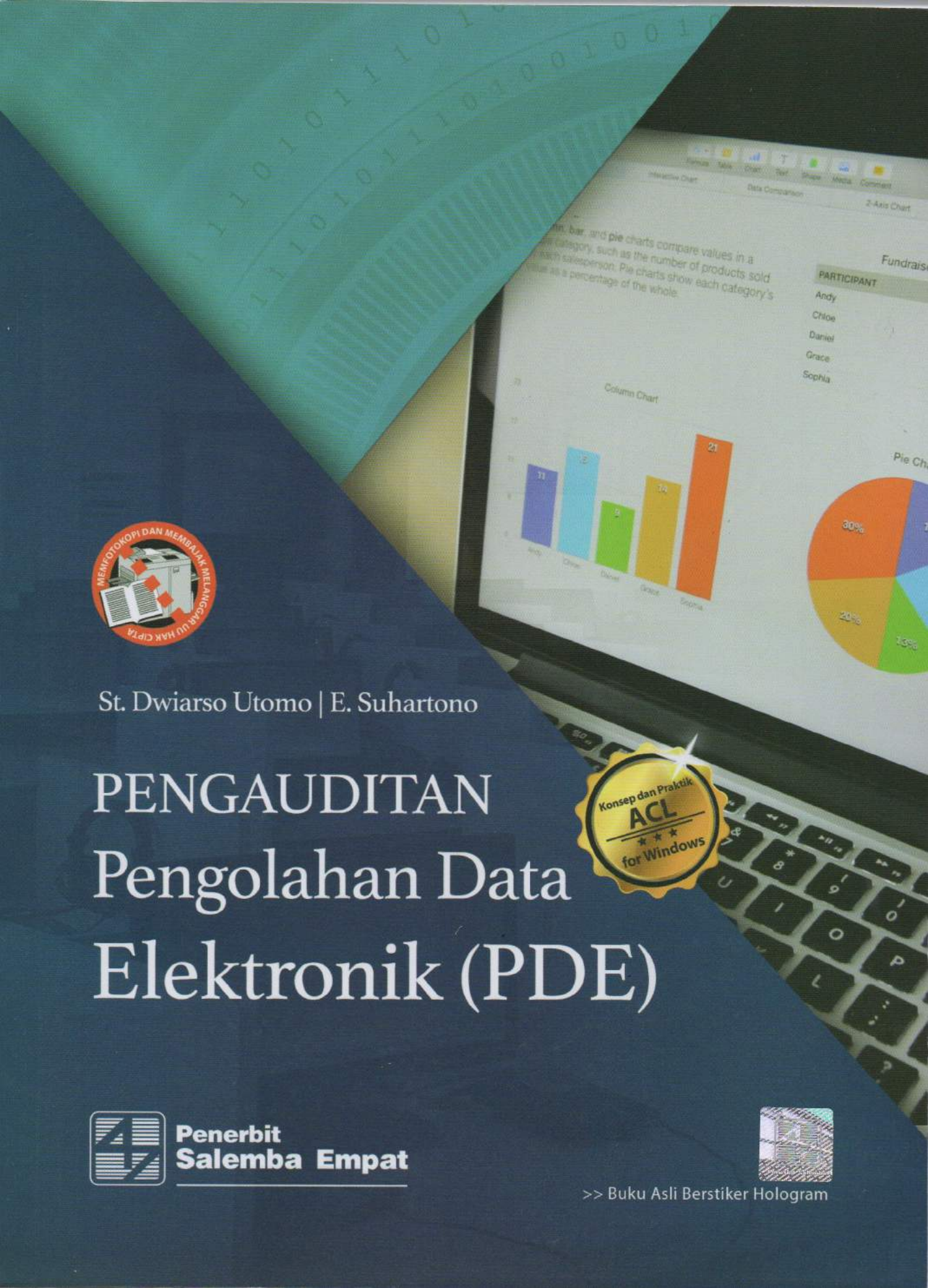



St. Dwiarso Utomo | E. Suhartono

# PENGAUDITAN Pengolahan Data Elektronik (PDE)

Penerbit Salemba Empat

>> Buku Asli Berstiker Hologram

**Pengauditan Pengolahan Data Elektronik (PDE)—Konsep dan Praktik ACL for Windows**

Dr. St. Dwiarso Utomo, S.E., M.Kom., Ak., C.A, Entot Suhartono, S.Kom, M.Kom

Manajer Penerbitan dan Produksi: Novietha Indra Sallama
Supervisor Editor: Ema S. Suharsi
Copy Editor: Bambang Hernalyk
Tata Letak: Dedy Juni Asmara
Desain Sampul: Asyfa Ainur Khasanah


Jln. Raya Lenteng Agung No. 101
Jagakarsa, Jakarta Selatan 12610
Telp. : (021) 781 8616
Faks. : (021) 781 8486
Website: http://www.penerbitsalemba.com
E-mail : info@penerbitsalemba.com



Utomo, St. Dwiarso
Suhartono, Entot

Pengauditan Pengolahan Data Elektronik (PDE)—Konsep dan Praktik ACL for Windows/St. Dwiarso Utomo, Entot Suhartono

—Jakarta: Salemba Empat, 2018
1 jil., 204 hlm., 19 × 26 cm

ISBN 978-979-061-800-8

1. Akuntansi
2. Pengauditan Pengolahan Data Elektronik (PDE)—Konsep dan Praktik ACL for Windows

I. Judul
II. St. Dwiarso Utomo, Entot Suhartono

000.1.07

# KATA PENGANTAR

Puji syukur penulis haturkan kehadirat Allah SWT karena dengan berkat rahmat, hidayah, dan karunia-Nya penulis berhasil menyelesaikan buku dengan judul "Pengauditan Pengolahan Data Elektronik (PDE)—Konsep dan Praktik ACL for Windows" ini. Perkembangan teknologi informasi (TI) pada saat ini, hampir merambah semua aspek kehidupan untuk mempermudah pekerjaan manusia, salah satunya adalah bidang pengauditan. Perkembangan TI memberikan dampak pada proses transaksi dan pelaporan akuntansi, semula proses-proses tersebut dilakukan secara tradisional (manual) sekarang dilakukan secara otomatis dan *online* sehingga dapat mendukung operasional yang lebih efisien dan terintegrasi. Namun demikian, perkembangan tersebut membawa dampak risiko baru yang membutuhkan pengendalian internal khusus. Kemajuan tersebut telah melahirkan kebutuhan akan berbagai teknik baru untuk pengauditan, mengevaluasi pengendalian, dan

memastikan keamanan serta akurasi data perusahaan dan sistem informasi yang menghasilkannya.

Buku ini terbagi menjadi 2 bagian, yaitu bagian pertama, terdiri dari konsep audit dan *assurance*, pengendalian internal di lingkungan sistem informasi, sistem informasi berbasis komputer, sistem manajemen data, serta alat dan teknik audit berbantuan komputer. Bagian kedua berisi mengenai penggunaan aplikasi ACL for Windows V.9, terdiri dari penggunaan aplikasi ACL mulai dari persiapan data yang akan diperiksa sampai dengan pengolahan data dasar (Filter, Count, Total, Klasifikasi, Stratify, Statistik, Umur, dan lain-lain), dan pembahasan terakhir adalah analisis data dengan aplikasi ACL, seperti uji keterurutan data, uji kelengkapan data, uji duplikasi data, dan membuat tabulasi silang.

Buku ini disajikan telah mendapatkan sejumlah masukan dari para dosen Fakultas Ekonomi dan Bisnis Universitas Dian Nuswantoro yang tentunya dapat tampil dalam keutuhan yang terjaga kualitas akademiknya, walaupun demikian disadari sepenuhnya bahwa selalu ada keterbatasan dalam setiap penulisan. Untuk itu, kritik dan saran selalu diharapkan. Semoga buku ini dari waktu ke waktu dapat disempurnakan dengan kualitas akademik yang lebih baik.

Tidak lupa kami sampaikan terima kasih kepada Rektor, Dekan Fakultas Ekonomi dan Bisnis, Ketua Program Studi Akuntansi, Dosen Akuntansi Universitas Dian Nuswantoro, dan seluruh pihak yang telah membantu dalam penyusunan buku ini.

Semarang, Juli 2018

Penulis

# TENTANG PENULIS

**Dr. St. Dwiarso Utomo, S.E., M.Kom., Ak., C.A.**, menyelesaikan Studi S1 Program Studi Akuntansi Fakultas Ekonomi UNDIP, menyelesaikan studi S2 Magister Komputer di STIMIK Banarif Indonesia, dan menyelesaikan Studi S3 Doktor Ekonomi di Universitas Merdeka Malang. Saat ini beliau adalah dosen di bidang Sistem Informasi Akuntansi, Audit, Akuntansi Manajemen. Selain itu, beliau juga menjabat sebagai Wakil Rektor II Universitas Dian Nuswantoro. Beliau aktif menulis artikel di Jurnal Nasional Akreditasi dan Internasional dan sebagai Anggota Ikatan Akuntansi Indonesia.

**Entot Suhartono, S.Kom., M.Kom.**, menyelesaikan Studi S1 Program Studi Sistem Informasi Fakultas Ilmu Komputer Universitas Dian Nuswantoro dan menyelesaikan S2 Pascasarjana Magister Sistem Informasi Universitas Diponegoro. Saat ini beliau adalah dosen di Universitas Dian Nuswantoro bidang Sistem Informasi, Audit Sistem Informasi, dan Sistem Informasi Akuntansi. Beliau juga aktif menulis artikel di Jurnal dan Prosiding Internasional.

# DAFTAR ISI

# BAB 1

# KONSEP AUDIT DAN ASSURANCE

Setelah mempelajari bab ini, Anda diharapkan mampu:

- Mengetahui definisi audit dan audit teknologi informasi, serta mengetahui perbedaan jenis jasa atestasi dan *assurance*, kemudian dapat menjelaskan hubungan antara kedua jenis audit tersebut.
- Memahami struktur audit dan memiliki pemahaman atas perusahaan dari sudut berbagai elemen kontekstual dalam proses audit.
- Memahami perlunya audit TI pada perusahaan atau organisasi dan dampak kegagalan dari TI.

Perkembangan terbaru dalam teknologi informasi (TI) telah memberikan dampak besar di bidang audit (auditing). TI telah menginspirasi rekayasa ulang berbagai proses bisnis tradisional untuk mendukung operasi yang lebih efisien dan untuk meningkatkan komunikasi dalam entitas serta antara entitas dengan para pelanggan dan pemasoknya. Akan tetapi, berbagai kemajuan tersebut membawa dampak risiko baru yang membutuhkan pengendalian internal khusus. Kemajuan tersebut telah melahirkan kebutuhan akan berbagai teknik baru untuk mengevaluasi pengendalian dan untuk memastikan keamanan serta akurasi data perusahaan dan sistem informasi yang menghasilkannya.

Bab ini akan membahas gambaran umum mengenai audit komputer. Bab ini dimulai dengan pembahasan umum mengenai definisi audit, berbagai alternatif pendekatan audit. Pada bagian lain dari bab ini akan diulas juga mengenai kebutuhan audit teknologi informasi di lingkungan perusahaan atau organisasi yang telah menerapkan teknologi informasi dalam proses bisnisnya.

## 1.1 JENIS-JENIS AUDIT

### 1.1.1 Definisi Audit

Proses sistematis untuk memperoleh dan mengevaluasi secara objektif bukti yang berkaitan dengan penilaian berbagai kegiatan dan peristiwa ekonomi untuk memastikan tingkat kesesuaian antara penilaian tersebut dan membentuk kriteria serta menyampaikan hasilnya ke para pengguna yang berkepentingan.

Profesi audit terdiri atas beberapa jenis audit yang masing-masing memiliki perspektif, tujuan, dan organisasi profesinya sendiri-sendiri. Namun demikian, semua jenis audit mengikuti proses, petunjuk, dan standar umum, yang masing-masing memiliki perbedaan dalam beberapa hal.

### 1.1.2 Audit Internal dan Audit Eksternal

Institute of Internal Auditors (IIA) mendefinisikan audit internal sebagai fungsi penilaian independen yang dibentuk dalam perusahaan untuk mempelajari dan mengevaluasi berbagai aktivitasnya sebagai layanan kepada perusahaan. Auditor internal melakukan audit keuangan, mempelajari kepatuhan suatu operasi terhadap kebijakan perusahaan, mengkaji kepatuhan perusahaan terhadap kewajiban hukumnya, dan mengevaluasi efisiensi operasional, mendeteksi dan mengejar pelaku penipuan dalam perusahaan, serta melakukan audit teknologi informasi (TI), di mana semua aktivitas tersebut atas nama perusahaan.

Auditor ini harus memiliki sertifikasi dari Certified Internal Auditor (CIA) atau Certified Information System Auditor (CISA). Auditor internal mewakili kepentingan perusahaan dan pada umumnya bertanggung jawab kepada pihak manajemen eksekutif perusahaan. Standar, petunjuk, dan sertifikasi audit internal pada umumnya diatur oleh IIA dan untuk tingkat tertentu oleh Information System Audit and Control Association (ISACA).

Auditor eksternal mewakili kepentingan pihak luar sedangkan auditor internal mewakili kepentingan perusahaan. Untuk melakukan audit keuangan, auditor internal biasanya bekerja sama dan membantu auditor eksternal. Kerja sama ini agar dapat mencapai efisiensi audit dan untuk mengurangi biaya kegiatan audit.

Auditor eksternal dapat bergantung pada bukti yang disediakan oleh departemen audit internal yang secara organisasi independen dan bertanggung jawab pada komite audit dewan komisaris. Audit internal dapat mengumpulkan bukti audit sepanjang periode fiskal, kemudian bukti tersebut dapat digunakan auditor eksternal pada akhir tahun untuk melakukan audit yang lebih efisien.

## 1.2 AUDIT KEUANGAN

Audit keuangan adalah atestasi (pembuktian) independen yang dilakukan oleh auditor yang menyatakan pendapatnya atas penyajian laporan keuangan. Konsep utama dalam proses audit adalah *independensi*, di mana auditor mengumpulkan dan mengevaluasi bukti serta memberikan pendapatnya berdasarkan bukti secara independen. Selain independen terhadap proses audit, auditor juga harus independen dari perusahaan klien. Kepercayaan publik atas keandalan laporan keuangan yang dihasilkan secara internal oleh perusahaan tergantung secara langsung pada evaluasi atas laporan tersebut oleh auditor independen.

Pernyataan publik atas pendapat auditor adalah puncak dari proses audit yang sistematis melibatkan tiga tahapan konseptual, yaitu:

1. Adaptasi terhadap bisnis perusahaan;
2. Mengevaluasi dan menguji berbagai pengendalian internal; dan
3. Menilai keandalan data keuangan.

### 1.2.1 Jasa Atestasi dan Jasa Assurance

#### A. Jasa Atestasi

Atestasi merupakan perjanjian di mana seorang praktisi yang dikontrak untuk mengeluarkan, atau telah mengeluarkan sebuah komunikasi tertulis yang menyatakan kesimpulan mengenai keandalan penilaian tertulis yang merupakan tanggung jawab pihak lainnya.

Persyaratan yang berlaku pada jasa atestasi:

- Adanya penilaian tertulis dan laporan tertulis dari praktisi terkait;
- Kriteria pengukuran yang formal atau penjelasannya dalam penyajiannya; dan
- Tingkat jasa dibatasi pada pemeriksaan, pengkajian, dan penerapan berbagai prosedur yang telah disepakati sebelumnya.

#### B. Jasa Assurance

Jasa *assurance* adalah layanan profesional yang didesain untuk meningkatkan kualitas informasi, secara keuangan maupun nonkeuangan, yang digunakan oleh *decision*

*makers*. Sebagai contoh, jasa *assurance* (penjaminan) dapat dibuat untuk menyediakan informasi mengenai kualitas atau nilai komersial suatu produk. Seorang klien dapat membutuhkan informasi mengenai efisiensi suatu proses produksi atau efektivitas sistem keamanan jaringannya. Jasa penjaminan ditujukan untuk membantu dalam membuat keputusan yang lebih baik atas fungsi yang diatestasi, atau merupakan hasil dari pengkajian yang disengaja dilakukan secara independen.

Unit organisasi yang bertanggung jawab untuk melakukan audit TI biasanya disebut sebagai:

- Manajemen risiko TI (*IT risk management*);
- Manajemen risiko sistem informasi (*information system risk management*); atau
- Manajemen risiko sistem operasional (*operational system risk managemen—OSRM*).

### 1.2.2 Standar Audit

Produk fungsi atestasi adalah laporan tertulis formal yang menyatakan pendapat mengenai keandalan penilaian yang terdapat dalam laporan keuangan. Laporan auditor tersebut menyatakan pendapat apakah laporan keuangan sesuai dengan prinsip-prinsip akuntansi yang diterima secara umum (Standar Akuntansi Keuangan—SKA atau *generally accepted accounting principles*—GAAP). Auditor menjalankan tugas dan tanggung jawab profesional berdasarkan standar audit yang diterima secara umum.

#### A. Proses Sistematis

Proses pelaksanaan audit dilakukan secara sistematis dan logis serta berlaku untuk semua bentuk sistem informasi. Kerangka kerja logis untuk melakukan audit di lingkungan sistem berbasis TI sangat penting untuk membantu auditor mengidentifikasi semua proses serta *file* data yang penting. Kurangnya prosedur fisik yang secara visual dapat diverifikasi dan dievaluasi, menambah kompleksitas dalam pengauditan pengolahan data elektronik.

#### B. Pernyataan Manajemen dan Tujuan Audit

Pengaturan laporan keuangan mencerminkan rangkaian pernyataan manajemen atas kesehatan atau kinerja keuangan suatu perusahaan atau organisasi. Tugas auditor adalah menetapkan apakah laporan keuangan tersebut disajikan secara wajar. Agar auditor dapat menetapkan hal tesebut, maka auditor menentukan tujuan audit, mendesain prosedur, dan mengumpulkan bukti yang mendukung atau menolak penilaian manajemen.

Pernyataan manajemen dapat dikategorikan menjadi lima, yaitu:

1. *Keberadaan atau Keterjadian*
   Pernyataan ini menguatkan bahwa semua aset dan ekuitas yang berada di dalam neraca benar-benar ada dan semua transaksi dalam laporan laba rugi benar-benar terjadi.

2. *Kelengkapan*
   Menyatakan bahwa tidak ada aset, ekuitas, atau transaksi yang material telah dihilangkan dari laporan keuangan terkait.
3. *Hak dan Kewajiban*
   Memiliki arti bahwa aset yang muncul dalam neraca dimiliki oleh entitas terkait dan bahwa kewajiban yang dilaporkan merupakan kewajiban perusahaan.
4. *Penilaian atau Alokasi*
   Menyatakan bahwa aset dan ekuitas dinilai berdasarkan prinsip-prinsip akuntansi dan bahwa jumlah yang dialokasikan seperti beban penyusutan dihitung secara sistematis dan rasional.
5. *Penyajian atau Pengungkapan*
   Menyatakan bahwa bagian dalam laporan keuangan telah diklasifikasikan dengan benar (misalnya, kewajiban jangka panjang tidak jatuh tempo dalam satu tahun) dan bahwa catatan atas laporan keuangan yang merupakan pengungkapan cukup memadai hingga dapat mecegah terjadinya penyesatan pengguna laporan keuangan.

Auditor mengembangkan tujuan audit dan mendesain prosedur audit berdasarkan penilaian sebelumnya.

| Pernyataan Manajemen | Tujuan Audit | Prosedur Audit |
|---|---|---|
| Keberadaan atau Kejadian | Persediaan yang dicantumkan dalam neraca benar-benar ada. | Mengamati perhitungan fisik persediaan. |
| Kelengkapan | Utang usaha meliputi semua kewajiban ke pemasok untuk periode terkait. | Membandingkan laporan penerimaan, faktur dari pemasok, pemesanan pembelian dan ayat jurnal untuk periode terkait, serta awal periode berikutnya. |
| Hak dan Kewajiban | Pabrik dan perlengkapan yang dicantumkan dalam neraca dimiliki oleh entitas. | Meninjau kembali perjanjian pembelian, kebijakan asuransi, dan berbagai dokumen lainnya. |
| Penilaian atau Alokasi | Piutang usaha dinyatakan berdasarkan nilai realisasi neto. | Meninjau kembali akun yang jatuh tempo dan mengevaluasi alokasi yang cukup untuk akun yang tidak dapat diperbaiki. |
| Penyajian dan Pengungkapan | Berbagai kontinjensi yang tidak dilaporkan dalam akun keuangan diungkapkan secara baik dalam catatan atas laporan keuangan. | Mendapatkan informasi dari para pengacara entitas mengenai status litigasi dan perkiraan potensi kerugian. |

Tujuan audit dapat diklasifikasikan ke dalam dua kategori umum. *Pertama*, adalah tujuan audit seperti dalam tabel di atas yang berkaitan dengan transaksi dan saldo akun yang secara langsung memiliki dampak terhadap laporan keuangan.

*Kedua*, kategori yang berkaitan dengan sistem informasi, yang meliputi tujuan audit untuk menilai pengendalian atas operasi manual dan berbasis komputer yang digunakan dalam pemrosesan transaksi.

### C. Mendapatkan Bukti

Auditor mencari bukti yang mendukung penilaian manajemen. Dalam lingkungan TI, proses ini melibatkan pengumpulan bukti yang berkaitan dengan keandalan pengendalian operasi berbasis komputer serta isi *database* yang diproses oleh program komputer. Bukti dikumpulkan dengan melakukan pengujian pengendalian, yang berguna untuk menentukan apakah pengendalian internal berfungsi dengan baik, pengujian substantif, yang berguna untuk menetapkan apakah *database* akuntansi secara wajar mencerminkan transaksi dan saldo akun perusahaan.

### D. Memastikan Tingkat Kesesuaian dengan Kriteria yang telah Ditetapkan

Auditor harus menetapkan apakah berbagai kelemahan dalam pengendalian internal dan kesalahan penyajian yang ditemukan dalam berbagai transaksi serta saldo akun material atau tidak. Pada lingkungan semua jenis audit, menilai materialitas merupakan pendapat auditor. Namun, di lingkungan audit TI, keputusan ini semakin sulit dengan adanya teknologi dan struktur pengendalian internal yang canggih.

### E. Mengomunikasikan Hasil

Auditor harus mengomunikasikan berbagai hasil pengujiannya kepada pengguna. Auditor independen harus memberikan laporan ke komite audit dewan komisaris atau pemegang saham perusahaan. Laporan tersebut berisi, di antaranya, pendapat audit. Pendapat ini diterbitkan bersama dengan laporan keuangan ke pihak-pihak yang berkepentingan dari internal dan eksternal perusahaan. Auditor TI sering kali mengomunikasikan berbagai temuannya kepada auditor internal dan eksternal, yang kemudian dapat mengintegrasikan berbagai temuan ini dengan aspek non-TI dari audit terkait.

## 1.3 AUDIT TEKNOLOGI INFORMASI

Audit dalam lingkungan teknologi informasi berfokus pada aspek berbasis komputer dalam sistem informasi perusahaan. Audit ini meliputi penilaian implementasi, operasi, dan pengendalian sumber daya komputer.

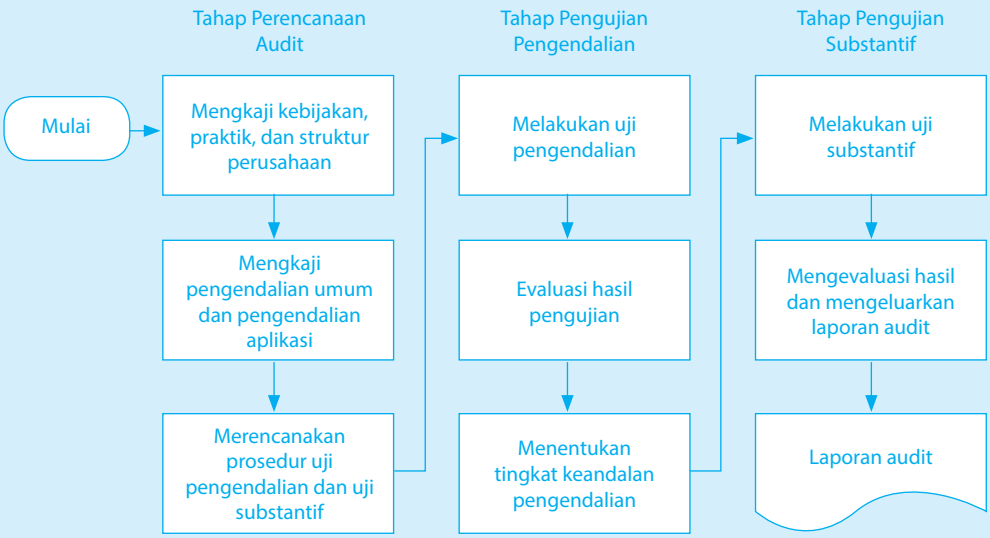
### 1.3.1 Lingkungan Teknologi Informasi

Sistem pengendalian internal yang efektif sangat dibutuhkan untuk melindungi integritas proses akuntansi dan data. Desain dan pengawasan atas sistem biasanya menjadi tanggung jawab akuntan, terutama auditor. Dengan sistem yang sudah menggunakan teknologi informasi, akan menambah kompleksitas desain pengendalian internal yang efektif.

Adanya pemusatan data dalam sistem informasi, serta adanya gabungan koneksi akses, akses jarak jauh, dan koneksi ke berbagai sistem atau komputer lainnya, hal ini akan menimbulkan lingkungan TI memiliki kompleksitas dalam desain pengendalian yang efektif. Semua pengguna terhubung ke sistem yang sama di mana *database* berada, menyimpan semua data, sehingga dimungkinkan timbul risiko akses yang tidak sah, pencurian, atau perusakan. Lingkungan sistem berbasis teknologi informasi akan meningkatkan aktivitas yang membahayakan sistem, *database*, dan aset. Lingkungan ini juga memudahkan pihak manajemen untuk melanggar pengendalian internal, dan hal ini dapat mengarah pada penipuan keuangan, pencurian aset yang dilakukan oleh karyawan, serta korupsi. Masalah komputer lainnya di tingkat perusahaan meliputi bencana, baik bencana alam atau yang disebabkan manusia, serta tidak berfungsinya sistem.

### 1.3.2 Struktur Audit Teknologi Informasi

Audit TI dibagi ke dalam tiga tahapan, yaitu perencanaan, pengujian pengendalian, dan pengujian substantif.




Gambar 1.1
Struktur Audit Teknologi Informasi

#### A. Perencanaan Audit

Sebelum auditor dapat menentukan sifat dan sejauh mana pengujian akan dilakukannya, maka auditor harus mendapatkan pemahaman secara lengkap mengenai bisnis kliennya. Tujuan auditor dalam perencanaan ini adalah untuk mendapatkan informasi yang cukup mengenai perusahaan agar dapat merencanakan tahapan audit. Bagian utama dari tahapan perencanaan adalah analisis risiko audit yang meliputi gambaran umum pengendalian internal perusahaan. Pada saat pengkajian pengendalian, auditor mencoba memahami kebijakan, praktik, dan

struktur perusahaan. Selain itu, auditor juga mengidentifikasi berbagai aplikasi dan usaha keuangan yang penting, untuk memahami pengendalian atas transaksi yang diproses oleh aplikasi tersebut.

Teknik untuk memperoleh atau mengumpulkan bukti pada tahapan ini adalah meliputi penyebaran kuesioner, wawancara dengan pihak manajemen, pengkajian dokumentasi sistem, dan pengamatan aktivitas yang ada. Selama proses ini, auditor TI harus mengidentifikasi berbagai eksposur (kelemahan) utama beserta pengendalian yang ada untuk mengurangi eksposur.

### B. Pengujian Pengendalian

Menentukan apakah ada pengendalian internal yang memadai dan berfungsi dengan baik. Untuk mencapai hal tersebut, auditor akan melakukan berbagai pengujian pengendalian. Teknik pengumpulan bukti yang digunakan dalam tahapan ini dapat secara manual atau dengan teknik komputer khusus.

Auditor akan menilai kualitas pengendalian internal, di mana tingkat keandalan yang dapat digunakan oleh auditor untuk pengendalian internal, memengaruhi sifat dan keluasan pengujian substantif yang harus dilakukan.

### C. Pengujian Substantif

Tahapan ketiga dalam proses audit ini fokus pada *database* (data keuangan). Tahapan ini melibatkan penyelidikan yang terperinci mengenai berbagai saldo akun dan transaksi melalui uji substantif. Misalnya, konfirmasi pelanggan adalah uji substantif yang kadang digunakan untuk memverifikasi saldo akun. Auditor akan memilih sampel saldo piutang usaha dan menelusurinya kembali ke sumbernya untuk menentukan apakah jumlah yang dicantumkan benar-benar utang pelanggan. Berdasarkan temuan dalam sampel ini, auditor akan dapat mengambil kesimpulan mengenai nilai wajar dari seluruh aset piutang usaha.

Dalam lingkungan TI, informasi yang dibutuhkan untuk melakukan uji substantif terdapat dalam *file database* yang sering kali diekstrak menggunakan perangkat lunak Computer-Assited Audit Tools and Tecgniques (CAATT). Pendekatan *database* dalam audit TI menggunakan CAATT dan uji substantif untuk meneliti integritas data dan keandalan data.

## 1.4 ALASAN MENGAPA AUDIT TEKNOLOGI INFORMASI DIPERLUKAN

Seiring dengan makin banyaknya institusi, baik pemerintahan maupun swasta, yang mengandalkan TI untuk mendukung jalannya operasional sehari-hari, maka kesadaran akan perlunya dilakukan *review* atas pengembangan sistem informasi semakin meningkat. Risiko yang mungkin ditimbulkan sebagai akibat dari gagalnya pengembangan sistem informasi, antara lain:

- Biaya pengembangan sistem melampaui anggaran yang ditetapkan.
- Sistem tidak dapat diimplementasikan sesuai dengan jadwal yang ditetapkan.

- Sistem yang telah dibangun tidak memenuhi kebutuhan pengguna.
- Sistem yang dibangun tidak memberikan dampak efisiensi dan nilai ekonomis terhadap jalannya operasi institusi, baik pada masa sekarang maupun masa datang.
- Sistem yang berjalan tidak menaati perjanjian dengan pihak ketiga atau memenuhi aturan yang berlaku.

Ron Weber, Dekan Fakultas Teknologi Informasi Monash University, dalam salah satu bukunya: *Information System Controls and Audit* menyatakan beberapa alasan penting mengapa audit TI perlu dilakukan, antara lain:

1. ***Kerugian akibat kehilangan data.***
   Saat ini, data telah menjadi salah satu aset terpenting bagi perusahaan. Bayangkan, jika seorang pimpinan perusahaan yang sebagian besar penjualan yang diraih dilakukan dengan cara kredit di mana para pembeli akan membayar tagihannya di kemudian hari. Untuk mencatat penjualan, diperlukan TI. Akibat terjadinya gangguan virus atau terjadi kebakaran pada ruangan komputer yang dimiliki, misalnya, maka seluruh data tagihan tersebut akan hilang.

   Kehilangan data tersebut mungkin akan mengakibatkan perusahaan tidak dapat melakukan penagihan kepada para pelanggan. Selain itu, kalaupun masih dapat dilakukan, waktu yang dibutuhkan menjadi sangat lama karena harus melakukan verifikasi manual atas dokumen penjualan yang miliki.
2. ***Kesalahan dalam pengambilan keputusan.***
   Banyak kalangan usaha yang saat ini telah menggunakan bantuan sistem pendukung keputusan (*decision support system*—DSS) untuk mengambil keputusan penting. Dalam bidang kedokteran, misalnya, keputusan dokter untuk melakukan tindakan operasi mungkin ditentukan dengan menggunakan bantuan perangkat lunak tersebut. Risiko yang mungkin dapat ditimbulkan apabila dokter salah memasukkan data pasien ke sistem TI yang digunakan. Taruhannya bukan lagi material, melainkan nyawa seseorang.
3. ***Risiko kebocoran data.***
   Data bagi sebagian besar sektor usaha merupakan sumber daya yang tidak ternilai harganya. Informasi mengenai pelanggan, misalnya, bisa jadi merupakan kekuatan daya saing suatu perusahaan. Bayangkan, seorang direktur perusahaan telekomunikasi yang memiliki 5 juta pelanggan. Tanpa disadari, satu per satu pelanggan perusahaan telah beralih ke perusahaan pesaing.

   Setelah melalui proses audit, akhirnya diketahui bahwa data pelanggan perusahaan telah jatuh ke tangan perusahaan pesaing. Berdasarkan data tersebut, perusahaan pesaing kemudian menawarkan jasa yang sama dengan jasa yang ditawarkan ke pelanggan yang sama, tetapi dengan biaya yang sedikit lebih rendah. Kebocoran data ini tidak saja berdampak terhadap kehilangan sejumlah pelanggan, tetapi lebih jauh lagi bisa mengganggu kelangsungan hidup perusahaan.

4. ***Penyalahgunaan komputer.***
Alasan lain perlunya dilakukan audit TI adalah tingginya tingkat penyalahgunaan komputer. Pihak-pihak yang dapat melakukan kejahatan komputer sangat beraneka ragam. Kita mengenal adanya *hackers* dan *crackers*. *Hackers* merupakan orang yang dengan sengaja memasuki suatu sistem teknologi informasi secara tidak sah. Biasanya mereka melakukan aktivitas *hacking* untuk kebanggaan diri sendiri atau kelompoknya, tanpa bermaksud merusak atau mengambil keuntungan atas tindakannya itu.

Sementara itu, *crackers* di sisi lain melakukan aktivitasnya dengan tujuan mengambil keuntungan sebanyak-banyaknya dari tindakannya tersebut, misalnya mengubah atau merusak atau bahkan menghancurkan sistem komputer.

Kejahatan komputer juga bisa dilakukan oleh karyawan yang merasa tidak puas dengan kebijakan perusahaan, baik yang saat ini masih aktif bekerja di perusahaan yang bersangkutan maupun yang telah keluar. Sayangnya, tidak semua perusahaan siap mengantisipasi adanya risiko-risiko tersebut.

Survei yang dilakukan oleh *Ernst & Young* menemukan bahwa 34% dari total perusahaan yang ada saat ini tidak memiliki mekanisme yang memadai untuk mendeteksi kemungkinan adanya serangan terhadap sistem mereka. Lebih dari 33%, bahkan menyatakan bahwa mereka tidak memiliki kemampuan yang cukup untuk menindaklanjuti ancaman yang mungkin timbul.

5. ***Kerugian akibat kesalahan proses perhitungan.***
Sering kali, TI digunakan untuk melakukan perhitungan yang rumit. Salah satu alasan digunakannya TI adalah kemampuannya untuk mengolah data secara cepat dan akurat (misalnya, penghitungan bunga bank). Penggunaan TI untuk mendukung proses penghitungan bunga bukannya tanpa risiko kesalahan. Risiko ini akan semakin besar, misalnya, ketika bank tersebut baru berganti sistem dari sistem yang sebelumnya digunakan. Tanpa adanya mekanisme pengembangan sistem yang memadai, mungkin terjadi kesalahan penghitungan atau, bahkan *fraud*. Kesalahan yang ditimbulkan oleh sistem baru ini akan sulit terdeteksi tanpa adanya audit terhadap sistem tersebut.

6. ***Tingginya nilai investasi perangkat keras dan perangkat lunak komputer.***
Investasi yang dikeluarkan untuk proyek TI sering kali sangat besar. Bahkan, dari penelitian yang pernah dilakukan tercatat bahwa 20% pengeluaran TI terbuang secara percuma, 30–40% proyek TI tidak mendatangkan keuntungan. Selain itu, sulit mengukur manfaat yang dapat diberikan TI.

Skala investasi dan biaya yang telah diinvestasikan pada saat ini maupun masa depan ke sistem informasi sangat besar. Berapa investasi yang ditanamkan oleh perusahaan seperti PT Telkom, PT Semen Gresik atau investasi yang ditanamkan dalam industri perbankan seperti Bank BNI, Bank BCA dan sebagainya?

Untuk Indonesia, alokasi anggaran untuk investasi di bidang TI relatif tidak lebih besar dibandingkan di luar negeri. Di Indonesia besarnya alokasi

anggaran berkisar 5-10%, sementara di luar negeri bisa mencapai 30% dari total anggaran belanja perusahaan. Namun, bila dilihat dari nilai absolut besarnya rupiah yang dikeluarkan, jumlahnya sangat besar. Perusahaan besar nasional, seperti Garuda Indonesia, Telkom, dan Pertamina semuanya, saat ini sudah menerapkan sistem ERP (*Enterprise Resource Planning*) dan bahkan berbagai aplikasi lainnya yang melibatkan investasi yang signifikan.

Untuk mengantisipasi hal itu, perusahaan menginginkan adanya *assurance* dari pihak yang berkompeten dan independen mengenai kondisi sistem TI yang akan atau sedang digunakan perusahaan. Pihak yang paling berkompeten dan memiliki keahlian untuk melakukan *review* tersebut adalah Auditor Sistem Informasi (Auditor TI).

Audit teknologi informasi bertujuan untuk mengevaluasi dan memperbaiki efektivitas proses manajemen risiko, pengendalian, dan *good governance*. Nilai penting dilakukannya audit TI sejalan dengan pentingnya mencapai tujuan perusahaan. Artinya, bagaimana perusahaan dapat mengelola berbagai risiko yang dihadapinya, terutama terkait dengan penerapan TI, dalam upayanya mencapai tujuan bisnisnya. "Jadi, bukan mencari siapa yang salah dan kesalahannya apa."

Kebanyakan perusahaan sudah merasa yakin dengan perencanaan penerapan dan pengembangan sistem TI. Keyakinan tersebut yang membuat umumnya banyak perusahaan menampik untuk melakukan audit TI. Selain ada juga pertimbangan yang menyangkut biaya pelaksanaan audit TI. "Pada prinsipnya audit TI hampir sama dengan audit keuangan. Namun, bedanya kalau audit keuangan sudah merupakan keharusan, sementara audit TI dilakukan hanya sebatas jika diperlukan." Perusahaan akan diuntungkan dengan dilakukannya audit teknologi informasi, karena dengan audit TI suatu perusahaan bisa didorong melakukan perencanaan dan pengembangan secara lebih terarah dan terfokus sesuai dengan tujuan bisnisnya. Perusahaan didorong untuk menerapkan secara tepat dan benar, bukan sekadar menggelar sistem yang mahal, tetapi tidak berdampak terhadap peningkatan kinerja karyawan dan perusahaan itu sendiri.

Di sisi lain, audit tersebut harus dilakukan terhadap sistem informasi secara keseluruhan, bukan pada perangkat TI yang digunakan. Selain itu, bukan hanya soal *software*, *hardware*, dan jaringannya. Audit dilakukan terhadap seluruh aspek yang terlibat dan relevan dalam sistem informasi. Meskipun menjadi tulang punggung, TI hanyalah salah satu aspek dari sistem informasi. Para profesional di bidang sistem informasi pasti mengetahui hal-hal mendasar semacam ini.

Misalnya, di bidang keamanan sistem informasi, ada beberapa prinsip nonteknis yang harus dipegang. Di kalangan profesional yang bergelut di bidang keamanan sistem informasi, ada prinsip yang dikenal sebagai prinsip multidisipliner (*multidisciplinary principle*), yang menegaskan bahwa segala macam pengukuran, praktik, dan prosedur keamanan sistem informasi harus melayani segala pertimbangan dan sudut pandang berbagai disiplin yang relevan, termasuk aspek sosial budaya, hukum dan politik. Ada juga prinsip demokrasi (*democracy principle*),

yang menegaskan bahwa keamanan sistem informasi perlu mempertimbangkan hak-hak pengguna dan pihak-pihak lain yang dipengaruhi oleh sistem tersebut.

Oleh karena itu, yang penting bukan "harus diaudit," tetapi perlu lebih jelas akan audit apa yang perlu dilakukan, siapa yang berkompeten melakukannya, dan mengapa audit TI sangat diperlukan?

Audit TI, seperti juga audit keuangan, sesungguhnya sangat dibutuhkan oleh perusahaan yang telah menerapkan TI untuk kepentingan usahanya. Namun, masih banyak perusahaan yang belum merasakan perlunya melakukan audit TI.

Ada sejumlah alasan utama yang muncul, misalnya, karena perusahaan merasa bahwa TI yang diterapkan masih berperan sebatas *support tools*, belum menjadi *strategic tools*." Selain itu, masih banyak perusahaan yang kebijakan dan tujuan penerapan TI tidak begitu jelas.

Alasan lainnya lebih menyangkut nilai investasi TI yang belum dianggap cukup berarti dibandingkan nilai keuangan perusahaan. Termasuk juga masih terdapatnya sejumlah jargon teknis TI yang sulit dipahami oleh manajemen puncak. Para teknisi atau spesialis TI sering kali berbicara menggunakan jargon teknis tanpa melihat perlunya membangun kesepahaman terlebih dahulu dengan lawan bicaranya. "Akibatnya lawan bicaranya langsung menolak untuk melanjutkan pembicaraan," jelasnya.

Tidak berarti semua itu negatif, karena resistensi terhadap dilakukannya audit TI juga bisa muncul karena tingginya tingkat kepercayaan perusahaan terhadap bagian TI. Apalagi kalau selama ini belum pernah ada masalah serius yang terjadi pada sistem TI, misalnya mengalami *crash*, terkena gangguan virus, hilangnya *file*, hingga dijebolnya sistem oleh para *hackers*.

Pekerjaan auditor TI ini belum banyak dikenal di Indonesia. Di samping itu, jumlah tenaga auditor TI yang menyandang sertifikasi internasional (Certified Information System Auditor—CISA) juga masih sangat terbatas.

Dari pengalamannya selama ini, perusahaan mulai mempertimbangkan kemungkinan dilakukannya audit TI, terutama setelah manajemen puncak mulai mempertanyakan beberapa hal yang dirasa cukup penting. Misalnya, mengenai ROI (*return on investment*) dari investasi perangkat dan sistem TI yang sudah dilakukan selama ini. Itu bila investasi TI mulai dirasakan membengkak, tetapi hasil atau daya gunanya justru tidak sebanding dengan yang diharapkan.

### 1.4.1 Kegagalan Penerapan TI dan Audit TI

Dalam berbagai kasus, kebanyakan manajer TI atau personal TI tidak bisa menjelaskan secara baik, melalui parameter kuantitatif, antara investasi TI dan peningkatan produktivitas karyawan. Biasanya, profesional TI cenderung menyebutkan hal itu sebagai hal yang *intangible*. Akibatnya, manajemen puncak dan senior terpaksa meraba-raba dan mencari tahu apa yang dimaksud dengan *intangible* itu. Tentunya, itupun setelah perusahaan berinvestasi TI dalam jumlah yang tidak bisa dipandang kecil.

Kegagalan bagian TI dalam menjelaskan hasil guna seperti yang diharapkan mendorong manajemen puncak mengambil langkah melakukan audit TI. Kegagalan tersebut biasanya terkait dengan tidak terpenuhinya ROI secara memadai. Selain itu, juga berkurangnya kredibilitas bagian TI di mana bagian-bagian lain di dalam perusahaan.

Isu lain yang juga mendorong dilakukannya audit TI adalah karena dirasakan kurang selarasnya kegiatan bagian TI dengan tujuan bisnis perusahaan secara keseluruhan. Di samping itu, dirasakan perlunya mengerem investasi TI yang sudah menggelembung lebih besar dari perencanaan awal.

Hal lain yang juga mengemuka, yang mendorong dilakukannya audit TI adalah masalah keamanan sistem internal, terutama ketika hendak dihubungkan dengan Internet. Bagi banyak perusahaan, hal itu jarang sekali diperhatikan.

Selama ini, Internet dianggap sebagai cara yang paling efektif dan efisien untuk berkomunikasi dengan kantor cabang atau pihak luar, termasuk pemasok barang dan jasa. Oleh karena Internet sesungguhnya merupakan *public domain*, maka faktor keamanannya perlu diperhatikan secara sungguh-sungguh. Tanpa memerhatikan faktor itu, *hackers* akan mudah masuk dan menjebol sistem, serta mengambil *file* penting perusahaan.

Saat ini, masalah keamanan juga dihadapi banyak perusahaan, antara lain menyangkut persoalan transaksi dan pembukuan keuangan perusahaan. Itu sebabnya persoalan ini menjadi sangat penting untuk diperhatikan. Pengabaian terhadap persoalan keamanan berarti juga membuka peluang bagi masuknya *hackers*, bukan hanya sekadar mengacak-acak f*ile* yang ada, tetapi juga mematai-matai dan bahkan mencuri sejumlah *file* penting lainnya. Tidak heran bila faktor keamanan menjadi salah satu fokus dalam audit TI.

Selama ini, kalau berbicara audit selalu tertuju langsung ke audit keuangan. Dengan demikian, ketika muncul isu Sistem Informasi Komisi Pemilihan Umum (SI-KPU) diminta untuk diaudit, karena dinilai lambat dan tidak efektif dalam penghitungan suara pemilih Pemilu 2004 dan dilanjutkan pada Pemilu 2009 di mana terdapat kekisruhan mengenai Data Pemilihan Tetap (DPT) yang tidak valid, maka asosiasi itupun muncul. SI-KPU harus segera diaudit. Kalangan pakar dan praktisi menyuarakan harus dilakukannya audit, tetapi yang dimaksud lebih pada audit sistem informasi (audit TI), sedangkan kalangan partai lebih melihat audit sebagaimana audit keuangan.

Permintaan auditnya sama, tetapi penekanannya berbeda, antara mengaudit kesiapan dan keandalan sistem informasi yang merujuk pada rencana, desain dan tujuan penerapannya, dengan audit pengadaan sistem informasi, yang lebih melihat pada aspek nilai pentingnya TI dalam proses penghitungan suara dan bagaimana pengadaannya, apakah sesuai dengan rencana dan spesifikasi semula.

Kontroversi mengenai audit tersebut memunculkan banyak pertanyaan yang tak sepenuhnya terjawab. Belum lagi, kalau yang dipersoalkan adalah payung hukum yang memungkinkan dilakukannya audit, karena hal itu terkait dengan pertanggungjawaban ke publik. Pada saat yang sama, muncul dugaan "kekisruhan"

tersebut malah akan semakin mendorong ketidakpercayaan masyarakat terhadap arti pentingnya teknologi informasi (TI), bukan saja dalam proses penghitungan suara Pemilu, melainkan juga dalam banyak aspek penerapannya di masyarakat, misalnya *e-Commerce*, *e-Government,* dan sebagainya.

Kalangan praktisi melihat bahwa audit TI sangat perlu dilakukan, karena dengan begitu akan semakin jelas dan terbuka masalah apa yang sebenarnya dihadapi, khususnya SI-KPU, sehingga menyebabkan lambannya proses penghitungan suara berbasis TI tersebut. *Clearance* semacam ini sangat diperlukan, bukan hanya dari sisi pengadaannya, melainkan terutama dari sistem informasinya, yang menyangkut banyak aspek, baik teknologi (perangkat keras dan perangkat lunak, PC, sistem jaringan, dan lainnya) maupun sumber daya manusia yang mengoperasikannya, serta sistem yang terkait dengan kelancaran proses penghitungan suara tersebut.

Sebagaimana penerapan TI di lingkungan perusahaan, SI-KPU juga mencakup jaringan luas, didukung oleh ribuan tenaga operator, yang mungkin tidak semuanya "siap" mengoperasikan PC dan sistem penghitungan suara, sehingga sejak awal mestinya telah memperhitungkan berbagai risiko yang bakal dihadapi, termasuk jaringan dan kemampuan transfer datanya.

Pelaksanaan audit TI sebenarnya lebih pada bagaimana kita mengelola risiko yang mungkin muncul dari penerapan TI. Artinya, risikonya ada, tetapi bagaimana risiko itu disadari, tentunya dengan didukung oleh sistem TI yang secara nyata siap dijalankan, sehingga risiko dapat dikelola dengan baik, misalnya diantisipasi, dicarikan jalan alternatifnya jika terjadi sesuatu.

## 1.5 BAGAIMANA AUDIT TEKNOLOGI INFORMASI DILAKUKAN?

Audit TI merupakan proses pengumpulan dan evaluasi bukti-bukti untuk menentukan apakah sistem komputer yang digunakan telah dapat melindungi aset milik organisasi, mampu menjaga integritas data, dapat membantu pencapaian tujuan organisasi secara efektif, serta menggunakan sumber daya yang dimiliki secara efisien. Audit TI sendiri merupakan gabungan dari berbagai macam ilmu, antara lain: *Traditional Audit*, Sistem Informasi Manajemen, Sistem Informasi Akuntansi, Ilmu Komputer, dan *Behavioral Science*. Seperti audit keuangan, audit TI juga memiliki metode dan teknik yang andal, yang memungkinkan diperolehnya hasil evaluasi yang kredibel.

Dalam praktiknya, tahapan-tahapan dalam audit TI tidak berbeda dengan audit pada umumnya. Tahapan perencanaan, sebagai pendahuluan, mutlak perlu dilakukan agar auditor mengenal benar objek yang akan diperiksa. Selain itu, tentunya, auditor dapat memastikan bahwa *qualified resources* sudah dimiliki, dalam hal ini aspek SDM yang berpengalaman dan juga referensi praktik terbaik (*best practices*). Tahapan perencanaan ini akan menghasilkan program audit yang didesain sedemikian rupa, sehingga pelaksanaannya akan berjalan efektif dan efisien, dan dilakukan oleh orang-orang yang kompeten, serta dapat diselesaikan sesuai dengan waktu yang disepakati.

Dalam pelaksanaannya, auditor TI mengumpulkan bukti-bukti yang memadai melalui berbagai teknik termasuk survei, interviu, observasi, dan *review* dokumentasi (termasuk tinjauan *source code* bila diperlukan).

Satu hal yang unik, bukti audit yang diambil oleh auditor biasanya mencakup pula bukti elektronis (data dalam bentuk *file softcopy*). Biasanya, auditor TI menerapkan teknik audit berbantuan komputer, disebut juga dengan CAAT (Computer Aided Auditing Technique). Teknik ini digunakan untuk menganalisis data, misalnya data transaksi penjualan, pembelian, transaksi aktivitas persediaan, aktivitas nasabah, dan lain-lain.

Sesuai dengan standar auditing ISACA (Information Systems Audit and Control Association), selain melakukan pekerjaan lapangan, auditor juga harus menyusun laporan yang mencakup tujuan pemeriksaan, sifat dan kedalaman pemeriksaan yang dilakukan. Laporan ini juga harus menyebutkan organisasi yang diperiksa, pihak pengguna laporan yang dituju dan batasan distribusi laporan. Laporan juga harus memasukkan temuan, kesimpulan, rekomendasi sebagaimana layaknya laporan audit pada umumnya.

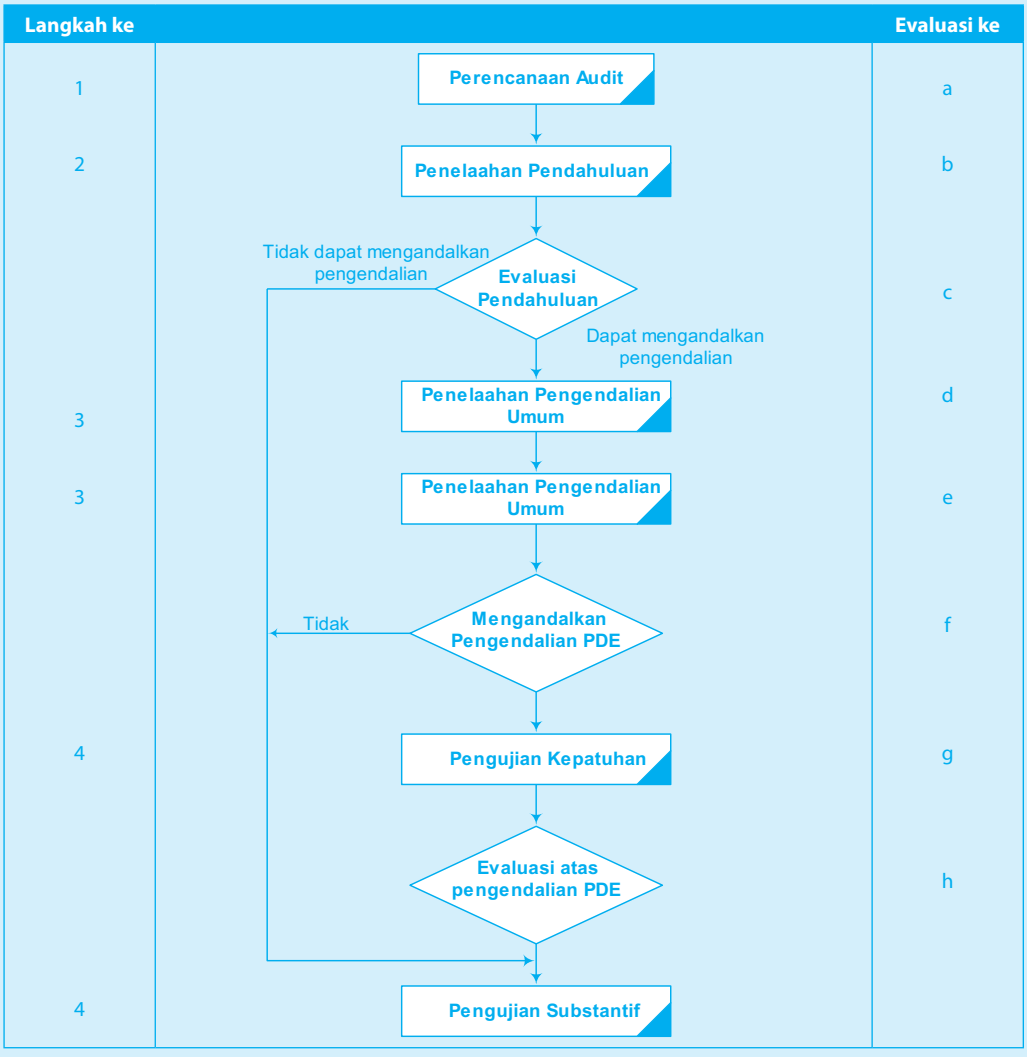
## Studi Kasus

Bank swasta terkemuka menunjuk tim audit TI, *Ernst & Young,* untuk melakukan *review* atas penerapan sistem perbankan yang terintegrasi. Berikut ini adalah contoh-contoh pemeriksaan yang dilakukan.

1. ***Manajemen Proyek***
   Melakukan *review* atas manajemen proyek untuk memastikan bahwa semua *outcome* yang diharapkan tertuang dalam rencana proyek. Pada tahapan ini, auditor TI melakukan *review* atas *project charter*, sumber daya yang akan digunakan, alokasi penugasan dan analisis tahapan pekerjaan proyek.
2. ***Desain Proses dan Pengendalian Kontrol Aplikasi***
   *Review* mengenai desain pengendalian dalam modul-modul Perbankan tersebut, yaitu pinjaman dan tabungan. Untuk itu dilakukan *review* atas desain proses di mana auditor mengevaluasi proses, risiko dan pengendalian mulai dari tahapan *input*, proses, maupun *output*.
3. ***Desain Infrastruktur***
   *Review* ini mencakup analisis efektivitas dan efisiensi desain infrastruktur pendukung (*server*, *workstation*, sistem operasi, *database*, dan komunikasi data). Hasil *follow-up* dijadikan dasar oleh manajemen untuk memulai implementasi sistem perbankan yang terintegrasi tersebut. Berdasarkan nilai tambah yang diberikan melalui rekomendasi pada fase pertama, perusahaan menunjuk kembali auditor untuk melakukan fase *review* kedua secara paralel pada saat implementasi dilakukan, yaitu *review* terhadap:
   - Migrasi data, pada saat "*roll-out*" ke cabang-cabang, termasuk kapasitas pemrosesan dan penyimpanannya.
   - Aspek lainnya termasuk persiapan *help-desk*, *contingency,* dan *security*.

- Kesiapan pengguna dalam menggunakan sistem ini, kualitas pelatihan yang diberikan dan dokumentasi pengguna (*user manual*).
- Prosedur-prosedur manajemen perubahan (*change management*) dan *testing.*

Auditor selanjutnya diminta memberikan saran mengenai risiko yang masih tersisa, sebelum manajemen memutuskan sistem barunya dapat "*go-live.*" Sedangkan langkah-langkah untuk melakukan audit TI bisa dilihat pada Gambar 1.2 yang menggambarkan urutan langkah-langkah yang dilakukan dalam audit TI.




Gambar 1.2
Langkah-Langkah Audit TI

## 1.5.1 Perencanaan Audit

Standar pekerjaan pertama menurut SPAP (Standar Profesional Akuntan Publik) menyatakan bahwa auditor harus merencanakan pengauditan dengan sebaik-baiknya. Perencanaan sangat penting untuk alasan-alasan berikut.

1. Perencanaan memungkinkan bagi auditor untuk memperoleh bukti yang kompeten dan cukup, dan bukti kompeten ini selanjutnya dapat memperkecil kewajiban hukum dan menjaga reputasinya auditor.
2. Perencanaan memungkinkan bagi auditor untuk dapat melaksanakan pengauditan secara efisien dengan biaya yang memadai.
3. Perencanaan memungkinkan bagi auditor untuk menghindari kesalahpahaman yang dapat timbul dengan pihak-pihak yang diperiksa.

Berikut hal yang perlu dipertimbangkan auditor dalam merencanakan pengauditan.

1. Memperoleh pemahaman yang menyeluruh mengenai kondisi kegiatan usaha dan industri organisasi yang akan diperiksa serta kegiatan (operasinya).
2. Mengidentifikasi alasan mengapa auditor memerlukan pengauditan tersebut.
3. Memperoleh informasi mengenai kewajiban hukum organisasi yang akan diperiksa serta informasi penting yang harus diperiksa, seperti kontrak kerja, hasil RUPS, dan sebagainya.
4. Mengantisipasi seberapa jauh struktur pengendalian internal organisasi yang akan diperiksa dapat diandalkan serta pengauditan yang akan dihadapi.
5. Menilai atau mengantisipasi permasalahan yang mungkin timbul dari laporan keuangan serta kondisi yang mungkin memerlukan perluasan atau modifikasi prosedur pengauditan.
6. Mengantisipasi jenis laporan akuntan (*audit report*) yang akan dikeluarkan.
7. Membuat program pemeriksaan yang menunjukkan langkah-langkah pengauditan yang akan ditempuh untuk mencapai tujuan pengauditan yang telah ditetapkan.

### 1.5.2 Pemahaman terhadap Lingkungan Sistem Informasi Berbasis Komputer

Dalam *audit arround computer* atau pengauditan konvensional langkah ini tidak dilakukan karena auditor menggunakan prinsip "jika masukan dan keluarannya benar, maka *ouput*-nya berarti benar." Namun demikian, *audit arround computer* dianggap riskan karena perubahan teknologi sistem informasi berbasis komputer yang semakin kompleks dan terintegrasi. Tahapan kedua, salah satu atau lebih auditor mengetahui mengenai konsep sistem informasi berbasis komputer. Pada tahapan ini pengetahuan atau kompetensi auditor tentang PDE atau sistem informasi berbasis komputer diperlukan, baik auditor tersebut seorang spesialis komputer atau yang memiliki keahlian auditing dan akuntansi, tetapi sudah dilatih dengan pengetahuan tentang sistem informasi berbasis komputer. Pemahaman terhadap konsep sistem informasi berbasis komputer selain merupakan pengetahuan atau kompetensi lain yang disyaratkan oleh SPAP juga berguna bagi auditor dalam memahami sistem akuntansi yang berbasis komputer dan dalam memilih serta menerapkan prosedur audit organisasi tersebut secara memadai.

### 1.5.3 Mengevaluasi Pengendalian Internal

Kegiatan ini sering kali dikatakan mengakses risiko pengendalian, yaitu menilai efektivitas kebijakan dan prosedur struktur pengendalian internal dalam mencegah atau mendeteksi salah saji (*misstatement*). Dalam mengakses risiko pengendalian tersebut auditor dapat melakukan pengujian terhadap pengendalian yang ada.

### 1.5.4 Pelaksanaan Pengujian Kepatuhan dan Pengujian Substantif

Tujuan dari pengujian kepatuhan adalah untuk menentukan apakah sistem pengendalian internal berjalan sebagaimana yang dikehendaki. Sementara itu, pengujian substantif dimaksudkan untuk memvalidasi bahwa transaksi tertentu telah diotorisasi secara memadai, disertai bukti pendukung dan telah dicatat. Selain itu, pos-pos yang dicatat tersebut merupakan hasil dari transaksi yang telah diotorisasi, disertai bukti pendukung dan diklasifkasikan dengan benar. Davis, Adams, dan Schaller menyebutkan adanya lima jenis substantif tes yang dapat digunakan dalam instalasi pengolahan data. Kelima jenis pengujian substantif tersebut adalah sebagai berikut.

1. Pengujian untuk mengidentifikasi pemrosesan yang salah.
2. Pengujian untuk mengakses kualitas data.
3. Pengujian untuk mengidentifikasi data yang tidak konsisten.
4. Pengujian untuk membandingkan data dengan perhitungan fisik.
5. Mengonfirmasikan data dengan yang berasal dari sumber ekstern.

Pengujian kepatuhan dan pengujian substantif dapat dilakukan dengan atau tanpa bantuan program komputer atau biasa disebut dengan istilah *computer assisted audit techniques* (CAAT) atau teknik audit berbantuan komputer (TABK).

### 1.5.5 Penyelesaian Audit

Tahap terakhir dalam setiap audit adalah penyampaian laporan audit (*audit report*) sesuai dengan penugasan dan tujuan audit yang dilakukan. Dalam laporan ini dikemukakan apa yang telah dilakukan oleh auditor yang bersangkutan serta kesimpulan yang diambilnya. Bagi auditor fungsional, laporan audit merupakan media untuk menyatakan tujuan dan ruang lingkup pemeriksaannya serta melaporkan temuan dan kesimpulan audit berikut saran tindak (rekomendasi) perbaikannya. Audit operasional terhadap aktivitas sistem berbasis komputer juga menggunakan pola seperti ini yang biasanya terstandardisasi, yaitu auditor harus menyebutkan mengenai kondisi, kriteria, sebab-akibat, dan rekomendasi yang disarankan.

Sementara itu bagi auditor independen, menerbitkan laporan bukan hanya mengikuti standar pelaporan sebagaimana yang dipersyaratkan Standar Profesional Akuntan Publik, melainkan tidak jarang hanya laporan ini yang dapat dilihat oleh para penggunanya.

Dengan kata lain, laporan ini adalah produk dari auditor yang bersangkutan yang dapat digunakan oleh pengguna laporan keuangan. Bagi auditor independen juga diiwajibkan untuk memberikan opini mengenai kewajaran laporan keuangan.

Saran tindak (rekomendasi) perbaikan atau jenis-jenis opini auditor dalam audit finansial tetap sama, baik dalam objek yang diaudit menggunakan atau tidak menggunakan komputer dalam memproses data bisnis. Jenis-jenis opini ini adalah sebagai berikut.

1. ***Pendapat wajar tanpa pengecualian.***
   Dalam jenis opini ini auditor menyatakan bahwa laporan keuangan menyajikan secara wajar, dalam semua hal yang material, posisi keuangan, hasil usaha, dan arus kas satuan usaha tertentu sesuai dengan prinsip akuntansi yang berlaku umum.
2. ***Pendapat wajar tanpa pengecualian dengan bahasa penjelasan yang ditambahkan dalam laporan audit bentuk baku.***
   Pendapat semacam ini diberikan apabila karena suatu keadaan tertentu auditor harus menambahkan paragraf atau bahasa penjelasan yang lain dalam laporan auditnya, meskipun hal tersebut tidak memengaruhi pendapat wajar tanpa pengecualian atas laporan keuangan yang diauditnya.
3. ***Pendapat wajar dengan pengecualian (qualified opinion).***
   Dalam jenis pendapat ini auditor menyatakan bahwa laporan keuangan telah menyajikan secara wajar, dalam semua hal yang material, posisi keuangan, hasil usaha, dan arus kas satuan usaha tertentu sesuai dengan prinsip akuntansi yang berlaku umum, kecuali untuk dampak hal-hal yang berhubungan dengan yang dikecualikan.
4. ***Pendapat tidak wajar (adverse opinion).***
   Dalam pendapat tidak wajar seorang auditor menyatakan bahwa laporan keuangan tidak menyajikan secara wajar posisi keuangan, hasil usaha, dan arus kas satuan usaha tertentu sesuai dengan prinsip akuntansi yang berlaku umum.
5. ***Pernyataan tidak memberikan pendapat (disclaimer).***
   Dalam pernyataan tidak memberikan pendapat auditor menyatakan bahwa ia tidak memberikan pendapat atas laporan keuangan.

## 1.6 KAPAN AUDIT TEKNOLOGI INFORMASI DILAKUKAN?

*Best Practice* menyarankan agar dalam proses pengembangan sistem informasi yang signifikan, perlu dilakukan *review*, baik sebelum atau pada saat implementasi (*pre-implementation system*) maupun setelah sistem "*live*" (*post-implementation system*).

Manfaat *pre-implementation review* adalah sebagai berikut.

- Institusi dapat mengetahui apakah sistem yang telah dibuat sesuai dengan kebutuhan ataupun memenuhi kriteria keberterimaan (*acceptance criteria*).
- Mengetahui apakah pengguna telah siap menggunakan sistem tersebut.
- Mengetahui apakah *outcome* sesuai dengan harapan manajemen.

Manfaat *post-implementation review* adalah sebagai berikut.

- Institusi mendapat masukan (*input*) atas risiko yang masih ada dan saran untuk penanganannya.
- Masukan tersebut dimasukkan dalam agenda penyempurnaan sistem, perencanaan strategis dan anggaran pada periode berikutnya.
- Bahan untuk perencanaan strategis dan rencana anggaran di masa datang.
- Memberikan *reasonable assurance* bahwa sistem informasi telah sesuai dengan kebijakan atau prosedur yang telah ditetapkan.
- Membantu memastikan bahwa jejak audit (*audit trail*) telah diaktifkan dan dapat digunakan oleh manajemen, auditor, atau pihak lain yang berwenang untuk melakukan pemeriksaan.
- Membantu dalam penilaian apakah *initial proposed values* telah terealisasi dan saran tindak lanjutnya.

Di sisi lain, sistem TI tidak hanya terkait dengan aspek teknologi, melainkan lebih luas dari itu. Aspek sumber daya manusia yang mengoperasikan, hal-hal pendukung lainnya, yang secara langsung akan terkait dengan keberhasilan sistem TI dalam mencapai tujuan penerapannya, merupakan bagian dari sistem yang harus diperhitungkan dengan baik sejak awal penerapannya. Oleh karena itu, keberhasilan sistem mestinya dilihat dari tujuan yang ingin dicapai melalui penerapan TI, sehingga tidak dapat memisahkan bahwa keberhasilan sistem TI hanya dilihat semata-mata dari aspek teknologinya, melainkan keseluruhan yang membentuk sistem itu.

Pelaksanaan audit TI itu sendiri sebenarnya sudah banyak dilakukan di lingkungan perusahaan, terutama yang menyadari arti pentingnya penerapan TI bagi pencapaian tujuan bisnis perusahaan. Sebagian besar perusahaan memiliki kebijakan dan tujuan penerapan TI yang tidak begitu jelas. Selain itu, masih banyak juga yang beranggapan bahwa nilai investasi TI yang ditanamkan belum cukup berarti dibandingkan nilai keuangan perusahaan sehingga mereka belum berpikir perlunya dilakukan audit TI.

Kepentingan audit TI bukan sekadar melakukan penilaian dan evaluasi, tetapi audit tersebut juga harus sejalan dengan tujuan perusahaan. Bagaimana dengan dilakukannya audit TI, perusahaan dapat terbantu dalam mencapai tujuan bisnisnya. Artinya, bagaimana perusahaan dapat mengelola risiko agar tidak terlalu *overload*, terlalu agresif, atau terlalu ekspansif. Sebaliknya, dengan audit TI perusahaan dapat melakukan *internal control*.

Memang terjadi penolakan terhadap dilakukannya audit TI, misalnya, sebagaimana yang terjadi di beberapa perusahaan, meski penerapan TI sudah cukup meluas dan intensif, tidak serta merta berarti negatif. Oleh karena yang dimaksud lebih pada audit TI yang dilakukan pihak ketiga, misalnya, auditor TI eksternal. Hal itu mungkin karena tingginya tingkat kepercayaan perusahaan terhadap kemampuan bagian TI dalam melakukan pengendalian internal. Apalagi selama ini belum pernah ada masalah serius yang terjadi terhadap sistem TI, misalnya mengalami *crash*, terkena gangguan virus, hilangnya *file*, hingga dijebolnya sistem oleh para *hackers*.

Audit TI sendiri sebenarnya bukan semata-mata melakukan penilaian atau evaluasi terhadap sistem yang telah berjalan, yang ketika mengalami masalah, baru kemudian dirasakan perlunya melakukan audit. Itu sebabnya, auditor TI yang memiliki pengalaman cukup lama, sebaiknya audit TI dilakukan sebelum sistem TI dijalankan. Mengapa? Karena, dengan begitu penerapan TI dapat dilihat dari kesesuaiannya dengan rencana dan upaya pencapaian tujuan perusahaan, dan pada saat yang sama, dapat diketahui apakah sistem telah siap dan mampu memenuhi harapan itu.

Berbeda halnya, jika audit TI dilakukan setelah sistem TI berjalan. "Karena, selain mungkin saja ada penolakan, mengauditnya pun tidak mudah, karena bisnis yang dilakukan harus tetap berjalan," tambah Isnaeni. Jika pada awal, misalnya sejak perencanaan dan pengembangan sistem, maka sejak dini pula akan diketahui apakah sistem TI yang dibangun sudah pas, sudah layak untuk menjalankan aplikasi yang diinginkan, yang tentunya semua itu dalam upaya perusahaan untuk mencapai tujuannya. Jika belum, misalnya, maka perusahaan dapat mengambil tindakan yang diperlukan, sebelum sistem TI benar-benar dijalankan.

## 1.7 KOMPONEN AUDIT TEKNOLOGI INFORMASI

Audit TI sedikitnya mencakup enam komponen yang sangat esensial: pendefinisian tujuan perusahaan: penentuan isu, tujuan dan perspektif bisnis antara penanggung jawab bagian dengan bagian TI; *review* terhadap pengorganisasian bagian TI yang meliputi perencanaan proyek, status dan prioritasnya, *staffing levels*; belanja TI; *IT change process management*: penilaian (*assessment*) infrastruktur teknologi, *assessment* aplikasi bisnis; temuan, dan laporan rekomendasi. Sedangkan subjek yang perlu diaudit mencakup aspek keamanan, keandalan, kinerja, dan kemampuan mengelola (*manageability*).

Subjek audit TI lebih terfokus pada keamanan, keandalan, kinerja, dan *manageability*. Masalah keamanan mencakup tidak hanya keamanan *file server* dan penerapan metode cadangan, melainkan juga penerapan standar tertentu, seperti C-ICT. Keandalan meliputi penerapan RAID V disk subsystems untuk *server* dengan *critical applications* dan prosedur penyimpanan data pada *file server*, bukan pada *drive* lokal C. Kinerja mencakup persoalan standardisasi PC, penggunaan LAN serta cadangan yang sesuai dengan beban kerja.

Sementara *manageability* menyangkut penerapan standar tertentu dan pendokumentasian secara teratur dan berkesinambungan. "Payahnya, orang Indonesia ini tidak begitu peduli terhadap pendokumentasian, sehingga melakukan perubahan tanpa dokumentasi apapun haruslah dihindari sejak awal," jelasnya.

Setelah audit TI dilakukan terdapat sejumlah *assessment* yang masih harus dikerjakan guna memenuhi persyaratan dan standar yang sudah disepakati bersama. Dari pengalaman diketahui bahwa *assessment* audit tidak hanya terhadap konfigurasi sistem dan jaringan yang diterapkan, melainkan juga organisasi bagian TI.

## Studi Kasus pada PT Tiga Raksa

Konfigurasi sistem dan jaringan PT Tiga Raksa terfokus pada konsep dan penerapan WAN, Internet, *e-mail system* serta pendokumentasiannnya. Konsep WAN jelas dan tersentralisasi pada kantor cabang di seluruh Indonesia. Jaringannya memiliki kecapatan yang cukup tinggi, yaitu 1.000 Megabit, agar bisa mengimbangi beban kerjanya. Sebelumnya kantor perwakilan perusahaan memiliki empat *e-mail address* domain yang terpisah. Namun, setelah diaudit keempatnya dilebur menjadi satu. Koneksi ke Internet yang tadinya terbuka diubah melalui satu *gateway* yang telah dilengkapi *firewall*.

Untuk *server*, *assessment* audit mewajibkan penerapan konfigurasi RAID. Ini dilakukan terutama untuk *main critical applications*. Dengan konfigurasi ini terdapat minimal 3 *disk*, sehingga bila salah satunya gagal atau *crash*, maka 2 *disk* lainnya akan menggantikan sambil menunggu perbaikan. Ini untuk mencegah terjadinya kehilangan data penting perusahaan. *Assessment* terhadap *server* juga menghendaki penerapan skema cadangan C-ICT dan perangkat lunak pengendali jarak jauh C-ICT.

Untuk PC dan *printer*, *assessment* menghendaki virus *scanner* dilakukan secara otomatis ketika pengguna *log-in*. Penyimpanan data tidak lagi dilakukan pada *drive* lokal C, tetapi langsung pada *file server* kecuali pada kantor cabang yang dilakukan dengan CD-RW. *Assessment* juga menghendaki C-ICT *common desktop application* terhadap semua PC yang ada, dan dibatasi usia PC sehingga tidak ada yang obsolet (usang) dari sisi teknologi. "Dahulu ketika datang pertama kalinya, masih ada teman yang masih menggunakan PC berprosesor 486 dan Pentium I, padahal saat itu umumnya di pasar sudah ada PC berprosesor Pentium III," ungkapnya.

Hal yang sering terlupakan adalah pengaturan *printer* sebagai alat bantu. Kebanyakan perusahaan berusaha menyediakan *printer* untuk hampir setiap pegawainya. Namun, hal itu tidak dilakukan di sini. *Assessment* audit menyarankan perlunya berbagai *printer* untuk satu grup kerja tertentu, sehingga mengurangi keengganan karyawan menggunakan printer di luar keperluan kerja.

*Assessment* terhadap perangkat lunak dilakukan dengan menyeragamkan sistem operasi (OS). "Saat itu diputuskan untuk mengambil Windows 2000 *server based* dengan *service pack* terbaru," jelas Nofrins. Keseragaman diperlukan agar tidak hanya efisien, tetapi juga mudah dalam perawatannya.

Pengoraganisasian bagian TI juga ditetapkan dalam audit *assessment*. Ini terbagi atas IT management, IT support dan IT staffing. Untuk pertama kalinya diperkenalkan visi jangka panjang mengenai IT management yang merujuk pada tujuan bisnis perusahaan. Ini didukung visi *business support* yang jelas dan orientasinya dipersiapkan untuk penerapan ERP (*enterprise resource planning*) sebagai infrastrukturnya. Selain itu, tanggung jawab dibebankan pada setiap karyawan pengguna, sedang manajemen TI lebih bertanggung jawab dalam mendukung dan memecahkan masalah yang muncul.

Untuk IT support, *assessment* menghendaki terjadinya restrukturisasi organisasi TI dengan pendekatan operasional yang baru. Penekannya pada karyawan yang

terlatih dengan baik, sehingga cepat tanggap dan terbuka terhadap semua keluhan yang muncul.

Namun, yang terpenting bukan terletak pada organisasi dan *staffing* bagian TI, melainkan karena audit TI masih terbatas pada fokus utama dan yang terkait dengan penerapan dan pengembangan sistem TI dalam suatu organisasi atau perusahaan. Namun, masalahnya adalah bagaimana mensosialisasikannya ke seluruh bagian dan staf lainnya di dalam organisasi tersebut.

Di sini justru letak tantangan terbesarnya, yakni bagaimana menyebarluaskan dan mengubah kebiasaan pengguna TI. "Jangan sampai muncul keluhan dari teman atau pengguna betapa sulitnya mengingat *password*, terutama karena seringnya ganta-ganti *password*."

## 1.8 AUDIT TEKNOLOGI INFORMASI DAN REGULASI

Industri keuangan, misalnya perbankan, lebih dahulu mengenal audit TI dibanding banyak industri lainnya. Maklum, industri ini memang dikenal sebagai *the most computerized industries*. Itu sebabnya industri perbankan sangat mengenal dan sebagian besar telah akrab dengan audit TI.

Kebutuhan untuk melakukan audit TI di kalangan perbankan lebih disebabkan karena kebutuhan dasarnya. Hal ini mirip dengan kebutuhan industri terhadap audit keuangan. Itu sebabnya Bank Indonesia, sebagai bank sentral, juga mendorong dikeluarkannya regulasi tentang audit TI. Menurut Mohamad Ishak, Direktur Direktorat Akunting dan Sistem Pembayaran Bank Indonesia, sebelum regulasi dikeluarkan BI sebenarnya sudah menyadari pentingnya dilakukan audit TI untuk kalangan perbankan nasional.

Mungkin hal itu dipicu oleh rawannya masalah keamanan TI, sehingga sewaktu-waktu memungkinkan sistem dibobol oleh *hacker*. Namun, terlepas dari masalah itu, ternyata tidak hanya persoalan sekuriti yang menjadi titik fokus. Akan tetapi, juga menyangkut sistem dan perangkat, jaringan, dan manajemen TI yang menyeluruh dan berkesinambungan dari waktu ke waktu. Ini berguna tidak hanya untuk kepentingan perbankan semata-mata, melainkan juga melindungi kepentingan nasabah.

Jauh sebelum regulasi audit TI dikeluarkan, BI sebenarnya sudah mendorong dilakukannya audit TI di kalangan perbankan nasional. Tampaknya, BI ingin membiarkan munculnya *awareness* yang sungguh-sungguh dari pengelola perbankan nasional akan pentingnya audit TI daripada meregulasinya secara terburu-buru dan menimbulkan resistensi yang kontraproduktif.

Selain itu, BI tampaknya juga berhati-hati dalam regulasinya, karena terdapat perbedaan akan standar dan prosedur audit TI. Sebagai bank sentral, BI tentu ingin mempelajari lebih jauh mengenai perbedaan dan keunggulan serta kelemahan masing-masing standar tersebut, sebelum mengakomodasikannya ke dalam regulasi yang akan ditetapkan secara mengikat ke seluruh kalangan perbankan nasional.

Namun, tampaknya apa yang dilakukan BI justru cukup tepat mengingat kalangan perbankan nasional saat ini sudah sedemikian akrab dengan audit TI dan menggangapnya sebagai kebutuhan mendasar. Artinya, audit dilakukan secara berkala dan bukan sekadar sebagai reaksi sesaat.

## SOAL DAN STUDI KASUS

1. Apa tujuan audit TI?
2. Apa definisi audit?
3. Sebutkan perbedaan antara auditor internal dan auditor eksternal!
4. Apa saja ke empat elemen utama yang dijelaskan dalam definisi audit teknologi informasi?
5. Sebutkan dan jelaskan tiga struktur di dalam audit TI!
6. Mengapa perusahaan atau organisasi memerlukan audit TI? Jelaskan!
7. Sebutkan secara berurutan langkah-langkah yang dilakukan dalam kegiatan audit TI?
8. Apa yang dimaksud dengan kegagalan TI?

# BAB 2

# PENGENDALIAN INTERNAL DI LINGKUNGAN SISTEM INFORMASI

Setelah mempelajari bab ini, Anda diharapkan mampu:

- Memahami berbagai tujuan pengendalian internal yang diatur dalam Statement on Auditing Standards (SAS) 78.
- Memahami bagaimana berbagai fitur unik lingkungan komputer harus dipertimbangkan untuk mencapai tujuan pengendalian yang disebutkan dalam SAS 78.
- Mengetahui berbagai area utama risiko dalam lingkungan TI.
- Mengetahui berbagai macam klasifikasi pengendalian internal.
- Mengetahui dan memahami pengendalian di lingkungan TI.
- Memahami peranan COBIT dalam penyelenggaraan *good governance* TI.

Selain perencanaan *(planning),* pengorganisasian *(organizing)* dan pelaksanaan *(actuating),* pengendalian *(controlling)* adalah juga merupakan fungsi manajemen untuk mencapai tujuan organisasi. Dalam perencanaan, manajemen memikirkan mengenai alternatif-alternatif tindakan, memilih yang paling sesuai dan menjabarkannya dalam bentuk ramalan dan anggaran. Untuk menyesuaikan dengan perubahan yang terjadi, manajemen tidak jarang harus mengubah perencanaan tersebut. Pekerjaan ini dikerjakan sebagai fungsi pelaksanaan sebagaimana disebut di atas. Sementara itu dalam pelaksanaan fungsi pengendalian, manajemen membandingkan kinerja yang sebenarnya dengan yang seharusnya dan mengukur selisihnya. Hal yang terakhir ini mungkin mengarah pada tindakan korektif bila memang diperlukan. Dengan adanya pengendalian maka manajemen dapat menghindari atau mengurangi risiko kegagalan dalam melakukan aktivitas bisnis.

Pada bab ini akan membahas mengenai masalah pengendalian internal yang menjadi dasar SAS 78. Pada bagian lain dari bab ini akan mengulas mengenai kerangka kerja untuk penaksiran risiko yang mengidentifikasi tujuan area risiko komputer. Pada bagian terakhir adalah pembahasan umum mengenai audit teknologi informasi dengan beberapa kasus yang ada.

## 2.1 APA ITU PENGENDALIAN INTERNAL?

Pembentukan dan pemeliharaan sistem pengendalian internal adalah kewajiban pihak manajemen yang penting. Aspek fundamental dari tanggung jawab pelayanan pihak manajemen adalah untuk memberikan para pemegang saham suatu jaminan yang wajar bahwa bisnis telah cukup terkendali. Selain itu, pihak manajemen memiliki tanggung jawab untuk melengkapi pemegang saham dan calon inverstor lainnya dengan informasi keuangan yang dapat diandalkan secara tepat waktu. Sistem pengendalian internal yang memadai sangat dibutuhkan agar pihak manajemen dapat melaksanakan berbagai kewajibannya.

Pengendalian internal terdiri atas kebijakan, praktik, dan prosedur yang digunakan oleh perusahaan untuk mencapai empat tujuan umum, yaitu:

a. Mengamankan aset;
b. Memastikan akurasi dan keandalan catatan dan informasi akuntansi;
c. Menyebabkan efisiensi dalam operasi perusahaan.
d. Mengukur kepatuhan dengan berbagai kebijakan dan prosedur yang ditetapkan oleh pihak manajemen.

Di dalam tujuan pengendalian terdapat empat asumsi penjelas yang menjadi petunjuk bagi para desainer dan auditor sistem pengendalian internal, yaitu:

***a. Tanggung Jawab Pihak Manajemen***

Konsep ini meyakini bahwa pembentukan dan pemeliharaan sistem pengendalian internal adalah tanggung jawab manajemen.

***b. Jaminan yang Wajar***

Sistem pengendalian internal harus memberikan jaminan yang wajar bahwa keempat tujuan umum pengendalian internal telah terpenuhi. Kewajaran di sini artinya adalah tidak ada sistem pengendalian internal yang sempurna dan biaya untuk mencapai pengendalian yang lebih baik tidak boleh melebihi manfaatnya.

***c. Metode Pemrosesan Data***

Sistem pengendalian internal harus mewujudkan keempat tujuan umumnya apapun metode pemrosesan data yang digunakan. Teknik tertentu yang digunakan untuk mewujudkan keempat tujuan tersebut akan berbeda, bergantung pada jenis teknologi yang berbeda.

***d. Keterbatasan***

Setiap sistem memiliki keterbatasan dalam hal efektivitasnya. Keterbatasan ini meliputi (1) kemungkinan terjadinya kesalahan—tidak ada sistem yang sempurna, (2) pembelotan (*circumvention*), yaitu tindakan kecurangan yang dilakukan oleh karyawan dengan membelokkan sistem melalui kolusi, (3) pengesampingan pihak manajemen—pihak manajemen mengabaikan semua prosedur pengendalian dengan secara pribadi menyimpangkan transaksi atau dengan mengarahkan bawahan untuk melakukan hal tersebut, dan (4) kondisi yang berubah-ubah—kondisi dapat berubah sepanjang waktu hingga pengendalian internal efektif yang ada mungkin menjadi tidak efektif lagi.

### 2.1.1 Kebutuhan Manajemen terhadap Pengendalian

Manajemen menganggap pengendalian sebagai sesuatu yang penting dan harus ada di dalam organisasi antara lain karena alasan-alasan berikut.

1. Untuk membuat kinerja sistem organisasi berjalan secara efektif. Pengendalian, sebagaimana disebutkan di atas, adalah membandingkan kinerja yang sebenarnya dengan yang seharusnya. Kinerja yang seharusnya ini dinyatakan dalam perencanaan. Selanjutnya, dengan adanya perencanaan ini maka manajemen telah membuat arah untuk mencapai tujuan. Oleh karena itu, apabila tujuan tersebut tercapai maka dikatakan efektif.
2. Agar dapat menghindari kerugian yang material dan penyimpangan dari tujuan yang telah ditetapkan. Kemungkinan kerugian dan penyimpangan akan selalu ada. Dengan demikian, tanggung jawab manajemen adalah memperkecil atau bahkan menghilangkan, bila mungkin, kemungkinan terjadinya kerugian dan penyimpangan tersebut. Selain itu, apabila kerugian dan penyimpangan tersebut benar-benar terjadi, maka kewajiban manajemen lainnya adalah membatasi pengaruh keadaan tersebut pada organisasi. Caranya adalah dengan menerapkan struktur pengendalian internal yang memadai.

Kerugian dan penyimpangan yang mungkin terjadi *(potential loss)* dalam suatu organisasi dapat berbentuk salah satu atau lebih dari hal-hal sebagai berikut.

- Data yang tidak dapat diandalkan sehingga laporan yang dibuat menjadi salah dan/atau menyesatkan.
- Pengolahan data menjadi tidak layak.
- Menyimpang dari prinsip akuntansi yang umum.
- Tidak dapat mencapai tujuan organisasi atau bahkan mengakibatkan terhentinya kegiatan usaha.
- Manajemen salah dalam membuat keputusan.
- Timbulnya kecurangan dan penyalahgunaan.
- Timbulnya tuntutan hukum.
- Timbulnya/biaya yang berlebihan, inefisiensi operasi atau hilangnya penerimaan.
- Hilang atau rusaknya aset dan catatan yang dimiliki.
- Hilangnya daya saing perusahaan.

Hal lain yang tidak dapat dihilangkan sama sekali dan hanya dapat dihindari atau dikurangi adalah risiko. Dengan adanya pengendalian maka manajemen dapat menghindari atau mengurangi risiko kegagalan dalam melakukan aktivitas bisnis. Untuk mengadakan pengendalian semacam itu bukan berarti tidak ada biayanya, bahkan beberapa pengendalian lebih mahal dibandingkan dengan pengendalian lainnya. Oleh karena itu, biaya pengendalian tersebut harus sebanding dengan manfaat yang akan diperoleh dari pengendalian tersebut, yaitu dibandingkan dengan nilai dari hal yang dikendalikan tersebut, yang dikendalikan dari kesalahan (*error*), penyalahgunaan (*fraud*), atau kemungkinan kerugian yang akan terjadi.

Untuk memperkecil risiko kesalahan auditor dalam membuat laporan audit ini maka standar auditing mengharuskan kepada auditor untuk antara lain mengevaluasi pengendalian internal dan menguji kepatuhan terhadapnya. Dalam evaluasi ini, auditor menilai kualitas pengendalian tersebut dalam artian seberapa efektif pengendalian tersebut membatasi penyebab terjadinya kerugian dan/atau seberapa baik pengendalian tersebut mengurangi kerugian yang terlanjur terjadi. Apabila dalam evaluasi tersebut ditemukan hal-hal yang belum baik, maka standar auditing juga mewajibkan kepada auditor untuk merekomendasikan dilakukannya perbaikan terhadap pengendalian tersebut.

Oleh karena alasan-alasan di atas maka menjadi sangat penting bagi manajemen untuk menciptakan pengendalian dan bagi auditor untuk mereviunya. Dalam kaitannya dengan sistem informasi berbasis komputer, AICPA menyebutkan adanya 19 pengendalian pokok sebagai berikut.

- Departemen PDE/Sistem Informasi harus dipisahkan dari departemen pengguna.
- Personel pada departemen PDE tidak diizinkan untuk memulai *(originate)* atau mengotorisasi transaksi, bertanggung jawab terhadap aset non-PDE ataupun memulai perubahan *file* induk *(master file).*
- Fungsi-fungsi dalam departemen PDE harus dipisahkan dengan baik.

- Prosedur untuk perancangan sistem, termasuk perolehan paket perangkat lunak, harus disertai dengan partisipasi aktif dari perwakilan para pengguna, dan, bila memungkinkan, partisipasi dari departemen akuntansi dan auditor internal.
- Masing-masing sistem harus memiliki spesifikasi tertulis yang ditelaah dan disetujui oleh manajemen dengan tingkatan yang memadai serta oleh departemen pengguna yang bersangkutan.
- Pengujian sistem harus merupakan usaha bersama antara pengguna dengan personel PDE, dan harus mencakup fase-fase sistem secara manual maupun yang dikomputerisasikan.
- Persetujuan akhir harus diperoleh sebelum mengoperasikan sistem yang baru tersebut.
- Semua konversi *file* induk dan *file* transaksi harus dikendalikan untuk mencegah timbulnya perubahan yang tidak ada otorisasinya sehingga dapat diperoleh hasil yang akurat dan lengkap.
- Setelah sistem yang baru berjalan, semua perubahan program harus disetujui sebelum diimplementasikan guna menentukan apakah perubahan tersebut ada otorisasinya, diuji dan didokumentasikan.
- Manajemen harus menetapkan beberapa jenis dokumentasi dan prosedur formal untuk mendefinisikan sistem dengan perincian yang cukup.
- Karakteristik (*feature*) pengendalian yang ada di dalam perangkat keras komputer, sistem operasi dan perangkat lunak pendukung lainnya harus dimanfaatkan semaksimal mungkin untuk mengendalikan pelaksanaannya dan untuk mencegah dan melaporkan ketidakberesan (*malfunctions*) perangkat keras.
- Perangkat lunak sistem harus dikendalikan sebagaimana manajemen melakukan pengendalian terhadap pemasangan dan perubahan program aplikasi.
- Akses ke dokumentasi program harus dibatasi hanya kepada personel yang membutuhkan untuk melaksanakan pekerjaannya.
- Akses ke *file* data dan program harus dibatasi hanya kepada personel yang berhak untuk memproses atau memelihara sistem tertentu.
- Akses ke perangkat keras komputer harus dibatasi hanya kepada personel yang berhak (yang memiliki otorisasi).
- Fungsi pengendalian harus bertanggung jawab untuk penerimaan data yang akan diproses, untuk memastikan bahwa seluruh data tersebut telah dibukukan, untuk menindaklanjuti kesalahan yang ditemukan selama pengolahan guna melihat bahwa kesalahan tersebut telah dikoreksi dan dikirimkan kembali oleh pihak-pihak yang berwenang, serta untuk melakukan verifikasi terhadap pendistribusian keluaran yang memadai.
- Pedoman sistem dan prosedur yang terlulis harus dibuat untuk seluruh aktivitas komputer yang memberikan otorisasi khusus atau umum kepada manajemen untuk memproses transaksi.

- Auditor internal atau kelompok independen lainnya dalam organisasi harus menelaah dan menilai usulan sistem pada tahap-tahap kritis pengembangan sistem tersebut.
- Secara rutin auditor internal atau kelompok independen lainnya di dalam organisasi harus menelaah dan menguji aktivitas pengolahan data.

Sementara itu, Clowes menyebutkan adanya jenjang dalam pengendalian (*control layering concept*) di mana masing-masing jenjang berfungsi untuk mencegah, mendeteksi, mengoreksi dan/atau menghambat (*recovery barrier*) terjadinya penyimpangan berupa kecurangan (*fraud*), bencana (*disaster*), kegagalan perangkat keras (*hardware failure*), kesalahan operator mesin, kesalahan *input,* dan kesalahan program. Jenis-jenis pengendalian yang ada dalam jenjang atau lapisan yang melindungi sistem tersebut adalah pengendalian program (*program controls*), pengendalian perubahan program (*control program changes*), pengendalian pengujian (*testing controls*), audit internal, asuransi terhadap bencana yang mungkin timbul, pengamanan ruang komputer, pengendalian kualitas, pemisahan tugas, serta pengendalian produksi dan peralatan (*production and machine controls*).

Pengendalian tersebut secara keseluruhan dimaksudkan agar sistem menjadi terlindungi dengan baik. Artinya, apabila sistem tersebut dikendalikan dengan berbagai metode yang berlapis-lapis maka masing-masing sumber penyimpangan sebagaimana disebutkan di atas (kecurangan, bencana, kegagalan perangkat keras, dan sebagainya) akan dapat diinterupsi dan ditolak oleh jenis-jenis pengendalian komputer tersebut. Dari pemikiran ini muncul prinsip dasar pengendalian yang berkali-kali dan ada penggantinya bila yang satu tidak berfungsi *(overlapping, redundant, and compensating controls).*

### 2.1.2 Pengendalian Internal dan Struktur Pengendalian Internal

Pengendalian yaitu pengendalian internal dan struktur pengendalian internal. Pengendalian internal adalah seluruh kebijakan, prosedur, dan praktik akuntansi yang dibuat oleh manajemen untuk membantu dalam melindungi organisasi dari kesalahan (*error*) dan penyalahgunaan (*fraud*).

Di Amerika Serikat, penerapan pengendalian internal dalam organisasi merupakan keharusan karena dinyatakan dalam undang-undang, yaitu yang disebut dengan Foreign Corrupt Practices Act tahun 1977. Oleh karena itu, tidak mengherankan apabila American Institute of Certified Public Accountants (AICPA) dalam *Statement on Auditing Standard No. 1,* mendefinisikan pengendalian internal menjadi sebagai berikut dengan mendasarkan pada undang-undang tersebut.

Pengendalian internal terdiri dari rencana organisasi serta seluruh metode koordinasi dan pengukuran yang diterapkan oleh perusahaan untuk menjaga aset, menguji keakuratan dan keandalan data akuntansi, mendukung efisiensi operasional serta mendorong dipatuhinya kebijakan manajerial yang telah ditetapkan.

Definisi dari AICPA tersebut menjelaskan pengendalian internal sebagai suatu kerangka kerja tertentu untuk mencapai tujuan tertentu pula. Kerangka kerja tersebut terdiri dari rencana organisasi serta metode dan alat-alat pengukuran lainnya yang digunakan oleh organisasi untuk:

1. Menjaga aset organisasi yang bersangkutan.
2. Menguji keakuratan dan keandalan data akuntansi.
3. Mendukung efisiensi operasional.
4. Mendorong dipatuhinya kebijakan manajerial yang telah ditetapkan.

Dari rencana organisasi, metode dan alat-alat pengukuran lainnya tersebut, pengendalian internal yang utama meliputi hal-hal sebagai berikut.

- Adanya pemisahan tugas yang memadai.
- Adanya dokumentasi dan catatan yang memadai.
- Adanya otorisasi yang memadai dari manajemen.
- Adanya pengendalian yang memadai atas aset dan catatan.
- Adanya penilaian yang independen terhadap kinerja para pegawai.
- Adanya pegawai yang kompeten.
- Adanya uraian tugas.
- Adanya struktur organisasi yang baik dengan garis wewenang dan tanggung jawab yang jelas.
- Adanya pengelolaan (manajemen) yang baik dengan tingkat integritas yang tinggi.

Sementara itu, struktur pengendalian adalah seluruh kebijakan dan prosedur yang ditetapkan oleh manajemen untuk memperoleh kepastian bahwa tujuan organisasi yang telah ditetapkan oleh manajemen tersebut akan dapat tercapai. Dalam Standar Profesional Akuntan Publik, struktur pengendalian ini dibagi menjadi lima unsur, yaitu lingkungan pengendalian, penaksiran risiko, aktivitas pengendalian, informasi dan komunikasi serta pemantauan. (SA Seksi 319 [PSA No. 69] Paragraf 07). Kelima komponen ini sama dengan lima komponen yang dinyatakan oleh Committee of the Sponsoring Organizations (COSO) dari Komisi Treadway yang dibentuk oleh AICPA yang dinyatakan dalam *COSO Report.* Uraian mengenai kelima unsur tersebut sebagai berikut.

## A. Lingkungan Pengendalian

Lingkungan pengendalian (*control environment*) mencakup seluruh tindakan, kebijakan, dan prosedur yang merefleksikan atau menggambarkan seluruh sikap dari manajemen, direktur, dan pemilik satuan usaha tentang pengendalian internal yang dapat menimbulkan kesadaran bagi para anggota organisasi tersebut mengenai pentingnya pengendalian semacam itu bagi satuan usaha yang bersangkutan. Sebagai contoh, apabila manajemen dengan sengaja mengesampingkan kebijakan dan/atau prosedur yang ada, maka lingkungan pengendalian organisasinya akan sangat

terpengaruh karena para pegawainya akan menganggap pengendalian tersebut tidak terlalu penting sehingga mereka cenderung akan mengabaikan juga.

Sebagian dari lingkungan pengendalian ini dapat dikendalikan oleh manajemen dengan menggunakan kebijakan dan prosedur tertentu, seperti misalnya hal-hal sebagai berikut.

1. Penggunaan anggaran dan laporan keuangan sebagai sarana untuk memformulasikan dan mengomunikasikan tujuan, perencanaan dan kegiatan perusahaan yang bersangkutan.
2. Penggunaan pegawai yang saling menguji (*check and balance*) untuk memisahkan kegiatan yang tidak boleh digabung (tidak kompatibel) serta untuk mengadakan penyeliaan (supervisi) oleh tingkatan manajemen yang lebih tinggi.
3. Seberapa jauh pengendalian terhadap penggunaan metode pengolahan data serta terhadap pengembangan dan pemeliharaan sistem informasi berbasis komputer oleh perusahaan yang bersangkutan.

Struktur pengendalian internal menurut IAI, sebagaimana menurut *COSO Report,* mencakup beberapa subkomponen yang terdiri dari unsur-unsur sebagai berikut.

1. Integritas dan nilai etis yang harus dimiliki oleh seluruh anggota organisasi. Kewajiban dari manajemen, direktur, dan pemilik satuan usaha untuk menciptakan kebijakan dan/atau situasi di mana setiap pegawai tidak boleh melakukan tindakan yang tidak jujur, ilegal atau tidak etis.
2. Pertimbangan pada keahlian yang dibutuhkan untuk melaksanakan pekerjaan *(commitment to competence).* Artinya, manajemen mempertimbangkan bahwa untuk mencapai tujuan organisasi maka tingkat pengetahuan dan keahlian dalam melakukan pekerjaan dijadikan dasar penentu bagi manajemen untuk mengangkat seseorang.
3. Partisipasi Dewan Direksi dan Komisi Audit. Selain dari siapa yang akan menjadi anggota Dewan Direksi dan Komisi Audit, bagaimana cara kerja dan independensi mereka terhadap manajemen juga sangat berpengaruh terhadap lingkungan pengendalian.
4. Falsafah dan gaya kepemimpinan (gaya bekerja) dari manajemen. Sebagaimana dalam komponen struktur pengendalian internal menurut IAI, falsafah dan gaya kepemimpinan manajemen seperti bagaimana gaya mereka dalam mengambil keputusan dan/atau dalam memantau risiko bisnis sangat berpengaruh terhadap lingkungan pengendalian.
5. Struktur organisasi. Dengan adanya struktur organisasi yang memadai maka satuan usaha dapat mencapai tujuan organisasi yang bersangkutan karena struktur organisasi dapat memberikan kerangka kerja yang baik untuk merencanakan, melaksanakan, mengendalikan, dan memantau aktivitas satuan usaha yang bersangkutan.

6. Penetapan otoritas dan tanggung jawab sehingga setiap pegawai dapat mengetahui siapa yang berwenang dan bertanggung jawab mengenai aktivitas dalam organisasi yang bersangkutan.
7. Kebijakan dan praktik mengenai sumber daya manusia. Pengendalian internal menjadi efektif maka harus dibuat kebijakan mengenai sumber daya manusia sehingga dapat diperoleh kepastian bahwa personel satuan usaha tersebut memiliki integritas, nilai etika, dan kompetensi pada tingkatan yang dikehendaki.

Dalam kaitannya dengan lingkungan pengendalian ini IAI menghendaki agar auditor memperoleh pengetahuan yang memadai tentang lingkungan pengendalian guna memahami sikap, kesadaran dan tindakan manajemen dan dewan komisaris terhadap lingkungan pengendalian internal, dengan mempertimbangkan substansi pengendalian maupun dampaknya secara kolektif. Auditor harus memusatkan pada substansi pengendalian daripada bentuk luarnya, karena pengendalian mungkin dibangun tetapi tidak dilaksanakan.

## B. Penaksiran Risiko

Penaksiran risiko entitas untuk tujuan pelaporan keuangan merupakan identifikasi, analisis, dan manajemen terhadap risiko yang relevan dengan penyusunan laporan keuangan yang wajar sesuai dengan prinsip akuntansi yang berlaku umum di Indonesia. Sebagai contoh, penaksiran risiko dapat ditujukan mengenai bagaimana entitas mempertimbangkan kemungkinan transaksi tidak dicatat atau mengidentifikasi dan menganalisis estimasi yang dicatat dalam laporan keuangan. Risiko yang relevan dengan pelaporan keuangan yang andal juga berkaitan dengan peristiwa dan transaksi khusus.

Risiko yang relevan dengan pelaporan keuangan mencakup peristiwa dan keadaan intern dan ekstern yang mungkin terjadi dan secara negatif berdampak terhadap kemampuan entitas untuk mencatat, mengolah, meringkas, dan melaporkan data keuangan konsisten dengan asersi manajemen dalam laporan keuangan. Sekali risiko diidentifikasi, manajemen mempertimbangkan signifikan atau tidaknya, kemungkinan terjadinya, dan bagaimana hal itu dikelola. Manajemen dapat membuat rencana, program, atau tindakan yang ditujukan ke risiko tertentu atau dapat memutuskan untuk menerima suatu risiko karena pertimbangan biaya atau yang lain. Risiko dapat timbul atau berubah karena keadaan seperti berikut ini.

- Perubahan dalam lingkungan operasi. Perubahan dalam lingkungan peraturan dan operasi dapat mengakibatkan perubahan dalam tekanan persaingan dan risiko yang berbeda secara signifikan.
- Personel baru. Personel baru mungkin memiliki fokus yang berbeda atas atau pemahaman terhadap pengendalian internal.

- Sistem informasi baru atau yang diperbaiki. Perubahan signifikan dan cepat dalam sistem informasi dapat mengubah risiko berkaitan dengan pengendalian internal.
- Pertumbuhan yang pesat. Perluasan operasi yang signifikan dan cepat dapat memberikan tekanan terhadap pengendalian dan meningkatkan risiko kegagalan dalam pengendalian.
- Teknologi baru. Pemasangan teknologi baru ke dalam operasi atau sistem informasi dapat mengubah risiko yang berhubungan dengan pengendalian internal.
- Lini produk, produk, atau aktivitas baru. Dengan masuk ke bidang bisnis atau transaksi yang di dalamnya entitas belum memiliki pengalaman dapat mendatangkan risiko baru yang berkaitan dengan pengendalian internal.
- Restrukturisasi korporat. Restrukturisasi dapat disertai dengan pengurangan staf dan perubahan dalam supervisi dan pemisahan tugas yang dapat mengubah risiko yang berkaitan dengan pengendalian internal.
- Operasi luar negeri. Perluasan atau pemerolehan operasi luar negeri membawa risiko baru atau sering kali risiko yang unik yang dapat berdampak terhadap pengendalian internal, seperti risiko tambahan atau risiko yang berubah dari transaksi mata uang asing.
- Penerbitan standar akuntansi baru. Pemakaian prinsip akuntansi baru, atau perubahan prinsip akuntansi dapat berdampak terhadap risiko dalam penyusunan laporan keuangan.

## C. Aktivitas Pengendalian

Aktivitas pengendalian (*control activities*) adalah kebijakan dan prosedur tambahan selain dari empat komponen lainnya yang dimaksudkan untuk membantu memberikan jaminan bahwa tindakan yang harus dilakukan benar-benar telah dilaksanakan dalam mencapai tujuan organisasi yang bersangkutan. Sebagaimana komponen prosedur pengendalian dalam kategori struktur pengendalian internal menurut IAI yang juga merupakan prosedur pengendalian tambahan, aktivitas pengendalian ini dapat berbentuk apa saja, tetapi biasanya berkaitan dengan salah satu dari lima kategori umum, sebagai berikut.

1. Otorisasi yang memadai atas transaksi dan kegiatan.
2. Adanya pemisahan tugas yang memadai.
3. Adanya dokumentasi dan pencatatan yang memadai.
4. Adanya pengendalian yang memadai atas akses dan penggunaan aset perusahaan dan catatan.
5. Adanya pengecekan atas kinerja yang dilakukan secara independen.

## D. Informasi dan Komunikasi

Tujuan dari sistem informasi dan pelaporan akuntansi dari satuan usaha adalah untuk mengidentifikasi, menggabungkan, mengklasifikasikan, menganalisis, mencatat, dan

melaporkan transaksi satuan usaha yang bersangkutan serta untuk mempertahankan akuntabilitas aset dan kewajiban yang terkait.

Agar sistem informasi ini menjadi efektif maka sistem informasi tersebut harus:

1. mengidentifikasi dan mencatat transaksi yang valid;
2. mengidentifikasi dan mencatat seluruh transaksi yang valid tersebut;
3. memberikan jaminan bahwa aset dan kewajiban yang sudah dicatat adalah dari hasil transaksi yang menyebabkan timbulnya hak atau kewajiban perusahaan;
4. mengukur nilai transaksi sedemikian rupa sehingga memungkinkan untuk mencatat nilai moneter transaksi tersebut secara memadai dalam laporan keuangan;
5. mencakup perincian semua transaksi secara memadai sehingga memungkinkan untuk disajikan dalam laporan keuangan, termasuk klasifikasi yang memadai dan pengungkapan yang perlu disajikan dalam laporan keuangan.

Dalam kaitannya dengan informasi dan komunikasi ini, IAI menghendaki agar auditor memperoleh pengetahuan yang memadai mengenai sistem informasi yang relevan dengan pelaporan keuangan dengan tujuan untuk memahami hal-hal sebagai berikut.

1. Golongan transaksi dalam operasi entitas yang signifikan bagi laporan keuangan.
2. Bagaimana transaksi tersebut dimulai.
3. Catatan akuntansi, informasi pendukung, dan akun tertentu dalam laporan keuangan yang tercakup dalam pengolahan dan pelaporan transaksi.
4. Pengolahan akuntansi yang dicakup sejak saat transaksi dimulai sampai dimasukkan ke dalam laporan keuangan, termasuk alat elektronik (seperti komputer dan *electronic data interchange*) yang digunakan untuk mengirim, memproses, memelihara, dan mengakses informasi.

## E. Pemantauan (Monitoring)

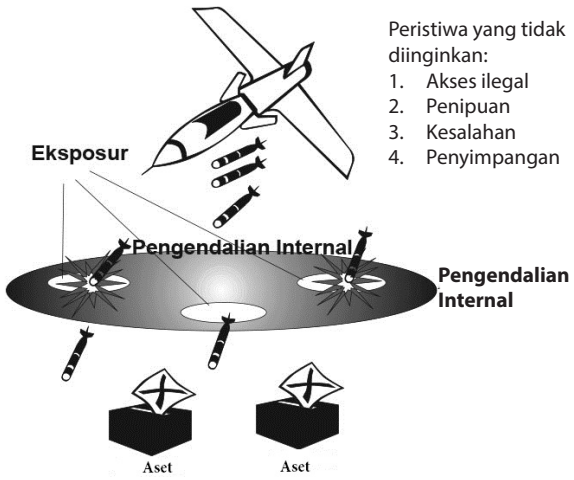
Kegiatan pemantauan berkaitan dengan penilaian atas kualitas kinerja struktur pengendalian internal yang dilakukan oleh manajemen untuk menentukan bahwa pengendalian yang sudah ditetapkan tersebut memang benar-benar dilaksanakan sesuai dengan tujuannya semula serta untuk menentukan bahwa pengendalian tersebut memang benar-benar perlu diperbaiki agar sesuai dengan berubahnya kondisi yang ada. (SA Seksi 319 [PSA No. 39] Paragraf 37–39). Kegiatan ini dapat dilakukan dengan berbagai cara dan sumber seperti menilai pengendalian yang ada, mempelajari laporan auditor, reviu atas laporan pengecualian (*exception report*), melalui saran dari anggota organisasi atau rapat pimpinan, bahkan dapat pula dilakukan melalui keluhan dari pelanggan.

Dalam kaitannya dengan pemantauan ini, IAI menghendaki agar auditor memperoleh pengetahuan yang memadai mengenai "tipe utama aktivitas entitas yang digunakan untuk memantau pengendalian internal terhadap pelaporan keuangan, termasuk bagaimana aktivitas tersebut digunakan untuk melaksanakan tindakan koreksi."

## 2.3 EKSPOSUR DAN RISIKO

Risiko adalah potensi acaman yang dapat membahayakan penggunaan atau nilai berbagai aset perusahaan. Ketiadaan atau kelemahan pengendalian tersebut dapat dikatakan sebagai eskposur.

Gambar 2.1 menggambarkan sistem pengendalian internal sebagai tameng yang melindungi aset perusahaan dari berbagai peristiwa tidak diinginkan yang menghujani perusahaan. Peristiwa yang tidak diinginkan meliputi usaha untuk mengakses secara tidak sah aset perusahaan; penipuan yang dilakukan oleh perorangan baik dari luar maupun dari dalam perusahaan; kesalahan karena karyawan yang tidak kompeten, program komputer yang salah, dan data yang di-*input* rusak; serta tindakan yang menyimpang, seperti akses tidak sah oleh *hacker* komputer dan ancaman dari berbagai virus komputer yang menghancurkan program dan *database*.




Gambar 2.1
Tameng Pengendalian Internal

Eksposur yang digambarkan sebagai lubang pada tameng pengendalian internal, meningkatkan risiko perusahaan mengalami kerugian keuangan atau kerusakan akibat dari berbagai peristiwa yang tidak diinginkan. Kelemahan di dalam pengendalian internal dapat mengekspos perusahaan ke salah satu atau lebih dari berbagai jenis risiko berikut ini.

a. Kerusakan aset.
b. Pencurian aset.
c. Korupsi informasi atau sistem informasi.
d. Gangguan atas sistem informasi.

### Penaksiran Risiko (Risk Assessment)

Untuk tujuan pelaporan keuangan, akses terhadap risiko menunjukkan tentang identifikasi, analisis dan pengelolaan risiko perusahaan yang berkaitan dengan pembuatan laporan keuangan sesuai dengan standar akuntansi yang berlaku. Pentingnya manajemen memperhitungkan risiko yang dapat membuat perusahaan tidak mencapai tujuannya atau bahkan dapat menimbulkan kebangkrutan.

Akses manajemen terhadap risiko bisnis perusahaannya pada dasarnya menyerupai akses auditor terhadap pengendalian, yaitu bahwa manajemen mengakses risiko sebagai bagian dari perancangan dan pelaksanaan pengendalian internal untuk meminimalkan kesalahan dan penyalahgunaan, sedangkan auditor mengekases risiko untuk menentukan bukti yang diperlukan dalam pelaksanaan pekerjaan auditnya. Bila manajemen merespons atau dapat mengakses risiko secara efektif, maka auditor biasanya akan dapat mengumpulkan bukti dalam jumlah yang lebih sedikit dibandingkan dengan bila manajemen gagal dalam melakukannya.

Dalam kaitannya dengan penaksiran risiko ini, IAI menghendaki agar auditor memperoleh pengetahuan yang memadai mengenai "proses penaksiran risiko entitas untuk memahami bagaimana manajemen mempertimbangkan risiko yang relevan dengan tujuan pelaporan keuangan dan memutuskan tentang tindakan yang ditujukan risiko tersebut. Pengetahuan ini mungkin mencakup pemahaman tentang bagaimana manajemen mengidentifikasi risiko, melakukan estimasi risiko yang signifikan, menaksir kemungkinan terjadinya, dan menghubungkannya dengan pelaporan keuangan.

## 2.4 KLASIFIKASI PENGENDALIAN INTERNAL

Pembahasan mengenai komponen struktur pengendalian internal di atas menunjukkan bahwa pengendalian internal dapat diklasifikasikan menjadi beberapa kriteria. Untuk mempermudah pembahasan, dalam buku ini pengendalian internal diklasifikasikan menjadi beberapa kategori, yaitu (1) berdasarkan saat dilakukannya pengendalian atau menurut waktunya; (2) berdasarkan sifat pengendalian tersebut atau menurut sifatnya; (3) berdasarkan tujuan yang hendak dicapai atau menurut tujuannya; dan (4) berdasarkan kategori atau menurut klasifikasi lainnya. Uraian tentang keempat klasifikasi pengendalian ini tampak pada beberapa subbagian berikut.

### 2.4.1 Pengendalian Menurut Waktunya

Menurut waktu dilakukannya pengendalian tersebut, maka dapat dikelompokkan menjadi tiga jenis pengendalian, adalah sebagai berikut.

### A. Pengendalian sebelum terjadinya suatu kegiatan

Pengendalian yang dilakukan sebelum terjadinya suatu kegiatan disebut juga dengan istilah pengendalian pratindakan (*pre-action control* atau *precontrol*). Tujuan pengendalian ini adalah untuk memperoleh keyakinan bahwa segala sesuatunya telah sesuai dengan ketentuan yang berlaku *sebelum* suatu kegiatan dilaksanakan, sehingga manajemen dapat mencegah suatu masalah sebelum timbul. Jenis pengendalian ini bertujuan preventif. Contoh dari pengendalian pratindakan ini misalnya persetujuan kepala bagian keuangan terhadap setiap pengeluaran kas. Apabila tidak ada tanda tangan kepala bagian keuangan tersebut maka kasir tidak diperkenankan untuk melakukan pembayaran.

### B. Pengendalian selama berlangsungnya kegiatan

Pengendalian yang dilakukan selama berlangsungnya kegiatan disebut juga dengan istilah pengendalian sibernatik (*steering control* atau *cybernetic control* atau *feedforward control*). Tujuan pengendalian ini adalah untuk mendeteksi adanya penyimpangan dari ketentuan yang telah ditetapkan, baik berupa standar ataupun tujuan, dan memperbaikinya sebelum suatu kegiatan berakhir.

### C. Pengendalian setelah berlangsungnya kegiatan

Pengendalian yang dilakukan setelah kegiatan berlangsung disebut juga dengan istilah pengendalian pascatindakan (*post-action control*). Pengendalian jenis ini merupakan kebalikan dari pengendalian pratindakan. Artinya, dalam pengendalian ini pembandingan dengan standar dilakukan *setelah* kegiatan tersebut berakhir. Contoh dari pengendalian pratindakan ini misalnya audit yang dilakukan oleh auditor independen atas kewajaran laporan keuangan, ataupun penelaahan yang dilakukan oleh auditor internal. Namun demikian, auditor internal dan eksternal banyak pula melakukan penelaahan selama berlangsungnya kegiatan perusahaan (*steering control*) dan dilanjutkan sampai pekerjaan tersebut selesai (pengendalian pascatindakan). Dengan demikian, tren yang terjadi dalam lingkungan audit adalah menggabungkan antara jenis pengendalian yang kedua (pengendalian sibernatik) dan jenis ketiga (pengendalian pascatindakan).

## 2.4.2 Pengendalian Menurut Sifatnya

Menurut sifatnya, pengendalian dapat diklasifikasikan menjadi dua kelompok, yaitu:

### A. Pengendalian akuntansi

Pengendalian akuntansi meliputi rencana organisasi serta prosedur dan pencatatan yang berkaitan dengan penjagaan aset serta keandalan catatan finansial. Tujuannya adalah untuk menjaga aset dan catatan perusahaan serta untuk memverifikasi ketepatan dan dapat diandalkannya data akuntansi.

### B. Pengendalian administratif

Pengendalian administratif antara lain mencakup rencana organisasi serta prosedur dan pencatatan yang berkaitan dengan proses pengambilan keputusan yang mengarah pada otorisasi transaksi dan merupakan titik awal dari pembuatan pengendalian akuntansi. Tujuan dari pengendalian administratif adalah:

- untuk menyediakan informasi, baik informasi strategis, informasi operasional ataupun informasi taktis, yang relevan bagi manajemen dalam melaksanakan fungsinya.
- untuk menumbuhkan atau mendorong efisiensi operasional serta efektivitas dan ekonomi sistem.
- untuk mendorong ditaatinya kebijakan, prosedur, dan standar yang telah ditetapkan.
- untuk menaati peraturan dan perundang-undangan yang berlaku.

## 2.4.3 Pengendalian Menurut Tujuannya

Menurut tujuan dari sistem tersebut, pengendalian biasanya diklasifikasikan menjadi tiga kelompok berikut yang sangat erat kaitannya dengan aplikasi dan pengembangan sistem komputer. Ketiga kelompok pengendalian menurut tujuannya tersebut adalah sebagai berikut.

### A. Pengendalian preventif

Pengendalian preventif bertujuan sama dengan pengendalian pratindakan, yaitu untuk mencegah terjadinya kerugian atau penyimpangan, selain untuk mengarahkan kegiatan agar sesuai dengan yang direncanakan. Pengendalian preventif sering dikatakan lebih baik dibandingkan dengan pengendalian detektif dan korektif karena alasan-alasan sebagai berikut.

- Pengendalian preventif sifatnya lebih mudah.
- Biaya pelaksanaannya lebih murah.
- Lebih baik mencegah sebelum suatu masalah timbul daripada mendeteksi atau memperbaiki persoalan setelah hal tersebut terjadi.

Pengendalian ini bersifat mencegah timbulnya kejadian yang tidak diharapkan. Di dalam lingkungan sistem informasi berbasis komputer, pengendalian preventif pada umumnya dicapai dengan cara mengimplementasikan prosedur otomatis untuk mencegah akses terhadap sistem oleh pihak yang tidak mempunyai otorisasi serta untuk mengharuskan dilaksanakannya tindakan yang tepat dan konsisten oleh *user.* Contoh pengendalian preventif adalah:

- Operator memerlukan *password* untuk dapat melakukan *entry* data.
- Melakukan verifikasi ulang untuk data masukan.
- Sistem tidak dapat menutup buku jika terdapat transaksi yang hilang.

Penerapan pengendalian ini dengan jaminan yang tinggi akan banyak membutuhkan biaya. Oleh karena itu, biasanya pengendalian ini dikombinasikan dengan pengendalian detektif untuk mengidentifikasi kesalahan yang tidak dapat dicegah terjadinya. Termasuk dalam pengendalian preventif misalnya formulir yang diberi nomor urut *(prenumbered),* penggunaan kata sandi *(password)* untuk dapat mengakses komputer, dan sebagainya.

## B. Pengendalian detektif

Pengendalian detektif dimaksudkan untuk menentukan dan mengidentifikasi adanya kesalahan yang terjadi dalam pelaksanaan kegiatan tertentu, selain untuk mengurangi frekuensi terjadinya kesalahan tersebut. Keterbatasan pengendalian ini adalah bahwa pengendalian detektif hanya dapat memberitahukan mengenai masalah yang timbul saja, tetapi tidak dapat mencegah terjadinya persoalan tersebut. Contoh pengendalian ini misalnya akses ke komputer yang terhenti apabila sampai tiga kali seseorang memasukkan kata sandi yang salah.

Pengendalian ini bersifat mengidentifikasi atau mendeteksi peristiwa yang tidak diharapkan setelah hal tersebut terjadi. Pengendalian detektif ini mempunyai dua komponen, yaitu:

- terdapat suatu sistem yang mengidentifikasi dan mencatat kegiatan *user*, transaksi kunci, dan menghasilkan laporan "pengecualian" (*exceptional report*).
- terdapat hierarki manajemen, yang mempunyai tugas mereviu laporan luar biasa tersebut dan melakukan tindakan perbaikan bila diperlukan.

Titik kunci dari pengendalian ini adalah adanya petugas khusus yang menerima laporan tersebut dengan tugas untuk melakukan tindak lanjut yang diperlukan.

Oleh karena pengendalian preventif dapat gagal atau terlewat oleh *user*, pengendalian detektif harus melakukan pengecekan kembali agar dapat mendeteksi kesalahan yang tidak dapat dicegah oleh pengendalian preventif. Pengecekan ulang ini dari segi biaya mungkin akan mahal, untuk itu penerapannya dapat secara selektif pada hal-hal yang penting atau secara *sampling*.

Pada setiap pengendalian detektif seharusnya terdapat satu atau lebih pengendalian korektif, tergantung pada jenis dan kompleksitas kesalahan. Apabila pengendalan detektif tersebut terdeteksi kesalahan atau hal yang tidak diharapkan, maka tindakan korektif dapat berupa:

- hanya melaporkan atau menginformasikan terjadinya kesalahan atau yang memenuhi kriteria tertentu.
- memisahkan dari pemrosesan dan dilaporkan untuk tindakan lebih lanjut.
- melakukan tindakan koreksi dan perbaikan lebih dahulu sebelum meneruskan pemrosesan.

**Contoh:**

Sistem aplikasi yang ada melakukan identifikasi atas transaksi yang melewati suatu jumlah yang telah ditetapkan oleh manajemen, memprosesnya dan menginformasikan kejadian tersebut dengan cara mencetaknya dalam format laporan yang ditetapkan.

Dalam pengendalian yang lain, sistem yang ada dapat dirancang untuk memisahkan suatu transaksi yang diidentifikasi mengandung kesalahan dan melaporkannya kepada petugas tertentu untuk dilakukan koreksi yang diperlukan.

**Contoh:**
Berhubungan dengan pendeteksian kesalahan dalam proses produksi. Bila ditemukan kesalahan atau cacat produksi, maka produk cacat tersebut harus diproses ulang atau diperbaiki dahulu sebelum dapat diteruskan ke proses produksi selanjutnya.

### C. Pengendalian korektif

Tujuan dari pengendalian korektif adalah untuk memberikan informasi yang diperlukan oleh personel yang terlibat dalam penyelidikan dan perbaikan kesalahan yang telah terdeteksi oleh pengendalian detektif. Meskipun namanya ''korektif' dan terdapat pula dalam lingkungan komputer, pengendalian ini hanya mengumpulkan bukti dan komputer tidak memperbaiki sendiri kesalahan yang terjadi. Perbaikan kesalahan tersebut tetap dilakukan oleh personel yang mempunyai otoritas untuk itu. Artinya, komputer hanya memberikan informasi yang dapat digunakan dalam menentukan mengapa persoalan terjadi dan bukan ia sendiri yang memperbaikinya. Contoh pengendalian korektif misalnya catatan aktivitas komputer (*transaction log*) yang dapat diprogramkan untuk menyebutkan mengenai siapa yang mengakses (dari identitas pengguna), kapan aktivitas tersebut dilakukan (dari waktu), dari terminal mana suatu aktivitas dilakukan (dari nomor terminal yang diakses), dan sebagainya. Dari data ini para personel yang terlibat dapat mengetahui di mana kesalahan terjadi atau siapa yang melakukannya.

## 2.4.4 Pengendalian Menurut Klasifikasi Lainnya

Ada tiga klasifikasi menurut Weber, Ikatan Akuntan Indonesia, serta menurut Vasarhelyi dan Lin. Mullen menyebutkan adanya beberapa acuan referensi mengenai klasifikasi pengendalian yang lainnya.

### A. Klasifikasi Menurut Ron Weber

Menurut Ron Weber, pengendalian dalam sistem komputer dapat diklasifikasikan secara berbeda-beda. Beberapa jenis pengendalian yang dikaitkan dengan upaya untuk mendorong keandalan pengolahan data menurutnya adalah sebagai berikut. (Weber, 1999).

1. ***Pengendalian terhadap keaslian (authenticity controls).*** Pengendalian ini dimaksudkan untuk memverifikasi identitas individual atau proses yang akan melakukan beberapa kegiatan di dalam sistem seperti kata sandi, *personal identification numbers* (PIN), tanda tangan digital, dan sebagainya.
2. ***Pengendalian terhadap akurasi (accuracy controls).*** Pengendalian ini dimaksudkan untuk memberi jaminan mengenai kebenaran data dan proses di dalam sistem seperti *program validation check* untuk mengetahui bahwa unsur data (*field*) numerik memang hanya berisi angka, *overflow check, hash total,* dan sebagainya.

3. ***Pengendalian atas kelengkapan (completeness controls).*** Pengendalian ini dimaksudkan untuk memberi jaminan bahwa tidak ada data yang hilang dan bahwa semua pemrosesan dilaksanakan melalui kesimpulan yang benar seperti *program validation check* untuk memverifikasi bahwa tidak ada unsur data yang kosong, dan sebagainya.
4. ***Pengendalian ulangan (redundancy controls).*** Pengendalian ini dimaksudkan untuk memberikan jaminan bahwa pos-pos data tertentu hanya diproses satu kali sehingga tidak ada data yang dobel atau terulang. Contoh dari jenis pengendalian ini misalnya tanda pembatalan dalam *batch,* nomor urut *record*, dan sebagainya.
5. ***Pengendalian atas privasi data (privacy controls).*** Pengendalian ini dimaksudkan untuk memberikan jaminan bahwa data diproteksi sedemikian rupa sehingga tidak dapat diakses oleh orang yang tidak berwenang. Contoh pengendalian ini misalnya penggunaan metode enkripsi data *(encryption),* kata sandi, dan sebagainya.
6. ***Pengendalian atas jejak audit (audit trail controls).*** Pengendalian ini dimaksudkan untuk memberikan jaminan adanya pencatatan yang kronologis tentang semua kegiatan yang terjadi di dalam sistem.
7. ***Pengendalian atas eksistensi (existence controls).*** Pengendalian ini dimaksudkan untuk memberikan jaminan mengenai ketersediaan semua sumber daya sistem yang dibutuhkan dalam pengolahan data, seperti misalnya duplikat perangkat keras, pemeliharaan preventif, pengendalian tentang memulai lagi komputer setelah terhenti *(restart controls),* dan sebagainya.
8. ***Pengendalian atas perlindungan aset (asset safeguarding controls).*** Pengendalian ini dimaksudkan untuk memberikan jaminan bahwa semua sumber daya yang terdapat di dalam sistem benar-benar dilindungi dari kerusakan atau kehilangan. Contoh pengendalian ini misalnya pemadam kebakaran, kata sandi, kepustakaan data, dan sebagainya.
9. ***Pengendalian atas efektivitas sistem (effectiveness controls).*** Pengendalian ini dimaksudkan untuk memberikan jaminan bahwa sistem yang digunakan mencapai sasaran yang ditetapkan. Contoh pengendalian ini misalnya pemantauan reguler terhadap kepuasan para pengguna sistem, pemantauan atas frekuensi penggunaan sumber daya komputer, analisis biaya-manfaat secara periodik, dan sebagainya.
10. ***Pengendalian atas efisiensi penggunaan sumber daya komputer (effectiveness controls).*** Pengendalian ini dimaksudkan untuk memberikan jaminan bahwa sistem komputer yang digunakan tersebut memerlukan sumber daya yang sedikit untuk mencapai sasaran yang ditetapkan oleh manajemen. Contoh pengendalian ini misalnya dengan melakukan wawancara dengan para pengguna sistem secara reguler, catatan atas penggunaan sumber daya komputer, dan sebagainya.

Meskipun pengendalian menurut Weber dapat diklasifikasikan berdasar pada sepuluh kategori di atas, Weber sendiri mengklasifikasikan pengendalian komputer dalam dua kategori atau subsistem, yaitu subsistem manajemen dan subsistem aplikasi

di mana pengendalian manajemen terletak mengelilingi pengendalian sistem aplikasi. Komponen pengendalian dalam subsistem manajemen adalah sebagai berikut.

1. ***Pengendalian manajemen tertinggi (top management controls).***
   Pengendalian ini bertujuan untuk memberikan jaminan bahwa instalasi pengolahan data dikelola dengan baik.
2. ***Pengendalian manajemen sistem informasi berbasis komputer.***
   Pengendalian ini bertujuan untuk memberikan jaminan bahwa tanggung jawab perencanaan dan pengendalian aplikasi komputer dari manajemen sistem informasi berbasis komputer kepada pimpinan tertinggi mengenai telah dilaksanakan secara memadai.
3. ***Pengendalian pengembangan sistem (system development controls).***
   Pengendalian ini bertujuan untuk memberikan jaminan bahwa manajemen pengembangan sistem telah melaksanakan fungsinya untuk merancang/ mendesain, menerapkan, dan memelihara setiap sistem aplikasi.
4. ***Pengendalian pengelolaan pemrograman.***
   Pengendalian ini bertujuan untuk memberikan jaminan bahwa manajemen pemrograman telah melaksanakan fungsinya untuk membuat sistem yang baru, memelihara sistem yang lama dan menyediakan bantuan bagi terlaksananya sistem yang baru tersebut.
5. ***Pengendalian administrasi data.***
   Pengendalian ini bertujuan untuk memberikan jaminan bahwa administrator data telah melaksanakan fungsinya untuk mengendalikan penggunaan data organisasi yang bersangkutan termasuk *database* dan kepustakaan (*library*) sistem aplikasi.
6. ***Pengendalian administrasi keamanan.***
   Pengendalian ini bertujuan untuk memberikan jaminan bahwa manajemen administrasi keamanan telah melaksanakan fungsinya untuk mengamankan sumber daya fisik instalasi pengolahan data perusahaan yang bersangkutan.
7. ***Pengendalian operasi.***
   Pengendalian ini bertujuan untuk memberikan jaminan bahwa manajemen operasi telah melaksanakan fungsinya mengendalikan aktivitas pengolahan data sehari-hari seperti mengendalikan penyiapan data, arus transaksi data melalui sistem aplikasi, pemeliharaan perangkat keras dan sebagainya.

Komponen pengendalian dalam subsistem aplikasi komputer adalah sebagai berikut.

1. ***Pengendalian pembatasan akses (boundary controls).***
   Pengendalian ini dimaksudkan untuk menetapkan identitas dan keabsahan dari pengguna yang akan menggunakan sumber daya komputer sebelum mengakses sistem komputer tersebut. Artinya, begitu pengendalian ini selesai melaksanakan fungsinya, maka para pengguna baru dapat melanjutkan pengaksesan pada sistem komputer. Beberapa jenis pengendalian ini misalnya

kata sandi, kriptografi (*cryptographic controls*), *personal identification numbers* (PIN), tanda tangan digital, jejak audit, kartu plastik, dan sebagainya.

2. ***Pengendalian masukan.***
   Pengendalian ini dimaksudkan untuk memberikan jaminan bahwa proses penyiapan dan pemasukan transaksi data ke dalam sistem sesuai dengan tujuan yang telah ditetapkan oleh organisasi atau perusahaan yang bersangkutan. Pengendalian ini dibagi dalam dua kategori, yaitu (1) data dan masukan instruksi seperti dalam bentuk *check digits*, *batch controls*, jejak audit dan sebagainya, dan (2) pengendalian validasi dan kesalahan yang antara lain dapat pula menggunakan jejak audit.
3. ***Pengendalian komunikasi.***
   Pengendalian ini dimaksudkan untuk memberikan jaminan bahwa transmisi data yang dilakukan oleh satuan usaha terbebas dari kegagalan komponen transmisi data seperti saluran komunikasi serta perangkat keras (*modem, multiplexor,* dan sebagainya) dan perangkat lunak komunikasi antarkomputer, selain untuk mencegah masuknya pihak ketiga (*intruder*) yang secara sengaja masuk ke jaringan komunikasi satuan usaha untuk tujuan yang tidak baik.
4. ***Pengendalian pengolahan.***
   Pengendalian ini dimaksudkan untuk memberikan jaminan bahwa komponen yang melakukan proses perhitungan, pengklasifikasian, pengurutan, dan pengikhtisaran data di dalam sistem telah berfungsi sebagaimana mestinya. Misalnya, pengendalian integritas operasi sistem, pengendalian *software* aplikasi, dan sebagainya.
5. ***Pengendalian database.***
   Pengendalian ini dimaksudkan untuk memberikan jaminan bahwa pendefinisian, penambahan, pengaksesan, perubahan dan penghapusan data di dalam sistem sudah sesuai dengan tujuan satuan usaha yang bersangkutan. Misalnya, pengendalian akses, pengendalian penanganan data, pengacakan data, jejak audit, dan sebagainya.
6. ***Pengendalian keluaran.***
   Pengendalian ini dimaksudkan untuk memberikan jaminan bahwa pemanggilan atau pengaksesan dan penyajian data kepada para penggunanya adalah memang hanya terbatas pada mereka yang berwenang untuk melakukannya. Misalnya, pengendalian penyajian data seperti siapa yang dapat membaca data apa dan siapa yang tidak boleh melakukannya, siapa yang dapat mengubah data apa dan siapa yang tidak dapat melakukannya, dan sebagainya.

## B. Klasifikasi Menurut IAI

Sementara itu Ikatan Akuntan Indonesia mengklasifikasikan pengendalian dalam lingkungan komputer menjadi dua kelompok, yaitu pengendalian umum dan pengendalian aplikasi. Secara ringkas IAI mendefinisikan kedua jenis pengendalian tersebut sebagai berikut.

- Pengendalian internal atas pengolahan komputer, yang dapat membantu pencapaian tujuan pengendalian internal secara keseluruhan, mencakup prosedur manual dan prosedur yang didesain dalam program komputer. Prosedur pengendalian manual dan komputer terdiri atas pengendalian menyeluruh yang berdampak terhadap lingkungan SIK [Sistem Informasi Komputer] (pengendalian umum SIK) dan pengendalian khusus atas aplikasi akuntansi (pengendalian aplikasi SIK).
- Pengendalian umum (*general controls*) meliputi kebijakan dan prosedur mengenai semua aktivitas PDE yang bertujuan untuk membuat kerangka pengendalian yang menyeluruh mengenai aktivitas PDE serta untuk memberikan tingkat keyakinan yang memadai bahwa seluruh tujuan pengendalian internal dapat tercapai. Pengendalian umum ini mencakup hal-hal sebagai berikut.
  1. Rencana organisasi dan pelaksanaan aktivitas PDE. Pengendalian umum yang berkaitan dengan hal ini disebut dengan istilah pengendalian organisasi dan manajemen.
  2. Prosedur untuk mendokumentasikan, menelaah, menguji, dan menyetujui sistem atau program serta perubahan terhadap sistem atau program tersebut.
  3. Pengendalian yang dibentuk dalam komputer oleh pabrik komputer atau pembuat perangkat lunak (*software house*), atau yang biasa disebut dengan istilah pengendalian perangkat keras (*hardware controls*) dan pengendalian perangkat lunak (*software controls*).
  4. Pengendalian terhadap akses penggunaan peralatan, program, dan data.
  5. Pengendalian data dan prosedural lainnya yang memengaruhi seluruh aktivitas PDE.

Pengendalian aplikasi (*application controls*) dimaksudkan untuk memberikan kepastian bahwa pencatatan, pengklasifikasian, dan pengikhtisaran transaksi yang sah serta pemutakhiran *file* induk akan menghasilkan informasi yang akurat, lengkap, dan tepat waktu.

Pengendalian aplikasi ini bertujuan untuk "menetapkan prosedur pengendalian khusus atas aplikasi akuntansi untuk memberikan keyakinan memadai bahwa semua transaksi telah diotorisasi dan dicatat, serta diolah seluruhnya, dengan cermat dan tepat waktu." (SA Seksi 314 [PSA No. 60] Paragraf 08). Pengendalian aplikasi ini dikelompokkan menjadi tiga kategori pengendalian, yaitu pengendalian atas masukan, pengendalian atas pengolahan dan *file* data komputer, serta pengendalian atas keluaran.

Perbedaan utama antara pengendalian umum dan pengendalian aplikasi adalah bahwa sifat pengendalian umum adalah prosedural, sedangkan pengendalian aplikasi bersifat lebih berorientasi pada data. Oleh sebab itu, bagi auditor mungkin menilai pengendalian umum secara terpisah dari penilaian terhadap pengendalian aplikasi.

## C. Klasifikasi Menurut Vasarhelyi dan Lin

Pengolahan data dapat dilakukan secara kelompok (*batch processing*) ataupun secara seketika (*online*, *realtime*, atau OLRT). Dalam kaitannya dengan metode

pengolahan data tersebut, Vasarhelyi dan Lin membuat klasifikasi pengendalian aplikasi, yaitu ditandai dengan kategori pengendalian atas masukan, pengolahan dan keluaran. Pengendalian yang tercakup meliputi kedua jenis pengendalian PDE, yaitu pengendalian umum dan pengendalian aplikasi. Selain itu, Vasarhelyi dan Lin juga mengklasifikasikan pengendalian internal menjadi 89 jenis serta kesalahan yang mungkin timbul dalam 48 jenis.

#### D. Pengendalian dalam Lingkungan Komputer Mikro

Sebagaimana dikemukakan pula dalam SA Seksi 343, pengendalian dalam lingkungan komputer mikro kadang-kadang sulit untuk dipisahkan antara pengendalian umum dan pengendalian khusus, (SA Seksi 343 [PSA No. 63] Paragraf 13–15) sehingga pembahasan Bab 2 mungkin akan menjadi rumit atau bahkan sulit untuk diaplikasikan dalam lingkungan komputer personal. Oleh karena itu, IAI memerinci lebih lanjut dalam SA Seksi 343 pengendalian yang lebih spesifik untuk lingkungan komputer personal. Selain itu, Canadian Institute of Chartered Accountants (CICA) juga memberikan pedoman mengenai pengendalian dalam lingkungan komputer personal ini. Secara umum pengendalian dalam lingkungan komputer personal mencakup lima permasalahan sebagai berikut.

1. Penentuan penggunaan komputer.
2. Pengamanan fisik.
3. Penyimpanan data dan program serta metode pencadangan.
4. Perjanjian lisensi.
5. Perlindungan terhadap virus.

### 2.4.5 Pengendalian Menurut Waktu Penerapan Model PDC

Pengendalian internal menurut waktu penerapan terdiri tiga tingkat penerapan, yaitu (1) pengendalian preventif, (2) pengendalian detektif, dan (3) pengendalian korektif. Pendekatan ini disebut juga pendekatan model PDC.

#### A. Pengendalian Preventif

Pengendalian yang bersifat mencegah atau sebelum risiko terjadi, pencegahan ini merupakan lini depan dari pertahanan dalam struktur pengendalian. Pengendalian ini adalah teknik pasif yang didesain untuk mengurangi frekuensi terjadinya risiko. Pengendalian preventif menegakkan kepatuhan melalui tindakan yang seharusnya atau yang diinginkan sehingga mencegah tindakan yang menyimpang.

Pencegahan kesalahan dan penipuan jauh lebih efektif dari segi biaya daripada mendeteksi dan memperbaiki masalah setelah masalah tersebut terjadi. Misalnya, suatu layar entri data yang dilengkapi dengan fasilitas validasi data untuk mencegah data tidak valid di-*input* dan diproses lebih lanjut. Pengendalian akses yang efektif terhadap akses jarak jauh untuk mencegah akses tidak sah ke sistem perusahaan, yaitu dengan memasang *password* atau dengan teknologi biometrik.

## B. Pengendalian Detektif

Pengedalian lini kedua dari pertahanan adalah pengendalian detektif, yaitu teknik untuk mengidentifikasi dan mengekspos risiko yang lolos dari pengendalian preventif. Pengendalian detektif mengungkapkan berbagai jenis kesalahan tertentu dengan cara membandingkan kejadian sesungguhnya dengan standar yang telah ditetapkan. Ketika pengendalian detektif mengidentifikasi adanya penyimpangan dari standar, maka sistem akan merespons atau memberi peringatan terkait dengan permasalahan tersebut. Misalnya, adanya kesalahan entri data, *record* pesanan penjualan seorang pelanggan berisi data seperti berikut ini.

| Kuantitas | Harga Per Unit | Total |
|---|---|---|
| 10 unit | Rp100.000 | Rp10.000.000 |

Sebelum memproses transaksi ini dan memasukkannya ke dalam akun, pengendalian detektif seharusnya menghitung kembali nilai total dengan menggunakan harga dan kuantitas.

## C. Pengendalian Korektif

Tindakan perbaikan yang harus dilakukan untuk membalikkan pengaruh negatif dari kesalahan yang telah dideteksi. Pengendalian korektif sebenarnya hanya memperbaiki masalahnya, untuk setiap masalah yang terdeteksi, maka akan ada lebih dari satu tindakan perbaikan yang mungkin dapat dilakukan. Misalnya, dengan melihat kesalahan sebelumnya, kecenderungan pertama Anda mungkin adalah mengubah total dari Rp10.000.000 menjadi Rp1.000.000 untuk memperbaiki kesalahan tersebut. Tindakan ini menganggap bahwa nilai kuantitas dan harga dalam *record* terkait sudah benar; padahal mungkin saja tidak demikian. Pada kasus ini, Anda tidak dapat menentukan penyebab sebenarnya dari masalah tersebut; kita hanya tahu bahwa ada masalah.

Menghubungkan tindakan korektif dengan kesalahan yang terdeteksi, sebagai respons otomatis, dapat menghasilkan tindakan yang salah dan memperburuk masalah daripada kesalahan aslinya itu sendiri. Oleh karena itu, perbaikan kesalahan harus dipandang sebagai tahap pengendalian yang terpisah dan yang harus dilakukan dengan hati-hati.

## D. Pengendalian Prediktif

Kini dengan kemajuan dan perkembangan teknologi bagi auditor benar-benar dapat memprediksi kejadian menyimpang. Sebagai contoh adalah melalui jaringan syaraf buatan (*artificial neural network*—ANN) dan Internet Storm Center (*http://isc.incidents.org*). ANN memiliki kemampuan belajar atau mengenali pola dalam berbagai transaksi yang berisi kesalahan atau penyimpangan dengan mengekspos sistem tersebut ke berbagai kejadian sesungguhnya di masa lalu. ANN menggunakan modul audit melekat (*embeded audit module*—EAM), sistem tersebut dapat menyaring

berbagai transaksi untuk mencari transaksi yang mencurigakan, dan memberikan respons atau peringatan kepada pihak yang terkait segera setelah transaksi tersebut dimasukkan. Contoh yang kedua berkaitan dengan keamanan Internet. Internet Storm Center (ISC) menggabungkan daftar dari berbagai *host* Internet untuk menelusuri aktivitas pada *port* Internet tertentu. Kemudian, dengan menelusuri tingkat aktivitasnya, ISC memiliki kemampuan untuk melihat "anomali" atau aktivitas yang tidak biasa. Sistem ini merupakan sistem peringatan dini yang memperingatkan pihak-pihak yang terkait mengenai virus, *worm*, serangan penolakan layanan, dan aktivitas perusakan lainnya.

### 2.4.6 Pengendalian dalam Sistem Online

SA Seksi 344, Lingkungan Sistem Informasi Komputer—*Online Computer System,* selain menjelaskan mengenai metode pengolahan data secara elektronik juga mengemukakan mengenai pengendalian umum dan pengendalian aplikasi dalam sistem *online,* di samping menjelaskan pula mengenai dampak metode pengolahan data tersebut terhadap sistem akuntansi dan pengendalian internal serta terhadap prosedur audit. Pengendalian yang diuraikan dalam SA Seksi 344 mencakup pengendalian umum dan pengendalian aplikasi. Jenis-jenis pengendalian yang tercakup dalam pengendalian umum dalam pengolahan *online* adalah sebagai berikut.

1. Pengendalian akses, dengan tujuan untuk membatasi akses ke dalam program dan data sehingga dapat dicegah atau dideteksi hal-hal sebagai berikut.
   - Akses yang tidak semestinya.
   - Pemasukan transaksi dan perubahan *file* tanpa otorisasi.
   - Penggunaan program oleh personel yang tidak berwenang.
   - Penggunaan program yang belum memperoleh otorisasi.
2. Pengendalian terhadap kata sandi, dengan tujuan untuk membatasi akses hanya untuk petugas yang berwenang.
3. Pengendalian atas pengembangan dan pemeliharaan sistem, dengan tujuan untuk menjamin bahwa pengendalian yang dibutuhkan benar-benar dimasukkan ke dalam sistem selama pengembangan dan pemeliharaan sistem.
4. Pengendalian pemrograman, dengan tujuan untuk mencegah atau mendeteksi perubahan terhadap program secara tidak sebagaimana mestinya.
5. Pencatatan atas aktivitas yang dilakukan (*transaction log*).

Sementara itu jenis-jenis pengendalian yang tercakup dalam pengendalian aplikasi dalam pengolahan *online* adalah sebagai berikut.

1. Adanya otorisasi sebelum pengolahan.
2. Adanya pengeditan, pengujian kelayakan, dan validasi lainnya.
3. Adanya prosedur pisah batas agar transaksi diolah sesuai dengan periode akuntansinya.
4. Adanya pengendalian terhadap *file* dan *file* induk.

Selain dari pengendalian umum dan pengendalian aplikasi sebagaimana dinyatakan oleh IAI dalam SA Seksi 344 di atas, sebagaimana dinyatakan dalam Bab 2, perkembangan lebih lanjut dari jaringan antarkomputer antara lain adalah dalam bentuk Electronic Data Interchange (EDI). Sebagaimana dikemukakan oleh Chan dkk., EDI selain memberikan banyak manfaat bagi organisasi penggunanya juga menimbulkan risiko baru terhadap auditabilitas, risiko menjadi bergantung pada pihak ketiga, risiko apabila program aplikasinya tidak berfungsi, serta munculnya risiko dalam bentuk *domino effect* apabila terjadi kesalahan dari salah satu pihak pengguna EDI. Oleh karena, itu mereka menyarankan berbagai jenis pengendalian yang dimaksudkan untuk mengurangi berbagai risiko tersebut yang dikategorikan sebagai *application and environmental controls.*

## 2.5 PERNYATAAN STANDAR AUDIT NO. 78

Statement On Auditing Standard (SAS) No. 78 sesuai dengan rekomendasi komputer organisasi pendukung dari Komisi *Treadway* (*Committee of Sponsoring Organizations of the Treadway Commission*—COSO), menyatakan pengendalian internal terdiri dari lima komponen, yaitu lingkungan pengendalian, penaksiran risiko, informasi dan komunikasi, pengawasan, dan aktivitas pengendalian.

### 2.5.1 Lingkungan Pengendalian

Lingkungan pengendalian (*control environment*) adalah dasar untuk keempat komponen pengendalian lainnya. Lingkungan pengendalian menetapkan arah perusahaan dan pengaruh kesadaran pihak manajemen dan para karyawannya akan pengendalian. Lingkungan pengendalian memiliki beberapa elemen penting:

- Nilai integritas dan etika pihak manajemen.
- Struktur perusahaan.
- Keterlibatan dewan komisaris dan komite audit perusahaan, jika ada.
- Filosofi pihak manajemen dan gaya operasi.
- Prosedur untuk mendelegasikan tanggung jawab dan wewenang.
- Metode pihak manajemen untuk menilai kinerja.
- Pengaruh eksternal, seperti pemeriksaan oleh lembaga yang berwenang.
- Kebijakan dan praktik perusahaan untuk mengelola sumber daya manusia.

SAS 78 mengharuskan auditor memiliki pengetahuan yang memadai untuk menilai sikap dan kesadaran pihak manajemen perusahaan, dewan komisaris, dan para pemilik atas pengendalian internal. Berikut contoh berbagai teknik yang dapat digunakan untuk mendapatkan pemahaman mengenai lingkungan pengendalian.

1. Perusahaan harus menilai integritas pihak manajemen perusahaan dan dapat menggunakan lembaga penyelidik untuk memberikan laporan mengenai latar belakang para manajer pentingnya.

2. Auditor harus memerhatikan berbagai kondisi yang akan memungkinkan pihak manajemen suatu perusahaan melakukan penipuan.
3. Auditor harus memahami bisnis dan industri kliennya serta harus mengetahui berbagai kondisi luar biasa dalam industri tersebut yang mungkin dapat memengaruhi audit terkait.
4. Auditor harus menentukan apakah dewan komisaris perusahaan secara aktif dilibatkan dalam pembentukan kebijakan bisnis dan apakah dewan tersebut memonitor pihak manajemen serta operasi perusahaan.
5. Dari struktur organisasi dan deskripsi pekerjaan, auditor dapat menilai apakah pemisahan antarfungsi dalam perusahaan telah memadai.

### 2.5.2 Penaksiran Risiko

Perusahaan harus melakukan penaksiran risiko (*risk assessment*) untuk mengidentifikasi, menganalisis, dan mengelola risiko yang berkaitan dengan pelaporan keuangan. Berbagai risiko dapat timbul dari berbagai perubahan lingkungan, seperti berikut ini.

- Perubahan dalam lingkungan operasional yang membebankan berbagai tekanan persaingan baru atas perusahaan.
- Personel baru yang memiliki pemahaman berbeda atau tidak memadai atas pengendalian internal.
- Sistem informasi baru atau yang direkayasa ulang sehingga memengaruhi pemrosesan transaksi.
- Pertumbuhan yang signifikan dan cepat hingga mengalahkan pengendalian internal yang ada.
- Implementasi teknologi baru ke dalam proses produksi atau sistem informasi yang berdampak pada pemrosesan transaksi.
- Pengenalan lini baru produk atau aktivitas di mana perusahaan hanya memiliki pengalaman sedikit mengenai produk tersebut.
- Restrukturisasi organisasi yang mengakibatkan pengurangan dan/atau relokasi personel hingga operasi bisnis serta pemrosesan transaksi yang terpengaruh.
- Masuk ke pasar asing yang dapat berdampak pada operasi (contohnya risiko yang berkaitan dengan transaksi mata uang asing).
- Adopsi prinsip akuntansi baru yang berdampak pada pembuatan laporan keuangan.

SAS 78 mengharuskan para auditor mendapatkan pengetahuan yang cukup atas prosedur penaksiran risiko perusahaan untuk memahami bagaimana cara pihak manajemen mengidentifikasi, membuat prioritas, serta mengelola berbagai risiko yang berkaitan dengan pelaporan keuangan.

### 2.5.3 Informasi dan Komunikasi

Sistem informasi akuntansi (SIA) terdiri dari atas *record* dan metode yang digunakan untuk memulai, mengidentifikasi, menganalisis, mengklasifikasi, serta mencatat

berbagai transaksi perusahaan dan untuk menghitung aset serta kewajiban yang terkait. Kualitas informasi yang dihasilkan oleh SIA berdampak pada kemampuan pihak manajemen untuk melakukan tindakan dan mengambil keputusan sehubungan dengan operasi perusahaan serta untuk membuat laporan keuangan yang andal. SIA yang efektif akan dapat melakukan berbagai hal-hal sebagai berikut.

- Mengidentifikasi dan mencatat semua transaksi keuangan yang valid.
- Menyediakan informasi secara tepat waktu mengenai berbagai transaksi mendetail yang memadai untuk memungkinkan klasifikasi dan pelaporan keuangan yang benar.
- Secara akurat mengukur nilai keuangan berbagai transaksi agar pengaruhnya dapat dicatat ke dalam laporan keuangan.
- Secara akurat mencatat berbagai transaksi dalam periode waktu terjadinya.

SAS 78 mengharuskan para auditor mendapatkan pengetahuan yang cukup mengenai sistem informasi perusahaan.

### 2.5.4 Pengawasan

Pengawasan (*monitoring*) adalah proses di mana kualitas dari desain dan operasi pengendalian internal dapat dinilai. Penilaian ini dapat dicapai dengan prosedur yang terpisah atau melalui aktivitas yang berjalan.

Auditor internal perusahaan dapat memonitor aktivitas entitas terkait dalam berbagai prosedur yang terpisah. Mereka dapat mengumpulkan bukti kecukupan pengendalian dengan menguji pengendalian, kemudian mengomunikasikan kekuatan serta kelemahan pengendalian ke pihak manajemen. Selain itu, auditor internal membuat rekomendasi khusus untuk perbaikan pengendalian.

*Monitoring* yang berjalan dapat dicapai dengan mengintegrasikan berbagai modul komputer khusus ke dalam sistem informasi yang menangkap data penting dan/atau memungkinkan uji pengendalian dilakukan sebagai bagian dari operasi rutin. Modul audit yang melekat (*embedded audit modules*—EAM) tersebut memungkinkan pihak manajemen dan auditor mempertahankan keberlanjutan atas berfungsinya pengendalian internal serta integritas data transaksi.

Teknik lain pengawasan berjalan adalah penggunaan secara hati-hati laporan manajemen. Laporan yang tepat waktu memungkinkan para manajer pada area fungsional seperti penjualan, pembelian, produksi, dan pengeluaran kas untuk mengawasi serta mengendalikan operasi mereka. Dengan merangkum berbagai aktivitas, memperlihatkan berbagai tren, serta mengidentifikasi berbagai perkecualian dari kinerja normal, laporan manajemen yang didesain dengan baik dapat memberikan bukti atas berfungsinya atau gagalnya pengendalian.

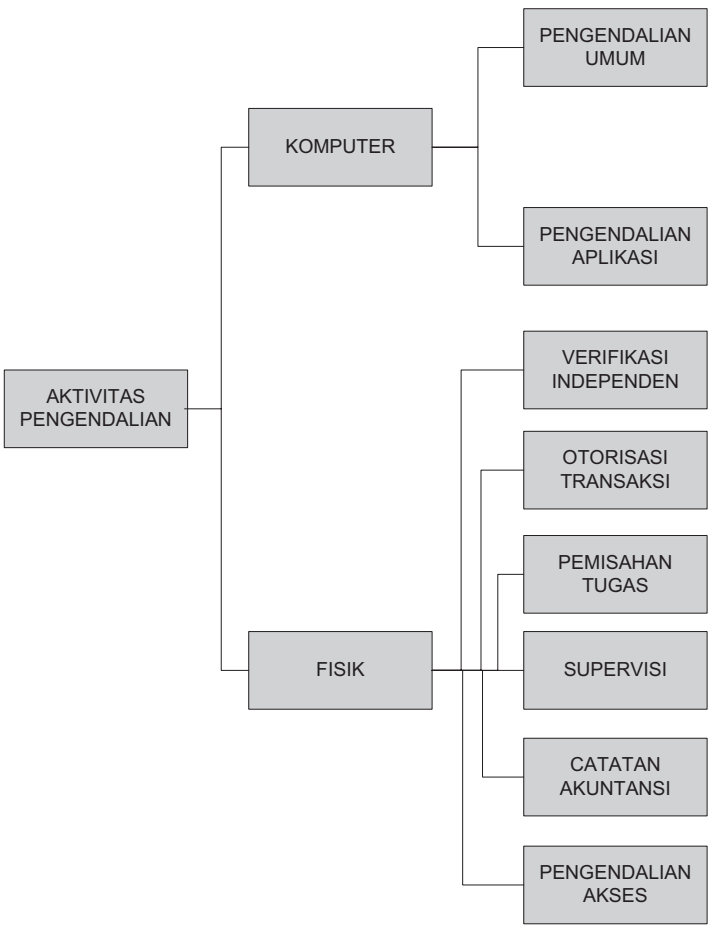
## 2.6 AKTIVITAS PENGENDALIAN

Aktivitas pengendalian (*control activities*) adalah berbagai kebijakan dan prosedur yang digunakan untuk memastikan bahwa tindakannya yang tepat telah dilakukan

untuk menangani berbagai risiko yang telah diidentifikasi perusahaan. Aktivitas pengendalian dapat dikelompokan ke dalam dua kategori, yaitu *pengendalian komputer* dan *pengendalian fisik*.

Pengendalian komputer digolongkan dalam dua kelompok umum, yaitu pengendalian umum (*general control*) dan pengendalian aplikasi (*aplication control*). Pengendalian umum berkaitan dengan perhatian pada seluruh tingkatan perusahaan, seperti pengendalian pada pusat data, *database* perusahaan, akses sistem, pengembangan sistem, dan pemeliharaan sistem. Pengendalian aplikasi memastikan integritas sistem tertentu seperti pemrosesan order penjualan, utang usaha, dan aplikasi penggajian.

Pengendalian fisik berkaitan dengan sistem akuntansi tradisional yang menggunakan prosedur manual. Namun, pemahaman atas konsep pengendalian ini juga memberikan padangan atas berbagai risiko dan kekhawatiran dalam pengendalian yang berkaitan dengan lingkungan TI.




Gambar 2.2
Kategori Aktivitas Pengendalian

## A. Otorisasi Transaksi

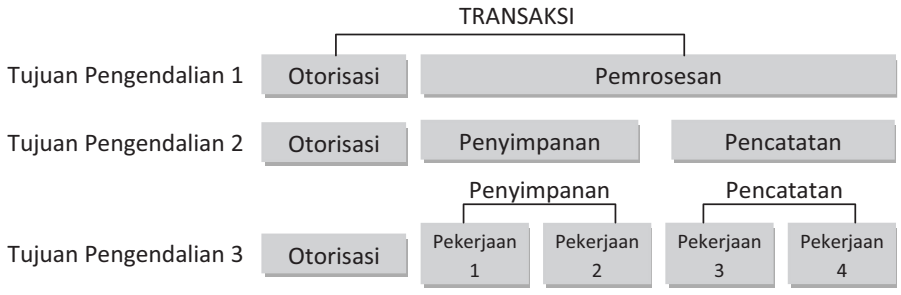
Tujuan dari otorisasi transaksi adalah untuk memastikan bahwa semua transaksi material yang diproses oleh sistem informasi valid dan sesuai dengan tujuan pihak manajemen. Otorisasi dapat bersifat umum atau khusus. Otorisasi umum diberikan pada personel operasional untuk melakukan operasi rutin. Sebagai contoh prosedur untuk mengotorisasi pembelian persediaan dari pemasok yang ditunjuk hanya ketika tingkat persediaan berada pada titik pemesanan kembali (*reorder point*) yang telah ditetapkan. Otorisasi khusus berkaitan dengan keputusan kasus per kasus yang berhubungan dengan transaksi nonrutin. Sebagai contoh keputusan untuk meningkatkan batas kredit pelanggan tertentu di atas jumlah normal. Otorisasi khusus ini biasanya merupakan tanggung jawab pihak manajemen.

Di dalam lingkungan TI, otorisasi transaksi dapat terdiri atas aturan berkode yang melekat pada program komputer. Contohnya, modul program pada sistem pembelian yang secara otomatis akan menentukan kapan, seberapa banyak, dan dari mana pemasok untuk persediaan yang akan dipesan. Tanggung jawab untuk tujuan pengendalian otorisasi transaksi terletak langsung pada akurasi dan konsistensi (integritas) program komputer yang melakukan berbagai pekerjaan ini. Berikut ini adalah ilustrasi mengenai tujuan dari pemisahan tugas.

## B. Pemisahan Tugas

Salah satu aktivitas pengendalian yang penting adalah pemisahan tugas karyawan untuk meminimalkan fungsi-fungsi yang tidak sesuai. Tiga tujuan pemisahan tugas adalah sebagai berikut.

1. Pemisahan tugas seharusnya sedemikian rupa sehingga otorisasi untuk suatu transaksi terpisah dari pemrosesan transaksi tersebut. Sebagai contoh pembelian tidak boleh dilakukan oleh bagian pembelian sampai diotorisasi oleh bagian pengendalian persediaan.
2. Tanggung jawab untuk penyimpanan aset seharusnya terpisah dari tanggung jawab pencatatan. Sebagai contoh bagian yang memiliki unsur penyimpanan fisik persediaan barang jadi (gudang) tidak boleh membuat catatan persediaan resmi. Akuntansi untuk persediaan barang jadi dilakukan oleh bagian pengendalian persediaan, yang merupakan fungsi akuntansi.
3. Perusahaan seharusnya distrukturisasi agar jika ada penipuan maka penipuan hanya dapat dilakukan lewat kolusi antara dua atau lebih individu dengan pekerjaan yang tidak saling berkesesuaian.




Gambar 2.3
Tujuan Pemisahan Tugas

Dalam lingkungan TI, program komputer biasanya melakukan berbagai pekerjaan yang dianggap tidak kompatibel dalam sistem manual. Program komputer dapat saja menjadi penanggung jawab satu-satunya atas otorisasi pembelian, pemrosesan order pembelian, dan pencatatan utang usaha. Ketika faktur dari pemasok sampai, program komputer akan menentukan waktu serta jumlah pembayaran yang harus dilakukan.

Pemisahan tugas masih memiliki peran penting dalam lingkungan TI, tetapi perhatian auditor TI harus diarahkan pada berbagai aktivitas yang mengancam integritas aplikasi. Misalkan ketika program telah berfungsi baik dalam implementasi sistem, integritasnya harus dipertahankan sepanjang siklus hidup program tersebut. Aktivitas pengembangan program, operasi program, dan pemeliharaan program adalah fungsi-fungsi TI yang sangat penting dan harus cukup terpisah.

## C. Supervisi

Di dalam perusahaan kecil atau dalam area fungsional yang kekurangan personel, pihak manajemen harus menyeimbangkan ketidakberadaan pengendalian pemisahan tugas dengan supervisi. Supervisi sering kali disebut pengendalian penyeimbang yang dapat berupa supervisi fisik, laporan, atau cara lainnya.

Dalam lingkungan TI, pengendalian supervisi harus lebih luas daripada sistem manual untuk tiga alasan. Alasan pertama berhubungan dengan masalah menarik karyawan yang kompeten. Teknologi pemrosesan data membuat lingkungan yang sangat rumit dan membutuhkan jenis karyawan tertentu. Mereka yang merancang, memprogram, memelihara, dan mengoperasikan sistem komputer perusahaan harus memiliki keahlian yang sangat khusus. Orang-orang ini bekerja dalam lingkungan dinamis yang dicirikan dengan perputaran karyawan yang tinggi. Pekerjaan untuk mengatur kembali staf akan semakin sulit dengan adanya perubahan teknologi yang cepat yang cenderung membuat mempersulit kemampuan pihak manajemen untuk menilai kompetensi calon karyawan.

Alasan kedua mencerminkan kekhawatiran pihak manajemen atas tingkat kepercayaan personel pemrosesan dalam area yang berisiko tinggi. Beberapa praktisi sistem bekerja dalam posisi yang memiliki wewenang tinggi hingga memungkinkan

akses langsung dan tidak terbatas ke program dan *database* perusahaan. Gabungan dari keahlian teknis dan peluang dalam genggaman seseorang yang mungkin melakukan penyimpangan atau perusakan, merupakan risiko besar bagi perusahaan.

Alasan ketiga ketidakmampuan pihak manajemen untuk secara memadai mengamati karyawannya dalam lingkungan TI. Aktivitas para karyawan yang terlibat dalam pemrosesan data sering kali tersembunyi dari observasi pihak manajemen. Sebagai contoh adalah para personel pemrosesan data yang tersebar di seluruh area dan melakukan fungsinya secara jarak jauh melalui jaringan. Pengendalian supervisi harus dirancang ke dalam sistem komputer untuk mengimbangi kurangya supervisi langsung.

## D. Catatan Akuntansi

Catatan akuntansi (*accounting record*) tradisional suatu perusahaan terdiri atas dokumen sumber, jurnal, dan buku besar. Catatan ini merekam aspek ekonomi transaksi dan menyediakan jejak audit peristiwa ekonomi. Jejak audit tersebut memungkinkan auditor menelusuri setiap transaksi melalui semua tahapan pemrosesannya dari awal peristiwa hingga laporan keuangan. Perusahaan harus mempertahankan jejak audit dengan dua alasan, yaitu (1) informasi yang dibutuhkan untuk melakukan operasi harian, (2) jejak audit memainkan peran penting dalam audit keuangan perusahaan.

Mempertahankan jejak audit pada lingkungan TI, catatan akuntansi otomatis dan jejak auditnya sangat berbeda dengan yang ada pada lingkungan manual. Beberapa sistem komputer tidak menyimpan dokumen sumber fisik. Jurnal dan buku besar sering kali tidak ada dalam antrian tradisional. Sebagai gantinya, berbagai *record* transaksi dan peristiwa ekonomi lainnya terfragmentasi melintasi beberapa tabel *database* yang dinormalisasi. Jejak audit dapat berbentuk *pointer*, teknik *hashing*, indeks, atau kunci melekat yang menghubungkan fragmen *record* antara dan antartabel *database*.

## E. Pengendalian Akses

Tujuan dari pengendalian ini adalah memastikan bahwa hanya personel yang telah diotorisasi yang dapat mengakses ke aset perusahaan. Akses yang tidak sah mengekspos aset ke penyalahgunaan, perusakan, dan pencurian. Akses ke aset dapat bersifat langsung maupun tidak langsung. Peralatan keamanan fisik seperti kunci, lemari besi, pagar besi, dan sistem alarm serta infra merah, dapat mengendalikan akses langsung ke aset. Akses tidak langsung dapat dicapai dengan mendapatkan akses ke berbagai catatan dan dokumen yang mengendalikan penggunaan, kepemilikan, dan disposisi aset. Dalam lingkungan manual, catatan akuntansi bersifat fisik dan cenderung akan disebarkan ke beberapa lokasi. Pengendalian akses tidak langsung dapat diwujudkan dengan mengendalikan penggunaan dokumen dan catatan serta dengan memisahkan tugas mereka yang harus mengakses dan memproses berbagai catatan ini.

Dalam lingkungan TI, catatan akuntansi sering kali terkonsentrasi dalam pusat pemrosesan data pada perangkat penyimpanan data yang besar. Dua bentuk ancaman dalam masalah ini adalah (1) penipuan dengan komputer, dan (2) kerugian akibat bencana.

Masalah lain yang hanya ada dalam lingkungan TI adalah mengendalikan akses ke program komputer. Selama tahap pengembangan, aplikasi komputer berada di bawah pengamatan dan pengujian yang dimaksudkan untuk mengekspos adanya kesalahan logika di dalamnya. Kekhawatiran atas integritas aplikasi tetap ada ketika setelah implementasi dalam periode operasional siklus hidup sistem (*tahap pemeliharaan*). Selama periode ini, aplikasi biasa dapat dimodifikasi beberapa kali. Modifikasi ini menimbulkan peluang adanya kesalahan yang masuk tanpa sengaja dalam aplikasi dan peluang bagi pelaku kejahatan komputer untuk melakukan penipuan dengan membuat perubahan program yang ilegal.

Pengendalian akses dalam lingkungan TI mencakup banyak tingkatan risiko. Pengendalian yang menangani risiko ini meliputi berbagai teknik yang dirancang untuk membatasi akses otoritas personel, membatasi akses ke program komputer, memberikan keamanan fisik pusat pemrosesan data, memastikan adanya cadangan yang memadai atas *file database*, serta menyediakan kemampuan untuk pemulihan dari bencana. Tiap-tiap orang seharusnya diberikan akses ke data, program, dan area terbatas hanya ketika ada kebutuhan yang terkait dengan pekerjaan yang ditugaskan dapat ditunjukkan.

## F. Verifikasi Independen

Prosedur verifikasi adalah pemeriksaan independen terhadap sistem akuntansi untuk mendeteksi kesalahan dan kesalahan penyajian. Verifikasi berbeda dengan supervisi, karena verifikasi dilakukan setelah kejadian, oleh seseorang yang bukan secara langsung terlibat dalam transaksi terkait atau pekerjaan yang diverifikasi. Supervisi dilakukan ketika aktivitas dilaksanakan oleh seorang supervisor yang tanggung jawab langsung atas pekerjaan yang terkait. Dengan prosedur verifikasi independen, pihak manajemen bisa menilai (1) kinerja perorangan, (2) integritas sistem pemrosesan transaksi, dan (3) kebenaran data yang berada dalam catatan akuntansi. Contoh verifikasi independen meliputi:

- Rekonsiliasi total *batch* di beberapa titik saat pemrosesan transaksi.
- Perbandingan aset fisik dan catatan akuntansinya.
- Rekonsiliasi akun buku pembantu dengan akun pengendalinya.
- Kajian atas laporan manajemen yang meringkas aktivitas bisnis.

Pengendalian verifikasi independen di lingkungan TI, program komputer melakukan banyak pekerjaan rutin, maka hal yang perlu diperhatikan adalah pada integritas aplikasi. Misalnya, setelah data di-*input* ke dalam sistem, program pemeriksa dapat dijalankan untuk mencari anomali seperti *field* kosong, nilai di luar

kisaran, atau kunci luar yang hilang. Laporan dari program pemeriksa ini dapat digunakan untuk memverifikasi integritas data setelah pengetikan. Para auditor TI melakukan fungsi verifikasi independen dengan mengevaluasi pengendalian atas pengembangan sistem dan aktivitas pemeliharaan sistem dengan mengkaji logika internal program.

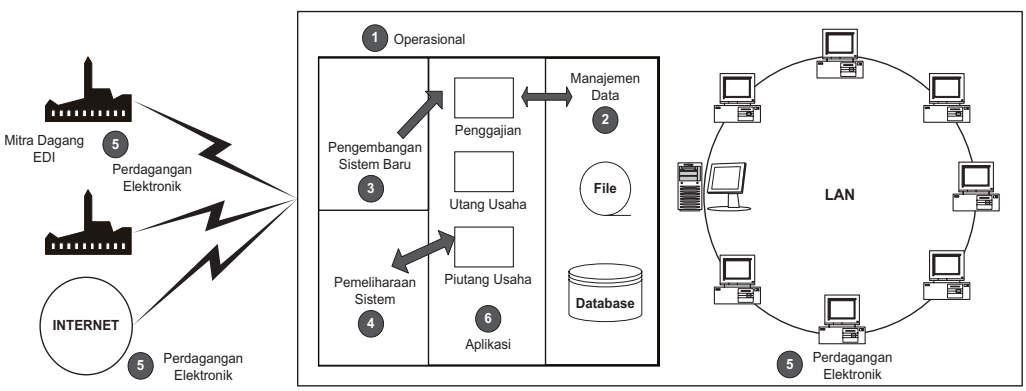
## 2.7 KERANGKA KERJA UMUM UNTUK MELIHAT RISIKO TEKNOLOGI INFORMASI DAN PENGENDALIANNYA

Berbagai area yang memiliki potensi risiko terbesar ditunjukkan dengan angka-angka yang dilingkari, berdasarkan enam topik berikut ini.

1. Operasional.
2. Sistem manajemen data.
3. Pengembangan sistem baru.
4. Pemeliharaan sistem.
5. Perdagangan elektronik (*e-commerce*).
6. Aplikasi komputer.

Seperti yang telah disebutkan sebelumnya, pengendalian teknologi informasi dibagi ke dalam dua kategori umum, yaitu pengendalian umum dan pengendalian aplikasi. Pengendalian umum berlaku untuk berbagai jenis risiko yang secara sistematis mengancam integritas semua aplikasi yang diproses di dalam lingkungan TI. Pengendalian umum adalah topik yang diberi nomor 1 sampai 5.

Pengendalian aplikasi secara sempit berfokus pada berbagai risiko yang berhubungan dengan sistem tertentu, seperti penggajian, piutang, dan pembelian. Berbagai uji pengendalian aplikasi berhubungan dengan tujuan audit tertentu, seperti verifikasi kelengkapan utang usaha.




Gambar 2.4
Kerangka Kerja untuk Melihat Risiko TI

### Penaksiran Risiko (Risk Assessment)

Untuk tujuan pelaporan keuangan, akses terhadap risiko menunjukkan mengenai identifikasi, analisis dan pengelolaan risiko perusahaan yang berkaitan dengan pembuatan laporan keuangan sesuai dengan standar akuntansi yang berlaku. Pentingnya manajemen memperhitungkan risiko yang dapat membuat perusahaan tidak mencapai tujuannya atau bahkan dapat menimbulkan kebangkrutan.

Akses manajemen terhadap risiko bisnis perusahaan pada dasarnya menyerupai akses auditor terhadap pengendalian, yaitu bahwa manajemen mengakses risiko sebagai bagian dari perancangan dan pelaksanaan pengendalian internal untuk meminimalkan kesalahan dan penyalahgunaan, sedangkan auditor mengakses risiko untuk menentukan bukti-bukti yang diperlukan dalam pelaksanaan pekerjaan auditnya. Bila manajemen merespons atau dapat mengakses risiko secara efektif, maka auditor biasanya akan dapat mengumpulkan bukti dalam jumlah yang lebih sedikit dibandingkan dengan bila manajemen gagal dalam melakukannya.

Dalam kaitannya dengan penaksiran risiko ini IAI menghendaki agar auditor memperoleh pengetahuan yang memadai mengenai "proses penaksiran risiko entitas untuk memahami bagaimana manajemen mempertimbangkan risiko yang relevan dengan tujuan pelaporan keuangan dan memutuskan tentang tindakan yang ditujukan risiko tersebut. Pengetahuan ini mungkin mencakup pemahaman tentang bagaimana manajemen mengidentifikasi risiko, melakukan estimasi signifikan atas risiko, menaksir kemungkinan terjadinya, dan menghubungkannya dengan pelaporan keuangan." (SA Seksi 319 [PSA No. 69] Paragraf 30).

## 2.8 PENGENDALIAN INTERNAL DI LINGKUNGAN TEKNOLOGI INFORMASI

Tujuan sistem pengendalian internal berbasis komputer adalah untuk membantu manajemen untuk mencapai keseluruhan pengendalian internal termasuk di dalam kegiatan manual, mekanis, dan program komputer yang terlibat dalam pemrosesan data berbasis komputer.

Jika audit dilaksanakan atas satuan usaha (organisasi/perusahaan) yang berbasis komputer, maka auditor harus mengetahui dan memahami bagaimana pengendalian dalam sistem berbasis komputer yang berlaku pada satuan usaha yang akan diperiksa.

SAS (*section* 321.07) pengendalian dalam sistem berbasis komputer meliputi hal-hal sebagai berikut.

1. Rencana struktur dan pengoperasian sistem berbasis komputer.
2. Prosedur pendokumentasian, audit, pengujian dan persyaratan atau sistem perubahannya.
3. Pengendalian yang tercakup dan melekat dalam komputer tersebut (*hardware control*).
4. Pengendalian pada manusia yang mengerjakan dan mengakses pada komputer dan arsip.

5. Prosedur pengendalian lainnya yang berkaitan dengan operasi berbasis komputer.

Penggunaan semua pengendalian di lingkungan berbasis komputer mungkin tidak praktis bila ukuran bisnis adalah kecil atau komputer mikro digunakan tanpa melihat ukuran bisnis. Demikian pula, bila data diolah pihak ketiga, pertimbangan lingkungan karakteristik sistem komputer dapat bervariasi tergantung atas tingkat akses pengolahan yang dilakukan oleh pihak ketiga tersebut. Secara garis besar pengendalian di lingkungan sistem informasi berbasis teknologi informasi ada dua, yaitu pengendalian umum dan pengendalian aplikasi.

### 2.8.1 Pengendalian Umum

Pengendalian yang berkaitan dengan keseluruhan atau beberapa sistem aplikasi yang ada. Pengendalian ini memberikan keyakinan bahwa tujuan pengendalian internal umum ini mencakup pengendalian organisasi dan manajemen. Pengendalian ini berupaya mengawasi struktur organisasi dan manajemen kegiatan Sistem Informasi Berbasis Komputer (SIBK).

| **Tujuan** |
| --- |
| Adalah untuk membuat kerangka pengendalian menyeluruh atas aktivitas PDE, dan untuk memberikan tingkat keyakinan memadai bahwa tujuan pengendalian internal secara keseluruhan dapat tercapai. |

Pengendalian umum meliputi:

- Pengendalian organisasi dan manajemen.
- Pengendalian terhadap pengembangan dan pemeliharaan sistem aplikasi.
- Pengendalian terhadap operasi sistem.
- Pengendalian terhadap perangkat lunak sistem.
- Pengendalian terhadap entri data dan program.

#### A. Pengendalian Organisasi dan Manajemen

Pengendalian ini didesain untuk menciptakan kerangka organisasi aktivitas Sistem Informasi Berbasis Komputer yang mencakup:

(1) Kebijakan dan prosedur yang berkaitan dengan fungsi pengendalian.
Pusat pengolahan data harus memiliki rencana rekonstruksi dan perbaikan program dan *file* data untuk menghadapi kegagalan atau gangguan sistem informasi. Untuk itu perusahaan harus melakukan analisis risiko dan rencana kontinjensi.

Di dalam rencana perbaikan harus dihindari kehilangan atau kerusakan data. Selain itu, kegiatan pengolahan data diharapkan dapat berjalan terus. Berikut ini prosedur *backup* dan perbaikan yang memadai.

- Membentuk *file backup* yang disesuaikan dengan jenis pengolahan data.

- Melindungi sistem dari gangguan listrik dengan menyediakan generator listrik atau *battery backup*, dan pemakaian UPS.
- Melindungi sistem dari kemungkinan terinfeksi virus dengan membangun *firewall*, pemakaian antivirus, *update* antivirus secara periodik, dan pembatasan akses ke sistem.

(2) Pemisahan fungsi seperti penyiapan transaksi masukan, pemrograman, dan operasi komputer.

- Pemisahan antara fungsi-fungsi dalam unit pengolahan data, yaitu analisis sistem, pemrograman, operasional, librarian, dan pengendali data.
- Pemisahan antara *user* dengan unit pengolah data.

## B. Pengendalian Pengembangan dan Pemeliharaan Sistem Aplikasi

Pengendalian ini didesain untuk memberikan keyakinan yang memadai bahwa sistem dikembangkan dan dipelihara dalam suatu cara yang efisien dan melalui proses otorisasi. Pengendalian ini juga didesain untuk menciptakan pengendalian atas: (a) pengujian, perubahan, implementasi, dan dokumen sistem baru atau sistem yang direvisi, (b) perubahan terhadap sistem aplikasi, (c) akses terhadap dokumentasi sistem, (d) pemerolehan sistem aplikasi dan *listing* program dari pihak ketiga.

(1) Pengendalian Pengembangan Sistem

Studi kelayakan dari segi ekonomi, operasional, dan teknik harus dilakukan terhadap sistem yang berjalan dan sistem yang akan dikembangkan.

Pengendalian di dalam pengembangan sistem baru atau modifikasi sistem:

- Proposal atau permintaan pengembangan dan modifikasi sistem harus dibuat secara tertulis oleh *user* dan diotorisasi oleh komite atau manajer yang ditunjuk untuk hal itu.
- Menetapkan standar desain sistem dan pemrograman yang menggambarkan persyaratan sistem dan kebutuhan *user*.
- Untuk modifikasi program, hendaknya proses desain dilakukan terhadap salinan sistem, tidak boleh sistem yang sedang digunakan.
- Setelah sistem didesain, sistem harus diuji dengan menggunakan data uji yang memadai dan melibatkan *user*, auditor internal, serta personel PDE yang tidak terlibat pengembangan sistem.
- Setiap kegiatan pengembangan sistem harus didokumentasikan.
- Untuk menghindari hukum, hindari penggunaan sistem atau program bajakan.

(2) Pengendalian Dokumentasi Sistem

Dokumentasi sistem mencakup kumpulan dokumen yang mendukung dan menjelaskan sistem aplikasi, termasuk di dalamnya dokumentasi pengembangan sistem. Jenis-jenis dokumentasi, di antaranya adalah sebagai berikut.

- Dokumentasi pendefinisian masalah, yaitu dokumen yang berisi gambaran umum sistem secara cepat dan mudah, tanpa perlu melihat program secara terperinci.

- Dokumentasi sistem, yaitu dokumentasi alur data dalam sistem informasi.
- Dokumentasi program, yaitu dokumentasi atas kode program secara terperinci.
- Dokumentasi operasi, berisi berbagai macam petunjuk mengenai tata cara mengoperasikan sistem.
- Dokumentasi *user*, yaitu dokumen yang membantu pengguna dalam pelaksanaan pengolahan data.

### C. Pengendalian terhadap Operasi Sistem

Pengendalian ini didesain untuk mengendalikan operasi sistem dan untuk memberikan keyakinan memadai bahwa:

- sistem digunakan hanya untuk tujuan yang telah diotorisasi;
- akses ke operasi komputer dibatasi hanya petugas yang mendapat otorisasi;
- hanya program yang telah diotorisasi yang digunakan;
- kekeliruan pengolahan dapat dideteksi dan dikoreksi.

### D. Pengendalian terhadap Perangkat Lunak

Pengendalian ini didesain untuk memberikan keyakinan bahwa perangkat lunak sistem diperoleh atau dikembangkan dengan cara yang efisien dan melalui proses otorisasi termasuk:

- otorisasi, pengesahan, pengujian, implementasi dan dokumentasi perangkat lunak sistem baru dan modifikasi perangkat lunak sistem;
- pembatasan akses terhadap perangkat lunak dan dokumentasi sistem hanya bagi petugas yang telah mendapat otorisasi.

### E. Pengendalian terhadap Entri Data dan Program

Pengendalian ini didesain untuk memberikan keyakinan bahwa: (a) struktur organisasi telah ditetapkan atas transaksi yang dimasukkan ke dalam sistem, dan (b) akses ke data program dibatasi hanya bagi karyawan yang telah mendapat otorisasi.

### F. Pengendalian Lain untuk Keamanan SIBK

Pengendalian lain untuk menjaga keamanan SIBK adalah: (a) pembuatan cadangan data program komputer di lokasi diluar perusahaan; (b) prosedur pemulihan jika terjadi pencurian, kerugian, atau penghancuran data yang disengaja maupun tidak disengaja; dan (c) penyediaan pengolahan di lokasi diluar perusahaan dalam hal terjadi bencana.

## 2.8.2 Pengendalian Aplikasi

Tujuan pengendalian aplikasi (*application control*) PDE adalah: Untuk menetapkan prosedur pengendalian khusus atas aplikasi akuntansi dan untuk memberikan

keyakinan yang memadai bahwa semua transaksi telah diotorisasi dan dicatat serta diolah seluruhnya dengan cermat dan tepat waktu.

## A. Pengendalian Input

Pengendalian *input* adalah pengendalian yang dilakukan untuk menjamin bahwa data yang diterima untuk diproses dalam komputer telah dikonversi dalam sistem, dijumlahkan, dan dicatat dengan benar. Pengendalian ini didesain untuk memberikan keyakinan yang memadai bahwa: (a) transaksi diotorisasi sebagaimana mestinya sebelum diolah dengan komputer, transaksi yang diproses hanya transaksi yang sudah benar-benar disetujui; (b) transaksi diubah dengan cermat ke dalam bentuk yang dapat dibaca mesin dan dicatat dalam *file* data komputer, transaksi ini di-*input* ke mesin komputer dan dicatat pada *file* dengan tepat; (c) transaksi tidak hilang, ditambah, digandakan, atau diubah dengan tidak semestinya atau diubah secara salah; dan (d) transaksi yang keliru ditolak, dikoreksi, dan jika perlu, dimasukkan kembali pada waktu yang tepat.

Berikut ini adalah pengendalian *input* yang sering digunakan, antara lain:

- Error Listing
- Field Checks
- Financial Total
- Hash Total
- Limit Check
- Range Check
- Preformating
- Reasonableness Test
- Record Count
- Self Checking Digit
- Sequence Check
- Sign Check
- Validity Check
- Key Verification
- Redudancy Check
- Echo Check
- Completeness Check
- Internal Header dan Trailer Label

## B. Pengendalian Proses

Pengendalian ini disebut juga dengan pengendalian proses (*processing control*), yaitu pengendalian yang dilakukan untuk menjamin bahwa proses operasi PDE telah dilaksanakan sesuai dengan yang telah direncanakan. Misalnya, transaksi diproses setelah mendapat otorisasi, dan tidak ada transaksi yang diotorisasi, dihilangkan, atau ditambah.

Pengendalian ini didesain untuk memberikan keyakinan bahwa: (a) transaksi—termasuk transaksi yang dipicu melalui sistem—diolah dengan semestinya oleh komputer, (b) transaksi tidak hilang, ditambah, digandakan, atau diubah dengan cara yang tidak sah atau tidak semestinya, dan (c) kekeliruan dalam pemrosesan atau pengolahan data diidentifikasi dan dikoreksi pada waktu yang tepat.

## C. Pengendalian Output

Pengendalian keluaran (*output*) adalah pengendalian yang dilakukan untuk menjamin bahwa: (1) hasil *print out* komputer ataupun *display* telah dilakukan dengan teliti dan benar, dan (2) menjamin bahwa hasilnya diberikan kepada personel yang berhak.

Pengendalian ini didesain untuk memberikan keyakinan yang memadai bahwa: (a) hasil pengolahan atau proses komputer adalah akurat (cermat), (b) akses terhadap keluaran hasil *print out* komputer hanya dibenarkan bagi petugas tertentu yang berhak, (c) hasil komputer keluaran diberikan kepada atau disediakan untuk orang yang tepat pada waktu yang tepat pula, yang telah mendapat otorisasi sebagaimana mestinya. Berikut ini adalah pengendalian *output* yang sering digunakan, antara lain:

- Error Listing
- Console Log
- Distribution
- User Review

## D. Pengendalian Sistem Online

***Pengendalian masukan dalam sistem online.***

Pengendalian ini didesain untuk memberikan keyakinan yang memadai bahwa: (a) transaksi telah dilakukan entri data ke terminal yang semestinya, (b) dilakukan entri dengan cermat, (c) data yang dientri telah diklasifikasikan dengan benar pada nilai transaksi yang sah (valid), (d) data yang tidak sah (invalid) tidak dilakukan entri pada saat transaksi, (e) transaksi tidak dilakukan entri lebih dari sekali, dan (f) data yang dientri tidak hilang selama transaksi berlangsung.

***Pengendalian pengolahan dalam sistem online.***

Pengendalian ini didesain untuk memberikan keyakinan bahwa: (a) hasil perhitungan telah diprogram dengan benar, (b) logika yang digunakan dalam proses pengolahan adalah benar, (c) *file* yang digunakan dalam proses pengolahan adalah benar, (d) *record* yang digunakan dalam proses pengolahan adalah benar, (e) operator telah memasukkan data ke *computer console* sebagaimana mestinya, (f) label yang digunakan selama proses pengolahan adalah benar, (g) selama proses pengolahan telah digunakan standar operasi (*default*) yang semestinya, (h) data yang tidak sah tidak digunakan dalam proses pengolahan, (i) proses pengolahan tidak maenggunakan program dengan versi yang salah, (j) hasil perhitungan yang dilakukan secara otomatis oleh program adalah sesuai dengan kebijakan manajemen perusahaan, dan (k) data masukan yang diolah adalah data yang berotorisasi.

***Pengendalian keluaran pada sistem online.***
Pengendalian didesain untuk memberikan keyakinan bahwa: (a) keluaran yang diterima perusahaan adalah tepat dan lengkap, (b) keluaran yang diterima perusahaan telah diklasifikasi, dan (c) keluaran didistribusikan kepada pegawai yang telah diotorisasi.

## 2.9 PENGENDALIAN SISTEM KEAMANAN

Sistem keamanan komputer adalah suatu sistem proteksi fasilitas komputer yang terdiri dari *hardware*, *software*, dan data dari kerusakan atau gangguan yang disebabkan oleh manusia maupun lingkungan (alam).

- Lingkungan → Kebakaran; Banjir; Bencana Alam; Kondisi Udara; dan sebagainya.
- Manusia → Penyadapan; Pencurian; Perusakan; dan sebagainya.

### 2.9.1 Kejahatan Komputer

Berikut ini adalah contoh-contoh kegiatan atau kejahatan komputer yang dapat dikurangi dengan pengendalian administratif, fisik, dan teknikal.

1. Pencurian.
2. Penipuan (*fraud*).
3. Sabotase.
4. *Blackmail.*
5. Spionase industrial.
6. Penyampaian informasi tanpa izin.
7. Merusak kredibilitas.
8. Penghilangan kepemilikan informasi.

Berdasarkan data FBI pada tahun 2005 – 2006, ada tiga macam kasus kejahatan komputer yang paling sering terjadi, yaitu serangan virus, akses tanpa izin, dan pencurian informasi. Pada saat ini ada peningkatan kasus-kasus akses tanpa izin dan pencurian informasi.

#### A. Mengidentifikasi Pelaku

Saat ini pelaku kejahatan komputer tidak memerlukan kemampuan tingkat lanjut untuk melakukan kejahatan. Seseorang dapat dengan mudah melakukan mengakses sumber daya dan melakukan kejahatan. FBI melaporkan sejak tahun 2005 jumlah serangan dari dalam hampir sama dengan serangan dari luar.

***Hacker***
Istilah ini memiliki dua pengertian, yaitu seorang *programmer* yang dapat membuat program yang ada dapat digunakan. Akan tetapi, pelaku ini merupakan kriminal yang tidak diinginkan. Pelaku *hacker* fokus pada keinginan untuk memutuskan,

mengambil alih, dan merusak atau menghilangkan legitimasi pemrosesan komputer. Tujuan pertama dari *hacker* adalah bisa lolos dari level otorisasi sistem.

Pelaku *hacker* bisa dari dalam maupun luar organisasi. Usaha untuk mencegah pelaku ini dengan menggunakan otorisasi tingkat tinggi secara tegas, termasuk pemberhentian langsung terhadap karyawan tersebut.

***Cracker***

Istilah ini sangat bervariasi, istilah *cracker* bisa seseorang yang secara ilegal masuk ke sistem tanpa otorisasi.

***Script Kiddies***

Spesialisasi *programmer* dengan tujuan untuk melakukan *bypass* pengendalian keamanan. Orang ini membuat program yang digunakan agar dapat lolos dari sistem keamanan. Saat ini banyak program yang digunakan untuk memecahkan *password* atau *serial number*, sehingga *user* dapat menggunakan sistem atau program secara ilegal.

***Karyawan Sakit Hati (Orang Dalam)***

Karyawan yang memiliki hak akses yang lebih daripada yang lainnya dan memahami konstruksi dan kelemahan sistem keamanan. Kemudian, karyawan melakukan tindakan-tindakan yang dapat mengacaukan sistem karena motif balas dendam terhadap organisasi.

***Pihak Ketiga***

Personal dari luar seperti pengunjung, vendor, konsultan, personel *maintenance*, dan *cleaning service*. Individu tersebut berusaha memperoleh hak akses dan keadaan internal organisasi.

***Orang Tidak Sengaja***

Personal yang kurang akan pengetahuan, ia melakukan tindakan yang tanpa disadari merupakan tindakan kejahatan.

### 2.9.2 Metode Serangan

#### A. Serangan Pasif

Serangan ini dikategorikan dengan teknik pengamatan. Tujuan dari serangan ini adalah untuk memperoleh informasi tambahan sebelum melakukan serangan aktif. Contoh serangan pasif adalah menganalisis jaringan, analisis lalu lintas data, dan penyadapan.

#### B. Serangan Aktif

Serangan pasif relatif tidak terlihat, sedangkan serangan aktif sangat mudah untuk terdeteksi. Serangan aktif akan dilakukan setelah mendapat informasi cukup. Serangan ini didesain untuk melakukan tindakan pencurian atau membuat kekacauan pada sistem pengolahan data.

### 2.9.3 Proteksi Administratif

#### A. Manajemen Keamanan Informasi

Tujuan dari manajemen keamanan informasi adalah untuk memastikan atau menjamin kepercayaan, integritas, dan ketersediaan sumber daya informasi. Untuk menunjang hal tersebut, maka diperlukan manajemen sebagai berikut.

- *Chief Security Officer*
- *Chief Privacy Officer*
- *Information System Security Manager*

#### B. Undang-Undang Keamanan TI

Konsep ini merujuk pada kebijakan, standar, dan prosedur keamanan yang dikeluarkan oleh pemerintah.

Berikut ini merupakan peraturan otoritas data.

- Pemilik data
- Pengguna data
- Pengawas data
- Identifikasi kebutuhan media simpan
- Penghentian akses
- Penanganan insiden
- Pelaporan informasi pengecualian (*exception information*)

#### C. Proteksi Secara Fisik

- CCTV
- Kunci Khusus
- Biometric
- Alarm
- Sensor
- Lokasi Pengolahan Data
- Listrik Emergensi
- Pengaturan Udara

### 2.9.4 Menyediakan/Pengendalian Sistem yang Aman

#### A. Manajemen Keamanan

- *Membangun keamanan objektif*, yaitu memproteksi peralatan komputer, fasilitas, program, dan data dari kebakaran, gempa bumi, kebanjiran, kerusuhan, dan sebagainya.
- *Evaluasi risiko keamanan*, manajemen harus mengevaluasi risiko keamanan pengolahan data dan biaya yang ditimbulkannya.
- *Membuat perencanaan keamanan*, manajemen harus merencanakan tingkat keamanan yang dapat disesuaikan dengan biaya yang ada.

- *Pemberian tanggung jawab*, manajemen harus memberikan tanggung jawab untuk sistem keamanan. Tanggung jawab ini meliputi implementasi perencanaan, dan *monitoring* sistem keamanan.
- *Tes sistem keamanan,* pengendalian sistem keamanan harus sudah dites oleh manajemen apakah dapat mendeteksi dan mencegah gangguan keamanan dan dapat menyediakan sistem perbaikan.
- *Evaluasi sistem keamanan*, hasil dari pengujian (*testing*) pengendalian sistem keamanan harus dapat digunakan untuk mengevaluasi tingkat keefektifan pengendalian.

### B. Pengendalian Akses Fasilitas

Didesain untuk melindungi gedung komputer dan peralatannya dari kerusakan fisik.

### C. Pengendalian Lokasi

Walaupun dimungkinkan lokasi pusat komputer dapat dikendalikan dari jarak jauh dari lingkungan, teknologi, dan sosial. Akan tetapi, risiko kerusakan dari gempa bumi dan banjir dapat dikurangi dengan menyediakan area bebas banjir dan gedung antigempa.

### D. Pengendalian Konstruksi

Menyediakan konstruksi bangunan untuk fasilitas komputer yang dapat mengurangi risiko kerusakan dari bahaya lingkungan atau keamanan. Konstruksi tersebut digunakan untuk mengisolasi secara fisik fasilitas komputer.

### E. Pengendalian Akses Data

Pengendalian akses data ke fasilitas komputer diperlukan untuk mencegah akses yang tidak berhak.

### F. Pengendalian Library

Digunakan untuk membatasi akses *file* data, program komputer, dan dokumentasi sistem. Akses *library* sama halnya akses aset perusahaan.

### G. Pengendalian Akses Online

Pengedalian akses fisik langsung ke ruang komputer mungkin sudah tersedia, tetapi belum cukup untuk sistem *online*. Sistem *online* menggunakan suatu terminal yang biasanya ditempatkan di luar ruang komputer. Akses *online* harus dibatasi dengan beberapa cara, yaitu: keamanan fisik terminal, pengendalian otorisasi (terminal dan *user*), pengendalian identifikasi, dan pengendalian akses komunikasi data.

### 2.9.5 Deteksi Kegagalan Sistem Keamanan

- *Peralatan Deteksi*, alat deteksi kebakaran, otorisasi akses, dan sebagainya.
- *Autentikasi*, identifikasi *user* yang berhak untuk akses.
- *Sistem Monitoring*, memantau jalannya sistem keamanan yang sedang berjalan.

**Perbaikan kegagalan sistem keamanan**

1. Pemadam kebakaran.
2. Asuransi.
3. Prosedur *bypass*, kegagalan sistem dapat dikurangi dengan metode prosedur *bypass offline* di dalam sistem *online*. Prosedur *bypass* merupakan metode alternatif secara *offline* selama sistem *online* rusak.
4. Perencanaan perbaikan, untuk memastikan manajemen telah mengevaluasi kegagalan sistem keamanan dan mengadopsi prosedur tersebut jika diperlukan untuk meminimalisasi kehilangan data dan aset perusahaan.
5. Prosedur perbaikan, mencakup fasilitas dan peralatan komputer, *software*, data, personal, dan *supplay*.

## 2.10 STANDAR (COBIT)

Control Objectives for Information and Related Technology (COBIT), saat ini edisi ke-5, adalah sekumpulan dokumentasi *best practices* untuk *IT governance* yang dapat membantu auditor, manajemen dan pengguna (*user*) untuk menjembatani gap antara risiko bisnis, kebutuhan pengendalian, dan permasalahan teknis. COBIT dikembangkan oleh IT Governance Institute, yang merupakan bagian dari Information Systems Audit and Control Association (ISACA). COBIT memberikan arahan (*guidelines*) yang berorientasi pada bisnis, sehingga *business process owners* dan manajer, termasuk juga auditor dan *user*, diharapkan dapat memanfaatkan *guideline* ini dengan sebaik-baiknya.

### 2.10.1 Definisi COBIT

COBIT adalah alat bantu untuk pengelolaan teknologi informasi (*IT Governance Tool*), dan diharapkan dapat membantu para profesional TI bekerja dengan benar. COBIT mengembangkan tatanan atau ketentuan *IT Control Objective* yang dapat diterima oleh umum berikut panduan lengkap (*audit guidelines*) yang memungkinkan penerapan kerangka kerja dan tujuan pengendalian dapat berjalan dengan mudah. Tatanan atau ketentuan tersebut dapat digunakan oleh manajer dalam menjalankan perusahaan atau auditor dalam menjalankan profesinya. Diharapkan COBIT dapat dijadikan satu-satunya model pengelolaan dan pengendalian teknologi informasi (*information technology governance*).

COBIT merupakan serangkaian kebijakan, prosedur, praktik, dan struktur organisasi yang dirancang untuk menyiapkan keyakinan yang mendasar, bahwa tujuan organisasi/perusahaan akan dapat dicapai, dan hal-hal yang tidak dikehendaki

akan terdeteksi atau terkoreksi. COBIT juga mengadaptasi definisi dari *IT Control Objectives* yang dikeluarkan oleh System Auditability and Control (SAC), yaitu suatu pernyataan tentang hasil yang dikehendaki atau direncanakan untuk dicapai dengan menerapkan prosedur pengendalian di dalam kegiatan yang terkait dengan teknologi informasi.

### 2.10.2 Ruang Lingkup COBIT

COBIT *framework* menyatakan bahwa ruang lingkup, batasan atau *scope* audit harus termasuk *internal control systems* menyangkut penggunaan dan perlindungan sumber daya teknologi informasi (sistem informasi). Ada sumber daya sistem informasi:

a. ***Data***, dalam pengertian luas data bukan hanya meliputi angka-angka, teks, atau tanggal, tetapi termasuk juga objek lain yang ditampilkan sebagai suara, grafik, maupun video.
b. ***Aplikasi Sistem***, seluruh prosedur baik yang manual maupun yang terprogram.
c. ***Teknologi***, mencakup/berisi pengertian perangkat keras, sistem operasi, perangkat jaringan, dan sejenisnya.
d. ***Fasilitas***, sumber daya yang dipergunakan untuk menampung dan mendukung sistem informasi.
e. ***Manusia***, SDM yang dimaksud sebagai individu yang terampil dan memiliki kemampuan untuk merencanakan, mengorganisasi, menghimpun, membagikan, mendukung, dan memonitor sistem informasi dan pelayanan informasi.

Konsep pengendalian internal telah dikembangkan oleh berbagai profesi auditing dan profesi akuntansi, seperti AICPA (American Institute of Certified Public Accountant), GAO (Government Accounting Office), IIA (Institute of Internal Auditing), serta COSO (Committee of Sponsoring Organization of the Treadway Commission), adalah beberapa lembaga yang telah mengembangkan konsep pengendalian. Walaupun cara masing-masing lembaga tersebut dalam mengembangkan konsep pengendalian berbeda-beda, tetapi memiliki substansi pembahasan yang sama. Kesamaan substansi tersebut karena kosenp pengendalian adalah konsep deskriptif, yaitu konsep yang dikembangkan berdasarkan pada pengamatan atas pengelolaan yang dilakukan oleh manajer organisasi dan perusahaan.

Pada perkembangan pengendalian selanjutnya, berdasarkan hasil riset yang dilakukan ISACF (Information System Audit and Control Foundation), muncul konsep pengendalian di lingkungan sistem informasi yang diluncurkan pada bulan April 1998 dengan nama COBIT (Control Objectives for Information and Related Technology).

Kerangka kerja COBIT terdiri atas beberapa arahan (*guidelines*), meliputi:

1. ***Control Objectives*:** Terdiri atas 4 tujuan pengendalian tingkat tinggi (*high-level control objectives*) yang tercermin dalam 4 domain, yaitu: *planning & organization, acquisition & implementation, delivery & support*, dan *monitoring*.

2. ***Audit Guidelines*:** Berisi sebanyak 318 tujuan pengendalian terperinci (*detailed control objectives*) untuk membantu para auditor dalam memberikan *management assurance* dan/atau saran perbaikan.
3. ***Management Guidelines*:** Berisi arahan, baik secara umum maupun spesifik, mengenai apa yang mesti dilakukan, terutama agar dapat menjawab pertanyaan-pertanyaan berikut:
   - Sejauh mana Anda (TI) harus bergerak, dan apakah biaya TI yang dikeluarkan sesuai dengan manfaat yang dihasilkannya?
   - Apa saja indikator untuk suatu kinerja yang bagus?
   - Apa saja faktor atau kondisi yang harus diciptakan agar dapat mencapai sukses (*critical success factors*)?
   - Apa saja risiko yang timbul, apabila kita tidak mencapai sasaran yang ditentukan?
   - Bagaimana dengan perusahaan lainnya—apa yang dilakukannya?
   - Bagaimana Anda mengukur keberhasilan dan bagaimana pula membandingkannya?

   Kerangka kerja COBIT memasukkan juga hal-hal sebagai berikut.
   - *Maturity Models*—Untuk memetakan status *maturity* proses TI (dalam skala 0 – 5) dibandingkan dengan "*the best in the class in the industry*" dan juga *international best practices.*
   - *Critical Success Factors* (CSF)—Arahan implementasi bagi manajemen agar dapat melakukan pengendalian atas proses TI.
   - *Key Goal Indicators* (KGI)—Kinerja proses TI sehubungan dengan *business requirements.*
   - *Key Performance Indicators* (KPI)—Kinerja proses TI sehubungan dengan *process goals.*

COBIT dikembangkan sebagai "*generally applicable and accepted standard for good information technology (IT) security and control practices.*"

Istilah "*generally applicable and accepted*" digunakan secara eksplisit dalam pengertian yang sama seperti *Generally Accepted Accounting Principles* (*GAAP*). Sedangkan, *COBIT's "good practices"* mencerminkan konsensus di antara para ahli di seluruh dunia. COBIT dapat digunakan sebagai *IT Governance Tools*, selain membantu perusahaan mengoptimalkan investasi TI mereka. Hal penting lainnya, COBIT dapat juga dijadikan sebagai acuan atau referensi apabila terjadi kesimpangsiuran dalam penerapan teknologi.

Perencanaan audit sistem informasi berbasis teknologi (audit TI) oleh auditor internal dapat dimulai dengan menentukan area-area yang relevan dan berisiko paling tinggi, melalui analisis atas ke-34 proses tersebut. Sementara untuk kebutuhan penugasan tertentu, misalnya audit atas proyek TI, dapat dimulai dengan memilih proses yang relevan dari proses-proses tersebut.

Auditor dapat menggunakan *audit guidelines* sebagai tambahan materi untuk merancang prosedur audit. Singkatnya, COBIT khususnya *guidelines* dapat dimodifikasi dengan mudah, sesuai dengan industri, kondisi TI pada perusahaan atau organisasi, atau objek khusus di lingkungan TI.

Selain dapat digunakan oleh auditor, COBIT dapat juga digunakan oleh manajemen sebagai jembatan antara risiko TI dengan pengendalian yang dibutuhkan (*IT Risk Management*) dan juga referensi utama yang sangat membantu dalam penerapan *IT Governance* di perusahaan.

Seiring dengan makin banyaknya institusi, pemerintahan dan swasta, yang mengandalkan TI untuk mendukung jalannya operasional sehari-hari, maka kesadaran akan perlunya dilakukan *review* atas pengembangan sistem informasi semakin meningkat.

Risiko yang mungkin ditimbulkan sebagai akibat dari gagalnya pengembangan sistem informasi, antara lain:

- Biaya pengembangan sistem melampaui anggaran yang ditetapkan.
- Sistem tidak dapat diimplementasikan sesuai dengan jadwal yang ditetapkan.
- Sistem yang telah dibangun tidak memenuhi kebutuhan pengguna.
- Sistem yang dibangun tidak memberikan dampak efisiensi dan nilai ekonomis terhadap jalannya operasi institusi pada masa sekarang atau masa mendatang.
- Sistem yang berjalan tidak menaati perjanjian dengan pihak ketiga atau memenuhi aturan yang berlaku.

Untuk mengantisipasi hal itu, perusahaan menginginkan adanya *assurance* dari pihak yang berkompeten dan independen mengenai kondisi sistem TI yang akan atau sedang digunakan. Pihak yang paling berkompeten dan memiliki keahlian untuk melakukan *review* tersebut adalah Auditor Sistem Informasi (Auditor TI). Pekerjaan auditor TI ini belum banyak dikenal di Indonesia. Di samping itu, jumlah tenaga auditor TI yang menyandang sertifikasi internasional (*Certified Information System Auditor—CISA*) juga masih sangat terbatas.

*Best Practice* menyarankan agar dalam proses pengembangan suatu sistem informasi yang signifikan, perlu dilakukan *review*, sebelum atau pada saat implementasi (*pre-implementation system*) atau setelah sistem "*live*" (*post-implementation system*).

**Manfaat *pre-implementation review:***

- Institusi dapat mengetahui apakah sistem yang telah dibuat sesuai dengan kebutuhan atau memenuhi *acceptance criteria*.
- Mengetahui apakah pengguna telah siap menggunakan sistem tersebut.
- Mengetahui apakah *outcome* sesuai dengan harapan manajemen.

**Manfaat *post-implementation review:***

- Institusi mendapat masukan atas risiko yang masih ada dan saran untuk penanganannya.

- Masukan tersebut dimasukkan dalam agenda penyempurnaan sistem, perencanaan strategis, dan anggaran pada periode berikutnya.
- Bahan untuk perencanaan strategis dan rencana anggaran di masa datang.
- Memberikan *reasonable assurance* bahwa sistem informasi telah sesuai dengan kebijakan atau prosedur yang telah ditetapkan.
- Membantu memastikan bahwa jejak audit (*audit trail*) telah diaktifkan dan dapat digunakan oleh manajemen, auditor, atau pihak lain yang berwenang untuk melakukan pemeriksaan.
- Membantu dalam penilaian apakah *initial proposed values* telah terealisasi dan saran tindak lanjutnya.

## SOAL DAN STUDI KASUS

1. Jelaskan bagaimana dengan definisi pengendalian internal?
2. Sebutkan empat tujuan umum dari pengendalian internal!
3. Apa tujuan dari SAS No. 78?
4. Jelaskan bagaimana perusahaan atau organisasi membutuhkan pengendalian?
5. Sebutkan struktur pengendalian menurut IAI?
6. Apa yang dimaksud dengan lingkungan pengendalian?
7. Apa yang dimaksud dengan istilah eksposur dan risiko? Bagaimana melakukan penaksiran terhadap risiko?
8. Pada dasarnya pengendalian internal dapat diklasifikasikan menjadi beberapa, di antaranya adalah menurut *Ron Weber*. Sebutkan dan jelaskan apa saja pengendalian internal tersebut!
9. Berikan contoh pengendalian preventif!
10. Berikan contoh pengendalian detektif!
11. Berikan contoh pengendalian korektif!
12. Jika pengendalian detektif menandakan adanya kesalahan, mengapa pengendalian tersebut tidak secara otomatis melakukan perbaikan atas kesalahan yang teridentifikasi tersebut? Mengapa dibutuhkan adanya pengendalian korektif secara terpisah?

# BAB 3

# SISTEM INFORMASI BERBASIS KOMPUTER

Setelah mempelajari bab ini, Anda diharapkan mampu:

- Memahami tentang konsep sistem informasi berbasis komputer dan perananannya di dalam dunia bisnis.
- Mengetahui dan memahami aspek-aspek yang membentuk sistem informasi serta struktur fungsi TI.
- Memahami berbagai isu pengendalian dan audit yang berkaitan dengan strukturisasi fungsi TI terpusat dan berbagai pendekatannya.
- Mengetahui pengendalian pusat komputer dan berbagai prosedur yang digunakan untuk mengujinya.
- Memahami peran penting dalam elemen-elemen dasar pemulihan dari bencana yang efektif dan berbagai isu pengendalian serta audit yang berkaitan dengannya.
- Memahami berbagai risiko khusus yang berkaitan dengan lingkungan TI dan berbagai pengendalian yang membantu untuk mengatasinya.

Bab ini akan membahas mengenai konsep sistem informasi berbasis komputer dan beberapa isu audit yang berkaitan dengan operasi komputer. Pembahasan dalam bab ini akan mengulas mengenai: (1) konsep sistem informasi, (2) peranan sistem informasi dalam dunia bisnis, (3) konsep teknologi informasi berserta peranannya di dalam berbagai aspek, (4) aspek-aspek yang membentuk sistem informasi berbagai informasi, (5) struktur fungsi teknologi informasi, (6) pengendalian operasi pusat komputer, sistem operasi komputer, dan pengendalian sistem keseluruhan, serta lingkungan komputer pribadi.

Pada bab ini selain membahas konsep-konsep mengenai sistem informasi berbasis komputer, juga akan membahas mengenai kelebihan dan kekurangan dari struktur organisasi TI terpusat maupun terdistribusi. Tujuan audit teknik pengendalian, serta berbagai prosedur auditing. Bab ini akan menjelaskan gambaran umum mengenai berbagai fitur sistem operasional *multi-user* yang umum digunakan pada *networking* dan *mainframe*, serta pembahasan mengenai tujuan dasar sistem operasional dan berbagai macam risiko mengancam keamanan sistem. Pada bab ini juga akan menjelaskan berbagai isu pengendalian dan audit pada perusahaan secara umum.

## 3.1 KONSEP SISTEM INFORMASI

***Sistem*** pada dasarnya adalah sekelompok unsur atau unit yang saling terkait satu sama lainnya untuk bekerja sama untuk mencapai suatu tujuan. Definisi ini dapat diperinci lebih lanjut mengenai pengertian sistem, yaitu:

1. Setiap sistem terdiri dari unsur-unsur.
2. Unsur tersebut merupakan bagian terpadu yang saling terkait.
3. Unsur sistem saling bekerja sama untuk mencapai tujuan.
4. Sistem merupakan bagian dari sistem lainnya yang lebih besar.

Para pakar juga mendefinisikan sistem sebagi berikut.

- Stephen A. Maschope, *sistem* adalah suatu kesatuan yang terdiri dari interaksi subsistem yang berusaha untuk mencapai tujuan yang sama (4: 1984).
- Frederick H, *sistem* beroperasi dan berinteraksi dengan lingkungannya untuk mencapai tujuan tertentu, sistem menunjukkan tingkah laku melalui interaksi di antara komponen di dalam sistem dan di antara lingkungannya (6: 1984).
- Robert H. Blissmer, *sistem* adalah suatu kumpulan dari bagian yang ditata, berinteraksi bersama-sama untuk melakukan suatu fungsi.

Suatu sistem harus memiliki tiga unsur, yaitu *input*, proses, dan *output*. *Input* merupakan penggerak atau pemberi tenaga di mana sistem itu dioperasikan. *Output* adalah hasil dari operasi. *Output* di sini bisa berarti yang menjadi tujuan, sasaran, atau target pengorganisasian dari suatu sistem. Sedangkan proses adalah aktivitas yang mengubah *input* menjadi *output*.

Informasi adalah rangkaian data yang mempunyai sifat sementara, tergantung dengan waktu, mampu memberi nilai tambah bagi penggunanya. Karakteristik dari informasi adalah pengguna informasi mengalami perubaan dari kondisi belum mengetahui menjadi kondisi mengetahui.

Sistem informasi merupakan kumpulan unsur *hardware*, *software*, *networking*, *database* untuk mengumpulkan, mengubah, dan menyebarkan informasi dalam suatu organisasi. Setiap sistem informasi terdiri dari blok-blok bangunan (komponen utama) yang membentuk sistem informasi. Komponen utama sistem informasi terdiri dari enam blok (*system information building block*), yaitu: *input*, model, *output*, teknologi, *database*, dan pengendalian.

- ***Komponen Input***
  Data yang dimasukkan ke dalam sistem informasi beserta metode dan media yang digunakan untuk merekam atau menangkap dan memasukkan data tersebut ke dalam sistem. Cara memasukkan *input* ke dalam sistem tergantung dari jenis data.
- ***Komponen Model***
  Terdiri dari model matematika dan logika (*logical mathematical models*) yang mengolah masukan (*input*) dan data yang disimpan, dengan berbagai macam cara, untuk menghasilkan *output* yang dibutuhkan. Model ini dapat mengombinasikan unsur-unsur data untuk menyediakan jawaban atas suatu pertanyaan, atau dapat meringkas atau menggabungkan data menjadi suatu laporan ringkas.
- ***Komponen Output***
  Terdiri atas informasi yang memiliki nilai tambah dan dokumen untuk semua tingkat manajemen dan semua pengguna informasi, baik pengguna internal maupun pengguna eksternal organisasi.
- ***Komponen Teknologi***
  Komponen ini merupakan perangkat untuk menangkap *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan menyampaikan keluaran (*output*), serta mengendalikan seluruh sistem. Di dalam sistem informasi berbasis komputer, teknologi terdiri dari tiga komponen, yaitu komputer dan media simpan, telekomunikasi, dan perangkat lunak.
- ***Komponen Database***
  Suatu wadah untuk menampung atau menyimpan data yang digunakan untuk melayani kebutuhan pengguna informasi. *Database* dapat diperlakukan dari dua sudut pandang, yaitu secara fisik dan secara logis. *Database* secara fisik berupa media untuk menyimpan data, seperti pita magnetik, *disk*, *chip* memori, dan *microfilm*. *Database* dari sudut pandang logis terkait dengan bagaimana struktur penyimpanan data sehingga menjamin ketepatan, ketelitian, dan relevansi pengambilan informasi untuk memenuhi kebutuhan pengguna.

- ***Komponen Pengendalian***
  Sistem informasi harus dilindungi dari ancaman atau risiko dari dalam maupun dari luar. Komponen ini untuk mengendalikan jalannya sistem informasi, dengan kata lain untuk memastikan sistem informasi berjalan sesuai dengan yang telah ditetapkan.

## 3.2 PERANAN SISTEM INFORMASI DALAM BISNIS

Tiga peran penting yang dilakukan sistem informasi untuk perusahaan atau lingkungan bisnis, yaitu:

- Mendukung proses dan operasi bisnis.
- Mendukung pengambilan keputusan para pegawai dan manajernya.
- Mendukung berbagai strategi untuk keunggulan kompetitif.

### 3.2.1 Tren Peranan Sistem Informasi

*Era tahun 1950 – 1960*: Pemrosesan data atau sistem *electronic data processing*. Pada masa ini sistem informasi masih sederhana, penggunaanya untuk pemrosesan transaksi, pencatatan, akuntansi, dan aplikasi pemrosesan data elektronik (*electronic data processing*—EDP).

*Era tahun 1960 – 1970*: Pelaporan manajemen, peranan sistem informasi sederhana berkembang seiring dengan konsep sistem informasi manajemen (SIM). Laporan manajemen untuk tingkatan manajerial untuk informasi yang telah ditentukan telebih dahulu dalam rangka mendukung pengambilan keputusan.

*Era tahun 1970 – 1980*: Pendukung pengambilan keputusan (*decision support system*—DSS). Produk informasi yang dihasilkan sistem informasi manajemen belum cukup memenuhi banyak kebutuhan pengambilan keputusan. Pada era ini lahir konsep sistem pendukung pengambilan keputusan. Peranan baru ini memberi dukungan interaktif kepada para pengguna akhir tingkat manajerial dalam proses pengambilan keputusan. Dukungan ini akan dibentuk sesuai dengan gaya pengambilan keputusan yang unik dari para manajer, ketika mereka dihadapkan pada jenis masalah tertentu dalam dunia nyata.

*Era tahun 1980 – 1990*: Dukungan strategis dan pengguna akhir (*sistem end-user computing*). Dukungan komputer langsung untuk produktivitas pemakai akhir dan kerja sama kelompok kerja. *Executive information system* (EIS), yaitu informasi penting untuk pihak manajemen puncak. *Expert system*, yaitu saran ahli berbasis pengetahuan untuk pemakai akhir. Sistem informasi strategis, yaitu produk dan layanan strategis untuk keunggulan kompetitif.

*Era tahun 1990 – 2000*: *E-Business* dan *E-Commerce*. Sistem *e-business* dan *e-commerce* berbasis Internet. Perusahaan yang dijalankan melalui *web* dan operasi *e-business* global serta *e-commerce* melalui Internet, intranet, ekstranet, dan jaringan lainnya.

### 3.2.2 Jenis-Jenis Sistem Informasi

Secara konsep, aplikasi sistem informasi yang diimplementasikan dalam dunia bisnis saat ini dapat diklasifikasikan menjadi sistem informasi operasi atau manajemen.

#### A. Sistem Pendukung Operasi

Sistem informasi selalu dibutuhkan untuk memproses data yang dihasilkan oleh dan digunakan dalam operasi bisnis. Sistem pendukung operasi ini menghasilkan berbagai informasi yang paling dapat digunakan oleh para manajer. Peran dari sistem pendukung operasi di dalam perusahaan atau bisnis adalah untuk secara efisien memproses transaksi bisnis, mengendalikan proses industrial, mendukung komunikasi dan kerja sama perusahaan, serta memperbarui *database* perusahaan.
Rangkuman sistem pendukung operasi:

- **Sistem pemrosesan transaksi**. Memproses data yang dihasilkan dari transaksi bisnis, memperbarui *database* operasional, dan menghasilkan dokumen bisnis.
- **Sistem pengendalian proses**. Mengawasi dan mengendalikan berbagai proses industrial.
- **Sistem kerja sama perusahaan**. Mendukung komunikasi dan kerja sama tim, kelompok kerja, dan perusahaan.

#### B. Sistem Pendukung Manajemen

Pada saat aplikasi sistem informasi berfokus pada penyediaan informasi dan dukungan untuk pengambilan keputusan yang efektif oleh para manajer, maka aplikasi sistem tersebut dapat dikatakan sebagai sistem pendukung manajemen. Memberikan informasi dan dukungan untuk pengambilan keputusan semua jenis manajer serta praktisi bisnis adalah tugas yang rumit.
Jenis utama sistem pendukung manajemen:

- **Sistem informasi manajemen**. Memberikan informasi dalam bentuk laporan yang telah ditentukan sebelumnya untuk mendukung pengambilan keputusan bisnis.
- **Sistem pendukung keputusan** (*decision support system*—DSS). Memberikan dukungan interaktif khusus untuk proses pengambilan keputusan para manajer dan praktisi bisnis lainnya.
- **Sistem informasi eksekutif** (*executive information system*—EIS). Memberi informasi penting dari SIM, DSS, dan sumber lainnya yang dibentuk sesuai kebutuhan informasi para eksekutif.

## 3.3 DEFINISI TEKNOLOGI INFORMASI

Teknologi informasi adalah suatu perangkat yang dapat membantu bekerja dengan informasi dan melakukan tugas-tugas yang berhubungan dengan pemrosesan informasi (Haag dan Keen—1996).

Teknologi informasi tidak hanya terbatas pada teknologi komputer (*hardware* atau *software*) yang digunakan untuk memproses dan menyimpan informasi, melainkan juga mencakup teknologi komunikasi untuk mengirimkan informasi (Martin—1999).

Teknologi informasi adalah teknologi yang menggabungkan komputasi (proses komputer) dengan jalur komunikasi berkecepatan tinggi yang membawa data, suara, dan video (Williams dan Sawyer—2003).

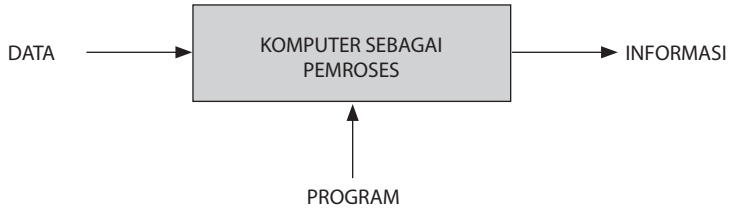
Teknologi informasi adalah gabungan antara teknologi komputer dan teknologi telekomunikasi.

### 3.3.1 Teknologi Komputer

Teknologi yang berhubungan dengan komputer, termasuk peralatan yang berhubungan dengan komputer seperti *printer*, monitor, CD-ROM, dan sebagainya. Komputer adalah suatu mesin elektronik serba guna yang dapat dikendalikan oleh program, digunakan untuk memproses data menjadi informasi.

Program adalah suatu kumpulan atau sederetan instruksi yang digunakan untuk mengendalikan komputer sehingga dapat melakukan tindakan sesuai yang dikehendaki pembuat program (*programmer*).

Data adalah materi bahan mentah yang belum diolah oleh komputer, data ini dapat berupa teks, gambar, video, dan suara. Informasi merupakan hasil dari data yang diolah sehingga dapat digunakan atau bermanfaat untuk pengambilan keputusan.




### 3.3.2 Teknologi Telekomunikasi

Teknologi yang berhubungan dengan komunikasi jarak jauh. Misalnya, telepon, radio, televisi, dan sebagainya.

## 3.4 ASPEK KOMPUTERISASI (SISTEM TERKOMPUTERISASI)

Ada tiga aspek agar sistem komputerisasi dapat berjalan dengan baik sesuai dengan prosedur yang ada, adalah sebagai berikut.

### 3.4.1 Aspek Hardware

Aspek ini berkaitan dengan peralatan pendukung komputerisasi yang mencakup peralatan *input/output* dan peralatan yang pendukung *networking* maupun *internetworking* (Internet).

### 3.4.2 Aspek Software

Sistem komputerisasi tidak akan berjalan dengan baik bila aspek *hardware* yang telah dimiliki tidak dilengkapi dengan prosedur yang digunakan untuk menjalankan suatu proses data agar dapat diolah sedemikian rupa sehingga menghasilkan informasi. Prosedur tersebut dapat berupa:

**Software Sistem Operasi Komputer**
*Software* yang berfungsi untuk mengoperasikan unit komputer agar dapat berjalan dengan semestinya, selain itu *software* OS juga befungsi sebagai perantara antara manusia (*user*) dengan mesin (komputer).

Contoh:
Windows 98, Windows 2000, LINUX, UNIX, Windows NT, Novell Netware, dan lain-lain.

**Software Aplikasi**
*Software* yang memiliki fungsi untuk membantu menyelesaikan kasus atau pekerjaan khusus.

Contoh:
Statistik ➔ SPSS, Akuntansi ➔ DEC Easy Accounting, MS Word, MS Excel, dan lain-lain.

**Software Bahasa Pemrograman**
*Software* ini digunakan untuk membuat aplikasi oleh *programmer* yang dibuat khusus untuk menangani pekerjaan/aplikasi tertentu.

Contoh:
Visual FoxPro, Visual Basic, Delphi, Visual C, C+, dan lain-lain.

**Software Utility**
*Software* yang yang digunakan untuk membantu kinerja komputer, seperti antivirus, *scandisk*, dan lain-lain.

### 3.4.3 Aspek Brainware

Aspek ini berkaitan dengan sumber daya manusia yang menjalankan komputerisasi. Ketiga aspek di atas saling bekerja sama dan saling keterkaitan satu sama lainnya untuk melakukan fungsi-fungsi sistem pengolahan data (komputerisasi).

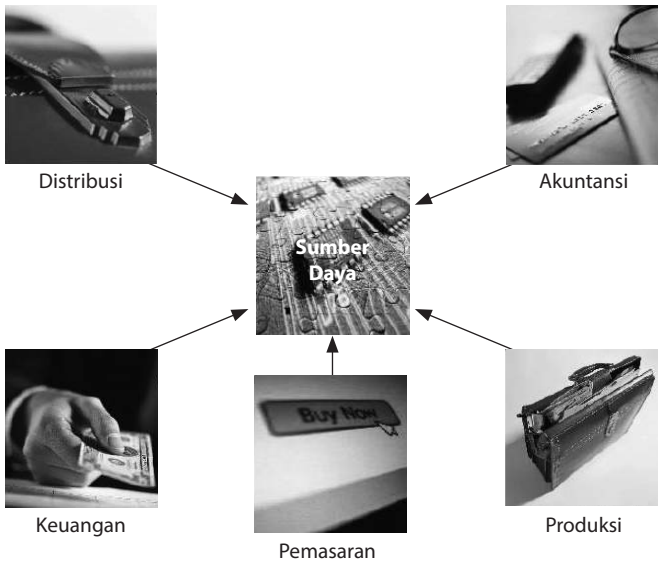
## 3.5 STRUKTURISASI FUNGSI TEKNOLOGI INFORMASI

Pengaturan teknologi informasi memiliki berbagai dampak pada sifat pengendalian internal, yang selanjutnya akan memiliki dampak pada auditnya. Pada bagian ini akan membahas mengenai risiko, pengendalian, dan isu audit pada dua model organisasi, yaitu pendekatan terpusat dan terdistribusi.

### 3.5.1 Pemrosesan Data Terpusat (Centralized Data Processing)

Model organisasi ini memungkinkan semua pemrosesan data dilakukan oleh satu atau lebih komputer besar yang ditempatkan pada satu lokasi terpusat yang melayani berbagai pengguna.

Pada Gambar 3.1 menggambarkan aktivitas layanan komputer yang dikonsolidasikan dan dikelola sebagai sumber daya bersama. Para pengguna akhir bersaing untuk mendapatkan sumber daya ini berdasarkan kebutuhannya. Fungsi layanan komputer biasanya diperlakukan sebagai pusat biaya (*cost center*) yang biaya operasionalnya akan dibebankan kembali kepada para pengguna akhir. Pada Gambar 2.2 berikut ini menggambarkan struktur layanan komputer terpusat dan menunjukkan area layanan utamanya, yaitu: (1) administrasi *database*, (2) pemrosesan data, (3) pengembangan sistem, dan (4) pemeliharaan.



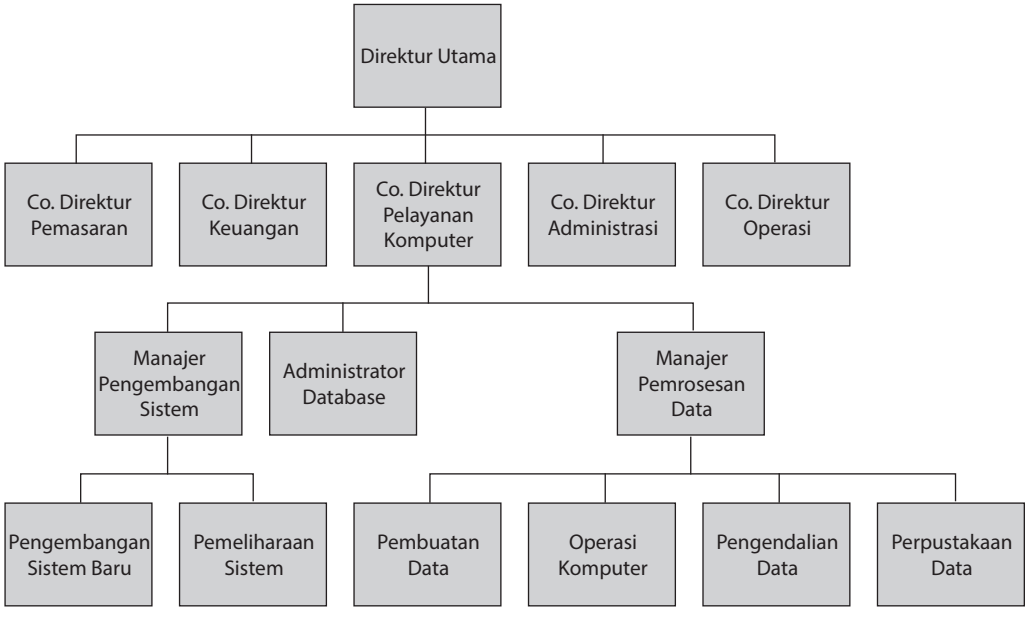

Gambar 3.1
Pemrosesan Data Terpusat

#### (1) Administrasi Database

Perusahaan yang dikelola secara terpusat dalam pemeliharaan sumber daya datanya dalam lokasi yang digunakan secara bersama oleh semua pengguna akhir. Area independen (administrasi *database*—DBA) ini dikepalai oleh Administrator Database yang bertanggung jawab atas keamanan dan integritas *database*.

#### (2) Pemrosesan Data

Bagian ini yang mengelola sumber daya komputer yang digunakan untuk melakukan pemrosesan harian berbagai transaksi. Bagian ini terdiri dari atas fungsi organisasi: pengendalian data, konversi data, operasi komputer, dan perpustakaan data.




Gambar 3.2
Bagan Struktur Organisasi dari Fungsi Layanan Komputer Terpusat

**(3) Pengembangan dan Pemeliharaan Sistem**

Kebutuhan sistem informasi para pengguna dipenuhi melalui fungsi ini. Fungsi pengembangan sistem bertanggung jawab menganalisis berbagai kebutuhan pengguna dan mendesain sistem baru. Pihak yang terlibat di dalam aktivitas pengembangan sistem adalah profesional sistem, pengguna akhir, dan pemegang kepentingan.

Ketika sistem baru telah didesain dan diimplementasikan, fungsi pemeliharaan akan meneruskan tanggung jawab untuk menjaganya tetap sesuai kebutuhan pengguna. Selama masa *running* sistem, sebanyak 80 sampai 90 persen dari biaya totalnya berasal dari aktivitas pemeliharaan.

## 3.5.2 Pemisahan Fungsi

Terdapat tiga tujuan dasar diadakannya pemisahan fungsi yang tidak saling bersesuaian, yaitu:

(1) Pemisahan fungsi otoritas transaksi dari pemrosesan data.
(2) Pemisahan fungsi pencatatan dan pengamanan aset.
(3) Pembagian fungsi pemrosesan transaksi ke beberapa personel agar pelaku penipuan akan melakukan kolusi dengan dua atau lebih personel.

Di lingkungan sistem berbasis komputer, sebuah aplikasi dapat melakukan otorisasi, memproses, dan mencatat semua aspek transaksi. Dengan demikian, fokus pengendalian dengan pemisahan fungsi pekerjaan bergeser dari tingkat operasional (pemrosesan data dilakukan oleh program komputer) ke hubungan organisasi yang lebih tinggi dalam fungsi layanan komputer.

## A. Memisahkan Pengembangan Sistem dari Operasi Komputer

Para staf pengembangan dan pemeliharaan sistem seharusnya menciptakan dan memelihara sistem bagi para pengguna, dan seharusnya tidak terlibat dalam proses memasukkan data atau menjalankan aplikasi. Staf operasional seharusnya menjalankan sistem dan tidak terlibat dalam tahap desain sistem.

Dengan pengetahuan mengenai logika dan parameter pengendalian aplikasi serta akses ke sistem operasi komputer dan perlengkapannya, seseorang yang memiliki hak akses istimewa dapat melakukan perubahan secara tidak sah atas aplikasi sistem. Misalnya, jauh lebih mudah bagi para pemogram untuk mengelabui sistem daripada operatornya karena mereka lebih tahu kode programnya.

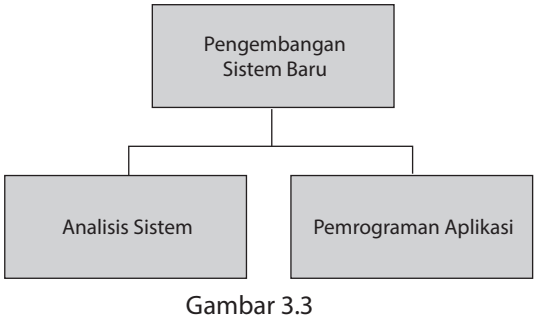
## B. Memisahkan Administrasi Database dari Fungsi Lainnya

Fungsi administrasi *database* (ADB) bertanggung jawab atas sejumlah pekerjaan yang berkaitan dengan keamanan *database*, pembuatan skema *database* dan tampilan pengguna, pemberian otoritas akses ke *database* kepada para pengguna, pengawasan penggunaan *database*, dan perencanaan perluasan di masa yang akan datang. Pendelegasian beberapa pekerjaan atau tanggung jawab ADB ke fungsi lainnya yang tidak saling berkesesuaian akan mengancam integritas *database*.

## C. Memisahkan Fungsi Pengembangan Sistem dengan Pemeliharaan Sistem

Beberapa perusahaan mengatur fungsi pengembangan sistem internalnya ke dalam dua bagian, yaitu analis sistem dan pemrograman. Pendekatan ini dihubungkan dengan dua jenis masalah pengendalian: *dokumentasi yang tidak memadai* dan *potensi terjadinya penipuan program*.

Struktur alternatif untuk fungsi pengembangan sistem dapat dilihat pada ilustrasi berikut ini, di mana fungsi tersebut dipisahkan ke dalam dua kelompok, yaitu: *analisis sistem* dan *pemrograman aplikasi*.




Gambar 3.3
Struktur Organisasi Alternatif untuk Pengembangan Sistem

## D. Memisahkan Fungsi Perpustakaan Data dan Operasional

Fungsi ini memiliki tanggung jawab atas penerimaan, penyimpanan, penarikan, dan pengamanan berbagai *file* data serta harus mengendalikan akses ke perpustakaan

tersebut. Pada masa sekarang ini, peran pustakawan tambahan meliputi penyimpanan cadangan data di lokasi kantor dan pengamanan perangkat lunak komersial beserta berbagai lisensinya.

### 3.5.3 Tujuan dan Prosedur Audit

***Tujuan***

Ada tiga tujuan audit dalam hal pemisahan fungsi di dalam lingkungan teknologi informasi, adalah sebagai berikut.

- Melakukan penilaian risiko mengenai pengembangan, pemeliharaan, dan operasi sistem.
- Memverifikasi bahwa para personel dengan pekerjaan yang tidak bersesuaian telah dipisah sesuai dengan tingkat potensi risikonya.
- Memverifikasi bahwa pemisahan tersebut dilakukan dalam cara yang dapat mendorong lingkungan kerja di mana hubungan formal, bukan informal, ada antarpekerjaan yang tidak saling bersesuaian tersebut.

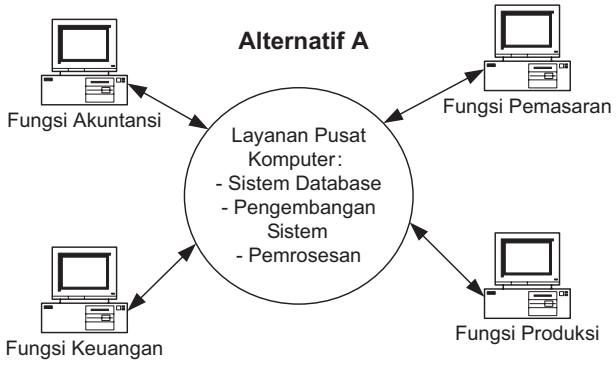
***Prosedur Audit***

- Memperoleh dan mengkaji kebijakan perusahaan atas keamanan komputer. Memverifikasi bahwa kebijakan dikomunikasikan kepada para karyawan dan supervisor yang bertanggung jawab atasnya.
- Mengkaji dokumentasi yang terkait, termasuk struktur organisasi saat ini, pernyataan misi, dan deskripsi tugas untuk berbagai fungsi penting, agar dapat menentukan apakah ada orang atau kelompok yang menjalankan fungsi-fungsi yang saling tidak bersesuaian.
- Mengkaji dokumentasi sistem dan catatan pemeliharaan untuk mencari sampel aplikasi. Memverifikasi bahwa para pemogram pemeliharaan yang ditugaskan untuk proyek tertentu bukan merupakan pemogram desain awalnya.
- Melalui observasi, menentukan apakah kebijakan pemisahan tugas diikuti dalam praktiknya. Mengkaji daftar akses ruang operasi untuk menentapkan apakah pemogram masuk ke fasilitas tersebut untuk alasan lain selain dari alasan kegagalan sistem.
- Mengkaji hak-hak dan keistimewaan para pengguna untuk memverifikasi bahwa para pemogram memiliki izin akses yang sesuai dengan deskripsi tugasnya.

### 3.5.4 Model Terdistribusi

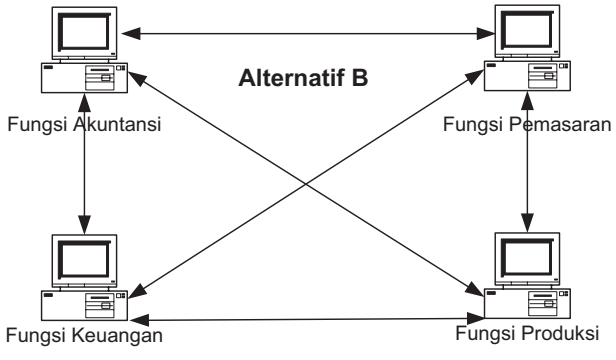
Alternatif dari konsep terpusat adalah konsep pemrosesan data terdistribusi (*distributed data processing*—DDP). DDP melibatkan reorganisasi fungsi layanan komputer menjadi beberapa unit TI kecil yang diletakkan di bawah kendali pengguna akhir. Unit tersebut didistribusikan berdasarkan fungsi bisnisnya, lokasi geografis, atau keduanya. Tingkat distribusi aktivitas TI akan berbeda tergantung pada filosofi dan tujuan manajemen perusahaan.

Ilustrasi berikut ini menunjukkan dua alternatif DDP, yaitu:
***Alternatif A*** adalah varian dari model terpusat; perbedaannya adalah terminal-terminalnya didistribusikan ke para pengguna akhir untuk menangani *input* dan *output*. Bentuk ini meniadakan kebutuhan akan kelompok pengendalian data dan konversi data terpusat, karena para pengguna akhir akan mengerjakan pekerjaan tersebut. Namun demikian, pengembangan sistem, operasional komputer, dan administrasi *database* masih terpusat.




Gambar 3.4
Model Terdistribusi Varian dari Model Terpusat

***Alternatif B*** merupakan perubahan radikal dari model terpusat. Alternatif ini mendistribusikan semua layanan komputer ke semua pengguna akhir, di mana mereka dapat beroperasi sebagai unit yang berdiri terpisah. Hasilnya adalah menghilangkan fungsi layanan komputer pusat dalam struktur organisasi. Hubungan antarfungsi mewakili penataan dalam bentuk jaringan yang memungkinkan komunikasi serta transfer data antarunit atau fungsi.




Gambar 3.5
Model Terdistribusi Perubahan Radikal dari Model Terpusat

***Arsitektur Jaringan Client-Server—C/S*** memiliki beberapa fitur dari kedua alternatif di atas. Arsitektur ini menyediakan sentralisasi *database* atau *software* ke server, dengan tetap menyediakan fungsional dan independensi komputer yang berdiri sendiri.

## A. Risiko Berkaitan dengan DDP

Potensi masalah yang terdapat pada lingkungan DDP adalah ketidakefisienan penggunaan sumber daya, kerusakan jejak audit, pemisahan fungsi tugas yang tidak saling bersesuaian, peningkatan potensi kesalahan pemrograman, dan kegagalan sistem, serta kurangnya standar.

***Ketidakefisienan Penggunaan Sumber Daya***

- Risiko terjadinya kesalahan manajemen atas sumber daya keseluruhan perusahaan, terutama oleh pengguna akhir.
- Risiko perangkat keras dan lunak tidak sesuai satu sama lainnya (tidak kompatibel) di tingkat pengguna akhir.
- Risiko pekerjaan yang *redudance* (rangkap/ganda) berkaitan dengan aktivitas dan tanggung jawab.

***Kerusakan Jejak Audit***

Jejak audit pada masa sekarang ini, cenderung bersifat elektronik sehingga sebagian atau seluruh jejak audit berada di dalam berbagai komputer pengguna akhir. Contoh risiko kerusakan jejak audit:

- Pengguna akhir secara tidak sengaja menghapus jejak audit, sehingga jejak audit tersebut dapat hilang dan sulit untuk dipulihkan kembali.
- Pengguna akhir secara tidak sengaja meng-*input* suatu kesalahan yang lolos dari pengendalian ke dalam catatan (*log*) audit, sehingga jejak audit dapat secara efektif akan hancur.

***Pemisahan Fungsi Tidak Memadai***

Dalam satu unit orang yang sama dapat menulis program aplikasi, melakukan pemeliharaan program, memasukkan data transaksi ke dalam komputer, dan mengoperasikan perlengkapan komputer (periferal). Kondisi ini akan menjadi pelanggaran pengendalian internal yang mendasar.

## B. Kelebihan dari DDP

— ***Penurunan Biaya***, banyaknya jenis kebutuhan yang harus dipenuhi oleh sistem yang terpusat membutuhkan komputer umum dan penggunaan operasional yang rumit. Beban *overhead* yang timbul akibat dari kegiatan yang menjalankan sistem tersebut dapat menurunkan kelebihan kekuatan pemrosesan aslinya. Jadi, sistem besar yang terpusat mewakili biaya yang sangat mahal yang harus dihindari. Penerapan DDP dapat mengurangi biaya dalam dua hal, yaitu (1) data dapat di-*input* dan diedit pada area pengguna, hingga menghilangkan pekerjaan terpusat untuk pembuatan dan pengendalian data; dan (2) kerumitan aplikasi dapat dikurangi, yang akhirnya akan mengurangi biaya pengembangan dan pemeliharaan.

— ***Peningkatan Tanggung Jawab Pengendalian Biaya***, para manajer diberdayakan secara tepat dengan otoritas untuk membuat keputusan mengenai sumber daya yang memengaruhi keberhasilan mereka.

— ***Peningkatan Kepuasan Pengguna***, hal ini berasal dari tiga area kebutuhan yang sering kali tidak dipenuhi dalam pendekatan model terpusat, yaitu (1) para pengguna menginginkan pengendalian sumber daya yang memengaruhi tingkat keuntungan mereka; (2) para pengguna menginginkan para profesional sistem yang responsif terhadap situasi khusus mereka; dan (3) para pengguna ingin lebih secara aktif dilibatkan dalam pengembangan dan implementasi sistem.
— ***Fleksibiltas Cadangan***, kemampuan untuk membuat cadangan fasilitas komputer agar terlindungi dari potensi bencana alam.

### C. Tujuan dan Prosedur Audit

**Tujuan Audit**

- Melakukan penilaian risiko atas fungsi TI pada DDP.
- Memverifikasi bahwa unit TI yang terdistribusi menggunakan berbagai standar kinerja keseluruhan perusahaan yang mendorong kesesuaian antara perangkat keras, aplikasi perangkat lunak, dan data.

**Prosedur Audit**

- Memverifikasi bahwa berbagai kebijakan dan standar perusahaan untuk desain sistem, dokumentasi, dan pengadaan perangkat keras dan lunak telah dikeluarkan dan disebarluaskan ke berbagai unit TI.
- Mengkaji struktur organisasi, misi, dan deskripsi tugas terkini berbagai fungsi yang utama, untuk menentukan apakah ada karyawan yang melakukan tugas yang tidak saling berkesesuaian.
- Memverifikasi bahwa ada pengendalian pengganti seperti supervisi dan pengawasan manajemen dilakukan ketika pemisahan tugas yang tidak saling bersesuaian secara ekonomi tidak mungkin dilakukan.
- Mengkaji dokumentasi sistem untuk memverifikasi bahwa berbagai aplikasi, prosedur dan *database* didesain dan berfungsi sesuai dengan standar perusahaan.
- Memverifikasi bahwa setiap karyawan diberikan izin akses ke berbagai program dan data sesuai dengan deskripsi tugas.

## 3.6 PENGENDALIAN PUSAT KOMPUTER

Para akuntan mempelajari lingkungan fisik pusat komputer sebagai bagian dari audit tahunan. Eksposur pada area ini memiliki potensi dampak yang besar atas informasi, catatan akuntansi, pemrosesan transaksi, dan efektivitas berbagai pengendalian lainnya yang lebih konvensional dan internal.

Berikut ini adalah beberapa pengendalian yang secara langsung dapat berkontribusi pada keamanan lingkungan pusat komputer.

1. ***Lokasi fisik pusat komputer***, sedapat mungkin harus jauh dari berbagai bahaya dan ancaman yang ditimbulkan oleh manusia maupun alam.

2. ***Konstruksi pusat komputer***, idealnya ditempatkan dalam bangunan berlantai satu yang konstruksinya solid dengan akses terkendali. Akan tetapi, jika berada di gedung beberapa lantai, maka idealnya ditempatkan pada lantai paling atas.
3. ***Akses pusat komputer***, seharusnya dibatasi hanya untuk para operator dan karyawan lainnya yang bekerja pada perusahaan tersebut.
4. ***Pengaturan suhu udara***, komputer akan berfungsi dengan baik apabila lingkungan memiliki pengatur suhu udara. Kesalahan logika dapat terjadi dalam perangkat keras komputer jika suhu menyimpang jauh dari kisaran optimal. Risiko kerusakan sirkuit akibat gelombang listrik statis akan meningkat apabila terjadi kelembaban udara yang tinggi.
5. ***Pemadam kebakaran***, ancaman yang paling umum atas perangkat komputer perusahaan adalah kebakaran. Hampir setengah dari perusahaan yang mengalami kebakaran bangkrut karena hilangnya berbagai catatan yang sangat penting, seperti piutang usaha.
6. ***Pasokan listrik,*** masalah sumber daya ini sering memberikan masalah yang dapat mengganggu operasi pusat komputer. Diperlukan perlengkapan yang digunakan untuk mengendalikan berbagai masalah mengenai pasokan listrik.

### 3.6.1 Tujuan dan Prosedur Audit

**Tujuan Audit**

Tujuan umum untuk mengevaluasi berbagai pengendalian yang mengatur keamanan pusat komputer, sedangkan tujuan khusus, auditor harus memverifikasi bahwa:

- Pengendalian keamanan fisik memadai secara wajar untuk melindungi perusahaan dari eksposur fisik.
- Jaminan atas perlengkapan telah memadai untuk dapat memberikan kompensasi pada perusahaan jika terjadi kehancuran atau kerusakan atas pusat komputer.
- Dokumentasi operator memadai untuk dapat menangani kegagalan sistem.

**Prosedur Audit**

- Pengujian konstruksi fisik.
- Pengujian sistem deteksi kebakaran.
- Pengujian pengendalian akses.
- Pengujian pasokan listrik cadangan.
- Pengujian cakupan asuransi.
- Pengujian pengendalian dokumentasi operator.

### 3.6.2 Perencanaan Pemulihan Bencana Alam

Pada ilustrasi berikut ini menunjukan tiga jenis peristiwa yang dapat mengganggu atau menghancurkan pusat komputer perusahaan dan sistem informasinya. Ketiga peristiwa tersebut adalah bencana alam, bencana akibat manusia, dan kegagalan sistem.

Akibat bencana alam biasanya sangat menghancurkan bagi pusat komputer dan sistem informasi, walaupun kemungkinan kejadian tersebut tidak besar. Terkadang peristiwa bencana alam tidak dapat dicegah dan dihindari. Misalnya, seperti tsunami di Aceh, gempa bumi di Yogyakarta, kebakaran ruang pusat komputer di gedung kantor Pertamina Jakarta, dan pengeboman di gedung Bursa Efek Indonesia.

Daya bertahan hidup perusahaan yang terpengaruh oleh berbagai bencana, tergantung pada seberapa baiknya dan seberapa cepatnya perusahaan bereaksi. Dengan perencanaan kontinjensi yang hati-hati, dampak penuh dari bencana dapat dikurangi dan perusahaan masih dapat pulih kembali. Bencana yang diakibatkan oleh manusia, seperti sabotase, atau kelalaian dapat sama menghancurkannya.

Untuk dapat bertahan hidup dari peristiwa semacam itu, perusahaan mengembangkan prosedur pemulihan dan merumuskannya dalam bentuk rencana keberlanjutan perusahaan, rencana pemulihan bencana, atau rencana pemulihan bisnis. Rencana pemulihan bencana (*disaster recovery plan*—DRP) adalah pernyataan yang komprehensif tentang semua tindakan yang harus dilakukan sebelum, selama, dan setelah adanya bencana jenis apa pun, bersama dengan berbagai prosedur yang didokumentasikan dan diuji akan memastikan keberlanjutan operasi perusahaan. Semua rencana yang dapat berhasil dengan baik memiliki tiga fitur, yaitu:

1. Mengidentifikasi aplikasi yang sangat penting.
2. Membentuk tim pemulihan bencana.
3. Menyediakan cadangan lokasi.

## 3.7. PENGENDALIAN SISTEM OPERASI DAN PENGENDALIAN KESELURUHAN

Sistem operasi (*operating system*) adalah program yang mengendalikan jalannya atau operasi komputer. Sistem ini memungkinkan para pengguna dan aplikasi dapat berbagi dan mengakses sumber daya komputer secara bersamaan. Akuntan yang modern perlu mengenali peran sistem operasi dalam gambaran umum pengendalian untuk dapat secara tepat menilai berbagai risiko yang mengancam sistem akuntansi. Jika integritas sistem operasi menjadi lemah, pengendalian dalam setiap aplikasi akuntansi mungkin akan lemah pula. Dengan semakin banyaknya sumber daya komputer yang digunakan dalam komunitas pengguna yang selalu bertambah, keamanan sistem operasi menjadi isu pengendalian yang penting.

Sistem operasi melakukan tiga hal pekerjaan utama, yaitu (1) sebagai penerjemah bahasa manusia dengan bahasa mesin (komputer), (2) berperan dalam mengalokasikan berbagai sumber daya komputer ke pengguna, kelompok kerja, dan aplikasi, dan (3) bertugas mengelola berbagai pekerjaan penjadwalan tugas dan multipemrograman.

Untuk melakukan pekerjaan ini secara konsisten dan andal, maka sistem operasi harus mencapai lima tujuan pengendalian fundamental berikut ini.

1. Sistem operasi harus bisa melindungi dirinya dari para pengguna.
2. Sistem operasi harus bisa melindungi para penggunanya dari satu pengguna ke pengguna lainnya.

3. Sistem operasi harus bisa melindungi para penguna dari dirinya sendiri.
4. Sistem operasi harus bisa dilindungi dari dirinya sendiri.
5. Sistem operasi harus bisa dilindungi dari lingkungan sekitar.

### 3.7.1 Keamanan Sistem Operasi

Keamanan sistem operasi (*operating system security*) melibatkan kebijakan, prosedur, dan pengendalian yang menentukan siapa saja yang dapat mengakses sistem operasi, sumber daya mana yang dapat diakses, dan tindakan apa yang dapat dilakukan. Berikut ini adalah komponen keamanan yang dapat diterapkan dalam sistem operasi.

1. Prosedur Logon
2. Access Token
3. Daftar Pengendalian Akses
4. Pengendalian Akses Mandiri

### 3.7.2 Ancaman terhadap Integritas Sistem Operasi

Tujuan pengendalian sistem operasi terkadang tidak dapat tercapai karena adanya berbagai kesalahan dalam sistem operasi yang terjadi secara tidak disengaja atau disengaja. Ancaman yang tidak disengaja meliputi kegagalan perangkat keras yang menyebabkan sistem operasi gagal (*crash*). Kegagalan sistem operasi juga dapat disebabkan oleh penggunaan program aplikasi pengguna yang tidak dapat diterjemahkan oleh sistem operasi.

Ancaman yang dilakukan dengan disengaja biasanya berupa usaha untuk dapat mengakses data secara ilegal atau melanggar privasi pengguna, untuk mendapatkan keuntungan finansial. Akan tetapi, pada masa sekarang ini, bentuk ancaman yang berasal dari program penghancur yang tidak jelas tujuan maupun keuntungan yang akan dicapai. Berbagai eksposur ini berasal dari tiga sumber, yaitu:

1. Personel dengan hak tertentu yang menyalahgunakan wewenangnya.
2. Orang-orang yang menjelajahi sistem operasi untuk mengidentifikasi dan mengeksploitasi kelemahan keamanan.
3. Orang yang menyelipkan virus komputer atau bentuk lain program perusak lainnya ke dalam sistem operasi.

### 3.7.3 Pengendalian Keseluruhan Sistem

Hak akses pengguna diberikan pada beberapa orang dan ke seluruh kelompok kerja yang diotorisasi untuk menggunakan sistem. Hak tersebut untuk menentukan sumber daya apa saja yang dapat diakses atau digunakan oleh seseorang atau kelompok, dan operasi apa saja yang diperbolehkan.

Keamanan keseluruhan sistem dipengaruhi oleh bagaimana hak akses diberikan. Hak semacam ini harus dikelola dengan hati-hati dan diawasi secara dekat agar sesuai dengan kebijakan perusahaan dan prinsip pengendalian internal.

**Tujuan Audit**

Memverifikasi bahwa hak akses diberikan dalam cara yang konsisten dengan kebutuhan untuk memisahkan berbagai fungsi yang tidak saling bersesuaian, dan sesuai dengan kebijakan perusahaan.

**Prosedur Audit**

- Mengkaji kebijakan perusahaan atas berbagai fungsi yang tidak saling bersesuaian dan memastikan bahwa kebijakan tersebut mendorong adanya keamanan yang wajar.
- Mengkaji berbagai hak kelompok pengguna dan orang-orang tertentu untuk menentukan apakah hak akses mereka sesuai dengan deskripsi tugas dan posisinya.
- Mengkaji catatan personalia untuk menentukan apakah para karyawan yang diberikan hak tersebut menjalani pemeriksaan keamanan intensif yang cukup atau tidak, sesuai dengan kebijakan perusahaan.
- Mengkaji catatan karyawan untuk menentukan apakah para pengguna telah secara formal mengetahui tanggung jawabnya untuk menjaga kerahasiaan data perusahaan.
- Mengkaji waktu *logon* yang diizinkan untuk para penggguna.

### 3.7.4 Pengendalian Password

*Password* atau kata sandi adalah kode rahasia yang di-*input* oleh pengguna untuk mendapatkan akses ke sistem, aplikasi, *file* data, atau server jaringan. Jika pengguna tidak dapat memberikan kata sandi yang benar, maka sistem operasi akan menolak untuk memberikan hak akses.

Meskipun kata sandi dapat memberikan keamanan pada tingkat tertentu ketika diterapkan pada pengguna yang tidak memiliki konsep keamanan, prosedur kata sandi dapat mengakibatkan perilaku pengguna yang melemahkan keamanan. Bentuk paling umum perilaku yang bertentangan dengan keamanan meliputi:

- Lupa kata sandi dan akhirnya dikeluarkan dari sistem.
- Tidak sering mengubah kata sandi.
- Sindrom *post-it*, di mana kata sandi ditulis pada kertas kecil dan dipajang sehingga dapat dilihat orang.
- Kata sandi yang terlalu sederhana sehingga mudah ditebak oleh pelaku kejahatan komputer.

Pihak manajemen eksekutif dan manajemen SI harus membentuk kebijakan kata sandi yang dapat mengatasi berbagai risiko dan menyediakan potensi pengendalian. Adapun kebijakan yang disarankan (T. Singleton, *Managing the Audit Function*) adalah sebagai berikut.

- *Penyebaran yang tepat*, menyebarluaskannya, menggunakannya dalam pelatihan atau orientasi karyawan, dan mencari berbagai cara untuk meningkatkan kesadaran dalam perusahaan.
- *Panjang kata sandi yang sesuai*, menggunakan paling tidak delapan karakter. Makin banyak karakternya, makin sulit untuk ditebak atau dipecahkan. Delapan karakter adalah panjang yang efektif untuk mencegah proses penebakan jika digabungkan dengan hal di bawah ini.
- *Kekuatan yang sesuai*, menggunakan alfabet, angka, dan berbagai karakter khusus. Makin banyak karakter nonhuruf, makin sulit untuk ditebak atau dipecahkan. Buatlah kata sandi *case sensitive* dan menggabungkan huruf kapital dan nonkapital.
- *Tingkat akses atau kompleksitas yang sesuai*, menggunakan beberapa level akses yang membutuhkan beberapa kata sandi. Dengan menggunakan matriks kata sandi data untuk memberikan hak *read only*, *write/ read*, atau tidak memberikan akses untuk setiap *field* data per *user*, menggunakan teknologi biometrik (sidik jari, retina mata, suara). Selain itu, menggunakan alat akses tambahan seperti *smart card*, atau kata sandi *beeper* untuk *login* jarak jauh dan prosedur yang ditetapkan berdasarkan pengguna.
- *Perubahan tepat waktu yang sesuai*, dalam rentang waktu yang teratur, memerintahkan karyawan untuk mengubah kata sandi mereka.
- *Perlindungan yang sesuai*, melarang praktik untuk berbagi kata sandi atau adanya kata sandi dituliskan pada kertas dan diletakan di komputer salah seorang karyawan.
- *Penghapusan yang tepat*, memerintahkan penghapusan segera akun karyawan yang sudah berhenti, untuk mencegah seorang karyawan yang kecewa melakukan aktivitas negatif.

**Tujuan Audit**

Memastikan bahwa perusahaan memiliki kebijakan kata sandi yang memadai dan efektif untuk mengendalikan akses ke sistem operasi.

**Prosedur Audit**

- Memverifikasi bahwa semua pengguna diharuskan memiliki kata sandi.
- Memverifikasi bahwa semua pengguna diberikan arahan dalam penggunaan kata sandi mereka dan peran penting pengendalian kata sandi.
- Menentukan apakah telah ada prosedur untuk mengidentifikasi berbagai kata sandi yang gampang dipecahkan. Proses ini melibatkan penggunaan perangkat lunak untuk melakukan *scan file* kata sandi secara teratur.
- Menilai kecukupan standar kata sandi seperti dalam hal panjangnya dan interval kedaluwarsanya.
- Meninjau kembali kebijakan dan prosedur penguncian akun dan membutuhkan aktivasi kembali.

## SOAL DAN STUDI KASUS

1. Jelaskan apa yang dimaksud dengan sistem informasi?
2. Sebutkan tiga peranan sistem informasi dalam bisnis! Berikan contoh dari masing-masing peranan sistem informasi tersebut!
2. Sebutkan dua jenis sistem informasi yang diimplementasikan dalam dunia bisnis!
3. Sebutkan tiga aspek yang membentuk sistem komputerisasi! Jelaskan hubungan di antara ketiga aspek tersebut!
4. Apa yang dimaksud dengan pemrosesan data terdistribusi?
5. Apa kelebihan dan kelemahan pemrosesan data terdistribusi?
6. Apa jenis pekerjaan yang menjadi redundan dalam sistem pemrosesan data terdistribusi?
7. Apa saja lima tujuan umum dari sistem operasi?
8. Apa jenis *output* yang dianggap sangat sensitif dalam lingkungan universitas? Berikan tiga contoh dan jelaskan mengapa informasi tersebut dianggap sensitif! Diskusikan siapa yang seharusnya dan tidak seharusnya memiliki akses ke tiap-tiap jenis informasi tersebut!

# BAB 4

# SISTEM MANAJEMEN DATABASE

Setelah mempelajari bab ini, Anda diharapkan mampu:

- Memahami masalah operasional yang ada dalam pendekatan *file* datar (*flat file*) terhadap manajemen data sehingga menimbulkan pendekatan *database*.
- Memahami hubungan antarkomponen fundamental dari konsep *database*.
- Mengetahui karakteristik yang menentukan dari tiga model *database*: hierarkis, jaringan, dan relasional.
- Memahami fitur operasional yang berkaitan dengan risiko penyebaran model *database* terpusat, terpartisi, dan tereplikasi dalam lingkungan DDP.
- Mengetahui tujuan dan prosedur audit yang digunakan untuk menguji pengendalian manajemen data.

Bab ini akan membahas audit terhadap sistem yang mengelola dan mengendalikan sumber daya data perusahaan. Manajemen *database* bisa dibagi menjadi dua pendekatan, yaitu (1) model *flat file* dan (2) model *database*. Pembahasan awal mengenai konsep manajemen *flat file* yang digunakan dalam berbagai sistem yang lama hingga saat ini. Selain pembahasan konsep, terdapat ilustrasi bagaimana masalah yang berkaitan dengan *flat file* diselesaikan.

Bagian kedua dari bab ini akan menjelaskan fungsi utama dan fitur penentu dari tiga model *database* yang umum digunakan: model hierarkis, jaringan, dan relasional. Ketiga model tersebut ditampilkan dari perspektif fungsi TI terpusat.

Bagian ketiga membahas mengenai peranan teknologi *database* dalam lingkungan terdistribusi. Pemrosesan *database* terdistribusi (DDP) memungkinkan pengguna akhir untuk memperoleh kepemilikan dan pengendalian atas sumber daya teknologi informasi, termasuk *database*. Bagian ini menyajikan teknik untuk mencapai tujuan DDP sambil mempertahankan prinsip pembagian dan integrasi data.

## 4.1 PENDEKATAN MANAJEMEN DATA

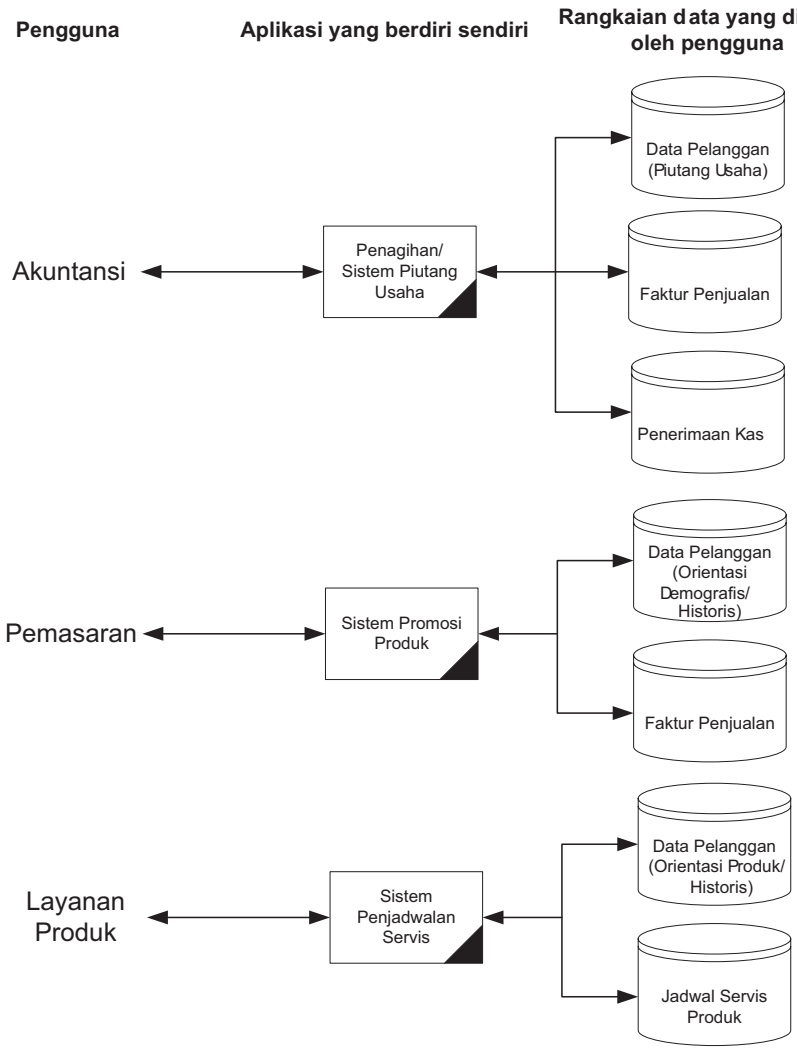
Ada dua pendekatan umum terhadap manajemen data, yaitu: model *flat file* dan model *database*.

### 4.1.1 Pendekatan Model Flat File

Pendekatan model ini merupakan pendekatan warisan masa lalu yang merupakan sistem *mainframe* besar. Pendekatan model *flat file* diimplementasikan pada akhir tahun 1960-an hingga tahun 1980-an. Saat ini sistem tersebut telah tergantikan dengan pendekatan model *database* yang lebih modern.

Model *flat file* menggambarkan lingkungan di mana *file* data individual tidak berhubungan dengan *file* lainnya. Pengguna akhir dalam lingkungan ini memiliki *file* data dan tidak berbagi dengan pengguna lainnya. Dengan demikian, pemrosesan data dilakukan oleh aplikasi yang berdiri sendiri bukan oleh sistem yang terintegrasi. Ketika beberapa pengguna membutuhkan data yang sama untuk tujuan berbeda, maka harus mengambil rangkaian data yang terpisah dan terstruktur sesuai dengan kebutuhan khusus.

Ilustrasi berikut (Gambar 4.1) menggambarkan bagaimana data penjualan pelanggan disajikan untuk tiga pengguna yang berbeda dalam suatu perusahaan. Redudansi data yang terdapat pada ilustrasi tersebut menyebabkan tiga masalah yang signifikan dalam lingkungan *flat file*, yaitu: **penyimpanan data, pembaruan data, dan kekinian informasi.**




Gambar 4.1
Redudansi pada Data Pelanggan

## A. Penyimpanan Data

Sistem informasi yang efisien mengambil dan menyimpan data hanya satu kali dan membuat sumber tunggal ini tersedia bagi semua pengguna yang membutuhkannya. Dalam lingkungan *flat file*, hal ini tidak mungkin dilakukan.

## B. Pembaruan Data

Perusahaan menyimpan sejumlah besar data ke dalam *file master* dan *file* referensi yang memerlukan *update* secara berkala untuk mencerminkan perubahan. Pada lingkungan *flat file*, para pengguna menyimpan *file* yang terpisah, semua perubahan harus dibuat secara terpisah juga untuk masing-masing pengguna.

### C. Update Data

Jika informasi pembaruan tidak disebarkan secara tepat, maka perubahan tersebut tidak akan merefleksikan ke beberapa data pengguna, sehingga keputusan yang diambil akan didasarkan pada informasi yang lama.

### D. Ketergantungan Task Data

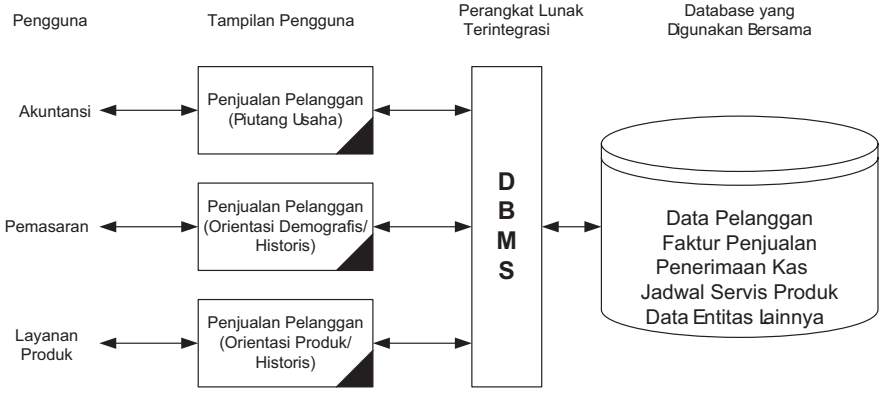
Ketidakmampuan pengguna untuk memperoleh informasi tambahan ketika kebutuhannya berubah. Rangkaian informasi pengguna dibatasi oleh data yang dimiliki dan dikendalikan. Para pengguna bertindak secara terpisah; mereka tidak berinteraksi sebagai sesama anggota dari suatu masyarakat pengguna.

### E. Flat File Membatasi Integrasi Data (Inklusi Terbatas)

*File* distrukturisasi, diformat, dan disusun agar sesuai dengan kebutuhan khusus dari pemilik atau pengguna utama dari data tersebut. Strukturisasi semacam ini dapat membatasi atribut data di dalam organisasi.

## 4.1.2 Pendekatan Database

Permasalahan yang ada pada *flat file* diatasi dengan mengimplementasikan pendekatan *database* terhadap manajemen data. Akses ke sumber daya data dikendalikan oleh sistem manajemen *database* (*database management system*—DBMS). DBMS adalah sistem perangkat lunak khusus yang diprogram untuk mengetahui elemen manajemen data yang boleh diakses oleh masing-masing pengguna. Pendekatan ini memusatkan data perusahaan dalam satu *database* umum yang dibagikan dengan pengguna lainnya. Dengan penggunaan secara bersamaan, masalah tradisional pada pendekatan sebelumnya mungkin dapat diatasi dengan pendekatan *database*.




Gambar 4.2
Pengelolaan Data dengan Pendekatan DBMS

### A. Eliminasi Permasalahan Penyimpanan Data

Setiap elemen data disimpan hanya satu kali, sehingga mengurangi redudansi data (data kembar) serta mengurangi biaya pengumpulan dan penyimpanan data.

### B. Eliminasi Permasalahan Pembaruan Data

Dikarenakan setiap elemen data hanya muncul dalam satu lokasi, maka prosedur pembaruan hanya perlu dilakukan satu kali. Hal ini mengurangi waktu dan biaya untuk menjaga *update* data.

### C. Eliminasi Permasalahan Update Data

Satu perubahan pada atribut data akan secara otomatis tersedia bagi semua pengguna dari atribut tersebut.

### D. Eliminasi Permasalahan Ketergantungan Task Data

Dengan akses penuh ke domain data entitas, perubahan pada kebutuhan informasi pengguna dapat dipenuhi tanpa harus mengambil serangkaian data tambahan khusus. Para pengguna hanya dibatasi oleh ketersediaan data untuk entitas tersebut dan legitimasi dari kebutuhan mereka untuk mengaksesnya.

### E. Eliminasi Permasalahan Integrasi Data

Oleh karena data berada dalam lokasi yang dapat diakses secara umum dan global, data tersebut dapat diintegrasikan secara penuh ke semua aplikasi untuk semua pengguna.

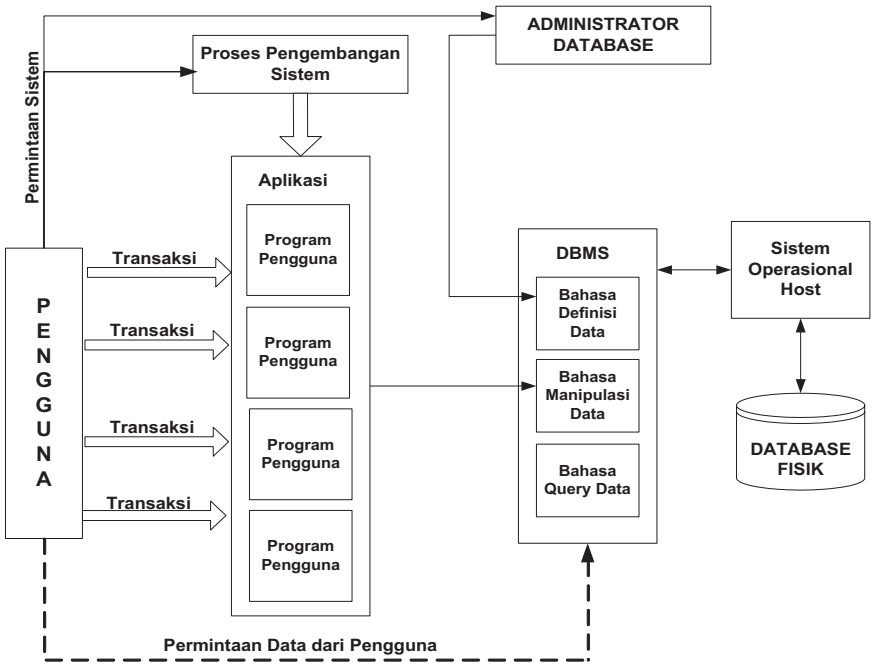
## 4.2 SISTEM DATABASE TERPUSAT

Pada ilustrasi berikut ini (Gambar 4.3) menyajikan pembagian lingkungan *database* menjadi empat elemen utama, yaitu: (1) DBMS, (2) Pengguna, (3) Administrator *database*, dan (4) *Database* fisik.

### 4.2.1 Sistem Manajemen Database

DBMS menyediakan lingkungan yang terkendali untuk membantu (atau mencegah) akses ke *database* dan untuk mengelola sumber daya data secara efisien. Setiap DBMS memiliki keunikan masing-masing dalam memenuhi tujuan ini, tetapi fitur yang umum adalah sebagai berikut.

- *Pengembangan program*. DBMS berisi perangkat lunak pengembangan aplikasi. Pemogram dan pengguna akhir dapat menggunakan fitur ini untuk menciptakan aplikasi untuk mengakses *database*.
- *Pembuatan backup dan pemulihan*. Selama pemrosesan, DBMS secara periodik membuat salinan *backup* dari *database* fisik. Jika terjadi bencana yang dapat mengakibatkan *database* tidak dapat digunakan, DBMS dapat memulihkan kembali ke versi sebelumnya yang dianggap benar.




Gambar 4.3
Metode Akses terhadap Database Terpusat

- *Pelaporan penggunaan database*. Fitur ini menghasilkan data statistik mengenai data apa saja yang digunakan, kapan digunakan, dan siapa yang menggunakannya. Informasi ini digunakan oleh administrator *database* dalam menetapkan otorisasi pengguna dan memelihara *database*.
- *Akses database*. Fitur yang paling penting dari DBMS adalah memungkinkan pengguna yang memiliki otorisasi untuk mengakses *database* secara formal dan informal. Pada ilustrasi di atas (Gambar 4.3) menunjukkan tiga modul perangkat lunak yang memfasilitasi tugas ini, yaitu bahasa definisi data, bahasa manipulasi data, dan bahasa *query* data.

### 4.2.2 Pengguna

Pada ilustrasi Gambar 4.3 menunjukkan bagaimana pengguna mengakses *database* dengan dua cara. Pertama, akses dimungkinkan oleh antarmuka (*interface*) aplikasi formal. Program pengguna, yang disiapkan oleh *programmer*, mengirimkan permintaan akses data DBMS, yang kemudian memvalidasi permintaan tersebut dan menelusuri data untuk diproses.

Kedua, akses informal: bahasa permintaan data, yaitu metode permintaan data secara informal. Permintaan data (*query*) adalah metodologi akses yang menggunakan perintah yang mirip dengan bahasa Inggris untuk membangun daftar atau informasi dasar lainnya dari *database*. Para pengguna dapat mengakses data melalui permintaan langsung, yang tidak memerlukan program pengguna formal.

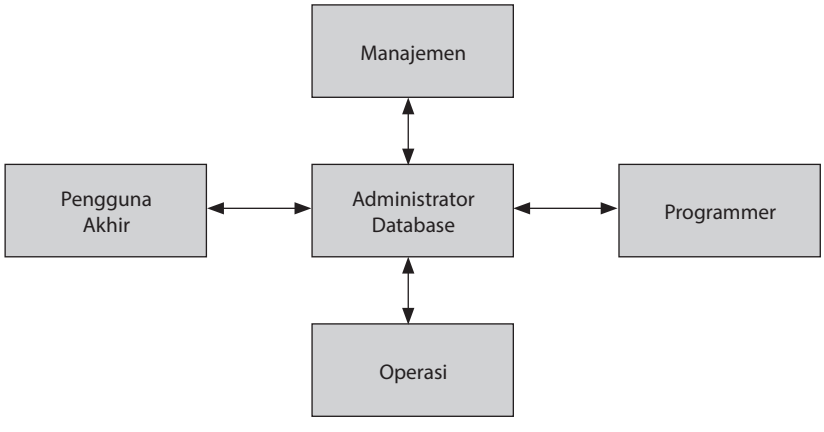
### 4.2.3 Administrator Database (DBA)

DBA bertanggung jawab untuk mengelola sumber daya *database*. Untuk melakukan *share database* yang sama antara banyak pengguna, perlu adanya pengaturan, koordinasi, peraturan, dan petunjuk untuk melindungi integritas *database*. Tugas DBA adalah meliputi bidang-bidang: (1) perencanaan *database*; (2) desain *database*; (3) implementasi, operasi, dan pemeliharaan *database*; serta (4) perubahan dan pengembangan *database*.

Tabel 4.1
Fungsi Administrator Database

| Perencanaan Database: | Implementasi: |
|---|---|
| Mengembangkan strategi *database* perusahaan<br>Mendefinisikan lingkungan *database*<br>Mendefinisikan persyaratan *database*<br>Mengembangkan kamus data | Menentukan kebijakan akses<br>Mengimplementasikan pengendalian keamanan<br>Menentukan prosedur pengujian<br>Menetapkan standar pemrograman |
| Desain: | Operasi dan Pemeliharaan: |
| *Database* logis (skema)<br>Tampilan pengguna eksternal (subskema)<br>Tampilan internal *database*<br>Pengendalian *database* | Mengevaluasi kinerja *database*<br>Mengatur kembali *database* sesuai dengan permintaan kebutuhan pengguna<br>Meninjau kembali standar dan prosedur |
| | Perubahan dan Pertumbuhan: |
| | Merencanakan perubahan dan pertumbuhan<br>Mengevaluasi teknologi baru |

Fungsi penting lainnya dari DBA adalah pembuatan dan pemeliharaan kamus data. Kamus ini mendeskripsikan setiap elemen data dalam *database*. Hal ini memungkinkan semua pengguna (dan *programmer*) untuk berbagi pandangan yang sama mengenai sumber daya data, sehingga sangat memfasilitasi analisis kebutuhan pengguna. Kamus data bisa berbentuk kertas atau *online*. Kebanyakan DBMS menggunakan perangkat lunak khusus untuk mengelola kamus data.




Gambar 4.4
Hubungan Administrator Database dengan Fungsi Lainnya

Interaksi organisasi DBA yang ditunjukkan pada ilustrasi, yaitu interaksi hubungan antara DBA, pengguna akhir, dan profesional sistem perusahaan. Ketika kebutuhan informasi meningkat, pengguna mengirimkan permintaan formal untuk aplikasi komputer ke *programmer* dari perusahaan. Permintaan ini dijawab melalui prosedur pengembangan sistem formal; jika bermanfaat, akan dibuat aplikasi programnya. Permintaan pengguna juga dikirim ke DBA, yang mengevaluasinya untuk menentukan kebutuhan *database* pengguna. Setelah terbentuk, DBA memberikan otoritas akses ke pengguna dengan memprogram tampilan pengguna (subskema). Hubungan ini terlihat pada garis antara pengguna dan DBA dan antara DBA dan modul DDL dalam DBMS. Dengan memisahkan akses dara ke pemrograman aplikasi, perusahaan lebih mampu mengendalikan dan melindungi *database*.

### 4.2.4 Database Fisik

Elemen utama yang keempat dari pendekatan *database* adalah *database* fisik. Elemen ini merupakan tingkat terendah dari *database* dan satu-satunya tingkat yang ada dalam bentuk fisik. *Database* fisik terdiri atas titik magnetis pada disket magnetis. Pada tingkat fisik, *database* membentuk kumpulan catatan logis dan *file* yang merupakan sumber daya data perusahaan. Efisiensi dalam DBMS untuk melaksanakan tugas ini merupakan penentu utama dari keberhasilan secara keseluruhan, dan sangat bergantung pada bagaimana menstrukturkan *file* tertentu.

Tabel 4.2
Operasi dalam Pemrosesan File

| Operasi dalam Pemrosesan File pada Umumnya |
|---|
| 1. Menelusuri catatan dari *file* berdasarkan nilai kunci primernya. |
| 2. Menyisipkan catatan ke dalam *file.* |
| 3. Memperbarui catatan dalam *file.* |
| 4. Membaca *file* catatan secara keseluruhan. |
| 5. Menemukan catatan selanjutnya dalam *file.* |
| 6. Memindai *file* untuk mencari catatan dengan menggunakan kunci skundernya. |
| 7. Menghapus catatan dari *file*. |

#### A. Struktur Data

Struktur data adalah dasar penyusunan *database*. Struktur data memungkinkan catatan untuk ditemukan, disimpan, dan ditelusuri, dan memungkinkan pergerakan dari satu catatan ke catatan lainnya. Struktur data memiliki dua komponen dasar: organisasi dan metode akses.

#### B. Organisasi Data

Organisasi suatu *file* mengacu pada cara catatan diatur secara fisik dalam alat penyimpanan sekunder. Alat penyimpanan bisa bersifat *squential* (berurutan) atau *random* (acak). Catatan dalam *file* berurutan disimpan dalam lokasi yang berkelanjutan yang menempati area tertentu di ruang disket. Catatan dalam *file* acak disimpan

tanpa melihat hubungan fisiknya dengan catatan lainnya dari *file* yang sama. *File* acak bisa memiliki catatan yang terdistribusi di semua bagian disket.

## C. Metode Akses Data

Metode akses adalah teknik yang digunakan untuk mencari catatan dan bernavigasi di *database*. Selama pemrosesan *database*, program metode akses, yang merespons permintaan data dari aplikasi pengguna, mencari dan menelusuri atau menyimpan catatan. Tugas yang dijalankan oleh metode akses bersifat transparan bagi aplikasi pengguna.

Pemilihan struktur melibatkan pertukaran antarfitur yang diinginkan. Adapun kriteria yang memengaruhi pemilihan struktur data mencakup:

- Akses *file* dan penelusuran data yang cepat.
- Penggunaan ruang penyimpanan disket yang efisien.
- Kapasitas untuk memproses transaksi yang tinggi.
- Perlindungan dari kehilangan data.
- Kemudahan pemulihan dari kegagalan sistem.
- Akomodasi pertumbuhan *file*.

## D. Hierarki Data

Pada bagian ini akan dipekernalkan istilah-istilah dalam *database*, terutama mengenai struktur yang ada di dalam *database* dan dimulai dari satuan data yang terkecil.

***Field / Atribut Data***. *Field* atau atribut data adalah *item* tunggal dari data, seperti nama pelanggan, saldo, atau alamat.

***Record***. *Record* adalah suatu kelompok yang erat kaitannya dengan *field* yang mendeskripsikan karakteristik yang relevan dari contoh entitas yang dilacak. *Record* dapat divisualkan sebagai baris dalam tabel data, sedangkan *field* dapat divisualkan dengan kolom.

***File / Tabel / Entitas***. Entitas adalah sumber daya, peristiwa, atau pelaku individual yang akan dipilih untuk mengumpulkan data. Sebagai contoh adalah tabel entitas yang berisi data persediaan, aktivitas penjualan, pelanggan, dan karyawan.

***Database***. *Database* adalah serangkaian tabel atau *file* entitas yang berkaitan erat yang secara bersama-sama membuat aplikasi untuk melayani kebutuhan pengguna dalam hal proses atau fungsi bisnis tertentu. Misalnya, *database* penggajian akan mencakup data yang relevan mengenai semua entitas yang diperlukan untuk menjalankan aktivitas penggajian sesuai dengan kebutuhan perusahaan.

***Database Enterprise***. *Database* ini berisikan serangkaian *file* atau tabel yang diperuntukkan bagi semua bagian yang ada di dalam suatu organisasi atau perusahaan. Saat ini telah banyak sistem aplikasi pengembang berbasis komputer untuk menangani *database* ini, seperti SQL Server, Oracle, dan banyak lagi. Sistem ini berfokus pada kemampuan untuk menggunakan *database enterprise* sebagai landasan untuk aplikasi yang menjadi *interface* di seluruh bagian dalam organisasi atau perusahaan. Salah satu contoh aplikasi yang menerapkan prinsip *database enterprise* adalah *enterprise resource planning* (ERP) yang digunakan untuk menangani

perencanaan semua sumber daya yang ada dalam perusahaan. Aplikasi ini sudah populer sejak tahun 1990-an, mulai masuk ke Indonesia dan sudah banyak diterapkan dalam perusahaan pada tahun 2000-an.

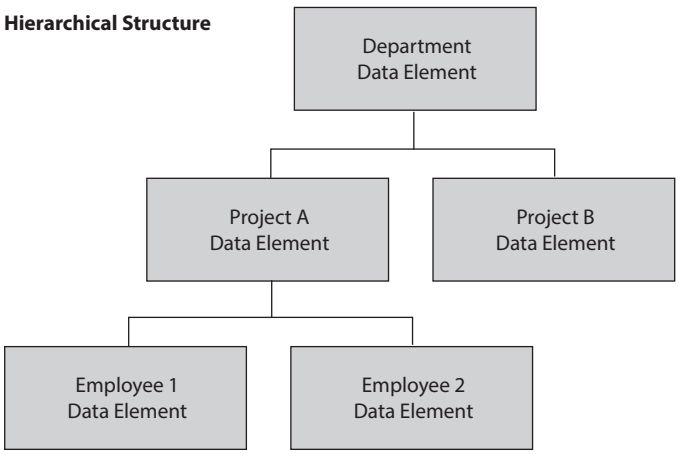
### 4.2.5 Model DBMS

Model data adalah representasi abstrak dari data mengenai entitas, termasuk sumber daya (aset), peristiwa (transaksi), dan pelaku (personalia atau pelanggan) dan hubungan antar-entitas di dalam suatu perusahaan. Tujuan dari model data adalah untuk menyajikan atribut entitas dengan cara yang mudah dipahami oleh pengguna.

Ada tiga model DBMS yang umum, yaitu (1) model hierarkis, (2) jaringan, dan (3) relasional.

#### A. Model Hierarkis

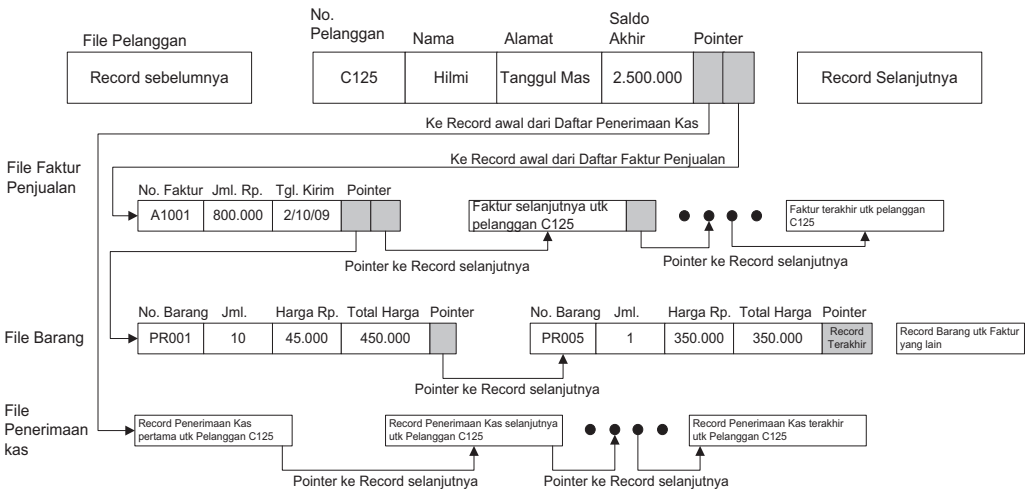
Sistem manajemen *database* ini didasari oleh model ***data hierarkis***. Model ini adalah metode yang paling populer untuk representasi data, karena model ini mencerminkan banyak aspek di dalam perusahaan yang menggunakan hubungan hierarkis.




Gambar 4.5
Model Database dengan Struktur Hierarkis

Struktur ini juga disebut struktur pohon, di mana tingkat tertinggi dari struktur ini adalah segmen *root*, dan *file* terendah dalam cabang tertentu disebut *leaf* (daun). Model kedua adalah ***database navigasional***. Model data ini disebut demikian karena lintas *file* memerlukan jalur yang telah ditentukan sebelumnya. Hal ini ditetapkan melalui hubungan eksplisit (*pointer*) antara berbagai *record* yang terkait. Satu-satunya cara untuk mengakses data pada tingkat yang lebih rendah dalam struktur pohon, yaitu melalui *root* dan via *pointer* turun melalui jalur navigasional ke *record* yang diinginkan.

***Integrasi Data dalam Model Hierarkis***. Model terakhir ini menunjukkan perincian struktur *file* untuk *database* parsial dari *database* navigasional. Lihat ilustrasi pada Gambar 4.6 mengenai *database* untuk menangani data penjualan.



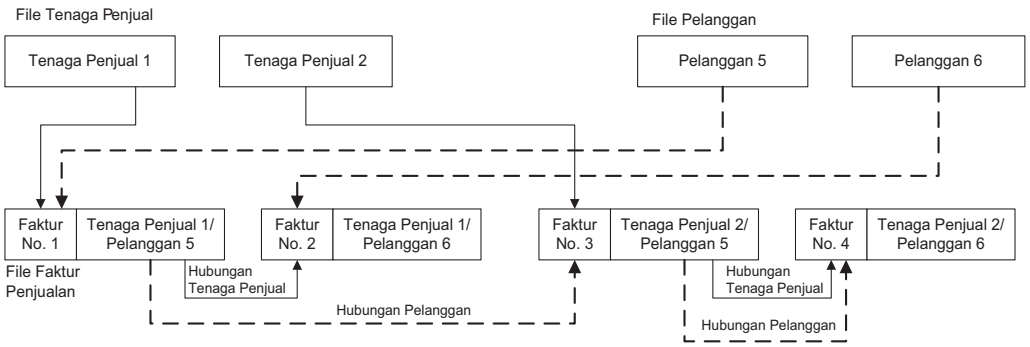

Gambar 4.6
Integrasi Data dalam Database Hierarkis

***Kelemahan Model Hierarkis***. Model ini menyajikan tampilan hubungan data yang terbatas secara artifisial. Berdasarkan proporsi bahwa semua hubungan bisnis bersifat hierarkis, model ini tidak selalu mencerminkan suatu realitas. Peraturan berikut ini, mengatur model hierarkis, menunjukkan kelemahan operasionalnya:

1. *Record parent* bisa memiliki satu atau beberapa catatan *child*. Misalnya, pelanggan adalah *parent* untuk faktur penjualan dan penerimaan kas.
2. Tidak ada *record child* yang boleh memiliki lebih dari satu *parent*.

## B. Model Jaringan

Model jaringan adalah *database* navigasional dengan hubungan eksplisit antara *record* dan *file*. Perbedaannya adalah model ini memungkinkan *record child* memiliki beberapa *parent*. Misalkan, mengacu pada Gambar 4.7, Faktur Nomor 1 adalah *child* dari Tenaga Penjual 1 dan Pelanggan 5. *Field pointer* pada kedua *record parent* secara eksplisit mendefinisikan jalur catatan faktur (*child*).




Gambar 4.7
Hubungan dalam Database Jaringan

Catatan faktur ini memiliki dua hubungan ke *record* yang terkait. Pertama adalah hubungan Tenaga Penjual ke Faktur Nomor 2. *Record* ini berasal dari penjualan oleh Tenaga Penjual 1 ke Pelanggan 6. *Pointer* kedua adalah hubungan pelanggan ke Faktur Nomor 3. Ini mewakili penjualan kedua ke Pelanggan 5, yang diproses oleh Tanaga Penjual 2.

Struktur ini dapat diakses pada *record* tingkat *root* (tenaga penjual atau pelanggan) dengan memasukkan data kunci primer yang sesuai (Nomor Tenaga Penjual atau Nomor Pelanggan).

## C. Model Relasional

Model relasional pertama kali diusulkan oleh E.F. Codd pada akhir tahun 1960-an. Model formal yang memiliki landasan pada aljabar relasional dan rangkaian, yang mana menyediakan dasar teoretis untuk sebagian besar operasi manipulasi data yang digunakan. Perbedaan yang paling nyata antara model relasional dan model navigasional adalah cara asosiasi data yang disajikan ke pengguna. Model relasional menampilkan data dalam bentuk tabel dua dimensi. Lihat ilustrasi pada Gambar 4.8 yang menyajikan contoh satu tabel *database* yang disebut pelanggan.

Pada bagian atas tabel terdapat atribut atau *field* data yang membentuk kolom. Bagian yang memotong kolom untuk membentuk baris dalam tabel disebut *tuple* (*record*). *Tuple* adalah susunan data yang dinormalisasi dan mirip, tetapi tidak sama sepenuhnya, dengan *record* dalam sistem *file* data. Tabel yang didesain dengan baik memiliki empat karakteristik sebagai berikut.

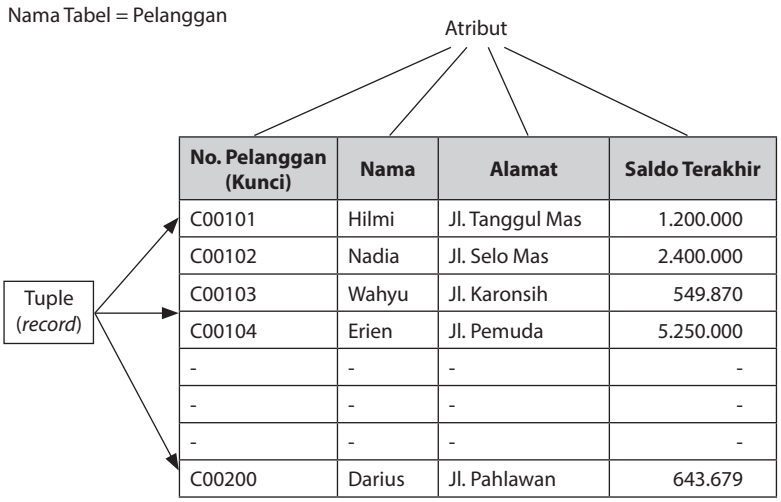
1. Semua kemunculan pada perpotongan baris dan kolom memiliki nilai tunggal. Tidak boleh ada nilai ganda (kelompok berulang).
2. Nilai atribut pada setiap kolom harus memiliki kelas yang sama.
3. Setiap kolom dalam satu tabel harus memiliki nama yang berbeda dengan lainnya. Akan tetapi, tabel yang berbeda bisa memiliki kolom dengan nama yang sama.
4. Setiap baris dalam tabel harus berbeda minimal pada satu atribut. Atribut ini yang disebut dengan *primary key*.

Tabel harus dinormalisasi, karena setiap atribut dalam baris harus bergantung pada *primary key* dan tidak terikat dengan atribut lainnya. Hubungan antara berbagai *record* dalam tabel yang terkait dibentuk melalui operasi logis dari DBMS, bukan melalui alamat eksplisit yang distrukturkan ke *database*.

Sifat asosiasi di antara dua tabel menentukan metode yang digunakan untuk menetapkan *foreign key*. Jika asosiasinya satu lawan satu (*one to one*), tidak masalah *primary key* dari tabel mana yang dihubungkan sebagai *foreign key* tabel lainnya. Dalam asosiasi satu ke banyak (*one to many*), *primary key* pada sisi "satu" dihubungkan sebagai *foreign key* pada sisi "banyak."

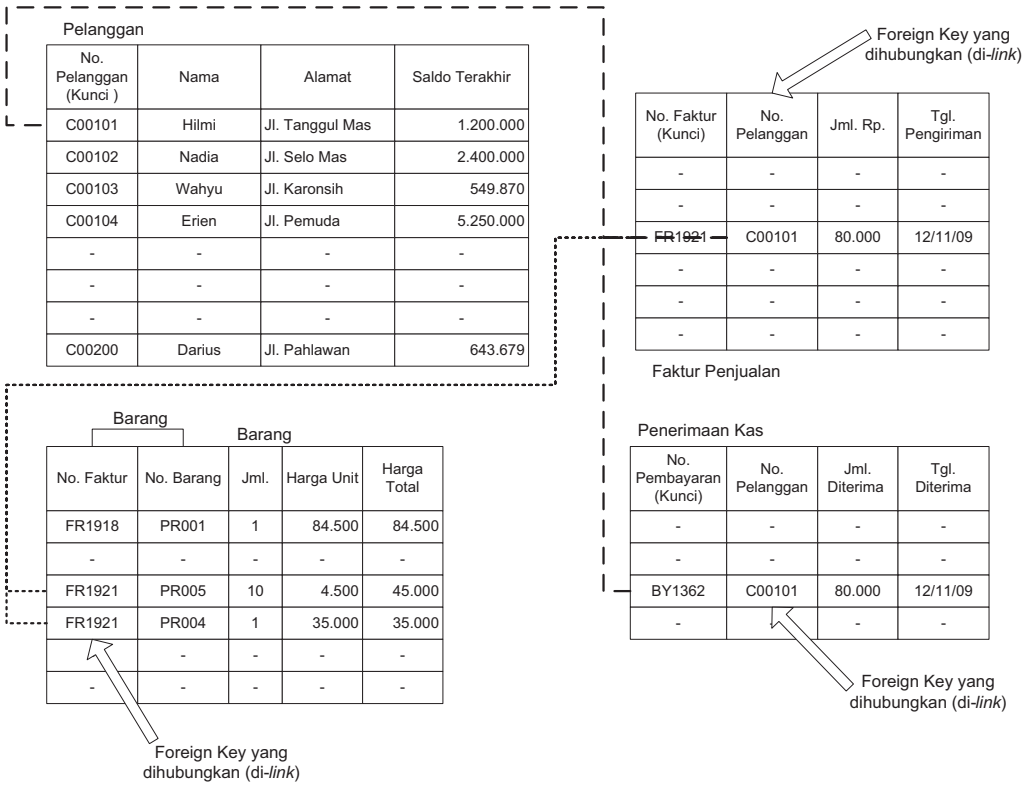
Pada pembahasan sebelumnya, dijelaskan bagaimana *database* navigasional menggunakan penghubung (*pointer*) yang eksplisit antarcatatan untuk membentuk hubungan. Untuk mengilustrasikan perbedaan ini, bandingkan struktur *file* dari tabel relasional pada Gambar 4.8 dengan contoh model hierarkis pada Gambar 4.6.

Hubungan konseptual antar-*file* adalah sama, tetapi tidak adanya *pointer* eksplisit dalam tabel relasional.




| No. Pelanggan (Kunci) | Nama | Alamat | Saldo Terakhir |
|---|---|---|---|
| C00101 | Hilmi | Jl. Tanggul Mas | 1.200.000 |
| C00102 | Nadia | Jl. Selo Mas | 2.400.000 |
| C00103 | Wahyu | Jl. Karonsih | 549.870 |
| C00104 | Erien | Jl. Pemuda | 5.250.000 |
| - | - | - | - |
| - | - | - | - |
| - | - | - | - |
| C00200 | Darius | Jl. Pahlawan | 643.679 |

Gambar 4.8
Tabel Relasi yang Disebut Pelanggan




| No. Pelanggan (Kunci ) | Nama | Alamat | Saldo Terakhir |
|---|---|---|---|
| C00101 | Hilmi | Jl. Tanggul Mas | 1.200.000 |
| C00102 | Nadia | Jl. Selo Mas | 2.400.000 |
| C00103 | Wahyu | Jl. Karonsih | 549.870 |
| C00104 | Erien | Jl. Pemuda | 5.250.000 |
| - | - | - | - |
| - | - | - | - |
| - | - | - | - |
| C00200 | Darius | Jl. Pahlawan | 643.679 |

| No. Faktur (Kunci) | No. Pelanggan | Jml. Rp. | Tgl. Pengiriman |
|---|---|---|---|
| - | - | - | - |
| - | - | - | - |
| FR1921 | C00101 | 80.000 | 12/11/09 |
| - | - | - | - |
| - | - | - | - |
| - | - | - | - |

| No. Faktur | No. Barang | Jml. | Harga Unit | Harga Total |
|---|---|---|---|---|
| FR1918 | PR001 | 1 | 84.500 | 84.500 |
| - | - | - | - | - |
| FR1921 | PR005 | 10 | 4.500 | 45.000 |
| FR1921 | PR004 | 1 | 35.000 | 35.000 |
| - | - | - | - | - |
| - | - | - | - | - |

| No. Pembayaran (Kunci) | No. Pelanggan | Jml. Diterima | Tgl. Diterima |
|---|---|---|---|
| - | - | - | - |
| - | - | - | - |
| BY1362 | C00101 | 80.000 | 12/11/09 |
| - | - | - | - |

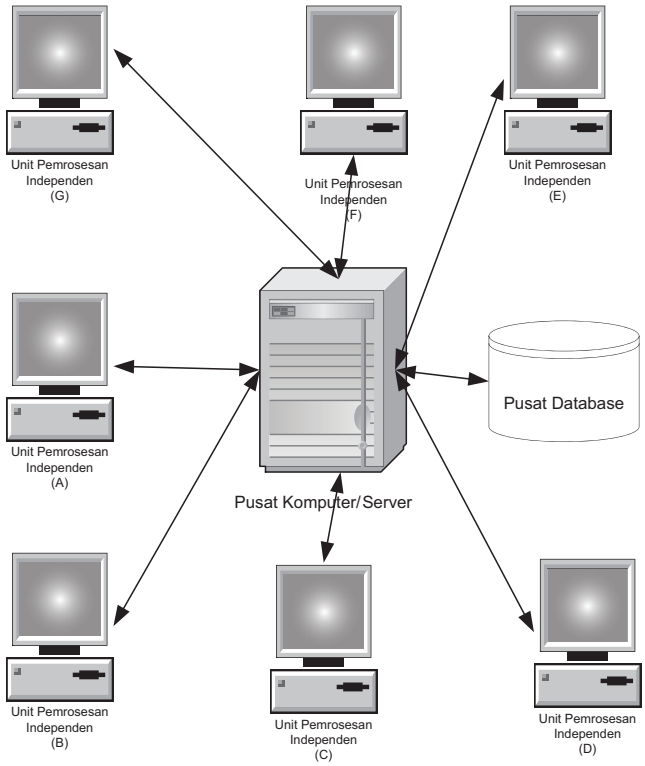
Gambar 4.9
Integrasi Data dalam Model Relasional

## 4.3 DATABASE DALAM LINGKUNGAN TERDISTRIBUSI

Struktur fisik data perusahaan merupakan pertimbangan penting dalam merencanakan sistem terdistribusi. Untuk mengatasi hal ini, perencana memiliki dua pilihan dasar, yaitu: *database* dipusatkan atau *database* didistribusikan. *Database* terdistribusi terdiri dari dua kategori: *database* terpartisi dan *database* tereplikasi.

### 4.3.1 Database Terpusat

Pendekatan pertama melibatkan penempatan pada lokasi pusat. Unit TI di lokasi terpisah mengirim permintaan data ke lokasi pusat, yang memproses permintaan dan mengirimkan data kembali ke unit TI yang memintanya. Pemrosesan sebenarnya dari data yang dilakukan pada unit TI, pusat melaksanakan fungsi sebagai manajer *file* yang melayani kebutuhan data dari unit-unit TI. Tujuan dasar dari pendekatan *database* adalah untuk memelihara kekinian data. Berikut ini adalah ilustrasi pendekatan *database* terpusat.




Gambar 4.10
Pengelolaan Database dengan Pendekatan Terpusat

### 4.3.2 Database Terdistribusi

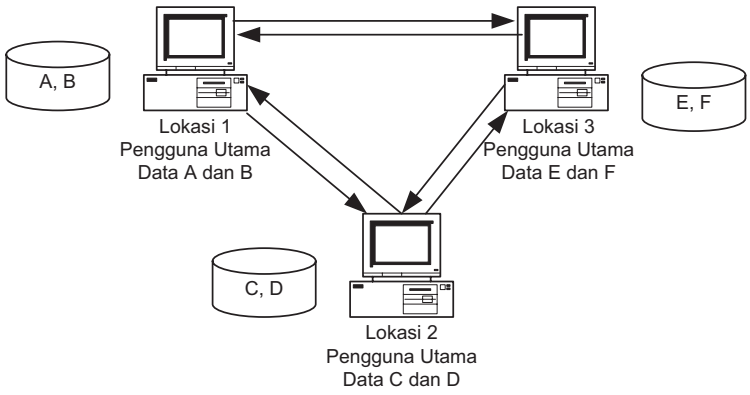
Ada dua sifat dalam *database* terdistribusi, yaitu *database* terpartisi dan *database* tereplikasi.

#### A. Database Terpartisi

Pendekatan *database* terpartisi membagi *database* pusat menjadi beberapa segmen atau partisi yang terdistribusi ke pengguna utama. Keuntungan pendekatan ini adalah sebagai berikut.

- Penyimpan data di lokasi lokal akan meningkatkan pengendalian pengguna.
- Waktu respons pemrosesan transaksi menjadi lebih bauk karena memungkinkan adanya akses lokal ke data dan mengurangi volume data yang harus di kirim antarunit TI.
- *Database* terpartisi bisa mengurangi potensi dampak bencana. Dengan menempatkan data di beberapa lokasi, kehilangan pada satu unit TI tidak akan menghentikan semua pemrosesan data di perusahaan.

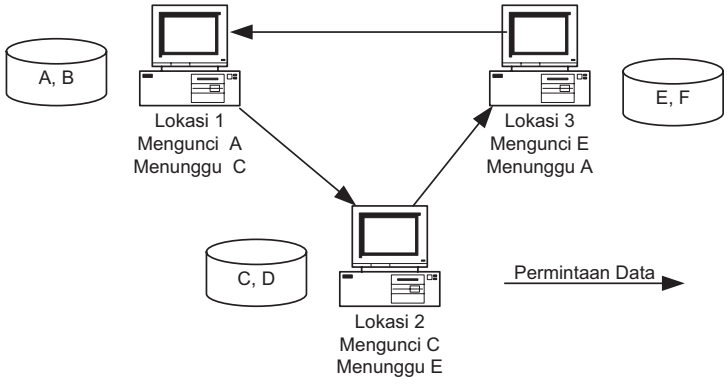
Pendekatan terpartisi, seperti pada ilustrasi di bawah ini paling baik digunakan untuk perusahaan yang memerlukan pembagian data minimal antarunit TI. Pengguna utama mengelola permintaan data dari lokasi yang lain. Untuk meminimalkan akses data dari pengguna yang berjauhan, perusahaan perlu memilih lokasi *host* secara hati-hati. Identifikasi *host* yang optimal memerlukan analisis yang mendalam mengenai kebutuhan data pengguna.




Gambar 4.11
Pengelolaan Database dengan Pendekatan Terdistribusi

#### B. Database Tereplikasi

Pendekatan *database* tereplikasi efektif pada perusahaan yang memiliki tingkat pembagian data yang tinggi, tetapi tidak memiliki pengguna utama. Oleh karena data umum direplikasi pada setiap situs unit TI, lalu lintas data antarlokasi banyak berkurang. Ilustrasi berikut ini menunjukkan model *database* tereplikasi.




Gambar 4.12
Pengelolaan Database dengan Pendekatan Tereplikasi

Justifikasi utama untuk *database* tereplikasi adalah untuk mendukung permintaan yang hanya bisa dibaca saja (*read only*). Dengan replikasi data pada setiap lokasi, akses data untuk tujuan permintaan data dapat dipastikan, dan jalan buntu serta penundaan karena lalu lintas data dapat diminimalkan. Masalah pada pendekatan ini adalah pemeliharaan versi terbaru dari *database* di setiap lokasi. Oleh karena setiap unit TI hanya memproses transaksinya, maka data umum yang direplikasi pada setiap lokasi dipengaruhi oleh berbagai transaksi dan mencerminkan nilai yang berbeda-beda.

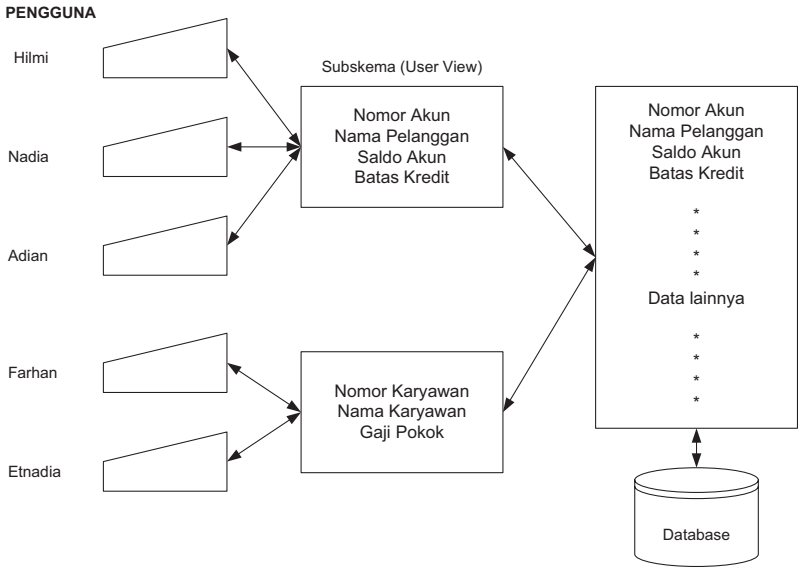
## 4.4 PENGENDALIAN DAN AUDIT SISTEM MANAJEMEN DATA

Pengendalian atas sistem manajemen data terdiri atas dua kategori umum, yaitu pengendalian akses dan pengendalian *backup*. Pengendalian akses didesain untuk mencegah individu yang tidak memiliki otorisasi untuk melihat, menelusuri, memanipulasi, atau merusak data entitas. Pengendalian *backup* memastikan bahwa jika terjadi kehilangan data karena akses yang tidak diotorisasi, kegagalan alat, atau bencana fisik, perusahaan dapat memulihkan *database* (*recovery*).

### 4.4.1 Pengendalian Akses

Pengguna *flat file* mempunyai kepemilikan eksklusif atas data mereka. Meskipun terdapat permasalahan integrasi data yang berkaitan dengan model ini, pengendalian akses menciptakan lingkungan di mana akses yang tidak memiliki otorisasi ke data dapat dikendalikan secara efektif. Ketika tidak digunakan oleh pemiliknya, *file flat* tertutup bagi pengguna lainnya dan bisa dibuat *offline* dan diamankan secara fisik dalam perpustakaan data.

Dalam lingkungan *database* yang dibagikan (*share*), risiko pengendalian akses mencakup manipulasi, pencurian, penyalahgunaan, dan perusakan data. Ancaman ini berasal dari penyusup yang tidak memiliki otorisasi dan pengguna yang memiliki otorisasi, tetapi melebihi hak akses yang dimilikinya. Berikut ini akan dibahas mengenai fitur-fitur pengendalian akses.




Gambar 4.13
Subskema yang Membatasi Akses ke Database

## A. Tampilan Pengguna

Subskema atau tampilan pengguna (*user view*) adalah bagian dari *database* total yang mendefinisikan domain data pengguna dan menyediakan akses ke *database*. Pada Gambar 4.13 mengilustrasikan peran tampilan pengguna. Dalam lingkungan *database* terpusat, administrator *database* memiliki tanggung jawab utama untuk mendesain tampilan pengguna, tetapi bekerja dekat dengan pengguna dan desainer sistem untuk melaksanakan tugas tersebut. Hak akses ke *database* harus sesuai dengan kebutuhan dari pengguna.

Meskipun *user view* dapat membatasi akses pengguna ke sekumpulan data yang terbatas, tampilan tersebut tidak mendefinisikan hak tugas seperti *read*, *delete*, atau *write*. Sering kali, beberapa pengguna memiliki *user view* yang sama, tetapi mereka memiliki level otorisasi yang berbeda (lihat ilustrasi pada Gambar 4.13).

## B. Tabel Otorisasi Database

Tabel otorisasi *database* berisi peraturan yang membatasi tindakan yang bisa diambil oleh pengguna. Teknik sama dengan daftar pengendalian akses yang digunakan dalam sistem operasi. Setiap pengguna diberikan hak akses tertentu yang dikodekan ke dalam tabel otoritas yang digunakan untuk memverifikasi permintaan tindakan pengguna. Misalkan, pada Gambar 4.13 menunjukkan *user* Hilmi, Nadia, dan Adian memiliki akses ke atribut data yang sama melalui *user view* yang umum, tetapi pada tabel otorisasi (lihat Tabel 4.3) menunjukkan bahwa hanya Nadia yang memiliki wewenang untuk memodifikasi dan menghapus data. Setiap baris dalam tabel otorisasi menunjukkan tingkat tindakan (*read, insert, modify,* atau *delete*) yang bisa dilakukan oleh setiap *user* setelah memasukkan *password* yang benar.

Tabel 4.3
Tabel Otorisasi Database

| Dept. | Piutang Usaha | | | Penagihan | |
|---|---|---|---|---|---|
| User | Hilmi | Nadia | Adian | Farhan | Etnadia |
| Password | Bugs | Dog | Katie | Lucky | Star |
| **Otoritas:** | | | | | |
| Read | ✓ | ✓ | ✓ | ✓ | ✓ |
| Insert | ✓ | × | ✓ | ✓ | × |
| Modify | ✓ | × | × | ✓ | × |
| Delete | ✓ | × | × | × | × |

## C. User-Defined Procedure

Prosedur ini memungkinkan *user* untuk membuat program atau rutinitas (*routine*) keamanan pribadi untuk menyediakan identifikasi *user* yang lebih positif daripada *password* tunggal. Jadi, selain *password*, prosedur ini juga akan memberikan serangkaian pertanyaan pribadi.

## D. Enkripsi Data

Saat ini banyak sistem *database* yang menggunakan prosedur enkripsi untuk melindungi data yang sangat sensitif, seperti formula produk, level gaji karyawan, *file passsword*, dan data keuangan. Enkripsi data menggunakan algoritma untuk mengacak data, sehingga tidak bisa dibaca oleh penyusup yang mengeksplorasi *database*. Selain melindungi data yang disimpan, enkripsi juga digunakan untuk melindungi data yang dikirim melalui jalur komunikasi.

## E. Teknologi Biometrik

Prosedur autentikasi pengguna yang terakhir adalah penggunaan teknologi biometrik, yang mengukur berbagai karakteristik pribadi, seperti sidik jari, suara, retina, atau karakteristik tanda tangan. Karakteristik pengguna ini dibuat dalam bentuk digital dan disimpan secara permanen dalam *file* keamanan *database* atau pada kartu indentitas yang dibawa oleh pengguna. Pada saat seseorang berusaha mangakses *database*, alat pemindai khusus akan menangkap dan merekam karakteristik biometriknya, kemudian membandingkannya dengan data yang disimpan dalam *file* atau kartu ID. Jika data tidak sesuai, maka akses akan ditolak.

## F. Pengendalian Inferensi

Salah satu keuntungan dari penerimaan data ke *database* adalah kemampuannya untuk menyediakan ringkasan dan data statistik ke pengguna untuk mengambil keputusan. Misalnya, para manajer yang mungkin mengajukan pertanyaan sebagai berikut.

- Berapa nilai total dari barang persediaan yang perputaran bulanannya kurang dari tiga?
- Berapa biaya rata-rata bagi pasien yang menginap di rumah sakit lebih dari delapan hari?

- Berapa biaya total untuk menggaji karyawan golongan II pada bagian produksi?

Jawaban di atas merupakan jenis pertanyaan yang diperlukan secara rutin oleh manajer SDM, perencanaan sumber daya, dan keputusan pengendalian operasi. Permintaan data kadang-kadang mencakup akses ke data rahasia. Jadi, pengguna mungkin diberi akses berupa ringkasan dari data rahasia, pengendalian inferensi harus ditempatkan untuk mencegah pengguna yang ingin mengacaukan nilai data tertentu melalui fitur permintaan data, meskipun pengguna tersebut tidak memiliki otorisasi untuk mengaksesnya. Pengendalian inferensi digunakan untuk mencegah tiga jenis kompromi *database*, adalah sebagai berikut.

- *Kompromi positif*—pengguna menentukan nilai tertentu dari *item* data.
- *Kompromi negatif*—pengguna menentukan bahwa *item* data tidak memiliki nilai tertentu.
- *Kompromi perkiraan*—pengguna tidak bisa menentukan nilai yang tepat dari *item*, tetapi mampu memperkirakannya dengan keakuratan yang memadai guna melanggar kerahasiaan data.

## Tujuan Audit

Memverifikasi bahwa otoritas akses *database* dan hak khusus diberikan ke pengguna sesuai dengan kebutuhan logis mereka.

## Prosedur Audit

***Tanggung Jawab untuk Tabel Otoritas dan Subskema***. Auditor harus memverifikasi bahwa personel administrasi *database* mempertahankan tanggung jawab yang eksklusif untuk membuat tabel otoritas dan mendesain tampilan pengguna. Bukti-bukti bisa berasal dari tiga sumber: (1) dengan meninjau kembali kebijakan perusahaan dan deskripsi tugas, yang memuat perincian tanggung jawab teknis; (2) dengan memeriksa tabel otoritas *programmer* mengenai hak akses khususnya ke perintah DDL; dan (3) melalui wawancara pribadi dengan *programmer* dan personel administrasi *database*.

***Otoritas Akses yang Sesuai***. Auditor bisa memilih sampel pengguna dan memverifikasi bahwa hak akses mereka yang disimpan dalam tabel otoritas sesuai dengan fungsi organisasi.

***Pengendalian Biometrik***. Auditor harus mengevaluasi biaya dan manfaat dari pengendalian biometrik. Secara umum, ini akan sangat tepat jika data yang sangat sensitif diakses oleh sejumlah pengguna yang sangat terbatas.

***Pengendalian Enkripsi***. Auditor harus memverifikasi bahwa data yang sensitif, seperti *password*, dienkripsi dengan baik. Hal ini dapat dilakukan dengan mencetak isi *file* ke kertas.

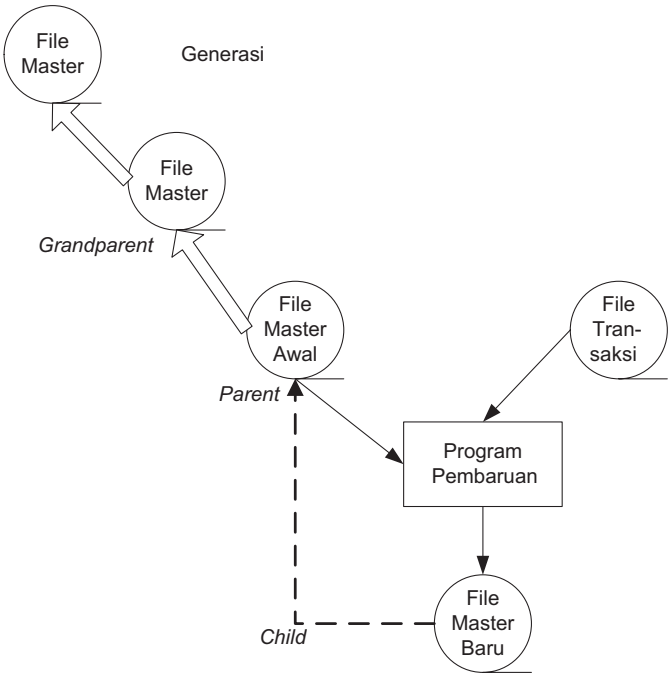
***Pengendalian Inferensi***. Auditor harus memverifikasi bahwa pengendalian permintaan data ke *database* ada untuk mencegah akses yang tidak memiliki otoritas melalui inferensi. Auditor bisa memeriksa pengendalian ini dengan melakukan simulasi akses dari sampel pengguna dan berusaha menelusuri data yang tidak diotorisasi melalui permintaan data inferensi.

### 4.4.2 Pengendalian Backup

Data bisa dimanipulasi dan dirusak oleh tindakan yang berbahaya dari *hacker* eksternal, karyawan yang sakit hati, kegagalan media simpan, kesalahan program, dan bencana alam. Untuk memulihkan bencana, perusahaan harus menerapkan kebijakan, prosedur, dan teknik yang secara sistematis dan rutin menyediakan salinan *backup* dari *file* penting.

#### A. Pengendalian Backup di Lingkungan File Datar

Teknik *backup* yang digunakan akan bergantung pada media dan struktur *file*. *File* berurutan (*sequential*)—pita magnet—menggunakan teknik pembuatan *backup* yang disebut *grandparent-parent-child* (GPC). Teknik *backup* ini merupakan bagian integral dari proses *update file* utama. Sebaliknya, *file* akses langsung (*direct access*) memerlukan prosedur pembuatan *backup* secara terpisah.

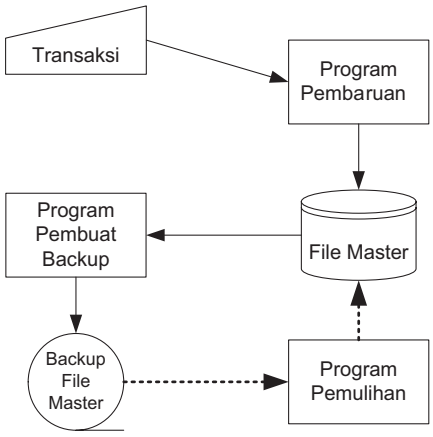



Gambar 4.14
Pendekatan Grandparent-Parent-Child

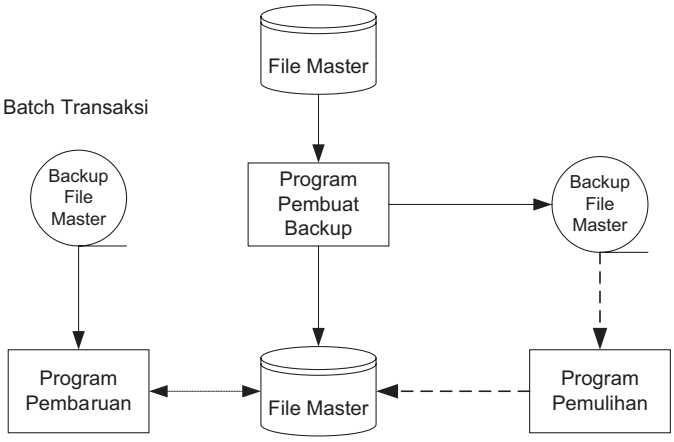
**Teknik *backup* GPC** (lihat Gambar 4.14), menggambarkan teknik yang digunakan dalam sistem *batch file* berurutan. Prosedur pembuatan *backup* dimulai ketika *file* master (*parent*) diproses berdasarkan *file* transaksi untuk menghasilkan *file* utama baru yang telah di-*update* (*child*). Pada *batch* transaksi berikutnya, *child* menjadi *file* utama yang sedang digunakan (*parent*), dan *parent* yang sebelumnya menjadi *file backup* (*grandparent*). *File* utama baru, yang dihasilkan dari proses *update* adalah *child*. Prosedur terus berlanjut pada setiap *batch* transaksi, sehingga menciptakan beberapa

generasi *file backup*. Ketika salinan *file backup* yang diharapkan tercapai, maka *file backup* yang paling lama akan dihapus. Jika *file* master yang sedang digunakan rusak atau dimanipulasi, pemrosesan *file backup* yang terbaru dengan *file* transaksi yang terkait dapat menghasilkan kembali *file* master tersebut.

***Backup File Direct Access***. Nilai data di dalam *file direct access* diubah di tempat melalui proses yang disebut *destructive change*. Oleh sebab itu, setelah nilai data diubah, nilai awalnya akan dihapus, sehingga hanya tinggal satu versi saja yang tersisa dari *file* (versi pertama). Untuk menyediakan *file backup*, akses langsung harus disalin sebelum di-*update* (lihat Gambar 4.15).




**Sistem Pemrosesan Real-Time**
Menggunakan pembuatan *backup* berdasarkan waktu. Transaksi yang diproses antara pembuatan cadangan akan harus diproses kembali setelah restorasi *file* utama.




**Sistem Pemrosesan Batch**
Pemrosesan menggunakan *file* akses langsung, *file* utama disalin sebelum di-*update*.

Gambar 4.15
Salinan File Direct Access

Penentuan waktu dari prosedur *backup direct access* akan bergantung pada metode pemrosesan yang digunakan *file*. *File backup* dalam sistem *batch* biasanya dijadwalkan sebelum proses *update*. Sistem *real-time* menyajikan masalah lebih sulit, karena transaksi diproses terus-menerus, sehingga prosedur pembuatan *backup* dilakukan pada interval tertentu selama sehari (misalnya, setiap 15 menit).

Jika versi *file* master yang sedang digunakan rusak, maka *file* tersebut dapat direkonstruksi dengan menggunakan program *recovery* khusus dari *file backup* yang terbaru. Dalam kasus sistem *real-time*, transaksi yang diproses sejak *backup* terakhir dan sebelum kegagalan akan hilang dan akan perlu diproses kembali untuk memulihkan *file* utama ke status terbaru.

**Penyimpanan ke Tempat Lain (*Offsite*)**. Sebagai perlindungan tambahan, *file backup* yang dibuat melalui pendekatan GPC dan *direct access* sebaiknya disimpan di lokasi lain yang aman.

### Tujuan Audit

Memverifikasi bahwa pengendalian pembuatan *backup* yang diterapkan berfungsi efektif dalam melindungi *file* data dari kerusakan fisik, kehilangan, penghapusan yang tidak disengaja, dan *data corrupt* karena kegagalan sistem dan kesalahan program.

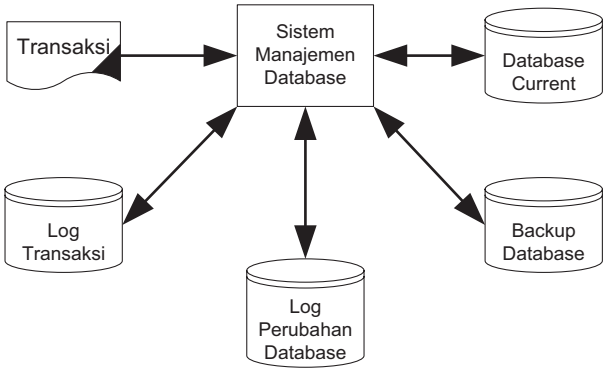
### Prosedur Audit

- ***Backup File Sequential* (GPC)**. Auditor harus memilih sampel sistem dan menentukan dari dokumentasi sistem bahwa jumlah *file* cadangan GPC yang ditentukan dalam sistem memadai. Jika terdapat versi cadangan yang tidak memadai, pemulihan dari beberapa jenis kegagalan mungkin tidak dapat dilakukan.
- ***File Transaksi Backup***. Auditor harus memverifikasi melalui pengamatan fisik bahwa *file* transaksi yang digunakan untuk merekonstruksi *file* utama juga dipertahankan. Tanpa *file* transaksi yang sesuai, rekonstruksi tidak dimungkinkan.
- ***Backup File Direct Access***. Auditor harus memilih sampel aplikasi dan mengidentifikasi *file dirrect access* yang di-*update* dalam setiap sistem. Dari dokumentasi sistem dan melalui pengamatan, auditor bisa memverifikasi bahwa setiap *file* disalin ke pita magnet atau media simpan lain sebelum di-*update*.
- ***Penyimpan ke Lokasi Lain***. Auditor harus memverifikasi keberadaan dan kelayakan lokasi penyimpanan lain. Prosedur audit ini bisa dilakukan sebagai bagian dari peninjauan rencana pemulihan bencana atau pengendalian operasi pusat komputer.

## 4.4.3 Pengendalian Backup di Lingkungan Database

Oleh karena salah satu tujuan dari pendekatan *database* adalah *share data*, maka lingkungan ini cukup rentan terhadap kerusakan dari pengguna individual. Satu prosedur yang tidak berotorisasi, satu tindakan yang membahayakan, atau satu

kesalahan program dapat merugikan seluruh pengguna sumber daya informasi tersebut. Selain itu, karena sentralisasi data, bahkan bencana kecil seperti kegagalan media simpan akan memengaruhi banyak atau semua pengguna. Ketika kejadian semacam itu terjadi, maka perusahaan perlu melakukan rekonstruksi *database* ke status prakegagalan. Hal ini hanya bisa dilakukan jika *database* telah dibuat *backup* dengan memadai.




Gambar 4.16
Pendekatan Backup dan Recovery Database

Pada umumnya *mainframe* DBMS memiliki sistem *backup* dan *recovery* yang mirip dengan yang diilustrasikan pada Gambar 41.6. Sistem ini menyediakan empat fitur *backup* dan *recovery*, yaitu: *backup database*, *log* transaksi, *checking point*, dan modul *recovery*.

***Backup***. Fitur *backup* membuat *backup* dari seluruh *database* secara berkala. Fitur ini adalah prosedur otomatis yang harus dilakukan minimal satu kali sehari. Salinan *backup* kemudian harus disimpan dalam area lain yang lebih aman.

***Log Transaksi* (*Jurnal*)**. Fitur ini menyediakan jejak audit dari semua transaksi yang diproses. *Log* ini membuat daftar transaksi ke dalam *file log* transaksi dan mencatat perubahan yang dihasilkan ke *database* dalam *log* perubahan *database* yang terpisah.

***Checking Point***. Fasilitas ini akan menunda semua pemrosesan data ketika sistem merekonsiliasi *log* transaksi dan *log* perubahan *database* dengan *database*. Pada saat sistem dalam keadaan diam (*idle*), *checking point* akan melakukan pemeriksaan secara otomatis beberapa kali dalam dalam satu jam. Jika terjadi kegagalan, biasanya pemrosesan diulangi mulai dari titik pemeriksaan terakhir. Jadi, hanya beberapa menit pemrosesan transaksi yang harus diulang.

***Modul Recovery***. Modul ini menggunakan *log* dan *file backup* untuk menjalankan kembali sistem setelah mengalami kegagalan.

### Tujuan Audit

Memverifikasi bahwa pengendalian atas sumber daya data memadai untuk menjaga integritas dan keamanan fisik *database*.

### Prosedur Audit

Auditor harus memverifikasi bahwa *backup* dilakukan secara rutin dan sering untuk memfasilitasi pemulihan data yang hilang, rusak, atau *corrupt* tanpa terlalu banyak pemrosesan. *Database* produksi harus disalin dalam interval tertentu. Dengan demikian, harus ada keseimbangan antara ketidaknyamanan karena aktivitas *backup* yang sering dilakukan dengan gangguan bisnis yang disebabkan oleh pemrosesan ulang yang terlalu banyak. Auditor harus memverifikasi bahwa prosedur otomatis untuk *backup* ada dan berfungsi, dan salinan *database* disimpan di lokasi lain untuk keamanan lebih lanjut.

## SOAL DAN STUDI KASUS

1. Apa yang disebut dengan model *file* datar (*flat file*)?
2. Sebutkan empat elemen utama pendekatan *database*!
3. Masalah apa dalam manajemen data *file* datar yang dapat diselesaikan dengan menggunakan konsep *database*?
4. Sebutkan dua komponen fundamental dari struktur data!
5. Apa yang disebut dengan atribut (*field*) data?
6. Apa yang disebut dengan *database*?
7. Mengapa model *database* hierarkis dipandang sebagai *database* navigasional? Apa saja kelemahan model *database* hierarkis?
8. Apa yang disebut dengan *database* tereplikasi, dan mengapa pengendalian bersamaan sulit untuk mengelola *background* ini?
9. Sebutkan empat fitur *backup* dan *recovery* yang diperlukan dalam DBMS! Jelaskan masing-masing fitur tersebut secara singkat!

# BAB 5

# ALAT DAN TEKNIK AUDIT BERBANTUAN KOMPUTER

Setelah mempelajari bab ini, Anda diharapkan mampu:

- Memahami mengenai salah satu tahapan penting dalam melakukan audit, yaitu pengujian substantif yang digunakan untuk menguji kemungkinan kesalahan parameter moneter yang berakibat langsung pada laporan keuangan.
- Mengenal dan memahami bermacam alat pengujian, yaitu media untuk membantu auditor, dan berguna dalam melakukan teknik pengujian.
- Membedakan antara alat pengujian dan teknik pengujian.
- Mengetahui dan memahami teknik dan alat pengujian substantif yang sering digunakan oleh auditor.
- Memahami pendekatan yang dapat digunakan untuk mendapatkan program komputer yang cocok digunakan untuk mengevaluasi dan menguji *record*.
- Mengetahui dan memahami enam standar audit di lingkungan PDE yang diterbitkan oleh IAI dalam SPAK.

Pada bab ini, khusus akan membahas mengenai alat dan teknik audit yang menggunakan alat bantu komputer atau teknologi informasi, khususnya pada saat melakukan pengujian substantif. Dimulai dengan pembahasan pengujian substantif, pada bagian ini akan mengulas secara mendalam mengenai pengujian substantif di lingkungan sistem informasi berbasis komputer. Kemudian akan dilanjutkan dengan pembahasan mengenai alat pengujian apa saja yang dapat membantu auditor dalam melakukan pengujian.

Pada bagian kedua dari bab ini akan membahas mengenai pendekatan yang digunakan untuk memperoleh program komputer untuk melakukan kegiatan audit. Selanjutnya akan dibahas pula mengenai kriteria *software* yang dapat digunakan dalam kegiatan pengujian.

Pada bagian ketiga akan membahas mengenai enam standar audit yang diterbitkan oleh Ikatan Akuntan Indonesia (IAI) dalam Standar Profesional Akuntan Publik (SPAP). Keenam standar auditing tersebut berkaitan dengan pengendalian internal di lingkungan Pemrosesan Data Elektronik (PDE), teknik audit berbantuan komputer (TABK), audit dalam lingkungan, dan tiga standar yang berkaitan dengan lingkungan PDE, yaitu untuk komputer mikro, sistem *online* dan sistem *database*.

## 5.1 PENGUJIAN SUBSTANTIF

Pengujian kepatuhan (*compliance testing*) dan pengujian substantif (*substantive testing*) merupakan kelompok pengujian yang saling melengkapi satu sama lain.

Pelaksanaan pengujian kepatuhan dapat berupa: 1) menguji apakah pengendalian benar-benar dijalankan sebagaimana dijabarkan oleh *auditee* di dalam dokumentasi program; dan 2) menguji apakah pengendalian yang dipasang telah sejalan atau mendukung kebijakan dan prosedur yang diterapkan manajemen.

Misalnya, auditor ingin menguji sejauh mana pengendalian *library* program (*program library control*) telah berjalan, maka auditor dapat mengambil sampel program, kemudian mengujinya apakah versi dari *source code* dan *object code* masih sama.

Pengujian substantif bertujuan mendapatkan keyakinan secara mendalam atas keandalan pengendalian yang diterapkan untuk melindungi organisasi dari kemungkinan tindakan kecurangan. Auditor finansial melakukan pengujian substantif untuk menguji kemungkinan kesalahan parameter moneter yang berakibat langsung pada laporan keuangan. Bagi auditor sistem informasi, tingkat pengujian substantif akan jauh lebih ekstensif.

Sebagai contoh kasus: Seorang auditor sistem informasi mungkin mengembangkan pengujian substantif untuk menguji apakah daftar persediaan (*inventory*) yang ada di *tape library* telah dilaporkan dengan benar. Untuk melaksanakan pengujian ini, auditor sistem informasi mungkin akan mengambil 100% persediaan, atau mungkin juga menggunakan *statistical sampling* yang memungkinkannya mengambil kesimpulan tentang keakuratan seluruh persediaan.

Berdasarkan uraian di atas dapat disimpulkan bahwa seorang auditor sistem informasi perlu memahami CAATs (*Computer Assisted Audit Techniques*). Istilah CAATs diterjemahkan oleh Ikatan Akuntan Indonesia menjadi TABK (Teknik Audit Berbantuan Komputer). Meningkatnya kebutuhan atas informasi yang lebih akurat, tepat waktu, mencerminkan kondisi terkini (*current*), maka pemakaian teknologi dalam sistem informasi akan semakin intensif.

Untuk memenuhi kebutuhan informasi yang mencerminkan kondisi terkini di antaranya adalah dengan pemrosesan transaksi secara *Online Real-Time* (OLRT), misalnya penggunaan *Optical Card Reader* (OCR) di pusat perkulakan (*grocery store*), Anjungan Tunai Mandiri (ATM =*Automated Teller Machine*) dan *Electronic Funds Transfer* (EFT) di dunia perbankan, serta *Electronic Data Interchange* (EDI) dalam transaksi pemesanan barang. Implementasi sistem informasi tersebut menuntut kebutuhan atas teknologi komputer yang lebih canggih. Namun demikian, terdapat kecenderungan bahwa semakin canggih teknologi komputer yang digunakan serta semakin besar kapasitas sistem dan volume transaksi yang diolah, semakin berkurang *audit trail* yang tersedia. Dalam kondisi ini mutlak diperlukan pengujian secara langsung terhadap sistem informasi yang digunakan untuk mengolah data transaksi. Teknik pengujian yang dapat digunakan antara lain: *Parallel Simulation*, *Integrated Test Facility* (ITF), dan *Test Deck*.

## 5.2 ALAT DAN TEKNIK PENGUJIAN

Alat pengujian (*test tools*) adalah berbeda dengan teknik pengujian (*test techniques*). Alat pengujian adalah media untuk membantu auditor, dan berguna dalam melakukan teknik pengujian. Alat pengujian dapat berbentuk pedoman, prosedur atau perangkat lunak (*software*). Sedangkan teknik pengujian adalah cara di mana proses pengujian berlangsung untuk mencapai tujuan audit. Teknik pengujian dilakukan untuk menguji suatu fungsi atau kondisi, di mana hasil pengujian tersebut akan digunakan untuk melakukan penilaian/penelaahan lebih lanjut.

Untuk membedakan antara alat dan teknik pengujian, gambaran berikut mungkin dapat membantu: General Audit Software seperti ACL (*Audit Command Language*), IDEA (*Interactive Data Extraction and Analysis*), dan sejenisnya, termasuk alat pengujian. Sedangkan proses atau cara yang dilakukan untuk melakukan pengujian seperti analisis statistik, stratifikasi, klasifikasi data, rekonsiliasi, dan sejenisnya, adalah teknik pengujiannya. Teknik pengujian rekonsiliasi ini, dalam praktiknya dapat menggunakan *tools* seperti *audit software*, atau dapat juga dilakukan secara manual.

Berikut ini dikemukakan hal-hal yang perlu dipertimbangkan oleh auditor sebelum melakukan TABK, antara lain:

- Pengetahuan dan keterampilan komputer dari auditor,
- Perangkat keras dan perangkat lunak yang tersedia,
- Perangkat keras dan perangkat lunak yang dimiliki *auditee*,

- Data uji yang digunakan,
- Waktu pengujian, dan
- Efektivitas dan efisiensi.

### 5.2.1 Alat dan Teknik Pengujian Substantif

Terdapat banyak alat dan teknik pengujian substantif yang tersedia bagi auditor dalam menguji pengendalian pada sistem informasi berbasis komputer. Namun, di sini hanya membahas beberapa alat dan teknik pengujian yang sering digunakan dalam *real audit*, yaitu:

1. Batch, Offline, Parallel Simulation
2. Embedded, Online, Parallel Simulation
3. Generalized Audit Software
4. Virtual Transaction Testing
5. Embedded Auditability Modules
6. Auditors's Database Segments
7. Embedded Audit Data Collection Modules
8. Extended Audit Records

#### 1. Batch, Offline, Parallel Simulation

Teknik pengujian ini umum digunakan oleh ekstern dengan cara membuat salinan program dari sistem yang diuji, disebut dengan *simulated systems*. Selanjutnya auditor memasukkan data, berupa transaksi riil periode sebelumnya. Validasi dilakukan dengan cara membandingkan laporan yang diperoleh dengan laporan yang dihasilkan oleh sistem yang diuji.

Teknik pengujian ini mempunyai perbedaan dengan *test deck,* yaitu dari segi program dan dari segi data yang digunakan. Program yang digunakan dalam *test deck* adalah program yang digunakan perusahaan (*production program*), sedangkan data yang digunakan adalah data yang dibuat oleh auditor atau biasa disebut *dummy data.*

Teknik pengujian *parallel simulation* ini relatif mahal biayanya, yaitu dalam kaitannya dengan penyusunan (pemrograman) *simulated systems*, terutama bila sistem yang diuji sangat kompleks/rumit. Namun demikian, teknik pengujian ini perlu dilakukan dalam kondisi-kondisi sebagai berikut.

- Tidak adanya dokumentasi atas sistem yang diuji.
- Auditor ingin lebih independen dengan menghindari penggunaan program yang tersedia dalam divisi sistem informasi.
- Perlu dilakukan pengujian atas *file* transaksi.
- Terdapat cara yang ekonomis untuk menyusun program, misalnya menggunakan *generalized audit software package*.

#### 2. Embedded, Online, Parallel Simulation

Teknik pengujian ini merupakan pengembangan dari *batch parallel simulation.* Perbedaannya adalah bahwa teknik pengujian ini berada dalam sistem yang diuji

atau dapat dijalankan secara *online* dengan sistem yang diuji. Pengertian *online* di sini berarti bahwa teknik pengujian (*simulated program*) tersebut tidak perlu berada dalam lingkungan sistem yang diuji.

Oleh karena adanya dua program, yaitu *production program* dan *simulated program,* maka penerapan teknik pengujian *parallel simulation* ini dipandang sangat mahal. Oleh karena itu, teknik pengujian ini biasanya hanya digunakan untuk menguji sistem yang sangat sensitif atau mempunyai risiko tinggi terhadap terjadinya kecurangan atau kesalahan.

Apabila *system analyst/programmer* merancang *parallel simulation* secara permanen, maka pengujian dapat dilaksanakan sepanjang tahun atau setiap saat diinginkan. Dalam keadaan demikian, teknik pengujian tersebut dinamakan *normative approach to testing computerized systems*.

### 3. Generalized Audit Software (GAS)

Dari sifatnya, *Generalized Audit Software* tidak melakukan pengujian atas suatu sistem, karena orientasinya lebih diarahkan sebagai alat untuk melakukan pengujian data (*data interogation*). Oleh karena itu, GAS lebih tepat digolongkan sebagai alat pengujian. Misalnya, seorang auditor menulis program *parallel simulation* dengan menggunakan GAS. Program bahasa komputer yang digunakan tersebut merupakan alat pengujian, sedang *parallel simulation* adalah teknik pengujian.

Dibandingkan dengan *higher level language*, suatu *Generalized Audit Software* lebih mudah dipelajari dan lebih sederhana. Teknik pengujian ini biasanya mempunyai fasilitas yang dibutuhkan auditor seperti *random sampling,* membandingkan *file*, mengurutkan (*sort*) *record*, menguji urutan dan duplikasi *(sequence and duplicate) record*, memilih *(extract) record,* konfirmasi, serta statistik. Secara singkat, fungsi-fungsi yang biasanya dilakukan secara manual oleh auditor, dapat ditulis programnya dengan menggunakan GAS ini.

### 4. Virtual Transaction Testing (VTT)

Dengan teknik pengujian ini, pengujian sistem dilakukan dalam partisi yang berbeda dengan sistem yang diuji. Oleh karena itu, sebelum melakukan pengujian, auditor harus mengopi sistem yang diuji ke dalam partisi yang berbeda serta memberikan nama yang berbeda. Selanjutnya auditor juga membentuk *database* untuk keperluan pengujian. Dengan menggunakan instruksi tertentu (*job control language*), sistem yang ada mengolah transaksi dari terminal auditor ke *database* pengujian, dan memproses transaksi rutin (*production transaction*) ke *database* rutin (*production database*).

Dengan prosedur ini tidak ada kekhawatiran bahwa transaksi penguji tercampur dengan transaksi rutin, sehingga auditor bebas melakukan pengujian.

Teknik pengujian ini disebut virtual karena pengujian dilakukan dalam lingkungan yang sama dengan sistem yang diuji, walaupun dalam partisi yang berbeda. Kelemahan teknik pengujian VTT adalah memerlukan perangkat keras yang memadai untuk dapat melakukan pengujiannya.

### 5. Embedded Auditability Modules

Sebelum pengujian dengan teknik ini dilaksanakan, auditor membuat suatu kriteria tertentu tentang data terkini yang ingin diujinya, dan memasukkan kriteria tersebut ke dalam modul yang dirancangnya. Perancangan dapat dilakukan oleh auditor sendiri ataupun dengan bantuan teknisi. Kemudian melalui proses *screening* dengan memanfaatkan program produksi rutin, dipilih oleh modul tersebut setiap transaksi yang memenuhi kriteria, dan disimpan dalam *file* tersendiri untuk dilakukan analisis lebih lanjut oleh auditor. Kriteria yang dipilih dapat berupa besarnya nilai transaksi sampai dengan jumlah tertentu dalam sistem pembelian dan sistem penjualan, pemberian suku bunga deposito di atas suku bunga yang berlaku bagi nasabah umum dalam sistem perbankan, pemberian diskon penjualan di atas ketentuan, dan sebagainya.

### 6. Auditors's Database Segments

Dalam teknik pengujian ini, segmen *database* ditambahkan ke *database* yang ada, semata-mata digunakan untuk keperluan auditor. Hal ini dilakukan dengan pertimbangan bahwa auditor, seperti halnya pemakai sistem yang lainnya, mempunyai hak dan keperluan untuk menetapkan elemen datanya sendiri. Sistem harus dirancang sedemikian rupa sehingga data auditor tidak digunakan oleh sistem yang reguler. Dengan kata lain, data yang ada di *database* auditor tersebut hanya *dummy* data.

### 7. Embedded Audit Data Collection Modules (EADCM)

Dalam teknik EADCM, auditor merancang suatu modul program dengan fungsi mengumpulkan data audit, untuk digunakan di masa yang akan datang. Jenis data, saat pengumpulan data, dan jumlah data yang dikumpulkan dapat bervariasi, tergantung dari jenis sistem, tujuan audit, dan tingkat kesulitan pengumpulan data yang *reliable.*

Terdapat beberapa keuntungan yang diperoleh dengan menggunakan teknik EADCM, yaitu data yang diperoleh dapat berupa data terkini (*current*), sehingga integritas data dapat lebih diandalkan. Bila direncanakan dan dirancang dengan baik, EADCM akan memberikan data audit yang dibutuhkan, sehingga akan menghemat waktu audit.

Data yang diperoleh dapat disimpan dalam jangka waktu yang lama tanpa tercampur dengan data lainnya, sehingga tidak mengganggu retensi *file*. Data yang diperoleh dapat berfungsi sebagai jejak audit (*audit trail*).

### 8. Extended Audit Records (EAR)

Teknik ini merupakan salah satu bentuk dari EADCM. Dalam teknik ini, informasi yang dikumpulkan adalah berupa data tambahan yang hanya diperlukan oleh auditor, sehingga disebut *extended*. Semula sistem ini berkaitan dengan *flat file* yang disimpan dalam *magnetic tape*. Sejalan dengan perkembangan ke sistem yang lebih canggih, *flat file* tidak digunakan lagi dalam *database*. Dalam sistem ini, informasi

tambahan dapat dikumpulkan dan disimpan dalam suatu *file* terpisah. Oleh karena itu, teknik ini akan lebih tepat disebut *audit reference rife of extended records*.

## 5.3 MEMPEROLEH PROGRAM KOMPUTER AUDIT

Ada tiga pendekatan yang dapat digunakan untuk mendapatkan program komputer yang cocok digunakan untuk mengevaluasi dan menguji *record*, yaitu program yang ditulis oleh klien sendiri, program yang ditulis oleh atau dengan pengawasan auditor, dan program audit yang digeneralisasi.

### 5.3.1 Program yang Ditulis Klien

Sering kali analisis yang diperlukan auditor juga bermanfaat bagi klien. Oleh karena itu, klien dapat menulis program komputer untuk diri sendiri atau menyediakan program tersebut jika auditor meminta analisis dan menggunakannya untuk keperluan internal.

### 5.3.2 Menulis Program Audit

Auditor menggunakan program audit yang dibuat sendiri sesuai dengan kebutuhan, tentunya dapat bekerja sama dengan *programmer* untuk membuatnya. Program audit komputer ini dibuat untuk melaksanakan aktivitas audit. Secara umum, ada lima aspek pengembangan program audit komputer.

- Menentukan tujuan dan prosedur audit yang diperlukan, dan menyiapkan dokumen yang mendefinisikan secara terperinci pemrosesan yang diperlukan.
- Mengembangkan bagan arus sistem yang mencakup semua *input*, *output*, dan langkah-langkah pemrosesan utama.
- Mengembangkan spesifikasi program dengan menggunakan bagan arus, tabel keputusan, dan/atau narasi yang menguraikan logika program serta langkah-langkah pemrosesan.
- Memberi kode, *debugging,* dan mengetes.
- Memproses dan mereviu hasil.

Berdasarkan aspek pengembangan program audit di atas, ada empat fase atau tahapan dalam menyusun atau membuat program tersebut, adalah sebagai berikut.

a. Fase tujuan, kelayakan (fisibilitas), dan perencanaan audit.
   - Mendefinisikan tujuan-tujuan audit.
   - Menyiapkan suatu program audit pendahuluan.
   - Mendapatkan informasi dari klien mengenai karakteristik komputer dan *file*, *layout record*, dan ketersediaan *file*.
   - Menentukan bahwa komputer dan *file* memang cocok dengan paket *software* audit.
   - Menyelesaikan prosedur audit dan menetapkan rencana kerja untuk fase-fase berikutnya.

b. Fase desain aplikasi.
   - Mendefinisikan ketentuan aplikasi *software* audit termasuk logika, kalkulasi, format laporan, dan bagaimana aplikasi itu dapat dikendalikan.
   - Mengembangkan suatu bagan arus keseluruhan dari aplikasi *software* audit yang direncanakan.

c. Fase pengodean dan pengetesan.
   - Memberi kode formulir-formulir spesifikasi *software* audit.
   - *Keypunch* formulir-formulir spesifikasi.
   - Mengetes aplikasi dengan menggunakan *file* tes atau bagian dari *file* klien yang sebenarnya.

d. Fase pemrosesan.
   - Memproses aplikasi dengan menggunakan *file* klien sebenamya yang diaudit.
   - Mereviu hasilnya untuk memastikan bahwa hasil itu memenuhi tujuan audit dan pengendalian.

Tujuan audit harus didefinisikan dengan jelas. Hal ini mencakup perencanaan program audit pendahuluan yang meliputi sebanyak mungkin perincian jika informasi yang tersedia memungkinkan. Jika tujuan audit telah dirumuskan, auditor harus mendapatkan informasi dari klien mengenai karakteristik komputer dan *file*. Informasi ini dapat mencakup hal-hal sebagai berikut.

- Produsen komputer dan nomor modelnya.
- Tipe unit pita dan *disk*.
- Peralatan *input-output* yang tersedia untuk digunakan dengan *sofware* audit.
- Alamat dari semua unit *input-output* yang digunakan dalam pemrosesan.
- Tipe dan nomor *release* dari sistem operasi.
- Tipe *file* yang digunakan: kartu, pita, *disk*.
- Tipe *record* data: panjangnya tetap atau variabel.
- Panjang (*length*) *record* data.
- Faktor *blocking*, jika *file* tersebut diblokir.
- Kepadatan (densitas) pita.
- Jumlah perkiraan *record* dalam *file*.
- Informasi label judul (*header label*) pada *file input*.

### 5.3.3 Program Audit yang Digeneralisasi

Program yang digeneralisasi melaksanakan fungsi audit komputer yang luas juga tersedia dan dapat digunakan hanya dengan sedikit latihan dan persiapan bagi para penggunanya. Pendekatan di antaranya adalah dengan menggunakan program yang dapat diterapkan untuk semua klien dalam suatu industri.

## 5.4 KEMAMPUAN SOFTWARE AUDIT

Fungsi-fungsi utama yang dapat dilaksanakan dengan menggunakan *software* audit komputer sebagai berikut.

1. *Create (membuat) file* kerja. Ini dilaksanakan dengan membaca *record* dari *file* yang diaudit dan membuat *file* baru yang meliputi *record* "kerja" dalam format yang dirumuskan auditor.
2. *Update (memperbarui) file* kerja dengan menggunakan data yang terkandung dalam *record* dari *file* klien yang kedua.
3. *Summarize (mengikhtisarkan) record* kerja, dengan memproduksi *record* ikhtisar yang berisikan total data yang terdapat dalam *field record* terperinci.
4. *Sort (menyortir) record* ke dalam urutan baru yang diperlukan untuk kalkulasi, pencetakan laporan, atau proses lainnya.
5. *Calculate (menghitung)* nilai tambahan yang akan dimasukkan ke dalam *record* kerja atau untuk mengetes nilai *file* yang ada.
6. *Generate (mengeluarkan) file* baru dalam format yang dirumuskan auditor.
7. *Print* laporan menurut format yang ditetapkan auditor.
8. *Select (memilih) record* tertentu untuk pemrosesan khusus, yang secara acak atau didasarkan pada tes nilai yang terkandung dalam *record* klien atau *record* kerja.

Selain fungsi utama di atas, program audit harus memiliki minimal tujuh aplikasi untuk tipe-tipe tugas dalam auditing.

1. ***Mengetes perkalian dan penjumlahan.*** Komputer dapat digunakan untuk melaksanakan penjumlahan dan perhitungan sederhana lainnya untuk mengetes kebenaran perkalian dan penjumlahan (*footing*).
2. ***Mengikhtisarkan data dan melaksanakan analisis yang berguna bagi auditor.*** Auditor sering kali perlu meminta data klien yang diikhtisarkan dengan pelbagai cara untuk analisis. Contohnya adalah penetapan umur piutang, penyiapan penggunaan tahunan, kebutuhan suku cadang dan persediaan, daftar semua saldo kredit dalam piutang dan semua saldo debit dalam utang, dan seterusnya.
3. ***Memeriksa kualitas record: kelengkapan, konsistensi, kondisi yang tidak sahih, dan seterusnya.*** Kualitas dalam *record* yang dapat terlihat (*visible*) akan jelas tampak begitu auditor menggunakannya dalam pemeriksaannya.
4. ***Memilih dan mencetak konfirmasi.*** Dengan menggunakan kriteria pemilihan yang dapat dikuantifikasi, komputer dapat memilih dan mencetak permintaan konfirmasi. Misalnya, satu kantor akuntan telah mendesain formulir multibagian yang dipersiapkan pada komputer.
5. ***Memilih dan mencetak sampel audit.*** Komputer dapat diprogram untuk memilih sampel audit dengan menggunakan angka acak (*random*) atau teknik pemilihan yang sistematis.

6. ***Membandingkan data yang sama yang disimpan dalam file-file terpisah untuk melihat kebenaran dan konsistensinya.*** Bila ada dua *record* terpisah atau lebih yang memiliki *field* data yang seharusnya sama, komputer dapat digunakan untuk mengetes kebenaran dan konsistensinya.
7. ***Membandingkan data audit dengan record perusahaan.*** Data audit seperti hasil perhitungan tes persediaan dapat dibandingkan dengan *record* persediaan yang menggunakan program komputer.

## 5.5 AUDIT PUSAT PDE DAN APLIKASI KOMPUTER

Ikatan Akuntan Indenesia (IAI) dalam Standar Profesional Akuntan Publik telah memasukkan enam standar audit. Keenam standar audit tersebut berkaitan dengan pengendalian internal di lingkungan PDE, teknik audit berbantuan komputer (TABK), audit dalam lingkungan, dan tiga standar yang berkaitan dengan lingkungan PDE, yaitu untuk komputer mikro, sistem *online*, dan sistem *database*.

Perbedaan prosedur audit antara audit konvensional dengan audit PDE sebagai pengaruh komputer terhadap auditing, yaitu akibat auditan mengolah datanya menggunakan komputer, maka menimbulkan beberapa hal yang sebelumnya tidak terjadi pada konvensional. Pengaruh komputer terhadap auditing mencakup:

1. Audit sekitar komputer (*audit around the computer*).
2. Audit melalui komputer (*audit through the computer*).
3. Audit dengan komputer (*audit with the computer*).

Hal tersebut menyiratkan adanya pengakuan dari auditor mengenai pengaruh komputer pada pengolahan data auditan. Adapun pengaruh tersebut antara lain sebagai berikut.

### 5.5.1 Perencanaan Audit

American Institute of Certified Public Accountants (AICPA) dalam SAS 48 menyebutkan bahwa auditor harus mempertimbangkan metode yang digunakan oleh auditan dalam mengolah data akuntansi, karena metode pengolahan data akan memengaruhi rancangan sistem akuntansi dan sifat prosedur pengendalian akuntansi. SAS 48 menyebutkan tentang perlunya auditor untuk menilai pengaruh PDE, sehingga auditor harus mempertimbangkan:

1. Cakupan penggunaan komputer dalam pengolahan data akuntansi.
2. Tingkat kompleksitas aplikasi komputer satuan usaha yang diaudit, termasuk penggunaan pengolahan data oleh pihak ketiga.
3. Struktur kegiatan pengolahan data.
4. Ketersediaan data, baik dalam bentuk *file* atau bukti fisik lainnya.
5. Penggunaan TABK untuk meningkatkan efisiensi pelaksanaan prosedur audit.

### 5.5.2 Penerapan Prosedur Pengujian Pengendalian dan Substantif

SAS 48 menyatakan bahwa metode yang digunakan oleh auditan dalam mengolah data akuntansi dapat memengaruhi sifat, waktu, dan luas prosedur audit. IAI secara eksplisit menyebutkan dampak pengolahan data elektronik ini pada penerapan prosedur pengujian pengendalian dan prosedur pengujian substantif.

### 5.5.3 Pengumpulan dan Evaluasi terhadap Bukti

*Ron Weber* menyatakan bahwa penggunaan komputer oleh auditan dalam proses bisnisnya bagi auditor menimbulkan pengaruh pada bagaimana bukti harus dikumpulkan dan dievaluasi. Mengumpulkan bukti mengenai keandalan sistem PDE adalah lebih kompleks, sehingga auditor harus memahami pengendalian internal di lingkungan PDE.

Perkembangan teknologi pengendalian berubah dengan cepat, sehingga auditor harus menyesuaikan terhadap perkembangan tersebut dalam mengumpulkan bukti mengenai keandalan pengendalian.

Dalam evaluasi bukti *Weber* menyebutkan:

- Semakin meningkatnya kerumitan sistem PDE dan teknologi pengendalian internal maka auditor juga akan menjadi lebih sulit untuk menilai keandalan sistem berdasarkan kekuatan dan kelemahan pengendalian sistem yang bersangkutan.
- Kesalahan PDE yang berulang-ulang menambah beban bagi auditor untuk memastikan bahwa pengendalian dalam satuan usaha sudah memadai untuk mengamankan aset, integritas data, efektivitas dan efisiensi sistem serta memastikan pengendalian yang ada benar-benar ada dan berfungsi.

### 5.5.4 Pengauditan Pusat PDE

Adapun langkah-langkah yang dilakukan dalam mengaudit di lingkungan PDE, antara lain meliputi: (a) Perencanaan audit, (b) Pemahaman terhadap lingkungan komputer, (c) Evaluasi terhadap pengendalian internal, (d) Pengujian kepatuhan dan pengujian substantif, dan (e) Penyelesaian audit.

#### A. Perencanaan Audit

Standar pekerjaan lapangan yang pertama dari SPAK menyatakan bahwa pekerjaan audit harus direncanakan dengan sebaik-baiknya. Perencanaan memungkinkan auditor dapat melaksanakan audit secara efisien dengan biaya yang memadai, serta memungkinkan bagi auditor untuk menghindari kesalahpahaman yang mungkin timbul dengan pihak-pihak yang diaudit.

AICPA memasukkan perencanaan ini dalam tahap penelaahan pendahuluan. Penelaahan ini bertujuan untuk memperoleh pemahaman mengenai sistem akuntansi berbasis elektronik dan non-elektronik melalui unsur-unsur berikut ini.

- Arus transaksi dan keluaran yang signifikan.
  Tujuannya adalah auditor dapat merancang dan menerapkan prosedur yang sesuai untuk menelaah dan menilai pengendalian akuntansi.
- Sejauh mana penggunaan komputer dalam aplikasi akuntansi.
  Agar dapat memahami sejauh mana PDE digunakan dalam aplikasi akuntansi, maka auditor harus mempertimbangkan:
  - Jumlah dan jenis transaksi yang diproses.
  - Nilai total rupiah setiap jenis transaksi.
  - Sifat dan sampai sejauh mana pengolahan menggunakan PDE, termasuk yang dilaksanakan oleh program komputer.
  - Pembagian arus transaksi antara aktivitas PDE dengan non-PDE.
  - Struktur dasar dari pengendalian akuntansi, baik pengendalian bagian PDE maupun pengendalian bagian pengguna.

Ada beberapa hal yang harus diperhatikan auditor pada saat mengaudit pusat PDE, adalah sebagai berikut.

1. Pengendalian yang ada.
2. Pembagian tanggung jawab terhadap pengendalian di dalam sistem antara bagian PDE dan non-PDE.
3. Hubungan antara pengendalian berdasarkan PDE maupun non PDE.
4. Sifat, sejauh mana dan tersedianya informasi yang memberikan jejak audit.
5. Metode yang digunakan untuk memperoleh pemahaman mengenai sistem akuntansi adalah dengan kuesioner dan wawancara, observasi, penelaahan terhadap dokumentansi; memonitor/mentrasir transaksi, kuesioner pengendalian serta daftar pengujian.

Secara umum penelaahan pendahuluan terdiri menjadi tiga tahapan, yaitu pengumpulan data umum, identifikasi terhadap aplikasi keuangan, dan penyiapan rencana pemeriksaan.

Pada tahapan pengumpulan data umum auditor bermaksud mengumpulkan informasi yang bersifat umum seperti struktur organisasi satuan usaha, bagan perkiraan yang ada, *hardware* dan *software* yang digunakan, termasuk bagan alur *(flowchart*), prosedur yang ada serta pengamanan fisik yang dilakukan. Berdasarkan informasi umum tersebut, seharusnya auditor dapat menentukan masalah-masalah penting yang berkaitan dengan pelaksanaan pekerjaan audit, seperti banyak waktu yang diperlukan, para personel dan kecakapan yang diperlukan untuk melaksanakan pekerjaan audit, serta kapan suatu pekerjaan audit harus dilaksanakan (penjadwalan).

Pada tahapan identifikasi terhadap aplikasi keuangan yang dapat ditentukan dengan mempertimbangkan banyak hal, seperti:

1. Keinginan dari pimpinan objek pemeriksaan, yang ditentukan dalam surat penugasan.
2. Kemungkinan terjadinya *potential error*.
3. Histori keuangan di masa lalu.

Setelah tahapan tersebut dilaksanakan, maka auditor dapat menyusun rencana audit antara lain meliputi lingkup audit, uraian mengenai prosedur dan pengendalian PDE yang ada, pengaruh kekuatan dan kelemahan pengendalian aplikasi yang ada, pengujian kepatuhan yang mungkin dilakukan. Dalam perencanaan ini auditor dapat menggunakan komputer untuk melakukan:

1. Perancangan audit program.
2. Pengembangan kuesioner pengendalian internal.
3. Pelaksanaan analisis terhadap risiko satuan usaha yang sedang diaudit.
4. Pelaksanaan analisis atas data keuangan.
5. Penjadwalan pekerjaan yang akan dilakukan dan biayanya.

## B. Memahami Lingkungan Komputer

Untuk memahami lingkungan komputer, auditor harus memiliki pengetahuan mengenai PDE. Tingkat pengetahuan proses-proses PDE yang dibutuhkan auditor berbeda tergantung pada sistem PDE, peranan masing-masing auditor, dan sifat dari prosedur audit yang dilakukannya.

Auditor harus mengumpulkan informasi tentang lingkungan PDE yang relevan dengan perencanaan audit, termasuk informasi berikut ini.

- Bagaimana fungsi sistem informasi dikelola dan lingkup atau distribusi pengolahan komputer dalam keseluruhan satuan usaha.
- Sistem *batch*, data dikumpulkan berdasarkan kriteria dan diproses bersama secara periodik.
- Sistem *real-time*, data di-*input* melalui terminal pada saat itu juga akan dicek oleh komputer dan diproses serta disimpan pada *file* yang relevan.
- Sistem *online*, data di-*input* melalui terminal pada saat itu juga dicek oleh komputer, tetapi data disimpan sementara dan di-*update* di kemudian hari.
- *Software* dan *hardware* yang digunakan.
- Sifat pengolahan data dan kebijakan penyimpanan data.
- Implementasi aplikasi baru atau pemeliharaan aplikasi yang ada.

Auditor harus memahami bagaimana sistem komputer satuan usaha yang akan diaudit menghasilkan data, yaitu mulai dari penyiapan dokumen sumber sampai pendistribusian dan penggunaan *output*. Untuk aplikasi akuntansi, maka auditor harus mempertimbangkan (AICPA) dalam:

- Memperhitungkan jenis-jenis kesalahan dan ketidakteraturan yang mungkin terjadi.
- Menentukan prosedur-prosedur pengendalian akuntansi yang mencegah atau mendeteksi kesalahan dan ketidakteraturan.
- Mengakses efektivitas pengendalian akuntansi PDE dan non-PDE

## C. Evaluasi atas Pengendalian Internal

Tujuan evaluasi atas pengendalian internal adalah sebagai berikut.

- Untuk mengidentifikasi jenis kesalahan yang mungkin timbul atau kelemahan yang mungkin ada sehingga auditor dapat merencanakan prosedur audit secara memadai.

- Untuk mempertimbangkan faktor-faktor yang memengaruhi risiko penyajian laporan keuangan yang secara material salah.
- Untuk merancang pengujian substantif.

Untuk memperoleh informasi yang dapat digunakan oleh auditor dalam mengevaluasi pengendalian internal, yaitu: penelaahan dokumentasi, *interview* dengan personel PDE, dan departemen pengguna, serta melakukan pengamatan terhadap praktik yang dilakukan dalam satuan usaha yang akan diaudit.

### D. Pengujian Kepatuhan dan Pengujian Substantif

1. ***Pengujian Kepatuhan***

   Tujuan dari pengujian kepatuhan adalah untuk menentukan apakah sistem pengendalian internal yang ada telah berfungsi sebagaimana dikehendaki oleh manajemen.

   Pada tahapan ini auditor melakukan validasi terhadap keandalan dan kelemahan pengendalian internal yang ditemukan dalam tahap evaluasi tersebut. Apabila pada tahapan evaluasi pengendalian internal auditor memercayai, maka selanjutnya auditor melakukan pengujian kepatuhan. Apabila auditor tidak memercayai pengendalian internal, maka auditor akan langsung melakukan pengujian substantif tanpa melalui pengujian kepatuhan.

   Pengujian substantif tetap akan dilaksanakan setelah pengujian kepatuhan, tetapi lingkupnya berbeda. Apabila setelah pengujian kepatuhan tersebut auditor juga tidak dapat memercayai pengendalian *user*, maka auditor melakukan pengujian substantif.

2. ***Pengujian Substantif***

   Salah satu prosedur audit yang dirancang untuk menguji kesalahan atau ketidakaturan suatu nilai uang yang secara langsung memengaruhi kewajaran saldo dalam laporan keuangan.

   Pengujian ini untuk memvalidasi bahwa suatu transaksi telah diotorisasi secara memadai, disertai bukti pendukung, dicatat, dan pos-pos yang dicatat tersebut merupakan hasil dari transaksi yang ada otorisasinya.

   Tujuan pengujian ini untuk menentukan apakah transaksi bisnis suatu usaha telah diotorisasi, dicatat, dan dibukukan ke dalam jurnal, buku tambahan dan buku besar secara benar.

## 5.5.5 Mengaudit Aplikasi EDP

Berbeda dengan mengaudit lingkungan sistem informasi yang bersifat umum, maka mengaudit aplikasi EDP di sini adalah mengaudit dari program aplikasi yang digunakan pada lingkungan sistem informasi yang berbasis komputer. Ada 7 langkah mengaudit aplikasi EDP, antara lain:

1. *Review* pendahuluan.

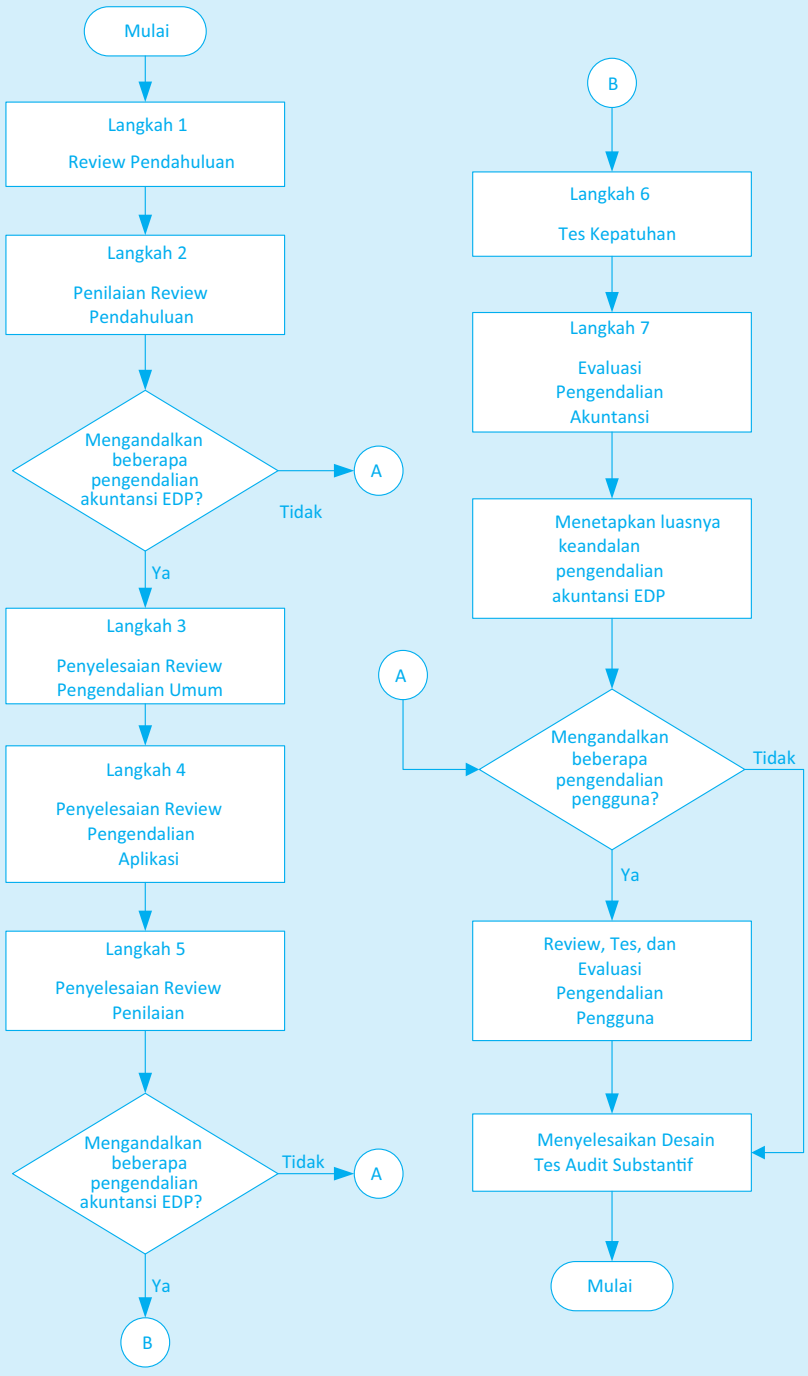
   ***Tujuan:***

   - Memahami sistem akuntansi termasuk segmen EDP dan non-EDP.
   - Arus transaksi dan signifikansi *output.*

- Luasnya penggunaan EDP dalam aplikasi akuntansi yang signifikan.
- Struktur dasar pengendalian EDP maupun pengguna.

***Metode:***

Penyelidikan dan pembahasan; Observasi; *Review* dokumentasi: penelusuran transaksi, kuesioner, dan daftar cek pengendalian.




Gambar 5.1
Langkah-Langkah Audit pada Aplikasi Komputer

2. Penilaian *review* pendahuluan.
   ***Tujuan:***
   - Menilai signifikansi pengendalian akuntansi EDP dan non-EDP.
   - Menetapkan luasnya *review* tambahan dalam EDP.

   ***Metode:***
   Pertimbangan (*judgment*).

3. Penyelesaian *review* pengendalian umum.
   ***Tujuan:***
   - Mengidentifikasi pengendalian umum yang dapat direncanakan pengendaliannya dan menentukan operasinya.
   - Menentukan pengaruh kekuatan dan kelemahan terhadap pengendalian aplikasi.
   - Mempertimbangkan uji kepatuhan yang dapat dilaksanakan.

   ***Metode:***
   - Pengujian dokumentasi yang terperinci.
   - Mewancarai auditor internal; personel EDP, dan departemen pengguna.
   - Observasi operasi pengendalian umum.

4. Penyelesaian *review* pengendalian aplikasi.
   ***Tujuan:***
   - Mengidentifikasi pengendalian aplikasi yang dapat direncanakan pengendaliannya dan menentukan bagaimana pengendalian tersebut beroperasi.
   - Mempertimbangkan uji kepatuhan yang dapat dilaksanakan.
   - Mempertimbangkan pengaruh potensial dari kekuatan dan kelemahan yang diidentifikasi terhadap uji kepatuhan.

   ***Metode:***
   - Pengujian dokumentasi yang terperinci;
   - Mewancarai auditor internal; personel EDP, dan departemen pengguna;
   - Observasi operasi pengendalian aplikasi.

5. Penyelesaian dan penilaian *review*.
   ***Tujuan:***
   - Untuk setiap aplikasi akuntansi yang signifikan.
   - Mempertimbangkan tipe-tipe kesalahan atau ketidakwajaran yang mungkin terjadi.
   - Menentukan prosedur pengendalian akuntansi yang dapat mencegah atau mendeteksi kesalahan dan ketidakwajaran tersebut.
   - Mengevaluasi efektivitas pengendalian akuntansi EDP dan non-EDP.

   ***Metode:***
   Pertimbangan.

6. Uji kepatuhan.
   ***Tujuan:***
   - Menentukan apakah prosedur pengendalian yang perlu telah digariskan dan diikuti dengan baik.
   - Mencari keyakinan yang cukup bahwa pengendalian benar-benar berfungsi dengan baik.
   - Mempertimbangkan dan, bila perlu, mendokumentasikan kapan, bagaimana, dan oleh siapa pengendalian tersebut akan dilaksanakan.

7. Evaluasi pengendalian akuntansi.
   ***Tujuan:***
   - Mempertimbangkan tipe-tipe kesalahan atau ketidakwajaran yang mungkin terjadi.
   - Menentukan prosedur pengendalian akuntansi yang dapat mencegah atau mendeteksi kesalahan dan ketidakwajaran tersebut.
   - Menentukan apakah prosedur pengendalian yang perlu telah digariskan dan diikuti dengan baik.
   - Mengevaluasi kelemahan dan menilai pengaruhnya terhadap sifat, waktu (*timing*), dan luasnya prosedur audit yang digunakan.

   ***Metode:***
   Pertimbangan.

## SOAL DAN STUDI KASUS

1. Apa yang dimaksud dengan pengujian substantif?
2. Apa perbedaan dari pengujian substantif dengan pengujian kepatuhan?
3. Apa tujuan utama dari pengujian substantif?
4. Jelaskan! Apa perbedaan dari alat pengujian dan teknik pengujian?
5. Sebutkan hal-hal yang perlu dipertimbangkan oleh auditor sebelum melakukan pengujian dengan menggunakan komputer!
6. Sebutkan delapan alat dan teknik pengujian substantif!
7. Bagaimana cara memperoleh program komputer untuk audit?
8. Pada saat ini, sudah banyak program audit dalam perkembangannya. Sebutkan delapan fungsi utama dari program komputer!
9. Bagaimana pengaruh komputer setelah penyelenggaraan uji substantif?

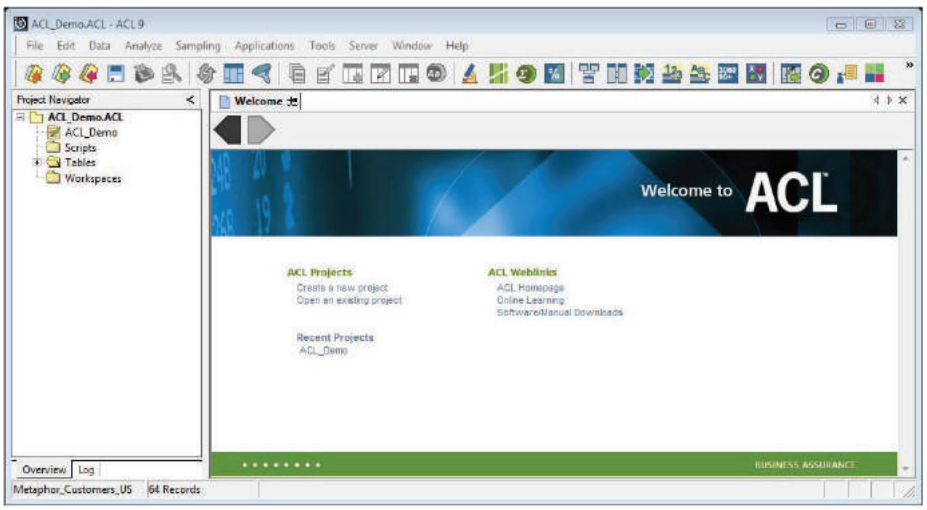
# BAB 6

# MENGGUNAKAN APLIKASI ACL FOR WINDOWS

Setelah mempelajari bab ini, Anda diharapkan mampu:

- Mengetahui kegunaan umum ACL for Windows.
- Mengetahui cara mengakses dan men-*download* data untuk diolah dengan ACL.
- Memahami bagaimana komputer menyajikan data.
- Mengetahui jenis-jenis data yang dapat dibaca oleh ACL.

Audit Command Language (ACL) merupakan salah satu jenis *audit software* yang termasuk dalam kategori Generalized Audit Software (GAS). Seperti halnya program GAS yang lainnya, ACL hanya dapat digunakan untuk mengumpulkan dan mengevaluasi bukti yang dihasilkan dari pemrosesan transaksi perusahaan sehingga ACL lebih cenderung digunakan untuk menilai *post transactions* daripada *current transactions*. Oleh karena itu, ACL lebih berorientasi pada pendekatan *auditing around the computer* dibanding *auditing through the computer*. Namun demikian, ACL memiliki manfaat yang cukup besar dalam membantu pelaksanaan tugas auditor.




Gambar 6.1
Langkah-langkah Mengaudit pada Aplikasi Komputer

ACL for Windows dirancang khusus untuk menganalisis data dan menghasilkan laporan audit untuk pengguna biasa (*common/nontechnical users*) atau pengguna ahli (*expert users*). ACL dapat membantu menyiapkan laporan audit secara mudah dan interaktif, sehingga pekerjaan *auditing* akan jauh lebih cepat daripada proses *auditing* secara manual yang memerlukan waktu sampai berjam-jam bahkan sampai berhari-hari. Audit terhadap data keuangan hanya secara *sampling* saja, tetapi dengan ACL dapat dilakukan pada keseluruhan dokumen yang diizinkan diakses sehingga bersifat komprehensif.

Dengan beberapa kemampuan ACL, analisis data akan lebih efisien dan lebih meyakinkan. Berikut ini beberapa kemampuan ACL.

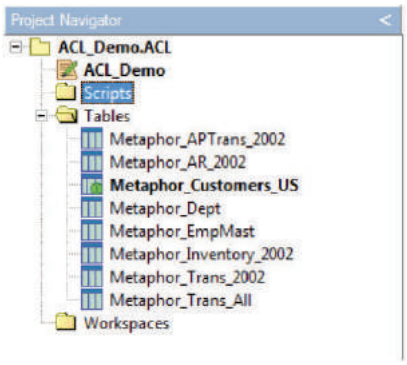
- ***Mudah dalam penggunaan.*** ACL for Windows, sesuai dengan namanya, adalah perangkat lunak (*software*) berbasis Windows, di mana sistem operasi Windows telah dikenal bersifat mudah digunakan (*user friendly*). Kemudahan ini ditunjukkan dengan pengguna (*user*) hanya mengeklik pada gambar tertentu (*icon*) untuk melakukan suatu pekerjaan, dan didukung pula dengan fasilitas Wizard untuk mendefinisikan data yang akan dianalisis.

- ***Built-in audit dan analisis data secara fungsional.*** ACL for Windows didukung dengan kemampuan analisis untuk keperluan audit/pemeriksaan, seperti analisis statistik, menghitung total, stratifikasi, sortir, indeks, dan lain-lain.
- ***Kemampuan menangani ukuran file yang tidak terbatas.*** ACL for Windows mampu menangani berbagai jenis *file* dengan ukuran *file* yang tidak terbatas.
- ***Kemampuan untuk membaca berbagai macam tipe data.*** ACL for Windows dapat membaca *file* yang berasal dari berbagai format, antara lain: Flat sequential, dBase (DBF), Text (TXT), Delimited, Print, ODBC (Microsoft Access Database, Oracle), Tape (½ inch 9 – track tapes, IBM 3480 cartridges, 8 mm tape dan 4 mm DAT).
- ***Kemampuan untuk mengekspor hasil audit ke berbagai macam format data.*** Berbagai format antara lain: Plain Text (TXT), dBase III (DBF), Delimit (DEL), Excel (XLS), Lotus (WKS), Word (DOC), dan WordPerfec (WP).
- ***Pembuatan laporan berkualitas tinggi.*** ACL for Windows memiliki fasilitas lengkap untuk keperluan pembuatan laporan.

## 6.1 KONSEP DASAR ACL

Untuk kepentingan pemeriksaan, pengguna harus membuat *file* proyek (*project file*) untuk setiap klien dan objek pemeriksaan. Hal ini untuk membedakan antara satu pemeriksaan dengan pemeriksaan lainnya, terutama jika unit usaha yang diperiksa sangat kompleks.

*File* proyek secara fisik sama dengan *folder*, untuk mengelola dokumen dan data setiap proyek agar tidak bercampur baur dengan yang lainnya.




Gambar 6.2
Isi dan Struktur Proyek ACL

**a. Table Layout**

Berisi semua struktur data asli dan yang dibuat oleh pemeriksa (*virtual fields* dan *expression*). Satu proyek ACL dapat memiliki beberapa tabel (*table*), di mana satu *table* dapat terhubung dengan berbagai *file* data sumber. Misalnya, *table* Inventori yang berisi struktur data inventori dapat dihubungkan dengan/digunakan oleh

*file* data Inventori_JAN, Inventori_FEB, Inventori_MAR, dan seterusnya atau Inventori_BDG, Inventori_SMG, Inventori_SBY, dan sebagainya. **File Input Definition** penting dalam proses audit, dalam tahapan ini pemeriksa perlu kerja sama dengan bagian administrasi *database* atau manajer IT klien untuk memperoleh informasi tentang *database layout*.

**b. View**
Digunakan untuk menampilkan seluruh atau sebagian data dan menghasilkan laporan sementara (*ad hoc reports*). Kita dapat menciptakan berbagai macam *ad hoc reports* dari satu *file* data.

**c. Scripts**
Perintah ACL dapat dikumpulkan dalam suatu *file batch* dan dijalankan berurutan dari satu *file* data.

**d. Workspace**
Untuk menyimpan formula/rumus yang dapat digunakan lagi pada beberapa *file*/*table* yang berbeda.

**e. Log Files**
Untuk menyimpan/mencatat semua pekerjaan (perintah dan hasilnya). Hal ini akan membantu dalam penyusunan kertas kerja audit dan *review* pekerjaan audit.

## 6.2 AKSES DAN DOWNLOAD DATA

ACL for Windows dapat bekerja menggunakan *database* relasional modern, selain menggunakan sistem penyimpanan data secara tradisional. Pada sistem *legacy*, untuk membuat dan memproses data tanpa menggunakan program, sedangkan ACL for Windows memiliki kemampuan untuk mengakses data.

ACL for Windows dapat mengakses data dalam berbagai macam format dan pada berbagai macam tipe media penyimpanan. ACL for Windows mampu menguji *output* atas suatu aplikasi di mana data yang digunakan kurang meyakinkan, atau mungkin aplikasi tersebut tidak berjalan dengan benar. ACL for Windows dapat digunakan untuk keperluan *View*, *Explore*, dan menganalisis seluruh data serta membuat laporan atas hasilnya.

### 6.2.1 Bagaimana Komputer Menyajikan Data?

Pada bentuk yang paling sederhana, setiap karakter dalam data disimpan dalam bentuk *byte*. Nilai dari *byte* menentukan karakter mana yang diwakilinya. Ada 2 skema utama pengodean yang memetakan nilai atas *byte* pada setiap karakter tertentu, yaitu:

- EBCDIC (Extended Binary Coded Decimal Interchange Code).
  Format ini umumnya ditemukan hanya pada komputer *mainframe* dan *mid-range* IBM.

- ASCII (American Standard Code for Information Interchange).
  Format ini biasanya ditemukan pada hampir semua *personal computer* (PC) dan beberapa komputer *mid-range* dan *mainframe*.

### 6.2.2 Bagaimana Mengakses File Data?

Langkah-langkah dalam memperoleh *file* data untuk dianalisis adalah sebagai berikut.

#### A. Identifikasi Sumber File Data

*File* data yang tersimpan pada komputer *mini* atau *mainframe*, harus di-*download* terlebih dahulu sehingga bisa diakses oleh komputer PC. Metode umum yang sering digunakan untuk keperluan tersebut adalah dengan menghubungkan komputer PC atau LAN (Local Area Network) ke *mainframe*, dengan menggunakan *terminal emulation*, *tape,* atau *cartridges*.

Kebanyakan produsen komputer *mini* dan *mainframe* menawarkan sarana untuk menghubungkan PC ke produk mereka, sehingga memungkinkan dilakukan transfer *file* data. Data tidak perlu diubah ke format ASCII untuk keperluan *download* ini, karena ACL for Windows bersifat *compatible* dengan sebagian besar tipe data dan seharusnya mampu membaca data apa pun. Namun, sebelumnya harus dipastikan bahwa pengguna ACL for Windows memiliki *harddisk space* yang cukup pada PC untuk menyimpan *file* yang akan di-*download*. Dalam beberapa kasus, diperlukan jumlah *space* tertentu.

#### B. Bekerja sama dengan Departemen Sistem Informasi

Ketika akan melakukan *download file* data dari komputer mini atau *mainframe*, sebelumnya harus dilakukan perencanaan untuk mengidentifikasi di mana data tersimpan dan dalam format apa. Langkah pertama adalah berbicara langsung dengan departemen komputer di perusahaan dan meminta *layout file* atas suatu sistem, misalnya: definisi *record*, *data dictionaries*, *schemas*, dan lain-lain yang pasti berisi daftar *field* pada suatu *file*.

Langkah berikutnya adalah melakukan *review* atas data (dengan atau tanpa *layout file*) untuk memastikan bahwa seluruh *field* berada pada suatu sistem. Hal ini bisa dilakukan dengan menguji data yang tersedia ke layar monitor atau dicetak terlebih dahulu. Sering kali suatu sistem memiliki *field* tertentu, tetapi pada kenyataannya, *field* tersebut sebenarnya tidak digunakan. Selain itu, sering terdapat beberapa kode dan konvensi yang harus dipahami oleh pengguna ACL for Windows.

#### C. Mempersiapkan File Data

Sebelum melakukan *download* atas data, perlu dipersiapkan dahulu datanya. Jika seluruh data ada pada *file* tertentu dan mempunyai format tertentu yang dapat dibaca langsung oleh ACL for Windows, maka transfer bisa langsung dilakukan dalam bentuk *native state* ke PC. Mungkin tidak perlu mentransfer seluruh data untuk *file*

yang berukuran besar. Sebaiknya, minta bantuan dari pihak departemen komputer untuk memberikan *copy flat file* atas suatu *database*.

*File output report* dalam bentuk elektronik bisa diakses oleh ACL for Windows, karena hampir semua *software* komputer mampu menghasilkan laporan (*report)*. ACL for Windows mampu membaca informasi yang disimpan dalam bentuk laporan tercetak. Aplikasi ini sangat berguna ketika Anda ingin mengakses data yang tersimpan dalam format *database* yang rumit.

Langkah pertama dalam memproses laporan sebagai data adalah dengan cara melakukan *capture* informasi ke dalam disket. Dalam kebanyakan lingkungan (*environment*) komputer mini dan *mainframe*, hal tersebut bisa langsung dilakukan, karena umumnya *file* akan di-*spooled* sebelum dicetak. Anda tidak perlu mencetak *file* terlebih dahulu, cukup mengopi *spool file* sebelum dihapus oleh sistem. Jika perlu men-*download spooled report file*, sekali lagi tanyakan kepada departemen komputer untuk mengopi *file* yang diperlukan ke disket, untuk selanjutnya dilakukan *download* atas *file* tersebut.

### D. Mengunduh File Data ke PC, Server Jaringan, atau Tape

Ketika mengunduh (*download*) data dari komputer mini atau *mainframe* melalui *terminal emulation*, PC akan mengemulasi terminal pada komputer *mainframe* atau mini melalui *software* untuk *terminal emulation*. Paket program seperti *IBM's Personal Communications* dan *PC 3270*, *PC Support/400*, *IRMA*, dan *Rumba* menyediakan perangkat *terminal emulation* untuk PC. Beberapa *software* dalam melakukan *download* akan mencoba mengonversi data dari komputer mini atau *mainframe*, seperti EBCDIC ke salah satu yang digunakan pada PC, misalnya ASCII. Sedangkan paket lainnya sering mengasumsikan seluruh data sebagai *text* dan konsekuensinya beberapa angka akan rusak. Dalam kasus ini, *download* yang terbaik dilakukan adalah tanpa *convertion options* (biasanya disebut *binary transfers*).

## 6.3 UTILITAS KONVERSI ACL

ACL for Windows berisi utilitas konversi yang memungkinkan untuk mengonversi beberapa tipe *file* tertentu untuk bisa digunakan oleh ACL.

### 6.3.1 Tipe File Data yang Dapat Dibaca ACL

ACL for Windows mampu membaca tipe *file* data berikut.

**a. Flat sequential**

*Flat sequential file* data berisi baris atas *consecutive* data, yang diatur satu per satu setelah yang lainnya. Sama dengan baris atas informasi yang dibagi menjadi bagian-bagian seperti pada buku telepon, *flat sequential file* memiliki baris data yang dibagi menjadi beberapa *field*. Misalnya, satu *field* bernama nama akhir, dan *field* lain bernama nama awal, dan lain-lain.

**b. Dbase**

ACL for Windows secara otomatis dapat mendeteksi, menganalisis, dan kemudian membuat suatu format *dBASE file*. Hal ini berlaku juga untuk *dBASE file* yang dibuat dengan *dBASE compatible products*, seperti *FoxPro*, *Visual FoxPro*, dan *Clipper*. Catatan: ACL tidak mampu membaca *associated file*, semacam Index atau *file* Memo.

Ketika *dBASE file* dibuka, ACL menampilkan pesan *"File is likely a dBASE file. Create fields now?" Click [OK]* untuk membuat *input file definition*. ACL akan memberitahukan bahwa *field definition* sudah dibuat secara otomatis. Data selanjutnya bisa diproses seperti *file* yang lainnya.

**c. Text**

*File* data berupa *text* berisi hanya karakter yang bisa dicetak, semacam huruf dari a sampai z, angka 1 sampai 9 dan *punctuation* (sebagian besar tombol pada *keyboard*). *Text file* bisa/tidak berupa *print file*.

**d. Delimited**

Kebanyakan *file* data berisi *field* yang tidak memiliki posisi tetap dalam sebuah *record*. *File* yang setiap *field*-nya dipisahkan dengan yang lain dengan karakter pemisah *field* disebut *delimited files*. ACL mendeteksi dan kemudian meminta pengguna untuk mengonversi *delimited file* tersebut.

**e. Print Files**

*Print file* adalah *text file* dalam bentuk laporan tercetak. Format ini mungkin berisi baris nondata semacam *header*, *subheader*, baris kosong, dan baris total di mana ACL mampu memfilter pada saat membaca *print file*.

**f. ODBC**

ODBC adalah singkatan dari Open Database Connectivity. ODBC merupakan teknologi API (Application Programming Interface) standar yang memungkinkan aplikasi mengakses *multiple database* dari pihak ketiga. Developer dapat membuat aplikasi ODBC *enabled* (juga dikenal dengan nama aplikasi ODBC *compliant* atau ODBC *client*). Mereka juga bisa menuliskan ODBC *drivers* untuk sistem manajemen *database* tertentu (*data sources*).

Teknologi ODBC memberikan para pengembang (*developer*) dan pengguna sebuah alat penting untuk mengakses sumber data (*data source*) yang beraneka ragam. Sebelum ada ODBC, aplikasi harus menggunakan antarmuka *proprietary* khusus atas *data source*. Aplikasi ini sulit untuk dibuat, sulit dalam pemeliharaan, dan rumit untuk dikembangkan.

ODBC meningkatkan *application portability* dengan menghilangkan kebutuhan pemanggilan spesifik *data source*. ODBC memungkinkan aplikasi agar dapat dalam waktu bersamaan dengan mengakses, *view*, dan modifikasi data dari *data sources*, tanpa memerhatikan bahwa *source* tersebut berupa *hierarchical object-oriented*, ISAM atau *plain text*.

**g. Tape**

ACL for Windows dengan mudah mengakses dan membaca data dari *reel tape* atau *cartridges*. Mengakses suatu *file* pada *tape* hampir sama dengan memproses *file* dengan *disk-based file*. ACL langsung membaca data yang belum diterjemahkan (jangan gunakan *backup copy*), dan data akan tetap berada dalam *tape*.

ACL dapat membaca data dari hampir semua tipe *tape*, termasuk ½ inch 9-track tapes, IBM 3480 cartridges, 8 mm tape, dan 4 mm DAT. Untuk ACL versi PC harus dihubungkan dengan *drive tape* yang sesuai dengan PC untuk bisa menggunakan metode ini.

## 6.4 SPESIFIKASI SISTEM ACL

ACL for Windows versi 9 dapat di-*install* pada sebuah PC *stand alone* atau jaringan (Local Area Network) dengan spesifikasi sistem yang dibutuhkan sebagai berikut.

- Windows 95/98 atau Windows NT/2000
  - *Memory* (RAM) minimal 8 MB dianjurkan 16 MB untuk *Windows 95*
  - *Memory* (RAM) minimal 24 MB dianjurkan 32 MB untuk *Windows 98/NT*
- Minimal 10 MB ruang kosong pada *harddisk* untuk menyimpan program *ACL*.

## 6.5 MEMBUAT DOKUMEN

Dokumen ACL digunakan untuk melihat tampilan dan mencatat semua aktivitas Anda dalam ACL. Dokumen ACL berupa *file* dengan extension .ACL. Data *file* merupakan *file* yang berisi data dengan format fil, dbf, txt, prn atau csv. Penamaan *file* pada ACL dapat menggunakan 32 karakter termasuk spasi.

### 6.5.1 Memulai ACL

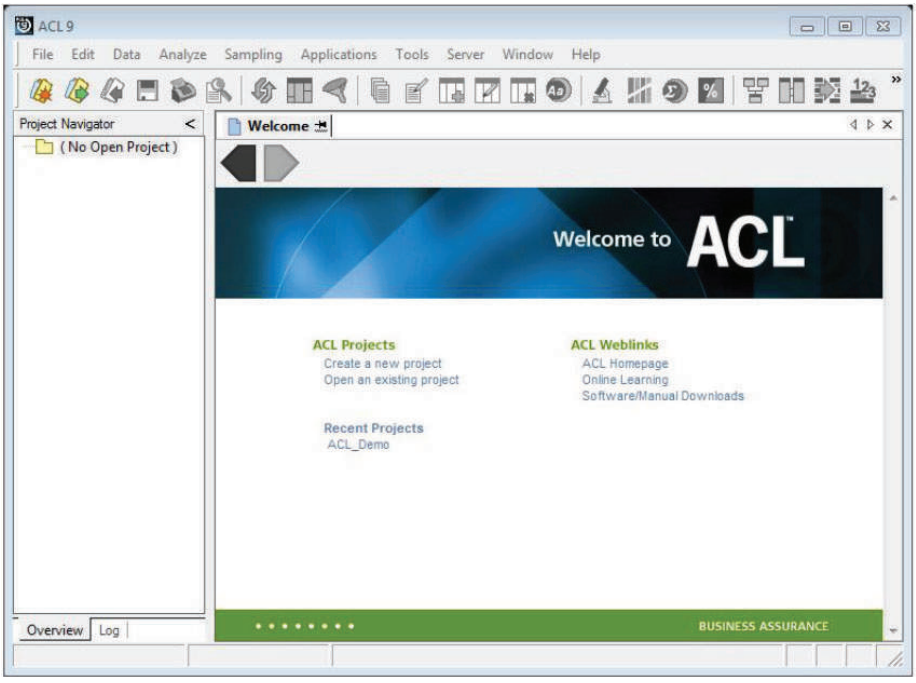
ACL menyimpan semua informasi mengenai data yang digunakan ke dalam *file* yang disebut dengan dokumen. Data yang diolah oleh komputer dan informasi yang sedang dikerjakan disimpan dalam sebuah *file* yang disebut *file* data.

Untuk dapat bekerja dengan ACL dibutuhkan sebuah dokumen. Dokumen dapat dibuat dari dokumen baru atau dari dokumen yang pernah dibuat. Untuk lebih jelasnya, ikuti langkah-langkah berikut untuk membuat dokumen baru.

Langkah-langkah untuk mulai bekerja dengan ACL.

1. Pilih dan tekan **Start** pada tampilan *Windows*;
2. Pilih menu **Program**;
3. Pada menu **Program – ACL Desktop Education Edition**;

Pada layar akan tampil **ACL Application Screen** seperti pada gambar berikut.

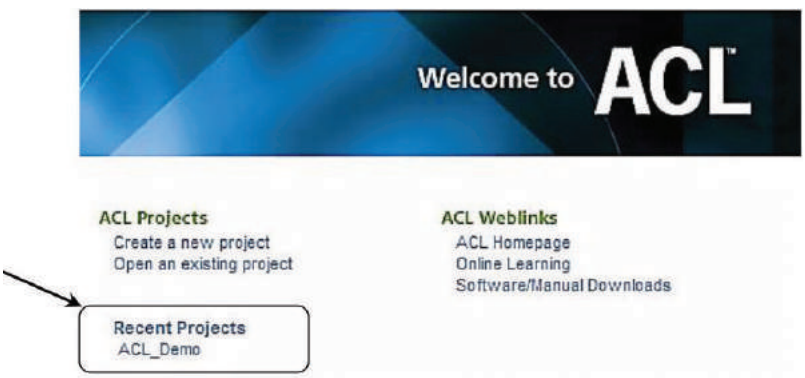



Gambar 6.3
Layar Aplikasi ACL (ACL Application Screen)

### 6.5.2 Memulai dengan Project

*File* proyek ACL memiliki *extension* **.acl** yang berisi Table Layout (struktur/format data), View, Batches, Workspace, dan Log.

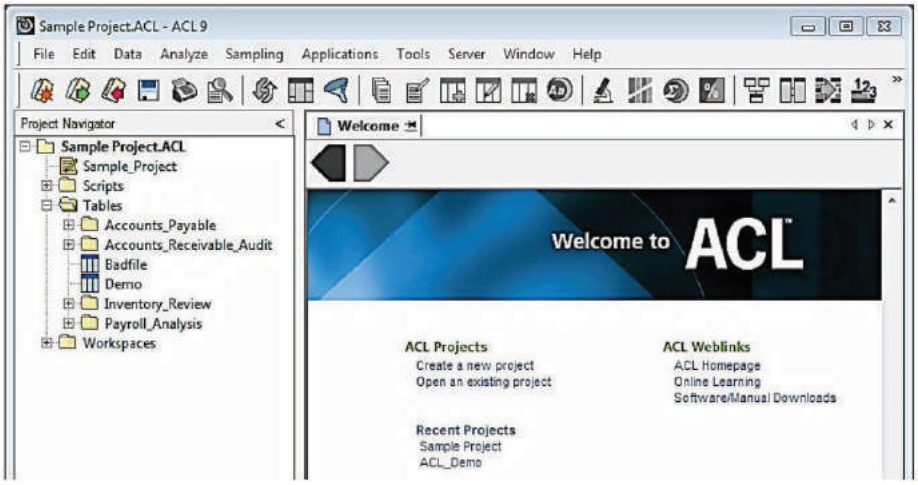
**a. Membuka proyek yang sudah ada.**




Gambar 6.4
Layar Utama Aplikasi ACL

Pada layar utama ACL, di bawah teks **ACL Projects**, klik teks **Open an existing project** atau

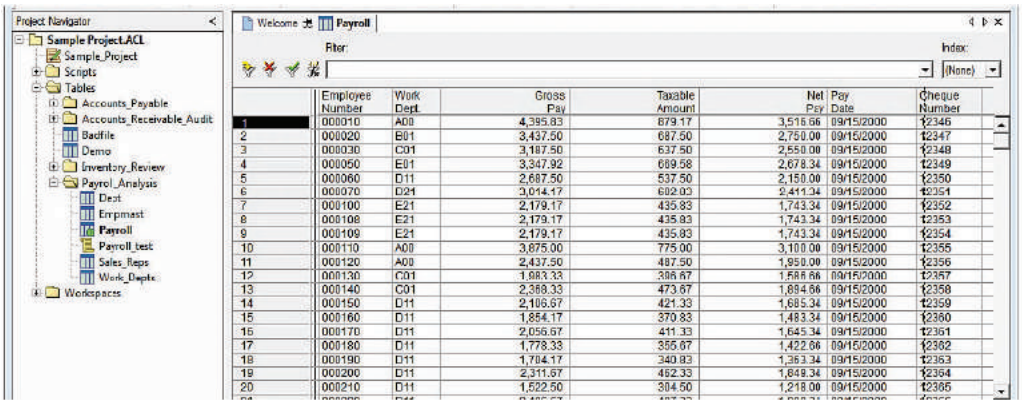
1. Pada menu **File** – klik perintah **Open Project** (**Ctrl + O**).
2. Kemudian pilih salah satu *file project* yang akan dibuka (misalkan, Sample Project).
3. ACL akan menampilkan *project* seperti berikut ini.




Gambar 6.5
Project Navigator dengan Proyek Sample Project ACL

Layar utama dibagi menjadi dua bagian, bagian kiri adalah **Project Navigator** (seperti *Windows Explorer*), untuk memilih *file-file* kerja yang ada dalam *project*, sedangkan pada bagian kanan adalah untuk menampilkan data dan hasil analisis.

Klik tanda **+** pada kiri **Payroll Analysis** untuk menampilkan *table* yang ada, kemudian klik 2 kali pada *table* yang akan ditampilkan (misalkan **Payroll**).




| | Employee Number | Work Dept | Gross Pay | Taxable Amount | Net Pay | Pay Date | Cheque Number |
|---|---|---|---|---|---|---|---|
| 1 | 000010 | A00 | 4,395.83 | 879.17 | 3,516.66 | 09/15/2000 | 12346 |
| 2 | 000020 | B01 | 3,437.50 | 687.50 | 2,750.00 | 09/15/2000 | 12347 |
| 3 | 000030 | C01 | 3,187.50 | 637.50 | 2,550.00 | 09/15/2000 | 12348 |
| 4 | 000050 | E01 | 3,347.92 | 669.58 | 2,678.34 | 09/15/2000 | 12349 |
| 5 | 000060 | D11 | 2,687.50 | 537.50 | 2,150.00 | 09/15/2000 | 12350 |
| 6 | 000070 | D21 | 3,014.17 | 602.83 | 2,411.34 | 09/15/2000 | 12351 |
| 7 | 000100 | E21 | 2,179.17 | 435.83 | 1,743.34 | 09/15/2000 | 12352 |
| 8 | 000108 | E21 | 2,179.17 | 435.83 | 1,743.34 | 09/15/2000 | 12353 |
| 9 | 000109 | E21 | 2,179.17 | 435.83 | 1,743.34 | 09/15/2000 | 12354 |
| 10 | 000110 | A00 | 3,875.00 | 775.00 | 3,100.00 | 09/15/2000 | 12355 |
| 11 | 000120 | A00 | 2,437.50 | 487.50 | 1,950.00 | 09/15/2000 | 12356 |
| 12 | 000130 | C01 | 1,983.33 | 396.67 | 1,586.66 | 09/15/2000 | 12357 |
| 13 | 000140 | C01 | 2,368.33 | 473.67 | 1,894.66 | 09/15/2000 | 12358 |
| 14 | 000150 | D11 | 2,106.67 | 421.33 | 1,685.34 | 09/15/2000 | 12359 |
| 15 | 000160 | D11 | 1,854.17 | 370.83 | 1,483.34 | 09/15/2000 | 12360 |
| 16 | 000170 | D11 | 2,056.67 | 411.33 | 1,645.34 | 09/15/2000 | 12361 |
| 17 | 000180 | D11 | 1,778.33 | 355.67 | 1,422.66 | 09/15/2000 | 12362 |
| 18 | 000190 | D11 | 1,704.17 | 340.83 | 1,363.34 | 09/15/2000 | 12363 |
| 19 | 000200 | D11 | 2,311.67 | 462.33 | 1,849.34 | 09/15/2000 | 12364 |
| 20 | 000210 | D11 | 1,522.50 | 304.50 | 1,218.00 | 09/15/2000 | 12365 |

Gambar 6.6
Data Table Payroll

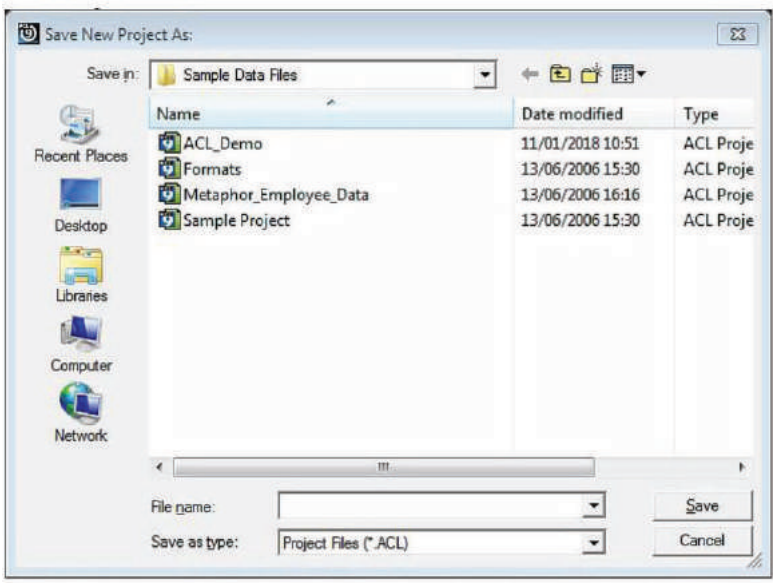
Data tabel Payroll ditampilkan dalam bentuk kolom menyamping seperti *spreadsheet*, tetapi data ini tidak bisa diubah secara langsung. ACL akan bekerja

dengan *virtual field* (**Field** – kolom tabel yang ditampilkan pada layar atau **View**, tetapi di dalam *database* tidak ada datanya, yang ada hanya formulanya).

### b. Membuat proyek baru.

Selanjutnya kita akan membuat proyek baru dengan nama yang menggambarkan proyek pemeriksaan yang dilakukan atau objek audit. Dalam kegiatan pekerjaan sesungguhnya sebaiknya penamaan proyek terdiri dari: nama klien_objek audit_ periode (misalnya, Indomobil_piutang_2016).

1. Pada bagian *toolbar* klik tombol atau pada bagian menu bar, klik menu **File** – **New** – **Project**.



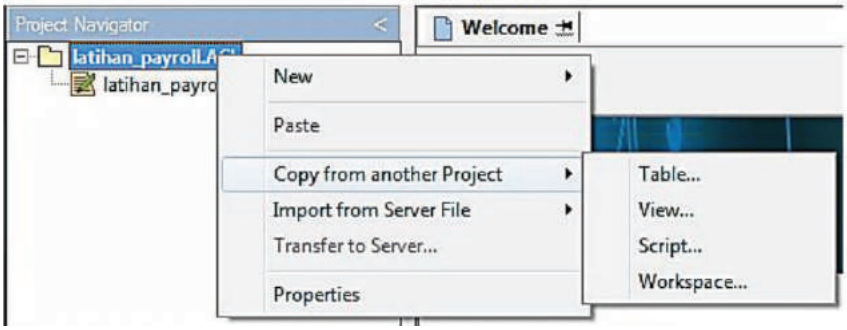

Gambar 6.7
Kotak Dialog Save New Project As

2. Pada isian *file name*, ketikkan nama proyek yang diinginkan (misalkan, *latihan_ payroll*).
3. Klik tombol **Save**.
   Selanjutnya ACL menampilkan menu **Data Definition Wizard**. Klik tombol **Cancel – Yes**.

### c. Mengakses data dari proyek lain.

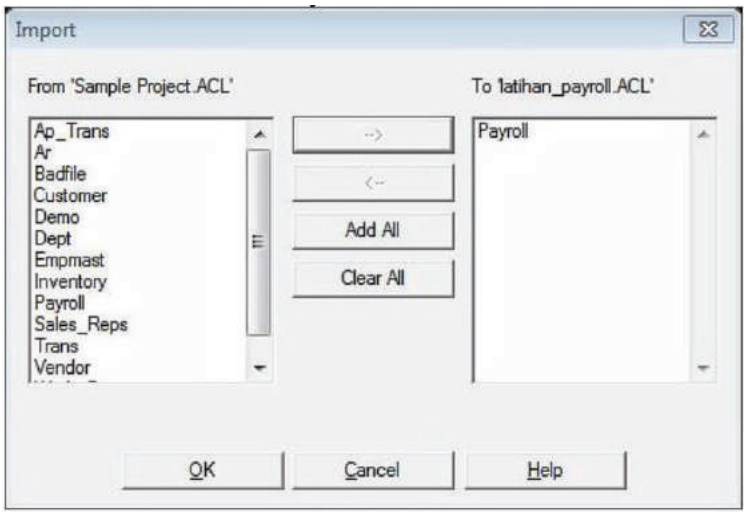
Mengambil atau menyalin *file* dari *project* lain yang sudah ada.

1. Pada bagian **Project Navigator**, klik kanan pada proyek *latihan_payroll.*
2. Kemudian klik perintah **Copy from another project – Table.**



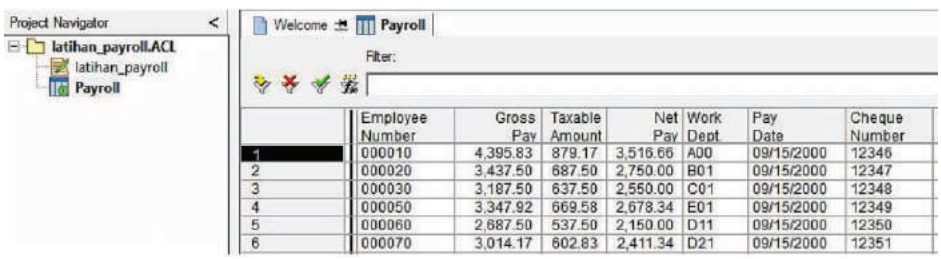

Gambar 6.8
Menu Copy from Another Project – Table

3. Pilih *file* proyek Sample Project untuk menampilkan daftar tabel yang ada. Untuk latihan awal, klik tabel Payroll kemudian klik tombol **-->**




Gambar 6.9
Kotak Dialog Import

4. Klik tombol **OK.**
   Pada bagian **Project Navigator** akan muncul nama tabel **Payroll**, untuk mengaktifkan dan menampilkannya klik dua kali nama tabel tersebut.




| | Employee Number | Gross Pay | Taxable Amount | Net Pay | Work Dept | Pay Date | Cheque Number |
|---|---|---|---|---|---|---|---|
| 1 | 000010 | 4,395.83 | 879.17 | 3,516.66 | A00 | 09/15/2000 | 12346 |
| 2 | 000020 | 3,437.50 | 687.50 | 2,750.00 | B01 | 09/15/2000 | 12347 |
| 3 | 000030 | 3,187.50 | 637.50 | 2,550.00 | C01 | 09/15/2000 | 12348 |
| 4 | 000050 | 3,347.92 | 669.58 | 2,678.34 | E01 | 09/15/2000 | 12349 |
| 5 | 000060 | 2,687.50 | 537.50 | 2,150.00 | D11 | 09/15/2000 | 12350 |
| 6 | 000070 | 3,014.17 | 602.83 | 2,411.34 | D21 | 09/15/2000 | 12351 |

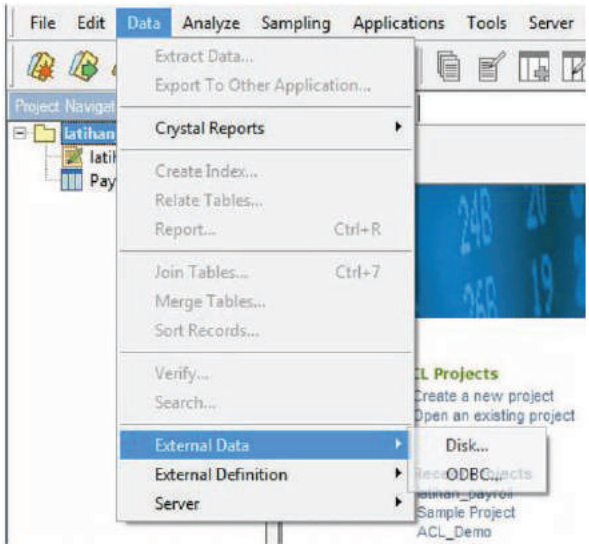
Gambar 6.10
Tabel Payroll Hasil Import

Sekarang Anda bekerja pada proyek Anda sendiri yang tidak akan bercampur dengan proyek lainnya. Perhatikan pada layar bagian kanan **Project Navigator** yang menampilkan data Payroll yang berisi data **Employee Number**, **Gross Pay**, **Taxable Amount**, **Net Pay**, **Work Dept.**, **Pay Date**, dan **Cheque Number**.

### d. Mengakses data dari *external disk*.

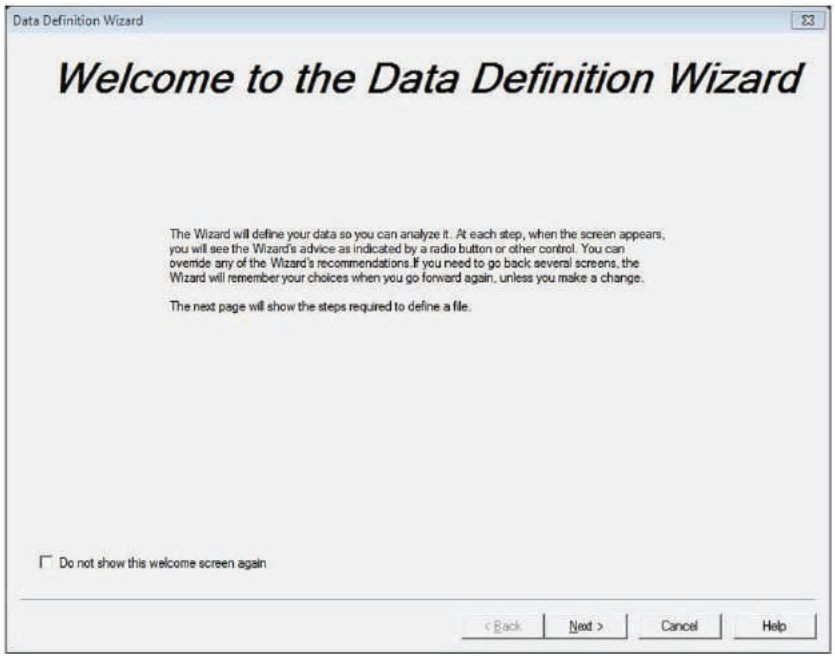
Untuk membuat dokumen baru, kita membutuhkan nama dokumen, yang secara khusus dapat disimpan dalam media simpan *harddisk*. Apabila data yang akan digunakan dalam ACL adalah data baru, maka secara otomatis, pembuatan *input file definition* menggunakan Wizard. Hal ini dimaksudkan agar kita dapat mengatur *file* data sesuai keinginan kita.

1. Pada bagian **Project Navigator**, klik kanan pada proyek *latihan_payroll*.
2. Kemudian klik perintah **New – Table.**
   atau
   Pada bagian menu bar, klik perintah **Data – External Data – Disk**



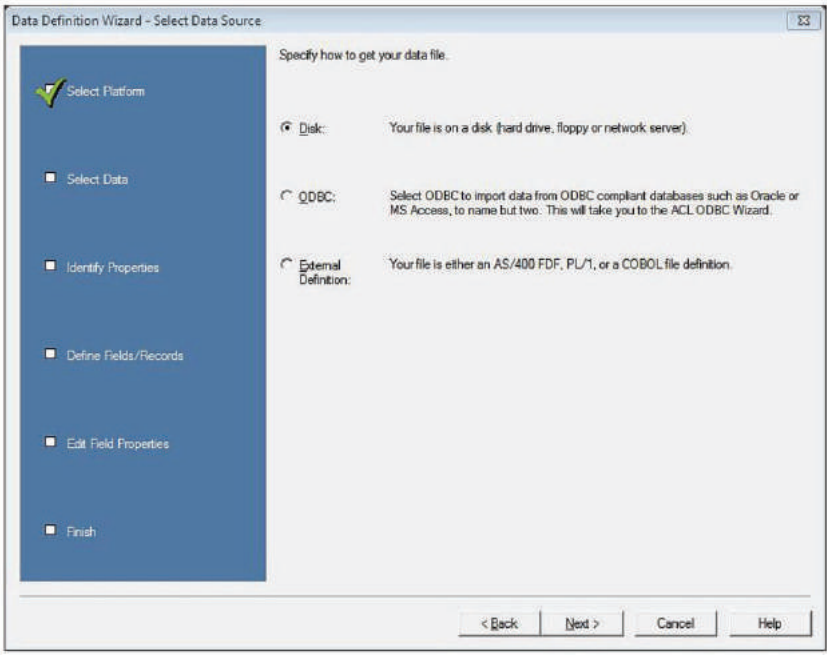

Gambar 6.11
Menu Data – External Data - Disk

3. Kotak dialog **Data Definition Wizard** akan muncul, klik tombol **Next**.



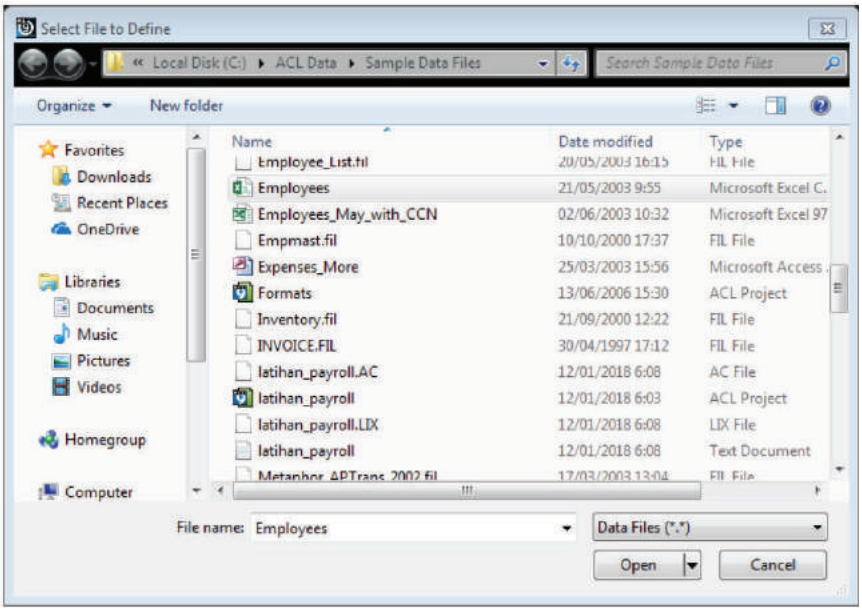

Gambar 6.12
Kotak Dialog Data Definition Wizard

4. Tahapan selanjutnya adalah memilih sumber data yang akan digunakan. Pilih opsi **Disk** jika data yang dimiliki berada di dalam media simpan seperti *harddisk*.



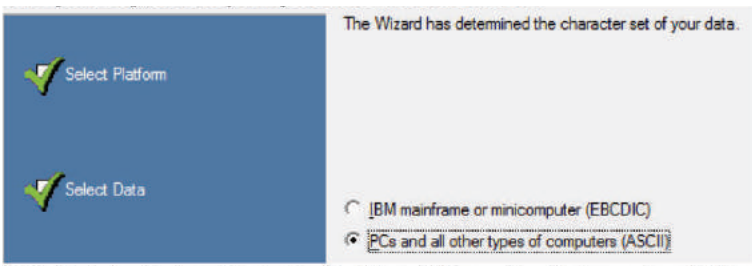

Gambar 6.13
Kotak Dialog Data Definition Wizard – Select Data Source

5. Pada kotak dialog **Select Files to Defines**, pilih *file* data yang akan diperiksa pada lokasi di mana *file* data tersebut disimpan, lalu klik tombol **Open**. Misalkan, pilih *file* Excel *Employees*.



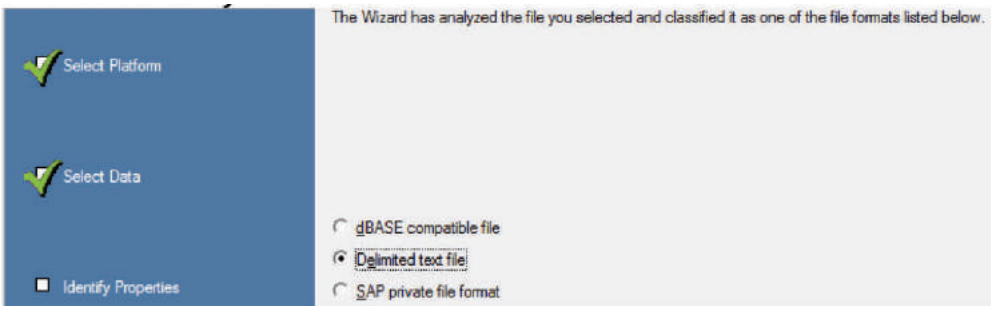

Gambar 6.14
Kotak Dialog Data Definition Wizard – Select Data Source

6. Gunakan opsi *default* dari wizard, yaitu **PCs and all other types of computers (ASCII)**, selanjutnya klik tombol **Next**.



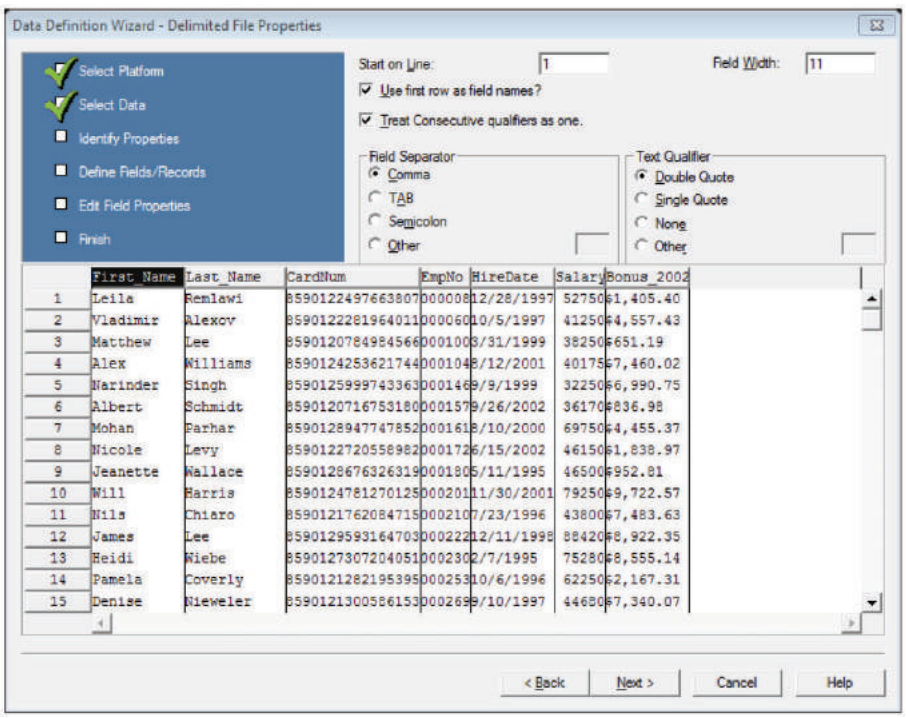

Gambar 6.15
Kotak Dialog Data Definition Wizard – Determined Character Set

7. Gunakan opsi *default* yang disarankan oleh ACL disesuaikan dengan jenis *file* yang dipilih sebelumnya. Dalam kasus ini, opsi *default*-nya adalah **Delimited text file**.



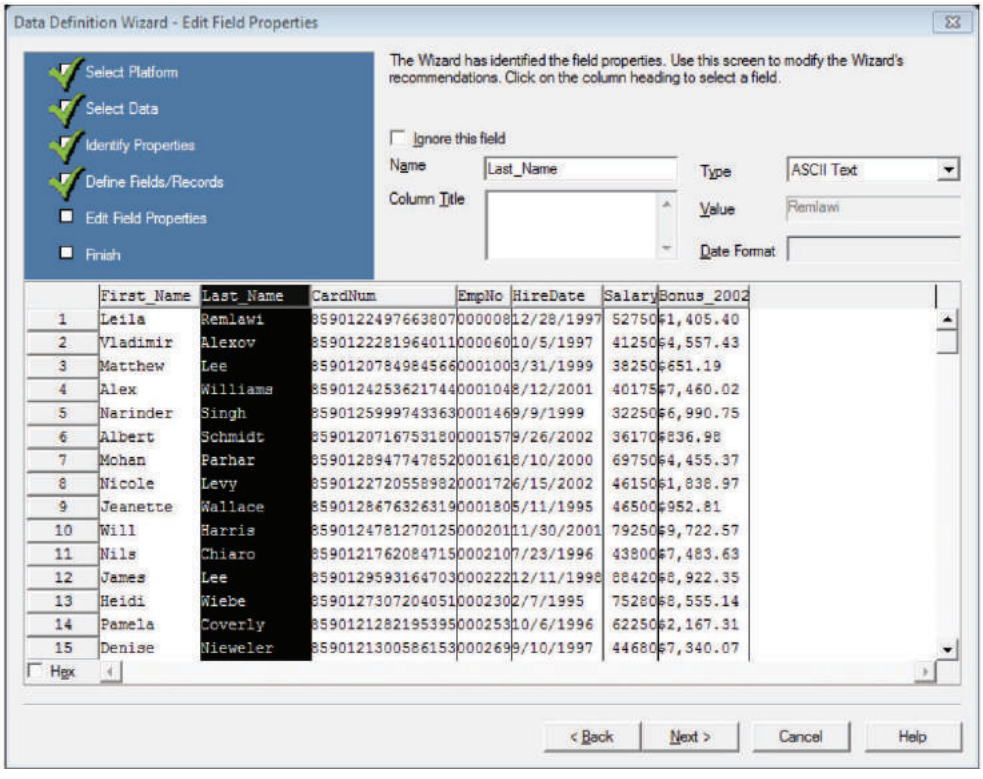

Gambar 6.16
Kotak Dialog Data Definition Wizard – File Format

8. Mengatur properti dari *file Employee*, yaitu penentuan kolom data yang akan digunakan sesuai dengan kebutuhan pemeriksaan. Setelah selesai klik tombol **Next**.




Gambar 6.17
Kotak Dialog Data Definition Wizard – Select Data Source

9. Hasil dari pengaturan properti *file* tersebut disimpan menjadi *file* kerja dengan ekstensi ***.fil***, simpanlah *file* kerja dengan nama *karyawan.fil*, kemudian klik tombol **Save**.
10. Setelah pengaturan properti *file* disimpan, tahapan berikutnya adalah kegiatan mengedit **Field Properties**, yaitu menentukan properti *field* seperti nama *field*, *type*, *decimal*, dan lain-lain. Klik tombol **Next** untuk lanjut ke tahap berikutnya.




Gambar 6.18
Kotak Dialog Data Definition Wizard – Edit Field Properties

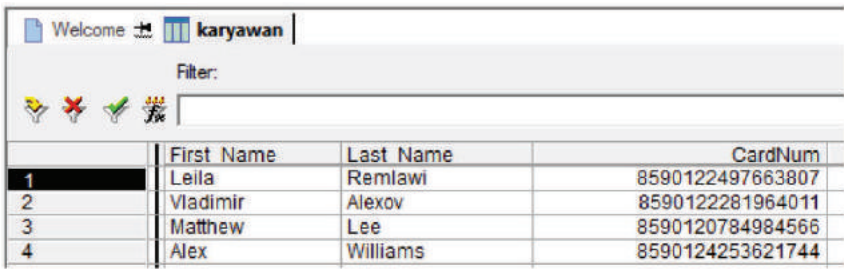
11. Hasil akhir proses pendefinisian *file input* (*table layout*) data karyawan adalah sebagai berikut (hasil pendefinisian *file* data).

| Value | Definition |
|---|---|
| Table Data Source File | Employees.csv |
| Character Set | ASCII |
| Record Length | 73 |
| Skip Length | 0 |
| Number of Fields | 7 |
| Field Name | Data Type |
| First_Name | ASCII |
| Last_Name | ASCII |
| CardNum | PRINT |
| EmpNo | PRINT |
| HireDate | ASCII |
| Salary | PRINT |
| Bonus_2002 | ASCII |

Gambar 6.19
Hasil Pendefinisian File Input

12. Klik tombol **Finish**, kemudian ketikkan nama *file* kerja (misalkan, *Karyawan*). Tujuan dilakukan pekerjaan ini agar memudahkan auditor dalam menganalisis

data yang akan diaudit karena data sudah terdefinisi dengan baik, seperti gambar di bawah ini.




| | First_Name | Last_Name | CardNum |
|---|---|---|---|
| 1 | Leila | Remlawi | 8590122497663807 |
| 2 | Vladimir | Alexov | 8590122281964011 |
| 3 | Matthew | Lee | 8590120784984566 |
| 4 | Alex | Williams | 8590124253621744 |

Gambar 6.20
Hasil Pendefinisan Tabel pada File Input Karyawan

Data *file* ACL dibuat dengan *default* ekstensi **.FIL**, tetapi data *file* pada *software* lainnya dapat dibuat dengan ekstensi yang lain, contohnya file *dBase* mempunyai ekstensi **.DBF**. Perhatikan bahwa status bar menampilkan nama dari dokumen yang dibuka pada sisi kiri dan melaporkan jumlah data yang terkandung dalam *file* tersebut. Nama dari *input file* ditampilkan pada sisi kanan dari status bar. *Path* dan nama dari data *file* ditentukan dalam *title bar* pada jendela *input file definition*.

### 6.5.3 Pengolahan Data Dasar

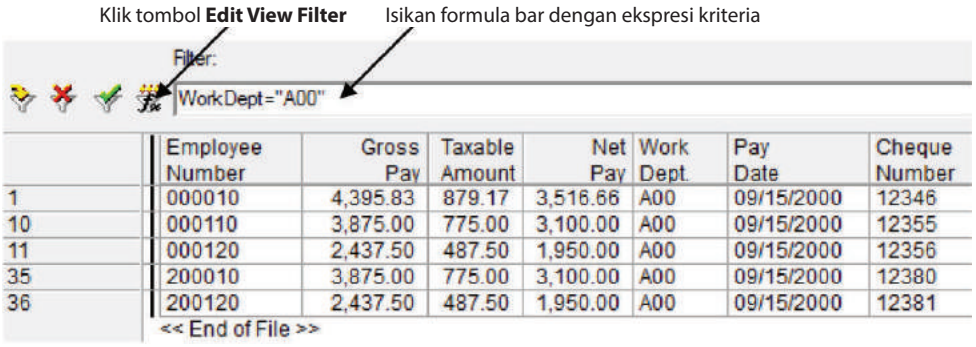
Setelah menyelesaikan tahapan membuat proyek serta mendefinisikan *table* data yang akan digunakan dalam pemeriksaan. Tahapan selanjutnya dapat menggunakan perintah dasar ACL untuk melakukan analisis awal yang berkaitan dengan kegiatan memeriksa integritas data, seperti kelengkapan, keunikan, kewajaran, kebenaran, dan validasi data.

#### a. Filter Data

Setiap data tabel kerja dapat menyaring (memfilter) data *record* dengan kriteria tertentu yang akan diproses. Untuk melakukan filter dapat dilakukan dua cara, yaitu dengan (1) mengetik ekspresi kriteria pada formula bar, atau (2) menggunakan **Edit View Filter.**

**(1) Formula Bar**

Sebagai contoh, kita akan memproses hanya data karyawan dengan Work Dept. adalah A00. Bukalah *file input* Payroll, kemudian dalam kotak isian formula bar ekspresi ditulis WorkDept=”A00”, kemudian **Enter**.




| | Employee Number | Gross Pay | Taxable Amount | Net Pay | Work Dept. | Pay Date | Cheque Number |
|---|---|---|---|---|---|---|---|
| 1 | 000010 | 4,395.83 | 879.17 | 3,516.66 | A00 | 09/15/2000 | 12346 |
| 10 | 000110 | 3,875.00 | 775.00 | 3,100.00 | A00 | 09/15/2000 | 12355 |
| 11 | 000120 | 2,437.50 | 487.50 | 1,950.00 | A00 | 09/15/2000 | 12356 |
| 35 | 200010 | 3,875.00 | 775.00 | 3,100.00 | A00 | 09/15/2000 | 12380 |
| 36 | 200120 | 2,437.50 | 487.50 | 1,950.00 | A00 | 09/15/2000 | 12381 |

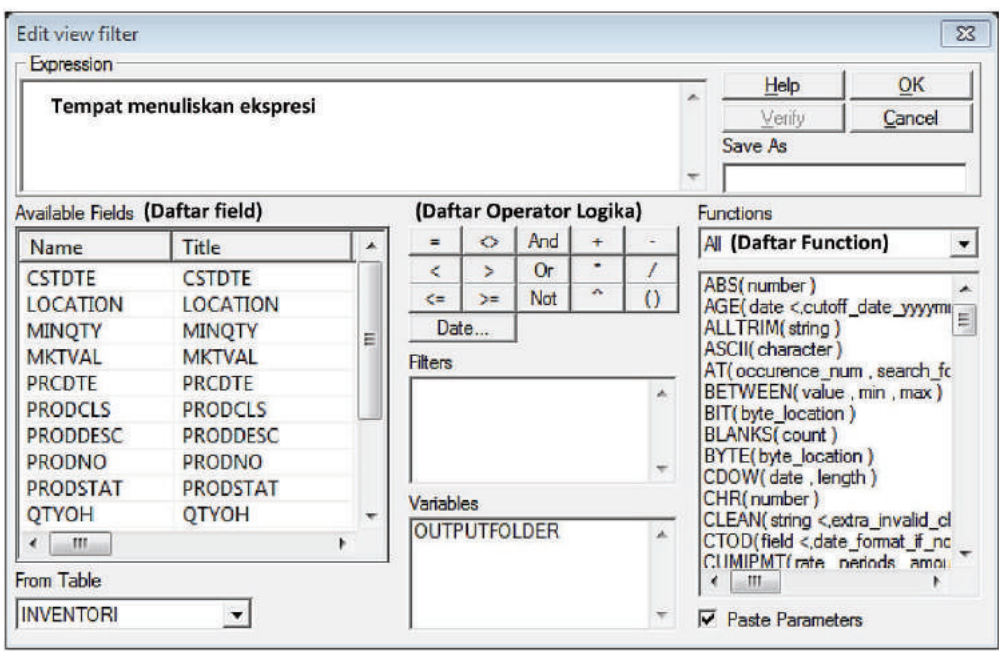
<< End of File >>

Gambar 6.21
Hasil Filter Data Karyawan dengan Filter WorkDept="A00"

Kriteria untuk memfilter data *record* dapat lebih dari satu, misalkan **WorkDept = "A00" AND NetPay <=2000**, artinya menampilkan data karyawan pada *Departemen* **A00** dan *Net Pay* lebih kecil dari **2000**.

### (2) Edit View Filter

Sebagai contoh, kita akan memproses hanya data inventori dengan *Location* adalah 06. Bukalah *file input* **Inventori**, kemudian maka klik tombol untuk menampilkan kotak dialog **Edit view filter**.




Gambar 6.22
Kotak Dialog Edit View Filter

Ketik ekspresi kriteria di dalam kotak **Expression**, atau dapat juga menggunakan *field* yang ada di dalam daftar **Available Fields**, operator logika, dan **Functions** yang tersedia.

Pada kasus di atas, klik dua kali pada *field Location* untuk menyalin *field Location* ke dalam kotak **Expression**. Demikian juga tombol operator "**=**", sedangkan "06" bisa diketik langsung.


Setelah selesai membuat ekspresi kriteria klik **OK**.

Klik tombol untuk menghapus efek filter yang telah dibuat, baik dari **Formula Bar** maupun dari kotak dialog **Edit View Filter**.

INVENTORI

Filter:

LOCATION = "06"

| | CSTDTE | LOCATION | MINQTY | MKTVAL | PRCDTE | PRODCLS | PRODDESC |
|---|---|---|---|---|---|---|---|
| 1 | 10/10/2000 | 06 | 980 | 8691.30 | 10/18/2000 | 07 | LATEX SEMI-GLOSS ORANGE |
| 2 | 10/10/2000 | 06 | 985 | 4595.40 | 10/18/2000 | 07 | LATEX SEMI-GLOSS CARAME |
| 3 | 10/10/2000 | 06 | 750 | 14785.20 | 10/18/2000 | 07 | LATEX SEMI-GLOSS LILAC |
| 4 | 10/10/2000 | 06 | 780 | 12887.10 | 10/18/2000 | 07 | LATEX SEMI-GLOSS APRICOT |
| 5 | 10/10/2000 | 06 | 420 | 14985.00 | 10/18/2000 | 07 | LATEX SEMI-GLOSS PINK |
| 6 | 10/10/2000 | 06 | 430 | 24175.80 | 10/18/2000 | 07 | LATEX SEMI-GLOSS YELLOW |
| 7 | 10/10/2000 | 06 | 670 | 18681.30 | 10/18/2000 | 07 | LATEX SEMI-GLOSS GREEN |

<< End of File >>

Gambar 6.23
Hasil Filter Data Inventori dengan Filter Location ="06"

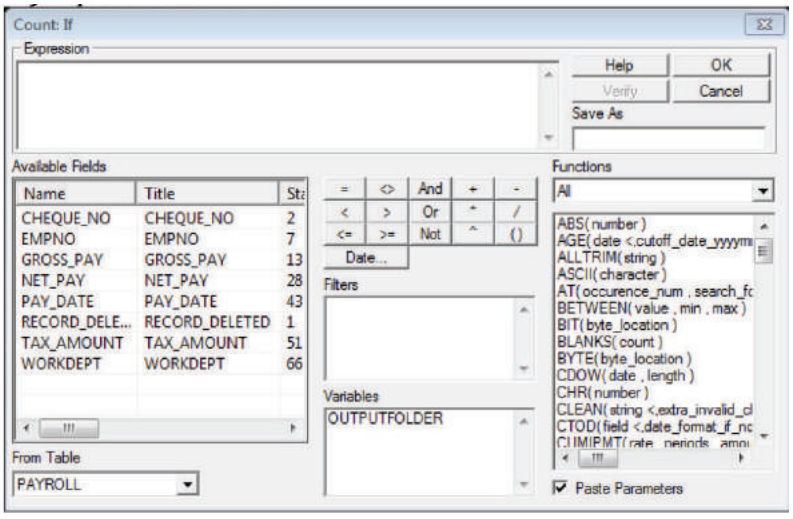
## b. Menghitung Jumlah Data (Record)

Hal ini bertujuan untuk menghitung jumlah data *record* di dalam *table*, atau sesuai dengan kondisi yang telah ditentukan dan untuk meyakinkan bahwa kita bekerja pada jumlah data yang benar dan lengkap. Hasil dari proses penghitungan ini akan ditampilkan dalam *view analysis*.

Misalkan akan menghitung jumlah data (*record*) karyawan yang berada dalam Kode Departemen A00 (WorkDept = A00). Bukalah *file input* Payroll.

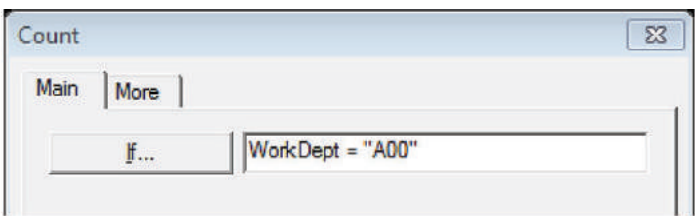
***Tahapan menghitung jumlah record***

1. Pada menu bar klik perintah menu **Analyze – Count Records.**
2. Pada kotak dialog **Count**, klik tombol **IF** untuk menampilkan kotak dialog **Count: If** seperti berikut ini.

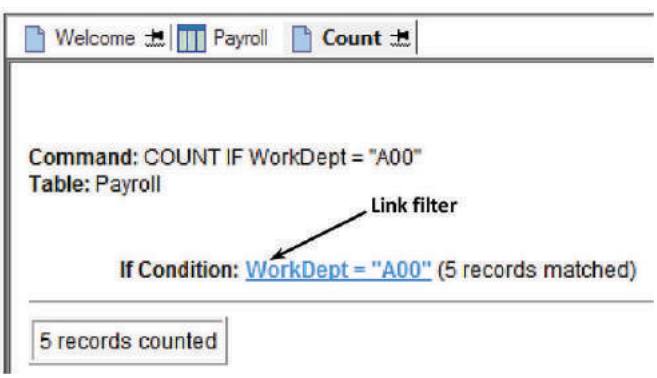



Gambar 6.24
Kotak Dialog Count: If untuk Menuliskan Ekspresi Kriteria

3. Pada daftar **Available Fields**, klik dua kali pada nama *field* **WorkDept** sehingga nama *field* disalin ke dalam kotak Expression, kemudian tambahkan tulisan "=A00" setelah nama *field* **WorkDept**. Klik tombol **OK** untuk lanjut ke tahap berikutnya.




4. Klik tombol **OK**, hasil proses ini akan ditampilkan pada **View Analyze** seperti berikut ini.




Gambar 6.25
View Analysis Count dengan Filter WorkDept=A00

Untuk melihat daftar karyawan yang berada pada WorkDept = A00, klik *link* filter untuk menampilkan hasil filter seperti berikut ini.

PAYROLL | Count

Filter:

(WORKDEPT = "A00")

| | CHEQUE_NO | EMPNO | GROSS_PAY | NET_PAY | PAY_DATE | TAX_AMOUNT | WORKDEPT |
|---|---|---|---|---|---|---|---|
| 1 | 12346 | 000010 | 4395.83 | 3516.66 | 09/15/2000 | 879.17 | A00 |
| 10 | 12355 | 000110 | 3875.00 | 3100.00 | 09/15/2000 | 775.00 | A00 |
| 11 | 12356 | 000120 | 2437.50 | 1950.00 | 09/15/2000 | 487.50 | A00 |
| 35 | 12380 | 200010 | 3875.00 | 3100.00 | 09/15/2000 | 775.00 | A00 |
| 36 | 12381 | 200120 | 2437.50 | 1950.00 | 09/15/2000 | 487.50 | A00 |

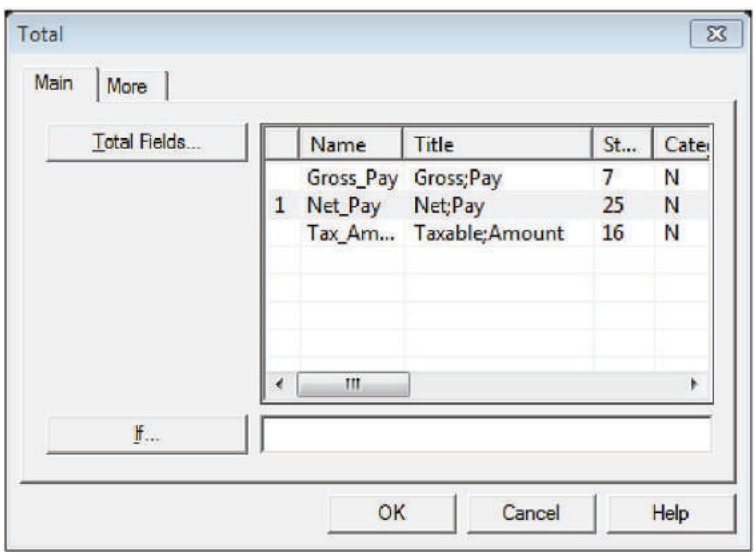
<< End of File >>

Gambar 6.26
Data Karyawan yang Berada pada WorkDept = A00

### c. Menghitung Total Field

Misalkan menghitung total gaji (*Total Net Pay*) karyawan pada *file input* Payroll. Tahapan menghitung total *field* sebagai berikut.

1. Pada menu bar klik perintah menu **Analyze – Total Fields.**
   ACL akan menampilkan kotak dialog **Total** dan hanya menampilkan *field* yang bertipe angka (*numeric*).




Gambar 6.27
Kotak Dialog Total

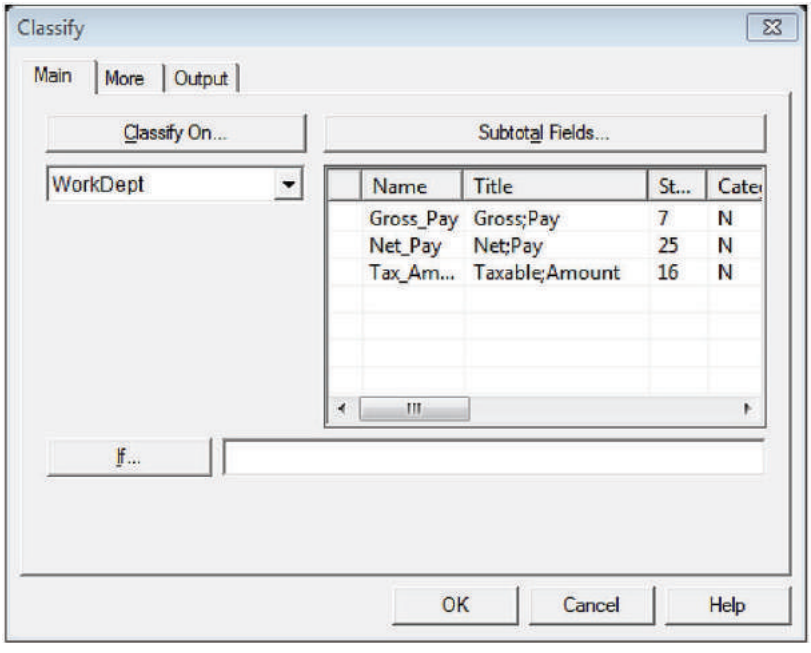
2. Pada bagian *list field*, pilih atau klik *field* mana yang akan dihitung totalnya (misalkan, **Net_Pay**).
3. Jika ada kriteria, isikan pada kotak isian **IF** atau tombol **IF** (lihat pembahasan sebelumnya).
4. Klik tombol **OK**.


Gambar 6.28
View Analysis Total Gaji Karyawan pada File Input Payroll

### d. Klasifikasi Data

Tujuan dari klasifikasi data adalah membuat pengelompokan dan rekapitulasi data berdasarkan *field character*. Misalkan akan membuat rekapitulasi gaji karyawan berdasarkan kelompok departemen pada *file input* Payroll.

1. Pada menu bar klik perintah menu **Analyze – Classify.**
   ACL akan menampilkan kotak dialog **Classify** dan hanya menampilkan *field* yang bertipe huruf/karakter (*character*) sebagai dasar klasifikasi dan daftar *field* bertipe *numeric* yang akan direkapitulasi.




Gambar 6.29
Kotak Dialog Classify

2. Klik tombol **Classify On**, pilih *field character* yang akan dijadikan dasar klasifikasi dan pada daftar **Subtotal Fields**, pilih *field numeric* yang akan direkapitulasi. Misalkan untuk dasar klasifikasi pilih *field* **WorkDept** dan *field* yang akan dihitung subtotalnya, pilih *field* **Net_Pay**.

3. Klik tombol **OK**.

**Command:** CLASSIFY ON WorkDept SUBTOTAL Net_Pay TO SCREEN
**Table:** Payroll

| Work Dept. | Count | Percent of Count | Percent of Field | Net Pay |
|---|---|---|---|---|
| A00 | 5 | 11.36% | 17.09% | 13,616.66 |
| B01 | 1 | 2.27% | 3.45% | 2,750.00 |
| C01 | 4 | 9.09% | 9.95% | 7,925.98 |
| D11 | 11 | 25% | 23.15% | 18,441.38 |
| D21 | 7 | 15.91% | 15.04% | 11,978.68 |
| E01 | 1 | 2.27% | 3.36% | 2,678.34 |
| E11 | 6 | 13.64% | 9.82% | 7,826.00 |
| E21 | 8 | 18.18% | 16.47% | 13,121.38 |
| E83 | 1 | 2.27% | 1.67% | 1,330.00 |
| Totals | 44 | 100% | 100% | 79,668.42 |

Gambar 6.30
View Analysis Klasifikasi Data Payroll berdasarkan WorkDept

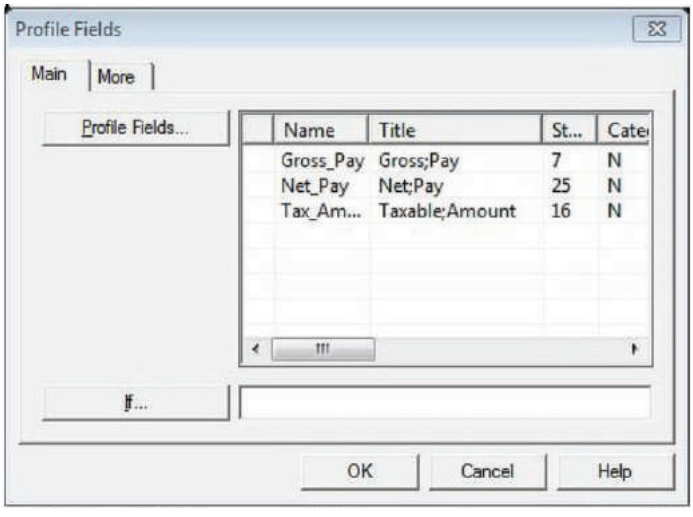
### e. Stratify Data

Tujuan dari stratifikasi (*stratify*) data adalah mengelompokkan data numerik (interval) dalam sebuah *file*. *Stratify* akan menghitung jumlah dan total *record* berdasarkan interval yang dapat ditentukan sendiri. Perintah **Stratify** melibatkan 2 operasi, yaitu **Profile Data** dan **Stratify Data**.

- **Profile Data**

Perintah ini digunakan untuk menampilkan nilai minimum dan maksimum dari *field numeric* yang akan dibagi dalam tingkatan.

- Pada menu bar klik perintah menu **Analyze – Statistical – Profile.**
  ACL akan menampilkan kotak dialog **Profile** dan menampilkan *field* yang bertipe numerik.
- Pada daftar *field*, pilihlah nama *field* numerik yang akan dihitung *total value*, *absolute value*, maksimum, dan minimum.
- Klik tombol **IF** untuk memberikan kriteria pada proses **Profile**.
  Misalkan *profile* data untuk kelompok pegawai pada departemen dengan kode B01.

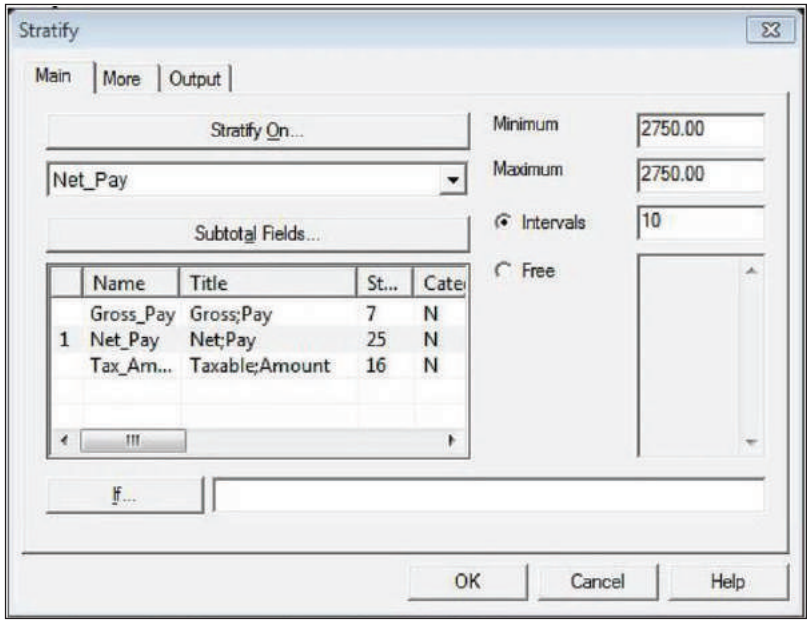



- Klik tombol **OK**.

**If Condition:** WorkDept = "B01" (1 records matched)

| Field Name | Total Value | Absolute Value | Minimum | Maximum |
|---|---|---|---|---|
| Taxable Amount | 687.50 | 687.50 | 687.50 | 687.50 |
| Net Pay | 2,750.00 | 2,750.00 | 2,750.00 | 2,750.00 |
| Gross Pay | 3,437.50 | 3,437.50 | 3,437.50 | 3,437.50 |

- **Stratify Data**
  - Pada menu bar klik perintah menu **Analyze – Stratify.**
    ACL akan menampilkan kotak dialog **Stratify** dan menampilkan *field* yang bertipe numerik
  - Pada *drop down list* [**Stratify On**], pilih *field numeric*. Misalkan pilih *field Net_Pay*, perhatikan pada bagian nilai Minimum dan Maximum secara otomatis diset oleh ACL. Demikian juga dengan **Intervals** diset 10, jika interval ingin ditentukan sendiri, klik opsi **Free**, lalu isikan nilai interval yang diinginkan.
  - Klik tombol **IF** untuk menambahkan kriteria operasi **Stratify**.




- Klik tombol **OK**.

*Range* untuk unit Net Pay (*range* telah ditentukan oleh ACL)

**Command:** STRATIFY ON Net_Pay SUBTOTAL Net_Pay INTERVALS 10 TO SCREEN
**Table:** Payroll

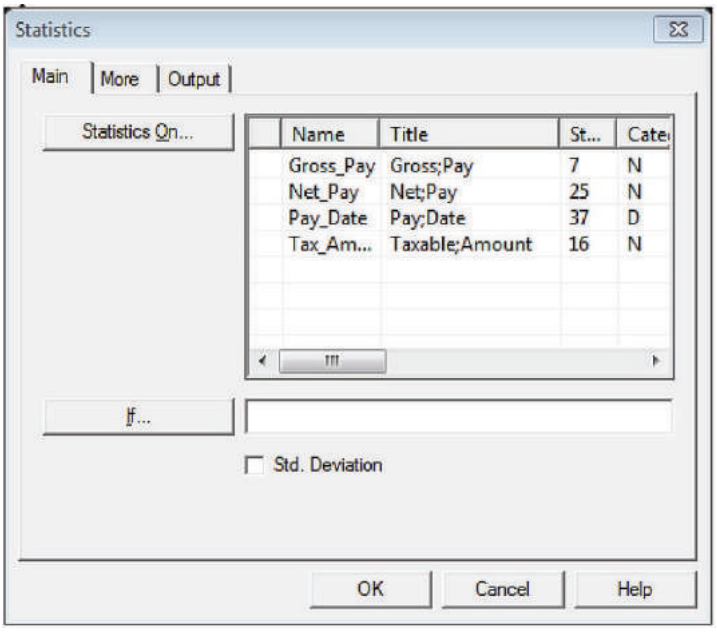
Minimum encountered was 1,022.66
Maximum encountered was 3,516.66

| Net Pay | Count | Percent of Count | Percent of Field | Net Pay |
|---|---|---|---|---|
| <2,750.00 | 40 | 90.91% | 84.35% | 67,201.76 |
| 2,750.00 - 2,750.00 | 1 | 2.27% | 3.45% | 2,750.00 |
| >2,750.00 | 3 | 6.82% | 12.2% | 9,716.66 |
| **Totals** | 44 | 100% | 100% | 79,668.42 |

## f. Statistic

Fasilitas ini digunakan untuk menampilkan nilai statisik dari kumpulan data tabel, seperti nilai maksimum, minimum, average, total, dan sebagainya. Perintah ini memberikan gambaran data berupa data statistik.

- Pada menu bar klik perintah menu **Analyze – Statistical – Statistics**.
  ACL akan menampilkan kotak dialog **Statistic** dan menampilkan *field* yang bertipe numerik.
- Pada *drop down list* [**Statistic On**], pilih *field numeric* yang akan dihitung statistiknya (bisa lebih dari satu *field*). Misalkan pilih *field Net_Pay*, untuk menampilkan data statistik dari *Net Pay*.




- Klik opsi *check box* [**Std. Deviation**] untuk menyertakan data standar deviasi pada *output* **Statistic**.
- Klik tombol **IF** untuk manambahkan kreiteria pada operasi **Statistic**.
- Klik tombol **OK**.

**Command:** STATISTICS ON Net_Pay STD TO SCREEN NUMBER 5
**Table:** Payroll

**Net Pay**

| | Number | Total | Average |
|---|---|---|---|
| Range | - | 2,494.00 | - |
| Positive | 44 | 79,668.42 | 1,810.65 |
| Negative | 0 | 0.00 | 0.00 |
| Zeros | 0 | - | - |
| Totals | 44 | 79,668.42 | 1,810.65 |
| Abs Value | - | 79,668.42 | - |
| Std. Dev. | - | 554.82 | - |

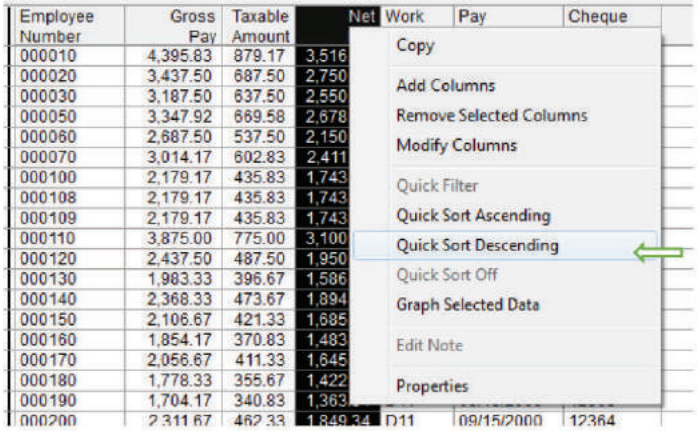
| Highest | Lowest |
|---|---|
| 3,516.66 | 1,022.66 |
| 3,100.00 | 1,060.00 |
| 3,100.00 | 1,060.00 |
| 2,750.00 | 1,150.00 |
| 2,678.34 | 1,183.34 |

### g. Mengurutkan Data

Fasilitas ini digunakan untuk mengurutkan data, baik secara *ascending* (dari nilai terkecil ke nilai terbesar) maupun *descending* (dari nilai terbesar ke nilai terkecil).

- Klik pada salah satu *heading* kolom yang akan dijadikan dasar pengurutan.
- Kemudian klik kanan pada *heading* kolom tersebut untuk menampilkan menu *shortcut*.
- Klik pada menu **Quick Sort Ascending** atau **Quick Sort Descending**.

Misalkan kita ingin mengetahui nilai *Net_Pay* (Gaji Neto) yang paling tertinggi, klik *heading* kolom pada *Net_Pay*, kemudian klik kanan – pilih **Quick Sort Descending**.



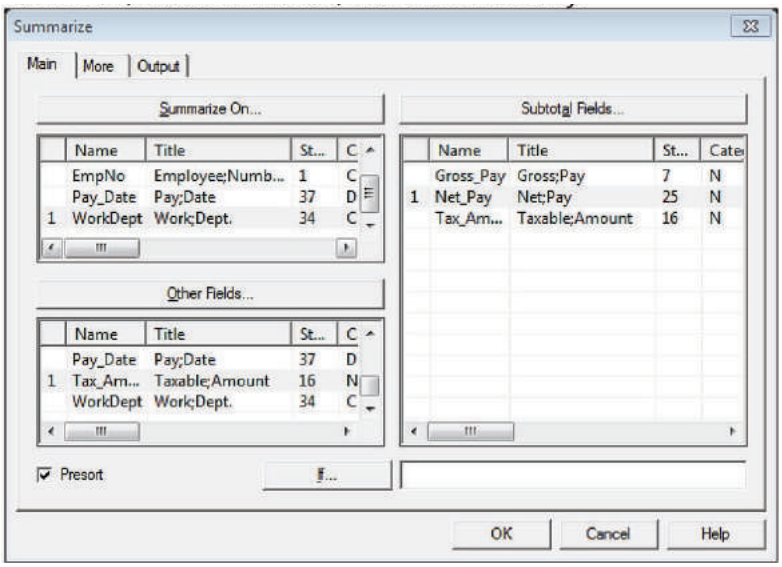

Untuk kembali ke urutan data semula, klik pada *heading* kolom tersebut, lalu klik kanan – pilih **Quick Sort Off**.

### h. Summarize

Fasilitas ini digunakan meringkas data berdasarkan *field* karakter. Perintah ini dapat menghitung jumlah *record* dan menjumlahkan nilai dari *field numeric* berdasarkan kelompok-kelompok yang diinginkan.

Misalkan kita akan melakukan analisis atas suatu tabel yang berisi data karyawan, maka perintah ini kita dapat mengelompokkan data tersebut berdasarkan Bagian (WorkDept). Berdasarkan pengelompokan Bagian, ACL akan menampilkan data berupa jumlah karyawan antarbagian, jumlah gaji karyawan antarbagian.

- Pada menu bar, klik menu **Analyze – Summarize**.
  ACL akan menampilkan kotak dialog **Summarize** terdiri dari 3 *list* (**Summarize On**, **Subtotal Fields**, dan **Other Fields**).




- Pada *list* **Summarize On**, klik *field WorkDept* untuk mengelompokkan berdasarkan bagian.
- Pada *list* **Subtotal Fields**, klik *field Net_Pay* untuk dihitung subtotalnya berdasarkan pengelompokan (**Summarize**). Pada bagian ini, Anda dapat memilih *field* lebih dari satu untuk diringkas.
- Klik tombol **IF** jika Anda akan menambahkan kriteria pada operasi **Summarize**.
- Klik tombol **OK**.

**Command:** SUMMARIZE ON WorkDept SUBTOTAL Net_Pay OTHER Tax_Amount TO SCREEN PRESORT
**Table:** Payroll

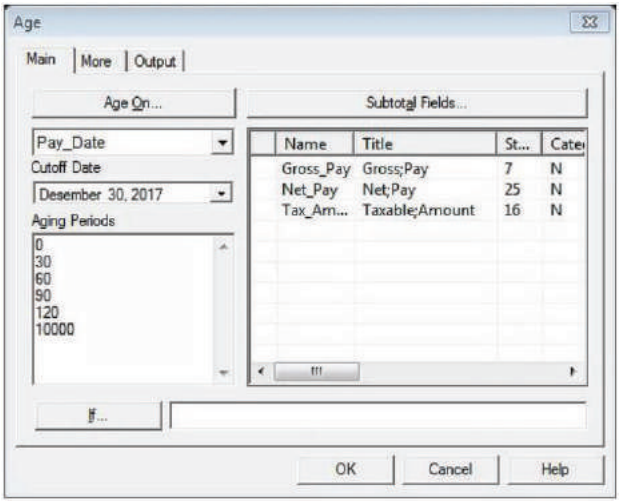
| Work Dept. | Net Pay | Count | Taxable Amount |
|---|---|---|---|
| A00 | 13,616.66 | 5 | 879.17 |
| B01 | 2,750.00 | 1 | 687.50 |
| C01 | 7,925.98 | 4 | 637.50 |
| D11 | 18,441.38 | 11 | 537.50 |
| D21 | 11,978.68 | 7 | 602.83 |
| E01 | 2,678.34 | 1 | 669.58 |
| E11 | 7,826.00 | 6 | 437.50 |
| E21 | 13,121.38 | 8 | 435.83 |
| E83 | 1,330.00 | 1 | 332.50 |
| Totals | 79,668.42 | 44 | 5,219.91 |

9 records produced

### i. Age

Perintah **Age** menghasilkan ringkasan umur dari data yang dapat digunakan untuk membuat *Aging Schedule*, berisi *range* umur dalam hari dan total nilai untuk setiap *range*. Perintah ini dapat disertai dengan kriteria berdasarkan data tanggal. ACL dapat secara otomatis meringkas umur berdasarkan tanggal sistem.

- Pada menu bar klik perintah menu **Analyze – Age …**
  ACL akan menampilkan kotak dialog **Age** dan menampilkan *field* yang bertipe tanggal (*Age On*) dan bertipe numerik (*Subtotal Fields*).




- Pada *drop down list* [**Age On**], pilih *field* tanggal yang akan dilakukan **Aging**, dalam hal ini adalah *field Pay_Date*. Misalkan tanggal *cuttof*.
- Pada *drop down list* [**Subtotal Fields**], pilih *field numeric* yang akan dihitung subtotal. Misalkan pilih *Net_Pay* dan *Tax_Amount*.
- Klik *drop down* [**Cutoff Date**] untuk memilih tanggal *cutoff*. Misalkan tanggal akhir periode perhitungan tahun 2017, set dengan tanggal 30 Desember 2000.
- Pada *list* [**Aging Periode**], ACL secara otomatis *setting* periode **Aging**, jika tidak Anda dapat *setting* sendiri dengan mengedit. Misalkan *setting* **Aging** periode dengan **0 30 60 90 120**.
- Klik tombol **IF** untuk menambahkan kriteria pada proses **Aging**.
  Klik tombol **OK**.

# BAB 7

# ANALISIS DATA DENGAN ACL FOR WINDOWS

## 7.1 BEKERJA DENGAN VIEW

Salah satu Window yang penting dalam ACL adalah View. Di dalam Window View diperlihatkan seluruh isi data yang akan diolah. Tampilan pada Window View dapat diubah sesuai dengan keinginan kita.

*View* adalah jendela yang menampilkan data dari *file* yang sudah didefinisikan (Tabel). Format tampilan seperti *spreadsheet*, yaitu terdiri dari baris dan kolom. Kolom menggambarkan *field* sedangkan baris menggambarkan *record*.

Dalam *view* ACL, masing-masing baris menyajikan sebuah *record* dan masing-masing kolom menyajikan sebuah *field*. Di samping kolom paling kiri adalah nomor *record* tertentu. Nomor *record* yang disorot disebut *current record* (berwarna gelap/ *highlight*). *Record* lain dapat disorot sebagai *current record* dengan mengklik *record* tersebut.

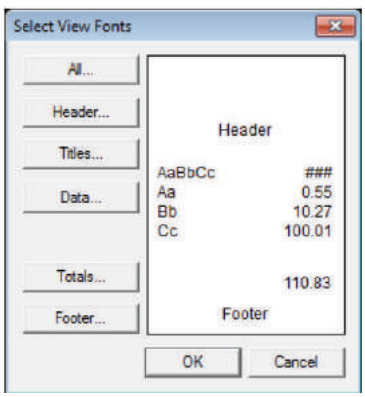
Welcome | Summarize | Inventory

Filter:

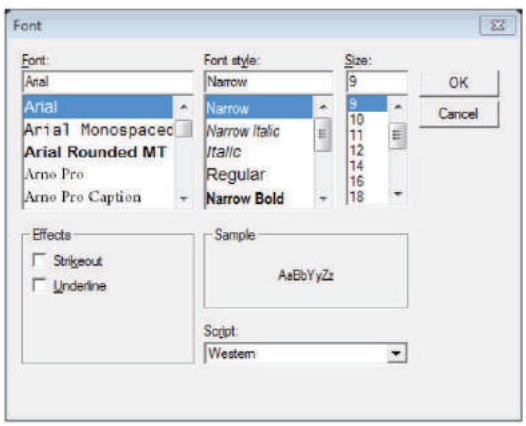
| | Product Number | Product Class | Location | Product Description | Product Status | Unit Cost |
|---|---|---|---|---|---|---|
| 1 | 070104347 | 07 | 06 | LATEX SEMI-GLOSS ORANGE | A | 6.87 |
| 2 | 070104397 | 07 | 06 | LATEX SEMI-GLOSS CARAMEL | A | 6.87 |
| 3 | 070104177 | 07 | 06 | LATEX SEMI-GLOSS LILAC | A | (6.87) |
| 4 | 070104677 | 07 | 06 | LATEX SEMI-GLOSS APRICOT | A | 6.87 |
| 5 | 070104657 | 07 | 06 | LATEX SEMI-GLOSS PINK | A | 6.87 |
| 6 | 070104327 | 07 | 06 | LATEX SEMI-GLOSS YELLOW | A | 6.87 |
| 7 | 070104377 | 07 | 06 | LATEX SEMI-GLOSS GREEN | A | 6.87 |
| 8 | 030414313 | 03 | 03 | METRIC TOOL SET 3/8" DR | A | 47.00 |
| 9 | 030414283 | 03 | 03 | METRIC SOCKET SET 11 PC | A | 18.00 |
| 10 | 030412553 | 03 | 03 | 6 PC OPEN END WRENCH SET | A | 11.53 |

### 7.1.1 Modifikasi View

Tampilan Window View dapat dimodifikasi dengan mengakses *icon change format* yang berada pada *toolbar*  untuk menampilkan kotak dialog **Select View Fonts**.




- Klik tombol [**Header**], [**Titles**], [**Data**], atau [**Footer**] untuk menampilkan window **Font**.



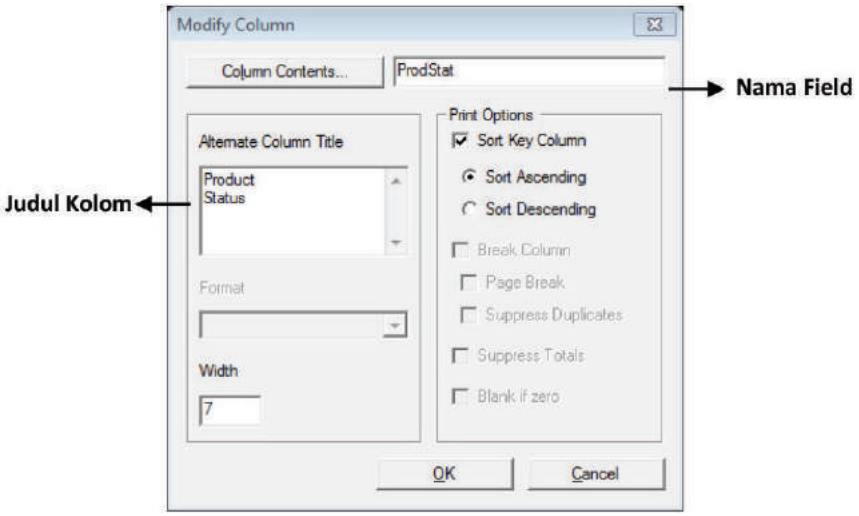
- Pilih **Font** atau dan efek format dari format *font*, klik **OK** jika sudah selesai.
- Klik **OK** untuk keluar dari kotak dialog **Select View Fonts**.

### 7.1.2 Edit dengan Kolom

#### 1. Memindahkan Kolom dan Mengubah Judul Kolom

Posisi urutan kolom dapat diubah sesuai dengan kebutuhan *user* dengan mengklik kolom yang diinginkan, kemudian *drag* sampai ke urutan posisi yang diinginkan baru kemudian dilepas.

Untuk mengubah judul kolom, *double* klik pada *heading* kolom yang diinginkan, kemudian akan tampil kotak dialog seperti berikut ini.

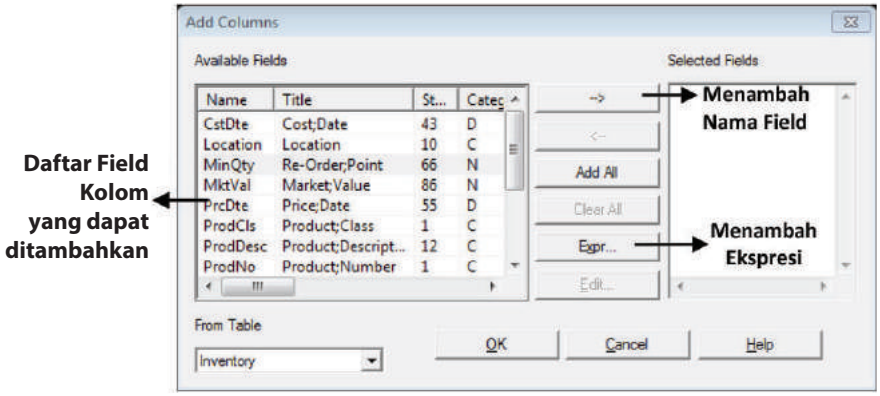



#### 2. Menghapus Kolom

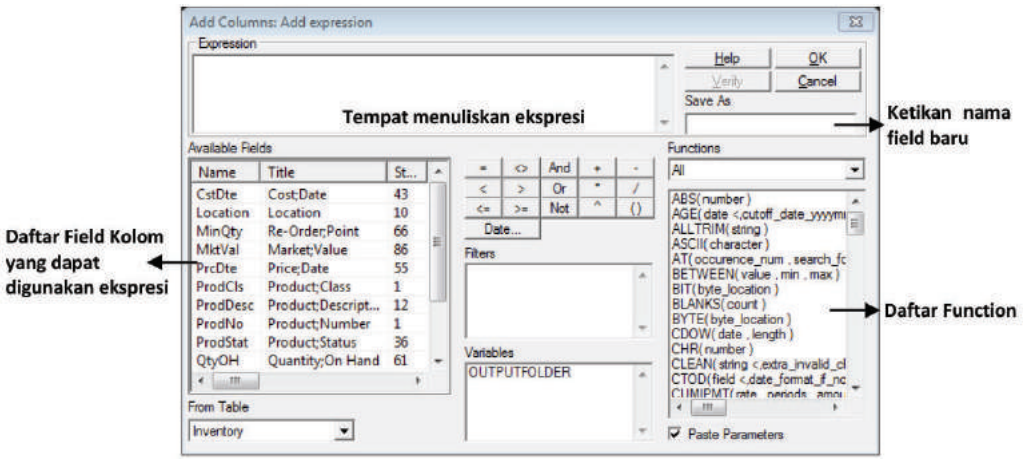
Apabila kolom tertentu tidak digunakan maka ACL dapat menghapus kolom dengan cara klik kolom yang akan dihapus, kemudian pilih *icon* **Remove Columns** untuk menghapus. Atau, klik kanan pada *heading* kolom yang akan dihapus, kemudian klik perintah **Remove Selected Columns**.

#### 3. Menambah Kolom

Untuk menambah kolom klik *icon* **Add New Columns** sampai muncul kotak dialog seperti berikut ini.




Menambahkan *columns* dengan ekspresi: klik tombol **Expr …** sampai muncul dialog *box* seperti di bawah ini.




### 7.1.3 View Filter

Filter adalah sejenis ekspresi yang digunakan untuk mengidentifikasi *record* sesuai dengan kriteria yang diinginkan. Kriteria didefinisikan dengan ekspresi yang bertipe logika.

Filter yang dibuat dapat bersifat sementara atau disimpan pada *input file* definisi sebagai *computed field*. Pada jendela ekspresi akan ditampilkan hasil filter yang telah dibuat. Ada dua buah jenis filter, yaitu:

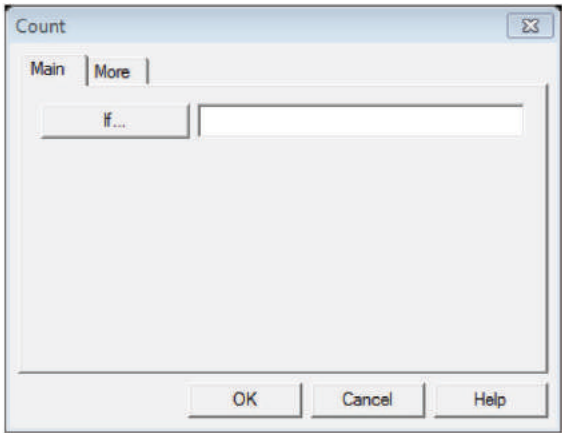
#### 1. Local Filter

Sebuah ekspresi yang digunakan sebagai filter harus mencakup kondisi logika, sehingga ACL dapat menghitung ekspresi tersebut benar atau salah. *Local filter* hanya digunakan pada perintah tunggal yang dilaksanakan pada satu kejadian saja. Suatu *local filter* dapat dikatakan sebagai satu contoh *logical expression*. Agar *expression* dapat digunakan sebagai *filter*, maka harus mencakup kondisi logika, sehingga ACL dapat

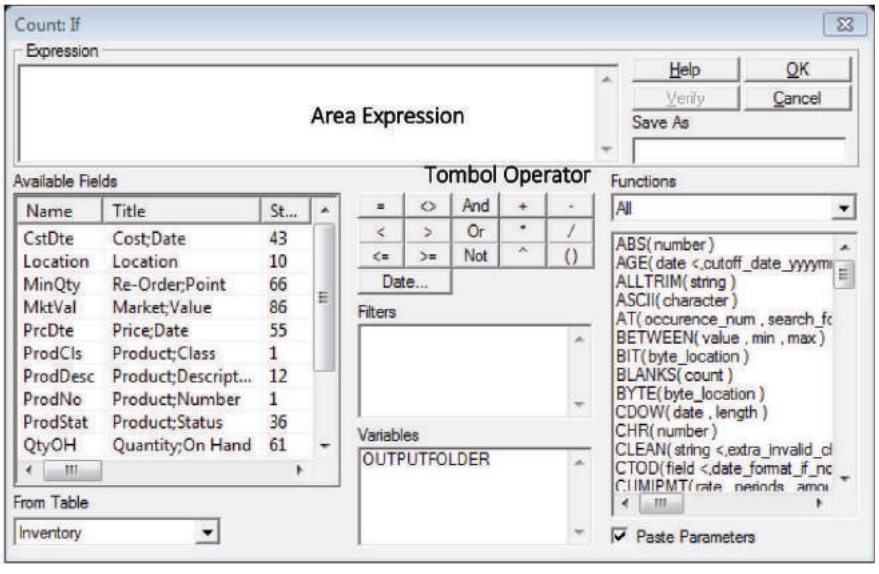
mengevaluasi setiap *record* berdasarkan *expression* tersebut. *Local filter* digunakan ketika perintah tunggal dijalankan satu kali saja.

**Contoh kasus:** Menghitung jumlah data inventori berdasarkan kriteria lokasi.

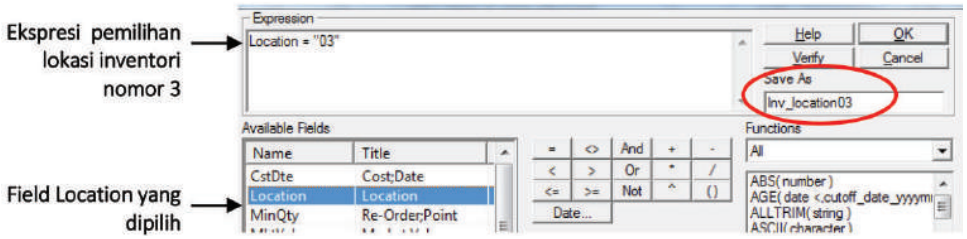
- Bukalah data **Inventory** pada proyek latihan.
- Pada menu **Analyze** – **Count Records**.




- Klik tombol **IF** untuk menampilkan kotak dialog **IF** seperti berikut ini.

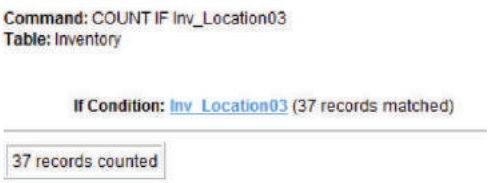



- Pada daftar **Available Fields**, *double* klik pada *field* **Location** sehingga *field* tersebut tampil pada **Area Expression**.
- Pada area tombol operator logika, klik tombol "**=**".
- Pada **Area Expression** setelah tanda "**=**", ketikkan nomor lokasi gudang tempat menyimpan inventori. Misalkan ketikkan nomor lokasi "03".




Jika tipe *field* yang dipilih adalah *numeric*, maka penulisan ekspresinya pada angka tidak disertai tanda kutip " ". Jadi, penulisannya adalah **Location = 3**.

- Pada bagian isian **Save As**, ketikan memberikan nama hasil filter (misalkan **Inv_location03**).
- Jika ingin memeriksa apakah penulisan ekspresi sudah benar, klik tombol **Verify**. Apabila penulisan ekspresi sudah benar, maka akan muncuk kotak dialog **Expression Valid**. Akan tetapi, jika penulisan ekspresi salah, maka akan muncul informasi sesuai dengan kesalahannya.
- Klik tombol **OK** jika sudah selesai dengan penulisan ekspresi, kemudian klik tombol **OK** untuk mengeksekusi ekspresi filter.

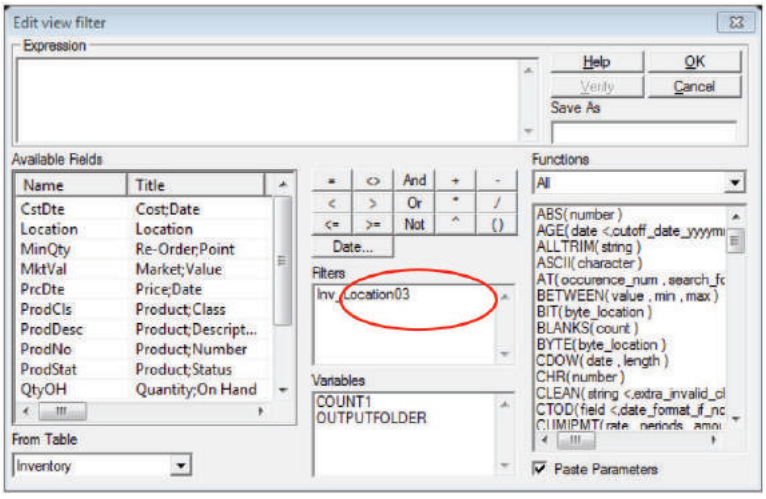



**Analisis**: Dari hasil pada **Command Log** diketahui bahwa dari 37 *record* inventori yang berada di lokasi gudang nomor 03.

## 2. Global Filter

Filter yang digunakan untuk semua *view* pada *input file* definisi selama kita melaksanakan perintah dari menu atau tombol. Filter tetap pada tempatnya sampai kita menutupnya. Ketika Anda menggunakan global filter, semua pemrosesan dalam *input file*, kecuali perintah yang Anda masukkan melalui *command log*, akan terpengaruh. Kondisi yang akan digunakan oleh global filter tersebut akan tetap berpengaruh sampai Anda mematikan filter tersebut, menggunakan filter lain, atau menggunakan *input file definition* yang berbeda. Anda dapat membuat dan menyimpan sejumlah filter yang berbeda, dan kemudian menggunakan salah satu di antaranya yang sesuai dengan kebutuhan.

- Bukalah data **Inventory** pada proyek latihan.
- Pada bar filter, klik tombol (Edit View Filter) untuk menampilkan kotak dialog **Edit View Filter**.




- Pada bagian **Filters**, filter yang sudah dibuat akan muncul (*Inv_Location03*). Klik 2 kali filter tersebut, maka filter *Inv_Location03* akan ditampilkan ke dalam bagian isian **Expression**.
- Klik tombol **OK** jika sudah selesai dengan penulisan ekspresi, kemudian klik tombol **OK** untuk mengeksekusi ekspresi filter.

## 7.2 ANALISIS DATA LANJUTAN

Perintah **Sequences, Duplicates,** dan **Gaps** digunakan untuk melihat keterurutan data, duplikasi, dan gap. *Sequences* berfungsi untuk mempercepat proses pencarian *record* pada *file* data untuk keperluan proses ada atau tidaknya kesenjangan antardata dan data yang sama (*duplicate*). Dengan kata lain, perintah tersebut digunakan untuk memeriksa *file* transaksi apakah terdapat data transaksi yang tidak sesuai urutan, tidak terekam (*missing record*), bahkan data transaksi yang terekam lebih dari satu.

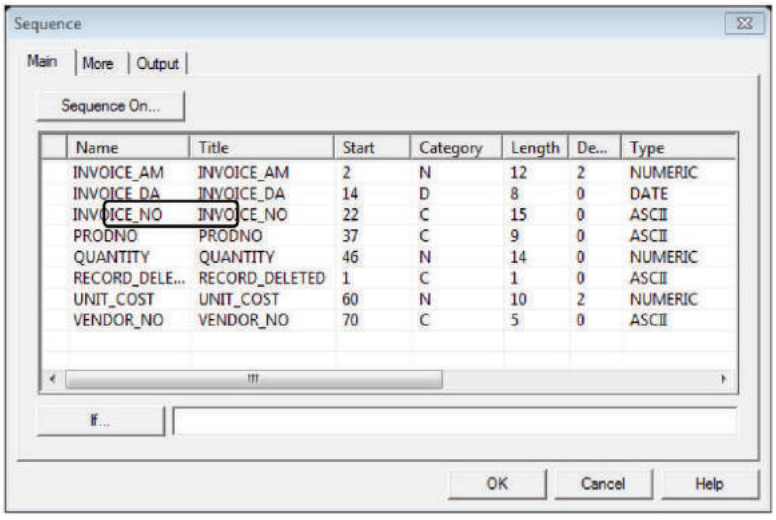
### 7.2.1 Menguji Keterurutan Data (Sequence)

Gunakan perintah **Sequence** untuk menguji bahwa *file input* sudah diurutkan berdasarkan *field* tertentu atau belum. Jika data *file* sudah sesuai urutannya, maka *window* **Command Log** akan menampilkan informasi *no sequence*. Jika *file input* tidak diurutkan berdasarkan *field* yang ditentukan, maka *window* **Command Log** akan menampilkan daftar sejumlah *record* yang tidak sesuai dengan urutannya.

Misalnya, pemeriksaan *file* data transaksi sudah diurutkan berdasarkan nomor *invoice* atau belum, sehingga akan dilakukan pemeriksaan **Sequence** pada *field* INVOICE_NO. Bukalah *file input* AP_TRANS dengan struktur data sebagai berikut.

| Value | Definition |
|---|---|
| Table Data Source File | D:\Dokumen\Buku 2018\DATA DBF\AP_T... |
| Character Set | ASCII |
| Record Length | 74 |
| Skip Length | 257 |
| Number of Fields | 8 |
| Field Name | Data Type |
| RECORD_DELETED | ASCII |
| INVOICE_AM | NUMERIC |
| INVOICE_DA | DATE |
| INVOICE_NO | ASCII |
| PRODNO | ASCII |
| QUANTITY | NUMERIC |
| UNIT_COST | NUMERIC |
| VENDOR_NO | ASCII |

Pada menu **Analyze – Examine Sequence**.




Klik field **INVOICE_NO** pada daftar kotak dialog **Sequence**.

Klik *tab* **Ouput**, pastikan opsi **Screen** terpilih untuk menampilkan hasil pemeriksaan pada layar monitor.

Hasil dari perintah **Sequence** adalah sebagai berikut.

**Command:** SEQUENCE ON INVOICE_NO ERRORLIMIT 10 TO SCREEN
**Table:** AP_TRANS

Sequence test error limit of 10 reached
10 sequence errors detected

**Sequence:**

| Record Number | INVOICE_NO |
|---|---|
| 2 | 2275301 |
| 4 | 5983947 |
| 5 | 589134 |
| 6 | 49545947 |
| 8 | 123196 |
| 10 | 49540141 |
| 12 | 58720114 |
| 14 | 58724783 |
| 16 | 54328931 |
| 18 | 296877 |

Daftar data ***no inovoice*** yang tidak sesuai urutannya.

### 7.2.2 Menguji Kelengkapan Data (Gaps)

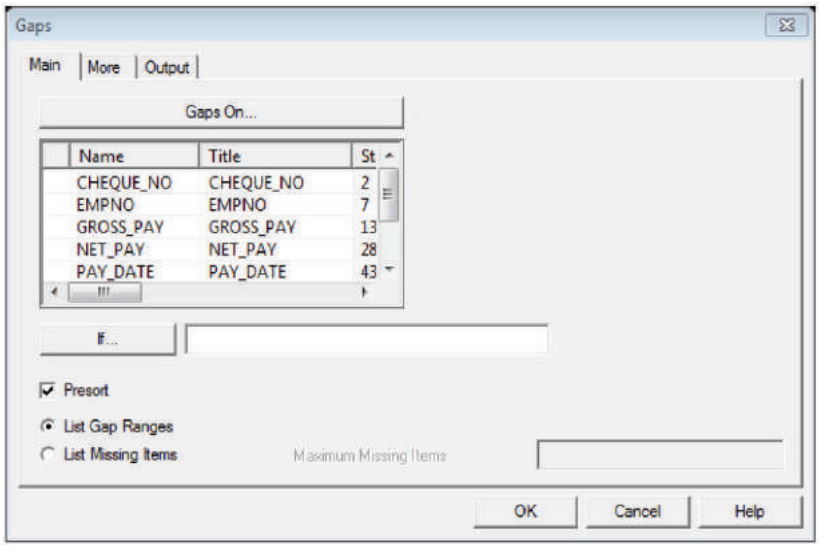
Gunakan perintah **Gaps** untuk menguji atau mendeteksi apakah *file* data berisi data yang tidak berada di dalam urutannya. Dengan kata lain, perintah ini memeriksa *file* data apakah di dalam urutan data terdapat data yang hilang atau tidak terekam.

Sebagai contoh akan dilakukan pemeriksaan pada *file* data Payroll (Penggajian) apakah terdapat nomor cek yang tidak tercantum di dalam urutannya. Jika hal tersebut terjadi, maka ACL menampilkan daftar urutan data yang tidak lengkap atau terjadi gap.

Bukalah *file input* **Payroll** dengan struktur data seperti berikut ini.

| Value | Definition |
|---|---|
| Table Data Source File | D:\Dokumen\Buku 2018\DATA DBF\PAY... |
| Character Set | ASCII |
| Record Length | 68 |
| Skip Length | 257 |
| Number of Fields | 8 |
| Field Name | Data Type |
| RECORD_DELETED | ASCII |
| CHEQUE_NO | ASCII |
| EMPNO | ASCII |
| GROSS_PAY | NUMERIC |
| NET_PAY | NUMERIC |
| PAY_DATE | DATE |
| TAX_AMOUNT | NUMERIC |
| WORKDEPT | ASCII |

Pada menu **Analyze – Look for Gaps**.




Pada *tab* **Output**, pastikan opsi **Screen** aktif.
Di dalam daftar *field*, klik *field* **CHEQUE_NO**
Pada *tab* **Main**, klik opsi

- **List Gap Ranges**: untuk menampilkan daftar *range* yang terdapat datanya tidak tercantum (hilang) dalam *file*.
- **List Missing Items**: untuk menampilkan daftar data-data yang tidak tercantum (hilang) dalam *file*.

Klik **OK**.

**Command:** GAPS ON CHEQUE_NO PRESORT TO SCREEN
**Table:** PAYROLL

1 gap ranges detected
4 missing items

**Gaps Found Between:**

| Gap Start (Exclusive) | Gap End (Exclusive) | Number of Missing Items |
|---|---|---|
| 12,388 | 12,393 | 4 |

**Command:** GAPS ON CHEQUE_NO PRESORT MISSING 5 TO SCREEN
**Table:** PAYROLL

4 missing items

**Gaps Found:**

| CHEQUE_NO | Gap Start (Inclusive) | Gap End (Inclusive) | Number of Missing Items |
|---|---|---|---|
| 12,389 | | | 1 |
| 12,390 | | | 1 |
| 12,391 | | | 1 |
| 12,392 | | | 1 |

*Output* dengan opsi **List Gap Ranges**

*Output* dengan opsi **List Missing Items**

### 7.2.3 Menguji Duplikasi Data (Duplicates)

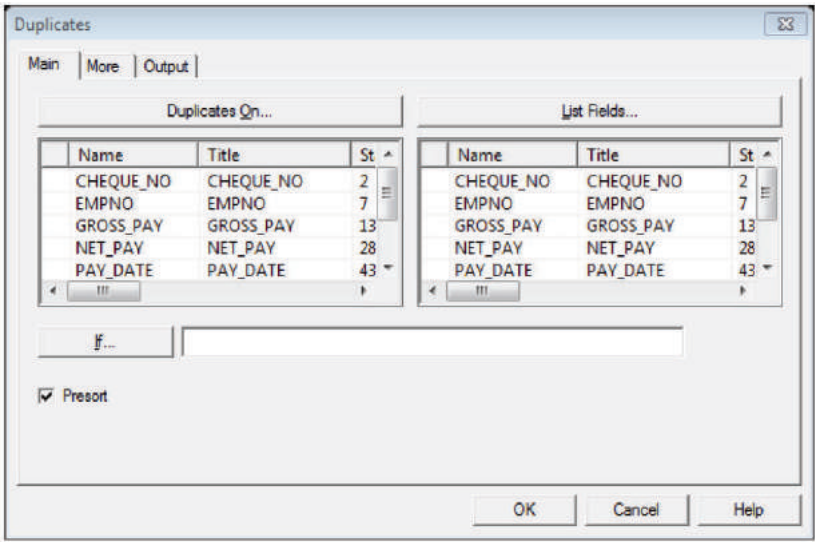
Perintah **Duplicates** digunakan untuk menguji atau mendeteksi apakah *file input* terdapat duplikasi data. Dengan kata lain, perintah ini memeriksa apakah *file input* terdapat data yang terekam lebih dari satu kali.

Misalkan pada *file input* **Payroll** akan diperiksa apakah terdapat data karyawan yang terekam lebih dari satu kali, yaitu dengan memeriksa duplikasi pada nomor karyawan. Jika hal ini terjadi, maka ACL akan menampilkan data nomor karyawan yang terindikasi duplikasi.

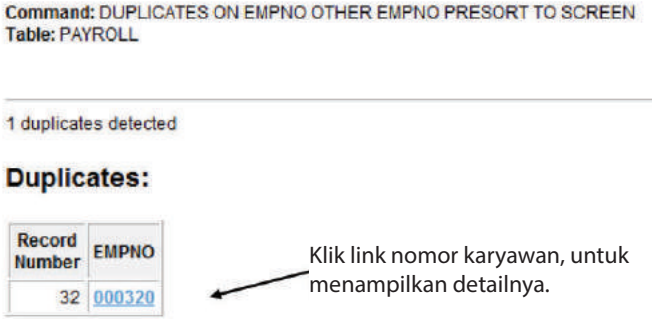
Bukalah *file input* **Payroll** dengan struktur data seperti berikut ini.

| Value | Definition |
|---|---|
| Table Data Source File | D:\Dokumen\Buku 2018\DATA DBF\PAY... |
| Character Set | ASCII |
| Record Length | 68 |
| Skip Length | 257 |
| Number of Fields | 8 |
| Field Name | Data Type |
| RECORD_DELETED | ASCII |
| CHEQUE_NO | ASCII |
| EMPNO | ASCII |
| GROSS_PAY | NUMERIC |
| NET_PAY | NUMERIC |
| PAY_DATE | DATE |
| TAX_AMOUNT | NUMERIC |
| WORKDEPT | ASCII |

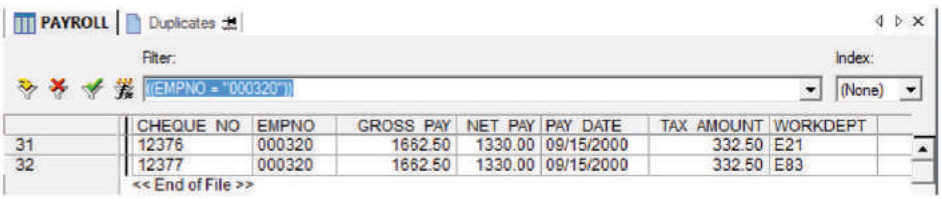
Pada menu **Analyze – Look for Duplicates**.




Pada *tab* **Output**, pastikan opsi **Screen** aktif.
Di dalam daftar **Duplicates On**, klik *field* **EMPNO**.
Di dalam daftar **List Fields**, klik *field* **EMPNO**.
Klik tombol **OK**.




ACL akan menampilkan hasil pemeriksaan duplikasi, yaitu menampilkan nomor *record* dan nomor karayawan yang terdeteksi duplikasi. Untuk melihat data-data mana yang terdeteksi duplikasi, klik *link* data **EMPNO** sehingga ACL akan menampilkan *file input* **Payroll** dengan filter **EMPNO = 000320**, yaitu menampilkan data karyawan dengan nomor "000320" seperti berikut ini.



### 7.2.4 Membuat Tabulasi Silang/Multidimensi (Cross Tabulate)

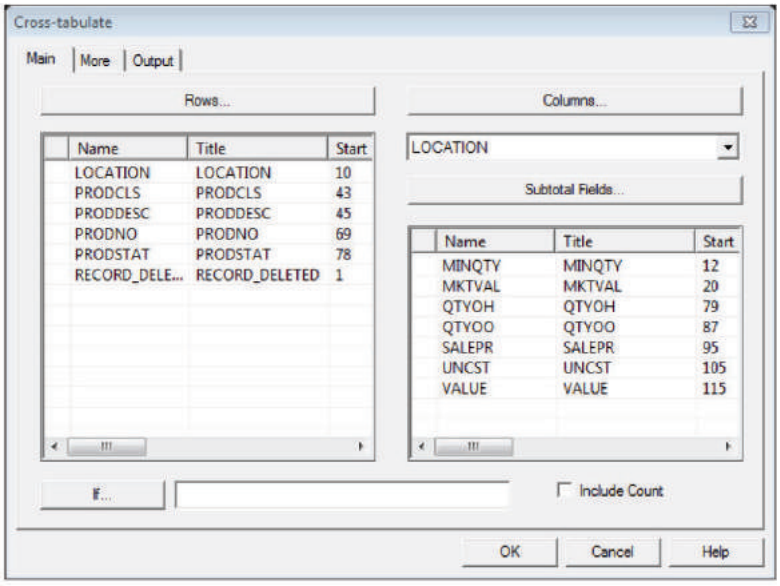
Fasilitas ACL ini hampir sama dengan fasilitas Pivot Table pada MS Excel, yaitu membuat tabel multidimensi dengan menggunakan pengelompokan berdasarkan kategori pada baris beserta kolom.

Sebagai contoh membuat tabel multidimensi yang terdiri dari **Qty**, **Value**, dan **Jumlah Unit Item** pada setiap **Kelas Produk** dan masing-masing diperinci setiap **Lokasi**.

Bukalah *file input* **Inventory** dengan struktur data seperti berikut ini.

| Value | Definition |
|---|---|
| Table Data Source File | D:\Dokumen\Buku 2018\DATA DBF\INVE... |
| Character Set | ASCII |
| Record Length | 129 |
| Skip Length | 481 |
| Number of Fields | 15 |
| Field Name | Data Type |
| RECORD_DELETED | ASCII |
| CSTDTE | DATE |
| LOCATION | ASCII |
| MINQTY | NUMERIC |
| MKTVAL | NUMERIC |
| PRCDTE | DATE |
| PRODCLS | ASCII |
| PRODDESC | ASCII |
| PRODNO | ASCII |
| PRODSTAT | ASCII |
| QTYOH | NUMERIC |
| QTYOO | NUMERIC |
| SALEPR | NUMERIC |
| UNCST | NUMERIC |
| VALUE | NUMERIC |

Pada menu **Analyze – Cross-tabulate.**



Pada *tab* **Output**, pastikan opsi **Screen** aktif.

- Untuk pengelompokan setiap baris data, pada daftar **Rows** pilih **ProdCLS** (kelas produk).
- Untuk pengelompokan setiap kolom data, pada daftar **Columns** pilih **Location**.
- Untuk menghitung subtotal pada setiap kelompok data, pada daftar **Subtotal Fields** pilih **QtyOH** dan **Value**.
- Untuk menghitung jumlah item pada setiap kelompok data baris (kelas produk), klik opsi **Include Count**.

Klik **OK**, ACL akan menampilkan *output* tabulasi silang seperti berikut ini.

Command: CROSSTAB ON PRODCLS COLUMNS LOCATION SUBTOTAL QTYOH VALUE COUNT TO SCREEN
Table: INVENTORI

| PRODCLS | QTYOH LOCATION 01 | VALUE LOCATION 01 | Count 01 | QTYOH LOCATION 02 | VALUE LOCATION 02 | Count 02 | QTYOH LOCATION 03 | VALUE LOCATION 03 | Count 03 | QTYOH LOCATION 04 | VALUE LOCATION 04 | Count 04 | QTYOH LOCATION 05 | VALUE LOCATION 05 | Count 05 | QTYOH LOCATION 06 | VALUE LOCATION 06 | Count 06 | QTYOH LOCATION 22 | VALUE LOCATION 22 | Count 22 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 2,634 | 31,064.68 | 17 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 |
| 02 | 110 | 707.30 | 1 | 2,977 | 19,836.50 | 18 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 |
| 03 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 11,907 | 99,585.24 | 19 | 624 | 3,107.52 | 1 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 |
| 04 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 4,760 | 69,018.95 | 17 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 |
| 05 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 82,466 | 42,479.35 | 13 | 0 | 0.00 | 0 | 0 | 0.00 | 0 |
| 06 | 0 | 0.00 | 0 | 2,792 | 59,458.40 | 16 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 163 | 2,021.20 | 1 |
| 07 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 9,890 | 47,602.10 | 7 | 0 | 0.00 | 0 |
| 08 | 0 | 0.00 | 0 | 31,090 | 160,230.08 | 18 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 |
| 09 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 10,410 | 80,646.05 | 21 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 |
| 13 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 804 | 11,352.48 | 1 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 |
| 18 | 0 | 0.00 | 0 | 1,330 | 4,461.00 | 1 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 | 0 | 0.00 | 0 |
| Totals | 2,744 | 35,961.98 | 18 | 38,517 | 268,988.05 | 54 | 17,471 | 199,965.67 | 37 | 11,034 | 83,753.57 | 22 | 82,466 | 42,479.35 | 13 | 9,890 | 47,602.10 | 7 | 163 | 2,021.20 | 1 |

Tabulasi silang yang dihasilkan ACL selain menampilkan ringkasan data dalam bentuk tabulasi silang, juga menyediakan *link* untuk melihat data lebih detail. Misalkan untuk melihat detail dari data inventori berdasarkan kelas produk "01", pada kolom **PRODCLS** klik link "**01**". Detail dari *link* tersebut adalah berupa daftar data *file input* **Inventory** dengan filter "**PRODCLS = 01**" seperti berikut ini.

INVENTORI | Cross tabulate

Filter: PRODCLS = "01"

Index: (None)

| | QTDATE | LOCATION | MINQTY | MKTVAL | PRGDTE | PRODCLS | PRODDESC | PRGNO | PRODSTAT | QTYOH | QTYOO | SALEPR | UNICST | VALUE |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 50 | 08/18/2000 | 01 | 220 | 1157.10 | 08/31/2000 | 01 | DIET SCALE | 0103111990 | A | 280 | 0 | 3.99 | 2.98 | 834.20 |
| 51 | 08/15/2000 | 01 | 110 | 2658.10 | 08/31/2000 | 01 | BLANCHER | 0103119640 | A | 190 | 200 | 13.99 | 8.00 | 1520.00 |
| 52 | 08/15/2000 | 01 | 40 | 562.60 | 08/31/2000 | 01 | CAKE PAN | 0103131140 | A | 140 | 200 | 3.59 | 3.09 | 432.60 |
| 53 | 08/15/2000 | 01 | 24 | 136.44 | 08/31/2000 | 01 | LOAF PAN | 0103131190 | A | 36 | 100 | 3.79 | 3.10 | 111.60 |
| 54 | 08/15/2000 | 01 | 12 | 169.50 | 08/31/2000 | 01 | VEGETABLE STEAMER | 0103134420 | A | 50 | 100 | 3.99 | 3.12 | 156.00 |
| 55 | 02/10/2000 | 01 | 24 | 767.52 | 08/31/2000 | 01 | CAKE DECORATING SET | 0103226620 | A | 40 | 100 | 15.99 | 10.80 | 518.40 |
| 56 | 02/10/2000 | 01 | 12 | 1150.55 | 08/31/2000 | 01 | ALUMINUM TEAPOT & CUP | 0103227710 | A | 144 | 0 | 7.99 | 5.99 | 862.56 |
| 57 | 04/10/2000 | 01 | 25 | 2467.88 | 08/31/2000 | 01 | DISH DRAINER | 0103511300 | D | 412 | 0 | 5.99 | 5.56 | 2702.72 |
| 58 | 04/18/2000 | 01 | 25 | 1719.14 | 08/31/2000 | 01 | MINCER | 0103708890 | A | 85 | 200 | 19.99 | 14.14 | 1216.04 |
| 59 | 12/20/2000 | 01 | 12 | 3519.36 | 08/31/2000 | 01 | PASTA NOODLE MAKER | 0103711800 | A | 84 | 0 | 54.99 | 24.88 | 1592.32 |
| 60 | 12/20/2000 | 01 | 24 | 335.52 | 08/31/2000 | 01 | 7 PC KITCHEN TOOL SET | 0103803760 | A | 48 | 100 | 6.99 | 3.21 | 154.08 |
| 61 | 08/15/2000 | 01 | 36 | 1714.08 | 08/31/2000 | 01 | STEP-ON CAN | 0103850150 | A | 132 | 0 | 12.99 | 8.40 | 1108.80 |
| 62 | 04/12/2000 | 01 | 48 | 783.44 | 08/31/2000 | 01 | 5 SHELF BREADBOX | 0103855160 | A | 56 | 100 | 13.99 | 9.93 | 556.08 |
| 63 | 08/12/2000 | 01 | 0 | 1055.04 | 08/31/2000 | 01 | 4 PC CANISTER SET | 0103855170 | A | 96 | 100 | 10.99 | 7.05 | 676.80 |
| 64 | 08/12/2000 | 01 | 144 | 1227.48 | 08/31/2000 | 01 | NAPKIN & RELISH HOLDER | 0103877230 | A | 212 | 0 | 5.79 | 3.22 | 682.64 |
| 65 | 11/19/2000 | 01 | 110 | 23992.00 | 08/31/2000 | 01 | PRESSURE COOKER 8QT | 0104102640 | A | 400 | 0 | 64.99 | 39.40 | 15760.00 |
| 66 | 11/19/2000 | 01 | 80 | 9195.40 | 08/31/2000 | 01 | 192 OZ DUTCH OVEN | 0104350000 | A | 230 | 240 | 39.99 | 27.00 | 6348.00 |

<< End of File >>

## 7.3 MANIPULASI DATA

### 7.3.1 Mengekstrak Data

Perintah **Extract** digunakan untuk mengekstrak data, yaitu memilih *record* tertentu dari *file* berukuran besar dengan tujuan agar bisa memproses *record* tersebut dalam

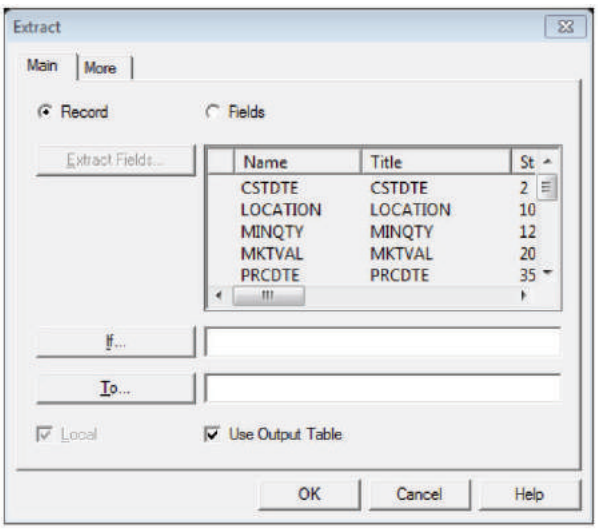
*file* yang lebih kecil. Pengekstrakan *record* atau *field* ke dalam *file* tersendiri akan mengurangi waktu yang diperlukan untuk memprosesnya dan mengurangi ruang (*disk space*) yang diperlukan.

Fungsi ini bertujuan untuk melakukan pemisahan (*extract*) berdasarkan *record* atau *field* dari suatu *file*, kemudian hasil ekstrak disimpan pada *file* yang lain. Hasil *file* ini akan menjadi *input file definition*. Ekstrak berguna agar kita dapat menganalisis data yang lebih kecil dan lebih khusus.

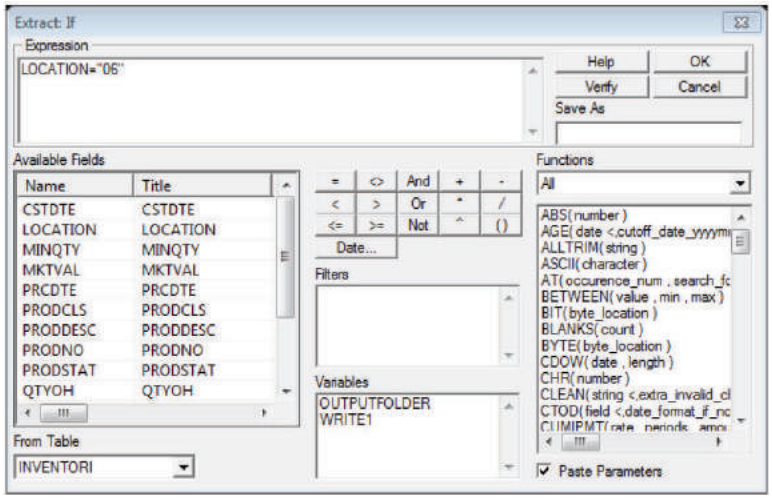
Pada fungsi ekstrak terdapat dua pilihan, yaitu ekstrak berdasarkan *record* dan berdasarkan *field* tertentu yang dipilih. Ekstrak berdasarkan *record* menghasilkan suatu *file* yang berisi semua *field* yang sama dengan *file* sumber apabila tidak terdapat kondisi yang mengikutinya, jika terdapat kondisi yang mengikutinya maka hasil ekstrak adalah semua *field* dengan kondisi tertentu.

### Mengekstrak data berdasarkan record

Misalkan akan mengekstrak data inventori berdasarkan lokasi gudang no "06".
Bukalah *file input* **Inventory** dengan struktur data seperti latihan sebelumnya;
Pada menu **Data – Extract Data ...**

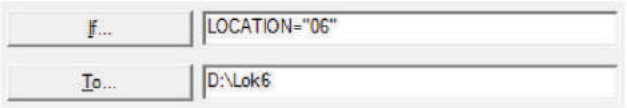



Pada **tab Main** pastikan opsi **Record** terpilih.
Klik tombol **If…** untuk menampilkan kotak dialog **Expression IF**



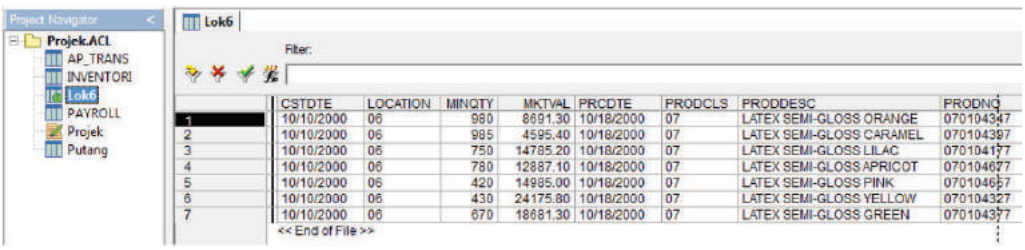

Pada kotak **Expression** isikan kriteria yang akan dijadikan dasar ekstrak, dalam hal ini adalah lokasi produk = 06. Pada daftar **Available Fields** klik dua kali pada *field* **Location** sehingga tulisan *field* "Location" berada di dalam kotak **Expression**, kemudian tambahkan tulisan "= 06", kemudian klik **OK** untuk lanjut.

Klik tombol **To…** untuk memberikan nama *file* hasil ekstrak dan lokasi *file* ekstrak tersebut akan disimpan. Misalkan hasil ekstrak disimpan dengan nama *file* "LOK6", sehingga tampilan kotak dialog **Extract** pada tombol **IF** dan **To** adalah seperti berikut ini.




Klik tombol **OK** untuk mengeksekusi ekstrak data.

ACL akan menghasilkan *file input* baru dengan nama *Lok6* hasil ekstrak data dari *file input* **Inventory** berdasarkan produk yang berlokasi gudang nomor 06.



### Mengekstrak data berdasarkan field

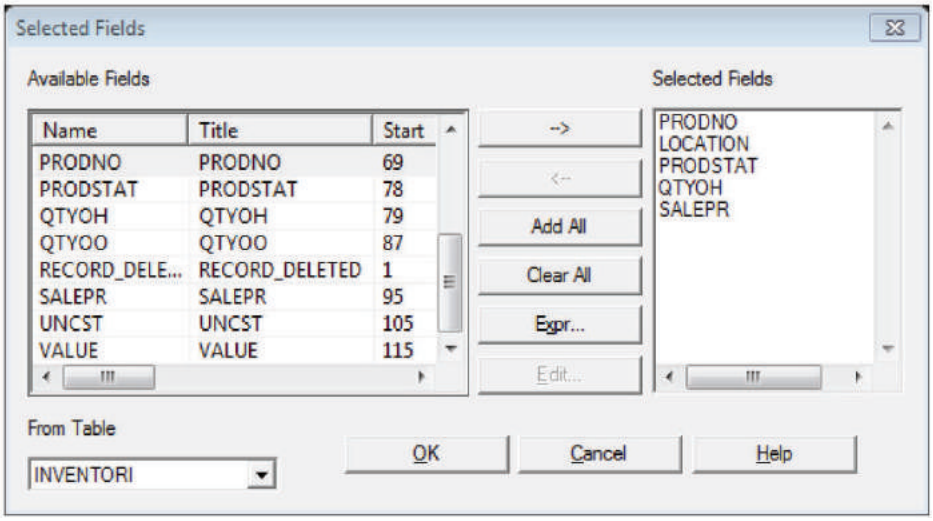
Misalkan akan mengekstrak data inventori berdasarkan lokasi gudang nomor "03" dengan *field* yang dipilih sesuai kebutuhan, misalkan *field* PRODNO, LOC, PRODSTAT, QTYOH, dan SALEPR.

Bukalah *file input* **Inventory** dengan struktur data seperti latihan sebelumnya;

Pada menu **Data – Extract Data …**

Pada *tab* **Main** pastikan opsi **Field** terpilih.

Klik tombol **Extract Fields** untuk menampilkan kotak dialog **Selected Fields** seperti berikut ini.

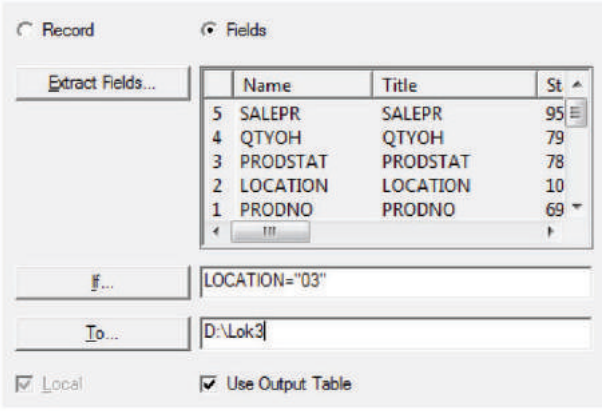



Pada daftar **Avaliable Fields** (kiri), klik dua kali pada *field* yang akan dipilih untuk ditempatkan ke dalam daftar **Selected Fields** (kanan). Klik **OK** untuk keluar dari pemilihan *field*.

Klik tombol ***If...*** untuk menampilkan kotak dialog **Expression IF**.

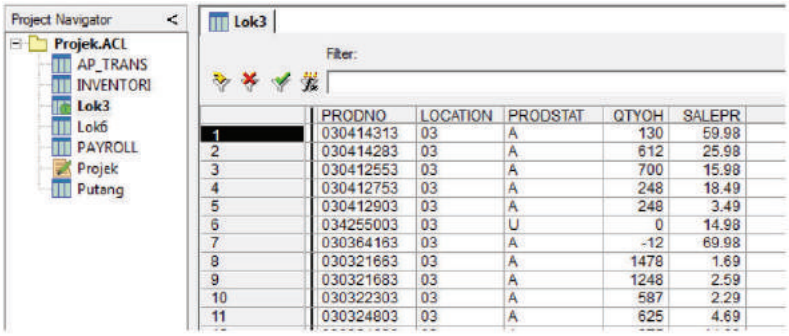
Pada kotak **Expression** isikan kriteria yang akan dijadikan dasar ekstrak, dalam hal ini adalah lokasi produk = 03. Pada daftar **Available Fields** klik dua kali pada *field* **Location** sehingga tulisan *field* "Location" berada di dalam kotak **Expression**, kemudian tambahkan tulisan "= 03", lalu klik **OK** untuk lanjut.

Klik tombol **To…** untuk memberikan nama *file* hasil ekstrak dan lokasi *file* ekstrak tersebut akan disimpan. Misalkan hasil ekstrak disimpan dengan nama *file* "LOK3", sehingga tampilan kotak dialog **Extract** pada tombol **IF** dan **To** adalah seperti berikut ini.




Klik tombol **OK** untuk mengeksekusi ekstrak data.

ACL akan menghasilkan *file input* baru dengan nama Lok3 hasil ekstrak data dari *file input* Inventori berdasarkan produk yang berlokasi gudang nomor 3 yang terdiri dari *field* PRODNO, LOCATION, PRODSTA, QTYOH, dan SALEPR.




### 7.3.2 Mengekspor Data

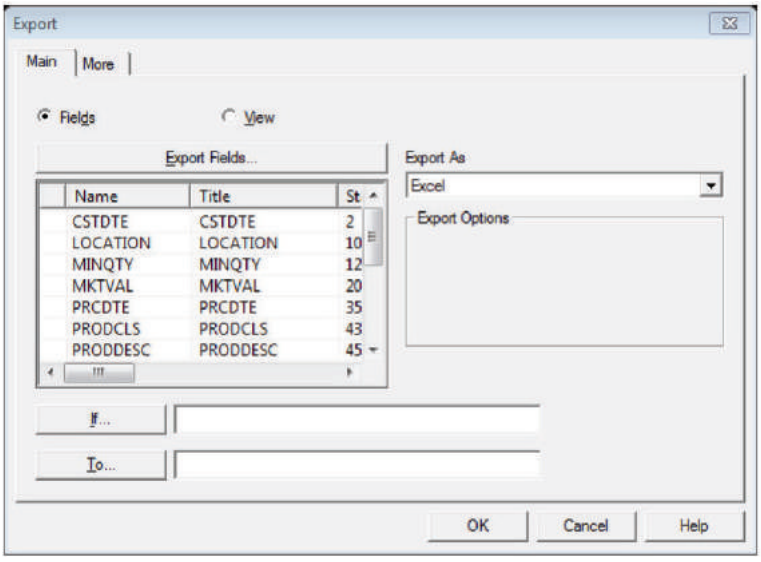
Anda dapat mengekspor data dari ACL ke dalam paket perangkat lunak (*software package*), seperti Microsoft Word atau Microsof Excel, untuk diproses atau diolah lebih lanjut. Apabila Anda menggunakan perintah **Export**, ACL akan mengonversikan data ke dalam format yang dapat dibaca (*readable*) oleh paket perangkat lunak lain dan menyimpannya (*save*) di dalam *file* ekspor. Format *file* yang didukung oleh ACL adalah Delimited Text, Dbase III plus, Lotus 123, MS Excel, MS Access, MS Word, Plain Text, Windows Clipboard, Word Perfect, dan XML.

Sebagai contoh kasus akan digunakan *file input* Inventori dengan struktur *file* seperti pada latihan sebelumnya.

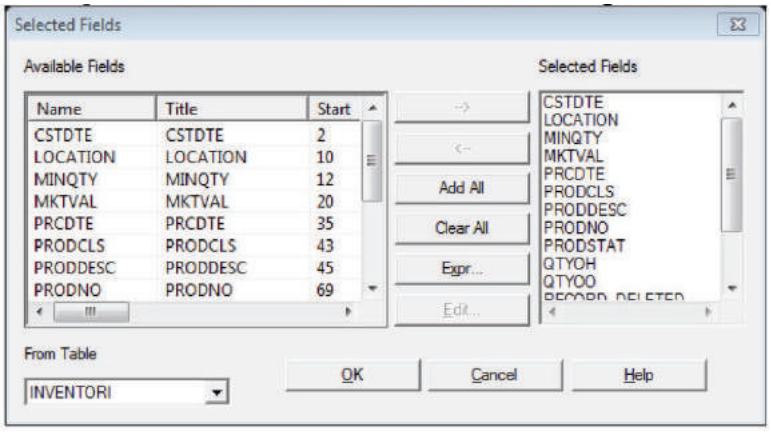
#### Mengekspor (Exporting) seluruh Field

Lakukan langkah-langkah sebagai berikut.

1. Pilih menu **Data** – **Export to Other Application.** ACL akan menampilkan kotak dialog seperti berikut ini.




2. Pada *tab* **Main**, pastikan opsi **Fields** terpilih atau aktif.
3. Klik tombol **Export Fields** untuk membuka kotak dialog **Selected Fields**.




Pastikan semua *field* terpilih, yaitu dengan mengklik **Add All** sehingga semua *field* yang ada di daftar **Available Fields** disalin (*copy*) ke dalam daftar **Selected Fields**.

4. Klik **OK** untuk lanjut kembali ke kotak dialog **Export**.
5. Pada daftar **Export As**, pilih salah satu nama aplikasi yang akan dijadikan tujuan. Misalkan, pilih *Excel*.

6. Klik tombol **To…**, tentukan lokasi *folder* tempat menampung hasil ekspor, kemudian ketikan nama *file* hasil ekspor pada teks isian. Misalkan *DataBarang.xls*
7. Klik **OK**, ACL akan menampilkan *command* dan informasi hasil ekspor data seperti berikut ini.

```
Command: EXPORT FIELDS CSTDTE LOCATION MINQTY MKTVAL PRCDTE PRODCLS PRODDESC
PRODNO PRODSTAT QTYOH QTYOO RECORD_DELETED SALEPR UNCST VALUE EXCEL TO
"D:\databarang"

23:47:07 - 04/02/2018
  152 records produced
  Output to D:\databarang.XLS is done
```

# DAFTAR PUSTAKA


____. 2006. ACL Getting Started.

____. 2006. ACL in Practice.

American Institute of Certified Public Accountants (AICPA). 2012. *Statements on Auditing Standards.* ISACA.

Hall, James A. dan Tommie Singleton. 2005. *Information Technology Auditing and Assurance,* 2nd Edition. Mason, Ohio: Thomson South-Western.

IAI. 2014. *Standar Profesional Akuntan Publik.*

IAI. 2015, *Pernyataan Standar Akuntansi Keuangan.*

Weber, R.1999. *Information Systems Control and Audit*. NJ, United State: Prentice-Hall.

# INDEKS

## C



## D



## E



## F



## G



## H



## I



## J



## K

## R



## S

## T



## U



## V