe ORANG. ORANG dibagi menjadi beberapa kategori, yaitu NAMA_DEPAN dan NAMA_BELAKANG.

Keuntungan Model Relasional objek adalah:

- Basisdata relasional memiliki lebih banyak arsitektur 3D.

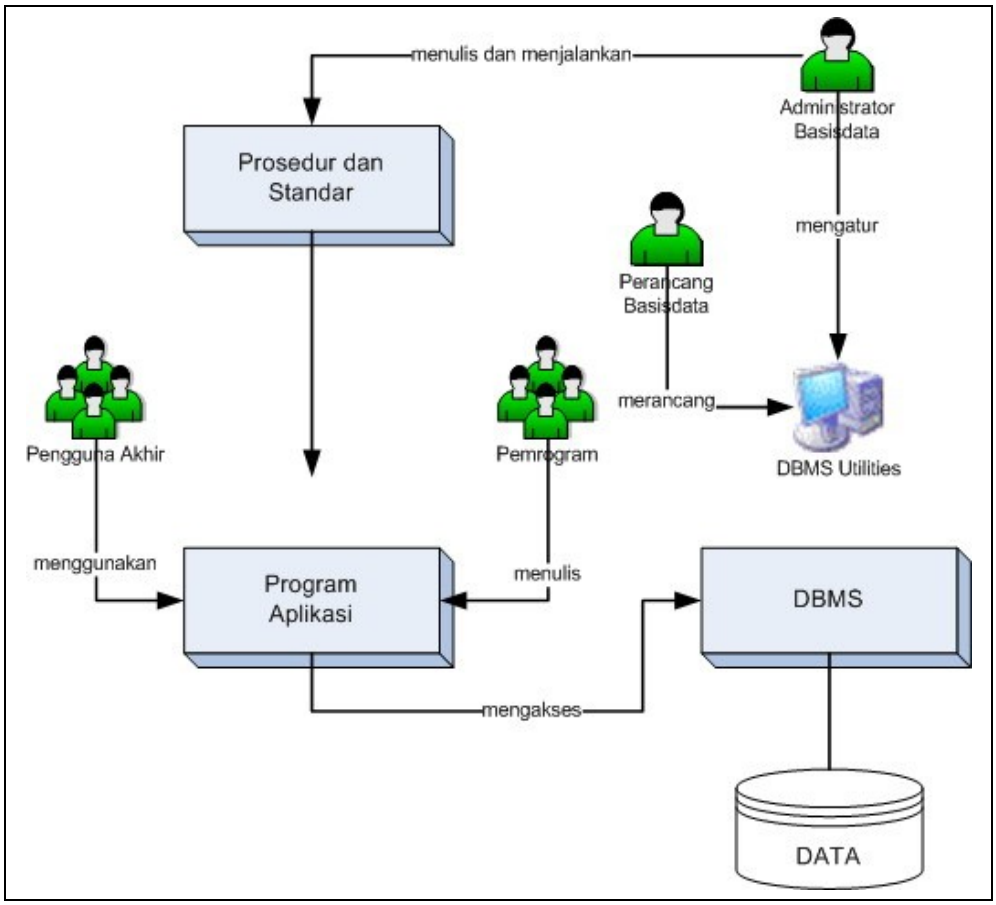
- Tipe yang ditentukan oleh pengguna dapat dibuat
- Tipe yang ditentukan oleh pengguna dapat dibuat.

Kekurangan model relasional objek adalah:

- Pengguna harus mengerti, baik konsep relasional maupun berorientasi objek.
- Beberapa penjual yang telah menerapkan konsep relasional objek tidak mendukung pewarisan objek (*object inheritance*).

2.2.7. Lingkungan Basisdata

Lingkungan basisdata adalah sebuah habitat dimana terdapat basisdata untuk bisnis. Dalam lingkungan basisdata, pengguna memiliki alat untuk mengakses data. Pengguna bisa datang dari dalam lingkungan basisdata atau dari luar lingkungan. Pengguna melakukan semua tipe pekerjaan dan keperluan mereka bervariasi seperti menggali data (*mining for data*), memodifikasi data, atau berusaha membuat data baru. Ada tiga macam lingkungan basisdata yaitu: Lingkungan Mainframe, Lingkungan Client-Server, dan Lingkungan Internet.



Gambar 2.11 Lingkungan Basisdata[Sumber:Janner Simarmata, Iman Prayudhi,"Basis Data"]

2.2.7.1. Lingkungan Mainframe

Lingkungan mainframe terutama terdiri atas komputer mainframe yang memungkinkan koneksi banyak pengguna. *Dumb terminal* dihubungkan ke komputer mainframe agar pengguna dapat berkomunikasi dengan mainframe. Pada dasarnya terminal adalah perpanjangan mainframe, terminal bukan komputer independen. Istilah *dumb terminal* mengisyaratkan bahwa terminal tidak bisa berpikir. Terminal tergantung pada komputer mainframe untuk melakukan semua pemrosesan. Salah satu masalah utama dalam lingkungan mainframe adalah keterbatasan penggunanya.

2.2.7.2. Lingkungan Client-Server

Lingkungan client-server terdiri atas sebuah komputer utama, yang disebut server, dan satu atau lebih PC yang dihubungkan ke server. Basisdata terletak pada server, yang merupakan bagian terpisah dari PC. Masing-masing pengguna yang ingin mengakses basisdata pada server harus memiliki PC sendiri. Karena PC adalah sistem komputer terpisah, palikasi dibuat dan diinstal pada PC di mana melalui aplikasi itulah pengguna dapat mengakses basisdata pada server. Aplikasi pada client mengirimkan permintaan data atau transaksi melalui jaringan secara langsung ke basisdata host server. Informasi dikirimkan melalui jaringan ke basisdata menggunakan *open database connectivity* (ODBC) atau software jaringan lainnya. Salah satu masalah dalam lingkungan client-server adalah ketika versi baru aplikasi dibuat, aplikasi harus diinstal dan dikonfigurasi ulang pada tiap-tiap komputer client.

2.2.7.3. Lingkungan Internet

Pada lingkungan internet, aplikasi hanya perlu diinstal pada satu server yang disebut *web server* . Pengguna harus memiliki koneksi internet dan web browser pendukung yang diinstal pada PC. *Web browser* digunakan untuk berhubungan dengan URL tujuan *web server*. Selanjutnya web server mengakses basisdata dibantu oleh aplikasi dan mengembalikan informasi yang diminta pada *web browser* pengguna. *Web browser* menampilkan hasilnya pada PC pengguna. Setup dan perawatan aplikasi disederhanakan paada lingkungan internet karena tidak ada yang harus diinstal, dikonfigurasi, atau dirawat pada PC pengguna dengan tujuan mengurangi resiko ketidakonsistenan konfigurasi dan ketidakcocokan versi perangkat lunak antara komputer client dan server.

2.3. MySQL

Database MySQL merupakan sistem manajemen basisdata SQL yang sangat terkenal dan bersifat open source. MySQL dibangun, didistribusikan, dan didukung oleh MySQL AB. MySQL AB merupakan perusahaan komersial yang dibiayai oleh pengembang MySQL. MySQL mempunyai dua macam lisensi yaitu lisensi yang bersifat open source dengan menggunakan GNU (*General Public License*) dan lisensi kedua berupa *Standard Commercial License* yang dapat dibeli dari MySQL AB.

1) MySQL merupakan sistem manajemen database. Database merupakan struktur dari penyimpanan data. Untuk menambah, mengakses, dan memproses data yang disimpan dalam sebuah database komputer maka diperlukan sistem manajemen database sepeti MySQL Server.
2) MySQL merupakan sistem manajemen database terhubung (*relational database management system*). Database terhubung menyimpan data padfa tabel-tabel terpisah. Hal tersebut akan menambah kecepatan dan fleksibilitasnya. Kata SQL pada MySQL merupakan singkatan dari *Structured Query Language*. SQL merupakan bahasa standar yang digunakan untuk mengakses database dan ditetpakan oleh ANSI/ISO SQL Standard.
3) MySQL merupakan software open source. Open source berartisemua orang diijinkan untuk menggunakan dan memodifikasinya. Semua orang dapat men *download software* MySQL dari internet dan menggunakannya tanpa harus membayar.

4) MySQL database server mempunyai kecepatan akses yang tinggi, mudah digunakan, serta handal. MySQL dikembangkan untuk menangani database yang besar secara cepat dan telah sukses digunakan selama bertahun-tahun. Karena konektivitas, kecepatan, dan keamanannya, MySQL server cocok untuk mengakses database di internet.
5) MySQL bekerja di Client/Server atau sistem embedded. Software database MySQL merupakan sistem client/server yang terdiri dari *multithread* SQL server yang mendukung software klien dan library yang berbeda, administrative tools, dan sejumlah *Application Programming Intrfaces* (APIs).
6) MySQL tersedia dalam beberapa macam bahasa.

Fitur utama dari MySQL adalah:

1) Bekerja dalam berbagai platform, misalnya Mac OS X, Solaris, Sun OS, Unix, Novell Netware, Windows, dan lain-lain.
2) Menyediakan mesin penyimpan (*engine storage*) transaksi dan non-transaksi
3) Server tersedia sebagai program yang terpisah untuk digunkan pada lingkungan jaringan client/server.
4) MySQL mempunyai library yang dapat ditempelkan pada aplikasi yang berdiri sendiri (*standalone application*) sehingga aplikasi tersebut dapat digunakan pada komputer yang tidak berada dalam jaringan.
5) Mempunyai sistem password yang fleksibel dan aman.

6) Dapat menangani basidata dalam skala besar. Basisdata dalam MySQL Server dapat berisi 50 juta record.
7) Klien dapat terkoneksi ke MySQL Server menggunakan socket TCP/IP pada platform manapun.
8) Server dapat mengirim pesan error ke klien dalam berbagai bahasa.

MySQL termasuk jenis RDBMS (*Relational Database Management System*). Pada MySQL sebuah database terdiri dari tabel-tabel, sebuah tabel terdiri dari baris dan kolom.

Dalam konteks bahasa SQL, informasi disimpan dalam tabel-tabel yang secara logis merupakan struktur dua dimensi yang tersimpan atas baris-baris data (*row* atau *record*) yang berada dalam satu atau lebih kolom (*column*). Baris pada tabel disebut sebagai *instance* dari data sedangkan kolom sering disebut sebagai *attributes* atau *field*.

MySQL mempunyai logo berupa ikan lumba-lumba yang bernama Sakila. Nama tersebut merupakan kontribusi dari pemenang kontes "Nama Lumba-lumba" yang diadakan oleh MySQL AB. Pemenang kontes tersebut bernama Ambroze Twebaze, seorang developer software open source dari Swaziland, Afrika. Menurut Ambroze nama Sakila berasal dari SISwati bahasa lokal Swaziland. Sakila juga merupakan nama kota di Arusha Tanzania.

MySQL yang berjalan di sistem operasi Windows terbukti sangat stabil. MySQL versi Windows mempunyai fitur yang sama dengan MySQL versi Unix, kecuali dalam hal-hal berikut:

1) Jumlah port di Windows terbatas. Sistem Windows mempunyai sekitar 4000 port yang tersedia untuk konekesi klien. Setelah koneksi pada sebuah port ditutup maka dibutuhkan waktu dua sampai empat menit untuk menggunakan kembali port tersebut. Apabila proses koneksi dan dsikoneksi antara klien dan server terjadi dalam kecepatan tinggi, koneksi MySQL akan memakai port-port lain sebelum port yang tertutup terbuka kembali. Hal ini akan menyebabkan server MySQL tampak tidak aktif meskipun sedang aktif. Port yang diset untuk MySQL dapat juga dipakai oleh aplikasi lain yang sedang berjalan di Windows sehingga jumlah port untuk MySQL berkurang jumlahnya.
2) Terbatasnya jumlah file yang dapat dibuka secara serempak. MySQL menjalankan fungsi pread() dan pwrite() untuk menjalankan perintah INSERT dan SELECT. Saat ini di Windows digunakan mutexes untuk menyamai pread() dan pwrite(). Proses itu akan berlangsung lama sehingga antarmuka filediganti dengan *virtual interface* sehingga readfile() dan writefile() pada NT, 2000, dan XP dapat lebih cepat. Implementasi ini menyebabkan file MySQL yang dapat dibuka pada Windows hanya terbatas 2048 buah. Jumlah ini sangat sedikit jika dibandingkan dengan jumlah file MySQL yang dapat dibuka di Unix.
3) Dukungan tipe data yang bermacam-macam seperti: integer 1-4, 8 bytes long, FLOAT, DOUBLE, CHAR, VARCHAR, TEXT, BLOB, DATE, TIME, DATETIME, TIMESTAMP, YEAR, SET, dan ENUM.