RECORD OF SOCIETY OF ACTUARIES
1981 VOL. 7 NO. 3

RECENT TRENDS IN COMPUTER TECHNOLOGY

Moderator: MATT TUCKER. Panelists: QUINTIN J. MALTBY, A .G. HASSELMEIER, DAVID B. ATKINSON

1. What is the "new" hardware?

2. What opportunities are offered by the "new" hardware?

   (a) Administrative/service-oriented possibilities
      (1) Totality of information storage
      (2) Paperless administration/elimination of hard copy records
      (3) Effect on user
      (4) Different/flexible structure of information
      (5) On-line processing concurrent with data processing

   (b) Product-oriented possibilities
      (1) Universal life
      (2) Flexible premium annuities
      (3) Adjustable life
      (4) Retired lives reserve
      (5) Bank annuities
      (6) Variable life/annuities

Discussion will include the administrative needs of each product.

MR. QUINTIN J. MALTBY: My job is to review some aspects of recent hardware and software developments and then to discuss some of the pressures that affect life insurance company data processors.

We have all read about the so-called "Microchip Revolution" and how it will do wonderful things for us and dreadful things to us. As is all too usual in such media presentations the truths -- and there are many -- lie throughout the middle ground. I might also add that different segments of society are affected at very different rates of change. This revolution is being fueled by the large scale integrated circuit silicon chip. These chips are cheap, they are realiable, and they pack tens of thousands of circuit elements into spaces measuring a very few cubic centimetres. Developments in the past decade have meant that sophisticated E.D.P. facilities are no longer the exclusive preserve of large organizations whose annual commitment runs into the millions.

Another facet of these recent developments is the widening array of offerings that make up the supply side of today's E.D.P. marketplace. The ratios between the fastest over the slowest, the biggest over the smallest, and the dearest over the cheapest are all much greater than they were even five years ago. There is also a change in the manner in which the hardware is controlled. In the past it was done purely via the stored program which was always loaded from an external storage medium. This "software" control is now being assisted by an intermediate built-in element called "firmware" which is really software at a more basic building block level that is subject only to engineering modifications. All of these opportunities pose a real challenge to those who have any responsibility associated with the procuring of equipment. They need both more information and more wisdom to ensure that tomorrow's problems will be solved when delivery takes place and that a base is in place from which to attach to problems of the next day.

In David Atkinson's presentation you will be treated to some figures comparing the past with the present, to some current systems development trends, and to an example of hardware designed to deal with an area that has heretofore not been subject to computer assistance.

Computer networks have evolved into very complex and sophisticated entities. My prediction is that they will not become such simpler in the future. We hear a lot about Distributed Processing today. Unfortunately it is subject to as many definitions as there are vendors in the marketplace. If you are buying a service that delivers what you need, all is well and good. However, if you are involved in building such a network, it is wise to remember Murphy's #8 which says "Whatever you want to do, you will always have to do something else first".

The data storage media of today are faster and much more capacious than in the recent past. We have read about research into exotica such as laser memories and magnetic bubbles which may give order of magnitude breakthroughs in the future. Meanwhile, back in the present, magnetic disks are the high-capacity on-line storage state of the art device. I would like to deliver two cautions. First, there is an unprecedented demand for such devices which keeps prices firm and stretches delivery dates. Second, just make sure that you can manage all the data that is out there. It has a tendency to grow in an uncontrolled fashion.

Graphics capabilities are another area which uses specialized cathode ray tube terminals. These are central to such applications as Computer Aided Design and other related engineering activities. There has been increasing business use in software packages that display data in a most readable form through the use of bar graphs, pie charts, map shading, and other means.

In the past, output to be viewed by humans was usually printed on paper -- a lot of paper. There is still a lot of paper being impact printed, but I would like to mention three specific improvements in this area. First, there is "C.O.M." which means computer output to microfilm. In this manner large volumes of historical type material can be very efficiently and compactly stored on microfilm or microfiche and accessed off-line from the computer. This approach saves trees and saves space, but does tend to push up the price of silver. Second, there are the laser printing subsystems such as the IBM 3800. These are fast and efficient devices that among other things can output in multiple fonts and simultaneously print the fixed business form along with its variable data. Third, there is the use of on-line systems for both development and production. In these, the screen can act as the primary means of communication with its human operator with printed output being limited to final copy that is specifically requested, audit trails, and transaction listings.

When the hardware such as I have been talking about is obtained and then interconnected to become a data processing system, a large sum of money has been committed. To make it produce, though, you must commit an even greater sum to system software, other purchased software, and your systems development and operations staff. People costs are now much more than half of most E.D.P. budgets. I will not bore you with a list of the various job skill sets that are found in a full scale E.D.P. operation. If you are curious, just ask the nearest computer systems executive -- and then remember that several categories were probably inadvertently omitted in the answer.

In order to make all this hardware operate for the user, it must be properly driven by software. This means operating systems, teleprocessing monitors, data base managers, other utilities, and finally applications programmes. The management of this complex is more difficult than it used to be but it is helped by the better modularization of functions that are more often connected through standard interfaces and by the greater use of high level languages. In this aspect one may contrast currently existing individual policy administration systems (which are clumsy monoliths) with those now being developed such as you will hear about later in the programme.

I would like to take just a minute or so to review today's hardware/software complex from the users' perspective. The central computer is no longer a strange and remote entity that takes one or more midnight shifts to eventually process our section's data. It is still a bit strange, but it is no longer quite as remote. We the users are on-line, tapping away impatiently at one or more of the computer's tentacles. Some of our operations are supported rather well. In these, our staff can conveniently access all the information needed to complete a transaction, can complete the transaction on the spot, and once it is accepted rest assured that they won't have to balance the batches days later. User management though is well aware that reaching such a Nirvana is a slow and painful process involving forward planning, interaction with systems they have never heard of, heavy leaning on the systems department, and great bribes that are called "budget allocations". The user is also much more subject to the ups and downs of the hardware itself so that when the main-frame sneezes, some users get pneumonia. In the situations where sophisticated data bases are being properly used, transactions and requests can be processed in a manner that is more friendly to the user, particularly where associated data bases are checked to ensure that incompatabilities do not creep into the totality of one's business data. In general, users will see fewer new batch systems being developed.

I will spend the next part of my time in going over some of the pressures that specifically affect Life Insurance Company data processing executives. It is within this milieu that the major computer efforts to deal with the Enigmatic Eighties will have to be made. In terms of types of systems activity, there are three general kinds, given in increasing order of severity. First is the hardware and its controlling operating system software. Second is the "normal" expansion and upgrading of the applications software base. Third and most discontinuous is the accelerating effect of external influences such as new and very different product demands, for much shorter reaction times for maintenance and development, and the economy in general. There are several specific ingredients in this mix which bear examination.

There is a sizeable installed base of existing systems, both large and small, that were built using the software technology and attitudes of the 1960's and kept that way because maintenance and upgrading seem easier using the old ways. However, we must not be too hasty in jumping on the fasionable bandwagon and considering such systems as just "problems". First and foremost they are "solutions". They have been solutions for a respectable span of years. They have paid their dues and a lot of the rent as well. Can you operate without them? I know that I cannot. The real problem is that these golden oldies are not the best solutions and that achieving better solutions requires major commitments.

There are the new projects, large and small, which receive much systems development attention. They need a judicious mix of patience and impatience in order to keep them moving forward in a manner that will solve the user's problems, keep them solved, and integrate well.

There are the pressures to adopt XYZ Corporation's minicomputer and consulting service to solve a specific problem. The figures look good, in particular when stacked up against what it would cost to run it on our own mainframe. Watch out -- a couple of tiny details may have been left out. How do you integrate this operation with the rest of your systems? How much overhead such as space, power, management cost, etc., has been included in the running costs shown in the proposal?

There are the pressures to react much more quickly in several dimensions. Maintenance of existing systems, addition of new product lines -- "we need it by yesterday", creation of new systems, and conversion to newer technologies are all areas in which such pressures arise.

There is the resource crunch. We can do anything, but we cannot do everything. Even with an unlimited budget, a shop simply cannot staff it and manage it with as many as are needed to do all the jobs that need doing. The people needed do not exist in sufficient numbers at appropriate skill levels.

Finally, there are basic corporate pressures. Some of these are competitive, both within the industry and with other portions of the financial services segment of the economy. Others are imposed, such as consumeristic legislation and practices. We must not forget the basic private sector pressure to produce adequate profit -- because if you don't, you die.

The last portion of my remarks will consist of a few comments about strategies to handle newer non-traditional products. Let us consider the situation in which our company is to issue a "Super Special Saver" policy. This policy just will not conveniently fit on our Individual Policy administration system because of peculiarities such as the absence of regular premium due dates. This policy was designed to meet competition from banks and trust companies.

Can we fit the new policy into our Individual Policy operations? Perhaps, if we expand the system to look after this new breed. However, this must be done at a cost that will make the new product competitive. It seems to me that the only basic systems that are really amenable to such treatment are those that are truly modular, with simple and understandable interfaces between the elements. Anything heavier will cost now, and it will cost in the future just to keep going. The total costs we are competing against tend to involve distribution by lower level salaried personnel helped by timely advertising and system costs that are lean because of the relative simplicity involved.

Can we engineer a special-purpose small system to look after our new line, and if it goes well integrate it into our main system later? Sure we can, but we had better be very certain at the start of the range of activities (and costs) that may reasonably ensue.

Can we contract-out the design and production of a new system? Yes, but even if we can find a software house to do the job, we must be aware that significant work must be done in-house to set up specifications, monitor progress, and finally to install and maintain the system.

Can we use an end-user language such as APL to deal with the administration of the new product? Yes, of course we can. However, we need to remember some systems truths. End-user departments are rarely versed in items such as integration with other systems, security and backup, and maintenance. Documentation, if it exists, is often a disaster with concentration on the algorithms (which are simple) and omission of the data delivery systems (which are usually much less simple).

These are the challenges. These are the tools. It is up to each group to use them properly to cope with the dynamic environment occupied by today's corporate survivors.

MR. A. GALE HASSELMEIER: In explaining the implications of traditional and non-traditional products on administrative systems, I will observe the following general format:

- Definition of Representative Products
- Plan Unit Values Storage Considerations
- Impact on Actuarial Functions

Traditional products can generally be characterized as having fixed death, endowment, and surrender benefits. Premiums are usually fixed with respect to both amount and payment period; however, variable costs can be affected by periodic variation in dividend scales or, more recently, by periodic revision of premium scales via adjustable (or indeterminate) premium provisions in our policy forms. Stored plan unit values have generally been used in administrative systems, thus permitting off-line, loosely-integrated actuarial functions employed to calculate plan unit values. This approach to automation of individual policy administration was in fact a direct adaptation of manual processing procedures and was a practical solution considering the limited computer capacity then available. The following figure highlights traditional product characteristics.

TRADITIONAL PRODUCTS

- Fixed Death, Endowment, Surrender Benefits.
- Fixed on Variable Cost.
  - --Premiums
  - --Dividends
- Stored plan unit values generally used.
- Off-line, loosely-integrated actuarial functions permitted.

Administrative systems oriented towards stored plan unit values do not, however, suffice for non-traditional products such as adjustable life, flexible term, and universal life. In the balance of my presentation, I will concentrate on these products to illustrate why additional administrative system features are required, what companies are doing today, and what they are likely to do tomorrow.

Adjusted Life Insurance (ALI) products have been offered by Minnesota Mutual, Bankers of Iowa, Combined Insurance, and College Life. Although ALI products are traditional life, term, or endowment products at issue, they embody unique unit values whenever certain types of adjustments, or policy changes, are effected after initial issue. The following example (depicting initial issue at age 25 as traditional whole life, with an adjustment at attained age 40 to term to age 65, followed by another adjustment at attained age 55 to endowment at age 85) illustrates why the ALI benefit embodies unique values after attained age 40, since the value already built up in the policy at that age results in "graded" unit values thereafter. This yields what is hoped to be a simple, practical way to process plan changes, and also affords high resistance to replacement efforts.

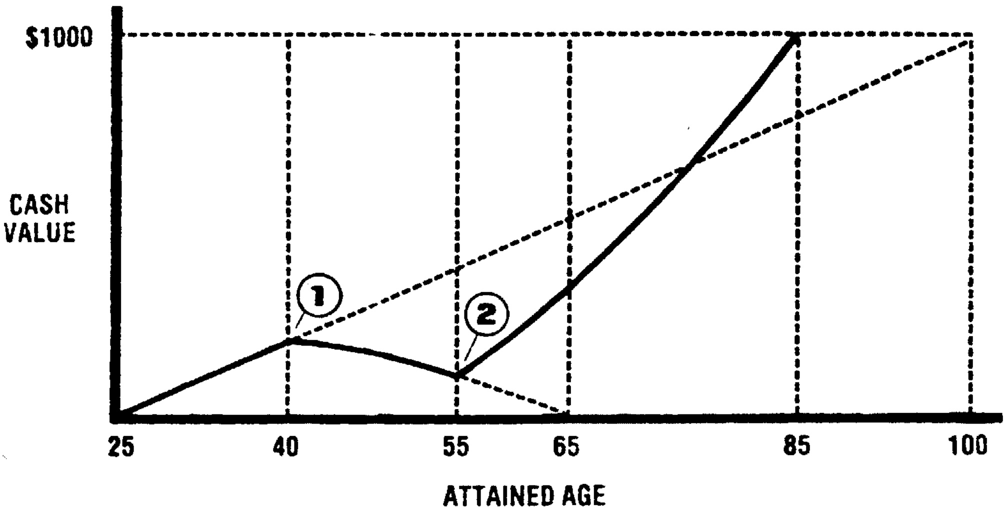
A recent variation, Lutheran Mutual's Flexible Term, supplements and is similar in concept to ALI but is used to provide level term (LT), uniform annual increasing term (UAIT), uniform annual decreasing term (UADT), mortgage protection (MP), and family income (FI) benefits. The general idea is to replace the entire family of current ratebook term products which provide for different premium periods, benefit periods, mortgage interest rates with one plan of insurance. "Shorthand" algorithms, including stipulated benefit and premium payment periods together with actuarial assumptions, enable computation of unit values at issue. The big difference is that these algorithms are kept in the in-force benefit record and not the plan description record. However, ALI-type future adjustment provisions are not included.

The following figure highlights Flexible Term product characteristics:

FLEXIBLE TERM

- Uses: LT, UAIT, UADT, MP, FI.
- "Shorthand" Algorithms.
- Variable Benefit and Premium Payment periods.
- Supplements ALI family of Level Death Benefit, Life, Term, and Endowment Plans.

PLAN CHANGE EXAMPLE



INITIAL ISSUE

AGE: 25
PLAN: WHOLE LIFE

---

1. ADJUSTMENT

AGE: 40
PLAN: TERM TO AGE 65

$$P^1 = \frac{A^1_{40:\overline{25|}} - CV_{40}}{\ddot{a}_{40:\overline{25|}}}$$

---

2. ADJUSTMENT

$$P^2 = \frac{A_{55:\overline{30|}} - CV^1_{55}}{\ddot{a}_{55:\overline{30|}}}$$

AGE: 55
PLAN: ENDOWMENT
AT AGE 85

In summary, ALI and Flexible Term products:

- embody multiple traditional plans.

- result in unique unit values per policy -- associated criteria must be stored for each policy. In general the approach taken in processing these benefits is to keep track of the adjusted net premium and the net valuation premium, as well as the "shorthand" death benefit schedule definition criteria. The general prospective formula, present value of future benefits less present value of future adjusted net premiums (or net valuation premium for reserves), is used to derive the cash value or reserve.

- new contract pages (specification pages, values pages, and in some cases, cost disclosure pages) must be produced whenever an ALI adjustment is effected.

- verification of proposal, valuations, etc., is more difficult than for traditional products.

- actuarial functions must be integrated into administrative system.

Universal Life (UL) products have been offered by E. F. Hutton Life, Life of Virginia, Occidental Life, and several other companies and have been billed as the replacement for all traditional permanent products. UL products emphasize the investment and risk protection elements of permanent life insurance products and are flexible with respect to premium and death benefit schedules.

However, in order to illustrate similarity with traditional concepts and compliance with the Standard Nonforfeiture Law (SNFL), consider some

"SPECIAL" UL PLAN ASSUMPTIONS

- 1958 CSO 4%.

- Level premiums paid annually in advance.

- Claims paid at the end of the year.

- Annual vs. Monthly determination of net amount at risk and associated costs.

- UL first year expense charge incurred at issue and equal to SNFL maximum expense allowance.

- Level percentage loading exactly offsets all non-first year expense charges.

- Level death benefit.

The constrained assumptions employed for this "special" UL plan will yield cash values equal to the minimum cash values prescribed for a traditional whole life plan in the SNFL.

$$P_x^A = \frac{A_x + E_x^1}{\ddot{a}_x}$$, where E^1 is the maximum expense allowance per the SNFL.

$$G_x = (1 + L)\, P_x^A$$, where L is the level percentage loading to offset all non-first year expenses.

This is an important relationship to explain because in policy filings it is necessary, in the actuarial justification, to show that the UL product will develop values that are indeed in conformance with current Standard Nonforfeiture Laws.

Now consider the more typical or

"NORMAL" UL PLAN ASSUMPTIONS

- Traditional Statutory Interest & Mortality Assumptions used to define maximum cost of Pure Insurance and minimum Savings Accumulation guarantees.
- Flexible Premium, with respect to amount and frequency of payment.
- Immediate payment of claims.
- First year expense charge payable monthly and equal to $x per policy + $y per $1,000 of initial face amount.
- Percent of premium expenses deducted from each payment when it is received.
- Monthly determination of net amount at risk and associated costs.
- Monthly deduction of insurance costs and first year expense charges.
- Death Benefit (total or corridor) may be changed on any policy month.

As long as the first year expense charge is less than or equal to the maximum prescribed by the SNFL (note that in some cases the per policy dollar expense requires rather large minimum face amounts), the following formulas can be used to ascertain that application of actual interest rates greater than the minimum specified as well as the charging of mortality costs less than the maximum specified will yield higher cash values.

From Life Contingencies, it follows that

$$ {}_1CV_x = (P_x^A + {}_0CV_x)\ (1 + i) - q_x\ (1 - {}_1CV_x) $$

However, in the interest of redefining the ${}_0CV_x$ and ${}_1CV_x$ terms in the right-hand portion of the equation

$$ {}_0CV_x = -E_x^1 $$

$$ {}_1CV_x = \left(\frac{1}{1-q_x} \right) [(P_x^A - E_x^1)\ (1 + i) - q_x] $$

the original formula becomes

$$ {}_1CV_x = (P_x^A - E_x^1)\ (1 + i) - \left(\frac{q_x}{1-q_x} \right) [1 - (P_x^A - E_x^1)\ (1 + i)] $$

where

the amount of pure insurance is $(1 - {}_1CV_x)$

and

the cost of pure insurance is $\left(\frac{q_x}{1-q_x} \right) [1 - (P_x^A - E_x^1)\ (1 + i)]$

Whereas the above is by no means a precise, complete justification, the resulting formula does illustrate the nature of the processing that must take place at the first of each policy month in order to achieve compliance with the savings and protection goals stipulated by the policyowner. In actual practice, premiums are billed and, whenever received, applied to the savings element after deduction of loadings to offset state premium taxes, cost of collection, and other similar expenses. However, regardless of receipt of premium payments, first year expense charges and insurance costs are determined and deducted from the savings élement on the first of each policy month.

SUMMARY - UL PRODUCTS

- Like ALI and Flexible Term:
  - Embody Multiple Traditional Plans.
  - Result in unique Unit Values per policy.
  - Require more effort to validate Proposals, Valuations, etc.

- Unlike ALI and Flexible Term:
  - Pure Insurance Benefit Amount and associated costs are recomputed monthly and depend upon the then current Total Death Benefit Goal of the Policyowner as well as the timing and amount of actual payments received.

  - Periodic Statements of Benefits, Payment History, etc., required.

- Actuarial Assumptions are applied to actual payment history and consider protection goal of policyowner at least monthly.

Therefore, these non-traditional products illustrate that the definition of a plan of insurance from a data processing viewpoint has been changed drastically. Plan description criteria must include actuarial assumptions and actuarial functions must be integrated into administrative systems in order to support these new products. In addition, greater computing capacity will be required. For example, although storage for plan unit values will decrease, increased storage for additional inforce benefit information (i.e., payment history) will more than offset that decrease.

MR. DAVID B. ATKINSON: My talk today consists of three parts. In the first part I will take you from 1956 to 1981 and show you what has happened in the last 25 years. Then I will take you beyond 1981 and look about ten years into the future to see what kind of changes we can expect.

The second part of my presentation is an example of one of the latest computers. It is quite an amazing machine, and I think you will be very impressed by it.

My talk concludes with a look at the next generation of insurance software.

FROM 1956 TO 1981 AND BEYOND

In 1956, Pacific Mutual acquired a Univac I computer, one of the very first computers. The Univac I was used to support our industry's first consolidated functions processing system. It was the forerunner of most of the processing systems used today by life insurance companies.

In 1956, everything was done in a batch mode. The idea of sitting at a terminal as we do today and interacting with a computer was unheard of. Punched cards had been used since the 1930's, so in 1956 they were the natural form of input. Today we tend to use video terminals wherever we can. The cost of a clerk has quadrupled over the last twenty-five years. This gives us incentive to automate procedures. On the other hand, the cost of a programmer has gone up by a factor of about six over the same period and this makes us think twice before automating certain things.

The memory used by the Univac I was curious, if nothing else. It consisted of mercury delay lines. A mercury delay line was made up of numerous mercury tubes. Electricity came in one end of one of these tubes, was converted to sound waves which passed slowly through the mercury and were converted back to electricity at the other end. A series of these tubes were connected end to end and constituted the memory. The memory's data circulated around the mercury tubes.

Because the Univac I used mercury delay lines, it was limited by the speed of sound, which meant that it could only process thousands of instructions per second. In contract, by using chips, IBM's latest machine, the 3081, handles up to twelve million instructions per second.

The Univac I had a memory capacity of about nine thousand bytes. Since there were no bytes in those days, that is a rough approximation. Before the end of this year, IBM expects to deliver 3081's with 32-million byte memory capacity. If you prorate the cost of these machines, the Univac I costs about $139,000 for every thousand bytes of memory. In contrast, the IBM 3081 only costs $6.40 per thousand bytes of memory.

Not only were punched cards used as input in 1956, they were also one of the primary means of storage. Today we use disks. The latest disk drive stores up to 2.5 billion bytes of data. If you consider what it costs to lease and maintain these new disk drives, it costs less than a penny to store one thousand bytes for a year.

Now take a look at the future. There is no doubt that computer and communications hardware will continue to become cheaper, faster, and especially more reliable. Reliability is an area in which there is quite a bit of room for improvement. We can also expect a trend towards machines with software that is more "user-friendly". By this I mean intelligible to the average person. In other words, we will not require a data processing professional to understand and use the machine. Over the next ten years, I think it is reasonable to expect an improvement of about an order of magnitude in most areas. We will certainly see much more powerful main-frames and minicomputers taking on many jobs in our user areas.

We are currently faced with a shortage of data processing personnel. This is one of the causes of our skyrocketing data processing costs. Another part of the problem is that it is expensive to do even the simplest thing through a data processing department. A startling statistic is that the data processing industry produces an average of about ten lines of executable code per man-day. In the future, we will be making changes to improve programmer productivity. The environment in which the programmer works is going to be substantially changed; it will be simplified and streamlined to make him more efficient. For example, job control language will certainly be replaced with something that is not nearly as archaic and awkward.

Tools to improve programmer productivity already exist, but they are not yet in general use. For example, Bell Labs has developed a system called UNIX, which is termed a "programmer workbench". It consists of many programming tools imbedded in a complete software environment dedicated to program development.

Our society is becoming more familiar with computers and more at ease with them. Within our companies, people in many departments are becoming more sophisticated users of computers. They are learning more and want to get more involved.

Certain of our expenses today are due to data processing middlemen - people that act as interfaces between our user departments and our data processing people. In the future, we can expect that more users will deal directly with data processing people or will do things themselves. We also shall see more systems that are oriented towards users doing things themselves. For example, users should be able to change formats on reports and screens. They should be able to update the files directly. There are also a lot of one-shot tasks that users can handle by themselves, given the right kind of support.

1980 DATA PROCESSING EXPENDITURES (In Billions)

Hardware	16.5	35%
Software	2.9	6%
Supplies	2.3	5%
Communication Lines	1.5	3%
Staff and Overhead	23.8	51%
TOTAL	47.0	

This chart makes it clear that the most expensive component today is programmer-related costs -- staff and overhead. The number of computers in use will continue to skyrocket. With more computers around, there will be even more demand for already scarce programmers, and programmer salary increases will certainly continue to outpace inflation. Because of increased demand, programmers will become even harder to find.

We must find ways to cut programmer costs. There are many possibilities. We can deemphasize program efficiency. There is not as much need today to save CPU cycles wherever we can. In most cases, it is more important to make our programs more straightforward, faster to write and to test, than it is to save a few CPU cycles. The same principle applies to data storage. Why save a few bits and bytes here and there, at the cost of making our programs more complex and expensive?

By purchasing software that is "user-friendly," we can offload more tasks to user departments. For example, time-sharing computers use languages and commands that are very easy to learn. They are not nearly as difficult to use as most other machines.

Rather than develop a large new system in-house, you can look to software companies. Software companies should have a bright future, with more and more companies turning to them to reap the benefits of shared development costs.

Some strategies cost a little more money initially, but may reduce costs in the long run:

- User-tailorable systems -- systems that allow the user to tailor screens, report formats, etc.

- User maintainable systems -- systems that allow the user to create data files, to control programs, and so forth. (Data processing people must be able to get away from the systems they create -- we cannot afford to shackle them to their creations.)

- Systems designed with future needs in mind -- systems that look beyond the short run, that allow for changes. While we cannot predict the changes that will be required, there are techniques we can use to make future changes easier.

DELPHI'S DELTA II

So much for the past and the future; now for the present. The second part of my talk concerns Delphi Communications Corporation and their new machine, the Delta II. Their first machine, the Delta I, was installed near San Francisco in 1976 and now answers 4,000 telephones, acting as a very large answering service. The first Delta II was installed last week in Los Angeles, where it supports a service center that will ultimately answer up to 11,000 telephones. The Los Angeles service center is the first in a national network of service centers. Delphi plans to establish a network that will allow customers in all the major cities to effectively share the same answering service.

As will become apparent, the Delta II is more than just an answering service -- it provides several services that regular answering services cannot. For example, when you receive a message, you can add comments to it and pass it on to your boss, people who work for you, etc. The message will be passed on in the original person's voice with added comments in your voice. You can send messages to a distribution list. The messages are not necessarily delivered when you send them. They are delivered according to your instructions. For example, my phone will ring and an operator will break in and say, "Hello, I have a message for David Atkinson from so and so." If I am not there, the Delta II will try again later to deliver the message. You can specify the maximum number of times that delivery should be attempted.

Delphi's answering service revolves around an in-box/out-box concept. Your in-box contains all of the messages that are waiting for your review. You can peruse your in-box and listen to selected messages; you can delete messages from your in-box. Your out-box keeps track of all the messages you have sent, including when and if each message was successfully delivered. For example, if you are trying to send a message to someone who is very hard to contact, it might take several days before the message finally gets through; your out-box lets you know the status of the message.

The Delta II is capable of much more than just answering telephones. It can process up to 240 MIPS (million instructions per second). IBM's most powerful machine, the 3081, can process about 12 MIPS; this means the Delta II is twenty times faster than the IBM 3081. The Delta II can accommodate up to 208 megabytes of memory. The IBM 3081 can accommodate up to 32 megabytes. So the Delta II has six to seven times as much memory capacity. The Delta II can service nearly 4,000 terminals. It can manage over 400 disk drives. It's also intended to handle video recording and video transmission.

You probably wonder why such a powerful machine is needed to answer telephones. First of all, all voices and pictures are digitized, which creates a vast amount of data. As if that were not enough, the machine must be able to answer many telephones at the same time. You cannot let phones ring while you answer them one at a time.

A Delta II machine is comprised of from eight to thirty-two independent processors; each processor is itself a powerful computer. The processors are specialized; some are CPUs (central processing units), some interface with disk drives, some interface with terminals, etc.

Although the Delta II is not cheap, prices range from $1 million for an eight-processor system to $3-1/2 million for a 32-processor system, the prices are very competitive. When you consider that the most powerful IBM 3081 (which is not even as powerful as a $1 million Delta II) costs $5 million, you can see why the Delta II has some promise.

Just as important as the power of this machine is its reliability. Few computers even begin to approach the reliability of the Delta II. Delphi has gone to great lengths to achieve this reliability.

First, Delphi designed the Delta II to have at least two of everything -- two power supplies, two sets of fans to cool the machine, at least two central processing units, two bus processors, etc. In general, the load is shared between units of the same type. For example, two CPUs can handle twice the workload of one. When one component fails, its partner takes over its responsibilities.

The processors continually check themselves and each other for problems. As soon as a problem is recognized, the bad processor is shut out, the tasks it was performing are reassigned, and an operator is notified of the problem. In many cases, the problem can be fixed by replacing a part, and the Delphi service centers will keep spare parts on hand. Even the cabinets which house the Delta II were specially designed. The Delta II is packaged in such a way that any component can be replaced in five minutes or less. In the case of a bad processor, the operator can simply walk over to the machine, open the door, pull a processor board out and plug a new processor board in: he does not have to turn the machine off or stop any processing to perform this replacement. The machine keeps running and performing its normal tasks while he pulls one board out and plugs a new board in. This same architecture allows you to add new terminals to the system, add new disk drives, or take away a printer without interrupting processing at all. That is a very important consideration for an organization such as the Federal Aviation Administration, which must have its computers operational 24 hours a day, every day of the year.

The net result of all of this attention to reliability and availability is certainly impressive. During the first quarter of 1981, the Delta I (the prototype machine) was unavailable for only fifteen minutes. In other words, it was able to answer telephones for all but fifteen minutes during a ninety-day period -- it was operational 99.99 percent of the time. Delphi's goal for the Delta II is fifteen minutes per year.

INSURANCE SOFTWARE -- THE NEW GENERATION

What will be the characteristics of this new product? What can we design today so that five or ten years from now our needs are met? There is no point in discussing the details of a hypothetical system, so I will limit myself to an overview of some of the fundamental ideas that would go into a brand new system and some of its resulting features.

A major design goal is to support different kinds of products and provide flexibility for the future. Rather than support specific products, specific features are supported which can then be used by many different products. Examples of features are:

- flexible premiums.
- deduction for mortality charges.
- rates within one plan that vary by risk classification, say smoker and non-smoker.
- continuous benefit modification, based on a cost-of-living index, asset index, or amortization schedule.
- modular expense charges that can be freely combined.

A company may then choose from among many different supported features to design a new product. Certain new products might also require new features, so the system architecture is designed to allow for easy addition of new features.

One strategy used in software development projects is to make the system as independent of its operating environment as is feasible. This new system will be independent of any data base management system, teleprocessing monitor, and even the type of terminals in use.

It is not a batch-oriented system, it is on-line oriented. Terminals will be used for inquiry, data entry, and to initiate processing. Although most existing on-line insurance systems allow you to enter a transaction on-line, the transaction is then stored for a nightly batch. In the new system transactions will execute immediately. A major policy change that requires three or four iterations does not have to wait for three or four nightly batches; it can be done all in one day, since changes are made immediately.

Most of the transactions that go on in the insurance business are periodic transactions such as billing and premium charges. Different categories of these periodic transactions can be scheduled at specific times for automatic processing. Automatic transactions will be processed when time is available -- they are said to be interleaved with on-line transactions. Of course, higher priority is given to the on-line transactions.

Because of the approach taken with on-line and automatic transactions, there is a departure from the way processing has traditionally been done in the insurance industry. Most systems revolve around the so-called "iron cycle": each night the system reads through every policy and processes each category of activity -- e.g., policy changes, cash processing, anniversary processing, billing, etc. -- in a certain predefined order. The "iron cycle" is a very rigid and expensive way of performing daily processing since it involves looking at every policy every night. With both on-line and interleaved automatic processing, the new system can obsolete the "iron cycle".

It emphasizes management reporting; it is meant to be a source of management information, a management tool. Various reports will provide information about product performance, agent performance, costs of administration, mortality experience, and so on. In spite of this, it is designed to be a paperless administrative system. Instead of printing out reams of paper, the new system will send notices, messages, reminders, and even reports to user queues for later display at terminals. Such data can also be printed out, either instead of or in addition to terminal displays.

No system can solve all the future needs of insurance companies. The way products are changing today, it is difficult even to keep up with current products. Because of this, the new system's architecture provides a framework that is easily enhanceable and customizable. For example, it uses all new data structures. The data will be organized into about two dozen data bases, of which three are the most important: The Plan Data Base, the Policy Data Base, and the Person Data Base. It also provides hundreds of "user exits". A user exit consists of a call to a dummy program which the user can easily replace with a meaningful program. As already mentioned, it is feature-oriented rather than product-oriented, facilitating the addition of new features to support new products.

It is oriented toward the end users; during development special attention is being paid to input from user departments. For example, extensive use will be made of menu screens -- these allow the user to make a selection by pressing a single key. Processing is either automatically performed, or the user is shown another menu screen with a new set of choices.

It is person-oriented. For example, a loan transaction can be performed without the user entering a policy number; only the person's name is necessary. This will also allow a transaction to apply to multiple policies if the person has more than one policy.

Use will be made of distributed processing. Agents in the field will be able to receive messages and proposals from the central computer and can inquire into its data bases. Regional centers with complete administrative capabilities could also be tied in. There will be support for on-line illustrations. Proposals can be requested on-line and immediately displayed at the terminal or printed out.

Automated correspondence will be an important feature, allowing personalized letters to be automatically sent to the policyholder. The letters consist of policyholder specific data which is merged into standard and optional text. The same correspondence features are used for follow-up notices and other internal communications. Correspondence may be generated automatically as a result of certain transactions or it may be requested by a user.

Many functions that are not automated by systems such as CFO and ALIS will be automated. Death claims will be handled from the time notice is received until benefits are paid. The new system also supports the pay-out of disability benefits and annuity benefits, and provides for the necessary security and recurring approval of benefits. It will automate conversions and reinstatements, as well as reentry, reversion, or exchange, or whatever you choose to call this new provision provided by certain term policies.

A common frustration with current systems is the long time needed to set up a new plan, even if no modifications to the current system are required. With the new system, once all the decisions have been made and the plan's rates have been determined, you will be able to set up a new plan in a matter of minutes. There will be flexibility in rate determination, with premiums, dividends, cash values, and reserves calculated as needed, obtained from a Rate Data Base, or stored with the policy.

Systems such as CFO and ALIS do not allow certain combinations of plans on the same policy. Great strides have been taken to eliminate such restrictions. In the new system, a policy can include the equivalent of many CFO or ALIS policies; it can have an unlimited number of riders. As might be expected, the new system will support all the products that have been supported by prior systems. This includes the traditional life, term, and endowment plans and their traditional riders. And of course, it will support the currently emerging products, universal life, flexible premium annuities, adjustable life, and retired life reserves. Health insurance, disability insurance, traditional annuities, and supplementary contracts will also be supported.

In summary, we can look forward to a greater number of advancements in both hardware and software over the coming years. As with all changes, it will take time for the information to be disseminated and for companies to take advantage of those changes. By 1990, many companies will be using hardware and software that doesn't exist today; but a surprising number of companies will just be starting to take advantage of 1981's advancements.

MR. LOUIS MARIE POMMAINVILLE: Would you not use any magnetic tape at all? How would you keep your backup?

MR. ATKINSON: All data is meant to be on-line, but every change to the data base will be recorded on a log tape. Also every week or so, all the data bases can be backed up on tape.

MR. CHRISTOPHER H. WAIN: The systems that we have had described to us seem geared to a non-participating type of operation; I wonder if any of the panelists can comment on the implications of these products and systems for participating business. Could these systems, with modest expansion, maintain individual asset shares for each policy or compute a three-factor dividend?

MR. HASSELMEIER: Some of the non-traditional products that I discussed are indeed participating products, in particular the College Life and Minnesota Mutual adjustable life products, as well as the Bankers Life product. These do require the ability to compute dividends on a unique basis for each policy. The same is true, by the way, of the flexible term product offered by Lutheran Mutual.

MR. MALTBY: A lot of the newer systems have a much more flexible architecture and record structure. You can expand the data base pretty well at will (within some constraints) to accommodate the keeping of funds on an individual policy basis. In the traditional, existing policy systems this is a little harder, but we have had to do it for a couple of our special products which involve funds that are maintained on an individual policy basis -- in one of them we use the cash value field in a reasonably unique way. But the newer breed of systems allows you to do this after issue if you have to, so you can change horses in mid-stream -- sometimes it will cost you, but the architecture is such that it is possible, and if you do it right, you do not have a patch job.