The AuditNet® Monograph Series
Computer Assisted Audit Techniques
Guide to Downloading Data

Student Copy
INTOSAI
AuditNet® Monograph Series - Guides for Auditors

This monograph series grew out of my desire to establish an online electronic communication network for auditors. Before online services, bulletin boards and the Internet many auditors were operating without the benefits of peer collaboration and information sharing on a major scale. The Internet, founded on the principle of sharing and communication, changed the interaction model between auditors. Auditors can now post messages in online discussion forums, upload and download audit work programs, checklists, surveys, questionnaires and other audit related material in warp speed. Small one-person audit shops can now communicate with others and feel like they are not paddling upstream with one oar when it comes to having access to audit resources. My vision of an online information communication network for auditors became a reality with AuditNet® as the foundation.

The AuditNet® Monograph Series or AMS provides auditors with guidance on different aspects of the audit process and other relevant topics to help them do their jobs. New auditors will seek these guides to learn some basics of auditing while experienced auditors will use them as a review. Each guide focuses on a specific subject.

If you have an idea for additions to the AMS please send a proposal via email to editor@auditnet.org.

Jim Kaplan, AuditNet® Founder and Principal
Acknowledgement

The material included in this monograph was created by INTOSAI, the professional organization of supreme audit institutions (SAIs) in countries that belong to the United Nations or its specialist agencies for a training program on downloading data. SAIs play a major role in auditing government accounts and operations, and in promoting sound financial management and accountability in their governments.

AuditNet® thanks Ian Pettigrew of the UK National Audit Office for permission to reproduce this student guide for downloading data in the form of a monograph for the benefit of the global audit community.
# Data Downloading Guide

## Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acknowledgement</td>
<td>4</td>
</tr>
<tr>
<td>1. Introduction</td>
<td>7</td>
</tr>
<tr>
<td>1.1 Aims and Objectives</td>
<td>7</td>
</tr>
<tr>
<td>Data identification and Acquisition</td>
<td>7</td>
</tr>
<tr>
<td>2.1 Planning a CAAT</td>
<td>7</td>
</tr>
<tr>
<td>2.2 Objectives and Assurances</td>
<td>8</td>
</tr>
<tr>
<td>2.3 What data is available?</td>
<td>8</td>
</tr>
<tr>
<td>2.4 The flow of data in a system</td>
<td>9</td>
</tr>
<tr>
<td>2.5 Identifying the file for interrogation</td>
<td>9</td>
</tr>
<tr>
<td>2.6 Specifying the information required for the CAAT</td>
<td>10</td>
</tr>
<tr>
<td>2.7 Specifying the format of the data file</td>
<td>10</td>
</tr>
<tr>
<td>2.8 File downloading and conversion</td>
<td>11</td>
</tr>
<tr>
<td>2.9 The Standard Requirements for data</td>
<td>11</td>
</tr>
<tr>
<td>2.10 Modes of data storage / transfer</td>
<td>12</td>
</tr>
<tr>
<td>2.11 Who or What can help with Data Identification / Acquisition</td>
<td>12</td>
</tr>
<tr>
<td>3. What is data?</td>
<td>13</td>
</tr>
<tr>
<td>3.1 Forms of data/communication</td>
<td>13</td>
</tr>
<tr>
<td>3.2 Electronic data formats</td>
<td>13</td>
</tr>
<tr>
<td>3.3 Binary and Hexadecimal</td>
<td>14</td>
</tr>
<tr>
<td>3.4 Bits and Bytes</td>
<td>14</td>
</tr>
<tr>
<td>3.5 The value of Hexadecimal</td>
<td>14</td>
</tr>
<tr>
<td>3.6 The Hex dump</td>
<td>15</td>
</tr>
<tr>
<td>3.7 Why use Hexadecimal</td>
<td>16</td>
</tr>
<tr>
<td>4. Character Representation</td>
<td>16</td>
</tr>
<tr>
<td>4.1 ASCII and EBCDIC</td>
<td>16</td>
</tr>
<tr>
<td>4.2 EBCDIC data</td>
<td>17</td>
</tr>
<tr>
<td>4.3 ASCII data</td>
<td>17</td>
</tr>
<tr>
<td>4.4 Zoned Decimal</td>
<td>18</td>
</tr>
<tr>
<td>4.5 Signing Zoned Decimal</td>
<td>18</td>
</tr>
<tr>
<td>4.6 Packed Decimal</td>
<td>19</td>
</tr>
<tr>
<td>4.7 Other forms of Packed Decimal data</td>
<td>19</td>
</tr>
<tr>
<td>4.8 Binary Data</td>
<td>20</td>
</tr>
<tr>
<td>4.9 Floating Point data</td>
<td>20</td>
</tr>
<tr>
<td>5. Record and File layouts</td>
<td>21</td>
</tr>
<tr>
<td>5.1 The Record Layout</td>
<td>21</td>
</tr>
<tr>
<td>5.2 HEX dump utilities</td>
<td>22</td>
</tr>
<tr>
<td>5.3 The typical HEX dump</td>
<td>22</td>
</tr>
<tr>
<td>5.4 The typical record layout</td>
<td>23</td>
</tr>
</tbody>
</table>
1. Introduction

1.1 Aims and Objectives

This guide is intended as an introduction to file downloading techniques. Its aim is to give auditors an understanding of the tools that are available to them and when and where to use them. This guide is also intended to provide the trainee with a detailed knowledge of file formats, data types and modes of data transfer.

At the end of the guide the auditor shall:

- Understand the definition of data - The forms which data can take and the problems involved with the communication of data. The student should have knowledge of electronic data types and how data is stored in the PC (Binary);
- Understand character representation - How data is represented on the PC covering Binary and Hexadecimal. The students will be aware of the function of the HEX display and the reasons for its use;
- Be aware of the types of data that exist - How data is represented in ASCII and EBCDIC, the varying forms of numeric data in EBCDIC. How to recognize these data types and the implications they have for file downloading;
- Understand the importance of record and file layouts - The structure of a data file and its storage on a tape. How to prove the completeness of a data file and how to recognise the various ways data can be stored on a tape;
- Downloading tools - The downloading tools currently available. Specific attention will be paid to Fdump, Looktape and “off the shelf” and “In house” utilities;
- Data transfer and Manipulation - The methods of transferring data to the PC and the manipulation of the data to produce a file compatible with the file interrogations software used for the audit. The student will also be aware of some of the hardware available for file downloading.

Data identification and Acquisition

2.1 Planning a CAAT

- It is essential to plan a CAAT properly before beginning development. The CAAT needs to be looked at as part of the whole audit. The objective of the CAAT is to assist the auditor in reaching the assurances he/she needs to certify an account;
- It is vital for the auditor to select the most appropriate file / files for CAATS interrogations and care needs to be taken to investigate the data files that could be available. The auditor should make use of flow charts of the client department’s system to ascertain which files are being used. It is important to select data that is as close to the original data entry process as possible;
- When the required data has been identified it is important to liaise with the audited department to arrange a suitable format for the data. The data needs to be in a format that
is compatible with the interrogation software being used by the audit department or a format that can be easily converted by the CAATs specialist;
• Data will also need to be provided on a magnetic media that is compatible with the hardware used by the audit department. If a decision is made to send the data across a network or via a modem, it is advisable to discuss any virus or security implications with the IS security officers of the client department and the audit body;
• It is also important for the auditor to establish contacts with both the CAATs specialists in his/her organisation and the IT specialists of the audited body. This information is useful for future development work and CAAT administration;
• Spending time in the planning of a CAAT can save a great deal more time during the audit.

2.2 Objectives and Assurances
• For any type of audit it is necessary to get to know the function and accounting practices of the body to be audited. This knowledge will enable the auditor to form a plan more clearly.
• It is essential to plan the forms of interrogation that will need to take place: For financial audit:
  * Reconciliation (completeness);
  * High value / Key item detection;
  * Exception reporting (population extraction);
  * Sampling;
• For VFM audit:
  • The nature of the study along with a knowledge of the departments practices will determine the forms of interrogation;
  • Ageing;
  • Gap Detection;
  • Duplicate Payments;
• A plan of action / interrogation must exist before a decision is made on the data to use;
• For Forensic audit:
  * A detailed knowledge of the procedures used in a system will determine the files required and the interrogations required before any work can take place. Forensic interrogations could include:
  ♦ Exception reporting;
  ♦ Fuzzy matching;
  ♦ File comparison;
  ♦ Gap detection.

2.3 What data is available?
It is always necessary to determine the data files that are being used by the audited body. This information is important to the proper planning of a CAAT. The auditor or the CAATs specialist will need to know:
• The operating system used by the audited body. Different operating systems will be able to create different types of output files. A thorough knowledge of the operating
system will allow the auditor or CAATs specialist to determine the most appropriate file type for their interrogation software;

- **The files created by the system.** Detailed knowledge of the system will allow an informed decision to be made as to the most appropriate data to use for the audit. For example, The CAATs specialist or auditor would be more likely to select a file containing all transactions entered into the system rather than a finalised General Ledger file that has been through several stages of manipulation.

There are various ways to gather this information:

- **Discussions with the IT personnel of the audited body.** It is usual for the CAATs specialist to assist the auditor to select the appropriate data and file type by accompanying the auditor to the client department for discussions with the IT personnel. The CAATs specialist having knowledge of both audit requirements and IT can make sure that the correct questions are asked and the answers are interpreted correctly;

- **Study of Installation/Application reviews.** The results of past reviews will give a lot of information regarding the system and the files used by the client department;

- **Inspection of system flow charts.**

### 2.4 The flow of data in a system

- It is possible to have more than one file in a client’s system. It is quite common for all of the transactions used by the client to be entered into one file. The data is sorted and transferred to other files at a later stage;

- It is important for the auditor to know the files that exist in the client’s system in order to identify the most appropriate file for audit interrogations;

- It is common for all financial transactions to be entered into a master file. The transactions may be split by a later process into separate files containing like transactions (Accounts, Pay.....). This data may then be posted to particular Votes or Accounts that go towards creating the overall annual Account. During the splitting process all transactions containing errors may have been corrected or removed or a separate Journal file created to balance the account. The final General Ledger file may only contain overall totals from all of the transaction for each discreet area of the account;

- The auditor will need to select the most appropriate file for sampling. In this case the Original transaction file or the Vote ledger files would probably be the file selected as each show individual transactions.

### 2.5 Identifying the file for interrogation

- When the auditor is selecting a file for CAATs interrogations It is always a good idea to select a file that is closest to the raw data or initial input file. In most cases this data is in its simplest format and therefore easier to import into a file interrogation package;

- If a file is created from several smaller files (feeder files) there has been some processing of the data prior to the creation of the file. The closer the file to the final account or General ledger the more data processing has taken place;
• If the auditor is confident in the data processing undertaken by the audited body (Systems Review) he/she may be satisfied to use the general ledger file with screened data;
• The data selected must be specific to the audit and relevant;
• It would be quite rare to use a summarised file as individual transactions no longer exist making the sampling process quite difficult;
• It is advisable to liaise with CAATs specialists and the IT operators at the audited body in order to make an informed decision.

2.6 Specifying the information required for the CAAT
• When the most appropriate file for the audit has been selected there may be additional questions that the auditor need to ask;
• In many cases the file will contain information that is not required for the audit at this stage of the CAAT it is a good idea to specify only those fields that are required as space will be saved on the PC and file processing may be quicker;
• It is possible that the data is not in an ideal form. For example the field may incorporate Account code and Cost Centre when only Account code is required. The file may also be missing information that would be helpful to the audit, for example Vote code;
• If information is not in the ideal format or key information is missing it may be necessary to liaise with the department or CAATs specialists to arrange some form of file manipulation such as an Embedded audit module or a file manipulation program to produce the data in a more useful format.

2.7 Specifying the format of the data file
• When the most appropriate file has been selected for CAATS interrogation and the information required from that file has been specified, it is necessary to agree a format for the data and a method of storing that data;
• The data should be provided in a format that can be readily accessed by the PC used by the auditor. The data needs to be stored/supplied on a media that can be read by the hardware used by the auditor;
• The data needs to be in the simplest form possible preferably a form that can be linked directly to the CAATs software being used;
• If the audit file cannot be provided in a readily useable form it may be necessary to manipulate the data prior to interrogation. If this is the case the data should be provided in a format specified by the CAATs specialist;
• If the client department is unable to provide data in a format readily useable by the audit department. It is advisable to ask for a report file. All computer systems are able to output some form of report. It is always possible to output the report to a magnetic media such as floppy disk. A report is fixed length in nature as it is bound by the size of the paper it is printed on. Today, many utilities exist that can manipulate report files to produce auditable tables.
It should be stressed by the auditor or the CAATs specialist that the client should not provide a backup of the system as unwanted data such as file formatting may be included with the data file.

### 2.8 File downloading and conversion
The CAATs specialist has an important role to play after the most appropriate data has been selected and provided in a format that is useable by the audit department.

- The CAATs specialist may:
  - **Download the data to the auditor’s PC.** This procedure could be as simple as providing a virus screening function before the data is copied by the auditor to his / her PC, to copying the data from a magnetic tape to a PC and manipulating the file into format that can be accessed by the audit department’s file interrogation software;
  - **Load the data into the auditor’s CAATs package.** The CAATs specialist may go as far as loading the data into the auditor’s file interrogation software;
  - **Use file manipulation utilities to convert the client’s data into a file compatible with the CAATs software.** If it was impossible for the client department to provide the data in a readily useable format. The CAATs specialist may need to perform some form of manipulation on the file to allow the data to be loaded into the file interrogation software.

  Such manipulations could cover:
  - Adding extra information to the file such as a Vote Code
  - Removing unwanted information or characters
  - Converting variable length data to fixed length
  - Translating the file form one format to another (Lotus to Excel)

- **Carry out tape and data management for the audit.** The CAATs specialist may keep records covering the serial numbers of the tapes received and the timing of any backups made of the client’s data. The CAATs specialist may be responsible for returning data to the client department when the interrogation is finished;

- **Initiate further technical liaison to further improve the CAAT or the layout of the audit file.** This will usually occur if the client department changes their accounting system. The CAAT will need to redeveloped. Liaisons will begin afresh determining the files available and the data types that can be used.

### 2.9 The Standard Requirements for data
These are the standard requests for data format and storage media as used by the NAO. Requirements will differ from audit body to audit body as the hardware and software used will probably be different:

- The data is fixed length text (variable length records are more difficult to link to file conversion software);
- The data is provided in a format compatible with the software used by the auditor (Windows files such as Lotus or Excel, Dbase, Comma Separated ASCII.......);
• If a client department is unable to produce data in one of the required formats it is always possible to print the data to a floppy disk instead of paper and use a report file utility to create a usable file;
• The data must be provided on a magnetic source that is compatible with the hardware used by the auditor. It is possible to download data using a modem or a PC to PC utility such as Laplink. If this is an option, it is advisable to discuss the implications of virus contamination with the IT security staff from both the client department and the audit department;
• In some instances it may be possible to set up a terminal in the audit department to access the client’s system. Data can then be extracted directly from the client body to the auditor’s PC;
• It is vital that the audited body provides a record layout with the data to be interrogated showing the fields used and the data types that exist. This information helps the CAATs specialist when downloading/converting data and linking the data to the interrogation software.

2.10 Modes of data storage / transfer
• Today there are many methods of data transfer available: Floppy Disk;
• Tape Cassette;
• Reel Tape;
• PC to PC link (Laplink);
• Across a Network;
• Via a Modem;
• Downloading via E-Mail / Internet;
• The method used to transfer data from the client department to the auditor’s PC will depend upon the hardware used by the audited body and the auditor;
• Clear specification must be made when requesting data;
• It may be necessary to consult the IT security department of both the audit department and audited body if data is being transferred from floppy disk, across a network or from PC to PC as there could be implications for data security and virus contamination.

2.11 Who or What can help with Data Identification / Acquisition?
• It is always advisable to involve both CAATs specialists and IT personnel from the audited body. In many cases the auditor does not have as complete a knowledge of the files to use or possible layouts;
• The CAATs specialist should be used as an intermediary between the IT personnel of the audited body and the auditor;
• If at all possible the auditor should consult previous system reviews or program flow charts to identify the best possible data. This consultation should take place with trained IT staff;
• In all cases the planning of a CAAT should take much longer than the actual development. A properly thought out approach using the correct files in the best possible format will minimize problems and limit the maintenance of the CAAT once it is developed.

3. What is data ?

3.1 Forms of data/communication
• There are many differing forms of communication. For instance something you see can convey a message as easily as something heard. For example a red traffic light is universally understood to mean stop. A police officer can give the same message with a word or a gesture;
• In more recent times the invention of the computer has allowed for a different form of information. Computers store information electronically. Ledgers on paper have been replaced in most cases with databases storing the same information in electronic form;
• To the question What is data ? ; a good explanation would be that data is information stored in a wide range of formats;
• Understanding data/information can be difficult. Not only is data represented in a variety of ways, these ways can be different from person to person, culture to culture and computer to computer. For example :
  * Language barriers;
  * Misunderstandings;
  * Gestures;
  * Differing computer operating systems and storage methods.

3.2 Electronic data formats
• Electronic data can be viewed/interpreted in a variety of ways :
  * Hexadecimal;
  * Binary;
  * ASCII;
  * EBCDIC;
  to mention but a few. The same problems arise when dealing with electronic data as with verbal or visual data. Most computer systems are not completely compatible, information may need to manipulated before it can travel from one system to another;
• To download files is to be an interpreter. Information in many different formats need to be understood and passed on in different forms.
3.3 Binary and Hexadecimal

- All data on a computer is stored in Binary (Base 2). There are however more than one way of interpreting binary in different number bases;
- One such number base is Hexadecimal (Base 16) 0-9, A-F. Hexadecimal is the commonest way of representing the binary information on a PC;
- All characters seen on a PC have a decimal value dependant upon the format in which they are being held. It would be impractical to represent these values in binary on the screen. Take the example “ABCD”. This in itself only takes up four characters and uses very little space on the screen. The PC holds this information in binary. To represent this information in binary would use a lot more space on the screen and would be complicated to understand;
- Each character has a decimal value A=65 B=66 ................... depending on the format used (in this case ASCII);
- in binary “ABCD” would be represented 10000011000011000011 10001;
- In Hexadecimal (base 16) the same characters would be represented as 41 42 43 44.

3.4 Bits and Bytes

- Each Binary number on the PC is called a Bit (Binary Digit) Four Bits combine to make a Nibble (One half of a Hexadecimal number) and two Nibbles combine to make a Byte(The full Hexadecimal code representing a character);
- All numbers and characters on a PC have a decimal value. This can also vary according to the system of coding used (ASCII, EBCDIC......). Using the example of the character A which has a decimal value of 65 in ASCII :
  * Binary representation = 0100001;
  * Hexadecimal = 41;
  * To calculate the numeric value in Hexadecimal use the following formula:
    ♦ Upper (leftmost) Hex nibble x 16 + Lower Hex Nibble;
    ♦ A = 65 Decimal = 41 Hex = 4x16 + 1;
- There are other forms of coding data such as EBCDIC which will be covered in more detail later. In essence the decimal value of characters are higher than those used in ASCII;
- Using other forms of coding such as EBCDIC characters have different decimal values. For example the character 0 is 30 Hex = 48 decimal using ASCII . Using EBCDIC 0 is stored as F0 Hex = 240 decimal.

3.5 The value of Hexadecimal

- As data on a PC is stored in byte format Hexadecimal is a very useful way to view data. As there are 8 Bits to a Byte the PC can use a value of 0 - 255 to store the decimal value of a character. Hexadecimal using base 16 will use the numbers in the range 0 - F;
- Again using the ASCII character A as an example. A has the decimal value 65;
  * In Binary this is represented as 0100 0001;
* In Hexadecimal 4 1;
  • Each side of the Hexadecimal number represents one half Byte or “Nibble”. Using this numbering convention is by far the easiest way of representing data;
  • When viewing data either from a tape or on the hard disk of a PC, the user will usually have the opportunity to use a Hex dump. A Hex dump shows the Hexadecimal values of the characters in the file you are viewing/editing;
  • When downloading data the use of a Hex dump to match data to record layout is usually the initial stage of the job. Matching a record layout to a Hex dump of the data enables the file downloading specialist to devise any record layouts that may be needed by the audit department’s file interrogation software and to plan any conversion software that needs to be written.

3.6 The Hex dump

• In the usual format of a Hex dump the screen is split into two halves. One half will show the Hexadecimal view of the data and the other will show the textual view of the same information.

There are many utilities available that present the user with a Hex dump of the data they wish to download / manipulate. Specific tape reading utilities such as Fdump or Tapeutil produce a Hex dump of data held on a magnetic tape. PC utilities such as Hexed or Qdos enable the user to view a Hex dump of files on their PC;
• A typical Hex dump will have the following characteristics:

<table>
<thead>
<tr>
<th>HEX TEXT</th>
<th>ASCII</th>
<th>EBCDIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>41 42 43 44 45 46 31 32 33</td>
<td>ABCDEFG123</td>
<td>c1 c2 c3 c4 c5 c6 f1 f2 f3 ABCDEFG123</td>
</tr>
</tbody>
</table>

• These two example show data in ASCII and EBCDIC formats. As mentioned earlier communication can be difficult when two different languages are being used. This is a good example of that. Both ASCII and EBCDIC use decimal values to represent all numbers and characters. The Values are however different;
  • The character A has a Hex value of 41 in ASCII = Decimal 65;
  • The same character has a Hex value of c1 in EBCDIC = Decimal 193;
• Two of the commonest Hex patterns to look for when downloading data are the Carriage Return and Line Feed characters;
• On a typewriter these are mechanical actions performed when a new line is to be started. These mechanical actions are stored as characters on the PC. Carriage Return Line Feed (CRLF) usually represent the end of one record and the beginning of another. In fixed length data the HEX pattern 0D 0A , decimal 13 10 should occur at regular intervals.

WHEN VIEWING RECORDS IN TEXT FORM THE CRLF CANNOT BE SEEN AS CHARACTERS. ALWAYS ADD 2 TO THE LENGTH OF THE RECORD. SOME SYSTEMS SUCH AS UNIX WILL HAVE A LINE FEED ONLY AT THE END OF A RECORD. IN THIS CASE 1 WOULD NEED TO BE ADDED TO THE RECORD LENGTH.
3.7 Why use Hexadecimal

- As the PC stores and receives all of its input in Binary (Base 2) it would be very confusing to display all text in its binary form. Hex is a simpler way of storing and viewing the decimal values of characters being in Base 16;
- Using the ASCII number 1:
  Decimal value = 49
  Hex value = 31 (16 x 3 + 1)
  Binary value = 110001 (32 + 16 + 1)
  Each Hex Nibble is represented in Binary and is 4 Bits long
  1 2 3 4 1 2 3 4
  3 = 0 0 1 1 1 = 0 0 0 1
  When recombined as an 8-bit Byte = 00110001 = 32 +16 + 1 = 49

4. Character Representation

4.1 ASCII and EBCDIC

- There are two ways of representing data in its decimal form, ASCII and EBCDIC. Each form uses a different series of decimal values to represent numbers, letters and control characters;
- The letter A is represented by decimal 65 in ASCII and decimal 193 in EBCDIC;

<table>
<thead>
<tr>
<th>ASCII</th>
<th>EBCDIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>65</td>
<td>193</td>
</tr>
</tbody>
</table>

- When downloading data a good rule of thumb is to expect Packed Decimal and Binary values when Using EBCDIC and decimal values in ASCII. EBCDIC is a Binary form of data representation and will use more forms of space saving in its numeric characters;
- Most Mainframe sites tend to produce data in EBCDIC format as there are many space saving numeric data types to use:
  * Signed Binary;
  * Packed Decimal;
  * Floating Point;
- These data types are not a form of EBCDIC storage but are likely to appear as they are represented in Binary environments.
4.2 EBCDIC data

- EBCDIC (Extended Binary Coded Decimal Interchange Code) is a binary representation of data. While EBCDIC can hold ordinary numeric data, some of the more complex data types can exist:
  - Binary data types
  - Packed Decimal
  - Signed and unsigned Binary
  - Float Forms

- These data types are used to store information in a more efficient way. EBCDIC data will be more commonly seen on tapes from mainframe sites;
- Numeric fields such as Date, Amount, Number of ....... etc. may well be stored in a format other than ordinary numeric;
- EBCDIC uses a different range of decimal values to represent characters, numbers and control characters. This becomes obvious when viewing Hex dumps. For instance numeric data begins at decimal 261;
- A numeric string in EBCDIC would appear thus in a Hexadecimal display:
  \[
  \text{F1 F2 F3 F4 F5} = 1 2 3 4 5 \\
  \text{F1} = 1 = 15 \times 16 + 1 = 261 \\
  \]

- If the data from a client department is in EBCDIC format it is recommended not to convert the data to ASCII. As it is common to see numeric data stored in other than pure decimal form in order to save space on the file, any conversion will destroy the numeric data.

4.3 ASCII data

- ASCII (American Standard Code for Information Interchange) data will usually appear in decimal character form only. It is unusual to see any numeric data in anything other than decimal form. ASCII data is easy to read normally. All information seen on a PC screen is in ASCII data;
- ASCII uses a different set of decimal values than EBCDIC to represent data;
- Consider a numeric string in Hexadecimal. Numeric characters begin at decimal 49:
  \[
  \text{31 32 33 34 35} = 1 2 3 4 5 \\
  \text{31} = 1 = 16 \times 3 + 1 = 49 \\
  \]

- This form of data representation of numeric characters in both ASCII and EBCDIC is known as “Zoned Decimal”. This is the commonest form of data storage. No attempt has been made to save any space. All of the byte is used. It is possible to save space when storing data by using the right hand 4 bits (lower Hex nibble) to store the sign of the numeric data;
- When requesting data from a client department for a data downloading exercise it is always preferable to request ASCII data wherever possible as all of the information can be read and understood when browsing the data with a tape reading utility. This enables the user to match record layouts to the data quickly and easily. As no binary space saving is going on there is no danger of destroying data if a conversion takes place.
4.4 Zoned Decimal

• There are more forms of Zoned decimal data representation. The sign of the numeric data can be “embedded” in either the first or the last digit of the number. This is a form of space saving common to EBCDIC data. It is possible to see this form of data in an ASCII file if the data has been converted from EBCDIC. This form of EBCDIC numbering is the only form that can safely be converted to ASCII;

• In ASCII sign embedded numeric data could look like this:

31 32 33 41.

• When translated to characters on the screen the following would appear:

1 2 3 A

• This is a very strange looking number. The characters {, A - I represent a positive sign and the numbers 0, 1 - 9 and the Characters }, J - R a negative sign and the characters 0, 1 - 9;

14

As can be seen one Byte of information can contain both sign and digit;

• Zoned decimal can also appear in EBCDIC data. The same convention is used but, as explained earlier, using a different set of decimal values;

• The following numbers would appear in this way in EBCDIC:

112.61 = F1 F1 F2 F6 C1

-112.68 = F1 F1 F2 F6 D8

• IN BOTH ASCII AND EBCDIC THE DECIMAL POINT IS IMPLIED. When a decimal point is implied it is not included in the number and has to be added by the operator. Usually by dividing the number by 100.

4.5 Signing Zoned Decimal

• In both ASCII and EBCDIC the second or lower Hex nibble is always represented in Binary. It is only the first or Upper HexNibble that is different;

• To assigning a sign to zoned decimal. No extra space is needed as the first half of the byte is always used simply to store a numeric value that decides whether the character is an ASCII or EBCDIC representation;

• This second half byte can be overwritten with the sign;

• In both ASCII and EBCDIC the second half of the Byte is always binary. The first half merely governs the format in which the data is coded;

31 32 36 41 = 112.6A (ASCII)

F1 F1 F6 F2 C4 = 116.2D = 116.24 (EBCDIC)

F1 F1 F6 F2 D4 = 116.2M = -116.24

• Many of the popular file interrogation packages now recognise this form of numeric data. In the past a routine would need to be written to translate the last byte of information into a meaningful number. This form of numeric representation is known as Numeric Sign Embedded by file interrogation software such as Idea and ACL.
4.6 Packed Decimal

- Packed decimal is a common form of numeric storage. Either the sign or the numeric digit itself are stored in the a single half Byte (Nibble). This form of “packing” enables numeric data to occupy less space than would otherwise be the case if simple Zoning was being used. A 15 digit number would occupy 15 bytes in Zoned decimal, but would only occupy eight bytes in Packed decimal;
- Again as with Zoned decimal, the decimal point is always implied. There are two types of packed decimal data; Signed and Unsigned;
- In signed Packed Decimal the sign is stored in the Lower Hex Nibble of the last Byte;
- There are three packed decimal signs:
  * Positive = C;
  * Negative = D;
  * Unsigned = F;
- In Unsigned Packed Decimal no sign appears;
- Examples of Packed Decimal data:
  00 00 01 33 44 55 6F = 133445.56 - Unsigned
  00 00 01 33 44 55 6C = 133445.56 - Positive
  00 00 01 33 44 55 6D = -133445.56 - Negative
- In unsigned Packed decimal the same number would appear thus:
  00 00 13 34 45 56

4.7 Other forms of Packed Decimal data

- There are more than one form of packed decimal data:
  * Borroughs Packed:
  - Very similar to standard packed decimal except the sign appears in the fist half byte. The number 133445.56 used in the previous examples would appear thus:
    00 00 0D 13 34 45 56 = -133445.56 - Negative
  * Packed Nibble:
  - Sometimes numeric data does not start on the boundary of a byte. Packed Nibble data appears in the same format as unsigned Packed data. For example
    4 3 4 5 6 7 8 9 = 34567.89
  IN BOTH OF THESE PACKED DECIMAL DATA FORMS THE DECIMAL POINT IS IMPLIED;
- When viewing packed decimal data in a HEX print the number of bytes taken up by the number itself may be shorter than the number quoted on a record layout. For example the department providing data for downloading may say that the amount filed is ten characters long. If this data was represented in Zoned Decimal the number would be then Bytes long. In unsigned packed decimal it could be as little as five bytes long;
- Consider the number 12345678.90;
- In ASCII Zoned Decimal the number would be represented 31 32 33 34 35 36 37 38 39 30;
- In EBCDIC Zoned Decimal F1 F2 F3 F4 F5 F6 F7 F8 F9 F0;
- In Unsigned Packed Decimal 12 34 56 78 90.
4.8 Binary Data

- With Binary data types. Both half bytes hold binary values. In a single byte a numeric value in the range 0 to 2 to the power of 8 can be stored:
  * 0 - 255;
  * A 2 byte Binary number can store a value of 0 to 2 to the power of 16 or;
  * 0 - 65,535;
- Binary numbers can go up to 8 Bytes, although this can vary depending on the machine used;
- Binary numbers can be signed. The sign is stored in the first bit of the number:
  - Positive = 0;
  - Negative = 1;
- A one byte signed Binary number can be in the range -128 to +127;
- As with Packed decimal data a binary numeric field will be much smaller than the length quoted on a record layout. A fifteen digit number will only occupy eight bytes in binary;
- When calculating a number in Binary on a Hexadecimal display it is necessary to manipulate the Hex first (reverse). Each half Byte of the is then multiplied by increasing powers of 16 and added together;
- Take the following HEX representations of Binary numbers:
  * C3;
  * 1. Reverse the HEX = 3C;
  * 2. 3 x 1 = 3;
  * 3. 12 x 16 = 192;
  * 4. 3+192 = 195;
  * EC03;
  * 1. Reverse HEX 30 6E;
  * 2. 3 x 1 = 3;
  * 3. 0 x 16 = 0;
  * 4. 6 x 256 = 1536;
  * 5. 14 x 4096 = 57344;
  * 6. 3+0+1536+57344 = 58883.

4.9 Floating Point data

- Not so commonly encountered but still appearing in EBCDIC files is the floating point number;
- Floating point numbers can vary according to the computer system being used. It is very difficult to convert them to Hex. A Floating point number has 3 distinct elements:
  - The sign bit - Positive or Negative
  - The Mantissa - The number as an Integer
  - The Exponent - The power by which the fraction is to be multiplied
5. Record and File layouts

5.1 The Record Layout

• When downloading data it is vital to have a record layout from the department sending the data. A record layout enables the file downloading expert to understand the nature of the records within the file and the elements that make up each record. This information is used for any software that will manipulate the downloaded data;
• A data file on the PC is made up of several elements in a hierarchical structure:

DATA FILE
RECORDS
FIELDS
CHARACTERS
BYTES

• A record layout will give the user a detailed picture of the fields that form each record including information about their length, data type, decimal places........;
• When viewing a record layout the user must be aware that some numeric fields will not be the same length on the file as displayed in the record layout. For example a 15 character numeric field stored in binary or packed decimal format will not use 15 bytes (probably 8) as binary and packed decimal are methods of saving space used by mainframe storage;
• It is a good discipline to regard a record layout as a guide to file downloading rather than a definitive picture of the data record;
• Consider the following basic record layout forms:
  * 99 = Numeric data length 2
  * 9(2) = Numeric data length 2
  * 9(7) v99 = Numeric data length 2 , 2 decimal places
  * s9(11) v99 = Signed Numeric data length 11 2 decimal places
  * (comp) = Binary
  * (comp-3) = Packed Decimal
• Textual data is represented in this way:
  * xx = Textual data length 2
  * x(25) = Textual data length 25
• This is a very basic record layout.
• A record layout shows the fields that exist in a particular file. All record layouts should contain the following elements:
  * Field Name Data Type length Decimal places
• The above example can be misleading as it is not evident how long each field is when the data is in its raw state. Consider the amount field. The fully converted number could well be 15 characters long. The data in its raw form will only occupy 8 bytes as it is Packed;
• Record layouts also vary in format for example:
01 RECORD
02 DEPARTMENT
03 AREA_CODE
03 DEPT_CODE

- In this case the record layout is in a hierarchical format. Only the type 3 fields need be included.
A flipchart demonstration may help clarify this layout;
Field Type length Start Position
AREA_CODE CHAR 4 1
DEPT_CODE PACKED 5 3

In all cases it is advisable to use a hex dump to check the record layout against the raw data.

5.2 HEX dump utilities

- There are several utilities that can be used to obtain a Hex dump of your audit data whether it is on tape or on your hard disk:
  * Norton Utilities - This software enables you to look at data on the hard disk of a PC or on a floppy disk. It is possible to directly edit a hex dump of your data.
  * Quick DOS - This is a file management utility. It is possible to view files in Hexadecimal format on the hard disk of a PC or on a floppy disk. It is not possible to edit Hex directly with this software.
  * Hexed (FRED) - This software enables direct editing of a Hex representation of your data on hard disk or floppy disk.
  * Fdump - A tape utility that enables the user to view a Hex dump of data on a magnetic tape.

5.3 The typical HEX dump

- This is a representation of a typical Hex dump. The Fdump tape reading utility will produce this type of display:
  - A Hex dump is split into 2 distinct parts. The left hand side of the display shows the Hexadecimal view of the data. There are 16 Bytes to a line. The right hand side of the display shows the textual representation of the data;
  - A record layout provided by the audited body can be used at this point to “Count the bytes” on the left hand side and match them to the text on the right hand side;
  - Data files consist of RECORDS. A Record can usually be termed as one transaction. For instance an accounts file will contain details of financial transactions. Each posted transaction will contain information about the payment. Paid Date, Invoice No., Amount, Ledger Code........;
  - Each record contains many elements. Each element is termed as a field. A field is an information unit within a record;
  - Each field consists of characters (Bytes) which can be represented in many ways. Binary,
Packed, float ..........;

- A record layout will consist of a record broken down into a number of fields. Each field will have a data type and length.

A RECORD LAYOUT SHOWS DATA AT A RECORD LEVEL ONLY

5.4 The typical record layout

- When faced with the prospect of downloading a data file, it is very important to have a record layout from the audited body.
- These fields are specific elements of a record:
  
  Name x (20)
  House No. 9 (3)
  Street x (30)
  Town x (15)
  Post code x (8)

- The example used is a very basic record layout using simple text and numeric characters. It is therefore very easy to calculate the length of the Address record:
  
  - It is possible that some address records do not have a street name of 30 characters or a Town name of 15 characters. For this example, it is best to assume a “FIXED LENGTH” record, i.e., all fields are the same length regardless of their content;
  - When using a Hex dump of this data, each field can be counted off to a total of 91 characters. It is possible that characters 92 & 93 may be a carriage return line feed (Hex 0D 0A) or that the next record begins at character 92. In either way, a pattern should be easily recognised in the Hex dump;
  - When downloading a data file, it is very important to use the information provided in the record layout. It is possible to calculate the length of the record by adding together the lengths of the individual fields (fixed length only).

5.5 Calculating the No. of records on the file

- By using a tape scanning utility such as Tapeutl or Looktape, details of the size of the data file can be obtained;
- When the length of the record is known and the size of the data files and blocks are known, a simple formula can be applied to calculate the number of records in a particular data file;
  
  \[((\text{Data block size} / \text{Record length}) \times \text{No of blocks}) - 1\) + (size of last data block/Record length);
• It is important to calculate the number of records on a data file. One of the objectives of an audit is to prove the completeness of an account. It is important to prove that all of the financial transactions in a file have been interrogated. Calculating the number of records on a data tape enables the file downloading specialist to prove the completeness of the tape. The number of records calculated for the file on the tape can be compared with information provided by the client department and checked against any conversion programs that are performed;
• When performing a file downloading exercise for a CAAT, the details of the calculations of the number of records on the data files throughout the financial year are kept as part of the operational documentation of the CAAT itself. If any problems occur when the data is interrogated, the CAATs / downloading specialist can quickly refer back to the download and confirm that the complete file was downloaded.

5.6 Types of tape layout
• These are the results obtained when scanning a file. The utility used in this case was Looktape (Looktape is a utility written in-house by the NAO). Most tape scanning utilities will give very similar output.
• The tape scanned is a typical ICL format tape;
• Each ICL tape has the following:
  * 3 Blocks of data showing header information (Volume no. File name etc.);
  * X Blocks of transaction data (Audit Records);
  * 2 Blocks of trailer information.
• The last block of audit information is shorter than the other blocks of audit data. This is because the number of records on tape does not divide exactly into the block size chosen. The remaining records are written into the last block;
• On some occasions the last block will be space filled to match the length of the other audit data blocks. Record calculation can then only be approximate as an unknown number of records are in the last block.

5.7 Calculation from a standard ISO tape
• This is the calculation of the number of records in the address file displayed earlier
Block size = 9100 bytes = 100 records per block
99 x 100 = 9900 = 182/91 = 2 = 9902

You have scanned a tape using Looktape and have the following information.
File 0
3 blocks read
Block size - maximum 80 bytes minimum 80 bytes
**********

File 1
100 blocks read
Block size - maximum 9100 bytes minimum 182 bytes
**********

File 2
2 blocks read
Block size - maximum 80 bytes minimum 80 bytes
• If the last block had been space filled the calculation would have equalled 9999 records on the file;
• Data files are divided into segments called data blocks a data block is decided by different factors:
  * A multiple of records per block (usually a factor of 10);
  * The default block size written by the audited department’s machine;
• Problems are sometimes encountered when faced with default block sizes as data records can begin in one block and continue in another (block spanning). If this is the case the formula to calculate record number is:
  * (no of blocks x block size) / Record length;
• Audited bodies will usually provide information as to the number of records on the data file.
The calculation of record number is a good way to ensure that the data tape is complete;
• It is a good discipline to carry out as many checks as possible on the data file that you have.
One of the audit disciplines is to ensure completeness of the file. If the audited body can provide the number of transactions on a file and their value it is a good idea to prove this information is correct before interrogating the data.

6. Types of data records

6.1 Fixed length data
• Records that are fixed length will always occupy the same space on a file. There are more than one type of record that can be defined a “Fixed Length”:
  * All records have the same fields filled to equal length;
  * All records have the same fields but empty and short fields are space filled;
  * records have varying data but end with a terminating character in the same place.
• Records that have the same number of fields but each field varies in length can be supplied with a unique character between the end of one field and the start of another “delimiter”. This type of record can be treated as fixed length by many packages, as it is possible to set a default field length equal to the longest field. Fields are then read up to the delimiter;
• When requesting data for file downloading fixed length data should be the preferred choice. As each record is the same length, all of the fields will appear in the same place on each record. This form of record layout makes it easy to apply a program to manipulate the data into a format acceptable by the audit body’s file interrogation software. Any changes that may subsequently occur to the record will easily be picked up.

6.2 Advantages of fixed length data
• Fixed length records have advantages when downloading data:
  * It is easy to calculate the number of records on a file if the length of the records is constant;
  * If files need some manipulation prior to interrogation. The programs needed are usually very simple. For instance a program that inserts Carriage Return Line Feed characters at the end of every record is much easier to write when the length of the records remains constant;
  * Most fixed length records are easier to import directly into interrogation software such as Idea when the record length is constant;
  * Generally fixed length records do not need very much manipulation, so less file processing is necessary.

6.3 Disadvantages of fixed length data
• Fixed length records also have disadvantages:
  * It is possible that a file of fixed length records has been created from a file of much longer variable length records with a fixed section and a variable number of fixed length elements. In a case such as this the fixed length file will have a great deal more records than its variable length counterpart. File processing could well take much longer due to the sheer volume of data.
  * It is also possible that fields that do not hold any data will be space filled thus occupying more disk space.

6.4 Variable length data
• There are different forms of variable length records. The commonest form is a record with a fixed length block and a varying number of fixed length elements. For example a payroll file;
  • The fixed length block holds the individuals details and each fixed length element is a payment detail. Gross Pay, Deductions, Overtime........ As each individual’s pay can have different elements there may be a varying number of pay blocks;
  • In order to save space records with a fixed number of fields can be written with no space filling in the fields. For example, consider a file with the following fields:
    NAME INITIALS DATE AMOUNT
• Records would be stored in this way:
  * SmithAB0101199512345
  * AbbottSBA1011199423456
• A delimited record has a unique character between each field for example:
  * Smith,AB,01011995,12345
• This record type is variable in length but some interrogation packages can use the length of the longest field as a default. Say the longest field is 20 Bytes:
  * Smith AB 01011995 12345

### 6.5 Disadvantages of Variable length data

• Variable length records have disadvantages for file downloading:
• Most analysis packages need a record layout in order to recognise the data to be interrogated and are expecting records of the same length. Variable length records do no usually import directly into interrogation software without some form of manipulation first;
• When manipulating variable length records, the programs needed are generally more complex than those required for fixed length data as the start and end positions of individual records within the data file are unknown;
• If a record layout changes it is more difficult to accommodate the changes for variable length records. As the change could occur more than once on the same record;
• It is more difficult to match variable length records to a record layout when using a Hex dump;
• The use of variable length records is very common in many Mainframe sites. It can sometimes be difficult for them to produce data in a fixed length format when requested, and a programming cost may be incurred.

### 6.6 Advantages of variable length data

• Variable length records do have advantages:
  * Generally there will be less records to process than on a corresponding fixed length file. Variable length records are a more efficient use of storage space. A variable length record format will often be the default layout on the computer system of the audited body.

### 6.7 Report Files

• A report file is a useful file type to use in the data downloading process. Quite often a department is unable to provide a file in a format that can be downloaded easily. The audited body may be reluctant to provide a record layout;
• Using a report file a record layout is no longer necessary as the field titles are generally found within the report itself;
• Report file are fixed length as they are bound by the page width (80 or 132 in most cases);
There are many software tools available to manipulate report files. For instance a basic report file definition tool exists in the Idea5 file interrogation package. There are “Off the Shelf” packages available dedicated to the manipulation of report files:

* Auto Import
* Monarch

With both of these packages particular elements of a report file can be highlighted and exported in a fixed length format to a new file that is easily imported into an interrogation package.

7. Data Transfer and Manipulation

7.1 Methods of data transfer

There are many ways in which data can be transferred from the computer of the audited body to the computer of the auditor:

- Small files can be copied directly from floppy disk to the hard disk of the auditor’s PC. Larger field may have been compressed using a utility such as PKZIP and will need to be decompressed during the transfer;
- Data can be transferred via a telephone link to the audited department (modem). The transfer rate is not as fast as copying data (@2mb per minute), but can be quite acceptable for smaller files;
- Data can be downloaded from a magnetic tape. The transfer rate is generally very fast. If the data is to be downloaded from a reel tape most tape readers will be able to access the data. If however, data is to be downloaded from a data tape or other form of cassette problems of tape formats may occur;
- Data can be restored from a floppy or a backup cassette tape. Problems can arise if the auditor does not have the same equipment and backup software as the audited department;
- Data can be transferred from PC to PC without the use of tapes or disks. Software is now available that enables data to be sent via a parallel or serial cable:
  * Laplink;
  * Brooklyn Bridge;
- It is also possible to send data across a network although the transfer rate can be quite slow.

7.2 Data Transfer in the Windows Environment

With the advent of Windows and Win 95 it has become easier to obtain data in a form that can be used directly for file interrogation. Most of the applications that run in a the Windows environment are able to import and export data in a wide variety of formats. It is usually the case that one of the formats used will be compatible with the file interrogation software you wish to use;
Such a method of transfer would generally be used for smaller files although it is possible to work on file of more than 100mb if database applications are being used;

An example of such a transfer method could work as follows:

The auditor requires interrogation on an accounting file that is held in an Excel database. The software to be used for the interrogation can access a Lotus 123 file. The data downloading specialist can easily save the Excel spreadsheet in Lotus format and import the data directly into the file interrogation package;

For smaller files it is possible to “Cut and Paste” data from one application to another or dynamically link the data to another application using OLE (Object Linking Environment). For example it is possible to update an Excel spreadsheet from within a Word document;

Windows 3.x and Win95 make use of many software drivers. One set of drivers are the ODBC (On Line Database Connectivity) drivers. It is now possible to “point” the file interrogation software to be used for CAATS to a particular database or spreadsheet and access the file directly.

7.3 On Line Database Connectivity (ODBC)

Both Windows and Win 95 are equipped with ODBC drivers. Some CAATS packages are able to access windows file by making use of these drivers. The CAATS user can specify the particular source of data for interrogation from within the file interrogation software and invoke the relevant ODBC driver for that source. In this way direct access between the file interrogation software and the data source can be maintained.

7.4 Data Manipulation

In some instances data needs to be manipulated after a download in order to put it into a format that can easily be imported into an interrogation package:

Some tapes that are written using a default block size may have padding characters at the end of each block to fill the remaining space. For example a block of 2048 characters may hold 20 records of 100 Bytes. The remaining 48 Bytes will be padding. A simple program will need to be run against the downloaded data to remove the padding;

Vme tapes have 8 Bytes at the front of each data block, 4 Bytes at the front of each record and block padding characters. All need to be removed to leave the records. VME is a standard method of file storage used by IBM mainframes. It is a good idea to write a standard VME unblocking routine to cater for all VME tapes;
• If data is not in fixed length format a program may be required to add a terminating character or to pad out fields to their full length. It should usually be possible however, to specify that the data should be fixed length during the liaison with the client department. ;
• It may be necessary to remove control characters such as page breaks from a report file. This form of file manipulation is now becoming less common as software now exists to manipulate report files;
• Sometimes a data file needs some additional information to improve the audit trail. Or the file is hierarchical and Account information needs to be added to the records;
• Some files have more than one record type requiring the removal of one particular type of record from the file.

7.5 Methods of data manipulation
• There are several ways to manipulate data:
• Data can be manipulated using a simple editor such as:
  * Hexed;
  * Norton Utilities;
  * DOS Text Editor;
• Most of the file downloading utilities have the option to convert data during the downloading process. Usually from EBCDIC to ASCII or vice versa. Only use this option if you are dealing with standard text or numeric data. A conversion from EBCDIC to ASCII for instance will destroy Packed and Binary numeric data;
• It is common to write file manipulation software in house to “Fix” downloaded data. This could take the form of removing padding characters from downloaded data blocks or splitting a downloaded file into smaller sub files containing different record types;
• There are a range of Off the shelf software to manipulate downloaded data. Each performing specific functions. For example, print files can be manipulated using Auto Import or Monarch. Software Bridge enables the user to translate files from one format to another;
• It is also common to import data into one package just to export it in a format that can be accessed by your interrogation software.

7.6 Editing
• Editing is not strictly advisable when dealing with audit data as the whole purpose of the audit is to analyse intact data. Editing can be used however to remove stray characters. For example Oracle will insert a Line Feed character at interval of 50,000 records. Hexed can be used to remove this phenomena.

7.7 Small Utilities
• Small utilities can be used (usually written in house) to:
  * Add Carriage Return Line Feed characters;
  * Remove commas from a file;
* Swap one Character for another;
* Data conversion utilities during file download are usually available with most tape handling utilities. The usual conversion is from ASCII to EBCDIC and vice versa;
* EBCDIC to ASCII will destroy Packed Decimal and Binary data;
* ASCII and EBCDIC use a different decimal value for characters. If you are converting ordinary text or Zoned Decimal from one to the other no problems will occur. If however an attempt is made to translate Binary or Packed numbers to ASCII, the Binary will be moved to a lower decimal value probably representing something other than a number.

7.8 “In house” file manipulation routines
* There are many uses of In House file manipulation utilities to :
  * Create fixed length records from variable length records;
  * Remove block and record headings from VME files;
  * Remove headings and control characters from print files;
  * Restructure columnar print file to a more conventional row format;
  * Separate different file types from a data file.

7.9 “Off the shelf” software
* There are now many “Off the Shelf” packages available that can do the work of the “In House” utilities. Print files can now be manipulated by preparatory software such as :
  * Auto Import;
  * Monarch;
* Both of these packages enable the user to highlight specific patterns in a print file (usually transaction lines) and write the information to another file ignoring the remaining data;
* It has been possible for quite some time to import data file into one package and export them in a different format more compatible with the file interrogation software to be used. Software now exists specifically for this purpose:
  * Brooklyn Bridge;
  * Software Bridge;
  * Link and Load;
* With the advent of Windows and Win95 the need for such software is declining due to the many types of file formats that can be read into and written from Windows based software.

7.10 Types of database structure
* Data files can be ordered in different ways :
  * The most common form encountered in file downloading today is the sequential flat file. Data records are in no particular order. This type of file is encountered on all
magnetic tapes used for data downloading. When the records were written to tape they lost any indexing that may have existed;
• Indexed file. This is a disk file. Data records have a key to determine their order in the file. An indexed file is faster to process than a sequential file;
• A random file is a file type used at many Mainframe sites. A #total (checksum) of the key enables the computer to locate the position of a record very quickly;
• This following are some of the more common “Database Management Systems”. A database is essentially a data file or a grouping of data files:
  • A flat file is a series of data records in the same format (same fields). Data downloaded from magnetic tapes is usually in flat file format. All records in a flat file bear a relationship to each other, for example all may be transaction records. The file however has no relationship to any other file. A flat file may have been one of a series of related files at the audited department. When it is provided on tape the relationship is lost. It may be necessary to manipulate the file to recreate the original relationship;
  • A relational database consists of a series of files with a common link (tables essentially). For example a file of creditors details can link to a file of invoices by supplier name or a particular reference code. When amending data to one table of a relational database, it is possible to amend the same information on any other linked table;
  • A hierarchical database has a pyramid like structure with records of different levels feeding into each other. For example.
    * Vote - 100 overall summary records;
    * Account - 5000 summary records (50 per vote);
    * Sub Account - 50000 summary records (10 per Account);
    * Transaction - 1000000 payment details (200 per Sub Account);
    * Manipulation may be needed to add Vote details to each transaction.

8. Data simplification and the use of downloading hardware

8.1 Essential questions to ask before beginning a file download
• There are essential questions an auditor must ask when developing a new CAAT (Computer Assisted Audit Technique) or simplifying an existing data download and interrogation routine:

  * **What data?** - A question the auditor must ask. What information will give me the best assurances of the audited body’s financial position;
  * **What files?** - A dump of the general ledger may be adequate for the auditor’s needs. It is essential however to look at all of the files produced by the audited body as other files may give more complete information in a more readily accessible format. For example
the General ledger may contain summarised data whereas a transaction feeder file contains each individual transaction;

* What Format? - Can the data be provided in fixed length ASCII, EBCDIC, a print file..... What media can the data be provided on ? Tape, Floppy or can the file be downloaded via modem Etc;

* Is all the information needed? - What fields are needed to give the auditor the information required. Maybe some fields are not necessary. File interrogation takes up a great deal of hard disk space. Using essential fields only saves on space;

* Downloading method - Tape, Floppy, Network,Modem........ select one that you can use.

**8.2 The standard data request**

- There are a series of standard requests given to audited bodies when the file to be used has been identified. These requirements will vary depending on the downloading equipment you use.
- The standard request used by the NAO is as follows :-
- Data
  - In order of preference:
    - Fixed length ASCII;
    - Fixed Length EBCDIC;
    - Dbase or Windows compatible file;
    - Comma separated (Delimited);
    - Variable length records;
- Storage
  - 9-Track half inch magnetic tape
  - Floppy Disk
  - Tape cassette

A RECORD LAYOUT IS ESSENTIAL FOR A SUCCESSFUL DATA DOWNLOAD.

**8.3 What if the department cannot meet the standard data request?**

- If the department cannot meet your request all is not lost. It may be possible to obtain a printout of the file in magnetic form;
- If the department does not have a data storage system that is compatible to yours, it may be possible to visit the audited body and download the data directly from their PC using a utility such as Laplink;
- If all avenues have been explored and no suitable transfer method has been found, it is possible to use an outside body to translate the data on your behalf and provide it in a format that can be accessed by your PC.
8.4 The 9-Track tape reader

- This is the internal layout of a typical 9-track tape reader.
  Reading head
  Vacuum

- The whole unit is sealed. A vacuum is used to draw the tape from the spool. A secondary spool picks up the end of the tape and it is wound via the guide wheels until the beginning of tape marker is detected by the ceramic read head;
- The purpose of the guide wheels is to keep the tape taught at all times.

8.5 The tape reader interface card

- Most tape readers are connected to a PC via an interface card. The card is inserted into the PC.
  It will hold the chips needed to run the tape software and will have a connecting port for the relevant cabling;
- The tape controller card will come with a set of “Software Tape Utilities” such as tape reading, tape scanning, tape writing, data conversion......

8.6 Tape drive maintenance

- It is essential to perform basic maintenance on your tape drive:
- It is advisable to set up a maintenance contract (usually with the supplier) for your tape drive. The contract should cover the replacement of faulty parts and a swift call out response for major problems;
- Regular routine maintenance can be performed on site:
- Tape readers use ceramic read heads. Oxide from magnetic tapes quickly builds up on the read head and inside the machine itself. To clean read heads it is essential to use an air blower (aerosol of compressed air) to remove the majority of the tape oxide;
- The read head should be cleaned with a non alcohol base cleaning fluid and a swab. Alcohol based cleaners will damage the read head.

**Topic Two Sub Heading**