Commentary on the CODASYL systems committee’s interim report on distributed database technology

by CHARLES BACHMAN
Honeywell Information Systems
Waltham, Massachusetts

The author has reviewed the Interim Report on Distributed Database Technology and is responding from the vantage point of the ANSI/X3/SPARC/Study Group on Distributed Systems. This study group is attempting to establish an architectural approach which will put all aspects of distributed systems into perspective, including distributed databases. The author has been studying this problem as part of Honeywell Information System’s investigation for the last three years.

My first comments must be addressed to the overall document which is well written and conveys its message well. I am in agreement with everything in it and will address myself to those aspects which it does not cover at the present time. As an interim report, one cannot criticize it severely for having some gaps. It has been circulated for comment and is soliciting comments so that the final report may be more accurate and more complete. I recommend the reading of this report to all interested parties and urge each reader to comment so that the final report will be as complete as time permits.

The interim report lists a number of motivations for distributed systems. All are valid in one situation or another. There are a couple of additional reasons which I would like to add to the list. The first reason is a human factor reason. This is to permit some element of the business to physically possess the part of the corporate database which holds its part of the data. This gives them the responsibility and the satisfaction of updating their database without removing it logically from the required access by others. Another human factor reason relates to the distribution of the database to reduce its vulnerability to strike action or acts of terrorism. Another reason is the matter of database size vs. computer power. There is no computer on earth which can by itself handle all the data processing for a large corporation. Therefore, processing must be distributed and is already distributed for most of the FORTUNE 500 companies. In most cases they are already physically distributed but not logically integrated and will not be logically integratable until “distributed database facilities” become available.

There should be one item added to the list of reasons for the centralization of data processing. This is to achieve a concentration of expertise sufficient to run a large data processing operation. From an administrative point of view, this is very important. Learning how to distribute the expertise along with distributing the processing and database may be a real roadblock.

From the vantage point of the ANSI Study Group on Distributed Systems, the report seems to lack a certain architectural overview which would put distributed database management into perspective. The ANSI distributed systems reference model begins with the notion that the information processing aspects begins with the notion that the information processing aspects of an enterprise should be viewed as a network of cooperating workstations. Each such workstation is a “location” where a certain kind of work takes place. This workstation may be manual, industrial or computerized. This workstation supports one or more processes which execute the procedures of the workstation and which primarily access the locally stored data. Access to remote stored data is acceptable, but known to be slower and more expensive. The processes of the workstations cooperate by exchanging messages which request certain actions and transfer such data as appropriate.

Where the process and the data it accesses are in the same machine, data access operates essentially as it did before it was distributed. Where the process and the data are in separate machines, more elaborate mechanisms are necessary. Those described in the CODASYL interim report are representative. One thing which appears missing from the report is any discussion of how processes and data would be partitioned and set up in the various machines in the network. While this is a packaging issue, my current guess is that it is a critical one. Probably, as critical to distributed database access as the “clustering” concept is to the performance of our current centralized database. Simply speaking, clustering is the placing together in the same page as many records of a set occurrence as possible. Where one set has been chosen in favor over another because of the
anticipated frequency of access. Frequency of access is also
the heart of the database vs. process distribution questions.
It has been said that there are two approaches: (1) take the
data to the process and (2) take the process to the data. The
carefully preplanned distribution of processes and the data
which they are most likely to access to the same machine
is the most important step which avoids much of the proc-
essing burden associated with distributed processing.

Within the ANSI reference model, three alternate sche-
mas have been discussed to achieve access to distributed
databases. From our point of view, the picture presented by
the CODASYL interim report describes the most difficult to
design and implement. It will also be the least efficient under
certain operating circumstances. The three approaches dis-
cussed are characterized as;

(1) cooperating user workstations
(2) cooperating storage management workstations
(3) cooperating database management workstations
(CODASYL)

All of these approaches assume that a facility for process to
process exchange of information is a basic capability of a
distributed system. The difference in the three approaches
is who is talking to whom and what is the nature of the in-
formation exchanged.

The cooperating user workstation approach assumes that
the users are knowledgeable of the data file vs. process
(workstation) configuration. Thus, if a process cannot get
what it needs from the local database, it will formulate a
message to be sent to another workstation which is co-lo-
cated with the data which needs to be processed. When the
message arrives at the workstation, a process of that work-
station will access the message and carry out whatever local
access and updating is necessary and respond with an an-
swering message. When workstations are considered as ele-
ments over which the total work load has been divided, then
it does not make much difference if one is designed to
approve a credit request for a new sales order or whether
one is designed to answer questions about the Minneapolis
inventory. It is only a slightly further move to design a
workstation which receives data manipulation commands as
a message text, executes these commands in terms of its
local database management system, and sends a response in
accordance with its success.

The cooperating storage management workstation ap-
proach is based upon the fact that most of our database
systems are built as virtual memory systems and do "page
turning" independent of the operating system and hardware.
When a page is requested by the database management
subsystem the storage management subsystem asks whether
the page is in a main memory buffer or whether it must be
fetched from the disc? If it must come from the disc, then
the question is which disc drive and address? With a dis-
tributed storage management system, the third question
must be asked, my disc or one managed by some other
storage management subsystem? If it belongs to someone
else, then a message must be formulated with the request.

A response message will either return the page, state the
process must be waited until the page is available, or stating
that a deadlock would result if the process were to be waited.
The cooperating database workstation is essentially that one
described in the CODASYL Interim Report.

The CODASYL report discusses several kinds of control
tables: network directory, concurrency control, integrity
control and security control. There is a discussion appearing
several places in the report which questions whether these
tables should be centralized or distributed. From my point
of view, there is only one answer: distributed. The central-
ized approach has all the problems which the distributed
database system is attempting to side-step. e.g., the vulner-
ability of the entire system to the failure of one element, the
congestion problem, etc.). What must be done is to give
each node the information about its local resources, process
and only information about remote objects of local interest.

One note about the concurrency control is relevant, the
problem in a distributed system takes on complexity beyond
just the aspect of multi-nodes. Most concurrency control
systems have a secondary responsibility to be sure that two
or more processes are not waiting for each other in order
to access data (deadlock). With the introduction of message
exchange between processes, it is possible for process (1)
to be waiting for a message from process (2) which is waiting
for a record held by process (1) . . . . deadlock! It also must
be considered that a process may be waiting upon a terminal
operator action. If that terminal operator could possibly be
waiting on the action of another process in the system, then
that anomie must also be included in the path of waiting
processes to determine if deadlock exists.

The question was raised concerning the effect upon con-
currency if a record were to be split between nodes. This
isn't a problem because the assignment and wait logic can
work at any level of granularity. It could work at the level
of the logical elements: item, item group, segment, record
or file. It could work at the level of physical elements: page,
group of pages, track, cylinder, or extent. As long as the
control is maintained non-ambiguously, it will work. The
only difference is the smaller element requires more book-
keeping and the larger elements force processes to wait and
possibly be recovered from deadlock when the effective
access desired would have caused no interference.

The text described two basically different forms of distri-
bution: distribution of parts of the only copy and distribution
of copies of the whole database. Where updating is a per-
tinent issue, the distribution of copies seems less efficient
than distributing parts of the only copy and getting it updated
at its only location. We have seen several large systems in
the field which have chosen a variation which is worth
describing. It could be considered as a form of the multiple
copy approach. In this case, major portions of the file are
copied and partitioned to be sent to a number of transaction
processing machines. These transaction machines handle
the on-line daytime processing of their partition of the da-
tabase. All on-line transactions are also sent to a centralized
system where they are batched with additional off-line trans-
actions and reprocessed in a nightly batch run. At the end

From the collection of the Computer History Museum (www.computerhistory.org)
of the nightly run, the updated centralized database is again copied and updated partitions are sent to each transaction machine.

So much for the extension of the generalities. The real question is where do we go from here. I see the only satisfactory long term answer in the form of a set of international standards which permit all kinds of computers, software, communication facilities and terminals to be joined together to form an extended network of cooperating workstations. I say extended because the workstations served will need to communicate within companies and between companies and across international borders. No one vendor or one country is large enough, smart enough, or influential enough to go it alone. This is the reason I have supported the ANSI effort and the ISO effort to move toward this standard. We have set ourselves an impossible schedule to reach the goal of international standards. The end of 1978 is the goal for an internationally accepted reference module (levels, protocols and interfaces). The end of 1980 is the goal for the first new protocol and interface standards which will support the interprocess exchange of information. Working groups are being established to study and prepare specifications. One of these is concerned with the distributed database access and control. I expect that the CODASYL System Committee with its members and report will make a significant contribution to making those standards. Are you ready to move and move rapidly? Are you ready to help write and negotiate these standards?