

---

# Sistem Manajemen Basis Data

Dr. Lily Wulandari

---

PERTEMUAN 1&2

## **PERENCANAAN, PERANCANGAN DAN ADMINISTRASI DATABASE**

## Pendahuluan

---

- Daur hidup pengembangan Sistem (SDLC) adalah proses yang digunakan oleh analisis sistem untuk mengembangkan sistem informasi , termasuk persyaratan , validasi , dan pelatihan.
- Setiap SDLC harus menghasilkan sistem berkualitas tinggi yang memenuhi atau melampaui harapan pelanggan, mencapai penyelesaian dalam waktu dan perkiraan biaya, bekerja secara efektif dan efisien dalam infrastruktur teknologi informasi saat ini maupun yang direncanakan, dan tidak mahal untuk memeliharanya dan efektif dalam biaya untuk pengembangan lebih lanjut.

## PENDAHULUAN

---

- Daur Hidup Pengembangan Sistem menyediakan kerangka urutan kegiatan untuk diikuti oleh desainer sistem dan pengembang.
- Sebuah daur hidup pengembangan Sistem (SDLC) menganut fase penting yang penting bagi para pengembang, seperti perencanaan , analisis , desain , dan implementasi
- Ada banyak model SDLC yaitu waterfall, fountain, rapid prototyping, RAD, spiral dll.

## PENDAHULUAN

---

- Database dan teknologinya berpengaruh besar terhadap pertumbuhan komputer. Database didefinisikan sebagai kumpulan dari data yang berhubungan. Misalnya nama, nomor telepon, dan alamat yang dapat disimpan (dalam buku, PC, disket).

## PENDAHULUAN

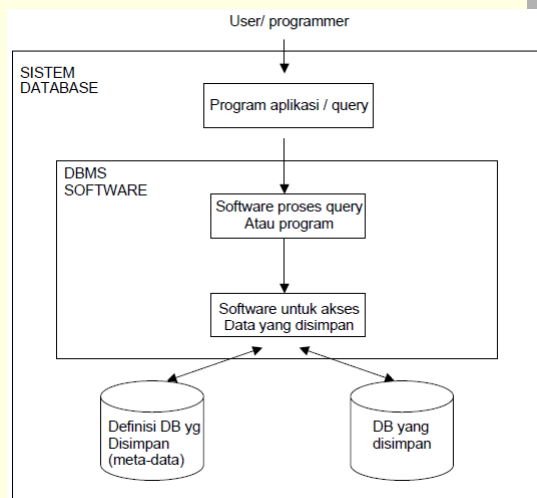
---

- Sistem manajemen database (DBMS) merupakan kumpulan program untuk membuat dan mengelola sebuah database oleh user. DBMS merupakan sistem software general-purpose yang memiliki fasilitas proses define, construct dan manipulate database untuk aplikasi yang bervariasi.

## PENDAHULUAN

- Define : spesifikasi tipe data, struktur dan constraint data yang akan disimpan dalam database
- Construct : proses menyimpan data itu sendiri ke dalam beberapa media penyimpanan yang dikontrol DBMS.
- Manipulate : fungsi seperti query database untuk memanggil data khusus, update database dan generate laporan dari data.

## PENDAHULUAN



Gambar 1. lingkungan sistem database

## PENDAHULUAN

### CONTOH

- Database UNIVERSITAS
- Database mengorganisasikan 5 buah file masing2 menyimpan record data yang bertipe sama.

#### STUDENT

Name	StudentNumber	Class	Major
Smith	17	1	CS
Brown	8	2	CS

## PENDAHULUAN

#### COURSE

CourseName	CourseNumber	CreditHours	Department
Intro to CS	CS1310	4	CS
Data structure	CS3220	4	CS
Discrete math	MATH2410	3	MATH
Database	CS3380	3	CS

#### SECTION

SectionIdentifier	CourseNumber	Semester	Year	Instructor
85	MATH2410	Fall	98	King
92	CS1310	Fall	98	Anderson
102	CS3320	Spring	99	Knuth
112	MATH2410	Fall	99	Chang
119	CS1310	Fall	99	Anderson
135	CS3380	Fall	99	stone

## PENDAHULUAN

### GRADE\_REPORT

StudentNumber	SectionNumber	Grade
17	112	B
17	119	C
8	85	A
8	92	A
8	102	B
8	135	A

### PREREQUISITE

CourseNumber	PrerequisiteNumber
CS3380	CS3320
CS3380	MATH2410
CS3320	CS1310

## PENDAHULUAN

### DEFINE :

- Struktur dari record per file dispesifikasikan dengan tipe elemen data yang berbeda untuk disimpan.
- File STUDENT terdiri dari StudentName, StudentNumber, Class dan Major.
- Tipe data untuk setiap elemen data dari record juga perlu dibuat.

## PENDAHULUAN

---

- StudentName merupakan string dari karakter alfabet, StudentNumber dibuat menjadi integer, dst.
- Pengkodean juga dibuat, misalnya Class dari STUDENT, 1 untuk freshman, 2 untuk sophomore, 3 untuk junior, 4 untuk senior dan 5 untuk graduate student.

## PENDAHULUAN

---

### CONSTRUCT :

- Data yang mewakili student, course, section, grade report dan prerequisite disimpan sebagai sebuah record dalam masing2 filenya. Ada beberapa record yang berrelasi. Misalnya record "Smith" di STUDENT berrelasi dengan 2 buah record di file GRADE\_REPORT.

## PENDAHULUAN

---

### MANIPULATE :

- Berhubungan dengan query dan update.  
Query : “retrieve the transcript – a list of all courses and grades – of Smith”; list  
Update : “change the class of Smith to Sophomore”; create; enter
- Sebelum diproses, query dan update yang informal seperti tersebut perlu diterjemahkan ke bahasa sistem database.

## PERKEMBANGAN DATABASE

---

- Aplikasi tradisional DB : deposit atau penarikan uang, reservasi hotel atau penerbangan, akses katalog perpustakaan yang terkomputerisasi, dll.
- Pada model ini, kebanyakan informasi disimpan dan diakses secara teks atau numeric.



## PERKEMBANGAN DATABASE

---

- Database multimedia merupakan teknologi lanjut yang dapat menyimpan gambar, video klip, pesan suara. GIS dapat menyimpan dan menganalisa peta, data cuaca, dan citra satelit.
- Data warehouse dan OnLine Analytical Processing (OLAP) digunakan untuk mengekstrak dan menganalisis informasi yang berguna dari database yang sangat besar untuk pengambilan keputusan.

## PERKEMBANGAN DATABASE

---

- Teknologi database aktif dan real-time digunakan untuk mengatur proses industri dan pabrikasi. Teknik pencarian database digunakan di WWW untuk meningkatkan pencarian informasi yang dibutuhkan user.

## PERENCANAAN SISTEM

---

- Perencanaan sistem menyangkut estimasi dari kebutuhan-kebutuhan fisik, tenaga kerja dan dana yang dibutuhkan untuk mendukung pengembangan sistem ini serta untuk mendukung operasinya setelah diterapkan.
- Pada perencanaan sistem, suatu sistem yang diusulkan harus layak dan mendukung faktor strategik. Untuk menilai kedua kemungkinan tersebut maka harus diadakan evaluasi terhadap faktor kelayakan TELOS dan faktor strategi PDM.

## PERENCANAAN SISTEM

---

### **FAKTOR KELAYAKAN (TELOS):**

- *Technical*  
Menunjukkan apakah sistem yg diusulkan dapat dikembangkan dan diterapkan dengan menggunakan teknologi yang ada atau jika membutuhkan teknologi baru

## PERENCANAAN SISTEM

---

- *Economic*

Menunjukkan apakah dana yg memadai tersedia untuk mendukung biaya dari sistem yg diusulkan

- *Legal*

Menunjukkan apakah ada konflik antara sistem yang sedang dipertimbangkan dan kemampuan perusahaan untuk menunaikan kewajibannya

## PERENCANAAN SISTEM

---

- *Operational*

Menunjukkan apakah prosedur dan ketrampilan personalia yg ada cukup untuk mengoperasikan sistem yg diusulkan atau apakah prosedur dan ketrampilan tambahan akan diberikan

- *Schedule*

Sistem yg diusulkan harus berlaku dalam suatu kerangka waktu yang logis

## PERENCANAAN SISTEM

---

### FAKTOR STRATEGIK (PDM) :

- *Produktivitas*. Mengukur jumlah output yang dihasilkan dari input. Tujuan untuk mengurangi atau menghapus biaya yang tidak menambah nilai. Diukur dengan RATIO, misal total biaya tenaga kerja mingguan dibandingkan dengan jumlah unit yang dihasilkan selama seminggu atau jumlah bahan mentah yang masuk selama seminggu dibandingkan dengan jumlah barang jadi yang dihasilkan selama seminggu.

## PERENCANAAN SISTEM

---

- *Diferensiasi*. Mengukur seberapa baik suatu perusahaan dapat menawarkan produknya atau pelayanannya yang secara nyata tidak serupa dengan jenis dan sifat dari produk dan pelayanan para pesaingnya. Dicapai melalui peningkatan kualitas, keanekaragaman, penanganan khusus, pelayanan cepat, biaya rendah dan sebagainya.
- *Manajemen*. Menunjukkan seberapa baik sistem informasi menyediakan informasi untuk membantu para manajer dalam perencanaan, pengendalian dan pengambilan keputusan.

## Daur Hidup (*Life Cycle*) yang Umum dari Aplikasi Basis Data

---

- Definisi Sistem
- *Database Design*
- Implementasi
- Loading/Konversi Data
- Konversi Aplikasi
- Testing & Validasi
- Operations
- Control & Maintenance

## Daur Hidup (*Life Cycle*) yang Umum dari Aplikasi Basis Data

---

- Definisi Sistem:
  - ruang lingkup basis data
  - pemakai
  - aplikasi
- *Design*:
  - *logical design* → ER/EER
  - *physical design* untuk suatu DBMS

## Daur Hidup (*Life Cycle*) yang Umum dari Aplikasi Basis Data

---

- Implementasi:
  - membuat basis data (kosong)
  - membuat program aplikasi
- Loading/ Konversi Data:
  - memasukkan data ke dalam basis data
  - mengkonversi file yang sudah ada ke dalam format basis data dan kemudian memasukkannya dalam basis data

## Daur Hidup (*Life Cycle*) yang Umum dari Aplikasi Basis Data

---

- Konversi Aplikasi:

Semua aplikasi dari sistem sebelumnya dikonversikan ke dalam sistem basis data.
- Testing dan Validasi:

Sistem yang baru harus ditest dan divalidasi (diperiksa keabsahannya).

## Daur Hidup (*Life Cycle*) yang Umum dari Aplikasi Basis Data

---

- Operasi:  
Pengoperasian basis data dan aplikasinya.
- *Monitoring dan Maintenance*:  
Selama operasi, sistem dimonitor dan diperlihara. Baik data maupun program aplikasi masih dapat terus tumbuh dan berkembang.

## Proses *Design* Sistem Basis Data

---

Basis Data biasanya merupakan salah satu bagian dari suatu sistem informasi yang besar yang antara lain terdiri dari:

- Data
- Perangkat lunak DBMS
- Perangkat keras komputer
- Perangkat lunak dan sistem operasi komputer
- Program-program aplikasi
- Pemrogram, dll

## Proses *Design* Basis Data

---

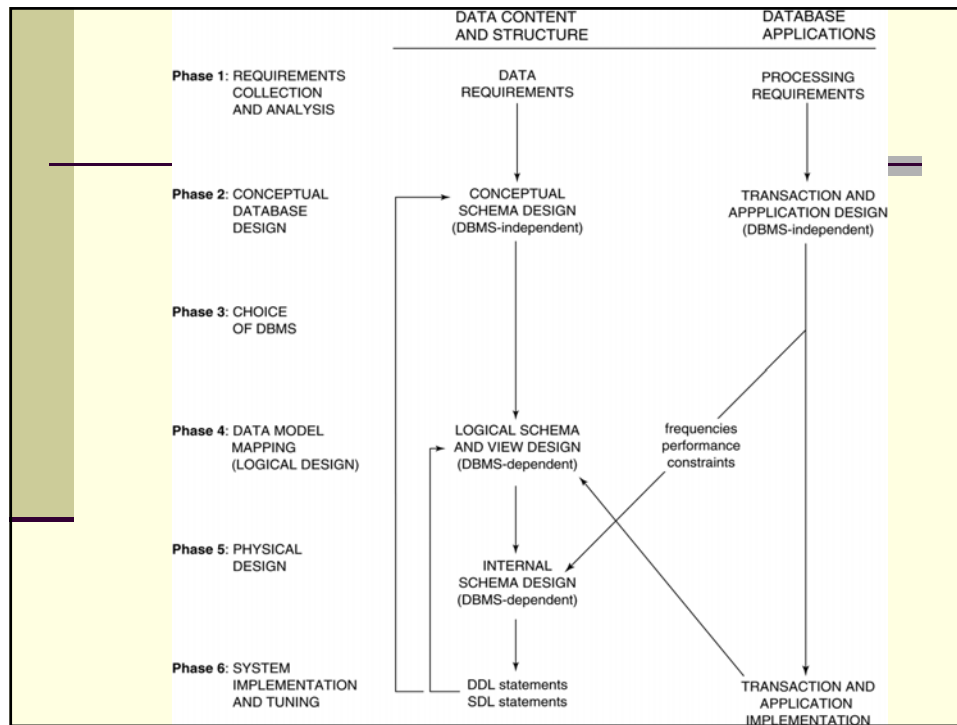
1. Pengumpulan dan analisa *requirement*
2. *Design* basis data *conceptual*
3. Pemilihan DBMS
4. *Mapping* dari *conceptual* ke *logical*
5. *Physical Design*
6. Implementasi

## Proses *Design* Basis Data (cont'd)

---

- Keenam *phase* dalam proses *design* tidak perlu dilaksanakan secara mutlak, mungkin ada umpan balik antar *phase* dan dalam masing-masing *phase*





## Proses *Design* Paralel

Proses design terdiri dari dua proses yang paralel yaitu:

- proses design dari data dan struktur dari basis data (*data driven*)
- proses design dari program aplikasi dan pemrosesan basis data (*process driven*)

## Mengapa Harus Paralel

---

Karena kedua proses tersebut saling bergantung.

Contoh:

1. Menentukan data item yang akan disimpan dalam basis data tergantung dari aplikasi basis data tersebut, juga dalam menentukan struktur dan akses path.
2. Design dari program aplikasi tergantung dari struktur basis datanya.
3. Biasanya condong ke salah satu.

## *Phase 1: Pengumpulan Data & Analisa Requirement*

---

- Pengidentifikasian group pemakai dan area aplikasi
- Penelitian kembali dokumen-dokumen yang sudah ada yang berhubungan dengan aplikasi → form, report, manual, organization chart, dsb
- Analisa lingkungan operasi dan kebutuhan dari pemrosesan, seperti tipe transaksi, input/output, frekuensi suatu transaksi, dsb

## *Phase 1: Pengumpulan Data & Analisa Requirement (Con'd)*

---

- Transfer informasi informal ke dalam bentuk terstruktur menggunakan salah satu bentuk formal dari requirement specification (bentuk diagram) seperti Flow Chart, DFD, UML Diagram, dll. Hal ini dilakukan untuk mempermudah pemeriksaan kekonsistenan, ketepatan, dan kelengkapan dari spesifikasi

## *Phase 2: Design Conceptual*

---

### **Phase 2A: Design Conceptual Schema**

- High level data model, bukan implementation-level data model
- Memberikan gambaran yang lengkap dari struktur basis data yaitu arti, hubungan, dan batasan-batasan.
- Conceptual schema bersifat tetap
- Alat komunikasi antar pemakai basis data, designer, dan analis

## Phase 2: Design Conceptual (cont'd)

---

### Phase 2A: Design Conceptual Schema

- Harus bersifat:
  - Mampu menyatakan relationship, batasan-batasan
  - Diagram
  - Formal, minimum dalam menyatakan spesifikasi data (tidak ada duplikasi)
  - *Simple*
- *Conceptual data model* harus *DBMS independent* → ER/EER

## Strategi untuk Design Schema

---

- *Top Down:*
  - mulai dengan beberapa *high level entity type*
  - bagi lagi (*top down*) menjadi beberapa *lower-level entity type* dan *relationship type*
- *Bottom Up:*
  - mulai dengan atribut
  - kelompokkan menjadi *entity type & relationship type*
  - tambahkan *relationship-relationship* baru bila ada

## Strategi untuk Design Schema (cont'd)

---

- *Inside Out:*
  - bentuk khusus dari *bottom-up*
  - mula-mula ditentukan *entity type* yang merupakan pusat/bagian terpenting
  - tambahkan *entity type* dan *relationship* lain yang berhubungan satu sama lain

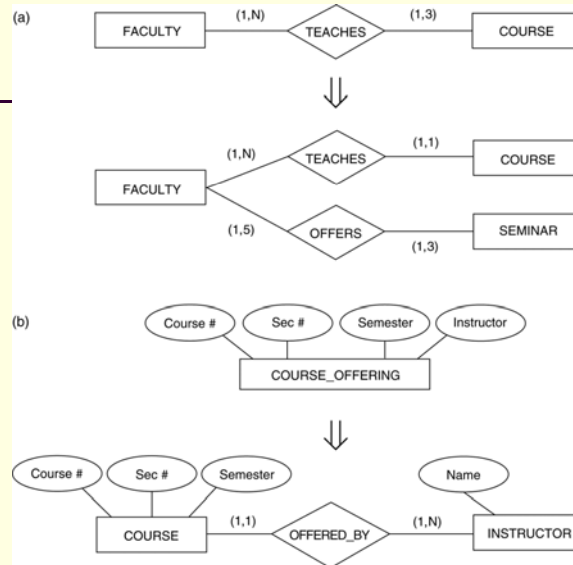
## Strategi untuk Design Schema (cont'd)

---

- *Mixed:*
  - *requirement* dibagi-bagi menggunakan strategi *top down*
  - sebagian dari *schema* di-*design* dari partisi-partisi menggunakan strategi *bottom-up*
  - bagian-bagian dari komponen-komponen tersebut kemudian digabungkan

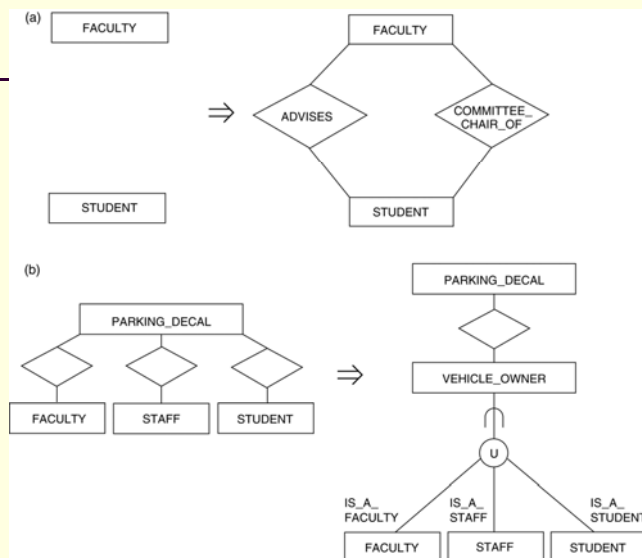
**Examples of top-down refinement.**

- (a) Generating a new entity type.
- (b) Decomposing an entity type into two entity types and a relationship type.



**Examples of bottom-up refinement.**

- (a) Discovering and adding new relationships.
- (b) Discovering a new category (union type) and relating it.



## *Phase 2b: Design Transaksi*

---

- Pada saat suatu basis data di-design, aplikasi dari transaksi utama harus sudah diketahui
- Transaksi-transaksi baru dapat didefinisikan kemudian
- Tentukan karakteristik dari transaksi dan periksa apakah basis data sudah memuat semua informasi untuk melaksanakan transaksi

## *Phase 2b: Design Transaksi (cont'd)*

---

- Transaksi dapat dibagi dalam 3 bagian yaitu:
  - *retrieval*
  - *update*
  - *mixed*
- *Phase 2a* dan *2b* sebaiknya dilaksanakan secara paralel dengan menggunakan umpan balik agar didapat *design schema* dan transaksi yang stabil

## Phase 3: Pemilihan DBMS

---

- Pemilihan DBMS ditentukan oleh sejumlah faktor antara lain:
  - **faktor teknis:** *storage, akses path, user interface, programmer, bahasa query*
  - **faktor ekonomi:** software, hardware, maintenance, training, operasi, konversi, teknisi, dll
  - **faktor organisasi:** kompleksitas, sharing antar aplikasi, perkembangan data, pengontrolan data, ketersediaan personal yg terbiasa dengan suatu sistem, adanya jaminan purna jual, dll

## *Phase 4: Mapping dari Data Model*

---

- Pemetaan dapat diproses dalam 2 tingkat :
  1. Pemetaan system-independent :  
pemetaan ke dalam model data DBMS dengan tidak mempertimbangkan karakteristik atau hal-hal yang khusus yang berlaku pada implementasi DBMS dari model data tsb.



### *Phase 4: Mapping dari Data Model*

---

2. Penyesuaian skema ke DBMS yang spesifik :  
mengatur skema yang dihasilkan pada langkah 1 untuk disesuaikan pada implementasi yang khusus di masa yang akan datang dari suatu model data yang digunakan pada DBMS yang dipilih

- Hasil pemetaan biasanya berupa DDL

### *Phase 5: Physical Design*

---

- Struktur *storage*, akses *path* untuk mendapatkan *performance* yang baik
- Kriteria baik dapat dilihat dari:
  - *response time*
  - pemakaian *storage*
  - *throughput* (jumlah transaksi per unit waktu)
- Perlu *tuning* untuk memperbaiki *performance* berdasarkan statistik pemakaian

## Phase 6: Implementasi Sistem Basis Data

- DDL dan SDL (Storage Definition Language) dari DBMS dikompilasi membentuk *schema* basis data dan basis data yang masih kosong
- Basis data dapat dimuati (di-load) dari sistem yang lama
- Transaksi dapat diimplementasikan oleh program aplikasi dan dikompilasi
- Siap dioperasikan

## Administrasi Database

- Fungsi pengelolaan dan pemeliharaan perangkat lunak manajemen sistem database (DBMS). Perangkat lunak DBMS yang utama saat ini seperti Oracle , IBM DB2 dan Microsoft SQL Server membutuhkan pengelolaan yang berkelanjutan. Dengan demikian, perusahaan yang menggunakan perangkat lunak DBMS sering mempekerjakan karyawan khusus TI ( Teknologi Informatika ) yang disebut Database Administrator atau DBA.

## Adiministrasi Database

---

### Tanggung Jawab DBA (Con'd)

- Instalasi, konfigurasi dan upgrade perangkat lunak server Database dan produk terkait.
- Mengevaluasi fitur Database dan produk terkait.
- Membentuk dan memelihara kebijakan dan prosedur backup dan recovery.
- Mengurus desain database dan implementasi.

## Adiministrasi Database

---

### Tanggung Jawab DBA

- Menerapkan dan memelihara keamanan database (menciptakan dan menjaga users dan roles, menetapkan privileges).
- Database tuning dan pemantauan kinerja.
- Aplikasi tuning dan pemantauan kinerja.
- Setup dan memelihara dokumentasi dan standar.

## Adiministrasi Database

---

### Tanggung Jawab DBA (*Con'd*)

- Rencana pertumbuhan dan perubahan (perencanaan kapasitas).
- Bekerja sebagai bagian dari tim dan memberikan dukungan 7x24 jam bila diperlukan.
- Mengatasi masalah teknis umum dan memberikan konsultasi

## Adiministrasi Database

---

### Ada tiga jenis DBA:

- Sistem DBAs (juga disebut sebagai Fisik DBA, DBA atau Dukungan Operasi Produksi DBA): fokus pada aspek fisik administrasi database seperti instalasi, konfigurasi, patch, upgrade, backup DBMS, mengembalikan, menyegarkan, kinerja optimasi, pemeliharaan dan disaster recovery.
- Development DBA: fokus pada pengembangan aspek logis dan administrasi database seperti desain dan pemeliharaan model data, menghasilkan DDL, menulis SQL dan tuning, coding stored procedure, bekerja sama dengan pengembang untuk membantu memilih fitur/fungsi DBMS yang paling sesuai dan kegiatan pra-produksi lain

## Adiministrasi Database

- Application DBA: biasanya ditemukan dalam organisasi yang telah membeli perangkat lunak aplikasi pihak ke-3 seperti sistem ERP ( [enterprise resource planning](#) ) dan CRM ( [customer relationship management](#) ). Contoh aplikasi perangkat lunak tersebut mencakup [Aplikasi Oracle](#) , Siebel dan [PeopleSoft](#) (sekarang keduanya bagian dari Oracle Corp) dan SAP. Aplikasi DBA tidak memihak antara DBMS dan software aplikasi dan bertanggung jawab untuk memastikan bahwa aplikasi sepenuhnya dioptimalkan untuk database dan sebaliknya. Mereka biasanya mengelola semua [komponen aplikasi](#) yang berinteraksi dengan database dan melakukan kegiatan seperti instalasi aplikasi dan patch, upgrade aplikasi, kloning database, membangun dan menjalankan rutinitas pembersihan data, manajemen proses load data, dll

## Adiministrasi Database

- Sementara individu biasanya mengkhususkan diri dalam satu jenis administrasi database, dalam organisasi yang lebih kecil, tidak jarang untuk menemukan individu atau kelompok melakukan lebih dari satu jenis administrasi database.

