الباب الثانى عشر

١٢

إدارة الملفات والبيانات

الملفات عبارة عن فئة أو مجموعة من السجلات المرتبطة منطقيا يتم تخزينها على وسائط التخزين الثانوية ، ويسمح نظام التشغيل للمستخدم بالتعامل معها عن طريق الأتصال ببرامج إدارة الملفات ، أحد أهم مكونات نظام التشغيل ، مستخدما فى هذا الأتصال بعض الأوامر الواردة بعد ، ويتعاون برنامج إدارة الملفات مع برنامج قيادة المعدات فى تحقيق الوظيفة التى يهدف إليها الأمر المستخدم مثل:

- أفتح الملف OPEN FILE لإعداد الملف للتعامل معه.
- أغلق الملف CLOSE FILE لإنهاء التعامل مع الملف.
- أنشئ ملف CREATE FILE لإنشاء ملف جديد.
- دمر ملفا DESTROY لمسح ملف منطقى من على وسائط التخزين.
- إعادة تسمية ملف RENAME لأعطاء مسمى جديداً.
- أطبع / أعرض ملف LIST لأستعراض محتويات ملف.
- أقرأ READ أظهر / أنقل سجل من الملف المنطقى.
- أكتب WRITE أكتب ناتج العملية التشغيلية على الملف.
- حدث UPDATE لتعديل بيانات مسجلة.

obeikandl.com

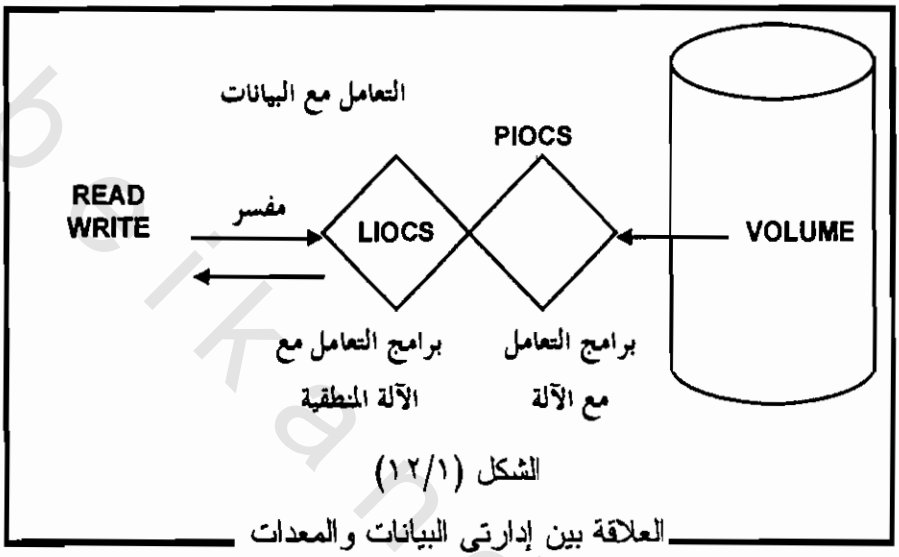
* أدخل INSERT لإضافة سجل جديد.
* أحذف DELETE لإلغاء سجل أو عدة سجلات.
* إلخ ETC.

وهذه الأوامر وغيرها كثير يتولى برنامج خاص أستقبالها وتفسيرها وتجزئتها إلى عدد من التعليمات السابق أعدادها بلغة الآلة وهى بدورها توعز إلى الآلات بأداء خطوات وأعمال محدده سلفا تحقيقا للغرض الذى يهدف إليه مستخدم الأمر.

ويمكن توضيح مثل هذه الإجراءات من إلقاء نظرة على بعض الخطوات التى يؤديها النظام / الحاسب / فور تلقى الأمر READ أقرأ من على قرص مغناطيسى ، حيث يتم الآتي:

أ ــ أخطار وحدة التشغيل المركزية بمقاطعة من نوع I/O.
ب ــ بعث SIO مناسبة محدد بها القناة المطلوبة والآلة المستهدفة.
ج ــ تحريك ذراع القراءة إلى المسار المطلوب على القرص.
د ــ وضع السجل المطلوب تحت رأس القراءة.
هـ ــ دفع رأس القراءة إلى السجل.
و ــ قراءة بيانات السجل.
ز ــ نقل البيانات إلى المنطقة العازلة المخصصة لذلك فى الذاكرة الأساسية.
ح ــ أخطار البرنامج المشرف بإنتهاء القراءة.
ط ــ مقاطعة جديدة...
ك ــ إلى آخره...

ومثل ما أسلفنا من خطوات وأن لم تشمل كل الخطوات ، فلا شك توضح لنا مدى أهمية التعاون بين برامج إدارة الملفات ، وإدارة المعدات والتى يمكن توضيحها فى الشكل (١٢/١).



الشكل (١٢/١)

العلاقة بين إدارتى البيانات والمعدات

مهام مدير الملفات:

يمكن توضيح مهام مدير الملفات من إجراء مناظرة بين مهامه ، والأعمال التى يؤديها أمين مكتبة يستخدم مجموعة كروت سطرت عليها البيانات اللازمة مثل [أسم الكتاب ــ أسم المؤلف ــ رقم الإيداع ــ تاريخ الإيداع... إلخ] محدداً بذلك كل الكتب وأوعية المعرفة المحفوظة بالمكتبة ، ونفترض أن أمين المكتبة يتبع فى تخزين هذه الأوعية المكتبية أسلوبا وسياسة محددة مسبقا بحيث تسمح للمترددين على المكتبة التعرف على مختلف الأوعية المكتبية والوصول إلى ما يحتاجونه للإطلاع.

ونفرض أن متردداً طلب الإطلاع على كتاب بذاته ، فإن أمين المكتبة يستخدم الكروت [الفهرس أو الكشاف] فى تحديد الكتاب المطلوب وموقعه ويأتى به ، بعدها يسجل أسم المستعير فى دفتر الاستعارة ، وفور إنتهاء المستعير من الإطلاع يعيد الكتاب إلى أمين المكتبة والذى بدوره يشطب بيانات المستعير ويعيد الكتاب إلى موقعه.

من هذه المقارنة نلمح أن مدير الملفات فى نظام التشغيل يتولى تنفيذ الواجبات التالية:

أ ــ تسجيل البيانات التوصيفية لجميع ملفات المنظومة فى جدول موحد يسمى جدول النظام أو ملف المنظومة والذى يعرف فى المراجع بأحد الأسماء التالية:

* FILE DIRECTORY.
* PERMENANT FILE DIRECTORY.
* SYSTEM CATALOG.
* FILE NAME TABLE.
* MASTER FILE DIRECTORY (MFD).

ب ــ تحديد مواقع ملفات المنظومة على وسائط التخزين الثانوية.

ج ــ تخزين الملفات على وسائط التخزين الثانوية محققاً الأستغلال الأمثل للموارد المتاحة مع تحقيق الوصول إليها بسهولة ويسر.

د ــ تخصيص الملف للمستخدم أو جملة مستخدمى الملف متى طلب منه ذلك بالأمر READ وإستعادة الملف متى أنتهت الحاجة إليه إستدلالاً بالأمر CLOSE مع تسجيل بيانات كاملة عن الملف والمستخدم وتوقيت الاستخدام على ملف آخر يدعى LOG FILE.

ملف النظام:

من المعروف أن الملفات النشطة فى أى منظومة يتم تحميلها بحيث تكون متاحة ON-LINE لمختلف المستخدمين فى حين يتم تخزين الملفات غير النشطة وفق

مفهومOF-LINE، وكلاهما يتم تخزين بياناتهم على ملف النظام ، وتشمل هذه البيانات بنوداً عديدة لعل أبرزها مايلى:

أ ــ أسم الملف.

ب ــ موقع الملف على المجلد/ VOLUME / الوحدة التخزينية.

ج ــ تنظيم الملف (متتالى ــ مفهرس ــ مباشر).

د ــ تاريخ إنشاء الملف.

هـ ــ تاريخ إلغاء الملف.

و ــ نوعية الوحدة الآلية المستخدمة فى تشغيل الملف.

ز ــ نشاط الملف.

ح ــ تاريخ آخر تعديل.

ط ــ أسلوب الإسترجاع.

ى ــ المدى الزمنى (مؤقت أو مستديم).

ك ــ التكتل ومداه وحيز السجل وحيز الكتلة المنطقية.

ل ــ أسم صاحب الملف.

م ــ .....

المجلد أو الوحدة التخزينية VOLUME :

سبق وأوضحنا أن هذا المسمى يطلق على وحدة التخزين الواحدة مثل القرص المغناطيسى ، وقد يحتوى المجلد على أكثر من ملف ويسمى فى هذه الحالة MULTI FILE VOLUME ، وفى حالة تضخم الملف الواحد عن حدود مجلد واحد ويسمى فى هذه الحالة [ملف متعدد الوحدات التخزينية MULTI VOLUME FILE] ، ويتم تعريف كل وحدة تخزينية فى النظام بأسم منطقى وأسم فيزيائى ، ولكل مجلد بطاقة تعريف LABEL تضم إلى جانب الأسم عدة بيانات منها:

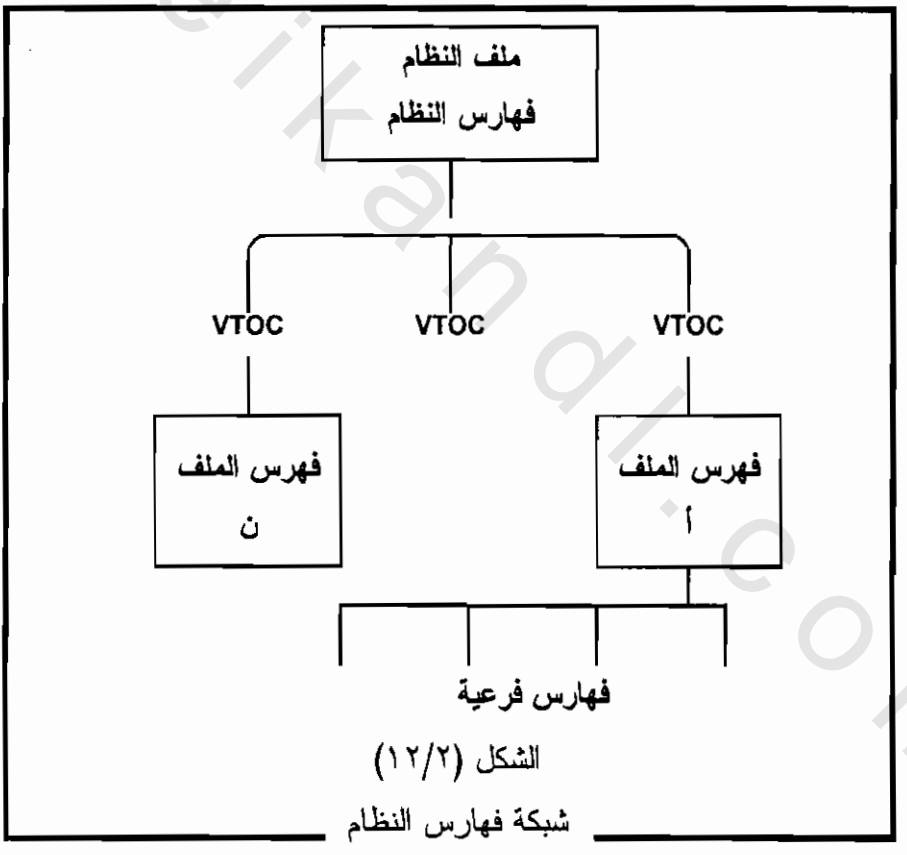
أ ــ تاريخ إنشاء المجلد.

obeikandl.com

ب ــ موقع فهرس المجلد.

ج ــ كتلة بداية الملف (رقم القطاع SECTOR فى حالة DASD) .

وعموما يمكن القول أن لكل مجلد VOLUME فهرسا خاصا به يسمى VOLUME TABLE OF CONTENT V.T.O.C ويتم تسجيله على المسارات الأولى فى القطاع الأول من وحدات DASD أو عند بداية الشريط المغناطيسى ، وذلك تسهيلاً لعمل نظام التشغيل فى الوصول إلى الملف عبر البحث المتتالى فى شبكة فهارس النظام الموضحة بالشكل (١٢/٢).



الشكل (١٢/٢)

شبكة فهارس النظام

خصائص الملفات:

توصف ملفات النظام وفق ثلاثة معايير هى:

أ ــ التلاشى أو عدم الأستقرار VOLATILITY :

وتعتبر هذه الخاصية مؤشراً إلى مدى الحذف والإضافة على الملف فإن زادت المعدلات لدرجة تجعل البيانات تتبدل وتتغير بأستمرار ، أطلق على الملف صفة التبخر ، أما إذا كانت عمليات الحذف والإضافة محدودة فإنه يطلق على الملف صفة الثبات أو الإستاتيكية.

ب ــ النشاط ACTIVITY :

وتعتبر هذه الخاصية مؤشراً عن كم السجلات التى يجرى تشغيلها أو إسترجاعها إرجاعا إلى عدد السجلات على الملف الواحد.

ج ــ حجم الملف SIZE :

وهو عبارة عن حجم الملف بوحدة الميجابايت (MB) .

تسمية الملفات:

يتوقف طول أسم الملف على قدرة نظام التشغيل ومدى كفاءته ــ ففى نظم التشغيل البسيطة لم يكن يسمح بأكثر من ثلاث حروف بينما النظم الحديثة تسمح بأستخدام عدد أكبر من الحروف ، وقد يتكون الأسم من مقطعين أو ثلاثة أو أكثر.

ويعتبر المقطع الأول هو المكون المسموح به للمستخدم وعليه إختياره بحيث يكون معبراً عن طبيعة البيانات ، ويتكون المقطع الثانى من حرفين أو ثلاثة تفصلها عن المقطع الأول نقطة (.)ويستخدم هذا المقطع فى توصيف نوعية الملف،ففى نظام مثل MS-DOS يستخدم المقطع الثانى[الأمتداد ــ EXTENSION] على النحو التالى:

FILE NAME . EXTENSION

برنامج فورتران	. FOR
برنامج كوبول	. COB

. BAT برنامج من نوع الدفعة

. DAT ملف بيانات

أسلوب أستدعاء الملفات:

بفرض أن هناك ملفا يدعى INVENTRY . FOR فإن أستدعائه يعتمد على نوعية نظام التشغيل فيما يوضحه المثال التالى:

أ ــ مع نظام تشغيل MS -DOS على حاسب شخصى مزود بأكثر من وحدة أقراص يكون الأمر على النحو:

CD \ PARTS \ INVENTRY. FOR

* أى أن الملف على القرص (C) تحت فهرس فرعى يسمى PARTS وأن الملف يحتاج إلى مترجم فورتران.

ب ــ مع نظام تشغيل UNIX يتم أستدعاء نفس الملف بالأمر:

/ USER / KLMN / XYZL / INVENTRY . FOR

* / تعنى مدخلا إلى الفهرس الرئيسى أو ملف النظام.
* USR / KLMN تعنى مدخلاً إلى فهرس فرعى.
* XYZL / تعنى مدخلاً إلى فهرس فرعى آخر.

السجلات منطقياً وفيزيائياً:

يوصف السجل المنطقى بتحديد وتسمية حقوله وتحديد طول كل حقل ونوعه.

وتندرج السجلات من حيث الطول ضمن أحد أنواع ثلاثة:

أ ــ سجل محدد الطول ويرمز له بالرمز (F).....FIXED .

ب ــ سجل متغير الطول ويرمز له بالرمز (V).....VARIBALE .

ج ــ سجل غير معلوم الطول ويرمز له بالرمز (U).... UNDEFIND .

وهذه السجلات قد تكون مكتلة أو غير مكتلة فيما توضحها الأشكال (١٢/٣) ، وجدير بالقول أن تفكيك السجلات أو تكتيلها مسئولية نظام التشغيل من خلال

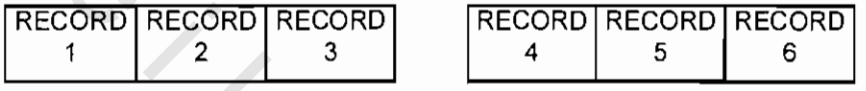
روتينات LIOCS ، PIOCS علما بأنه توجد نظم تشغيل لاتساند السجلات من نـــوع U.

LOGICAL RECORD FORMATS

FORMAT F RECORDS
FIXED - LENGTH RECORD
UNBLOCKED

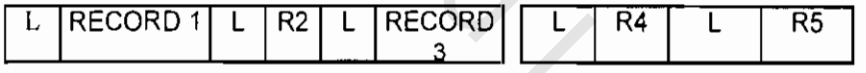
UNBLOCKED

BLOCKED



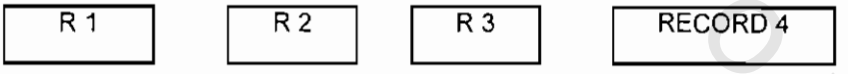
FORMAT V RECORDS
VARIBLE - LENGTH RECORD WITH STANDARD LENGTH - FIELD
UNBLOCKED

BLOCKED



L = STANDARD LENGTH - FIELD

FORMAT V RECORDS
VARIBLE - LENGTH RECORD WITHOUT STANDARD LENGTH - FIELD



FORMAT U RECORDS MAY BE INTERNALLY BLOCKED BUT USER MUST PROGRAM BLOCKING AND DEBLOCKING.
THIS IS NOT SUPPORTED FOR FORMAT U RECORDS BY DATA MANAGEMENT.

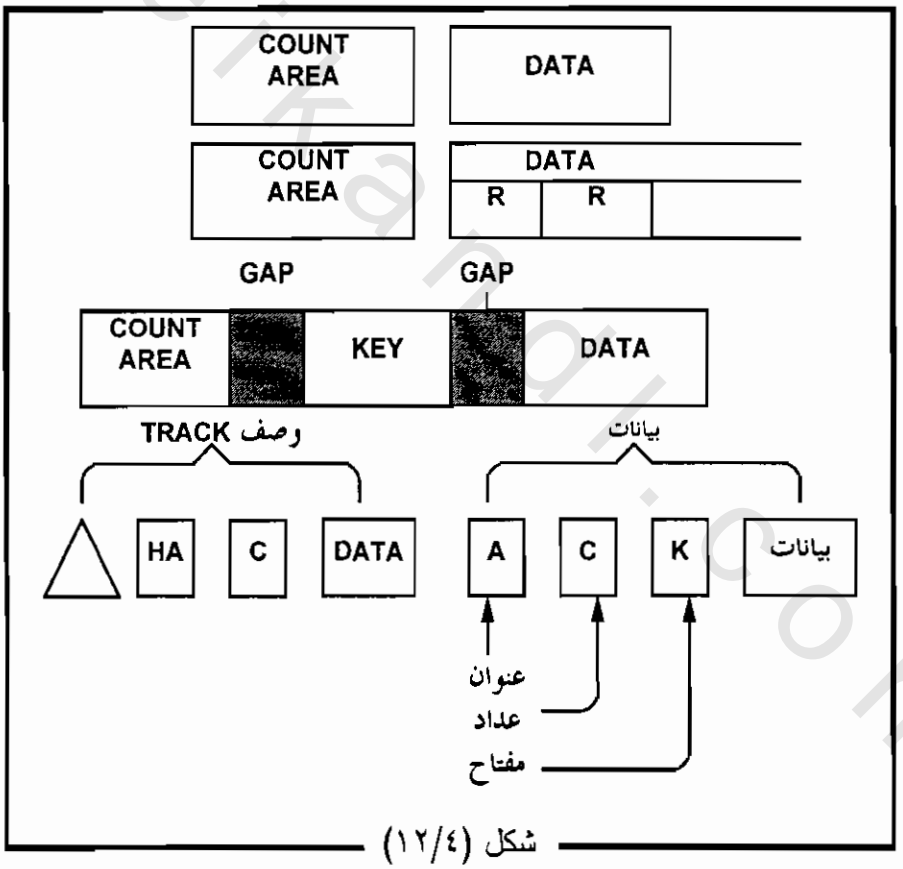
الشكل (١٢/٣)

بالسجلات المفككة وآخر للسجلات المكتلة لتطلب ذلك استهلاك حيز كبير من الذاكرة الأساسية لذلك تصمم IOCS على أساس التكتيل ويعامل السجلات غير المكتلة على أنها مكتلة بأعتبار عامل الكتلة يساوى الوحدة.

السجلات على وحدات DASD :

تتخذ السجلات شكلين أساسيين عند تسجيلها فى مسارات التسجيل : تعداد / بيانات أو تعداد / قيادة / بيانات ، كما فى الشكل (١٢/٤).

ومنطقة المفتاح هى عبارة عن نسخة من MAIN KEY فى السجل الذى تجرى عليه عملية الفهرسة مثل رقم العامل ــ رقم الصنف.



شكل (١٢/٤)

أ ــ فى النظام المكتل يشمل المفتاح أعلى فهرس فى الكتلة.

ب ــ فى النظام غير المكتل يشمل مفتاح فهرس السجل التالى.

عندما يقرأ أى بيان فأن جميع حقول السجل تقرأ كلها ويتولى برنامج إدارة البيانات إخفاء الجزء الغير متعلق بالمستخدم [عنوان / المفتاح / طول حقل البيان].

ويجرى كتابة السجلات على الأقراص المغناطيسية تحت أشـــراف مديـــر البيانات ويضاف على السجل المنطقى عدة موصفات لخصها الشكل (١٢/٤).

وهى أجزاء غير متعلقة بمستخدم النظام ويخفيها برنامج قيادة البيانات عـــن المستخدم وتصبح شفافة TRANSPARENT حياله.

وتدرج الموصفات المنطقية والفيزيائية ضمن جداول قيادة البيانات وبالتالى يمكن تغييرها حسب متطلبات برامج التطبيقات ، ويشمل ما سبق أن أسلفنا إليه حول الملفات إضافة إلى هيئة وفورمة FORMAT السجلات من وجهة النظـــر المنطقيـــة والفيزيائية ، وجدير بالذكر أن هناك بعض برامج قيادة البيانات والملفـــات تعـــامل السجلات المفككة على أنها سجلات مكتلة بمعامل تكتيل يساوى واحد.

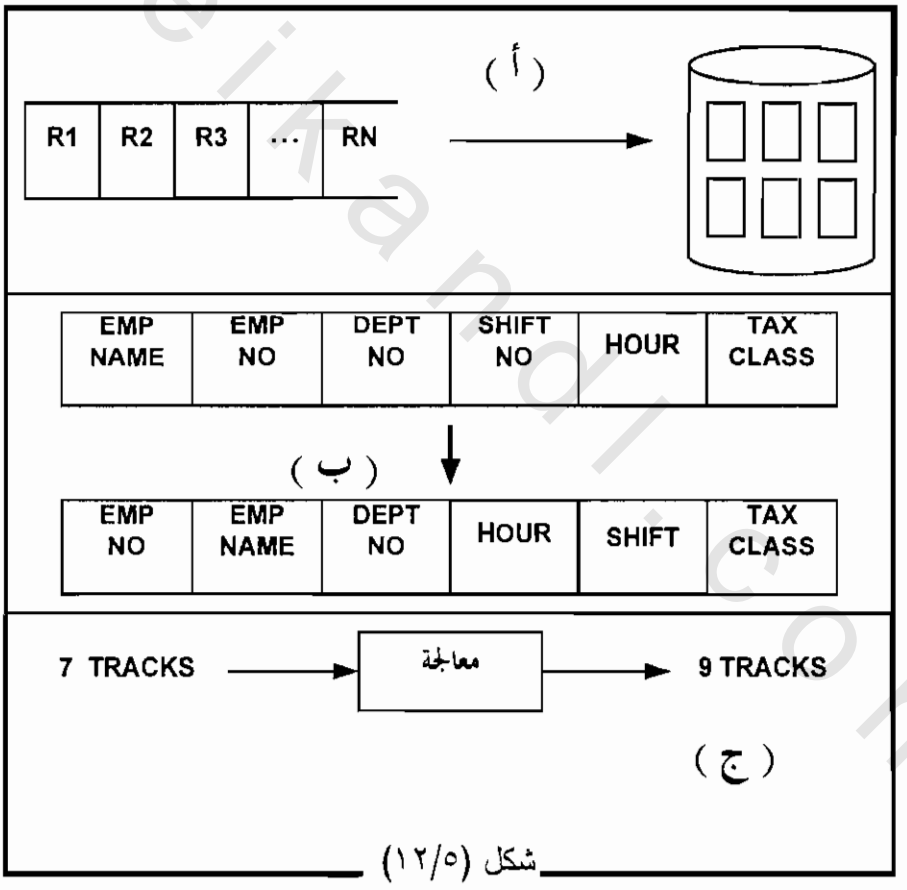
وإلى جانب الوظائف المعتادة لبرامج قيادة البيانات والملفات مشاركـــة مـــع برامج إدارة المعدات ممثلة فى روتينات LIOCS و PIOCS من حيث قراءة وكتابـــة الملفات وتكتيل وتفكك السجلات واستشعار نهاية الملف أو نهاية المجلـــد VOLUME فإنها تتأكد من صحة نقل السجلات ، ومراقبة وقياس طول السجل أسترشاداً بالطول المحدد فى بدايته مستخدمة فى ذلك عداد داخلى .. إلى آخر هذه الوظائف من فتـــح وغلق الملفات وفرض نقاط المراجعة فإنه يمكنها القيام بعدة وظائف أخرى هى:

أ ــ نقل السجلات من وسيط تخزين إلى وسيط آخر.

ب ــ تعديل هيئة السجل RECORD FORMAT.

ج ــ تعديل مسارات التسجيل وفق الأسلوب المطلوب.

وهذه الوظائف الهامة نراها موضحة فى الشكل (١٢/٥) ، حيث يوضح الشكل (أ) نسخ السجلات على شريط مغناطيسى إلى قرص مغناطيسى ، ويوضح الشكل (ب) تعديل فورمة السجل بأجراء تعديل على مواقع وأسبقيات الحقول المنطقية له ، ويوضح الشكل (ج) تعديل الملف من التسجيل على سبعة مسارات إلى التسجيل على تسعة مسارات.



شكل (١٢/٥)

دراسة الملفات على الوسائط التخزينية:

أولا: الملفات على الشرائط المغناطيسية:

تكتب السجلات من الذاكرة على الشرائط بالأمر "أكتب" وكل سجل يتم كتابته يتلوه سجل آخر وهكذا طالما أن هناك سجلات ، ونلحظ من ميكانيكية الكتابة على الشريط أن دوران الشريط يأخذ أقصى سرعة منتظمة من بداية الكتابة حتى نهايتها ثم يبطىء الدوران لدرجة التوقف بعدها يعاود أكتساب السرعة اللازمة مما يستغرق وقتا يسمح بمرور جزء من الشريط دون الكتابة عليه مسببا فجوة بين السجلات INTER RECORD GAB فإذا جرى تكتيل السجلات نجمت الفجوة بين الكتل المنطقية مما يحقق وفرا فى حيز التخزين.

والتكتيل يرفع من كفاءة نقل البيانات ويساعد على تحسين أداء وحدة التشغيل المركزية علاوة على الأستغلال الجيد للشرائط.

معامل التكتيل:

هو عدد السجلات المنطقية فى السجل الفيزيائى (الكتلة) ويتراوح بين سجل واحدا ، وعدد "ن" من السجلات فإذا كان معامل التكتيل يساوى واحد فإنه لا تكتيل ، وعموما فإن الأمر "أكتب" فى مرحلة التكتيل يكتب كتلة كاملة وليس سجلا مفردا.

تنظيم الملفات على الشرائط المغناطيسية:

١ ــ تنظيم الملفات وفق المعيار الزمنى لورود السجلات فمن يأتى أولا يسجل أولا ويسمى هذا التنظيم PILE كومه أو SERIAL متواليا.

٢ ــ إذا أعيد ترتيب سجلات الملف المسلسل وفق قيمة حقل المفتاح سمى الملف فى هذه الحالة ملف مسلسل SEQUENTIAL .

وكما نظمت الملفات مسلسلة أو متوالية فإن إسترجاع السجلات يكون بذات ترتيب وتنظيم الملف وهو أسلوب فرضته تقنية الشريط المغناطيسى مما حد من أستخدامها فى الأنظمة النشطة أو الملفات المتطايرة ، حتى عند تحديث بيانات سجل يعاد كتابته فى موقعة الأول مما يستدعى عند تحديث الملفات أن يكون شريط ملف المتغيرات له نفس تنظيم الملف الرئيسى ، وفى حالة حذف أى سجل يبقى مكانه خاليا على الشريط حتى تتم كتابة الشريط (الملف) مرة اخرى.

تنظيم الملفات على الأقراص المغناطيسية:

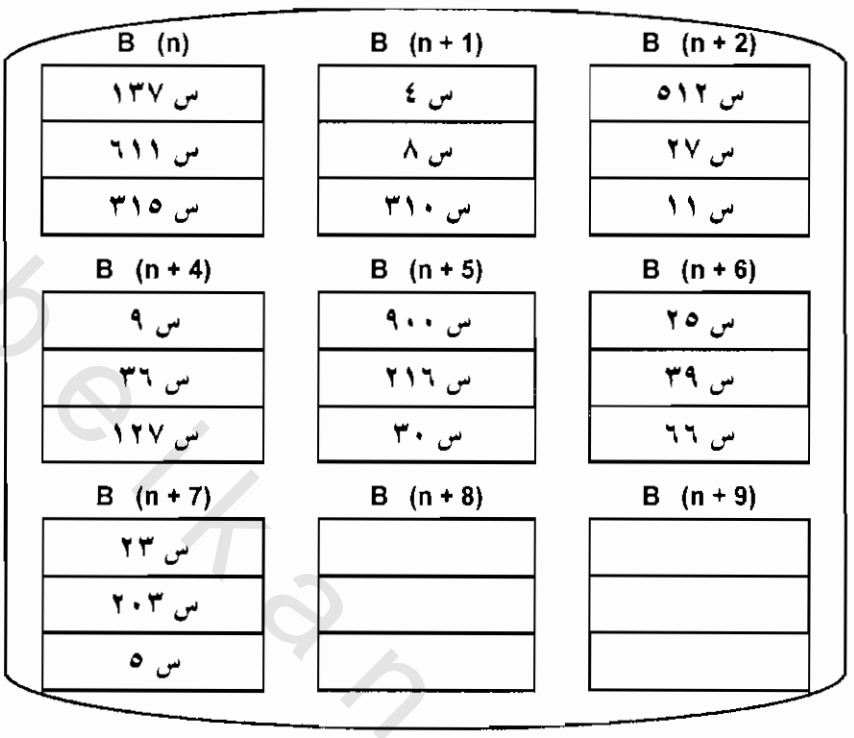
ناقشنا فى الأبواب الأولى من الكتاب طريقة التسجيل على مجموعة الأقراص الصلبة وكيفية تخليق الأسطوانات ومعنى المسارات حيث يقسم كل مسار إلى عدد من القطاعات ويسمى القطاع الواحد كتلة البيانات لأن بيانات كل قطاع تنقل أو تكتب دفعة واحدة فيما بين القرص والذاكرة فيما يناظر كتلة منطقية على الشريط المغنطيسى ، وقد يتم جمع الكتل المنطقية إلى بعضها البعض مكونة BUCKETS عناقيد فيما يوضحه الشكل (١٢/٦) مما يسرع عمليات القراءة الكتابة على الأقراص وتنظم الملفات على الأقراص بأى من أساليب التنظيم التالى:

أ ــ تنظيم مسلسل كما فى الشرائط.

ب ــ تنظيم متتالى كما فى الشرائط.

وكلا الأسلوبين يعتبر إهدارا لقدرات وسرعة التعامل مع الأقراص سيان فى تسجيل البيانات أو إسترجاعها.

ج ــ تنظيم مسلسل مفهرس ويتطلب إنشاء ملف فهرس إلى جانب ملف البيانات ، ولا يحقق هذا التنظيم سوى سرعة إسترجاع البيانات ويتطلب حيز تخزين أكبر من حيز ملف البيانات.



ملف متوالىSERIAL تم إنشاؤه إعمالا للقاعدة

"السجل الذى يأتى أولا يخزن أولا" مع ملاحظة أن

كل كتلة BLOCK تضم ثلاثة SLOTS

شكل (١٢/٦) ترتيب السجلات على الأقراص

د ــ تنظيم عشوائى (مباشر).

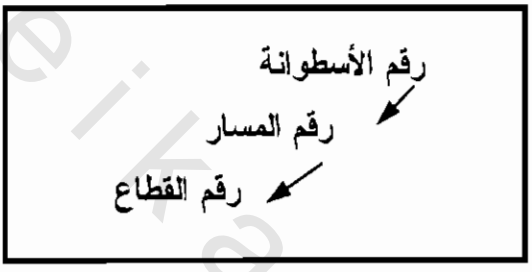
وإن كنا سنتناول هذه التنظيمات تفصيلا فى صفحات لاحقة فلابد من التأكيد على أن تقنية الأقراص تتيح الإسترجاع الفورى لأى سجل علـــى أى موقـــع مـــن الأسطوانات الوهمية للبيانات مما يتطلب عنونة السجلات ADDRESSING وتقســـيم مواقع التخزين إلى أجزاء يسهل الوصول إليها وفق هذا الترتيب الهرمى:

١ ــ الأسطوانة: تنشأ الأسطوانة الوهمية من كل المسارات ذات الرقم الواحد على جميع أسطح الأقراص الممغنطة وعدد المسارات فى الأسطوانة الواحدة يساوى عدد أوجه الأقراص المغناطيسية ناقص ٢.

٢ ــ المسار : هو المكان الفعلى الذى يتم عليه التسجيل.

٣ ــ القطاع : وهو أقل كتلة منطقية يمكن عنونتها وتقرأ أو تكتب دفعة واحدة.

وبذلك يكون عنوان السجل على النحو من اليسار إلى اليمين كما فى الإطار التالى:



١ ــ التنظيم المباشر DIRECT ORGANIZATION:

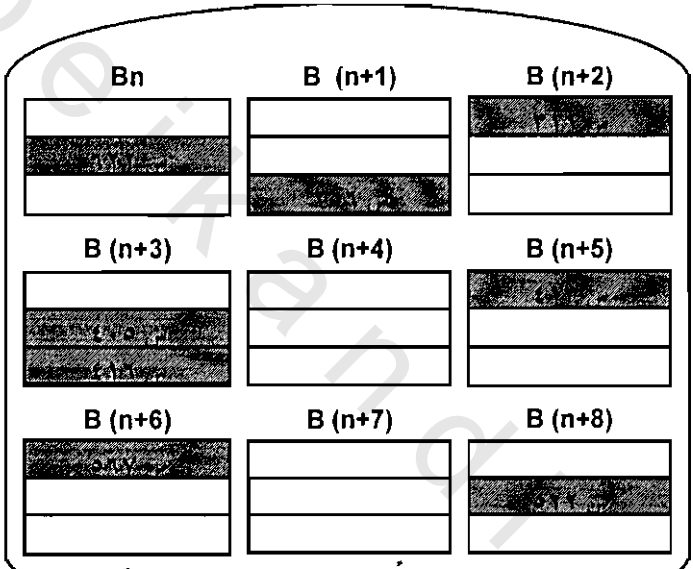
يعتمد هذا التنظيم على أساس تخزين السجلات واسترجاعها وفق عنوان لكل سجل يحسب من معادلة رياضية تطبق على حقل المفتاح الرئيسى فإن كان الحقل أبجدى فيجب تحويله إلى قيمة رقمية بعده يتم حساب قيمة العناوين بأحد أسلوبين:

أ ــ عنونة المفتاح الرئيسى KEY ADDRESSING :

يجرى حساب قيمة الموقع النسبى للسجل بتطبيق معادلة على قيمة حقل المفتاح ولتكن المعادلة على النحو الرقم النسبى = [(قيمة المفتاح – ٣) ÷ ٢] فإذا كانت قيمة حقل المفتاح ١٩ فإن رقمه النسبى يساوى "٨"وهذا الاسلوب يحقق سرعة استرجاع السجلات لان العنونه اتاحت تخريط كل الحيز المتاح للتخزين ووزعته على السجلات لكن السجلات لا تاتى تلو بعضها بانتظام مما يستدعى ترك حيز لسجلات لم ترد بعد وبالتالى فإن كفاءة الإسترجاع تكون على حساب فجوات واسعة

على وسيط التخزين فيما يوضحه الشكل (١٢/٧).

يوضح الشكل ملفاً مكون من تسع كتل منطقية وتم توزيع السجلات داخلها على إعتبار أن الكتلة تضم ثلاث سجلات داخل ثلاث SLOTS من كل كتلة BLOCK. الكتلة = حيز القطاع SECTOR كما تساوى صفحة من نظم التصفح ، وقد وزعت السجلات على أساس أن موقع السجل مساو لرقم العميل دون تطبيق أى منهج فى معالجة قيمة رقم العميل مما سبب فجوات واسعة على الملف.



يوضح الشكل ملفاً مكون من تسع كتل منطقية
وتم توزيع السجلات داخلها على إعتبار أن الكتلة
تضم ثلاث سجلات داخل ثلاث SLOTS فى كل كتلة BLOCK .
الكتلة = حيز القطاع SECTOR كما تساوى صفحة فى نظم التصفح.
وقد وزعت السجلات على أساس أن موقع السجل مساو لرقم العميل
دون تطبيق أى منهج فى معالجة قيمة رقم العميل مما سبب فجوات
واسعة على الملف.

شكل (١٢/٧)

ونؤكد على أنه يتحتم وجود حقل مفتاح فريد فى هيكل السجل المنطقى إذ بدونه يستحيل أستخدام هذا الأسلوب.

ب ـ التظليل Hashing :

ويشبه أسلوب عنونة المفتاح الرئيسى ويطبق فى حسابه معادلة رياضية على قيمة حقل المفتاح يستخدم فى عنونة السجل ، ولا تختلف هذه الطريقة عن طريقة العنونة سوى أن التظليل لا يقدم عنوان فريد ، وحيد ، لتخزين السجل فقد يتواجد أكثر من سجل لها نفس قيمة التظليل مما يسبب مشاكل تخزينية معقدة تتطلب حلولا جذرية ، هنا يجدر طرح السؤال ولماذا نستخدم التظليل إذا كان الأمر على النحو الذى أوضحنا من مشاكل؟ والإجابة [أن هذا الأسلوب يحقق أستخداما وإستغلالا أمثل لوسائط التخزين عكس الأسلوب الأول] مما يعطى ميزة لأنظمة المعلومات المباشرة On-Line .

منطق التظليل:

تستخدم معادلة أو خطة لحساب قيمة التظليل وتسمى فى المراجع الأجنبية بعدة مسميات هى:

Randomizing (Technique / Routine).
Hashing Algorithm
Hashing Routine
Hash Function

والآن نقترح أستخدام التظليل وفق خطة جمع المئات فى رقم الكتاب أو رقم العميل إلى الآحاد والعشرات ، ونطبق التظليل على تسع سجلات على النحو التالى ، وهى ذات السجلات الواردة بالشكل (١٢/٨):

رقم العميل	٤٠٥	٣١١	٤١٢	١٢٤	٥٢٢	٦٢٢	٢٥٦	٥٦٧
رقم التظليل	٩	١٤	١٦	٢٥	٢٧	٢٨	٥٨	٧٢

ويتضح من الجدول أن التسع سجلات وزعت على أمتداد ٧٢ موقعاً وكل سجل فى موقع فريد.

وماذا لو طلب إضافة السجل رقم ٥٥٣ إلى الملف ، علما بأن رقم تظليله ٥٨ وهو نفس رقم تظليل السجل ٢٥٦ المخزن فعلا على الملف بمعنى أنه مطلوب تخزين سجلين فى نفس الموقع على القرص المغناطيسى وسوف يحدث تصادم Collision بين السجلين ٢٥٦ ، ٥٥٣ حيث أصبحا Synoyms أى لهما نفس قيمة التظليل (٥٨).

حقيقة هناك أساليب فى إختيار وتقييم دالة التظليل مما يقلل من التصادم لحده الأدنى ومع هذا فلازال التصادم مشكلة قائمة تتطلب حلاً؟

ومن الحلول الشائعة فى تقليل التصادم إتخاذ عده إجراءات كما فى طريقة البحث الخطى Linear Search التى تعتمد على أستخدام:

أ ــ دالة تظليل تحسب رقم القطاع وليس موقع السجل مع زيادة حيز القطاع من ٥١٢ بايت إلى ١٠٢٤ بايت.

ب ــ حتى بعد زيادة حيز القطاع إذا حدث تصادم على قطاع عند إضافة أحد السجلات فيتم تخزين السجل فى اقرب قطاع به محل خال.

وتحقق هذه الطريقة تواجد عناقيد من السجلات المتقاربة على قطاعات متقاربة مما يرفع من كفاءة الإسترجاع ويجعل البحث عن السجل عملية سهلة على الحاسب لأن الآلات سوف تبحث عن السجل المطلوب فى أقرب قطاعات ورغم هذا تقل كفاءة الإسترجاع والتخزين فى حالة تضخم الملفات حيث تتطلب البحث فى آلاف السجلات قبل تخزين أو استرجاع أى سجل.

طريقة الفيض OVERFLOW :

وتعنى إتاحة قطاعات خالية تماماً ومحددة مسبقاً لإدخال وتخزين أى سجل يتصادم مع أى سجل آخر فى أى قطاع مع وضع مؤشر عن وجود هذا السجل فيما

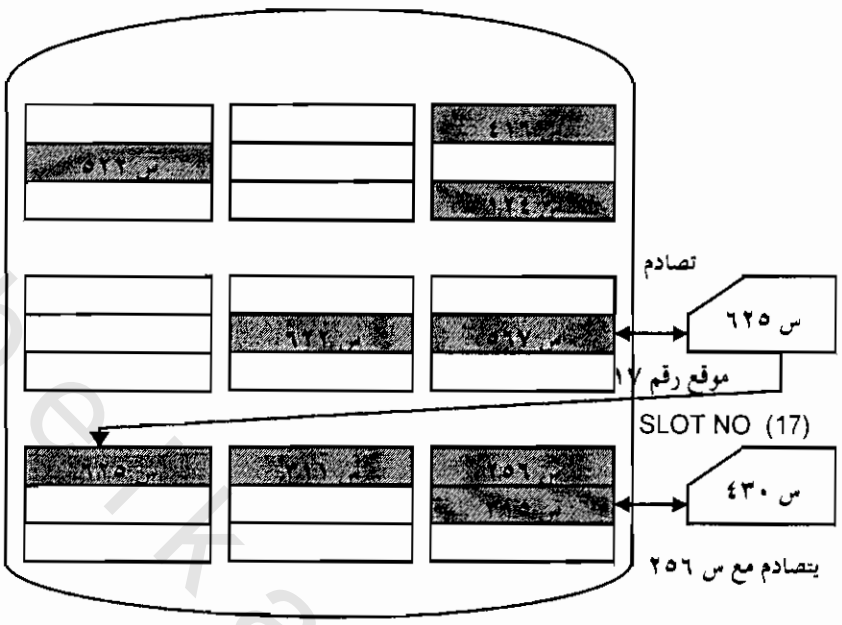
يوضحه الشكل (١٢/٩) وتمتاز هذه الطريقة عن طريقة البحث الخطى فى أن تسجيل أو استرجاع أى سجل متصادم يتم من خلال حيز محدود مما يساعد على الإسترجاع ، علاوة على أن البحث الخطى وإن أعتمد على تكوين عناقيد من سجلات متقاربة إلا أن السجل فى القطاع المتجاور التالى سوف يخلق مشكلة تصادم مستقبلية.

زيادة كفاءة البحث الخطى وطريقة الفيض POINTER CHAIN :

إشارة الإحالة ومؤشر الإحالة:

أحيانا كثيرة نجد فى فهارس المراجع عبارة أنظر أيضاً ، فإذا كنا نبحث عن معنى كلمة حاسب فإن الفهرس يحدد إن هذا الموضوع ورد فى صفحة كذا ونجد إحالات إلى مسميات أخرى مثل الحاسب الإلكترونى ، والحاسوب ، والعداد الإلكترونى... نفس هذا الأسلوب استخدم فى زيادة كفاءة طريقة البحث الحظى والفيض بأستخدام إشارة إحالة أو مؤشر Pointer ، مما يطلب أستخدام حقل إضافى أو بايت إضافية على حيز القطاع لتوضيح رقم القطاع المحال إليه كما هو موضح بالشكل (١٢/١٠) مما يقلل من عدد القطاعات التى يجب فحصها للوصول للسجل المتصادم.

أستخدم فى تسكين السجلات دالة تظليل أدت إلى تصادم بين السجلين أرقام ٥٦٧ و ٦٢٥ فيتم تسكين الأول فى الموقع النسبى ١٧ من الكتلة [(6) B] ولابد من التصرف حيال تخزين السجل رقم ٦٢٥ فإذا طبقنا مبدأ البحث الخطى نجد أن السجل ٦٢٥ يخزن فى الكتلة [B7] فى الموقع النسبى SLOT رقم (١٩).



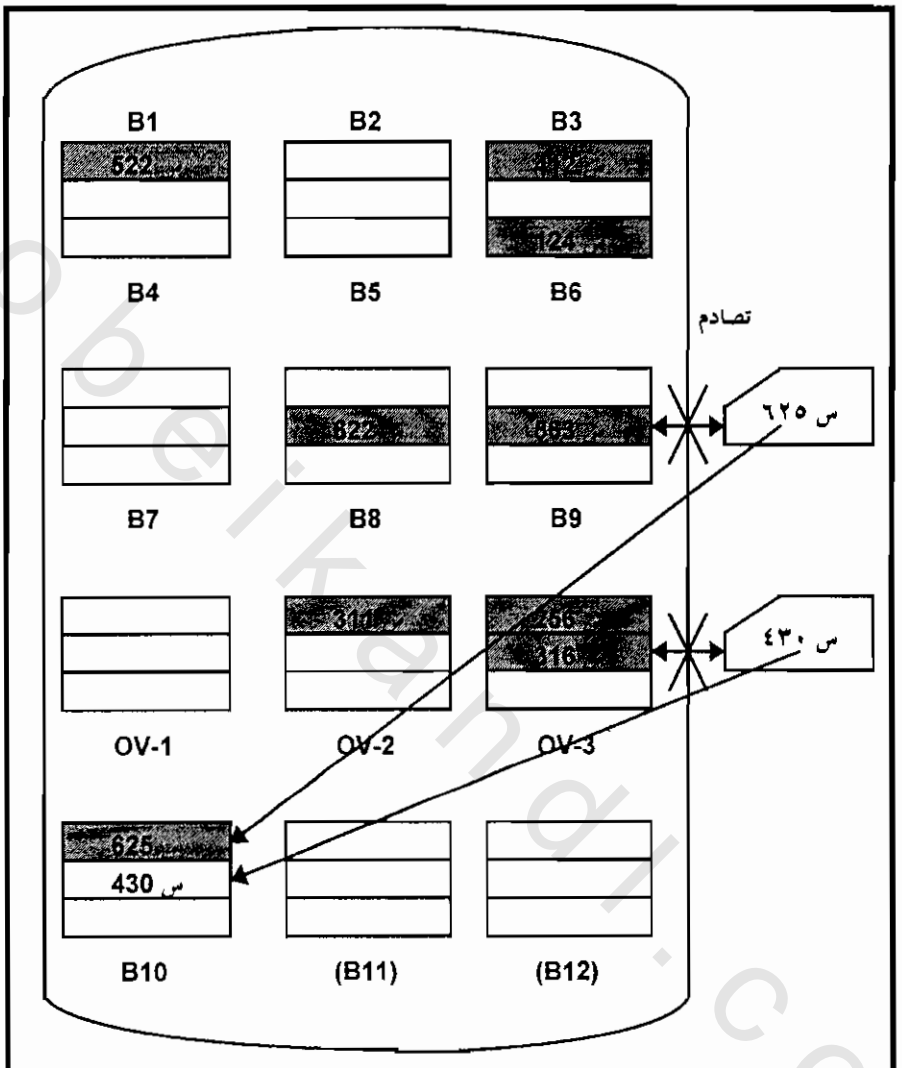
ملحوظة:

الكتل ذات حيز متساو وكذلك SLOTS وإن لم يبرز الرسم ذلك.

شكل (١٢/٨)

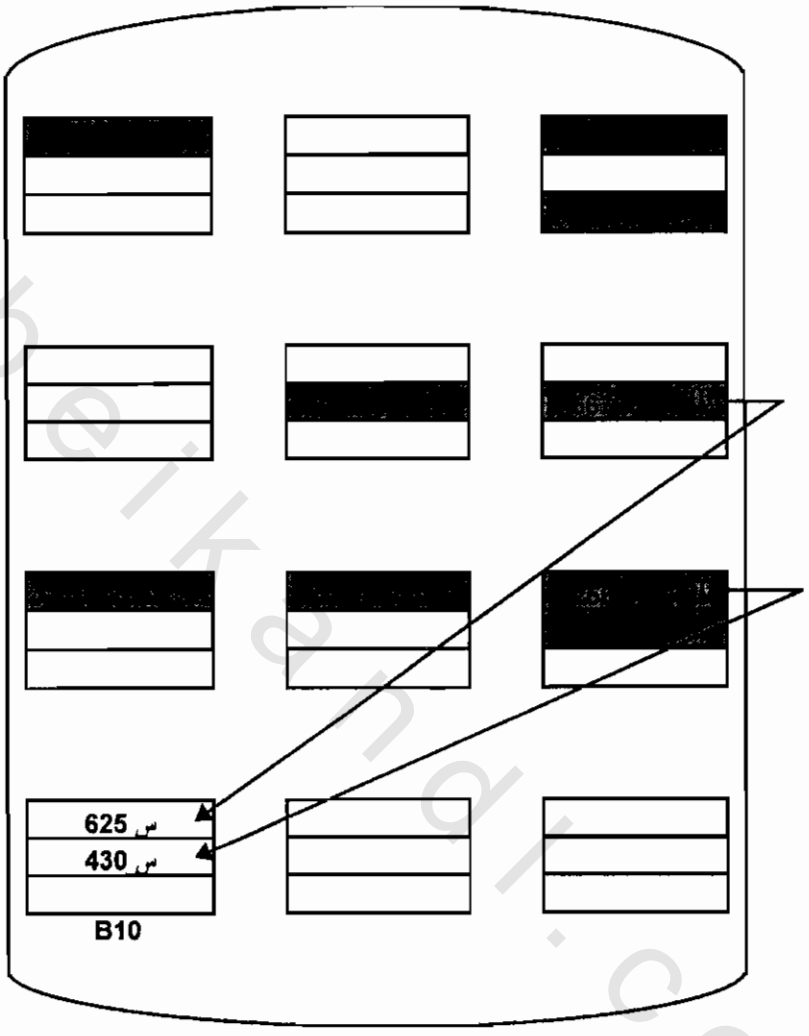
بديلا عن التسكين فى الكتلة التالية للتصادم يتم تسكين السجل المتصادم من منطقة فيض خاص بالملف.

ولتحسين أداء المنظومة وزيادة فاعليتها يستخدم حيز مقداره بايت واحدة يسجل به رقم منطقة الطفو التى تم تسكين السجلات المتصادمة وتسمى فى هذه الحالة:



شكل (١٢/٩)

أستخدام منطقة للفيض



شكل (١٢/١٠)

أسلوب الإحالة والمؤشرات

obeikandl.com

أو

POINTER CHAINS

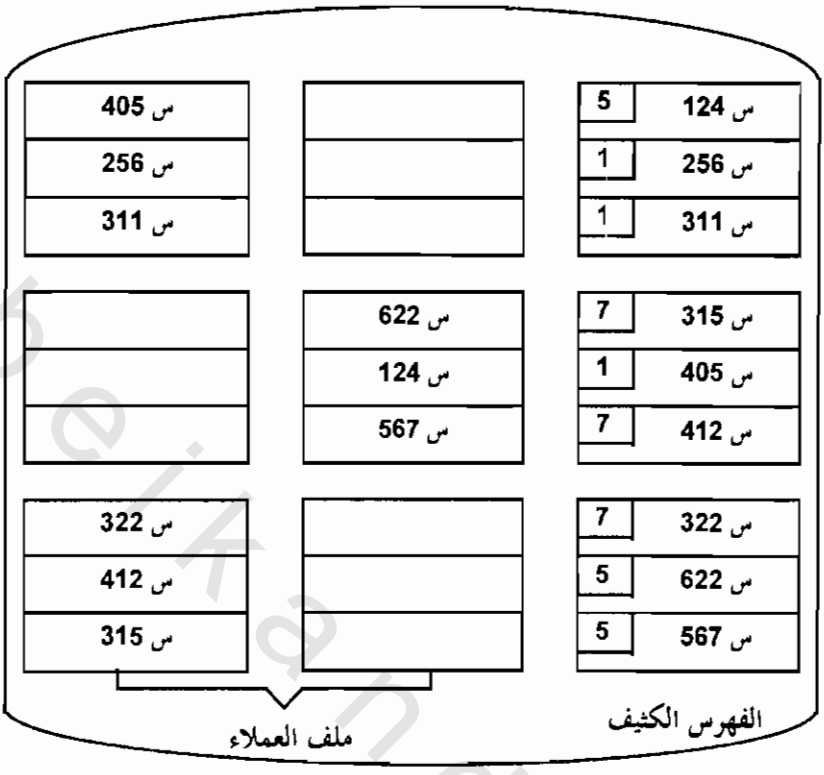
SYNOYM CHAINS
COLLISION CHAINS

٢ ــ التنظيم المفهرس Indexed Organization :

يرتكن هذا التنظيم على انشاء ملف صغير يسمى ملف الفهرس Index File إلى جوار ملف البيانات Data File ويعتبر ملف الفهرس المدخل الأساسى لملف البيانات ، ويتكون سجل ملف الفهرس من حقلين ، الأول المفتاح الرئيسى أو حقل المفتاح لسجل البيانات الى جانب حقل الفهرس ، مما يعنى أن كل سجل بيانات له مدخل وحيد فى ملف الفهرس ، ونؤكد مرة اخرى ملف مفهرس يعنى ملفين ، ملف بيانات وجواره ولصيق به ملف فهرس.

أ ــ تنظيم عشوائى مفهرس Indexed Random Organization :

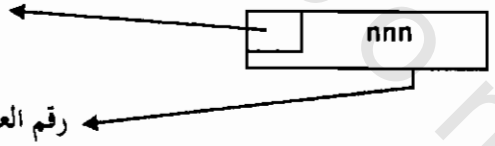
عندما يتم كتابة سجلات ملف البيانات دون ترتيب أو تحديد على وسائط التخزين فهذا ما نعنيه تماما بمفهوم التنظيم العشوائى المفهرس فيما يوضحه الشكل (١٢/١١) سجلات لعملاء موزعة على ثلاث قطاعات ثلاث كتل ــ وإلى جواره سجل الفهرس الذى يضم حقلين كما هو موضح مما دفع إلى عدة تساؤلات عن الفائدة المحققة من أستخدام الفهارس خاصة وهو ملف سوف يشغل حيزا لا يستهان به من وسائط التخزين ويتطلب إنشاء وصيانة وتحديث ومعالجة شأن كل الملفات؟ هل التكلفة الإضافية لإنشاء الفهرس تعطى مردودا يستدعى هذا الجهد والتكلفة؟



ملف عشوائى مفهرس

ذا فهرس كثيف

ملحوظة: غير خاف أن كل SLOT فى حيز ملف الفهرس يستوعب أكثر من سجل فهرس وأن كان الرسم لم يوضح ذلك .



شكل (١٢/١١)

التنظيم المفهرس

ومجموعه الأسئلة المثارة توضح مدى أهمية الفهارس ، فهذا التنظيم يتيح الإسترجاع المتتالى رغم أن سجلات البيانات موزعة عشوائيا أثناء التسجيل.. كيف؟ بإعادة ترتيب ملف الفهارس وسلسلة مفتاح سجل البيانات يتيح هذا النوع من الإسترجاع ، أيضا يمكن إسترجاع سجلات البيانات عشوائيا دون جهد باجراء البحث فى ملف الفهرس وصولا إلى القطاع ــ الكتلة ــ التى تضم السجل المنشود مباشرة مما يقلل من زمن الاسترجاع والذى ينعكس بدوره على كفاءة النظام لان التعامل مع ملف سجلاته لا تتعدى ٨ بايت أسرع بكثير جدا من التعامل ومعالجة سجلات قد يصل حيز السجل الواحد فى ملف البيانات مئات أو عده آلاف من البايت.

تطوير ملف الفهرس :

كقاعدة كلما تضخم ملف البيانات كلما أمتد وتضخم ملف الفهرس ويصبح عبئا على النظام من منظور الحيز التخزينى وسرعة التعامل لذلك تنشأ ملفات من الفهارس أو فهارس متعددة المستويات Multilevel Index (فهرس للفهرس) بحيث تعتبر الطبقة الأولى أو الفهرس الأصلى (فهرس مكثف Dense) وفهرس الفهرس فى الطبقة العليا فهرس مختصر Sparse لأنه لا يضم كل مداخل السجلات ، ويوضع الفهرس المختصر فى الذاكرة الأساسية مما يقلل من زمن إسترجاع أى سجل على ملف البيانات على النحو:

أ ــ يقرأ الحاسب المستوى الأعلى من الفهرس المخزن فى الذاكرة دون عمليات I/O على الأقراص وقطعا فى زمن ضئيل للغاية ومنه يحدد الكتلة المحتوية على رقم مفتاح سجل البيانات.

ب ــ من المستوى الكثيف للفهرس يحدد رقم الكتلة المناظرة لسجل البيانات المطلوب.

وهذا التطوير فى الفهارس وبنائها على أكثر من مستوى ما يعرف بمسمى B-Tree Index .

ولعل من أبرز مزايا هذه الفهارس إمكانية تحديد وجود أى سجل بيانات من عدمه وذلك باقتصار البحث فى ملفات الفهارس وليس على ملف البيانات.

التنظيم المتسلسل المفهرس Indexed Sequential :

يشمل مزايا ملف الفهارس إلى جانب أن سجلات البيانات مرتبة ترتيبا متتاليا وفق حقل المفتاح ، وفى هذه الحالة ليس هناك حاجة إلى فهرس كثيف ، ويكتفى بفهرس مختصر مما يساعد على سرعة البحث وتقليل زمن إستجابة النظام مع أستخدام حيز تخزين محدود وقد يوضع الفهرس المختصر ذاته فى الذاكرة مما يرفع من كفاءة وإستجابة النظام.

وتتعجب من مزايا التنظيم المتتالى المفهرس ونتساءل ولماذا نستخدم التنظيم العشوائى المفهرس ومشاكل تطوير فهارسه؟

الواقع أن سجل البيانات يمكن فهرسته على أى حقل مكونا ما يسمى الملفات المعكوسة Inverted List مما يساعد على وجود مداخل متعددة لدرجة أنه يمكن أستخدام جميع الحقول المتاحة فى سجل البيانات كمفاتيح ثانوية وهذا لا يمكن تطبيقه مع التنظيم المتتالى المفهرس لأنه فى هذا التنظيم لابد من أستخدام المفتاح الرئيسى لسجل البيانات دون سواه.

التعامل مع ملفات ISAM :

INDEXED SEQUENTIAL ACCESS METHOD :

عندما نحتاج إلى إضافة سجل جديد إلى هذا النوع من الملفات فإن من الأفضل عدم التعامل مع الملف مباشرة بل يضاف السجل الجديد إلى منطقة طفو مع أستخدام مؤشر الإحالات كما فى الشكل (١٢/١٢) مع إضافة حقلين لكل كتلة: الحقل

الأول يشير إلى رقم كتلة الطفو ويتكون من رقم القطاع والرقم النسبى للسجل داخل القطاع ، بينما يستخدم الحقل الآخر للإشارة إلى أعلى قيمة مفتاح متواجدة فى منطقة الطفو.

مميزات الملف المتسلسل المفهرس :

(١) يعمل ملف الفهارس كمداخل لملف البيانات.

(٢) مناطق الفيض للملف خارج حدود الملف وبالتالى يمكن علاج زيادة حيز السجلات كما تستخدم فى عمليات الإضافة على الملف.

(٣) الفهارس من نوع BEE TREE INDEX (B) .

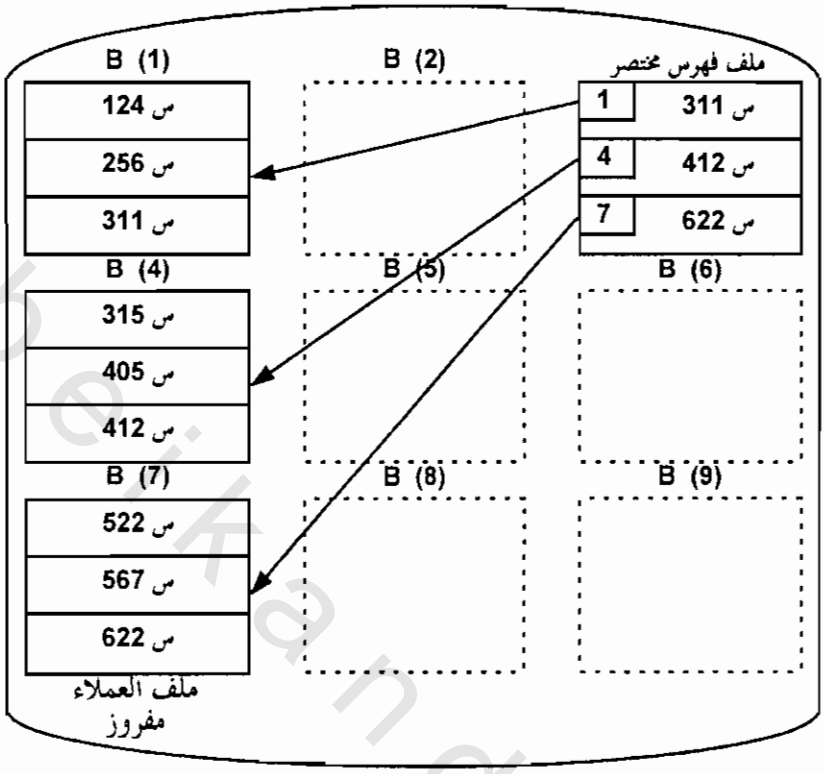
(٤) سهولة إعادة تنظيم الملف مقارنة بالملفات المباشرة.

مميزات الملفات المباشرة :

يتم تحديث الملفات بأسلوب أسرع مقارنة بالملفات المتتالية حيث يتم كتابة السجل بعد تحديثه فى نفس موقعه القديم مادام لم يتغير حيزه على وسيط التخزين ، كما أن عمليات الإضافة والحذف لا تستغرق وقتا يذكر.

عيوب الملفات المباشرة :

تنشأ معظم مشاكل هذه الملفات من أستخدام طريقة الـ HASHING فى حساب العناوين ، ناهيك عن المشاكل الناجمة عن تضخم (OVERFLOW فيض) الملف عن الحيز أو الحجم المقدر له عند بداية إنشائه والذى ينعكس بشدة على كفاءة إسترجاع المستخدمين وفى جميع الأحوال وعلاجا لتلك المشاكل يجب إعادة تنظيم الملف وكتابته من جديد ، وهى عملية تتطلب جهدا ليس يسيرا.



تنظيم متتالى مفهرس

مع أستخدام فهرس كثيف

ملحوظة: الكتل أرقام ٢ ، ٥ ، ٨ يمكن أستخدامها فى إنشاء فهرس غير كثيف يضم كافة مداخل السجلات.

ENTRIES OF ALL RECORDS

شكل (١٢/١٢)

التنظيم المتتال المفهرس

مع أستخدام فهرس كثيف

obeikandl.com

مزايا نظم الملفات:

١ ــ الملفات المتتالية تستهلك من حيز التخزين ادنى قدر متاح خصوصا إذا كانت مكتلة ، رغم أن الإسترجاع المتتالى يبرز عدم مرونة هـذا التنظيم.

٢ ــ تمتاز الملفات المباشرة بأقصى سرعة إسترجاع لكنها غير مناسبة من حيث الإسترجاع المتتالى.

٣ ــ التنظيم المفهرس يتيح للمستخدم الإسترجاع المتتالى والمباشر رغم أن الأخير ليس على السرعة المناسبة وهنا يتفوق التنظيم المتتالى المفهرس.

٤ ــ معظم نظم الملفات تلبى متطلبات النظم غير النشطة مثل أنظمة المرتبات والمخزون مما يستدعى أستخدام تكلفة إضافية.

٥ ــ من أبسط الأساليب فى أدارة البيانات رغم العيوب الكثيرة التى تعانى منها.

عيوب نظم الملفات:

حتى السبعينات ونظرا للقصور الكبير فى معدات الحاسبات صممت معظم التطبيقات على مبدأ الإرتباط المباشر بين البرنامج والبيانات الخاصة بها فيما عرف بأسم DATA - PROGRAM - DEPENDENCE ، فعندما كان يكتب برنامج بلغة الكوبول فإنه على الفور يتم إنشاء ملف يضم البيانات اللازمة لهذا البرنامج ، وعندما يكتب برنامج بلغة البيزك ينشأ معه ملف آخر تطابق صياغة بياناته برامج البيزك وهكذا.

حقيقة أن كل برنامج يمكنه التعامل مع أكثر من ملف لكنه يتعامل معها تتابعيا لأنه لا يمكنه التعامل مع أكثر من ملف واحد فى الوقت الواحد ، شريطة أن تكون الملفات مكتوبة بياناتها بطريقة تلائم لغة البرمجة المستخدمة كما أسلفنا.

هذه الحقيقة كان لها نتائجها السلبية التى نلخصها على النحو:

١ ــ حدت من مرونة النظام.

٢ ــ سببت نقص كفاءة النظام.

٣ ــ أتاحت وجود تكرارية فى البيانات الواحدة.

٤ ــ رغم الإرتباط المباشر والبيانات ولغة البرمجة فليس متاحا أستخدام بيانات برنامج آخر رغم أنهما مكتوبان بنفس لغة البرمجة مما حتم أستخدام نسخ جديدة من ملف البيانات ليلائم البرنامج الآخر.

٥ ــ تسببت تكرارية البيانات فى عدم تحقيق التكاملية بينها لأن تحديث أحد الملفات لا يعنى بالضرورة تحديث الآخر.

٦ ــ أنعكس كل هذا على زيادة كبيرة فى وسائط تخزين البرامج والبيانات.

كل هذه المشاكل قادت إلى التفكير فيما عرف بأسم قواعد البيانات لإحداث تكاملية بين مختلف الملفات وإنهاء مشكلة تكرارية البيانات... الخ.

وقد سميت هذه الفكرة تكاملية الملفات INTEGRATED FILES ثم شاع مسمى قواعد البيانات DATA BASE وتتلخص الفكرة فى وضع البيانات ضمن إطار موحد دعنا نطلق عليه للتبسيط حوض البيانات DATA POOL أو قواعد البيانات DATA BASE بحيث يستطيع مختلف مستخدمى النظام التعامل معها بشكل سهل ومبسط ، وهذا حديث لا محل له فى هذا الكتاب.

أساليب إسترجاع الملفات:

تعرف الملفات بواسطة أمرين مرتبطين ببعضهما البعض لدرجة أصبح ذكر أحدهما مرادفا للآخر وهما:

* **التنظيم.**
* **أسلوب إسترجاع.**

obeikandl.com

فالملف الذى تم تنظيمه متسلسلا يسانده نظام إسترجاع متسلسل سيان كانت السجلات من الفئة ذات الطول الثابت (F) أو الفئة ذات الطول المتغير (V) ، وبالتالى يستخدم مدير الملفات عنوان آخر بايت فى السجل لإسترجاع السجل التالي له وترتيبا على ذلك فإن هذا العنوان يجب تحديثه كلما قرأ النظام سجلا مثلما يحدث عند تنفيذ الأمر أقرأ. ويطلق على عنوان آخر بايت عنوان البايت الحالى CURRENT BYTE ADDRESS (C.B.A) .

١ ــ أساليب إسترجاع الملفات المتسلسلة:

فى هذه الملفات فإن تحديث ABC يتم بإضافة طول السجل RECORD LENGTH (R.L) إذا كان السجل من النوع (F) .

CBA = CBA + RL

أما إذا كان السجل متغير الطول (V) فإن مدير الملف يضيف إلى جانب طول السجل... عدد البايت المستخدمة فى تسجيل طول الملف وفق القانون:

CBA = CBA + RL + N

٢ ــ أساليب إسترجاع الملفات المباشرة:

فى حالة الملفات المباشرة وسجلاتها ذات طول ثابت (F) يتم حساب قيمة الـ CBA من المعادلة التالية:

(رقم السجل - ١) مضروبا فى طول السجل أى .....

[(ن -١) × طول السجل]

حيث ن تساوى رقم السجل المطلوب

فى حالة الملفات المباشرة ، وسجلاتها ذات طول متغير (V) :

فى هذه الحالة يصعب على مدير الملفات تحديد موقع السجل بإجراء عمليات حسابية كما فى الحالة السابقة وكحل بديل فإن مدير الملفات يحتفظ بفهرس

لأرقام السجلات و CBA المناظرة لها وعند أستدعاء سجل يقوم مدير الملف بفحص هذه القائمة وتحديد موقع السجل المطلوب.

وجدير بالقول أن هناك عدة نظم تشغيل تجبر مستخدميها على أستخدام سجلات ذات طول ثابت طالما سيتم أعداد الملف وفق أسلوب المباشر.

٣ ـ الملفات المتتالية المفهرسة:

يمكن النظر الى الملف المفهرس بإعتباره ملف مختصر ـ أقل حجما ـ من الملف الأصلى ، وفى حالة أستخدام شبكة هرمية من الفهارس مثلB -TREE فإن فهرس الفهارس يكون ذا حجم ضئيل وبالتالى يمكن مع بدء التشغيل تحميله فى الذاكرة الأساسية لسهولة أستخدامه.

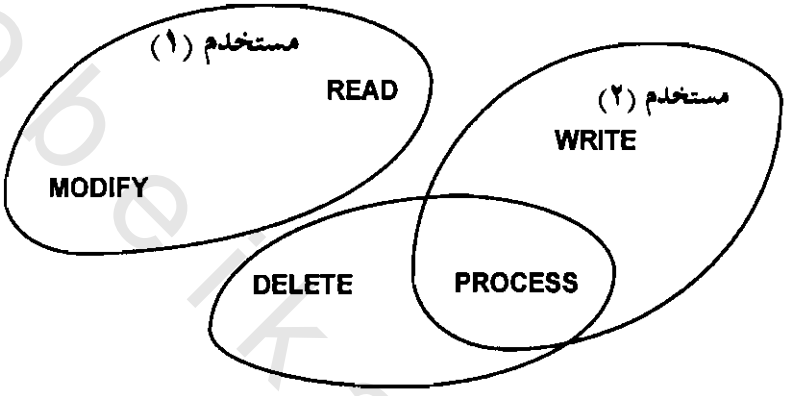
وإلى جانب أن الملفات المتتالية المفهرسة يمكن إسترجاعها بأى من الأسلوبين السابقين ، إلا أنه تسبقها خطوة ضرورية هى مراجعة شبكة الفهارس لتحديد الكتلة المنطقية والصفحة المحتوية على السجل المطلوب والتى يجرى قراءتها من على وسائط التخزين الثانوية بعدها يتم حساب عنوان أول بايت فى السجل (CBA) وتحديد موقع السجل المطلوب.

قيادة التعامل مع الملفات:

حققت نظم التشغيل الحديثة أهم إنجازاتها فى إمكانية المشاركة على الملف الواحد لعدة مستخدمين حيث حققت بذلك وفرا كبيرا فى موارد وسائط التخزين مع وضع عمليات تحديث البيانات تحت ضوابط ومعايير كثيرة مما حسن أداء النظم على المستوى المنطقى ووفر فى عمليات I/O على الملفات قدر الإمكان.

وكشأن أى تقدم علمى وتكنولوجى له وجهان ، وجه يحقق مزايا ووجه آخر يعبر عن مساوئ هذا التقدم ، وقد فرض هذا التقدم ضرورة وضع أساليب للسيطرة

على التعامل مع الملفات كرد على التساؤل [من يتعامل مع أى ملف وماذا يفعل خلال هذا التعامل.. هل يقرأ / يكتب / يحذف / ينفذ / يعدل / أو يجرى تعاملات مزدوجة] وما هى صلاحيات هذا المستخدم ، لذلك ضمت نظم التشغيل الحديثة عدة أساليب لقيادة التعامل مع الملفات هى:



١ ـ مصفوفة الملفات حيال المستخدمين:

وتضم المصفوفة أسماء الملفات وأسماء المستخدمين والصلاحيات وفئة العمليات على كل ملف من واقع شكل منه ، وتسجل المصفوفة فى النظام وتكون هى الإطار المرجعى لنظام التشغيل لأعطاء صلاحيات للمستخدم.

وتصلح مثل هذه المصفوفات فى النظم المحدودة من حيث الملفات وعدد المستخدمين ، ولا تصلح فى النظم الكبيرة لما تسببه من تأثيرات سلبية على كفاءة النظام.

المستخدمين / الملفات	م	ن	ل	ك	ع	ص
١	١م	٠	٠	٠	٠	٠
٢		٠	٠	٠		
٣	٠	٠	٠	٠	٠	٠

وتوجد من هذه المصفوفة أشكال معدلة لعل أبرزها:

• **قائمة قيادة التعامل:**

ويكون مدخلها أسم الملف وحياله أسماء المستخدمين وصلاحياتهم مثال:

ملف رقم (١) : حسن (الصلاحيات) ، ماجدة (الصلاحيـــات) ، موريـــس (الصلاحيات).

ملف رقم (٢) : حسن ( ) ، عويس ( ).....

٢ ــ **الكلمات الحاكمة LOCK WORD :**

تختلف الكلمة الحاكمة عن كلمة السر PASS WORD فى أن الثانية تســتخدم للسيطرة على التعامل مع النظام كله كوحدة متكاملة ، فــى حيــن تســتخدم الكلمة الحاكمة للسيطرة على التعامل مع الملفات ذاتها.

وتقع مسئولية إختيار الكلمة الحاكمة على صاحب الملـــف ، ويتــم رصــد الكلمات الحاكمة آليا على ملف خاص ضمن مجموعة فهارس النظام ومـــع ذلك ينال هذه الملف أعلى درجات سرية متاحة فى النظــام ويعتــبر شفافــا TRANSPARENT حيال المستخدمين ويستحيل طباعته أو أستعراضه علـــى الشاشات.

*

obeikandl.com