

TUGAS KELOMPOK

MK. Pengantar Komputer

Dosen

: Toto Haryanto

PEMROSESSAN TRANSAKSI

Kelompok 3 :

Dicky Tri Utama	D14090003
Dyah Prawesthi	D14090032
Edy Susanto	D14090039
Reza Hanifah	D14090073
Darojat Ulil Amri	D14090112



**DEPARTEMEN ILMU PRODUKSI DAN TEKNOLOGI PETERNAKAN
FAKULTAS PETERNAKAN
INSTITUT PERTANIAN BOGOR
2011**

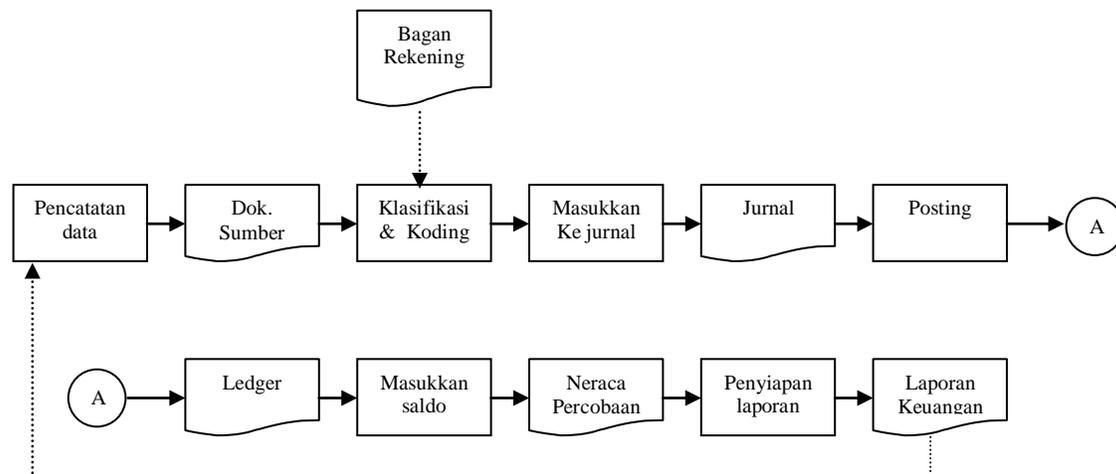
Pemrosesan Transaksi

Fungsi dari IS adalah untuk mendukung kegiatan operasional perusahaan sehari-hari.

Tujuan tersebut dapat dicapai dengan cara :

1. Memproses transaksi yang timbul dari sumber ekstern dan sumber intern.
2. Menyiapkan output seperti dokumen operasional atau laporan keuangan.

Kedua kegiatan diatas dikenal sebagai kegiatan **pemrosesan transaksi**.



Bagan Siklus Akuntansi

Sebelum transaksi diproses, kita terlebih dahulu harus mengidentifikasi data-data transaksi yang dibutuhkan dalam suatu laporan keuangan. Kita tidak perlu mengumpulkan transaksi yang tidak dibutuhkan oleh suatu laporan keuangan. Jadi kesimpulannya kita memulai pemrosesan suatu transaksi akuntansi harus diawali mulai dari atas bukan dari bawah (laporan keuangan). Dimulai dari desain laporan keuangan, kemudian pembuatan bagan rekening, setelah itu diidentifikasi berdasarkan siklus transaksi, dan yang terakhir baru dibuat ayat jurnalnya.

Bagan Rekening

→ Daftar yang menguraikan semua rekening yang digunakan dalam pembuatan laporan keuangan.

Kegiatan pembuatan bagan arus tersebut tidak terlepas dari kegiatan pengkodean.

Pengkodean

Untuk membantu dalam pengumpulan maupun pemrosesan transaksi biasanya diberi suatu kode. Dimana kode adalah suatu kerangka yang menggunakan huruf atau angka (atau kombinasi keduanya) untuk memberikan tanda terhadap klasifikasi yang sebelumnya telah dibuat (bagan rekening).

Tujuan Pengkodean

- Mengidentifikasi data akuntansi secara unik
- Meringkas data
- Mengklasifikasikan rekening/transaksi
- Menyampaikan makna tertentu

Metode Pengkodean:

Kode angka atau huruf berurutan.

Dalam metode ini tiap rekening diberi kode angka atau huruf secara berurutan.

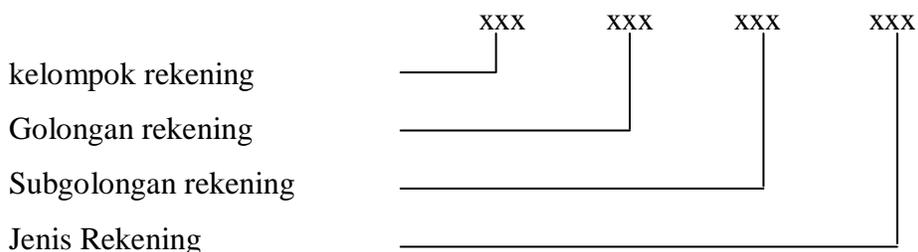
Kode angka blok

Dalam pemberian kode ini rekening dikelompokkan menjadi beberapa golongan dan tiap golongan disediakan satu blok angka yang berurutan untuk memberi kodenya.

Kode angka kelompok

Kode kelompok diberikan dengan memberikan nomor kode untuk setiap kelompok, golongan, subgolongan dan jenis rekening.

Klasifikasi :



Kode Mnemonic

Pemberian kode dengan menggunakan Huruf yang disingkat mendekati bentuk aslinya, misalnya, Aktiva lancar = AL, Jurnal Penjualan = JP, dsb.

Hal-hal yang perlu dipertimbangkan dalam pemberian Kode :

- ① Setiap kode harus mewakili secara unik unsur yang diberi kode.
- ② Desain kode harus mudah disesuaikan dengan tuntutan perubahan.

Pengklasifikasian Transaksi

Karena tingkat arus transaksi dalam suatu perusahaan sangat kompleks, untuk mempermudah dalam penyajiannya, maka tiap transaksi diklasifikasikan ke dalam beberapa siklus-siklus transaksi. Siklus transaksi mengelompokkan satu atau lebih transaksi yang mempunyai kesamaan tujuan. Siklus transaksi untuk satu perusahaan dengan perusahaan lain akan berbeda, disini diberi contoh siklus transaksi perusahaan dagang dan perusahaan manufaktur.

<i>Perusahaan Manufaktur</i>	<i>Perusahaan Dagang</i>
- Pendapatan	- Pendapatan
- Pengeluaran	- pengeluaran
- Produksi	- Manajemen Sumberdaya
- Keuangan	- laporan keuangan

Dari pengklasifikasian tersebut nantinya dapat dengan mudah dibuat suatu bagan rekening.

Komponen pemrosesan transaksi

Seperti layaknya suatu sistem, komponen pemrosesan terdiri dari *Input, Proses, Penyimpanan, Output*.

Input

Input dalam suatu proses transaksi adalah dokumen sumber yang dapat berupa formulir atau bukti transaksi lainnya. Sebelum suatu transaksi diproses terlebih dahulu kita harus melakukan pengumpulan data transaksi. Pengumpulan data-data transaksi ini tidak dapat dipisahkan dari desain suatu formulir, sebab suatu formulir merupakan gambaran atau rekaman dari suatu transaksi.

Tujuan dari formulir :

1. Formulir dibuat untuk meminta dilakukannya suatu tindakan.
2. Formulir digunakan untuk mencatat tindakan yang telah dilaksanakan.

Kegiatan yang berhubungan dengan penggunaan formulir biasa disebut sebagai **Record Management**.

Pertimbangan dalam merancang formulir :

1. Menentukan kebutuhan formulir.
2. Merencanakan formulir yang akan dibuat.
3. Menentukan kuantitas kebutuhan formulir.
4. Mengawasi penggunaan formulir.
5. Menentukan jangka waktu penyimpanan dan pemusnahan.
6. Menentukan alat untuk meyortir dan menyimpan formulir.

Proses

Dalam sistem manual, proses disini terdiri dari kegiatan memasukkan data transaksi kedalam **jurnal**. Dalam sistem komputer, prosesnya dilakukan dengan memasukkan data kedalam **file transaksi**.

Jenis jurnal :

Jika perusahaan masih dalam skala kecil, maka dapat digunakan jurnal umum, tapi jika perusahaan mulai membesar dan aktivitas perusahaan bertambah, tidak dapat lagi digunakan jurnal umum, harus digunakan jurnal khusus. Misalnya, Jurnal pembelian, jurnal penjualan, jurnal penerimaan kas, jurnal pengeluaran kas.

Langkah Perancangan Jurnal

1. Identifikasi karakteristik transaksi.
2. Buat jurnal standar.
3. Merancang jurnal (kolomnya) berdasarkan jurnal standar.

Penyimpanan

Media penyimpanan dari transaksi secara manual adalah Buku Besar. Buku besar ini menyediakan ikhtisar transaksi-transaksi keuangan perusahaan. Proses memasukkan data dari jurnal kedalam buku besar disebut **“POSTING”**.

Untuk sistem komputer, posting ini dilakukan dengan mengup-date file master menggunakan file transaksi.

Macam-Macam File penyimpanan :

1. Master File

⇒ Merupakan kumpulan catatan(record) yang bersifat tetap dan berisi data yang selalu disesuaikan dengan keadaan. Dalam operasi manual master file setara dengan Buku Besar dan Buku Besar Pembantu.

2. File Transaksi

⇒ Kumpulan catatan transaksi yang terjadi yang digunakan untuk up-date master file. Dalam operasi manual file transaksi ini sama dengan Jurnal.

3. File Indeks

⇒ Merupakan master file yang berisi data yang digunakan dalam proses menyesuaikan suatu master file. C/ : file pelanggan (berisi No.Pelanggan, alamat, maksimum kredit, dll), digunakan sebagai petunjuk untuk menyesuaikan file piutang (master file).

4. File Tabel

⇒ Suatu master file yang berisi data yang digunakan sebagai referens dalam memproses suatu file. Biasanya berisi data yang bersifat tetap yang digunakan dalam perhitungan-perhitungan, seperti file gaji karyawan yang digunakan untuk menyusun daftar gaji, file tarif pajak penghasilan yang digunakan untuk menghitung potongan pajak penghasilan karyawan.

Keluaran

Terdapat berbagai macam jenis keluaran yang dihasilkan dari proses transaksi, antara lain : Laporan keuangan, Laporan Operasional, Dokumen Pengiriman, faktur, dsb.

Metode pemrosesan Transaksi

1. Batch Processing (Pemrosesan Kelompok)

2. On-Line Processing (Pemrosesan Langsung)

Contoh Kode Blok :

Kelompok

100 – 199	Aktiva
200 – 149	Utang
250 – 299	Modal
300 – 399	Pendapatan
400 – 899	Biaya Usaha
900 – 999	Penghasilan & Rugi Di luar Usaha

Setiap kelompok akan dipecah lagi menjadi golongan :

100 – 149	Aktiva Lancar
150 – 159	investasi jangka Panjang
160 – 179	Aktiva Tetap Berwujud
180 – 189	Aktiva Tetap Tidak Berwujud

Setiap Golongan dipecah lagi menjadi Sub Golongan :

100 – 109	Kas & Bank
110 – 119	Piutang
120 – 139	Persediaan Barang

Setiap Sub Golongan dirinci menjadi jenis rekening :

100	Kas
101	Kas Kecil
102	Bank BDN
103	Bank BNI

Contoh Kode kelompok

1	Aktiva
1.1	Aktiva Lancar
1.1.1	Kas & Bank
1.1.1.1	Kas
1.1.1.2	Kas Kecil
1.1.1.3	Bank BDN

- 1.1.1.4 Bank BNI
- 1.1.1.11 Bank BCA

Contoh Kode Desimal

Hampir sama dengan kelompok tapi tidak ada kode kelompok yang lebih dari satu digit, misalnya 1.1.1.11 tidak bisa dipakai dalam kode desimal.

Pengertian Transaksi

Cat: Transaksi adalah satu atau beberapa aksi program aplikasi yang mengakses/mengubah isi basis data.

Transaksi merupakan bagian dari pengeksekusian sebuah program yang melakukan pengaksesan basis data dan bahkan juga melakukan serangkaian perubahan data. DBMS yang kita gunakan harus menjamin bahwa setiap transaksi harus dapat dikerjakan secara utuh atau tidak sama sekali. Tidak boleh ada transaksi yang hanya dikerjakan sebagian, karena dapat menyebabkan inkonsistensi basis data. Untuk itu transaksi selalu merubah basis data dari satu kondisi konsisten ke kondisi konsisten lain.

Sebuah transaksi berpeluang untuk ‘menggangu’ integritas basis data yang dapat membuat kondisi/hubungan antar data tidak seperti seharusnya. Untuk menjamin agar integritas dapat tetap terpelihara maka setiap transaksi harus memiliki sifat-sifat:

1. **Atomik**, dimana semua operasi dalam transaksi dapat dikerjakan seluruhnya atau tidak sama sekali.
2. **Konsisten**, dimana eksekusi transaksi secara tunggal harus dapat menjamin data tetap konsisten setelah transaksi berakhir.
3. **Terisolasi**, jika pada sebuah sistem basis data terdapat sejumlah transaksi yang dilaksanakan secara bersamaan, maka semua transaksi yang dilaksanakan pada saat yang bersamaan tersebut harus dapat dimulai dan bisa berakhir.
4. **Bertahan**, dimana perubahan data yang terjadi setelah sebuah transaksi berakhir dengan baik, harus dapat bertahan bahkan jika seandainya sistem menjadi mati.

Terhentinya suatu transaksi tidak selalu diakibatkan oleh kegagalan insidental baik dari perangkat keras (crash) ataupun kemacetan sistem operasi (hang). Tapi lebih sering terjadi karena user sengaja menghentikan transaksi atau karena penghentian transaksi oleh DBMS akibat adanya kondisi tak diinginkan, seperti deadlock atau timeout.

Sebuah transaksi dapat menghasilkan dua kemungkinan:

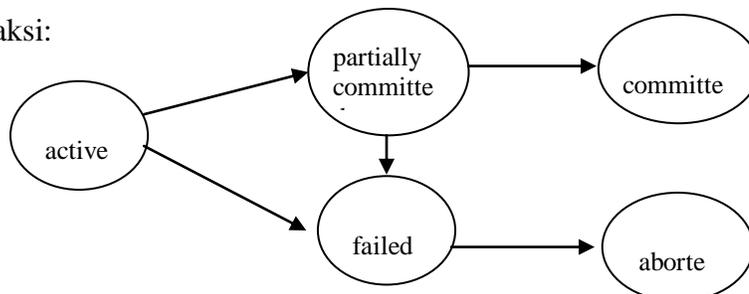
- a. Jika dilaksanakan lengkap seluruhnya, transaksi tersebut telah di *commit* dan basis data mencapai keadaan konsisten baru.
- b. Jika transaksi tidak sukses, maka transaksi dibatalkan dan basis data dikembalikan ke keadaan konsisten sebelumnya (*rollback*).

Transaksi yang sudah di commit tidak dapat dibatalkan lagi. Jika ada kesalahan, maka harus dilakukan transaksi lain yang membalik dampak transaksi sebelumnya.

Status-status yang dapat dicapai oleh sebuah transaksi sejak mulai dilaksanakan hingga selesai atau batal adalah:

1. **Aktif (Active)**, yang merupakan status awal (initial state) sebuah transaksi yang menunjukkan transaksi tersebut masih dieksekusi.
2. **Berhasil Sebagian (Partially Committed)**, yaitu keadaan yang dicapai transaksi tepat pada saat operasi terakhir dalam transaksi selesai dikerjakan.
3. **Gagal (Failed)**, yang merupakan keadaan dimana sebuah transaksi terhenti pengeksesuasiannya sebelum tuntas sama sekali.
4. **Batal (Aborted)**, yaitu keadaan dimana sebuah transaksi dianggap tidak/belum dikerjakan yang tentu dengan terlebih dahulu diawali dengan mengembalikan semua data yang telah diubah ke nilai-nilai semula. (yang menjadi tanggung jawab DBMS).
5. **Berhasil Sempurna (Committed)**, keadaan dimana transaksi telah dinyatakan berhasil dikerjakan seluruhnya dan basis data telah merefleksikan perubahan-perubahan yang memang diinginkan transaksi.

Diagram berikut ini menunjukkan aliran dan siklus peralihan status (state) dari sebuah transaksi:



Ketika sebuah transaksi mulai dikerjakan, maka transaksi itu berada dalam status aktif. Jika terjadi penghentian sebelum operasi berakhir, maka transaksi segera beralih ke status gagal/failed. Namun, bila keseluruhan transaksi selesai dikerjakan, maka transaksi itu berada pada status berhasil sebagian/partially committed, dimana perubahan-perubahan data masih berada di dalam memori utama yang bersifat *volatile*/tidak permanen. Transaksi dalam status ini masih mungkin untuk pindah ke status failed, karena ada pembatalan transaksi baik sengaja maupun tidak. Jika tidak beralih ke status failed, maka nilai-nilai data yang ada di memori utama akan direkam ke dalam disk yang bersifat permanen. Begitu proses perekaman selesai, maka transaksi beralih ke status committed. Sementara itu, transaksi yang berada pada status failed, maka DBMS harus menjalankan proses *rollback*. Proses tersebut dapat berupa:

- Mengulangi pelaksanaan transaksi / *restart*, yang dilakukan pada transaksi yang failed akibat kemacetan perangkat keras ataupun perangkat lunak dan bukannya penghentian transaksi secara sengaja oleh user.
- Mematikan transaksi / *kill*, yang dilakukan untuk transaksi yang dihentikan secara sengaja oleh user atau akibat adanya kesalahan logik dalam penulisan aplikasi.

Begitu salah satu dari pilihan proses tersebut selesai dilakukan, maka transaksi berpindah ke status batal (*aborted*). Status berhasil sempurna/*committed* maupun batal/*aborted* merupakan status terminasi, yaitu status akhir dalam pelaksanaan transaksi.

Pemrosesan Transaksi Berbasis Komputer

Penginputan data ke dalam sistem informasi dapat melalui beragam cara:

- dengan merekam data ke dalam sebuah formulir
- dengan menginputkan data langsung ke dalam computer
- dengan sms
- dengan menginputkan data di internet
- dengan barcode scanner
- scanner yang lain

Dokumen Input

Dokumen input adalah formulir atau dokumen sumber, yang datanya akan diinputkan ke dalam sistem informasi. Contoh adalah dokumen penjualan.

Fungsi dokumen input:

- sarana untuk mengotorisasi suatu transaksi
- sarana untuk memicu tindakan yang diharapkan (missal, memicu konsumen untuk membayar)
- refleksi akuntabilitas
- menyediakan data untuk membuat laporan

Desain dokumen input yang baik:

- judul yang bermakna
- marjin
- ukuran font
- warna
- berat kertas
- nomor urut tercetak untuk dokumen yang sensitif

Tampilan layar komputer untuk entry data yang baik:

- warna yang nyaman di mata user
- memudahkan user dalam menginput data
- perhitungan otomatis
- kursor otomatis berpindah ke elemen data yang mesti diinput oleh user
- pesan kesalahan yang mudah dipahami
- menu help yang tersedia untuk membantu user

Proses

Pemrosesan Input:

- off line (biasanya data diinputkan secara batch). Pemrosesan secara batch relative lebih murah dibandingkan dengan pemrosesan secara online
- on line

Dokumentasi Proses

- System flowcharts
- Computer system flowcharts
- (lihat yang sistem object)

Alternatif pendekatan pemrosesan data:

- Periodic processing approach (sistem batch)
 - Sequential updating from batched input
 - Sequential updating after online data entry
 - Direct updating
- Immediate processing approach (real time processing)

Perbandingan antara periodic processing approach dan immediate processing approach.

Periodic processing approach

Keuntungan:

meningkatkan efisiensi pemrosesan data, khususnya pada saat perusahaan mesti memproses batch yang sangat besar memungkinkan pengendalian menggunakan total batch untuk menjamin keakuratan dan kelengkapan data yang diproses. Pemanfaatan computer hardware dan software secara ekonomis

Kelemahan:

data dalam master file adakalanya *out of date* jika ada kesalahan dalam transaksi maka koreksi atas kesalahan akan tertunda, dan pembetulan mesti menunggu sampai siklus proses berikutnya tiba

Immediate processing approach

Keuntungan:

mampu menyediakan informasi yang uptodate transaksi dapat segera diedit pada saat transaksi direkam, koreksi atas kesalahan dapat langsung dilakukan tanpa delay

Kelemahan:

membutuhkan hardware dan software yang cukup canggih sistem yang lebih kompleks lebih sulit dan lebih kompleks dalam membangun jejak audit

