BAB III

KEGIATAN SELAMA KERJA PRAKTEK

3.1 Jadwal Kerja Praktek

Sesuai dengan Surat Keputusan PT. TELKOM No. Tel.114/PS000/RE3-DOC.C/2003 tentang panggilan Kerja Praktek, penulis diterima untuk melakukan Penelitian Kerja Lapangan (PKL)/Kerja Praktek di PT. TELKOM pada bulan Juli 2003 di bagian SISFO. Mengingat keterbatasan pembimbing lapangan dan masih aktifnya perkuliahan pada waktu tersebut maka disepakati untuk mengundur jadwal Kerja Praktek. Kerja Praktek dilaksanakan selama satu bulan terhitung mulai hari Senin tanggal 4 Agustus 2003 sampai dengan hari Jum'at tanggal 29 Agustus 2003.

3.2 Cara atauTeknik Kerja Praktek

Kerja Praktek dilaksanakan berkelompok sebanyak 3 orang. Dibawah bimbingan dari pembing di perusahaan penyusun melakukan observasi dan wawancara secara langsung, juga melakukan pengecekan koneksi server SISFO dan instalasi jaringan. Selain itu pembimbing juga memberikan tugas kepada penyusun untuk dikerjakan di rumah. Tugas tersebut adalah membuat program berbasis PHP yang dapat mengakses data laporan dalam bentuk file Excel dan Access pada jaringan.

Untuk memperoleh data-data dan mendalami tentang objek Kerja Praktek penyusun juga melaksanakan studi literatur. Literatur ini dapat berupa karya tulis, buku-buku, dokumen, file, ataupun media cetak yang menyajikan tulisan yang berkaitan dengan topik Kerja Praktek.

3.3 Data Kerja Praktek

3.3.1 *Backup* Data

3.3.1.1 Konsep Umum *Backup* Data

Peningkatan jumlah dan nilai data telah berkembang sangat pesat, dibandingkan dengan kecepatan peningkatan utilitas *backup*. Kurangnya kesempatan dalam melakukan pembackupan data menambah sulit keadaan, dimana menjadi semakin tergantung pada keberadaan data tersebut.

Perlindungan terhadap data membutuhkan volume data yang sangat besar dengan berbagai media penyimpanan yang sekarang telah tersedia diberbagai toko komputer. Masalah yang timbul pada proses perlindungan data yaitu adanya gangguan dari dalam sistem komputer dengan kata lain yaitu adanya *virus* atau gangguan lainnya seperti *bad sector*. Pembackupan data merupakan salah satu cara yang efektif yang sering dilakukan untuk menghindari resiko kehilangan data juga untuk meminimumkan data yang hilang.

Pada saat data tersebut dibutuhkan, hanya perlu melakukan pengembalian data (*restore*) dari data yang telah di*backup*.

3.3.1.2 Metode *Backup* Data

Ada 2 (dua) cara tradisional dalam melakukan pem*backup*an data, yaitu :

1. Metode *Batch*

   Metode *batch* ini akan melakukan penduplikasikan data pada saat data tidak digunakan. Isi data statik dan seluruh file ditutup. Jika peng*copy*an dilakukan melewati seluruh data, proses *recover* dapat dilakukan secara konsisten pada waktu tertentu.

2. Metode *Forward Recovery*

Pada metode *forward recovery*, penduplikasian terjadi ketika data sedang digunakan. Isi data berubah secara dinamis dan beberapa atau seluruh file dibuka.

3.3.1.3 Strategi Backup Data

Melakukan pembackupan untuk keseluruhan data mungkin tidak diperlukan, Hanya pada data tertentu yang dianggap penting yang dipilih untuk di*backup*, tetapi secara umum data yang harus dilakukan pem*backup*an, antara lain :

1. Keseluruhan file sistem (*full backup*)
2. Sebagian dari file sistem, yaitu :
   a) File yang telah berubah sejak *backup* terakhir (incremental)

Waktu untuk melakukan *backup* data juga tidak harus setiap saat, tetapi kita dapat melakukannya dengan mempertimbangkan :

1. Seberapa seringnya data tersebut berubah

Melakukan pemilihan strategi *backup* data yang akan digunakan dengan membedakan waktu target untuk melakukan *recover* data, kemudian baru memilih teknologi *backup* yang memenuhi kebutuhan tersebut. Pilih antara metode *Batch* atau *Forward Recovery*, dan pastikan kebutuhan terpenuhi.

Setelah strategi *backup* dan *recovery* ditentukan, pemroses harus diletakan di ruang yang dapat memastikan bahwa pendekatan yang strategis untuk *backup* data didapatkan. Permasalahan *backup* dapat berkembang melalui perubahan yang terjadi pada data atau teknologi yang digunakan. Juga, dapat terjadi dengan

adanya perubahan metode *backup* contohnya, dari peng*copy*an data statik ke metode dinamis.

3.3.1.4 Backup Data dengan HP-UX

Dari sekian banyak versi UNIX yang telah beredar, HP-UX salah satunya. HP-UX merupakan system operasi versi lain dari UNIX yang dikembangkan oleh perusahaan Hewlett Packard yang hanya dapat dijalankan pada mesin HP PA-RISC (HP9000 workstations dan serversw36).

Perkembangan HP-UX dimulai pada tahun 1986 pada keluarga HP9000/500, dengan CPU HP Focus, CPU multi system, lebih dari 7 CPUdalam satu box. Hewlett Packard mengenalkan HP 9000/400 yang disebut Apollo, sejak bergabung dengan Motorola, dan pada saat itulah mesin PA-RISC berbasis HP-UX datang. Keluarga HP300 dan HP400 telah didukung menjadi HP-UX 9.10, yang mencakup beberapa hal baru pada HP-UX 10.X

HP-UX mendukung server dari server skala kecil hingga skala besar, dan server database. Sedangkan untuk workstation, HP-UX hanya mendukung workstation yang mendukung UNIX. Sebagai *User Interface*, pada HP-UX dapat menggunakan 2 mode, yaitu : mode teks dan mode grafik. Untuk mode teks menggunakan POSIX shel dan untuk mode grafik menggunakan Visual User Environment (VUE) dan Common Desktop Environment (CDE).

3.3.1.5 Tipe-tipe Backup Data

Pada HP-UX, dapat menentukandan memilih sendiri file-file mana saja yang dibackup. Ada 2 (dua) tipe *backup* yang dapat dilakukan setelah pemilihan file-file yang akan di*backup*, yaitu :

1. *Full Backup*

Full Backup merupakan backup keseluruhan file. Tetapi bukan berarti setiap file yang terdapat pada sistem di*backup*, melainkan membackup seluruh file yang dipilih pada saat terakhir di*backup*.

2. *Incremental Backup*

Incremental backup merupakan pem*backup*an file yang hanya telah berubah atau telah dibuat sejak terakhir kali mem*backup* seluruh file.

3.4 Level Backup

Level *backup* adalah level yang didefinisikan untuk mengidentifikasikan perbedaan derajat dari *Incremental Backup*. Terdapat 10 level *backup* (0-9) yang dapat digunakan. Sebagai contoh, level 0 adalah *full backup*; level 1 mem*backup* file yang berubah sejak akhir *backup* level 0; dan seterusnya.

File **/var/adm/fbackupfiles/dates** berisi informasi tentang kapan terakhir *backup* pada setiap level *backup* dilakukan. Level *backup* akan membantu dalam menentukan strategi *backup* dan pembuatan penjadwalan dalam proses pembackupan.Misalkan serangkaian *backup* yang akan dilakukan sebagai berikut :

a) **Level 0** – *full backup* bulanan

b) **Level 1** – *backup* mingguan setiap jumat

c) **Level 2** – *backup* harian, kecuali jumat

Tabel III.1 Level *Backup*

Tanggal	1	2	3	4	5	6	7	8	9	10	11	12	13	..	1
Hari	M	Sn	Sl	R	K	J	Sb	M	Sn	Sl	R	K	J	..	
Level Backup	0	2	2	2	2	1	2	2	2	2	2		1		0

Jika data rusak pada hari Kamis tanggal 12, maka pada hari Jumat tanggal 13 yang harus dilakukan untuk *merestore* data hingga pada data hari Rabu tanggal 11, yaitu :

1. *Merestore* data bulanan dari tape *full backup* hari Minggu tanggal 1.
2. *Merestore* data mingguan dari tape *incremental backup* hari Jumat tanggal 6.
3. Kemudian *merestore* data *incremental backup* dari hari Rabu tanggal 11.

3.5 Media Penyimpanan

Melakukan pembackupan data tidak terlepas dari penggunaan media penyimpanan (*storage device*) sebagai tempat data yang dibackup. Terutama jika memilih menggunakan media penyimpanan luar (*external storage device*).

3.6 Tipe-tipe Media Penyimpanan

Media penyimpanan yang akan digunakan adalah media penyimpanan luar (*external storage device*). Tipe-tipe media penyimpanan luar yang dapat digunakan untuk *backup* antara lain :Tape drive dengan format DDS (DAT) atau tape magnetic, *Flexible* (floppy) disk, *Hardisk*, Tape kartidge singel Drive dengan format HP, Tape kartidge dengtan format *Quarter Inch Cartidge* (QIC) dan Tape magnetic *reel to reel*. Setiap media penyimpanan diatas memiliki kelebihan dan kekurangan, untuk lebih jelasnya tentang kriteria pemilihan dapat dilihat pada lembar lampiran.

3.7 Tape Drive DDS (DAT)

Beberapa macam tape drive yang memiliki perbedaan metode perekaman dan format yang dapat dikonfigurasikan pada HP-UX, yaitu : Tape drive *reel to*

reel nine-track, Tape *drive kartidge*, Tape drive QIC, Tape drive DDS(DAT),dan Tape drive 8 mm Exabyte, tetapi yang akan digunakan disini sebagai media penyimpanan *backup* ialah Tape drive DDS (DAT).

Tape drive DDS (DAT) yang akan digunakan pada HP-UX sebaiknnya menggunakan kartidge DDS Hewlett-Packard yang telah dirancang dan dites untuk memberikan maksimum data *reliability* dan *mechanical durability*, sehingga data yang tersimpan pada kartidge DDS dapat bertahan selama 5 - 30 tahun. Tape DDS memiliki 3 (tiga) kapasitas kartidge data, yaitu :

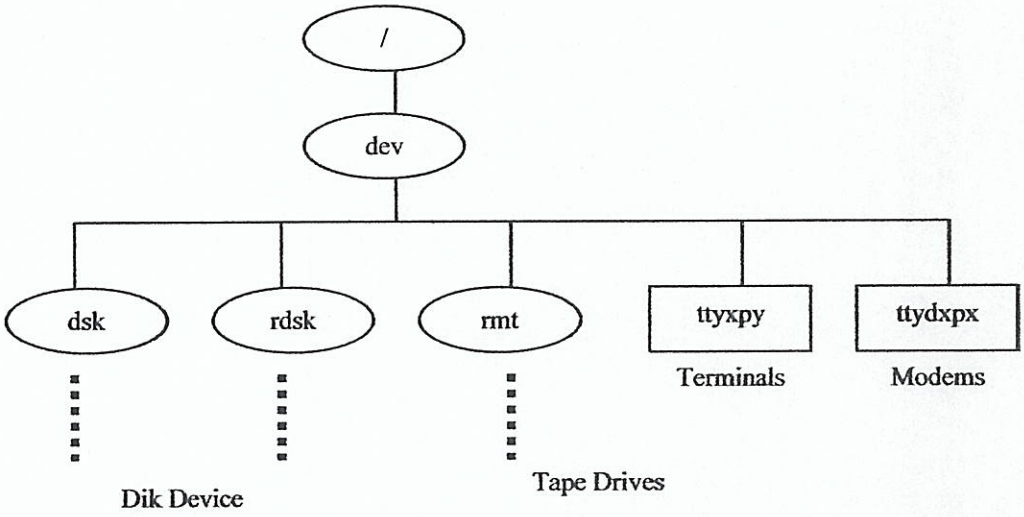


a. 60 meter ≈ 1,3 GB
b. 90 meter ≈ 2,0 GB
c. 120 meter ≈ 4,0 GB

Untuk menghindari resiko kerusakan pada system dan data *backup* yang dimiliki, sebaiknya menyimpan data pada media backup yang tempatnya terpisah dari sistem komputer.

HP-UX berkomunikasi dengan peripheral *device* (seperti tape drive, disk drive, printer, terminal, dan meodem) lewat file yang disebut file *device*. Sebelum HP-UX dapat berkomunikasi dengan periperal *device*, *device* tersebut harus memiliki file *device*. Contohnya, setiap terminal memiliki file *device* tersendiri, dimana HP-UX membaca dan menulis data. File *device* tidak berisi data seperti file regular biasa. Bagaimanapun, file *device* secara sederhana menspesifikasikan bagaimana HP-UX berkomunikasi dengan device. File *device* disimpan pada dierktori **/dev**. File device untuk media penyimpanan *backup* DDS2 pada HP-UX terdapat pada direktori **/dev/rmt.**

Pada banyak sistem HP-UX, seluruh file *device* terdapat dibawah direktori **/dev**. Beberapa file *device* berada pada direktori **/dev**, sementara ada file *device* yang disimpan pada sub direktori dalam direktori **/dev**.



Gambar III.1 Susunan Layout Direktori /dev

Beberapa direktori file *device* yang penting, antara lain :

1) **/dev** : berisi seluruh terminal, modem, lan, dan printer file device.
2) **/dev/dsk** : berisi seluruh blok file disk *device*.
3) **/dev/rdsk** : berisi seluruh file raw disk *device*.
4) **/dev/rmt** : berisi seluruh file tape *device*.

3.8 Konfigurasi HP-UX untuk Tape Drive

Proses pembackupan akan berjalan dengan baik, bila didukung dengan instalasi media *backup* dan pengkonfigurasian tape drive yang benar. Cara yang paling mudah untuk mengkonfigurasi tape drive ialah dengan menggunakan SAM (*System Administtator Manager*). Tetapi jika *command line* yang digunakan, maka hal-hal yang harus diperhatikan, antara lain :

1. Tentukan alamat I/O yang akan digunakan untuk *interface card* tape drive.Perintah **/usr/sbin/ioscan –fn** dapat digunakan untuk melihat alamat I/O.

2. Pilih driver device tape drive dan interface yang digunakan. Jika driver device yang diperlukan tidak ada, maka kernel harus diset ulang.

   a. Ubah direktori ke direktori **/stand/build**. Kemudian eksekusi script sistem preparasi , **system_prep**, yang akan mengekstrak file sistem dari *current file* dan menulis file **system** pada *current* direktori (membuat file **/stand/build/system**).

   b. Edit file **/stand/build/system** untuk menambahkan driver yang tidak ada.

   c. Set kernel dengan menggunakan perintah mk_kernel. Hal ini untuk membuat **/stand/build/vmunix_test**, dan kernel siap untuk diuji.

   d. Save file sistem dan kernel denegan menghapusnya.

```
mv /stand/system   /stand/system.prev
mv/stand/vmunix   /stand/vmunix.prev
```

   e. Pisahkan file sistem dan kernel yang baru, kemudian siap digunakan ketika *mereboot* sistem.

```
mv /stand/build/system  /stand/system.prev
mv/stand/build/vmunix_test   /stand/vmunix.prev
```

3. Kemudian install tape drive. Ketika memasang tape drive, set *switch* ke alamat I/O yang telah dipilih.

4. Nyalakan seluruh peripheral kembali, kemudian baru nyalakan SPU. Ketika mem-*booting*, HP-UX akan mendeteksi tape drive baru dan menyamakannya dengan *device* driver-nya.

5. Pastikan konfigurasi dengan menggunakan perintah **ioscan** untuk mengkonfirmasi bahwa tape *device* terdeteksi dan file *special device* untuk tape drive tersebut telah dibuat.

```
/usr/sbin/ioscan –c tape –fun

Class I H/W Patch   Driver S/W State    H/W Type     Description
=============================================================;
Tape  0 2/0/1.3.0   stape  CLAIMED      DEVICE       HP35480A
                    /dev/rmt/0m         /dev/rmt/c0td0BESTn
                    /dev/rmt/0mn        /dev/rmt/c0td0BESTb
```

6. Perintah **tar** dapat juga digunakan untukmemastikan *read* dan *write* dapat dilakukan ke/dari *device*. Perintah tar pertama menuliskan file **/etc/psswd** ke tape menggunakan file special *device* yang ditunjukan output ioscan. Perintah kedua menampilkan isi tape.

```
/usr/sbin/tar cvf /dev/rmt/c0t3d0BEST /etc/passwd
a /etc/passwd 2 blocks

/usr/sbin/tar cvf /dev/rmt/c0t3d0BEST
r-r-r-2/2 601 June 6 16:40 1994 /etc/passwd
```

3.9 SISKAMAYA 2.3 (Sistem Keamanan Dan Pelayanan)

SISKAMAYA 2.3 merupakan suatu aplikasi yang digunakan untuk pengolahan berbagai data yang sangat penting, dan merupakan alat untuk aktivitas pertukaran data, informasi pelayanan, informasi jaringan maupun pemasangannya, informasi tariff, informasi mutasi dan lain-lain.

Untuk lebih jelasnya kami jabarkan secara garis besar dibawah ini mengenai aktivitas yang dilakukan oleh pihak telkom dalam menggunakan aplikasi SISKAMAYA 2.3 ini dalam bertransaksi informasi.

3.9.1 Tabel-Tabel SISKAMAYA

Di SISKAMAYA terdapat ± 187 tabel yang dipilah dalam sebelas kelompok tabel, yaitu :

1. Kelompok Tabel Referensi
2. Kelompok Tabel Pelanggan
3. Kelompok Tabel Jaringan
4. Kelompok Tabel Tarif
5. Kelompok Tabel Transaksi PSB
6. Kelompok Tabel Transaksi Mutasi
7. Kelompok Tabel Transaksi Gangguan
8. Kelompok Tabel Transaksi *Claim* Pulsa
9. Kelompok Tabel Transaksi *Omzetting*
10. Kelompok Tabel Transaksi *Pending Order*
11. Kelompok Tabel Bantu

Dibawah ini penjelasan mengenai kelompok-kelompok tabel pada SISKAMAYA yang digunakan dalam berbagai transaksi, juga adanya perubahan yang terjadi akibat aktifitas *user* dalam melakukan perubahan atau pengolahan data dan hubungan antar tabel yang saling berkaitan dapat dilihat dibawah ini :.

1. Kelompok Tabel Referensi

Tabel III.2 Tabel Referensi

NAMA TABEL	BERISI DATA	KET
Referensi	Berbagai pengkodean kecil 1 dijit yang ada di berbagai tabel	√
Referensi_ultraphone	Detail statis tentang no.line dari tiap kanal/saluran untuk sambungan ultraphone	√
Referensi_wll	Detail statis tentang no.line dari tiap kanal untuk sambungan wll	√
Status	Referensi untuk status urat, kabel, alat, dan no.sambungan	√
Kode_feature	Referensi untuk feature-feature jasa telekomunikasi	√
Kode_layanan_kring	Jenis-jenis pelayanan untuk kring-nya pemasangan sambungan baru	√
Kode_usaha	Referensi untuk jenis-jenis usaha pelanggan	x

2. Kelompok Tabel Pelanggan

Tabel III.3 Tabel Pelanggan

NAMA TABEL	BERISI DATA	KET
Group_pelnggan	Identifikasi kelompok/group pelanggan jastel	x
Pembayar	Pembayar suatu sambungan telekomunikasi	√
Pelanggan_info	Data tambahan (typical data) untuk pelanggan sambungan telekomunikasi (untuk kebutuhan pembentukan data statistik)	x
Master_peralatan	Master untuk peralatan yang dekat ke sisi pelanggan/sambungan	√
Hunting	Nomor-nomor yang di-hunting dari suatu no.sambungan tertentu	√
Sambungan_info	Tambahan data non teknis per no.sambungan	√
Histori_sambungan	Histori pelaksanaan berbagai transaksi no.sambungan	√
Sambungan_teknis	Data tambahan mengenai peralatan-peralatan yang diberikan pada saat suatu sambungan diaplikasikan pada sisi pelanggan	√
Penumpang_pemakai	Penumpang pemakai sambungan telekomunikasi yang terdaftar di buku telepon. Digunakan juga sebagai acuan informasi 108	√
Alat_tambahan	Data alat_tambahan yang dimiliki oleh suatu sambungan, dilihat dari sisi pelanggan	√
Feature	Dat feature yang dimiliki oleh suatu sambungan pelanggan	√

3. Kelompok Tabel Jaringan

Tabel III.4 Tabel Jaringan

NAMA TABEL	BERISI DATA	KET
Master_sto	Master setiap STO	√
Master_mdf	Master stiap group klem mdf setiap STO	√
Master_kabel	Master kabel primer, kabel skunder, kabel tie atau kabel junction per STO	√
Master_rk	Master sk yang ada di setiap STO	√
Master_base_station	Master base station (untuk teknologi WLL dan Ultraphone) per STO	√
Master_sentral_lt	Master setiap SLT (Sentral Line Termination)	√
Komponen_dpg	Detail dari setiap urat kabel dpg, spt, info slot card SLMX di sentral, slot card AMX di LTC,dll.	√
Detail_pencabangan	Komponen (simpul) antara yang ada di antara komponen asal dan komponen tujuan dalam satu relasi	√
Urat_rusak	Urat-urat rusak dari satu kabel	√

4. Kelompok Tabel Tarif

Tabel III.5 Tabel Tarif

NAMA TABEL	BERISI DATA	KET
Tarif_master	Jenis-jenis pelayanan yang dikenai tarif yang sedang berlaku	√
Tarif_komponen	Parameter yang terlibat dan nilainya, yang menentukan/membedakan pentarifan	√
Tarif_biaya	Jenis biaya yang dikenai tarif dan nilainya	√
Tarif_komponen_histori	Data parameter dan nilai penentu tarif yang tidak berlaku lagi	√
Potongan_tarif	Besarnya potongan (jumlah dan prosentasinya) untuk tiap jenis pelayanan/biaya	√
Kode_tarif	Referensi kode tarif dan deskripsinya	√
Kode_komponen_tarif	Referensi kode komponen/parameter tarif dan deskripsinya	√
Tarif_biaya_histori	jenis biaya dan niali yang dikenai tarif yang tidak berlaku lagi	√
Tarif_master_histori	Jenis-jenis pelayanan yang sudah tidak berlaku lagi	√

5. Kelompok Tabel Transaksi PSB

Tabel III.6 Tabel Transaksi PSB

NAMA TABEL	BERISI DATA	KET
Psb_master_transaksi	Master transaksi setiap terjadinya pasang baru sambungan(PSB)	√
Psb_pembayaran	Besarnya biaya dan data-data pembayaran PSB	√
Psb_info_billing	Data tambahan pelanggan yang kelak dibutuhkan oleh system	x
Psb_detail_transaksi	No.sambungan dan uraiannya yang dialokasikan untuk setiap sambungan PSB	√
Psb_feature	Daftar feature yang dialokasikan untuk setiap sambungan	√
Psb_alat_tambahan	Daftar alat tambahan yang dialokasikan untuk setiap sambungan	√
Psb_hunting	Daftar no.terpanggil (hunting) yang dialokasikan untuk setiap sambungan	
Psb_instalasi	Informasi penginstalasian perangkat di sisi pelanggan	√

6. Kelompok Tabel Transaksi Mutasi

Tabel III.7 Tabel Transaksi Mutasi

NAMA TABEL	BERISI DATA	KET
Mutasi_master_transaksi	Master transaksi setiap terjadinya transaksi mutasi	√
Mutasi_pembayaran	Besarnya biaya dan data pembayaran untuk transaksi Mutasi	√
Mutasi_feature	Daftar feature yang dihapus/ditambah dalam Muatsi Jasnita	√
Mutasi_hunting	Daftar no. terpanggil yang dihapus/ditambah dalam mutasi	√
Muatsi_sistem	Daftar no.sambungan baru untuk ganti sejumlah no.sambungan sekaligus	√
Mutasi_instalasi	Perubahan data instalatur sambungan	√
Mutasi_ekspedisi	Data user dan waktu penanganan transaksi mutasi	
Muatsi_jenis	Jenis-jenis Mutasi yang dapat ditangani dan status keterlibatan user	√
Muasi_no_urut	No.urut transaksi Mutasi yang akan datang	√
Muatsi_batal	Transaksi mutasi yang dibatalkan	√
Mutasi_work_order	Status dan urutan keterlibatan user dalam penanganan transaksi Mutasi	
Mutasi_master_histori	Histori dari tabel mutasi_master_transaksi	
Muatsi_detail_histori	Histori dari tabel mutasi_detail_transaksi dan mutasi_ekspedisi	

7. Kelompok Tabel Transaksi Gangguan

Tabel III.8 Tabel Transaksi Gangguan

NAMA TABEL	BERISI DATA	KET
Gangguan_master_transaksi	Master transaksi setiap terjadinya transaksi gangguan	√
Gangguan_pengaduan_berulang	Catatan tentang pengulangan pengaduan oleh pelanggan	√
Gangguan_ekspedisi	Data user dan waktu penanganan transaksi gangguan	√
Gangguan_ekspedisi_ulang	Detail dari adanya pengulangan penanganan gangguan	√
Gangguan_detail_massal	Daftar no.sambungan yang terkena gangguan masal	√
Gangguan_master_massal	Master transaksi setiap terjadinya transaksi gangguan massal	√

8. Kelompok Tabel Transaksi *Claim* Pulsa

Tabel III.9 Tabel Transaksi *Claim Pulsa*

NAMA TABEL	BERISI DATA	KET
Cp_awal_transaksi	Pengaduan yang selesai dengan penjelasan	√
Cp_master_transaksi	Pengaduan yang akan di proses	√
Cp_detail_sentral	Data hasil cek sentral	x
Cp_detail_jarlok	Data hasil cek jaringan	√
Cp_nota	Pembuatan nota dinas	√
Cp_mob	Permintaan mob	√
Cp_mob_info	Tambahan informasi tentang mob	
Cp_hasil_investigasi	Hasil investigasi terhadap adanya klaim tagihan dari pelanggan	√

9. Kelompok Tabel Transaksi *Omzetting*

Tabel III.10 Tabel Transaksi *Omzetting*

NAMA TABEL	BERISI DATA	KET
Omzet_transaksi	Berisi data utama untuk omzetting (transaksi perubahan jaringan untuk no.sambungan tertentu)	√
Omzet_work_order	Berisi data utama work order (keterlibatan) user dalam transaksi	√
Omzet_ekspedisi	Berisi data utama pekerjaan user dalam transaksi omzetting	√
Omzet_histori	Histori dari tabel omzet_transaksi dan omzet_ekspedisi	√
Omzet_no_urut	No.urut transaksi omzetting yang akan datang	√

10. Kelompok Tabel Transaksi Pending Order

Tabel III.11 Tabel Transaksi *Pending Order*

NAMA TABEL	BERISI DATA	KET
Psb_datateknis_tel25po	Detail teknis perubahan data jaringan melalui transaksi Pending Order	√
Psb_kronologis_tel25po	Waktu dan identitas user yang menangani transaksi Pending Order	√
Psb_work_order_tel25po	Status dan urutan penanganan transaksi Pending Order	√
Psb_histori_datateknis_tel25po	Histori dari tabel psb_datateknis_tel25po	√
Psb_histori_kronologis_tel25po	Histori dari tabel_kronologis_tel25po	√
Psb_histori_work_order_tel25po	Histori dari table psb_work_order_tel25po	√

12. Kelompok Tabel Transaksi

Tabel III.12 Tabel Transaksi

NAMA TABEL	BERISI DATA	KET
Referensi_notel_pinjaman	Daftar no.sambungan milik suatu STO yang digunakan oleh STO lain	√
Status_sambungan	Status sebuah no.sambungan apakah sedang di-Isolir, di-Cabut, dan di PSK	√
Status_sambungan_histori	Histori dari table status_sambungan	√
Service_point_sto	Hubungan antar kode service point dengan kode_sto yang dapat dilayani oleh service point tersebut	√
Sto_sambungan	Hubungan antar kode_sto dan kepala no.sambungan yang dicatu oleh STO tersebut	√
Kode_wilayah	Deskripsi wilayah berdasarkan kode wilayah	√
Dinas	Pilihan data dinas di Telkom	
Sto_telepon	No.sambungan untuk setiap pos sto	√

3.9.2 Aktivitas User dan Perubahan Data

Bagian ini berisi informasi perubahan data yang terjadi pada basis data SISKAMAYA sebagai akibat adanya aktivitas user SISKAMAYA.

Pemakaian aplikasi SISKAMAYA oleh para user, kecuali untuk pengaksesan laporan-laporan, hampir selalu berdampak pada terjadinya perubahan data SISKAMAYA. Berikut ini akan diuraikan perubahan data yang terjadi dalam basis data akibat adanya interaksi (aktivitas) user yang memanfaatkan berbagai modul di SISKAMAYA.

3.9.2.1 Pemakaian Modul Fasilitas

Uraian pada table berikut ini hanya meliputi aksi entri. Perubahan data akibat aksi koreksi dan hapus (yang disediakan aplikasi) sengaja tidak diuraikan, karena table-tabel yang terpengaruh sama seperti yang terjadi akibat aksi entry, tetapi dalam bentuk manipulasi data yang berbeda yaitu perubahan atau penghapusan data ditabel-tabel tersebut.

UNIT PELAYANAN TEKNIS PERPUSTAKAAN UNIKOM

Tabel III.13 Tabel Perubahan Data

USER	AKTIFITAS(AKSI)	NAMA TABEL	PERUBAHAN YANG TERJADI
Dari menu Pengelolaan Data Jaringan submenu Pembangaunan Basis Data Jaringan			
Advalkab	Entri Data Master MDF	Master_mdf	Tambah data
	Entry Data Rumah Kabel	Master_rk	Tambah data
	Entry Data Segmen Skunder	Master_kabel	Tambah data, hanya untuk entry pertama
		Relasi	Tambah data
		Master_dp	Tambah data
	Entry Siap Pakai Kabel-Tie	Master_kabel	Tambah data, hanya untuk entry pertama
		Master_transaksi_kabel	Tambah data
		Detail_relasi_skunder	Tambah data. Untuk entri berikutnya, terjadi perubahan status, no_kleim & nama_komponen

3.9.3 *Procedure* dan *Rule* Basis Data SISKAMAYA 2.3

Demi peningkatan/pemaksaan integritas basis data, di SISKAMAYA telah disiapkan sejumlah *Rule* dan *Procedure* basis data. *Rule* basis data berisi pendefinisian table, jenis operasi basis data dan *Procedure* basis data. Sedang *Procedure* basis data berisi perintah-perintah SQL yang akan dipanggil atau dijalankan jika operasi terhadap table tersebut, seperti yang disebutkan dalam *rule*.

Tabel III.14 Tabel *Rule* dan *procedure*

NAMA RULE	JENIS OPERASI BASIS DATA	PROSEDUR YANG DIPANGGIL
Rlupdate_relasi	Update on relasi (jml_urat...)	Perubah_jumlah_urat
Rldelete_detail_relasi_primer	Delete on detail_relasi_primer	Prhapus_detail_relasi
Rlupdate_detail_relasi_primer	Update on detail_relasi_primer	Prubah_status_urat
Rldelete_detail_relasi_skunder	Delete on detail_relasi_skunder	Prhapus_detail_relasi
Rdelete_jaringan_terpasang	Delete on jaringan_terpasang	Prhapus_jaringan_terpasang
Rlinsert_jaringan_terpasang	Insert into jaringan_terpasang	Prset_kode_konfigurasi
Rlupdate_psb_detail	Insert into psb_detail_transaksi	Prset_kode_konfigurasi2

3.9.4 Pemeliharaan Integritas Basis Data SISKAMAYA 2.3

Integritas Data merupakan satu hal yang sangat penting yang sudah semestinya senantiasa dipelihara, agar kekuatan informasi dan kebenaran operasi dapat terjamin/tercapai. Dalam prakteknya, validitas data hingga 100% belum sepenuhnya bias tercapai. Hampir selalu terjadi adanya gangguan integritas data yang dihasilkan dari ketidakvalidan data dalam basis data. Ketidakvalidan itu dapat terjadi karena :

1. adanya kesalahan dalam aplikasi,

2. tidak terantisipasinya aktifitas *user* dalam melakukan suatu operasi yang sebenarnya tidak sesuai dengan petunjuk atau
3. karena memang kesalahan *user* dalam melakukan manipulasi (entri, koreksi dan hapus) data.

Ada 5 kelompok Integritas Basis Data yang harus dipelihara, yaitu :

1. Integritas Entitas (Keunikan Data)

   Sebuah *row* data yang disimpan harus memenuhi aturan keunikan data (apakah data harus unik atau boleh tidak unik).

2. Integritas Domain

   Nilai yang disimpan harus memiliki tipe yang sama atau sesuai dengan *field* penampungya. Nilai yang disimpan juga harus berada dalam batas-batas nilai yang diperbolehkan, misalnya data Kode STO harus 5 dijit diawali dengan 2 dijit pertama 50.

3. Integritas Antar Tabel

   Suatu record data atau *row* sebaiknya tidak disimpan di sejumlah table sekaligus.

4. Integritas Referensial

   Keberadaan atau nilai item data dari suatu *row* di sebuah table tergantung pada keberadaan *row* atau item data tsb. Ditabel yang lain.

5. Integritas Aturan Nyata

   Nilai yang disimpan untuk suatu *field* mengikuti aturan tertentu (tergantung sistem), misalnya nilai Biaya untuk PSB Dinas selalu tidak ada (*null*), dan sebagainya.Integritas Aturan Nyata (sebagian) telah diatur

di dokumen yang berisi stuktur table-tabel di SISKAMAYA. Contoh Integritas Aturan Nyata ini adalah :

1) No_sambungan yang bersetaus cabut tidak boleh ada yang berumur > 3 bulan,
2) Dalang merupakan user yang paling akhir dalam menangani transaksi PSB dan Mutasi dan sesudah menangani transaksi harus di-*close*,
3) Urat jaringan yang boleh dicatu untuk pemasangan baru haruslah urat yang kosong baik, dan lain-lain.

3.9.5 Konfigurasi Sistem Operasi

Bagian ini berisi contoh konfigurasi Sistem Operasi HP-UX hpsiska B.10.20 U 9000/879 yang dipakai pada mesin operasional SISKAMAYA.

1. Pemetaan Disk

   Pemetaan harddisk dilakukan mempertimbangkan performasi aplikasi SISKAMAYA 2.3

   1. Kebutuhan *Space* dan lokasi fisik *harddisk,* juga file *sistem*

Tabel 3.15. Tabel Pemetaan Disk

Volume group	Filesystem	Use	Mbytes	Mounted on	Ingress directory
Vg00-4gb	/dev/vg00/lvol1	HFS	48	/stand	
	/dev/vg00/lvol2	Swap/dump	1280		
	/dev/vg00/lvol3	HFS	84	/	
	/dev/vg00/lvol4	HFS	300	/home	
	/dev/vg00/lvol5	HFS	500	/opt	
	/dev/vg00/lvol6	HFS	100	/tmp	
	/dev/vg00/lvol7	HFS	500	/usr	
	/dev/vg00/lvol8	HFS	325	/var	
	/dev/vg00/lvol9	HFS	300	/oping	II_SYSTEM
	Jumlah:		3464		
Vg01-4gb	/dev/vg00/lvol1	HFS	500	/ingwork1	II_WORK
					II_ABF II_LOG_FILE

3.9.6 Konfigurasi Kernel (hpsiska)

Konfigurasi sistem operasi disesuaikan dengan kebutuhan sistem INGRES, untuk itu digunakan beberapa *Configurable Parameters* sebagai pemicu parameter-parameter lain, sehingga INGRES dapat memanfaatkan *resources* yang disediakan sistem operasi.

HP-UX hpsiska B.10.20 U 9000/879 di set sebagai General OLTP atau Database Server System

Tabel III.16. Tabel Parameter pada Server

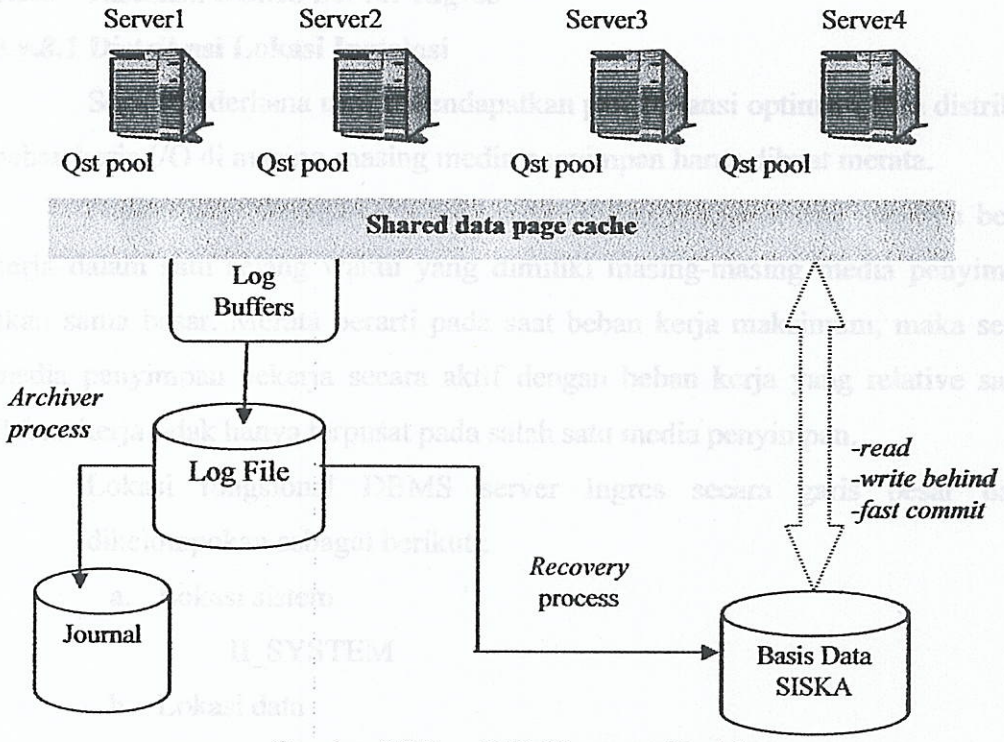
Parameter, bufpages, is now set to "(nproc*3)".
Parameter, dbc_max_pct, is now set to "0"
Parameter, dbc_min_pct, is now set to "0"
Parameter, maxfiles, is now set to"2048"
Parameter, maxswapchunks, is now set to"4096"
Parameter, maxpurc, is now set to"((nproc*9)/10)"

3.9.7 Konfigurasi DBMS

Bagian ini berisi contoh konfihurasi DBMS OpenIngres HP-UX version OI 1.2/01 (hp8.us5/00) yang dipakai pada mesin operasional SISKAMAYA.

3.9.7.1 DBMS Server

1. Untuk menghindari *swap* pada proses DBMS Server maka user Ingres di set sebagai Top CPU User.
2. Hpsiska menggunakan *multiple servers* dengan pertimbangan total MIPS dan jumlah *processor* mendukung untuk konfigurasi *multiple servers*, untuk itu dibuat beban 1 server DBMS untuk 1 prosesor.
3. Konfigurasi *multiple servers* menggunakan *shared cache* dengan pertimbangan karena semua DBMS server mengakses basis data yang sama dan ini diperlukan untuk menjaga konsistensi data.



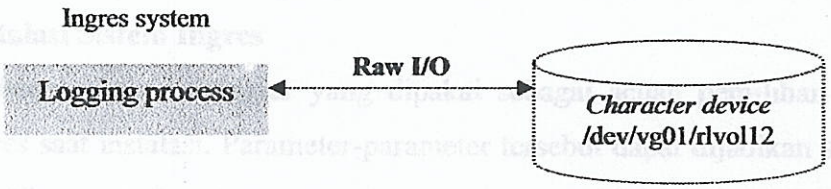
Gambar III.2 – DBMS server Hpsiska

3.9.7.2 Ingres Logging system

Ingres Log File menggunakan *character device* pda /dev/vg01/rlvoll2 sebesar 320 Mb.

Character device digunakan dengan tujuan :

1. Mem-*bypass* UNIX *buffer cache* dan *file system*, data secara langsung di transfer dari server DBMS ke disk,
2. Menghindari *update* pada inode untuk penulisan secara *syhronous* sehingga mengurangi proses I/O.



Gambar III.3 -Ingres *Logging system*

3.9.8 Instalasi DBMS Server Ingres

3.9.8.1 Distribusi Lokasi Instalasi

Secara sederhana untuk mendapatkan performansi optimal maka distribusi beban kerja I/O di masing-masing media penyimpan harus dibuat merata.

Pengertian "meratanya beban kerja" disini bukan berarti rata-rata beban kerja dalam satu selang waktu yang dimiliki masing-masing media penyimpan akan sama besar. Merata berarti pada saat beban kerja maksimum, maka setiap media penyimpan bekerja secara aktif dengan beban kerja yang relative sama. Beban kerja tidak hanya terpusat pada salah satu media penyimpan.

Lokasi fungsional DBMS server Ingres secara garis besar dapat dikelompokan sebagai berikut :

a. Lokasi sistem

   II_SYSTEM

b. Lokasi data

   II_DATABASE (dan juga lokasi definisi tambahan: II_DATA1 dan II_DATA2)

c. Lokasi kerja

   II_ABF, II_WORK, II_CHECKPOINT, II_JOURNAL, II_DUMP, II_TEMPORARY

d. Lokasi transaction log

   II_LOG_FILE, II_DUAL_LOG (pada instalasi ini II_DUAL_LOG tidak diaktifkan karena pertimbangan performansi).

Untuk melihat pemetaan lokasi *mount directory* yang dipakai sistem Ingres ke *logical volume* yang didefinisikan sistem operasi.

3.9.8.2 Instalasi Sistem Ingres

Berbagai parameter dasar yang dipakai sebagai acuan pemilihan lokasi sistem Ingres saat instalasi. Parameter-parameter tersebut dapat dijadikan sebagai referensi ketika muncul pertanyaan-pertanyaan mengenai lokasi sistem pada saat instalasi.

Contoh langkah-langkah instalasi DBMS server Ingres adalah sebagai berikut :

1. Buat *account* untuk Ingres Sistem Administrator
   a. Masuk ke user **root**
   b. Buat account dengan nama user **ingres**, alokasikan user id lebih besar 100 untuk user **ingres.**
      Contoh :
      Useradd –u 101 –g kmem ingres
   c. Masukan
2. Buat direktori *system Ingres*
3. Buat direktori Ingres
4. Pindahkan *default directory* dan salin program instalasi
   Contoh
   cd $II_SYSTEM/ingress
   tar-xpf /dev/rmt/0m (asumsi source Ingres)
5. Set *shared-libraries* sistem operasi
   Contoh
   SHLIB_PATH=/lib:$II_SYSTEM/ingress/lib:$SHLIB_PATH
   export SHLIB_PATH
6. Jalankan utility Ingres untuk instalasi
   Contoh :
   Install/ingbuild
   Untuk melakukan instalasi semua objek dengan lisensi yang dimiliki TELKOM, dapat dipilih menu *default*.
7. Lakukan pemilihan setting format mata uang, tanggal dan adaptasi tahun (contoh : 2004) dari pilihan yang didefinisikan Ingres.
   Contoh :
   Ingsetenv II_DATE_FORMAT INDONESIA
   Ingsetenv II_MONEY_FORMAT "l:Rp."
   Ingsetenv II_DATE_CENTURY_BOUNDARY 20

8. Sebagai alternatif lokasi penyimpanan *transaction log*, Ingres memungkinkan dialokasikannya dua tipe *log file* :
    a. *System File* biasa yang disediakan Sistem Operasi.secara default dibuat saat instalasi.
    b. *Raw device* (partisi disk tanfa file sistem, yang memungkinkan pengurangan *overhead* proses disk I/O). Pemanfaatan *log file* tipe ini akan meningkatkan performansi sistem secara keseluruhan.
9. Saat instalasi selesai, start Ingres dengan menjalankan ingstart.
    Contoh : **Ingstart**
10. Lakukan *tuning* Ingres dengan menggunakan utilitas **cbf.**

3.9.8.3 Raw log

Pertimbangan yang dipakai untuk pemilihan lokasi raw log_file:

2.7 *Raw log* harus ada pada partisi yang terpisah

2.8 Harus tidak ada file sistem di dalamnya

2.9 *Raw device* ini tidak digunakan oleh program lain.

Langkah-langkah pembuatan dan pengaktifan *raw log* :

1. Masuk ke user root.
2. Buat *raw* device pada lokasi yang ditetapkan.
    Pada Sistem Operasi HP dapat dilakukan melalui utilitas SAM, buat *logical* volume tanpa *buffer* dan juga tanpa file sistem.
3. Pastikan setelah perintah ini dilakukan pada lokasi yang dipilih ada dua file dengan tipe *block device* dan *character device.*
4. Pindahkan direktori ke direktori sistem file Ingres
    Contoh : **cd$II_SYSTEM/ingress**
5. Jalankan perintah Ingres untuk membuat *log file* :
    Contoh : **Mkrawlog**
    Utiliti sistem Ingres ini :
    1. harus diikuti dengan lokasi lengkap *raw device* yang ditentukan
    2. Set **ingres** sebagai pemilik log file.
    3. Memeriksa lokasi dan ukuran *raw device* valid.

4. Membuat sub *directory* dengan nama **log** pada lokasi $II_LOG_FILE/ingress
5. Mendaftarkan **log** sub *directory* ini ke file sistem operasi.

3.9.9 Lingkungan Kerja SISKAMAYA

Bagian ini berisi informasi contoh penyiapan lingkungan operasional SISKAMAYA versi 2.3

3.9.9.1 Sistem Operasi

Implementasi SISKAMAYA 2.3 dilakukan pada lingkungan sistem operasi HP UX B10.20 U 9000/879. Beberapa perubahan standar instalasi yang harus disiapkan untuk menjalankan DBMS Ingres dan aplikasi SISKAMAYA 2.3 adalah :

a) Konfigurasi kernel dan registrasi pemakai aplikasi

Variabel kernel yang harus diubah untuk menjalankan aplikasi SISKAMAYA 2.3 (kebutuhan dasar untuk menjalankan DBMS Ingres) dapat dilihat pada bagian "Konfigurasi Sistem Operasi".

Backup sistem operasi dapat dilakukan sebagai tindakan antisipasi terhadap kemungkinan rusaknya sistem operasi. *Backup* ini akan sangat membantu dalam melakukan restore semua konfigurasi sistem operasi yang ada. Ada beberapa tipe *backup* yang dapat dipilih dan dilakukan :

1. *Backup* dengan memanfaatkan utilitas *backup* spesifik sistem operasi, dalam hal ini pemanfaatan backup dari **SAM** atau pemakai utilitas **fbackup** sistem operasi.
2. *Backup* dengan memanfaatkan utilitas *backup* yang bersifat general, yaitu pemanfaatan utilitas *backup* seperti **tar, cpio,** dan lain sebagainya.

3.9.9.2 DBMS Ingres

DBMS server yang dipakai pada SISKAMAYA 2.3 ini adalah CA OpenIngres 1.2. Langkah–langkah instalasi dan penyiapan lingkungan kerja DBMS Ingres dapat dilihat kembali pada bagian" Instalasi DBMS Server Ingres ".

Beberapa perubahan yang harus ditambahkan di luar instalasi dan rekonfigurasi parameter di atas adalah :

1. Pendefinisian group pemakai dan pendefinisian pemakai,
2. Seting tambahan yang berkaitan dengan konkurensi pengaksesan table.

Setting tambahan yang berkaitan dengan konkurensi pengaksesan table adalah tambahan lingkungan kerja DBMS Ingres, yaitu :

ING_SYSTEM_SET=include/oping/ingress/set.ini

Isi dari file set.ini adalah sebagai berikut :

set lockmode session where readlock =nolock,level =page,maxlocks=960;

set session with on_error = rollback transaction;

Pemasukan linkungan kerja ING_SET dapat dilakukan melalui user ingress pada UNIX prompt dengan menuliskan :

ingsetenv ING_SYSTEM_SET "include /oping/ingress/set.ini

Instruksi pertama, **set lockmode session where readlock =nolock,level =page,maxlocks=960;** mendefinisikan proses *locking* tidak dilakukan untuk proses pembacaan table, dan proses *locking* dalam manipulasi data setiap session dilakukan per *page*. *Locking page* maksimum sebesar 960 *page*, jika batasan ini dilampaui maka *locking* akan diperbesar menjadi tabel. Sedangkan Instruksi kedua dipakai untuk memastikan bahwa untuk setiap *session* yang dibuka apabila terjadi *error* sistem Ingres akan mkelakukan *rollback* transaksi.

Backup DBMS Ingres dapat dipakai untuk antisipasi terhadap kemunkinan rusaknya DBMS Ingres. Backup ini akan sangat membantu dalam melakukan restore semua konfigurasi DBMS Ingres. *Backup* DBMS Ingres yang pernah dilakukan dengan memanfaatkan utilitas **tar.** *Restore* dapat dilakukan oleh user **root** dengan menuliskan perintah pada UNIX *prompt* :

tar-cvf/devrmt/0m

Saat memasukan perintah ini pastikan bahwa *backup* tape ada pada tape drive. Hasil adanya *directory* :

/oping

3.9.9.3 Aplikasi

Aplikasi yang dipakai di basis data SISKAMAYA 2.3, tersimpan pada basis data **siska2000**. Sedangkan *source* aplikasi tersimpan pada *directory*

/home/siska2000/apli. Apabila diinginkan, *restore* aplikasi dari *backup tape* dapat dilakukan oleh user siska dengan menuliskan pada UNIX *prompt* :

Tar –xvf/dev/rmt/0m

Saat memasukan perintah ini, harus pastikan bahwa *backup tape* ada pada *tape drive*.

Jika proses ini telah selesai dilakukan, *restore* aplikasi kedalam basis data **siska2000** dapat dilakukan dengan cara :

3. Pindah ke directory :

   /home/siska2000/apli/frame

4. Jalankan instruksi untuk memasukan aplikasi :

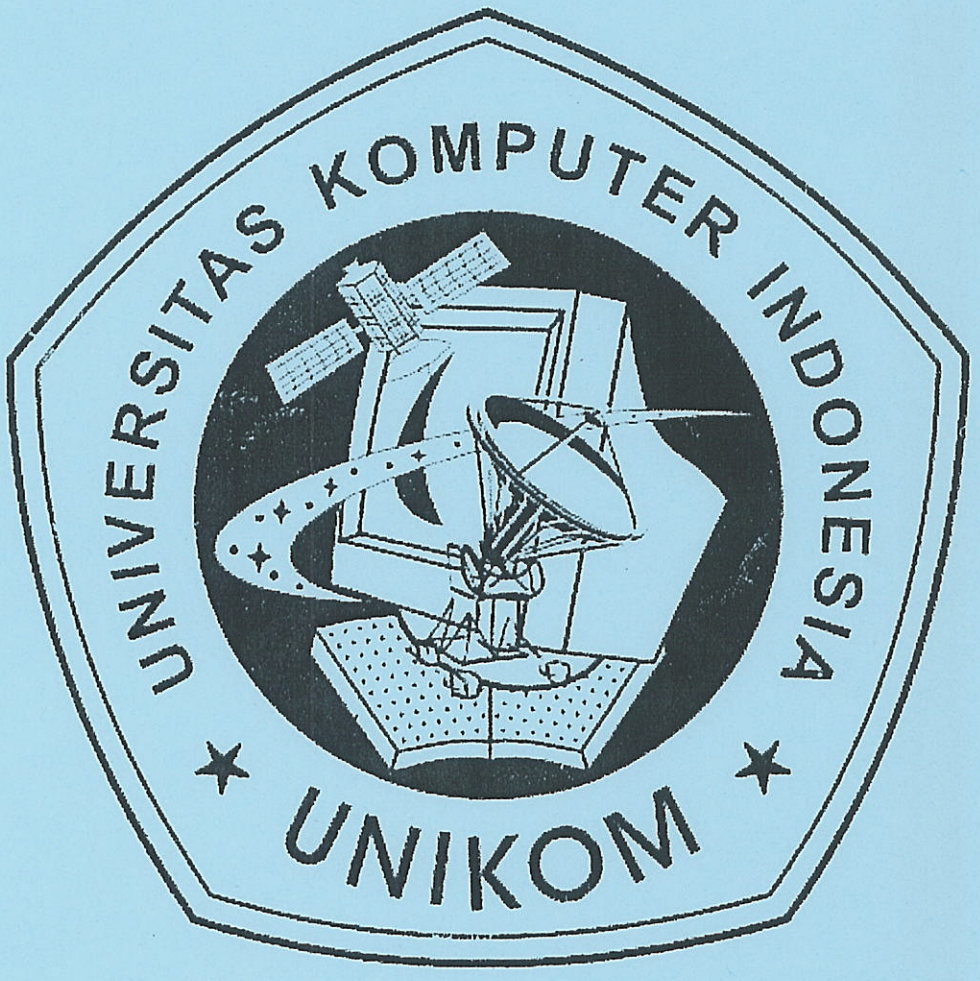
   /home/siska2000/apli/frame/backup.ini

Sebelum instruksi ini dijalankan pastikan bahwa basis data siska 200 sudah ada, jika belum buat dahulu basis data siska2000 dengan menuliskan pada UNIX *prompt* :

creatdb siska2000

Hasil dari langkah ini adalah terisinya aplikasi di basis data siska 2000. Aplikasi ini selanjutnya dapat diakses melalui utilitas **abf**, dengan menggunakan :

abf siska2000


UNIVERSITAS KOMPUTER INDONESIA
UNIKOM