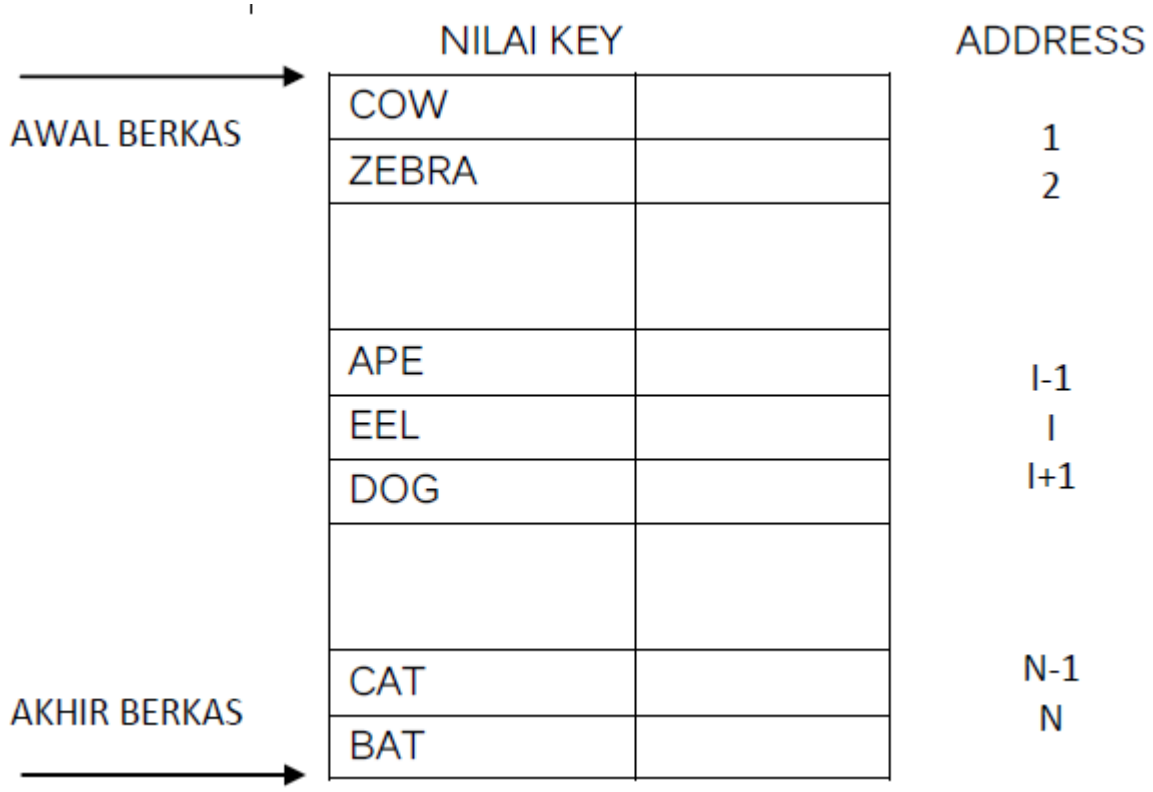
ORGANISASI BERKAS RELATIF

Salah satu cara yang efektif dalam mengorganisasikan sekumpulan record yang membutuhkan akses sebuah record dengan cepat adalah ORGANISASI BERKAS RELATIF. Dalam berkas relatif ada hubungan antara KEY YANG DIPAKAI untuk mengidentifikasi record dengan LOKASI RECORD dalam penyimpan sekunder. Perlu diperhatikan bahwa URUTAN RECORD SECARA LOGIC tak ada hubungannya dengan URUTAN SECARA FISIK. Record tidak perlu tersotir secara fisik menurut nilai key.



Bagaimana record yang ke-N dapat ditemukan? Dalam hal ini, perlu kita buat hubungan yang akan menerjemahkan antara NILAI KEY dan ADDRESS. Hubungan ini dinyatakan sebagai R, yang merupakan fungsi pemetaan.

dari nilai key ke address dalam penyimpanan sekunder. Pada waktu sebuah record ditulis ke dalam berkas relatif, fungsi pemetaan R digunakan untuk menerjemahkan NILAI KEY DARI RECORD menjadi ADDRESS, dimana record tersebut disimpan. Begitu pula pada waktu akan meretrive record dengan nilai key tertentu, fungsi pemetaan R digunakan terhadap nilai key tersebut, dimana record tersebut dapat ditemukan.

Berbeda dengan organisasi berkas sekuensial, organisasi berkas relatif ini tidak perlu mengakses record secara berurut (consecutive), karena sebuah record tertentu dapat diakses secara langsung. Organisasi berkas relatif ini tidak menguntungkan bila penyimpanan sekundernya adalah medium serial access seperti magnetic tape.

Berkas relatif harus disimpan dalam medium direct access storage device (DASD) seperti magnetic disk. Juga dimungkinkan untuk mengakses record-record dalam berkas relatif secara CONSECUTIVE. Tetapi perlu diketahui bahwa nilai key tidak urut secara logic. Sebagai contoh record dalam gambar di atas, di retrieve secara consecutive; COW, ZEBRA, …, APE, EEL, DOG, …, CAT, BAT. Karena kemampuan mengakses record tertentu secara cepat, maka organisasi berkas relatif paling sering digunakan dalam proses interactive. Sebagai contoh sebuah online sistem perbankan mempunyai berkas induk dengan struktur sebagai berikut :

CUSTOMER – ACCOUNT

ACCOUNT NUMBER	ACCOUNT TYPE	BALANCE	DATE-LAST CREDIT	DATE-LAST DEBIT

Sedangkan berkas transaksinya terdiri dari field-field sebagai berikut:

TRANSACTION

ACCOUNT NUMBER	TRANS TYPE	AMOUNT	DATE

Field ACCOUNT NUMBER dipakai sebagai nilai key untuk kedua berkas tersebut. Pada saat nilai key ACCOUNT NUMBER dimasukkan ke dalam transaksi, nilai key tersebut akan me-retrieve secara langsung record yang ada pada berkas induk. Jika TRANS-TYPE = 'T', maka BALANCE akan ditampilkan di layar. Jika TRANS-TYPE = 'C' atau 'D', maka record-record dari berkas induk CUSTOMER-ACCOUNT akan dimodifikasi dengan AMOUNT dan DATE yang ada di berkas transaksi, dimana ACCOUNT NUMBER yang menentukan lokasi record dalam berkas tersebut. Ada dua hal penting yang perlu dicatat; pertama, tidak perlu mengakses semua record berkas induk, cukup mengakses langsung record yang dikehendaki. Kedua, record dari berkas relatif dapat di up-date langsung tanpa perlu merekan kembali semua record. Bandingkan kedua hal ini pada organisasi sekuensial. Keuntungan dari berkas relatif ini adalah kemampuan mengakses record secara langsung. Sebuah record dapat di retrieve, insert, modifikasi, atau bahkan dapat di delete; tanpa mempengaruhi record laindalam berkas yang sama. Ada 3 teknik dasar yang digunakan untuk menyatakan fungsi pemetaan R, dimana

R (NILAI KEY) → ADDRESS

1. Pemetaan Langsung (Direct Mapping)
2. Pencarian Tabel (Directory Look-up)
3. Kalkulasi (Calculating)

A. TEKNIK PEMETAKAN LANGSUNG

Teknik ini merupakan yang sederhana untuk menerjemahkan nilai record key menjadi address. Ada dua cara dalam pemetaan langsung:

1) Pengalamatan mutlak (absolute addressing)
2) Pengalamatan relatif (relative addressing)

1. Pengalamatan Mutlak

Fungsi pemetaan R (NILAI KEY) ADDRESS, di mana NILAI KEY = ALAMAT MUTLAK. Fungsi pemetaan ini disebut PENGALAMATAN MUTLAK. Nilai key diberikan oleh pemakai program yang sama dengan ADDRESS SEBENARNYA dari record tersebut disimpan pada penyimpanan sekunder. Pada waktu record tersebut disimpan, lokasi penyimpanan record (nomor silinder, nomor surface, nomor record) bila dipakai CYLINDER ADDRESSING atau (nomor sector, nomor record) bila dipakai SECTOR ADDRESSING harus ditentukan oleh pemakai. Begitu pula pada waktu record tersebut di retrieve, lokasi mutlak itu harus diketahui dan diberikan pemakai. Ada 2 keuntungan dari pengalamatan mutlak:

- Fungsi pemetaan R sangat sederhana
- Tidak membutuhkan sekunder (retrieve lebih cepat)

Kelemahannya :

- Pemakai harus mengetahui dengan pasti record-record yang disimpan secara fisik.
- Pemakai tidak dapat menggunakan nilai key seperti ACCOUNT

- Alamat mutlak adalah device dependent. Perbaikan atau perubahan alat, dimana berkas berada, akan mengubah nilai key.
- Alamat mutlak adalah address space dependent. Reorganisasi berkas relatif akan menyebabkan nilai key berubah. Reorganisasi bertujuan memperbesar ruang alamat di mana sampah (garbage) akan dibuang atau dapat juga memperkecil ruang alamat.

2. Pengalamatan Relatif

Fungsi pemetaan R (NILAI KEY) ADDRESS, dimana NILAI KEY = ALAMAT RELATIF. Fungsi pemetaan ini disebut PENGALAMATAN RELATIF. Alamat relatif dari sebuah record dalam sebuah berkas adalah urutan record tersebut dalam berkas. Sebuah berkas dengan N record mempunyai record dengan alamat relatif dari himpunan (1, 2, 3, …, N-2, N, N-1) Record yang ke-I mempunyai alamat relatif I dan I-1.

Keuntungan dari pengalamatan relatif:

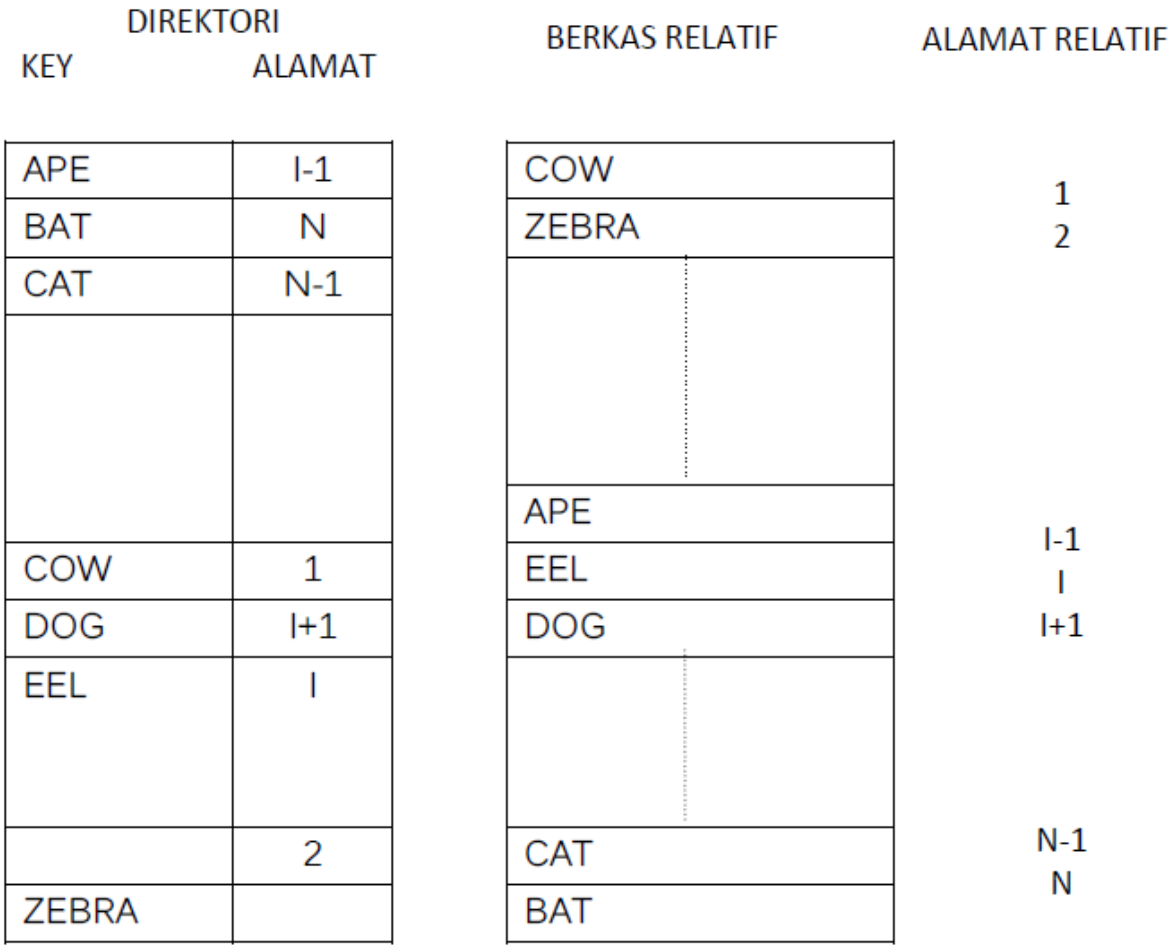
- Fungsi pemetaan R sangat sederhana
- Nilai key dari sebuah record dapat ditentukan lokasi recordnya dalam sebuah penyimpanan sekunder tanpa memerlukan waktu proses berarti.

Alamat relatif tidak tepat dikatakan sebagai device dependent seperti pada alamat multak, karena itu kelemahan tersebut dapat dihilangkan pada pengalamatan relatif. Namun seperti pada pengalamatan mutlak, pengalamatan relatif juga address space dependent. Nilai key dari sebuah field dapat dipakai sebagai alamat relatif. Misal dari 2000 jenis barang yang mempunyai nilai key PART NUMBER dipakai sebagai alamat relatif. Part

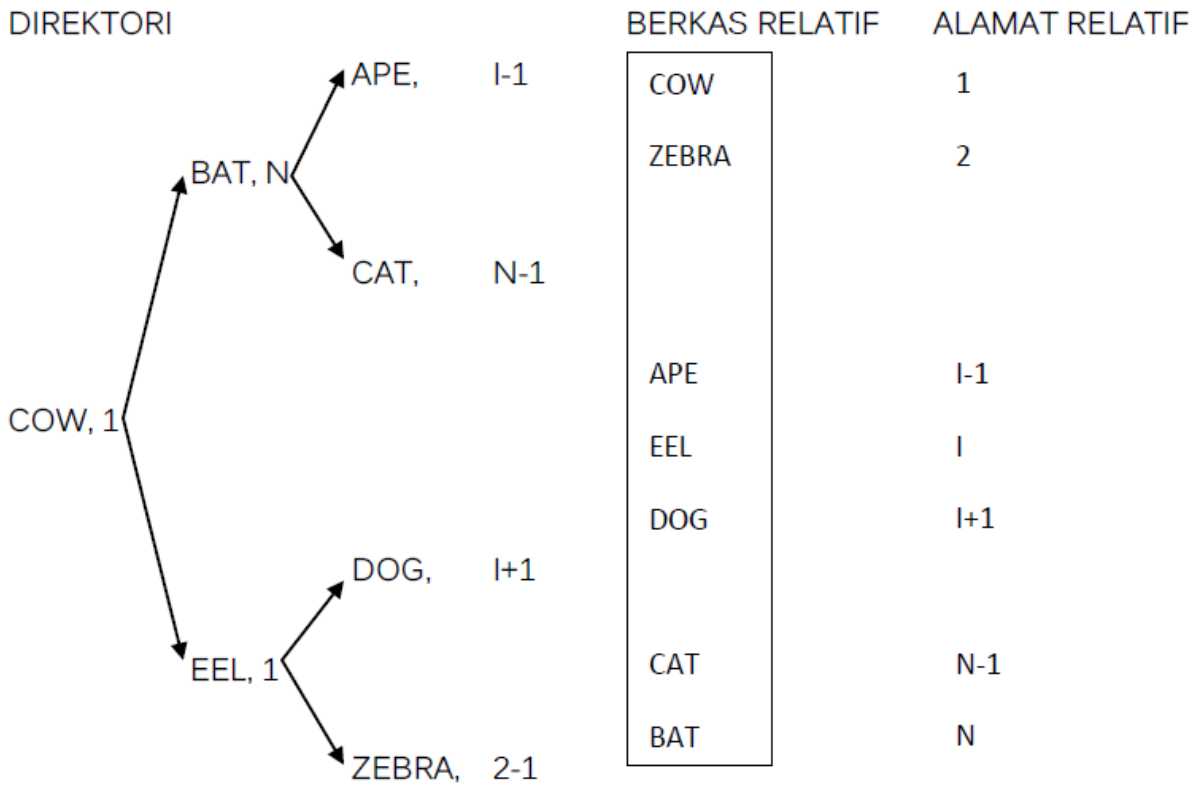
number 1001 mempunyai alamat relatif 1001. Kelemahan nilai key dari sebuah field dipakai sebagai alamat relatif adalah terjadinya pemborosan ruangan dimana berkas relatif menyediakan ruang (alamat relatif) untuk 9999 jenis barang dari sebenarnya 2000 jenis barang, terjadi pemborosan kurang lebih 80%. Untuk nilai key NIP yang terdiri dari 9 digit dari berkas pegawai yang berjumlah 2000 record akan terjadi pemborosan 99,99998% ruang dimana berkas relatif menyediakan alamat relatif untuk 999.999.999 pegawai. Untuk mengatasi pemborosan ruang adalah dengan cara mendapatkan sebuah key yang nilai jangkauannya mempunyai populasi tinggi. Sebagai contoh 8200 jenis barang akan lebih efisien mempunyai PART NUMBER sebanyak 4 digit sebagai nilai key-nya yang juga merupakan alamat relatif. Ruang kosong sebesar 20% dipakai untuk penambahan record.

B. TEKNIK PENCARIAN TABEL

Teknik pencarian tabel jauh lebih baik disbanding dengan teknik pemetaan langsung.Hanya saja memerlukan biaya baru dalam pemeliharaannya. Kita akan lihat bahwa pendekatan ini hamper serupa dengan teknik yang dipakai pada berkas indeks sekuensial. Dasar pemikiran pendekatan pencarian tabel adalah sebuah tabel atau direktori dari nilai key dan address. Untuk menemukan sebuah record dalam berkas relatif, pertama dicari dalam direktori nilai key dari record tersebut, akan menunjukkan alamat dimana record tersebut berada dalam penyimpanan. Dalam bentuk yang sedrhana, direktori diimplementasikan sebagai suatu array dari nilai key; record alamat, digambarkan sebagi berikut:



Disini data dalam direktori disusun secara urut menurut nilai key, sehingga pencarian nilai key dalam direktori lebih cepat dengan binary search dibanding sequential search. Alternatif lain, direktori dapat disusun dalam binary search tree, M-way search tree, atau B-tree.



Keuntungan dari pencarian tabel:

- Sebuah record dapat di akses dengan cepat, setelah nilai key dalam direktori ditentukan .
- Nilai key dapat berupa field yang mudah dimengerti seperti PART NUMBER, NMP, karena nilai key tersebut akan diterjemahkan menjadi alamat.
- Nilai key adalah address space independent, dimana reorganisasi berkas tak akan mempengaruhi nilai key, yang berubah adalah alamat dalam direktori.

Teknik ini banyak dipengaruhi oleh organisasi direktorinya. Apabila nilai key disimpan secara urut pada direktori, akses secara binary jauh lebih cepat dibanding secara sekuensial.

C. TEKNIK KALKULASI ALAMAT

Pendekatan lain yang umum dipakai untuk mengimplementasikan R (NILAI KEY) ADDRESS adalah dengan melakukan kalkulasi terhadap nilai key. Hasilnya adalah alamat alternatif. Teknik ini dapat dipakai sendiri atau bersama-sama dengan pencarian tabel. Ide dasar dari kalkulasi alamat adalah mengubah jangkauan nilai key yang mungkin, menjadi sejumlah kecil alamat alternatif. Salah satu kelemahan dari teknik pengalamatanrelatif adalah ruang harus disediakan sebanyak jangkauan nilai key, terlepas dari beberapa banyak nilai key. Sebagai contoh, dari 2000 jenis barang hendak dibuat alamat relatif sebanyak 2000 lebih sedikit. Cara ini dapat dilakukan dengan teknik kalkulasi alamat. Salah satu masalah dari teknik ini adalah ditemukannya alamat relatif yang sama untuk nilai key yang berbeda. Keadaan di alamat relatif yang sama untuk nilai key yang berbeda. Keadaan dimana R (K1) = R (K2) dan K1 # K2 disebut BENTURAN (COLLISION). Sedangkan nilai key K1 dan K2 disebut SYNONIM. Ada banyak cara untuk mengatasi benturan, antara lain:

- Scatter diagram techniques
- Randomizing techniques
- Key to address transformation methods
- Direct addressingbtechniques
- Hash tables methods
- Hashing