Data Bases

Recently I received a request to do an article on databases. It is always difficult to assess the general level of knowledge on such matters and as a consequence I have decided to start with a basic look at the differences in databases and then develop the art in subsequent articles.

Databases have fundamental building blocks:-

Bit – which is a binary "0" or "1", and as such conveys very little significant information.
Byte – which is a string of 8 Bits and is used to represent a character or letter.

Field – This is sometimes called an Attribute and is the smallest item that contains meaningful data; such as a Name, Address or a Quantity.

Record – this is a group of related fields; for example, all the information about a customer.

File – this is a collection of related records or data; for example all the information about all the customers interested in a specific product line.

Database – this is a group of related files; for example all the information about all customers interested in all product lines.

Organisation of Files

Sequential Files – The records in these files are stored sequentially, i.e. one after another and are sorted in an order such as alphabetically based on some key such as first letters of LastName.

These files are very useful when items require processing in order, or are being backed-up, but have the great disadvantage that each record must be read in order and there can be no jumping around within the file. It is very much like an audio cassette – you can fast forward, but you must go through each track in sequence.

Indexed File – This file has an Index that consists of a list of the values of one or more fields and the corresponding disk location for each record in the file, and using more than one index means that the records can be accessed much more quickly. In an Indexed File the records can be accessed either sequentially or directly, very much in the manner of a CD. The great disadvantage of this style of database is that as the files becomes larger access and record updating times can become longer and longer.

Direct File organisation – This uses a record's key value to determine its position on disk; the location where record are kept are called buckets, and these can have a number of

 Registered in England No: 06466302

slots to hold more than one record. This greatly reduces access and maintenance times.

Types of Databases

Hierarchical Database – These look very much like the old company organisation chart; so that each parent record can have many children, but each child record can only have one parent. The organisation splits in a tree and branch scheme from the top root record. This means that access to records is very quick, but records in different branches cannot easily be accessed at the same time. Also modifying or updating the database requires re-defining the whole database.

Networked Databases – These are similar to the hierarchical ones but in this case each child can have more than one parent record. In company organisation this is known as matrix working. Each parent record is now called an owner and each child record a member. This means that the speed of access is enhanced, but modifying or updating the database still requires re-defining the whole database

Query Language Databases – A Query language is a simple "English-like" language that allows you to specify the data you wish to see. This can be Query-by-Example (QBE); you are given a list of the fields available and you create an example of the data you require, or Structured-Query-Language (SQL) which is one of the most widely used languages with very definite structures and formats.

Object-Oriented Databases – When databases become very large, such as in Computer-Aided Design projects, hospital administration or aircraft manufacture, requiring the tracking of millions of different parts, the limitations of the relational databases become apparent. OOD are too complex to describe in detail here, but it is often considered that they represent data in a model that more closely aligns with the way we as humans understand complex structures of highly related information.

XML – With the advent of the World Wide Web we have all become used to seeing the HTML language. {If you have not seen such scripts click on "View, page source" on your internet browser and you will see lots of lines like: <paragraph> *text* <paragraph>. This is the HyperTextMarkupLanguage.} XML is very similar and allows tags (things in <> brackets) and attributes (i.e. name="value") to be used. This is very human friendly (after a time you can easily read the "raw data" but is very expensive on disk space – but the cost of storage is dropping every day. For more information see "XML in 10 points" at www.w3.org/xml/199/xml-in-10-points

So those are the different building block of Data Bases next time we will look at some applications.

So let's look at the story…

"Thank you and Good bye". And so ends another successful customer interaction… But that is not the end of the process. What data has been captured, or changed in that exchange of details? How will it be stored and will you ever be able to re-access it? But that is the problem of someone else, is it not? A Data Base Systems Manager? (or DBSM)

Data bases have been around for a long time. The old "card index" system frequently used in libraries is a great example. In the military there was a system called "shoe boxing" which quite literally comprised of shoe boxes (yes the ones that you bought shoes in!) and the data was torn from the tele-printer and placed in these boxes which represented different units or geographical locations. These systems worked well, but required someone (usually an individual) to update, correct and retrieve the information. This person knew where everything had been put and could retrieve it and even see possible patterns in that information over time. The old copper in the Inspector Frost television programmes on the TV is a wonderful example.

Once the use of computers became wide-spread, then the amount of data that could be stored and the frequency that it could be changed increased dramatically. They also made information "invisible", and opened the possibility for many people to store and retrieve information at the same time.

I used to be the manager for a production line that manufactured the optical receivers for the trans-oceanic telephone cables. The production line made 1000s of items over a 3 year period and each item had 100s of components, and the most important point was that each component must be traceable through its own production to its original materials. When you are putting these devices into 4 ½ miles of salty water – perfection is only just good enough!

I created a database – using the Dbase programme of the day (mid 1980s). I personally entered all the current data that had been collected from the production line during a week's work (yes this was a weekly batch system!), ordered all that data and cleansed the entries. This system was "state of the art" in those days, but looks so antiquated now… However, it did its job – all components were tracked all of the time. A wonderful example was an occasion when the man from HM Customs and Excise walked into my office and demanded "Where is every single PIN diode imported from Japan in the production line." (Now, PIN Diodes are so small that with a single mis-directed sneeze you could lose a thousand or more…) I turned to my "286" PC, opened the database, made a simple enquiry and the printer started printing line after line of the exact details he required. "Stop! Stop!" he cried – the question he really asked was "are you able to show me where ever PIN Diode is?" The system needed no further auditing and we had a pleasant lunch.

As a single data entry point, I was able to understand the data, and look for discrepancies, and even in a system as big as this one such "data cleansing" could be done by one person. But as databases got bigger the problems grew even faster… As a "rule of thumb" – the complexity of a database increases as "n^3" where "n" is the number of entries. Very soon the

 Registered in England No: 06466302

size of the database is so large that no one person can see it all. At this point "data cleansing" becomes both essential, difficult and expensive. One such case was in the early days of BT. We combined all our Sales, Marketing and Customer data. The first thing we saw was that the entries by different people were different! For example the "National Westminster Bank" appeared in 68 different ways – NWB, N.W.B., Nat West Bank, nat west bank etc. It was a very expensive process to trawl through the data base and capture and correct these entries – but it had to be done. With multiple data entry points – rules for naming conventions and entry restrictions had to be devised and implemented.

Just imagine the BT customer data base with 27 million customers all using the phone many times a day – over 6.8Gbytes of information is collected every day – and that has to be groomed so that when it arrives on your doorstep as a bill it is readily understandable.

Also mmO_2 and the other mobile companies have 30 million customers, who also use their phones many times a day – and these customers are moving around! Just imagine the data base that loges the position of every one of these customers all the time!

In the next article we will look at the means that are available to track data in these large data bases – and understand what is happening.

Above we looked at the possibility of databases becoming so large that they become almost impossible to manage…

The BT network is one of the largest machines ever built, and collects data in enormous amounts. Approximately 6.8Gbytes of information is collected each day and every three months this is amalgamated into phone bills that need to be completely accurate and easily understood by our customers.

But in a system so large, there will be fraud and fraudulent activities – there are enough villains out there and they are smart enough to guarantee it!

The big problem is to originally analyse exactly what fraud is being perpetrated. Just like in all law and security matters the villains can establish an activity at their choosing, the law enforcers need to be aware of everything and at all times. What is needed is to be able to look for patterns in these enormous amounts of data. Just such a case arose recently.

In the "old days" when creating data bases you had to have a pre-conception as to the types of enquiries you were going to make in the future and then construct the data in a form that could be so questioned. As data bases grow (and this growth is approximately the cube of the number of data entries involved) such pre structuring becomes not only impossible but also inadvisable.

In this case the enquiry of the data was "how many of the phones in the London area share at least 50% of their calls with other London area phones". The resulting data was turned in a coloured picture. The phone line identities being on the circumference of circles and the width of the lines between them indicated the proportion of the calls being made. Very rapidly the pictures developed into recognisable patens to the human observers. We humans are quite remarkable at recognising patterns. The eyes in the front of our heads can handle 1Gbyte of data each second and we possess a "wet-ware" computer in our heads that can process data at these rates!

The pictures clearly showed a scam of a group setting up a Premium Line number then exercising it themselves. The case went to court and the "best evidence" in a UK court for the first time was pictures not the written word. The data was so complex that a jury would have been totally unable to comprehend it. The pictures made the case easily understandable. In a similar way the discovery of people in a war torn part of the world ringing London and auto-call forwarding calls to their colleagues across the war zone has caused this type of behaviour to be constantly monitored throughout the network.

But data bases are getting more complex and dynamic. Just image a system that holds all your preferences, and these are tested against all the preferences of other people within a given distance even when you are moving around town. It could be the ultimate "Dating Agency"! As soon as someone within the given radius fits your desired profile, and you

equally fit theirs, you each receive notification via the handset of your 3rd generation mobile phone. Then it is up to you if you take any action, or not.

This is not only an incredibly large database but one which is incredibly dynamic as well. One company APAMA (www.apama.com) has inverted that thinking on data bases. The original way of looking at data bases was to have the data as a static unit and dynamically interrogate it by enquiries. But in the larger data bases the data is in practice quite dynamic and the enquiries mostly static. In the "dating agency" above one's preference are unlikely to change that much or that quickly!

By passing the dynamic data past the static enquiries massive interrogation rates can be achieved and handled by relatively simple data base engines.

The world of data bases is changing to meet our greater and greater demands for information.

Earlier we looked at the early days of data bases, where the information was manually added via the QWERTY keyboard by people. These days information is gathered more automatically.

Certainly there is still some data entry from the keyboard – but progressively the keyboard belongs to the customer… The World Wide Web has enabled the connectivity of customers and processes and enables them to enter their own data – and who should understand (and be able to correct) their own data better than customers.

But I am a customer, and I have to admit that I am becoming progressively dis-enchanted with having to enter and re-enter all my personal details every time. This is especially the case when, having entered all the details, there is a problem or glitch, and I find myself back at the beginning staring at a page full of little blank boxes…. Quite infuriating!

As we discussed in the Security article recently, I do see a time in the near future when we will carry SmartCards, which will contain all our personal details. Placing this card into the SmartCard reader (and yes I do see computers, TVs, phones etc. having these readers) will instantly transmit the required details simply and consistently. This will leave me free to concentrate on the important and interesting part of the transaction.

At Adastral Park, the R&D arm of BT Group, the process is being taken a step further. We are taking the machines that talk in a reasonably human manner and even listen and understand what you have said (see earlier article on talking machines) and combining these with the usual Web access. You log-on to the www site of your choice and at the same time telephone the site as well. The phone call can be via a fixed wired phone or a mobile phone – the system does not have any preferences…

You can, if you wish, simply type the details into the customary boxes in the old way, but over the phone line you hear a voice requesting data. For example it might well say “First we need to collect a few details about you. What is your family name?” You respond by saying your family name into the phone and instantly you see the word appear in the relevant box. More complex, or unusual, names might need be spelled out, and spelled in a manner that you would use to a human – no need to put long pauses between the letters! And if your name is so complex that you defeat the system – you can revert to simply typing the response.

The magical part is that whilst the system is addressing your family name you could be concurrently typing the details of your date of birth. The system is smart enough to know that you have entered the details and will skip over those details as it progresses through the form.

Voice commands also allow simple navigation through pages in your browser and the execution of instructions.

Registered in England No: 06466302

At this point most people in Call Centres are beginning to think “Hey! That is my job!” And I have to admit that yes it is….. The next reaction tends to be “Hey! This is putting me out of a job!” And there I have to disagree…

This type of system allows the customer to input data in a simple, enjoyable and fun way – personally I find it quite embarrassing to have to wait whilst a real person at the far end of the line types the details that I have just spoken into the machine. I really do prefer to enter the details myself – but I am no typist and the process is laborious!

Make the system so easy and fun, and I will not only want to be part of it, but I will want to return and use the system again. Once I have entered the boring data capture – then I need the assistance of the human to do the interesting parts. And as the systems become so much easier then the demand for their use will increase and more Call Centre agents will be required not less, but elevated to the level where the human intellect is needed.