Nineteen Sixties History of Data Base Management

T. William Olle

T.William Olle Associates, Walton on Thames, Surrey, UK
<BillOlle@aol.com>

Abstract: Data base management evolved during the sixties and seventies. The evolution period was protracted. Many driving forces impacted the evolution and it is the aim of this paper to analyze these driving forces (some technical and some political) and to discuss the impact of each. The driving forces are identified as follows: higher level languages, generalization of software, non-procedural approach, program maintenance, recognition of different levels of data definition, direct access storage, and relational theory.

1. Introduction and Terminology

The origins of the term "data base" and subsequently "database" go back a long way. The first sighting of the term was its use in 1963 by the System Development Corporation who sponsored a symposium with the title "Development and Management of a Computer-centered Data Base" [1]. The term "data base" was picked up by the contributors to the symposium in the titles of their papers. The defense industry in the USA was a major development force in those days.

The term "data base management" was a natural derivative from the juxtaposition of "data base" and "management". The terms "database" and "data base" have subsequently both been in widespread use. However, the latter is preferred in this paper.

At the time (namely 1963), the primary storage medium for data was magnetic tape. Early work on data processing focused on the optimization of the processing associated with magnetic tapes and on the required sequential processing through a file of data. The term "data base" was at the time and indeed still should be related to the term "file" in the following way. A data base is a collection of inter-related

files. Many authors confuse the two by using the term “data base” to refer to a “file”.

2. Higher Level Languages - COBOL

At the time the concept of “data base management” was beginning to emerge, programming languages for commercial application were rather more advanced. The initial developments were based on a series of earlier languages and systems. The US Department of Defense had used its influence to bring about the development of a common language for business oriented applications. This common language was COBOL - Common Business Oriented Language – the first version of which was available in 1960 [2]. This early COBOL had no facilities for, and indeed awareness of, database management. However, it is useful to note some of COBOL’s ideas had an impact on the development of data base management.

COBOL recognized the importance of separating the definition of the data from the definition of the procedures to be performed on that data. These concepts were provided in the Data Division and in the Procedure Division of COBOL. In the FORTRAN language, which was developed largely for scientific and engineering applications, the two definition processes were intermingled.

COBOL also offered what seems (with hindsight) to be a primitive way of structuring data. One could define files, record types, groups and data items. A file could have one or more record types. Each record type could have one or more items and zero or more groups. Each group could be either a fixed group (basically a naming convention) or a repeating group which allowed a group of data items to occur a number of times up to a prescribed maximum. These conventions allowed for the definition of different kinds of hierarchical structure, which could be represented on the widely used magnetic tape files of the time.

COBOL was above all a procedural programming language. It was a considerable advance on machine language programming available on the earliest machines and indeed on Symbolic Assembly Language, which was specified in 1957.

The terms “procedural” and “non-procedural” were often interpreted as follows. “Procedural” said “what” and “how”, “Non-procedural” said “what” but not “how”. These two concepts are not “black” and “white” but rather represent a spectrum of capability. COBOL represented a useful progression along this spectrum from the procedural to the non-procedural – but further progression was needed.

3. Generalization

In January 1959, William C. McGee of IBM published a paper with the title "Generalization: the key to successful data processing" [3]. This paper emphasized the idea that any computer program should be designed to address, not just one simple application, but a set of necessarily inter-related applications. Given the open interpretation in any business context of the term "application", it is perhaps more appropriate to think in terms "small" and "large".

However, the message is clear. It is inefficient and indeed time consuming to design several small applications, each with its own program, rather than one larger application in which the processes to be performed are generalized and the data which they reference is "integrated". The complex trade-off between pre-design and post-integration is still with us today. In the era before higher-level languages, the balance towards the latter must have been less telling. McGee's seminal paper clearly foresaw the emergence of data base management.

4. Disc Memory – IBM Ramac

The first computer to include a disc drive was introduced by IBM in September 1956 [4]. The computer was the IBM RAMAC and the disc drive was called the "IBM 350 disk drive". It had a capacity of five million 7-bit characters that translates to 4.8 megabytes in contemporary terms.

Despite this early introduction, IBM did not seem to push the new facility. This may have been because the capacity of the disk was modest and the cost high compared to the already widely used magnetic tapes. Even the first magnetic tapes on the Univac machine had a capacity of 1.4 megabytes and it was possible to have several such magnetic tape units on a single machine.

Another reason for the slow uptake of disc memory might well have been the uncertainty of how to make effective use of these devices.

5. Bachman and Dodd

Charles W. Bachman is widely recognized as a pioneer in the field of data base management. One of his papers [5] bears witness to the basis for this recognition. In October 1964, C.W. Bachman and S.B. Williams published this paper [5} in the Proceedings of the Fall Joint Computer Conference. It was entitled "A general purpose programming system for random access memories". The term "random access memory" was in early use to refer to what subsequently and more accurately became designated as "direct access memory" on the basis that the accesses to the data were certainly not random (or should not be).

In comparison to the sequential accesses to data necessary with magnetic tape storage, direct access memory certainly provided an extra dimension to the processing. The Bachman and Williams work certainly provided a means of exploiting this extra dimension. Bachman's approach was called "Integrated Data Store" and was developed in one of General Electric's divisions in which Bachman was employed.

Bachman subsequently claimed that the choice at the time of the more British term "store" was preferred to the American "memory" was quite deliberate. "Integrated Data Memory" certainly lacked something!

Also in the early sixties, George Dodd, of General Motors Research Laboratory, developed an approach to handling data, which was somewhat similar to Charles Bachman's. This was published the Proceedings of the Fall Joint Computer conference in 1966 [6] and entitled "APL - a language for associative data handling in PL/1". (This use of the acronym APL had nothing to do with Iverson's APL).

The two approaches had in common their departure from the hierarchical structures predicated on the sequential processing of magnetic tape files. The two approaches were both based on the less restrictive and more flexible network structures. In mathematics terms, the structuring facility could most accurately be described as a directed acyclic graph structure. This is based on the fact that circularity was not permitted in the structures.

6. Program Maintenance

Another driving force in the evolution of data base management systems was the realization that was creeping in concerning the cost of corrective maintenance on computer programs already in use.

Many companies went into computerized data processing with great enthusiasm first time round. Computers were a major advance on the electro-mechanical accounting equipment, which they replaced. Speed and flexibility were both achievable.

However, after the early programs had been in use for some time, it was found necessary to make changes to the code. While this was certainly easier if the code had been written using a higher level language, it was realized that, if more than one program was using the same file, any change to that file to support a new business requirement would have a knock-on effect on other programs using the same file but which otherwise did not really need to be modified.

This problem began to hit US industry in the mid-sixties. There was a need to redesign and reprogram application systems and hopefully not repeat the earlier mistakes of the first generation systems. The search for a "silver bullet" frequently found a home in the emerging database world.

7. COBOL List Processing Task Force

The development of COBOL discussed above and Bachman's IDS were two sets of ideas that had to be brought together. In 1965, COBOL was already gaining considerable momentum and the host organization, CODASYL (Conference on Data Systems Languages), had supported further work after the first draft of 1960.

It was in 1965 that one of the major non DOD non computer manufacturer protagonists, namely US Steel, proposed that the COBOL Language Committee of CODASYL should to create a Task Force (clearly a military term) to investigate the extension of the COBOL Language to incorporate the kind of facilities which Bachman had proposed in his IDS work. The "task force" was initially named the List Processing Task Force."

It should be noted that the term "data base" was already in use in 1965, but not in the COBOL Committee and not in the IDS work. Hence the term "list processing" which was in more widespread use at that time, thanks to the work of MacCarthy, in artificial intelligence (which was mostly Fortran based) [7].

Both Charles Bachman and George Dodd were involved on the early work of this List Processing Task Force. It was not until May 1967 at a meeting in Minneapolis (attended by the present author) that the meeting decided that the term "list processing" gave the wrong impression to practitioners and that the term "data base" would create less confusion. The Department of Defense oriented term "task force" was also replaced by the less emotive tem "Task Group" to generate the better-known term "Data Base Task Group". There is little doubt that this decision lead to the widespread acceptance of the term "data base". Given the emphasis, already in those early days, on "management" in all its forms, the term "data base management" was part of the package.

8. CODASYL Data Base Task Group

The CODASYL DBTG was a very hard working group between 1967 and 1971 holding meetings every two months. The first report to the parent committee (the CODASL COBOL Language Committee) was submitted in December 1968, followed by a second more complete report in October 1969. The final report was published in January 1973 entitled "Proposal for a data base facility in COBOL".[8]

The DBTG work introduced some ideas that were not in IDS or APL. The view that a data base was something central to an organization's data processing had emerged. The data base could be accessed by several programs which were written in COBOL and possibly in other languages. IBM was beginning to promote PL/I at the time and the DBTG felt that it was meaningful (and possibly also political) to support this multiple programming language thinking.

This evolved into the notion of a programming language independent schema and to the concept of a set of programming language schemas. (The term "schemata" was favoured by some language purists).

For reasons only a committee can conceive, much of the IDS terminology was changed. For example, IDS terminology referred to "chains" formed by two (or possibly more) record types, one of which was referred to as a master and the others as trailers. The IDS term chain was deemed to connote a specific representation of data in storage and the term "set type" was coined to depict this construct. A set type may have an owner and a member.

The work of the DBTG had a major impact. Implementations were produced by various vendors excluding IBM who initially were committed to an old style hierarchical system IMS that had been developed in the late.

Most international companies developed implementations of the DBTG proposals. These included ICL, Phillips Electrologica, CII, DataSaab and Hitachi. However, the implementation produced by an IBM customer, namely BF Goodrich and known as IDMS was taken over by a new company started by John Cullinane. Cullinane did rather well from their sales of IDMS to IBM customers. The early mastermind behind IDMS was Richard F. Schubert (indeed a descendant of his famous namesake). Dick Schubert had also been an active member of the DBTG.

Tribute must be paid to the two persons who served as chairman of the DBTG during most of its existence. The initial chair was Warren Simmonds of US Steel mentioned earlier. When he resigned because of other commitments, his responsibilities were taken over by Appollon Metaxides who was employed by the Bell Telephone Laboratory in Holmdel New Jersey.

9. CODASYL Systems Committee Publications

In the years from 1962 when Bachman published his initial paper on IDS, interest in what came to be called "data base management systems" burgeoned. Numerous systems emerged bearing the acronym DBMS – data base management system. In 1964, the CODASYL Systems Committee, a sibling committee of the CODASYL COBOL Language Committee (parent of the DBTG), turned its attention to these systems.

It was clear that several systems were becoming available reflecting various factors identified in this paper as contributing to the idea of database. In addition to the COBOL Language Committee, CODASYL had also created a committee with responsibility for taking a more long-term view. The CODASYL Systems Committee had been in existence since the formation of the CODASYL organization in 1959

In 1966, the Systems Committee started to analyze "generalized data base management systems". The term "generalized" was added in order to emphasize

that the systems being analyzed were each applicable to any kind of application area for which a data base could be defined. Such a system is different from a "tailored" system, which was defined for a specific application.

The Systems Committee published two reports in 1968 and 1971. The first report comprised 400 pages and was entitled "Survey of Generalized Data Base Management Systems" [9] and presented a survey of each of nine systems available at the time. The aim of the survey was quoted as being "to find a common basis for describing generalized data base management systems, to demonstrate its usability, and to make this information available for further theoretical and developmental work within CODASYL". The aim was not to assess evaluate any system. The reason for avoiding comparative evaluation was that any kind of collective and objective evaluation at the time by persons (many of whom were involved with one of the systems analyzed) would have proved impossible.

Following on publication of this survey, the CODASYL Systems Committee immediately started work on a more detailed analysis of generalized data base management systems. The 1971 report of over 500 pages was entitled "Feature Analysis of Generalized Data Base Management Systems" [10]. Several of the systems analyzed in this second report had been treated in the earlier report. These included IBM's Generalized Information System (GIS), GE/Honeywell's Integrated Data Store (IDS), Informatics Mark IV, System Development Corporation's Time-Shared Data Management System (TDMS) and Western Electric's System Control-1 (SC-1). The Feature Analysis also included an analysis of the CODASYL DBTG's Proposal [8].

During the seventies, it can safely be claimed that the three reports [8,9,10] added considerably to the understanding of what data base management was all about and helped to promote acceptance of the concepts and hopefully clarify some aspects of the confusion which prevailed about the systems being promoted.

10. Logical Data Structure and Storage Structure

One of the aspects of data base management, which caused confusion, was what is here identified as the level of data definition. In simple terms, the level of data definition refers to the extent to which a definition of the data in a given DBMS referred to the degree to which a DBMS represented the stored aspects of the data structure.

A definition of the logical perception of the data could be referred to the "logical data structure". A definition how the data structure was perceived to be represented in storage was referred to as the "storage structure".

A DBMS could possibly not any contain facilities for defining the storage structure. This meant that the system had an essentially unique way of representing

the data in storage, but the person defining the data (and ipso facto the person using the data) had no means of controlling or influencing this definition.

Alternatively, a DBMS could provide both a logical data definition and a storage definition. This, it would be argued, allowed the system to be "tuned" hence providing a more computer efficient system. The other argument was that the storage definition was not something that the user needed to both about when defining a process to access the data.

It should be pointed out that at the time much of the development work already described in this paper was being undertaken, computer hardware efficiency was a major issue for larger applications, that is to say larger data bases. The trade-off between hardware efficiency and people efficiency had already been moving towards the latter. However, in 1971, it was still a little premature to assert that compute time should be regarded as an inexpensive resource.

11. Relational Approach

In June 1970, the late Ted Codd published his seminal paper entitled "A relational model of stored data for large shared data banks" [11]. This paper did not build on earlier work in the data base field, but rather on some theoretical work in pure mathematics concerning relational calculus.

It developed its own set of terminology – relations, tuples, attributes and referential constraints were used to refer to concepts already well established as record types, records, data items and set types. Of the four terms in each of these two terminology set, the last one of the four is the one which was clearly an improvement on its predecessor.

The relational approach triggered more research work into data base management than any other published paper in the field of computer science. While data base management at that time had been largely of interest to practitioners involved in commercial applications, the academic world was now attracted to the whole issue of the "best" data base approach – in whatever way the "best" might be interpreted!

12. Debates on Data Base Management

During the early seventies, there were numerous debates held at technical conferences all over the world on the relative merits of various data base approaches. Interest focused on the CODASYL DBTG proposals for which commercially available implementations were being promoted, on the relational proposals, Cincom's TOTAL and IBM's Integrated Management System (IMS), a German system called ADDABAS.

Both TOTAL and IMS were aggressively promoted during the seventies. The CODASYL DBTG proposals were implemented by practically every computer manufacturer except IBM. The main implementation for the IBM 360 hardware was Cullinane's Integrated Data Management System.

13. Database Standardization

It is important to recall that the CODASYL work discussed earlier in this paper was never seen as a standardization effort. The CODASYL organization (now defunct) had established its roll in connection with COBOL in the mid-sixties as "pre-standardization) with full awareness that national organizations such as AFNOR, ANSI and BSI had responsibility for defining and approving standards.

As the world became smaller (in a certain sense) the role of the International Standards Organization became more significant. The database work was one of the areas where the major thrust of standardization took place in an international context.

The history of data base standardization is as extensive as the nineteen sixties history of data base management presented in this paper. Although many papers and articles have been written on this topic, an update remains a topic for a subsequent paper.

References

[1] "Development and Management of a Computer-centered Data Base" 1964. Published by System Development Corporation as TM-2624/100/00.
[2] COBOL - Common Business Oriented Language 1960.
[3] "Generalization: the key to successful data processing" Published in January 1969 in Communication of the ACM.
[4] "IBM 350 disk drive" 1956. http://www.cedmagic.com/history/ibm-305-ramac.html
[5] C.W. Bachman, S.B.Williams "A general purpose programming system for random access memories". Proceedings of Fall Joint Computer Conference, October 1964 Volume 26 Pp 411 – 422.
[6] G.G.Dodd "APL - a language for associative data handling in PL/1". Proceedings of Fall Joint Computer conference 1966. Vol 29 pp 677- 684.
[7] http://encyclopedia.thefreedictionary.com/Lisp
[8] "Proposal for a data base facility in COBOL". CODASYL Database Task Group. January 1973.
[9] "Survey of Generalized Data Base management Systems" Published 1968. CODASYL Systems Committee
[10] "Feature Analysis of Generalized Data Base Management Systems" Published 1971. CODASYL Systems Committee
[11] E.F. Codd. "A relational model of data for large shared data banks", Communications of ACM. Vol 13 No 6. Pp 377 – 387 (June 1970)