A CRAY RESEARCH, INC. PUBLICATION

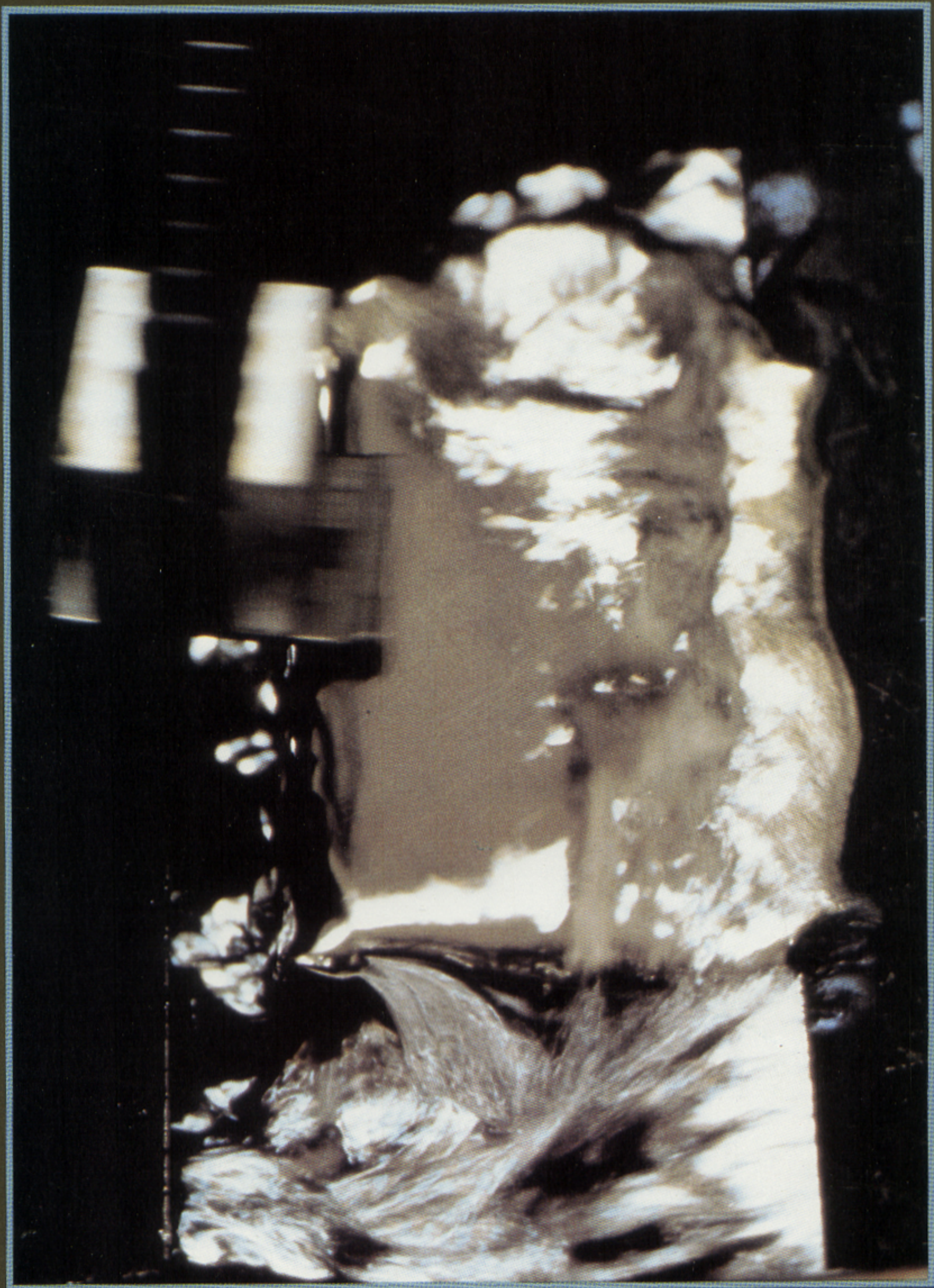
CRAY CHANNELS

Spring 1986

ANNOUNCEMENT!
Introducing the UNICOS operating system

FEATURE ARTICLES:

DEPARTMENTS:

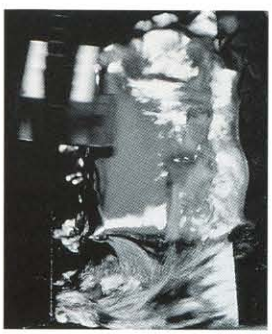


IN THIS ISSUE

Although the past few years have been a volatile time for the petroleum industry, the industry's demand for Cray computers has remained high. Industry leaders know that Cray computers provide the reliable processing capacity needed for fast and detailed exploration and production work. Cray systems used for seismic data processing and geophysical modeling help managers and engineers make the best drilling decisions. And potentially valuable exploration techniques that were previously impractical are now being re-examined, in light of the processing power Cray systems provide.

In this issue of CRAY CHANNELS we examine several applications of Cray computers in petroleum exploration. We also look at the potential uses of supercomputers in economic modeling, as well as current work in computer graphics and environmental risk assessment. Cray Research also announces the release of UNICOS, a new operating system for Cray computers.

The petroleum industry is a major market for supercomputers, and one that benefits manyfold from their use. New techniques that further exploit the special capabilities of supercomputers ensure that high-speed computing will continue to play an important role in geophysical exploration.

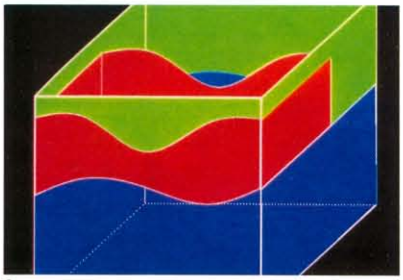


On the cover is an automatic tinning pot used to coat the leads of MOS memory chips. The gold leads on the chips are coated with molten solder, or tinned, before the chips are permanently soldered to memory modules. This ensures that the chips bond securely to the modules.

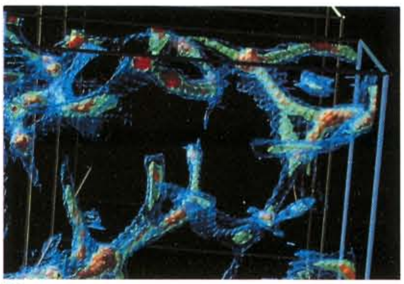
CRAY CHANNELS

A Cray Research, Inc. publication

Spring 1986



2



33

FEATURES

DEPARTMENTS

CRAY CHANNELS is a quarterly publication of Cray Research, Inc. intended for users of Cray computer systems and others interested in the company and its products. Please mail subscription requests, feature story ideas and news items to CRAY CHANNELS at Cray Research, Inc., 608 Second Avenue South, Minneapolis, MN 55402

Volume 8, Number 1
Spring 1986

Editorial Staff
Tina Bonetti
Ken Jopp
Carol Smith
Jack Thompson

Design
Cindy Erickson
Eric Hanson

Multitasking strategies for large-scale problems

Efficiently deploying multiple CPUs to execute a single problem — multitasking — is necessary for taking full advantage of multiprocessor computer systems. But doing so requires some ingenuity on the programmer's part. When multitasking, data storage and I/O management strategies are among the considerations that must be well thought out to maximize CPU usage. Effectively addressing these issues can improve performance enough to solve previously intractable problems.

An example of a particularly efficient use of multitasking, which runs 99 percent in parallel on a four-processor CRAY X-MP/48, is an acoustic forward modeling program based on the Fast Fourier Tranform (FFT) method. The program was developed by Dan Kosloff of the University of Tel Aviv in Israel with assistance from colleague Moshe Reshef, also of the University of Tel Aviv. It was recently use to solve a three-dimensional forward modeling problem of unprecedented size, representing a cube with 256 grid points per edge.

The seismic problem

Forward modeling is a computational method that complements seismic data processing and interpretation. Whereas seismic interpretation attempts to obtain a picture of the subsurface from field data gathered by reflection seismology, forward modeling predicts the type of field data that would be obtained from a known structure. Forward modeling results can be compared with field data to obtain a more accurate interpretation for a given volume of earth. "Given an interpreted model, forward modeling reveals the field data that the geology would produce," Kosloff explained. "If the results of forward modeling agree with the actual field data, chances are very good that the interpreted model is an accurate one."

This valuable technique has not been fully exploited by petroleum and geophysical contracting companies because the computational problem was intractable until the development of the CRAY X-MP/48 computer system and the Solid-state Storage Device (SSD). Input/output, in particular, has been a major obstacle to performing the calculations in a reasonable time frame. However, recent computation using Kosloff's model demonstrates the practical feasibility of solving large forward modeling problems. The computation was successfully carried out by exploiting the high performance capabilities of the CRAY X-MP/48 with a 128-million-word SSD and the multitasking and I/O handling expertise of Cray Research analyst Chris Hsiung and Cray Research geophysicist Mickey Edwards.

The methodologies

Several types of forward modeling methodologies exist. All of them rely on solving the wave equation, which relates the pressure of sound waves to the velocity and density of the material through which they pass. One class of methods is based on known closed-form solutions of the wave equation. These methods are completely accurate and often are very fast, but they are limited because closed-form solutions of the wave equation are known only for simple structures such as horizontally uniform ones.

Ray tracing methods are also used to solve the wave equation. These methods are based on the assumption, taken from geometric optics, that seismic energy travels along rays from its source to its receiver. Ray tracing involves tracing the rays for each event, such as a reflection from a given layer, and calculating the travel time and approximate amplitude of each event. Ray tracing methods are fast and therefore have been the only ones used for three-dimensional simulations until recently. However, the methods can suffer from inaccurate results and are often ineffective when modeling complicated structures.

The third class of methods uses a spatial grid to approximate the geological structure and solves the wave equation numerically to obtain values of the variables at the grid points. The input for a calculation includes values of the velocities of the pressure and shear waves at the grid points and the density and specification of the source.

Methods of this type are potentially very accurate but require extensive computing time. Using a coarse grid limits the high-frequency resolution, but improving resolution increases the required number of calculations dramatically. In two dimensions, a factor of two increase in resolution involves a factor of eight increase in the number of required calculations. In three dimensions, an increase in resolution of a factor of two results in a factor of sixteen increase in the number of required calculations. "The computational intensity of these methods has so far restricted their practical application to two-dimensional cases," notes Kosloff. "Only now with the four-processor CRAY X-MP system have we been able to perform realistic three-dimensional simulations."

The problem discussed in this article is an acoustic forward modeling problem that accounts for pressure waves only. Kosloff and Cray Research analysts have also solved three-dimensional elastic forward modeling problems, which account for pressure and shear waves, but for volumes smaller than the 256-cubed case discussed here.

The multitasking solution

Kosloff's algorithm for numerically solving the wave equation is a "global" scheme based on the FFT method for approximating the spatial derivatives in the wave equation. To most efficiently use the processing capacity of the CRAY X-MP/48 computer system, the problem was vectorized along the systems of FFTs. Sixty-four parallel FFTs were processed together each run, the optimum number for the CRAY X-MP computer's vector length of 64. While lower-level parallelism is exploited through vectorization, higher-level parallelism existing across independent plane sections presents another opportunity to attack the problem. This opportunity is never addressed through vector processing. However, this higher-level parallelism is fully exploited by integrating individual planes independently on the different processors of a CRAY X-MP multiprocessor computer system.

Multitasked for four CPUs, Kosloff's model recently ran at a sustained processing rate of over 513 million floating-point operations per second (MFLOPS) on a CRAY X-MP/48 at Cray Research's computer center in Mendota Heights, Minnesota. This run solved the forward modeling problem for a cubic grid with 256 nodes per edge, for a total of 16,777,216 nodes. For each node, five data values (density, velocity, pressure, rate of pressure change, and temporary data) were stored, for a total required storage capacity of over 83 million words. These data were stored on a 128-million word SSD.

Handling such large amounts of data without becoming hopelessly bogged down in I/O was a challenge successfully met by addressing three fundamental I/O management issues: how to store the data, how to access the data, and when to access the data.

Storing the data

The natural order in which to store three-dimensional data is in the order of (z,x,y); that is, to designate the z coordinate, the x coordinate, and the y coordinate, respectively. Because this scheme orders data in zx planes consecutively, an entire zx plane can be read with a single read statement. However, when data are structured in this way, the data in zy planes are not consecutive, and reading data in a zy plane requires a separate read for each line in the plane. In other words, each zy plane in the model would require 256 reads, seriously inflating I/O access time.

Because each SSD access request takes 0.4 milliseconds (msec) and processing one plane of the 256-node-per-edge cube would require 256 access requests, each plane would require 128 msec to access. By reading eight planes at a time, the total cube might be processed 32 times per time step, resulting in 3277 msec, or 3.3 seconds, of access request time per time step. A total I/O system overhead of 6.6 seconds is encountered per time step by adding in the write request. Multiply this by the 1000 time steps required to run a complete model, and one incurs an access request penalty of almost 2 hours!

This inefficient scheme was avoided by structuring the data in a novel way. The cubic database was divided into 8 x 8 x 256-node "pencils" and the data were stored in the *zxy* order *within each pencil*. Structuring data this way enabled Kosloff and Cray Research analysts to eliminate much of the I/O access time associated with the natural ordering. Although this storage strategy requires occasional data transposition, the penalty appears negligible.

The advantage of this method is that it allowed successive pencils to be read with a single read, such that the *zx* plane could still be read with one read, but the *zy* plane required only 32 reads instead of 256 (because each pencil was eight lines deep and 256/8 = 32). As a result, system overhead was reduced to one-eighth of what it would have been using the natural order. Because this method requires transposing data coordinates, or "rotating" the pencils, to return to the natural order, data handling is a delicate operation. But for very large problems, the time savings make the effort and extra caution worthwhile.

As an alternative, the user could simply allow the operating system to schedule an idle processor to handle I/O requests (if an idle processor is available). However, computational efficiency would be compromised because the maximum number of CPUs assigned to computation would be reduced from four to three.

Accessing the data: How

The I/O system overhead was further reduced through the use of a new I/O utility called (asynchronous) queued I/O. This utility allows the user to queue I/O requests together, instead of initiating each request separately, and thereby significantly reduces system overhead.

The overhead penalty for accessing the 256-cubed model without queued I/O totalled over 819 seconds for the SSD. This is because, in the *xy* plane, each call incurs 400 microseconds (μsec) of overhead, the plane spans 32 pencils per edge, and each time step reads the entire 32 x 32-pencil plane. Thus, the overhead for each timestep is 400 μsec x 32 x 32 x 2, or over 819 msec. This multiplied by 1000 time steps adds over 819 seconds of system overhead for the entire model.

However, queuing I/O reduces the overhead dramatically. Although queued I/O results in an overhead of 400 μsec for the first request, each subsequent request incurs only 50 μsec of overhead. Applied to the 256-cubed forward modeling problem, a run implementing queued I/O required 60 percent less wall-clock time than a run without queued I/O. (For more information on queued I/O, see this issue's related articles.)

Accessing the data: When

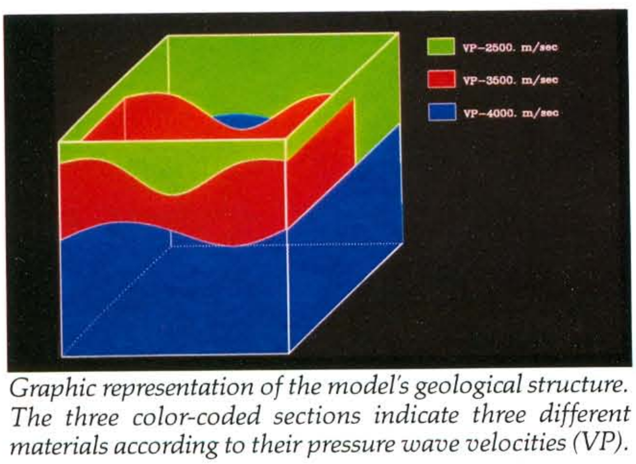
Along with the preceding time-saving strategies, double buffering was used to access data during CPU computation. Double buffering involves transferring a block of data from the SSD to memory at the same time the data from the previous transfer are being processed in a CPU. This method of overlapping I/O transfer and CPU computation completely eliminated I/O wait time, a goal that would have been impossible to realize without the SSD. I/O transfer from conventional disk storage would have been about 500 times slower than it was from the SSD.

"We were able to overlap I/O transfer and CPU computation completely with all four CPUs engaged," said Chris Hsiung, who devised the program's multitasking scheme and I/O strategies. "Each I/O transfer involved 2.5 million words, but by using double buffering there was never a wait for a transfer. The speed of the SSD was crucial to masking I/O in this way. Transfers that totaled about 11 minutes using the SSD would have taken over 42 hours using the DD-49 disk drive, Cray Research's fastest disk."

Double duty for data

Along with increasing I/O speed, Kosloff's forward modeling code was further structured by Cray Research analysts to minimize the total amount of I/O. This was achieved by having each CPU do more useful work on each subset of the data while the subset was in memory. Data from each spatial direction were retained in memory long enough for a CPU to perform calculations on them for two time steps.

Each time step requires two sweeps through the data — one in each of the directions that define the plane being processed (*xy* or *yz* planes). Thus, processing a *zx* plane requires the calculation of spatial derivatives by FFTs in the *z* and *x* directions. This is followed by a sweep through the *yz* plane for the calculation of derivatives in the *y* direction, and a time integration. But rather than repeat this order in a cyclic fashion for each time step, data from the last direction to reside in memory, in this case the *y* direction, are retained for the next time step. Thus, the next time step would first calculate the derivatives of the *y* direction and then the *z* direction. After all the *zy* planes are handled, *zx* planes are read in, and the new iteration is done by processing the *x* direction followed by a time integration. The data from the *x* direction would then be retained in memory for the following time step, and so forth. By processing data in this "zig-zag" fashion, the amount of required I/O is effectively halved, since the last part of each time step plays a dual role, being also the first part of the following time step.



Graphic representation of the model's geological structure. The three color-coded sections indicate three different materials according to their pressure wave velocities (VP).

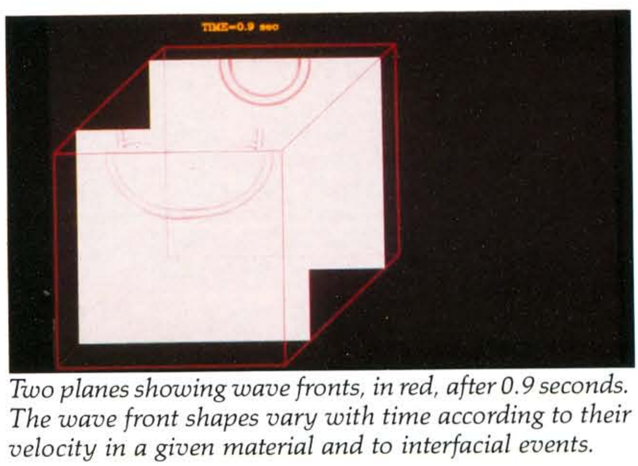
A new exploration tool

The several data-handling strategies described here were successfully combined to make practical the solution of a previously intractable problem: the direct numerical solution of a 256-cubed forward acoustic model. Solving such a problem was unimaginable until recently, but the advent of Cray multiprocessor systems and the ingenuity of dedicated scientists and programmers have expanded the boundaries of the possible.

Acoustic forward modeling on large data sets and elastic forward modeling on smaller data sets are becoming practical tools for geophysical exploration. As these tools come into popular use in the petroleum and geophysical exploration industries, and are combined with traditional data processing methods, drilling decisions will entail less risk than they do now. And as computational methods continue to be refined and ever-larger problems targeted for solution, Cray computer systems will remain one of the petroleum industry's most valuable and cost-effective exploration tools. □

Acknowledgment

Special thanks to Dan Kosloff and Moshe Reshef of the University of Tel Aviv and to Chris Hsiung and Mickey Edwards of Cray Research for their help in preparing this article.



Two planes showing wave fronts, in red, after 0.9 seconds. The wave front shapes vary with time according to their velocity in a given material and to interfacial events.

Seismic inversion
a "new" tool for exploration seismology

A. J. Berkhout, Delft University of Technology, The Netherlands
Don Larson, GeoQuest International, Houston, Texas

Exploration seismology involves artificially generating seismic energy at the Earth's surface, which then propagates into the subsurface and encounters discontinuities at boundaries between rock layers. The discontinuities partially reflect the energy back to the surface where the vibrations are recorded. As the amplitude of the vibrations varies, it provides information about the length of time required for sound to travel to the reflecting rock interface and back. The "character" of the vibrations contains information about the rocks forming the interface from which the energy was reflected.

Complex processing techniques are required to render this seismic survey data useful. Seismic inversion is a technique with great potential for improving the accuracy of exploration seismology results. But the technique has been largely ignored due to the computational resources it requires. Supercomputers such as Cray systems, however, now make it practical to explore and implement seismic inversion techniques.

Interpretation and inversion

After processing a recorded set of seismic survey data to maximize its resolution and signal-to-noise ratio, geophysicists and geologists analyze the data to derive information about the geologic layers. This analysis is called *seismic interpretation*. Broadly speaking, the information sought fits into two categories, *structural* and *lithologic*. Structural information relates to the location and orientation of interfaces between rock layers, faults, and folds. Lithologic information relates to acoustic and elastic properties of rocks.

The properties of the recorded waves are determined by two properties of the ground through which they traveled: the *propagation properties*, which depend on such "macro" elastic parameters as the average velocity and the average absorption, and the *reflectivity properties*, which depend on local variations in elastic parameters such as velocity and density. The macro parameters and the detail parameters together provide the full solution to the seismic inversion problem (Figure 1).

To analyze properly the reflectivity properties of seismic survey data, it is first necessary to remove the distortion caused by propagation of the seismic energy through rock layers above the reflecting interfaces. Inversion to remove these propagation effects requires the specification of a subsurface geologic model that contains macro features such as the location of the main velocity and density boundaries, average velocity and density gradients in each

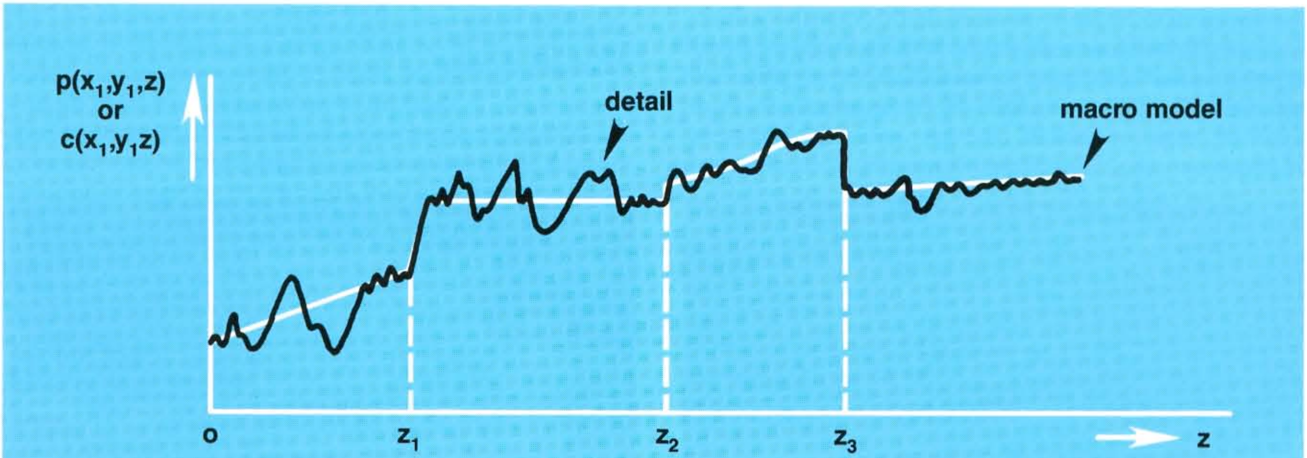


Figure 1. Propagation effects are determined by the average properties of the geologic layers while reflectivity is determined by the detailed properties.

main layer in the model, and the effective absorption parameter of each main layer.

Such a model is referred to as a *macro subsurface model*. The amplitude and travel time from the many separate physical experiments (measurements) made in the field should be used in the inversion process that determines the parameters of this macro subsurface model. Until effective parametric inversion algorithms for accurate determination of the macro subsurface model are developed and perfected, one of the most useful methods for determining the macro subsurface model is the use of an inversion algorithm to test successive models iteratively. This approach is described later in this article.

Inversion to obtain detail can be carried out by either seismic migration or inverse scattering. Both processes use a macro model to invert the propagation operators to eliminate propagation effects between the surface and the level of interest in the subsurface. In inverse scattering this model is known as the *reference medium* or the *background*. A homogeneous reference medium is not appropriate for seismic inversion. This inversion must be carried out in a band-limited way, meaning that the level of detail in the result is limited by the amount of information in the seismic survey data recording.

The output of seismic migration is the distribution of reflectivity in the subsurface. Inverse scattering, however, produces separate velocity and density estimates. Therefore, seismic migration is pre-eminently suited for estimating structural subsurface information (the macro model), and inverse scattering is best suited for estimation of lithologic subsurface information (the detail). Today the seismic industry displays growing interest in inverse scattering techniques.

Inverse scattering should always be applied to multiple measurements that were taken with varying separation between the location of the energy source and the receiver (*multi-offset*). Migration can be applied to data where the source-to-receiver separation is zero (*zero-offset*). Zero-offset data are not measured but they can be simulated by time-adjusting and summing (stacking) the data from multi-offset measurements.

In some geologic situations the "short cut" of stacking the multi-offset data to simulate zero-offset data followed by zero-offset migration fails because the required simplifying assumptions are violated. In such situations, it is necessary to migrate the multi-offset data to zero-offset, then stack the results. This type of *pre-stack migration* is described later in this article.

Macro model derivation

From the preceding discussion, it can be seen that a key requirement for inversion is the acquisition of a good background or macro model of the geologic section. The derivation of such a model is the task of the seismic interpreter.

Several methods exist for making such determinations. One widely used method involves a forward modeling program that generates synthetic seismic data to test model hypotheses. In this method, the interpreter states a hypothesis in the form required by the modeling program and uses the program to generate a synthetic seismic response to the model. These synthetic data are compared to the real seismic data, and if the two data sets match within acceptable limits then the interpretation is a reasonable one and represents a useful macro model.

Difficulties arise when the synthetic data do not match the real data within acceptable limits. The interpreter must then compare a view of the model, which has been distorted by propagation effects, with a view of the real world, which is also distorted but by different propagation effects, and reconcile the two. Significant differences must be mapped back through the still unknown propagation effects into changes in the model hypothesis.

An alternative approach involves using a depth migration to migrate the real data according to the proposed macro model and to remove the propagation effects as defined by the model. If the propagation effects in the data are the same as those described by the macro model, then the events in the migrated result representing velocity interfaces should match the corresponding interfaces in the macro model. When comparing in the depth-migrated domain, it is relatively easy to determine if an acceptable match has been achieved and, if not, what changes to the macro model are needed to improve the match. An additional advantage of working in the depth domain is that constraints based on geologic reasonableness can be applied.

Figure 2 illustrates this migration modeling process. As in forward modeling, an initial model is proposed by the interpreter. Next, the model is used with a depth migration program to migrate the real seismic data. The depth section that results from the migration is compared to the model, which is also in depth, to determine if the seismic events were migrated to the location predicted by the model. If they were, the model is acceptable.

If the events were not migrated to the predicted location, the model must be modified. Intuitively, this simply means that the model must be made to look more like the migrated data, which suggests that the migration result will be more correct than the model (when the model is incorrect). This implies that using the result of each migration as the model for the next migration will make the process converge. The following example demonstrates just such

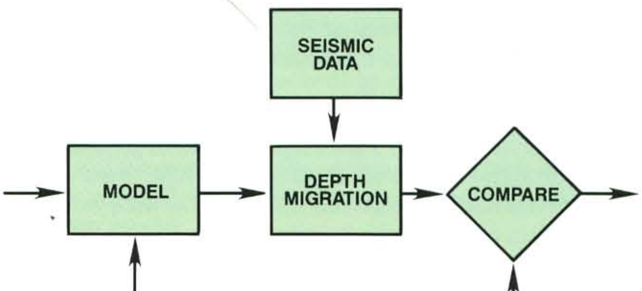


Figure 2. Depth migration is used iteratively to help determine the correct structure for the macro velocity model.

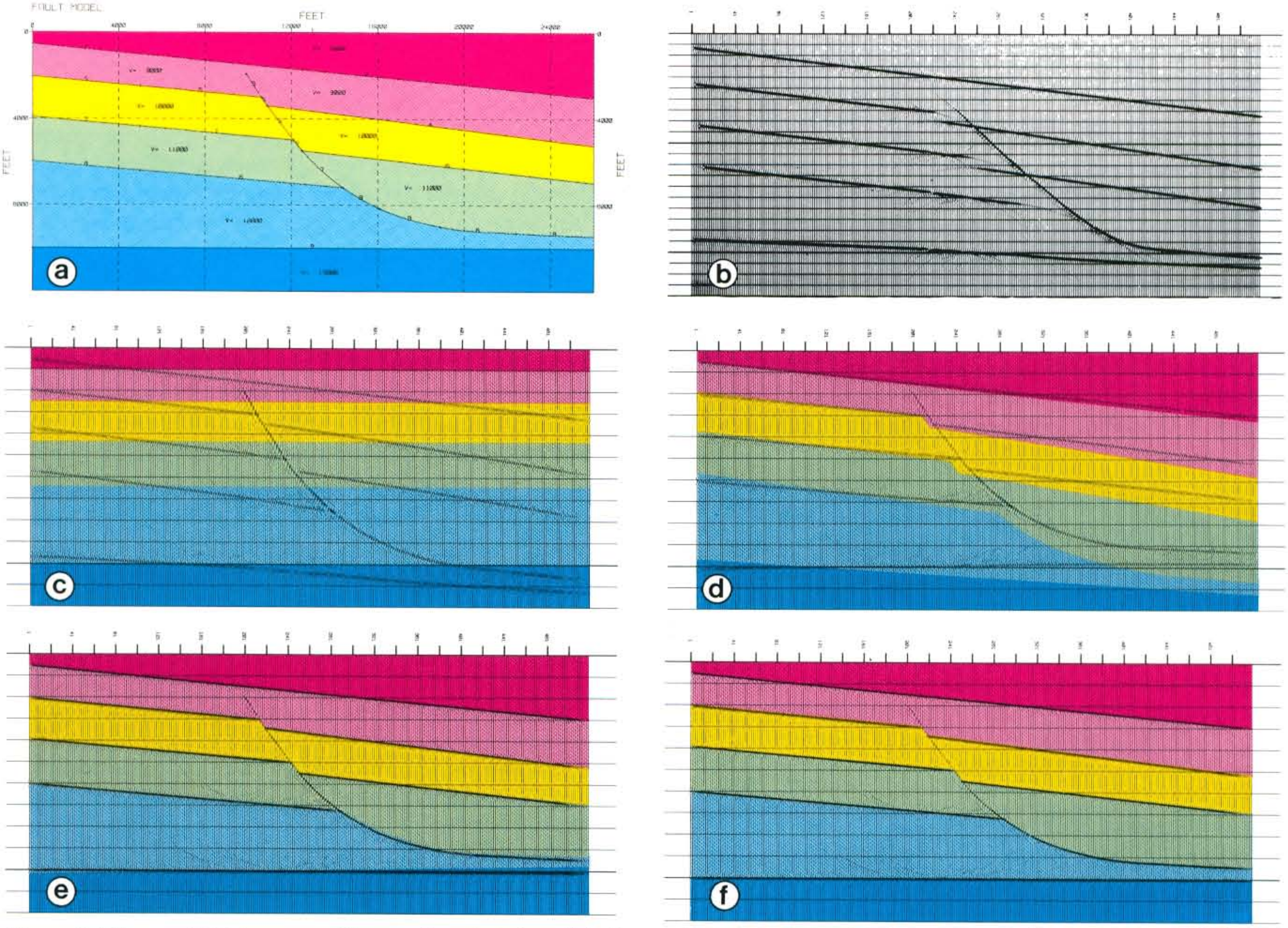


Figure 3. The model (a) was used to produce test data (b). Successive migration with incrementally improved models (c ,d,e, and f) converge on the correct image.

a convergence. It is impossible to prove that the process described always converges, but experience shows that it does if the initial model is not grossly in error. Gross error in the initial model estimate is best avoided by making that estimate a very simple one.

Macro inversion example

A set of test data was generated to illustrate the process of using a depth migration to derive a macro model. The model used is a simple growth fault shown in Figure 3a. In the illustrations used for this example, bands of color represent velocity. The layers of the model used to generate the synthetic data are colored, and the successive macro velocity models used to migrate that test data are superimposed in color over the migration results. The accuracy of the macro velocity models is judged by the extent to which the events (coherent sets of squiggles) in the migrated seismic data conform to the boundaries between different colored bands.

A Kirchoff wave theory algorithm was used to generate the synthetic data shown in Figure 3b. While there is an obvious relationship between the events in the synthetic seismic data and the velocity interfaces in the model, it is also obvious that propagation of energy through the model has produced a distorted image. In particular, the event that corresponds to the fault has been displaced significantly relative to the other events. The correct location of this fault would be crucial in an exploration context.

The velocities used in the macro models in the example were derived from the synthetic seismic data generated from the model. These velocities are within one or two percent of the actual velocities used to generate the synthetic data. This example shows how the migration technique helps determine the proper structures of the macro model.

The initial macro velocity model used for the first migration is structurally very simple and actually conforms to the assumption required by many seismic data processing methods, namely that the layers are all horizontal and of constant thickness. The average velocities in the layers are the ones derived from the synthetic data. This macro velocity model is overlaid on the results obtained by depth migrating the synthetic data with it in Figure 3c. The migrated result looks reasonable, in the sense that the obvious propagation distortions have been removed, but it is obvious that the events do not conform to the macro velocity model. This means that the velocity model must be changed.

The locations of the events in the first depth migrated result were used as the locations of the velocity interfaces in the second macro velocity model. In Figure 3d the second model is shown in color overlaid on the results obtained by depth migrating the data according to it. It is easy to see that the correspondence between the macro model and the migration result is now much improved. The difference between the result and the model is the same as the difference between the second migration result and the first.

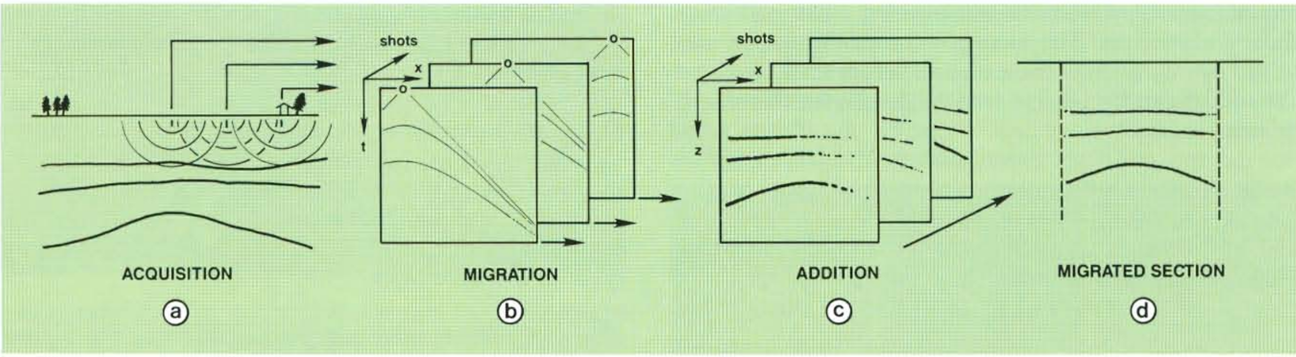


Figure 4. Separate field experiments (a) are recorded (b), migrated (c), and summed (d).

To appreciate the importance of this difference, consider that exploration targets often involve petroleum that has been trapped by a fault. In Figure 3d the trap would be on the right (downthrown) side of the fault. In the units of this model, the first result imaged the fault about 500 feet to the left of where the second migration imaged it. A well drilled on the basis of the first result could easily have been on the wrong side of the fault.

A third macro model defined from the events in the second migrated result is overlaid in color on the third migration result in Figure 3e. After the third iteration, correspondence between the model and the result is quite good. For good measure, a fourth macro model was defined from the third iteration result and used to migrate again. Figure 3f shows the model and result; the correspondence between model and migration result is almost exact.

Although this example is highly simplified, it shows the kind of iterative process actually used to derive macro velocity models through depth migration. In practice, judgment is usually needed to define successive macro velocity models. Also, each new velocity model tends to be more complex and more detailed than its predecessor.

Velocity determination

In the preceding example, velocities were determined from the synthetic seismic data. Analysis of travel times in a structurally simple part of the synthetic section was used to calculate average velocities for the macro model layers. This method can also be used with real data, but often velocities obtained this way are unreliable. Velocity information also can be obtained from well log data, well velocity surveys, geologic knowledge, and published data. The pre-stack migration example that follows can be used to test the accuracy of macro model velocities and to provide objective guidance for the refinement of these velocities.

Pre-stack depth migration

Seismic processing methods that simulate zero-offset data by stacking data from many different offsets have until recently relied on the assumption that all the geologic layers are horizontal and, most important, all the interfaces between layers with different velocities are horizontal. Recent developments in partial pre-stack migration relax these assumptions in certain limited ways, but the requirement for simple geologic structures remains.

Unfortunately, those parts of the Earth's crust with the greatest potential for containing hydrocarbon traps usually violate the condition required for successful common midpoint stacking. The problem arises because stacking the data traces with various source/receiver offset distances first requires removal of the differing propagation effects that result from different travel paths having been followed. The model used for this in common midpoint methods is the *horizontal layer model.*

If common midpoint methods cannot adequately remove the propagation effects to make stacking possible, it becomes necessary to use depth migration to remove those effects. Thus, there is a need to migrate the data so that it can be stacked, hence the term *pre-stack migration.*

Shot record migration

There are many methods available for pre-stack migration. The simplest involves migrating the seismic records as they were gathered (common source records) and to stack them after the propagation effects have been removed from each one. This method has a theoretical advantage in that the data sets are recordings of real physical experiments, not artificial and unrealizable constructs. It also has practical advantages: the records are independent of one another, can be migrated separately, and are each relatively small, thus minimizing the data management problem. Furthermore, with shot record migration it is only necessary to migrate the data needed to obtain the required signal-to-noise ratio in the result.

Figure 4 schematically shows how the method works. The shot records are gathered in the field using a separate experiment for each record as shown in Figure 4a. This shooting yields recordings as shown in Figure 4b. Migration removes propagation effects from the records and produces new records containing images that look like the actual geology as shown in Figure 4c. Finally, the individual images are summed (stacked) to improve the signal-to-noise ratio. In principle, if the macro velocity model used for the shot record migration is correct, then all the migrated images will be the same and will stack constructively. Any differences in the images indicate errors in the velòcity model.

Shot record migration example

This example uses the same model as the previous iterative migration example. Figure 5 shows three synthetic shot records generated from the model using a Kirchoff wave theory algorithm. The example shows the results obtained by migration of these records before stack and how the results can be used to test the quality of the macro velocity model.

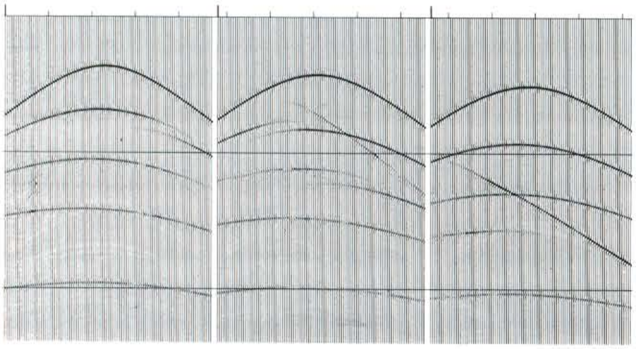


Figure 5. Synthetic common source records were produced from the model in Figure 3.

Figure 6a shows the three records after they have been migrated with a macro velocity model that has the velocity ten percent too high in the second layer from the top. The illustration shows each migrated record overlaid in color by the part of the model that was illuminated by that shot record. It can be seen that the second and deeper events were imaged deeper than the location of the corresponding interfaces in the macro velocity model. From the information available in each single record it is not possible to tell whether the locations of the interfaces were wrong in the model or if the velocities were wrong.

A total of 12 synthetic records were migrated and the traces sorted. This was done in such a way that the traces that would be summed in the stack were gathered together. Figure 6b shows these gathers. One can see that for the second and deeper interfaces the events from the different records did not image at the same depth. The traces near the outside of each gather traveled farther than the inner traces and they were imaged deeper in the section. This means that the velocity was too high in one or more layers above these events. Since the event from the shallowest interface imaged at the same depth within each gather, the velocity error must be below that interface. Therefore, the velocity is too high in the second layer and possibly in deeper layers.

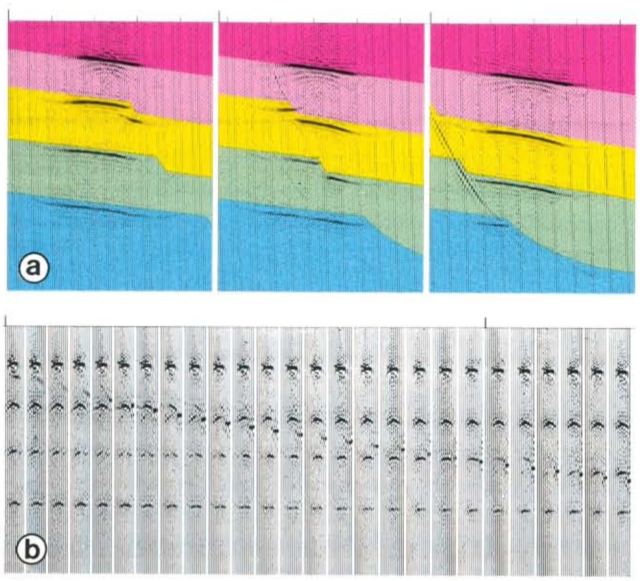
Figure 6c shows the stack of 12 records with the macro velocity model as a color overlay. The fault face in this image does not quite fit with the events that it cuts. There is also some weakness in the event from the third interface to the left of the fault. These are both caused by the velocity error in the third layer.

The shot records were migrated again using the correct velocities in the macro model. The three records shown in Figure 5 are shown again in Figure 7a with the model in color. Now the events in the migrated records fit well with the velocity interfaces in the model. Figure 7b shows the traces from 12 migrated shot records gathered for stacking and it can be seen that all the events were imaged at the same depth in each gather. This confirms the correctness of the model velocities. The stack of all 12 records in Figure 7c shows an improvement over Figure 6c. The position of the fault event is better and the weakness is no longer apparent in the third event.

This example shows that the pre-stack shot record migration can be used to remove propagation effects from separate records when the simplifying assumptions used in common midpoint methods are too restrictive. It can also be used to test the velocities in a macro velocity model when necessary.

The role of supercomputers

The common midpoint method was invented at about the same time digital computers came into use in seismic data processing. It is safe to say that these two developments complemented one another to a remarkable degree. Although originally necessary as a way to combine redundant data to improve signal-to-noise ratios, common midpoint stacking became a necessary technique for reducing the data volume to a size that conventional computers could manage. Pre-stack processing was limited to computationally simple tasks and pre-stack migration was not considered practical.



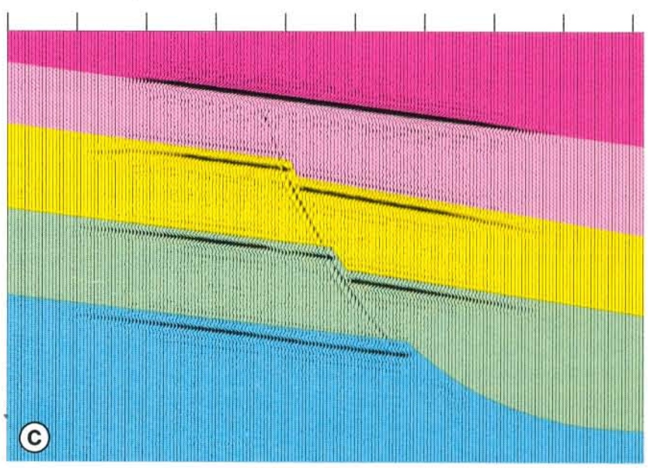


Figure 6. Records migrated with the wrong velocity conform neither to the velocity model (a), nor to one another (b). The stack (c) contains artifacts of the velocity model error.

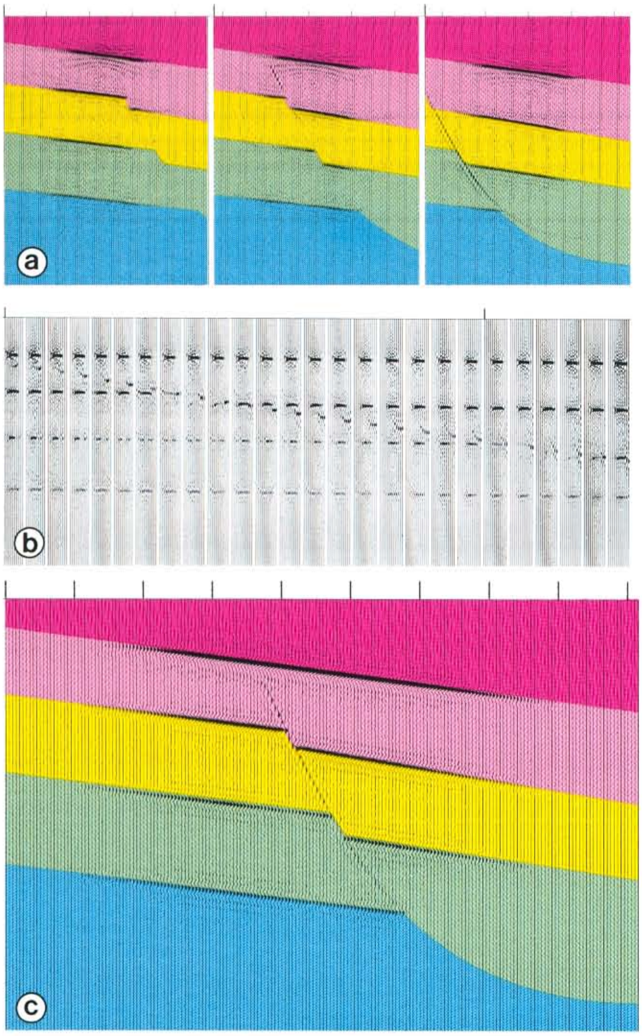


Figure 7. Records migrated with the correct velocity model conform to the model (a) and also to one another (b). The stack (c) is free of artifacts.

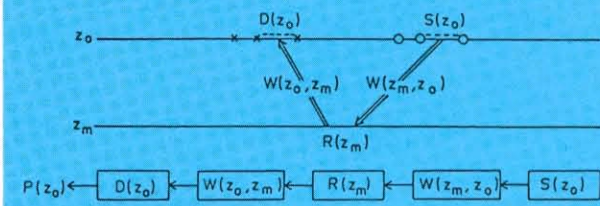
Supercomputers in general, and the CRAY X-MP system in particular, have made it possible to consider pre-stack depth migration as a serious alternative when indicated. As another process that needs to be applied before stacking, inverse scattering algorithms also are gaining attention because the required computers are now available.

The shot record depth migration described in this article was developed and tested several years ago but was shelved because it was impractical to use, given the computer power available at the time. However, experience with the CRAY X-MP system has led to renewed efforts in this area.

The iterative depth migration that was described here as a method to obtain a macro velocity model in preparation for seismic inversion becomes more interesting with the availability of Cray computers. Depth migration, even after stack, is computationally intensive. When it is used iteratively, each iteration can take an entire day. With a CRAY X-MP computer system and a suitable graphics device for result display and interactive model adjustment, it is not unreasonable to contemplate the whole iterative sequence being completed in a single sitting. In practice this means that the interpreter would be able to afford the time needed to obtain a more refined result than would otherwise be practical.

The iterative depth migration is a perfect example of a user being able to ask "What if?" and get a response quickly enough to make the answer interesting. Thus, one can not only solve the same problems faster, but also attempt to do what used to be dismissed as infeasible. It only seems natural that interest in seismic inversion methods would be sparked as supercomputers such as the CRAY X-MP system make it practical to apply such methods. □

A little bit of theory



The figure shows schematically how a seismic wave $S(z_o)$, generated by one or more sources at the surface $z=z_o$, propagates in the subsurface, is reflected by inhomogeneities at depth level z_m and propagates back again to the surface, where it is registered by one or more detectors, indicated by $D(z_o)$, at the surface $z=z_o$. Symbolically,

$$P(z_o) = D(z_o)[E_m W(z_o,z_m) R(z_m) W(z_m,z_o)] S(z_o).$$

The downward propagation operator $W(z_m,z_o)$ describes quantitatively how the downward traveling source wave field changes during propagation. Similarly, the upward propagation operator $W(z_o,z_m)$ describes how the upward traveling reflected wave field changes during propagation. At each depth level z_m the reflectivity operator $R(z_m)$ describes how (a small part of) the downward traveling source wave field is transformed into an upward traveling reflected wave field that is related to the inhomogeneities at depth level z_m. In *simulation* problems $R(z_m)$ is specified for each depth level z_m and the propagation effects of the medium together with the data acquisition properties are "added" to $R(z_m)$ according the above equation, yielding $P(z_o)$. In *inversion* (migration) problems $P(z_o)$ is measured and $R(z_m)$ is computed by compensating for

$$D(z_o)W(z_o,z_m) \text{ and } W(z_m,z_o)S(z_o)$$

at each depth level z_m.

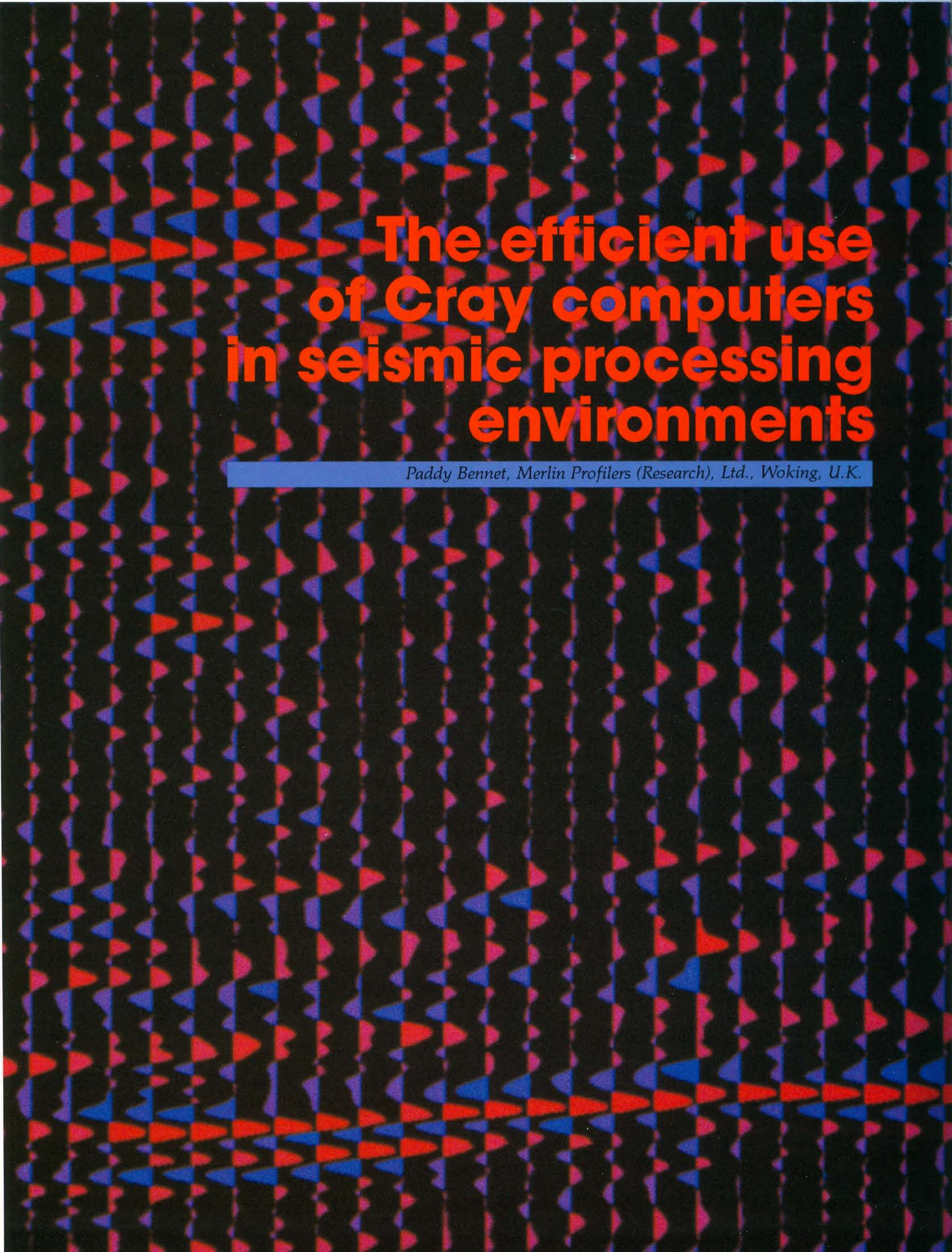
About the authors

A. J. Berkhout received his master of science and doctorate degrees from Delft University of Technology in The Netherlands. He was employed by the Shell Petroleum Company for several years and currently is professor of acoustics at Delft University. His research interests include signal processing, wave theory, and acoustics.

Don Larson received his B.S. degree in applied mathematics from Brown University. He is currently vice president responsible for software development at GeoQuest International. Larson's professional interests include seismic modeling and imaging.

The efficient use of Cray computers in seismic processing environments

Paddy Bennet, Merlin Profilers (Research), Ltd., Woking, U.K.



The key to effective use of available computer hardware lies not only in minimizing turnaround time for a particular task but also in maximizing throughput for several jobs. The nature of seismic data processing has made this task a challenge for even the largest computers. Computation, disk, and tape work must be balanced so that no job is hampered by others in the system. At the same time, the system's total resources must be used effectively.

This article describes field experiences using four standard Merlin Profilers' benchmark jobs run on several different computers, including the CRAY-1/S system at Merlin's Woking, U.K. processing center and the CRAY X-MP/48 system at Cray Research's computer center in Mendota Heights, Minnesota. The jobs are part of Merlin Profilers' Seismic Kernel System (SKS) portable software package, which comprises land and marine seismic software, including time migration and F-K filtering.

SKS software development started in April 1982 when Merlin Geophysical (as it was then called) took delivery of a Univac 1100/62 system with two FPS AP190L array processors. When the company needed to further upgrade its processing capacity, the portability of the software package made it possible to choose a different hardware system. A CRAY-1/S system was chosen and was installed in August 1984. It was in full production use one month later. The same basic software package is supported on several other manufacturers' computers as well. The combination of SKS software and Cray Research hardware proved so successful that Merlin Profilers has recently ordered a second Cray system, a CRAY X-MP/24, with delivery scheduled for late in 1986.

Interesting results were obtained during a demonstration run at Mendota Heights. They show that with modern software design, one can easily make full, efficient use of the most powerful state-of-the-art computers, even with inherently disk-oriented processing.

The demonstration job mix

The four jobs in our test were part of Merlin Profilers' suite of benchmarking and demonstration jobs, used at all SKS installations to test new systems and to measure relative performance of hardware or software configurations. The jobs are two identical copies of a job called "BM7" and one copy each of "BM8" and "BM80." The input data were from one of Merlin Profilers' non-exclusive surveys. The software used on the CRAY X-MP/48 system was the same as that used on our production CRAY-1/S system with minor changes to the disk allocation subroutines. Following are descriptions of the jobs.

BM7

BM7 is a basic stack job. The input data set is common depth point (CDP) gathered, 60-fold, 4-millisecond (msec) sample rate, with a 7-second record length; 240 CDPs are processed. The processing consists of deconvolution (a two-zone predictive deconvolution with a 240-msec operator length), normal move-out correction, trace balancing, exponential gain compensation, mute, and stack. After-stack spherical divergence amplitude compensation is performed and followed by a filter and automatic gain control. Plots are produced of the stacked data and of every 10th CDP prior to stack. For convenience, no output tape was generated, but one could be generated at any point in the processing sequence.

The combination of SKS software and Cray Research hardware proved so successful that Merlin Profilers has recently ordered a second Cray system, a CRAY X-MP/24 . . . .

The processing sequence involves very little disk work, with only the two trace plot processors using disk files. Consequently, during most of this job's duration, the factors limiting processing speed are CPU performance and tape speed. The input data tapes are recorded at 6250 bits/inch (bpi) and are read on 200-inch/sec (ips) tape drives, minimizing the tape read time. The method used by Cray Research for interfacing the tape drives to the system is such that the tape reading is effectively performed asychronously, though the programmer does not need to know this. Thus the tape read time is overlapped with computation (if sufficient computation is being performed). In this example, the times to read a trace and to process a single input trace are approximately equal, resulting in a job that stacks data at tape speed and has a high CPU utilization when running a single job in a single CPU.

BM8

BM8 is a basic CDP gather job. The input data set is shot ordered, with 60 traces per record, a 4-msec sample rate and a 7-second record length. There are 300 shot files, amounting to a full 2400-foot tape recorded at 6250 bpi. The input traces are first muted. This is followed by exponential gain control and trace balancing. Then an F-K domain coherency filter is applied to the common source, or shot, files to remove coherent noise. After that, the geometry is defined and the traces are gathered into CDP order. A bulk static shift is applied to compensate for the shot and cable depths. At this point the data would normally be sent to tape. Plots are produced of the near trace of every CDP and of every 50th CDP.

This job has quite a lot of disk work in it, apart from that involved in producing the trace plots. The F-K coherency filter processor is written to use disk files to hold a complete shot record if there is not sufficient room in the array spaces given to the subroutines. On many systems, the memory size limit would be modified to take advantage of more space and thus no use would be made of the disk, but Merlin's CRAY-1/S computer is a one-million-word system, so memory space is at a premium. To allow a better comparison between the two machines, this parameter was not changed, so only two million words of memory were used on the CRAY X-MP/48 computer by four concurrent jobs. The disk file used by this processor, however, was changed to reside in the fast 128-million-word Solid-state Storage Device (SSD) connected to the CRAY

X-MP/48, using only about a half million words of memory. On our CRAY-1/S computer, we specified that the file resides in the one-million-word solid-state buffer memory which, although it is considerably slower than the SSD, is considerably faster than the disk drives. Consequently, the F-K coherency processor showed very little overhead from the disk I/O performed within it. The CDP gather processor uses a disk file that is much larger than that used for the F-K processing. No attempt was made during this demonstration to use the SSD for this file. (The SSD was used in a later demonstration when the whole of this job was run in 3 minutes, 12 seconds — only slightly slower than the tape reading time.)

On a system such as the CRAY X-MP/24 or CRAY X-MP/48 with fast disks and an SSD, it is apparently not necessary to be so careful about the job mix.

This job is much less CPU-intensive than BM7, but the CPU utilization runs in bursts while it performs the F-K filtering. However, a comparison of CPU time against total elapsed time shows a very good average CPU utilization.

BM80

This job is similar to BM8, except that velocity spectrum analysis is performed and the results plotted at five locations after the CDP gather. CPU utilization is therefore similar to that of BM8, with periods of intense CPU activity and also periods of intense disk activity.

The mix of jobs

As previously mentioned, two copies of BM7 were run, using identical copies of the original data tape. One copy each of BM8 and BM80 were also run. No software changes were made to reflect that the jobs were running on a multiprocessor system or that (in effect) only one was run per CPU. One feature used to prevent jobs from competing with other jobs in the system was a disk allocation plan whereby each job has a disk pack (or two) allocated purely for its own use. Since the jobs did not overflow the disk pack allocated, there was no disk head contention between jobs. Disk contention between parts of one job was minimized by the sofware architecture of SKS. On Merlin Profilers' single-processor CRAY-1/S computer (which has DD-29 disks rather than the DD-39 or DD-49 disks configured on many CRAY X-MP systems), we normally try to run two jobs together; ideally, one will be CPU-intensive (such as BM7) and the other will be disk-intensive (such as BM8). On a system such as the CRAY X-MP/24 or CRAY X-MP/48 with fast disks and an SSD, it is apparently not necessary to be so careful about the job mix.

Efficiency of the demonstration mix

With only one job per CPU and very low system overhead, one might not expect particularly high CPU utilization figures. However, we were able, without any special effort, to produce a "snap" of the CPU monitor (a view of CPU status at a given instant) while the four jobs were running. The snap showed exceptionally high CPU utilization (Figure 1). Most of the system overhead shown is attributable to controlling the I/O requested directly by these jobs. Although this is a snap of a short time window, the long-term utilization was also high, showing that a high degree of efficiency was obtained in the I/O work. The software design made this easier to attain, allowing the geophysicist to specify a large amount of processing alongside the I/O-bound processors. With the system properly loaded (about eight jobs on the CRAY X-MP/48 system) we would then achieve almost 100 percent CPU utilization for the duration of the test jobs.

Vectorization

So far, no mention has been made of vectorization and the problems that can occur while achieving it. System efficiency has also been equated with high overall percentage CPU utilization. This assumes the actual computation work is performed efficiently and is properly vectorized, because otherwise it would be very easy to use up all the CPU time.

SKS was originally designed to be very modular. In particular, most of the computation work is performed by a set of subroutines called the Device-Independent Vector Arithmetic Package (DIVAP). This was first developed on a system with array processors and a FORTRAN version was written later. It was therefore very easy to ensure that the DIVAP routines run efficiently. In some cases very minor changes were made, such as the insertion of compiler directives to aid vectorization. In other cases, routines were rewritten to call subroutines from the Cray SCILIB or BENCHLIB libraries. In only a few cases did we need to spend any length of time vectorizing the code. As a result of this structure we have a computation system that is close to the maximum efficiency while still allowing ease of software development and maintenance.

Ideal system for seismic processing

There is no absolute formula describing the ideal Cray system for seismic processing using Merlin Profilers' SKS software because we have not fully loaded a CRAY X-MP/48 system. The thoughts presented below are extrapolations from the available data considering the hardware and software capabilities.

The advent of faster storage devices has effectively redressed the balance between computation speed and I/O capability.

The amount of memory used by a single job varies according to the complexity of the processing sequence. With SKS, very complicated processing sequences may be put together without the need for intermediate tapes or disk

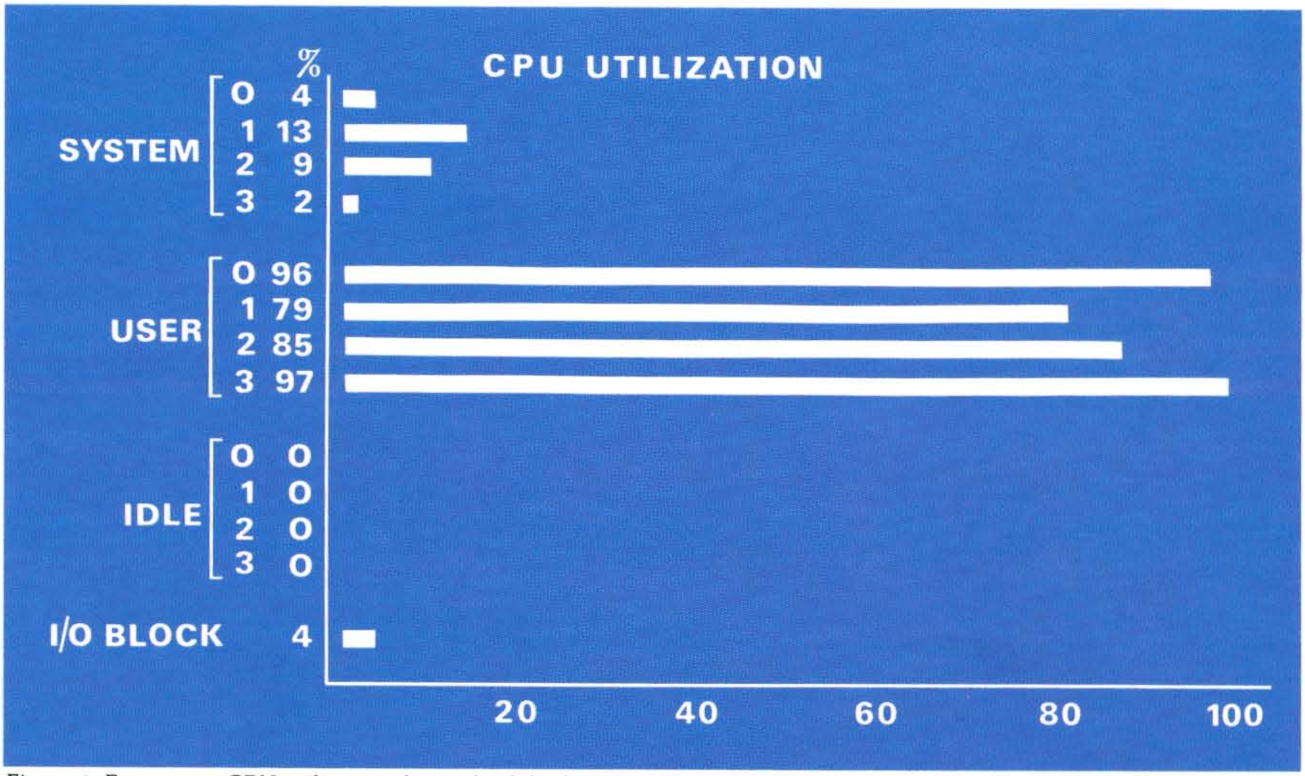


Figure 1. Percentage CPU utilization for each of the four CPUs on the CRAY X-MP/48 according to system usage (mainly I/O handling in this example), user program, and idle time. The I/O block represents idle CPU time spent waiting for outstanding I/O requests.

files, thus allowing the geophysical analyst greater flexibility and productivity. Such large processing sequences may generate large jobs, but one-half million words is the largest normally used in our experience. (Obviously, this size will vary on different machines with different hardware and software architectures.) To avoid rolling in and out of the jobs, it is highly desirable to have sufficient memory to fit all the required jobs into memory simultaneously. For efficiency of disk processing, it is best if two disk drives are allocated to each job.

The fast disk storage (that is, the solid-state buffer memory or SSD) requirement is harder to specify, but about one million words per job should be considered the minimum — although any more will be used to good effect. Two tape drives are needed per job. In the long-term, a job requires about half of a CPU (making allowances for things such as starting the jobs, mounting tapes, etc. which reduce the efficiency). Thus a CRAY X-MP/24 would be able to support four concurrent SKS seismic jobs with great efficiency if it had eight disk drives, eight 200-ips tape drives (plus at least two spares to allow for rewinds and breakdowns), and a minimum of four million words of fast-access storage.

A CRAY X-MP/48 system would obviously need twice the resources and twice the number of concurrent jobs to run with equal efficiency. However, I can envisage some logistical problems running such a system fully loaded with seismic work. Extrapolating from our experiences on the CRAY-1/S system, I would expect there would be tape mount requests occurring at least as frequently as one per minute continuously. This would require exceptionally good tape library management!

SKS and balance

The disproportionate relative speed of computation on Cray computers compared to older I/O capabilities using the DD-29 disk hardware compromised the ability to do very I/O-intensive seismic processing. (Cray Research no longer manufactures the DD-29 disk drives.) The advent of faster storage devices (such as DD-39 and DD-49 disks, solid-state buffer memory, the SSD, and memory-resident disk files combined with larger memory sizes) has effectively redressed the balance between computation speed and I/O capability. To best use the balanced computational and I/O capacities provided by Cray systems, a portable software package should be able to distinguish between a small, fast device and a relatively slow, large device and be able to use large amounts of memory, if available. The modern design of Merlin Profilers' SKS software (which has efficiency-dependent and system-dependent kernels) recognizes these potential differences and can quickly and easily be modified to make the best possible use of particular hardware configurations. □

About the author

Paddy Bennett received a mathematics degree from the University of Cambridge in 1973. He joined Merlin Geophysical Company (now Merlin Profilers Limited) shortly after its formation in 1979, after five years with another major seismic exploration company. He was part of the original design and development team for SKS and in July and August 1984 was responsible for moving the complete package to the CRAY-1/S. His current work involves aspects of processing efficiency, particularly on Cray systems, and portability of the SKS package to other systems.

Asynchronous queued input/output

Clayton D. Kirkland, Cray Research, Inc.

Vectorization has produced a dramatic reduction in the execution time needed for many application codes. This increased processing speed has brought new pressure on system I/O capabilities. Over the last 15 years, disk performance has only doubled, while single-processor speed has improved by a factor of ten or more. Multitasking or microtasking with several CPUs further increases the apparent computational rate and has accentuated the I/O component of programs even more. Since part of the operating system code for handling I/O on the CRAY X-MP supercomputer is single-threaded, only one I/O request initialization can be processed at a time. Hence, reducing system overhead is an important factor in improving both individual job performance and overall system throughput. The optimization of the I/O in application codes requires as much attention as the optimization of the computations.

I/O on Cray systems

Many types of I/O can be done on datasets. By default, a Cray operating system (COS) dataset is a blocked, buffered, and sequential dataset. Blocked datasets have a block control word every 512 words and an additional control word at the end of each logical record and file and at the end of data. A dataset optionally can be defined as an unblocked dataset. Unblocked datasets have no control words and are simply a sequence of blocks of pure data with a beginning and an end. In addition, they usually are unbuffered, so data moves directly between central memory and the secondary storage device.

Buffered datasets are managed by I/O library software and use a temporary area in memory called a buffer to prefetch read requests or to accumulate several write requests before a physical transfer occurs. All physical transfers are required to be in multiples of a sector (512 words) and must start on a sector boundary.

A sequential dataset has records that are processed sequentially from beginning to end. A random dataset allows the reading and rewriting of records in place and at random locations. When a synchronous I/O request is made on a dataset, the execution of the program is suspended until the request is completed. An asynchronous request allows program execution to continue while the I/O transfer occurs.

With versions of COS prior to 1.14, only one I/O request on a dataset can be active at a time. This means that each I/O request requires an operating system call. In COS 1.14 and later versions, this condition is relaxed. Additionally, the typical overhead for a COS call has been reduced to approximately 400 microseconds (μsec) per I/O request. For random datasets, system overhead can be reduced further by allowing multiple requests to be queued into a list and activated by a single call to COS. This allows COS to examine several I/O requests at a time. The overhead cost of processing the list of requests is approximately 400μsec for the first list item plus only 50μsec per subsequent list item or packet request. These times do not include the data transfer time.

AQIO and Compound AQIO defined

Asynchronous Queued Input/Output (called *AQIO* or just *queued I/O*) is a dynamic, circular, asynchronous, unblocked, and random I/O scheme implementing the technique previously described. It is available in COS 1.14 for Solid-state Storage Device (SSD) datasets and will be available in COS 1.15 for SSD datasets and disk files.

AQIO gives the user the ability to build up a list of requests for unblocked, random transfers between central memory and a COS dataset. The list is circular in that the first item in the list follows the last item. The user may specify when COS should begin processing the list and may monitor the completion of individual requests. Additional requests can be added to the list while COS is processing it as long as the list is not full. The capabilities of AQIO are important in many two- and three-dimensional applications because they significantly reduce the overhead in managing the plane and pencil access patterns commonly found in the data structures of these codes.

A performance enhancement to AQIO called *compound AQIO* is available. Several AQIO operations involving a repeating pattern of read or write requests can be specified with a single packet request. A compound request describes a collection of fixed-length dataset sections (a section is a group of sectors) with a fixed increment between sections, and an area in central memory containing sections separated by a constant stride. This technique effectively passes a DO-loop down to the lowest level in COS and is the most efficient way to do random I/O.

```
CALL AQOPEN                    (AQL,AQLSIZE,DN,STATUS)
CALL AQCLOSE                                (AQL,STATUS)
CALL AQREAD                (AQL,MEMADD,SECADD,
                     SECTORS,REQID,QUEUE,STATUS)
CALL AQWRITE               (AQL,MEMADD,SECADD,
                     SECTORS,REQID,QUEUE,STATUS)
CALL AQREADC              (AQL,MEMADD,MSTRIDE,
                       SECADD,SECTORS,DSTRIDE,
                      SECTNS,REQID,QUEUE,STATUS)
CALL AQWRITEC             (AQL,MEMADD,MSTRIDE,
                       SECADD,SECTORS,DSTRIDE,
                      SECTNS,REQID,QUEUE,STATUS)
CALL AQSTAT                 (AQL,REPLY,REQID,STATUS)
CALL AQWAIT                                 (AQL,STATUS)

WHERE:
        AQL = AQIO list array.
    AQLSIZE = size of list and DSP.
         DN = dataset name in ASCII.
     MEMADD = initial memory address.
     SECADD = initial sector address.
    SECTORS = number of sectors to transfer.
      REQID = request ID.
      QUEUE = queued flag.
              if ZERO, queue request and
              activate I/O.
              if NON ZERO, queue request only.

     STATUS = status flag to user.
    MSTRIDE = stride through memory.
    DSTRIDE = stride through dataset.
     SECTNS = number of sections.
      REPLY = currently unused.
```

Figure 1. AQIO routines and their argument definitions

Using AQIO

The user interface to AQIO is through FORTRAN call statements. Figure 1 shows the syntax for using the AQIO routines. The routines AQOPEN and AQCLOSE open and close files on which AQIO transfers are to occur. The argument AQL allocates AQLSIZE words that will contain the circular list of I/O packet requests. DN gives the dataset name. All AQIO routines return a STATUS flag indicating whether the request has been accepted or whether the queue is full. Simple AQIO reads and writes are performed using AQREAD and AQWRITE. Calls to these routines specify the starting memory address, MEMADD, and the starting sector address, SECADD, of the dataset. The number of sectors to transfer is given by SECTORS. Each I/O request is identified by a request number, REQID, which allows interrogation of the request at a later time. A QUEUE flag indicates either that all transfers in the list should begin after this request or that only a new item should be added to the list.

The compound AQIO routines, AQREADC and AQWRITEC, contain additional arguments to specify multiple data transfers at one time. A description of the memory area is given by the starting address, MEMADD, of the first section. The length of this section is SECTORS sectors, and the distance from the first section to the second is the memory stride, MSTRIDE. The number of sections is given by SECTNS. The description of the dataset area is similar. SECADD gives the starting sector number of the first section. Each section is SECTORS sectors long. The distance from the first dataset section to the next is the dataset stride, DSTRIDE. The number of sections is again SECTNS. Other arguments are the same as described earlier.

The user may interrogate the status of a particular request or wait for the completion of all requests on a dataset. Information about a specific request may be obtained by calling AQSTAT and by giving the identification number, REQID, of the request. AQSTAT returns a status flag, STATUS, indicating whether the request is completed, or remains queued, or whether the queue is empty (and the request is completed). A call to AQWAIT suspends program execution until all requests in the list have been satisfied.

Tests run on a CRAY X-MP/48 supercomputer with a 128-million-word SSD show the dramatic performance gains obtained with AQIO. Two very high-speed channels connect the SSD with central memory, giving a hardware bandwidth of 156.25 million words (Mwords) per second per channel. An experiment using the SSD consists of 50 write requests of a fixed-size record. Several runs are made for varying record sizes. Figure 2 shows the transfer rates observed using traditional single request calls, AQIO, and compound AQIO. While all three methods eventually produce transfer rates approaching the hardware rate as the record size increases, AQIO and compound AQIO converge to the limit more rapidly. The rate of 291.11 Mwords/sec represents 93 percent of the hardware bandwidth.

AQIO and compound AQIO enable the user to make full use of the CRAY X-MP hardware I/O capabilities. Both the user job and total system throughput benefit from the reduced operating system overhead and decreased I/O wait times. Those interested in obtaining more information about AQIO may contact me, Clay Kirkland, at Cray Research, Inc., 1333 Northland Drive, Mendota Heights, MN 55120; telephone: (612) 452-8520. □

Transfer rate in Mwords/sec

SSD record size in sectors	Single requests	AQIO	Compound AQIO
1	1.22	13.60	33.50
2	2.43	14.28	36.35
4	4.76	28.51	72.28
8	9.42	56.66	133.46
16	18.98	104.59	77.25
32	33.36	152.89	231.90
64	62.75	208.88	263.97
128	75.28	254.01	285.72
256	131.33	265.67	291.11

Figure 2. Transfer rates for 50 requests using several I/O techniques and record sizes

About the author

Clay Kirkland is a senior applications programmer with the Cray Research benchmark services group in Mendota Heights, Minnesota. His primary area of expertise is I/O optimization techniques. Kirkland joined Cray Research in 1979 after working with Control Data Corporation for 15 years. He received a Bachelor of Science degree in electrical engineering from the University of Minnesota in 1964.

A SLEDGE HAMMER APPROACH TO I/O

Doug Spragg, Exxon Production Research Company, Houston, Texas

Petroleum exploration research personnel at Exxon Production Research Company have written many wave equation processing programs for the company's CRAY-1/S computer. These programs require access to very large amounts of seismic data in a number of different orders. Because of this requirement, an efficient I/O strategy was needed. Exxon personnel met this need by developing WEDISK, a very efficient I/O subroutine package. WEDISK has since been installed in several wave equation programs, resulting in dramatic performance improvements. Cray Research is currently developing a queued I/O (also called asynchronous queued I/O) feature for its COS operating system that will encompass many of the WEDISK capabilities.

The need for queued I/O

Many of the wave extrapolation programs written at Exxon Production Research Company access seismic data in common depth point (CDP) order, common source point (CSP) order, and common receiver point (CRP) order. *CDP traces* are seismic data collected at the same midpoint (halfway between receiver and source); *CSP traces* are seismic data collected from the same shot point; and *CRP traces* are seismic data collected from a common receiver location.

Prior to the availability of the WEDISK package, it was necessary to go through a sorting routine whenever the using (calling) program switched from one order to another. A typical example is provided by the marine WVSTAT (wave equation datuming) program described by J. R. Berryhill ("Wave-equation Datuming Before Stack," *Geophysics* 49, 1984, pp. 2064-2066). The logical flow of WVSTAT is as follows.

First, CDP data are read from tape and modified to generate synthetic split-spread data. *Split-spread* data are acquired using receivers on both sides of the source and allow for better discrimination of geologic dip. The data are sorted into CSP order; the first phase of the WVSTAT process "moves" the geophones to the water bottom. A second sort gets the data into CRP order and the second phase of WVSTAT again "moves" the sources to the water bottom. The "movement" of the geophones and sources back to the water surface (with a replacement velocity) is accomplished with two more phases of WVSTAT and an intermediate sort. The synthetic split-spread data are then collapsed and a fourth (final) sort gets the data back into CDP order.

Sorting the data before each WVTSAT execution enables the program to read the data in sequential fashion in the desired order. Sorting represents a tedious and time-consuming job step that would best be avoided, particularly in this application because complete processing requires many passes through the data.

I/O development

These considerations led me, after discussions with associate J. R. Berryhill, to investigate developing an efficient I/O scheme on the CRAY-1/S computer that would permit Berryhill's wave equation programs to write all the data onto the disk in their original order (CDP) and permit access to one ensemble at a time in any of the three required orders (CSP, CRP, CDP), with no requirement for intermediate sorting. (An *ensemble* is a group of traces, typically on the order of 100.)

From discussions with Cray Research personnel, I knew that the way to minimize disk contention and spread the data across many disk packs was to initiate many distinct datasets. This is because Cray's COS operating system attempts to use as many scratch disk packs as are available when initiating a sequence of datasets. It was decided that, on the CRAY-1/S computer, a unique dataset would be assigned for each stack bin of data. A *stack bin* is a group of traces with a common depth point. A typical user job contains thousands of stack bins of data.

I also knew that efficient random access to the data could be realized only by using unblocked I/O. This format insists that data transfer to and from the disk be in multiples of 512 words (one sector on a Cray disk). If the data are arranged in memory so that a multiple of 512 words is allocated for each trace, then a natural mapping occurs between trace number in a stack bin and sector in the disk dataset where the data for that trace are written to (or read from). The use of unblocked I/O also eliminates the need for system I/O buffers. The I/O is done directly to and from user buffers. Since WVSTAT and other wave front extrapolation programs do I/O one ensemble at a time, the use of blocked I/O would make the space requirements for system buffers prohibitive. In addition, using unblocked I/O relieves the need for mapping the dataset onto a physical disk with 512 word sectors, and for various record control information entries interspersed with the data.

This knowledge made it clear that an I/O system was needed that could perform asynchronous (allowing computation to overlap I/O) unblocked I/O. A conference was held with Cray Research personnel to see whether such a system already existed. In fact, such a system had been developed at Cray Research's United Kingdom subsidiary and the system had performed impressively during extensive timing studies. Based on these results, the I/O package was adopted as the basis for the I/O package to be used in the wave equation application at Exxon. The package needed subroutines to:

- □ Initialize the I/O package
- □ Load the data onto the disk (one dataset per stack bin)
- □ Permit users to read a common source bin or a common receiver bin of trace data, modify the data by calculation, and then put the data back onto the disk (into the same place from which it had been read)
- □ Read the data back from the disk.

To accommodate these needs, the I/O package had to do a great deal more than call the I/O system. First, since a using program may want to retrieve and write data in any of three different orderings, a set of sort keys had to be created and sorted. These had to point to the disk location (stack-bin number and sequence number in that bin) of each trace for each ordering. Second, the complications of allocating a multiple of 512 words per trace such that the I/O can be done in unblocked format and the look ahead (buffered) feature had to be handled effectively by the I/O package — all this being transparent to the using program.

In addition, of course, all data had to be transferred one ensemble at a time. That is to say, one entire stacking bin (CDP order), or an entire common-shot bin (CSP order), or an entire common-receiver bin (CRP order) had to be transferred with one call to the I/O package. This is a substantial departure from most other seismic I/O at Exxon Production Research Company which works on only one trace at a time. Hence, this I/O package is sometimes referred to as an ensemble I/O package.

The WEDISK package has since been implemented by J. R. Berryhill in WVSTAT and in several other wave extrapolation programs. Because of the substantial improvement in I/O performance that the package affords, Cray Research is planning modifications to COS that will incorporate some of the package's features. □

About the author

Doug Spragg received his M.A. in mathematics from Michigan State University in 1972. He worked for eight years as a scientific programmer at the Bettis Atomic Power Laboratory in Pennsylvania before joining Exxon Production Research Company in 1981. His current work centers on seismic data processing applications.

Econometric supercomputing

"There are in the fields of economics no constant relations and consequently no measurement is possible." Ludwig von Mises

Contrary to the sentiments of Ludwig von Mises, the late economist and champion of free enterprise, today's economists are refining ways to measure and mathematically predict economic behavior. Although not a simulation of a physical phenomenon, such as air flowing over an aircraft or oil through a reservoir, the modeling of money, goods, and services flowing through an economy is nonetheless emerging as a practical application for supercomputers. Economists working in education, government, and industry are using the large storage capacities and fast processing speeds available on Cray systems to run large models in detail and to obtain faster turnaround.

In the past, economic forecasts have been compared disparagingly to weather forecasts. Despite the occasional horror story surrounding a wayward prediction, however, economic forecasting has matured in the past two decades into a multimillion-dollar industry. An outgrowth of methods originally designed to help government policymakers, this new information industry comprises service bureaus that market subscriptions to their periodically updated forecasts. In addition, economic modeling is conducted in-house by some government agencies and private companies.

Models and forecasts

An economic forecaster's primary tools are complex statistical models that chart the interactions among economic variables. The models are derived from econometrics, a discipline that applies statistical methods to economic data and problems. In a simple example of an econometric model, one might define the gross national product (GNP) as the sum of individual personal expenditures, business investment, and government purchases. (GNP can be defined in many ways, depending on the economic theory invoked.) In this example, the GNP is a *dependent variable*, because it depends on, or is a function of, other independent variables. The *independent variables* have values that are assigned, determined, or computed, often requiring that they assume the role of dependent variables that are functions of still other independent variables. Thus large, complex models rapidly evolve.

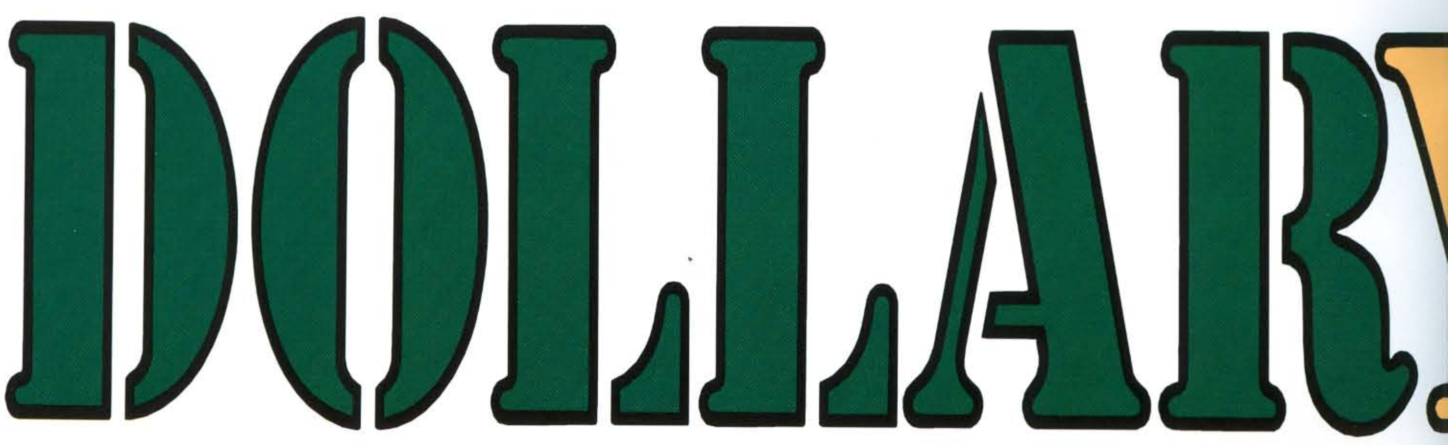
Model types and requirements

Econometric models are typically macroeconomic in that they attempt to model national and larger economies. Such a model might include variables such as money supply, tax and interest rates, the inflation rate, consumer income and spending, and stock prices. The economic theory on which the model is based determines the variables used and the ways and degrees to which they are correlated.

An *economic theory* is a set of assumptions about the cause-and-effect relationships among economic variables. Such sets of assumptions are the foundations for *structural* economic models. Models that rely primarily on historical correlations among variables, and downplay assumed cause-and-effect relationships, are called *reduced form* models.

Structural and reduced form econometric models forecast the economic future and simulate alternative economic scenarios. In government, for example, policy makers may want to predict the effects of deficit spending on tax revenues before drafting a budget. In business, managers may want to predict the effects of government policy on consumer spending before setting production quotas. In either case, the results of a good econometric model aid the intuition and expertise of the decision makers.

The three major econometric service bureaus, Chase Econometrics (a division of Chase Manhattan), Data



Resources, Inc. (a division of McGraw Hill), and Wharton Econometric Associates (a division of Ziff Davis), employ models that use from 455 equations (Chase model) to over 1000 equations (Data Resources model). Perhaps the most comprehensive model in existence, the Kent model, developed and used by Kent Economic and Development Institute at Kent State University in Kent, Ohio, uses 22,500 equations.

Although service bureaus are the backbone of the forecasting industry, econometric models also are developed and used as tools by large corporations and financial institutions such as banks. However, the largest models and those requiring the fastest turnaround time can be limited in their usefulness by the computing power available to such an institution. The users of such models can benefit significantly from the speed and large memory available with supercomputers such as Cray systems.

Potentials for supercomputers

Although the application of supercomputers to econometric modeling is less refined than applications in the physical sciences, there are some notable instances. For example, the Federal Reserve Bank of Minneapolis has used a CRAY-1 computer at the University of Minnesota to run monthly forecasts of the national economy. The forecasts are used as briefing material by the bank's president at meetings of the Federal Open Market Committee, which directs the Federal Reserve Board's open market operations.

The bank uses a reduced form model based on a forecasting technique known as *Bayesian vector autoregression* (BVAR). "The technique treats conventional economic theories in an agnostic way," said Robert Litterman, a senior analyst and manager of technical operations for the bank. "BVAR has been described as not being based on economic theories, but that's not quite true. The variables we choose necessarily imply an economic theory. For example, we look at interest rates but ignore sunspots."

BVAR is "agnostic" toward economic theories in that its equations do not specify strong relationships among the variables. Instead, the method relies on historical data; it forecasts a variable's performance based on the variable's own and others' past performance. But it does not assume rigid quantitative relationships among variables, as is done with *structural*, or *theory-based, models*.

A strict vector autoregression would mechanically include all past correlations among variables. This, however, would lead to inaccurate forecasts because of coincidental correlations, or noise, in the historical data. For this reason, some assumptions must be made regarding variable correlations. This modification is called a *Bayesian approach* — thus the name of the technique. These assumptions are points where economic theory enters the otherwise "agnostic" methodology.

Structural models use few variables with large coefficients relating them, whereas the BVAR model uses many variables with coefficients close to zero. The result is that structural models use many small equations and the BVAR model uses relatively few large ones. The national BVAR model comprises 46 equations, each with about 120 variables.

"It's a compute-bound problem, requiring about as much work as a matrix inversion based on the number of variables," Litterman explained. "For example, solving an equation with 120 variables requires as much work as inverting a 120 x 120 matrix. Cray Research's CFT compiler vectorizes our program extensively, and it runs in about 20 minutes of CPU time on the CRAY-1 computer at the University of Minnesota."

The VAR approach was first explored in the 1940s and refined somewhat in the 1960s, but the noise problem made it unpopular. In the 1970s, Litterman and Christopher Sims, a professor of economics at the University of Minnesota, resurrected the technique adding the Bayesian modification. The specific model Litterman wrote based on the technique is now a commercially available program called RATS (Regression Analysis of Time Series).

Litterman and Sims are currently using the CRAY-1 computer at the University of Minnesota for continuing research on VAR methods to resolve some technical issues and to develop an international VAR model linking the U.S. economy to those of Western Europe and Japan. The international model, which is being developed for a Brookings Institution conference on the global economy, will include variables for imports, exports, and interest and exchange rates.

"In developing the international model, we have been using the Cray system in somewhat of an interactive way," Sims explained. "We run the model in 200-second chunks,

PESO LIRA

then look over the results, make our modifications and rerun a chunk, look it over, modify it, and so forth."

Such an iterative approach relies heavily on the speed of the CRAY-1 computer. Because of the large number of interacting variables that must be mutually optimized, many iterations are required. The CRAY-1 computer's speed makes practical the repeated trials needed for this kind of development work. "What we're doing wouldn't be feasible on a conventional mainframe," Sims said. "The turnaround time would be prohibitively long."

Other econometric modeling approaches can also benefit from supercomputing. Large-scale cross-sectional statistical models, for example, require nonlinear fitting methods. These models have been reasonably successful, but limitations in available computing power have required making assumptions about probability coefficient distribution functions. "There's talk in the economic community about relaxing these assumptions now, because of the computing power that's available with supercomputers," said Sims.

Examples of applications of cross-sectional statistical models include a number of studies done in the 1970s to assess the effects of a proposed negative income tax as well as studies that are performed routinely to assess the efficacy of proposed urban mass transit systems.

"Computers with capabilities similar to Cray systems have a definite place in econometric modeling," Sims noted. "Unfortunately, the budget pressure in the social sciences means that few economists can get access to such systems."

The difficulties

Although supercomputers have the potential to greatly aid econometric modeling, some obstacles need to be overcome or worked around. Nobel Laureate Wassily Leontief, director of the Institute for Economic Analysis (IEA) at New York University, relies on a supercomputer to conduct large input-output analyses, a modeling methodology practiced worldwide and particularly popular in Japan. Leontief is less optimistic than Sims about the future potential of supercomputers in econometrics.

"Economists tend not to formulate problems for large-scale computers because of the lack of access," Leontief points out. "But more importantly, although large-scale computers are useful in any type of systems analysis, the overriding limitation for social scientists is not so much computer power as insufficient data. The United States is the only advanced country in the world that does not have a central statistical office. Each governmental department collects only the statistics that pertain to its own particular interests. Although, in the systematic gathering and updating of detailed information, large computers can play a significant role."

Daniel Szyld, an associate research scientist at IEA, currently working in the computer science department at Duke University, Durham, North Carolina, says he is optimistic about the future role of supercomputers, but with similar reservations. "Supercomputers have a role to play in economic modeling, but the extent to which that potential is realized depends on the availability of data, and in investments that need to be made in model building and numerical methods," he commented. "Unlike work in the physical sciences, where data is in some sense 'self-generated,' in computational economics the real numbers are needed."

In perspective

Econometrics is a developing supercomputer application and it exhibits the difficulties one would expect to find in an emerging discipline. But as these difficulties are overcome and the economic community gains familiarity with and learns to take advantage of supercomputers, many new research and production opportunities will present themselves. Fully developing these potentials, however, largely depends on the extent to which economists can access supercomputers.

Although current econometric models suffer from incomplete theoretical foundations and some types of data are of limited availability, researchers have already done much to dispel the pessimism of von Mises. Complexity in economics need not preclude detailed econometric modeling, at least not as far as the hardware capacity needed. Cray systems provide the fast processing and large memories economists need to get quick turnaround and precise results from large, comprehensive econometric models. □

Acknowledgement

Special thanks to Christopher Sims of the University of Minnesota and Robert Litterman of the Federal Reserve Bank of Minneapolis for their help in preparing this article.

POUNDF

Forecasting energy consumption

Consider the following task: predict the commercial, residential, and industrial consumption of coal, oil, natural gas, and electricity for three northwestern states during the next 25 years. Although a complex job, a computer program to perform it was developed in the United States during the mid-1970s. The program, called the NEPP (Northwest Energy Policy Project) model, was intended as a tool to help utilities and other interested organizations deal with rising oil prices and a perceived threat of impending oil shortages — the 1970s' energy crisis. The NEPP model delivered its projections from a CRAY-1 computer system at United Computing Services (now United Information Services, or UIS) between 1980 and 1981.

"With about 3000 lines of code, it was quite a massive econometric model — one of the most sophisticated for the Pacific Northwest at the time," said Stephen Scott, a developer of the model, now a systems analyst at Synergic Resources Corporation, a Seattle consulting firm working in economics and engineering.

The Northwest Regional Commission developed the NEPP model to forecast the energy consumption of Washington, Oregon, and Idaho. Explains Scott, "At the time, people were very concerned about the availability of oil. I rewrote sections of the model so it would constrain the supply of petroleum products and compute shadow prices. (A *shadow price* is the price a person is willing to pay for the next unit of a commodity that is unobtainable.) I then ran the enhanced model on the CRAY-1 computer for the Northern Energy Policy Workshop at Portland State University around 1980." The modified program was used for various calculations, such as determining the extent to which consumption would shift to other forms of energy if a given region or sector was unable to fill its petroleum needs.

"Originally, we ran the model on a CYBER 175 computer and it took about 2½ hours to turn around," Scott said. "After we converted it to the CRAY-1 system we got a 15-minute turnaround, and it cost us about one-fourth of what it did on the CYBER computer. The savings were due mostly to the sheer speed and power of the CRAY-1 system."

Although the NEPP model has since been retired, Scott says a broader potential in energy forecasting may be opening up for high-speed computers. "Because it's becoming clear that most of our new energy reserves will come from conservation — it's simply cheaper than developing new sources — there is a move toward so-called 'demand-side' management," Scott explained. "This type of energy management requires complex statistical manipulations of very large data sets."

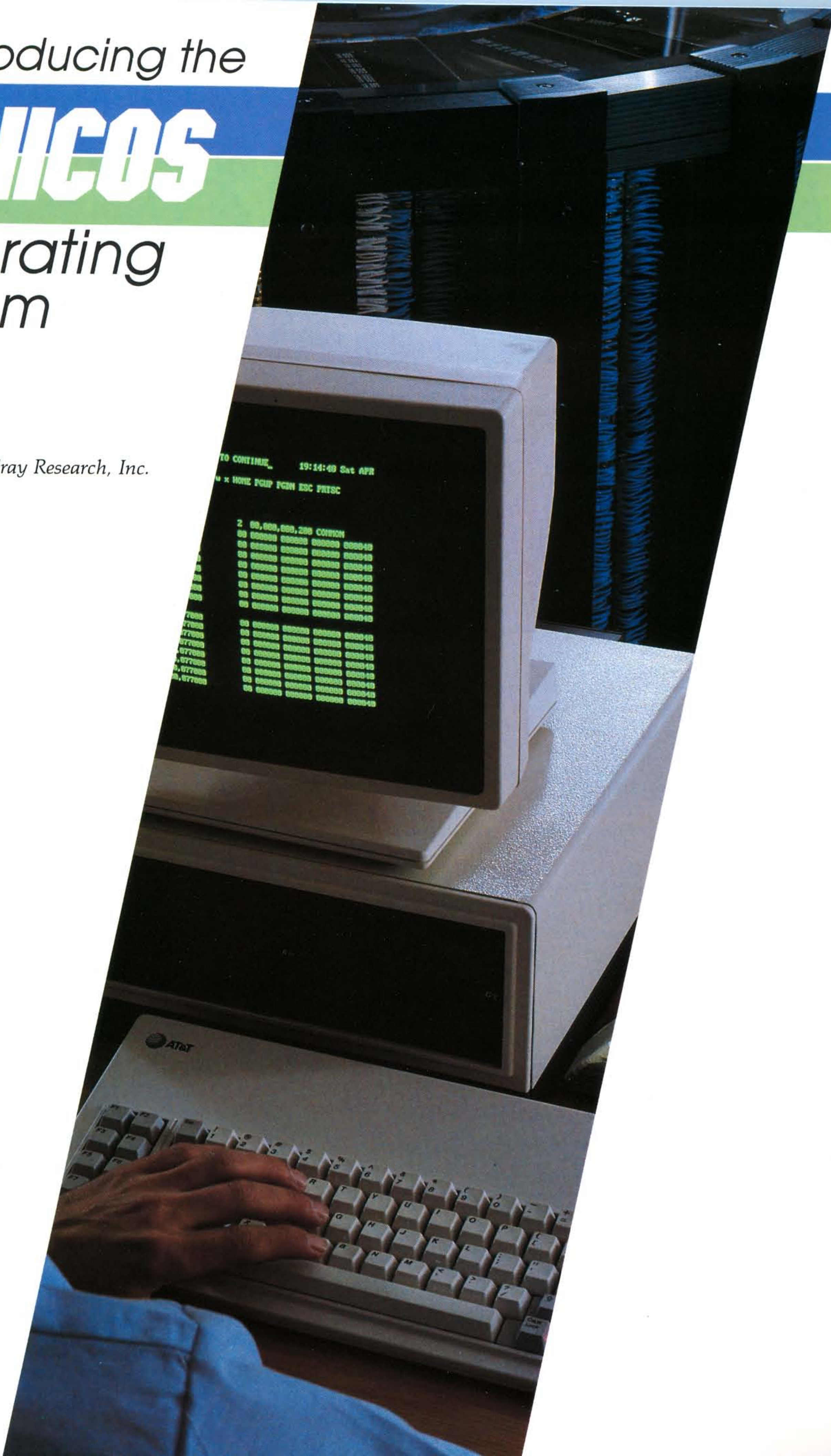
The idea behind demand-side management is essentially to graph the load shape, or energy consumption pattern, of individual residential appliances or units of industrial equipment over days or months, and to try to smooth out consumption peaks. The data could be statistically manipulated to unify load shapes of appliances in order to develop a "household" load shape, or to see how pricing policies affect load shapes. Weather and demographic adjustments could also be made.

"The goal is to move energy off the peaks, especially for coincident peaks," Scott said. "This approach would be especially useful for large utilities that deal with multiple service areas and various types of load shapes. For many applications the data sets would be so huge that processing power similar to that of Cray systems would be needed to manipulate them."

Although the energy crisis has subsided from headlines in recent years, the conservation measures it spawned are now facts of life for homeowners and industries. As a result, conservation is recognized as a valuable energy "source." To better exploit this source, increasingly complex methods are needed to measure the pulse of energy consumption. Accurately predicting patterns of regional as well as national and international energy use will require data availability, sophisticated algorithms, and sufficiently powerful computer hardware. Cray systems can contribute decisively to addressing the last of these needs.

Introducing the UNICOS operating system

Lynn Sayles, Cray Research, Inc.



Cray Research is pleased to announce the first release of the UNICOS operating system — providing a new software base that supports both current and future Cray computer systems. UNICOS, which is based on the AT&T UNIX System V operating system, runs on the CRAY-2 computer system and on CRAY X-MP or CRAY-1 systems configured with an I/O Subsystem.

Maximizing performance

Cray Research has a primary commitment to providing software that maximizes the performance of its computer systems by taking advantage of hardware features. At the same time, the company is committed to ensuring that its products fit a wide variety of user environments and to providing dependable service to users.

Because of these commitments, Cray Research formed a study group in early 1983 to identify a single operating system to support across Cray's product line. The study group chose AT&T UNIX System V because it was seen as a well-designed, portable, interactive operating system very adaptable to the Cray environment. Furthermore, UNIX has emerged as a widely accepted operating system, and a growing user community already exists. And because UNIX is designed to be interactive, it provides users with unique tools for the interactive environment that complement the computational power of Cray hardware.

Now, because UNICOS is portable and can be supported across Cray's product line, Cray Research can explore and develop new, faster computer technology without being restricted by architectural compatibility. In turn, customers will quickly be able to take advantage of the computational power of current and future Cray systems because of compatible application interfaces across the Cray product line. For example, an application running on a CRAY-2 system can be easily moved to a CRAY X-MP system and vice versa.

For current customers running the Cray operating system (COS), Cray Research will offer an environment and tools to migrate from COS to UNICOS. First, the proven Cray FORTRAN environment will be supported under both COS and UNICOS. Any FORTRAN application that runs under one system will run under the other, given that there are no specific system dependencies. And second, with UNICOS Release 1.0 and the COS 1.15 Guest Operating System (GOS) feature, CRAY X-MP multiprocessor users have the capability to run both COS and UNICOS simultaneously. Having both systems available at the same time allows the user ready access to either system, and the GOS feature can be used to facilitate migration to UNICOS by allowing a smooth and gradual transition. Customers are assured of retaining the functionality of COS through the migration process.

Offering flexibility

In addition to its portability, UNIX has a growing user community, making it easy for Cray systems running UNICOS to be integrated into existing UNIX environments. This enables Cray users to take advantage of the rapid advances in high-speed workstations, mainframes, and personal computers running UNIX and to work with the many graphics and communications tools available. With a standard operating system functioning on all machines, users are freed from having to know a different set of commands for each system.

Running UNICOS, users have available both batch and interactive processing. A primary function of every Cray system is to efficiently process long-running programs requiring large memory and high I/O bandwidth. UNICOS includes a batch service capability that is tailored to take advantage of the features of the Cray architecture — powerful CPUs, large memories, and high-speed I/O. The UNICOS batch capability will be extended to provide the same functionality COS provides today.

UNICOS also offers the advantage of being interactive. It is now possible for a user to communicate from a desktop computer to a Cray computer. For example, a user can combine graphics capabilities of a workstation and the computing power of a Cray system to perform computer modeling of physical phenomena interactively. A user at a workstation can monitor the progress of an application, suspend its execution, modify the program or data from the workstation, resubmit it, and get instant feedback as results are displayed. As applications grow larger, users can have a Cray UNICOS system performing heavy computational work and a workstation acting as a window to the Cray system, providing users with a "super" tool.

Current stations for other vendor mainframes available with COS will continue to provide powerful interactive access to UNICOS.

UNICOS structure

UNICOS is comprised of a small kernel surrounded by a large set of utilities that service system functions. The UNICOS kernel has been designed to maximize performance and functionality of Cray computer hardware. The design takes advantage of the Solid-state Storage Device (SSD) and the I/O Subsystem on CRAY X-MP and CRAY-1/S systems and the very large memory and large I/O bandwidth on CRAY-2 systems. In addition, networking features are provided, including TCP/IP protocol and NSC HYPERchannel support that allow Cray systems to integrate easily into existing environments. Support for Cray station connections will also be provided in future releases.

UNICOS

Because of the modular design of UNICOS, customers can make local enhancements using UNICOS utilities, making it more convenient to preserve their system as well as their application software investment. Also, new utilities may be written and tested without impacting the overall stability of other system functions. This allows users and system developers to add utilities to the system without modifying existing ones.

Cray UNICOS software products

Essential compilers and libraries have been developed to function with UNICOS. The Cray FORTRAN Compilers (CFT and CFT2), Cray C Compiler, Cray Pascal Compiler, and Cray Assembler (CAL) Version 2 are available with UNICOS Release 1.0. Also included are standard UNIX C and I/O libraries, optimized arithmetic and scientific libraries, and libraries to provide additional runtime support for FORTRAN and Pascal. These are the same products available under COS on CRAY X-MP systems.

FORTRAN is the predominant language used for solving mathematical problems. Available with UNICOS Release 1.0 are Cray FORTRAN Compilers that automatically vectorize and optimize code and allow users ready access to all multitasking features that have been developed under COS. CFT is offered with CRAY X-MP and CRAY-1 computer systems, and CFT2 is offered with the CRAY-2 computer system. Under development is a new, state-of-the-art Cray FORTRAN compiler which, when available, will run under UNICOS.

The C programming language is a high-level language available with Release 1.0 of UNICOS. The majority of the UNICOS kernel and utility programs comprising the system are written in the C language. The grammar and rules of Cray Research's C language are compatible with the emerging industry-standard. Because the C language is becoming more widely used in applications and system-oriented programs, its availability on Cray systems complements the scientific orientation of FORTRAN.

The Cray Pascal Compiler is an optimizing, vectorizing compiler. Cray Pascal is the implementation language for the Cray Assembler (CAL) Version 2, the new Cray FORTRAN Compiler being developed, and the Cray Pascal Compiler. Cray Pascal complies with the Level 1 requirements of standard ISO 7185, defined by the International Standards Organization (ISO). The Cray Pascal Compiler issues messages identifying Cray extensions to help transport a program to a machine running a different implementation of the language.

The Cray Assembler (CAL) Version 2 offers Cray users a powerful macro assembly language. CAL allows a user to express symbolically all hardware functions of a Cray mainframe. This detailed and precise level of programming is of special aid in tailoring programs to the architecture of a Cray computer and writing programs requiring code that is optimized to the specific hardware.

The future

Release 1.0 of UNICOS reflects Cray Research's commitment to provide Cray systems that are powerful tools for users and that integrate into different computing environments. Future releases of UNICOS will continue to reflect this commitment. These releases will include expanded networking capabilities, enhanced batch capabilities, and expanded application program support that will be superior in the large-scale scientific marketplace. □

About the author

Lynn Sayles manages the operating systems and on-line diagnostics writing group in Cray Research's technical publications department. She has been employed by Cray Research since 1981 and has been a UNICOS project writer since 1983. Prior to joining Cray Research, she was a technical writer for a consulting firm for two years. She also directed the technical writing staff of the Federal Reserve Bank of Minneapolis for six years.

UNICOS development

In July 1983, the CRAY-2 Software Development group, headed by Dave Judd, began porting a version of the AT&T UNIX System V to the CRAY-1/S computer system — the first step in porting the system to the CRAY-2. By September 1983, a demonstration system was running on the CRAY-1/S. Until a CRAY-2 computer system was available, the group used a CRAY-2 simulator on the CRAY-1/S system, enabling them to shorten the software development time. By April 1985, a UNICOS prerelease version was running at a customer site.

In early 1984, a second software development group began to develop a transportable version of UNICOS for both CRAY X-MP and CRAY-1/S computers. This software development group, headed by Jim Miller, used as its base system the version of the operating system that had been previously ported to the CRAY-1/S. By November 1985, a UNICOS prerelease version was running on a CRAY X-MP at a customer site.

Thus, beginning from a common base system, both software groups worked simultaneously to develop the UNICOS operating system for their targeted machines.

CORPORATE REGISTER

Cray Research announces SUPERLINK/ISP

SUPERLINK/ISP is a high-performance link between Cray computer systems and IBM or IBM-compatible systems. Offered on CRAY X-MP and CRAY-1 computer systems, it provides sequential bulk data transfer between Cray's COS operating system and the International Business Machines Corporation MVS environment more efficiently than ever before. SUPERLINK/ISP is designed to complement Cray Research's MVS Station features, expanding the range of possible applications and providing greater flexibility in applications development.

With SUPERLINK/ISP, COS users can take advantage of IBM/MVS device support and IBM/MVS data management services support. Users of the Cray FORTRAN compiler (CFT) can sequentially access IBM/MVS data at the record level and customize IBM/MVS data access.

SUPERLINK/ISP takes advantage of MVS to provide error recovery, security systems, accounting, and dataset management. It uses standard interfaces and makes no modifications to any IBM-supplied software. With SUPERLINK/ISP, CFT users can sequentially access IBM/MVS data at the record level. This capability provides the following advantages:

- □ Local access to MVS data without dataset staging. The MVS datasets are not copied to Cray disk. The COS application program executes immediately; there is no waiting for data to move to the Cray disk and back again.
- □ Data transfer rates improved over Cray MVS Station transfer rates.
- □ Access to datasets too large for Cray storage.

COS users can benefit from the IBM/MVS device support and data management services support on IBM/MVS. SUPERLINK/ISP supports a wide range of MVS devices and provides a link to the many data management services present in an IBM/MVS environment. These services include MVS catalog management, MVS sequential access methods, dataset recovery routines, MVS dataset security, and compatibility with optional program products such as tape management systems or archiving systems.

SUPERLINK/ISP allows CFT users to customize IBM/MVS data access, meaning that users can replace the SUPERLINK/ISP data access module (DAM) with their own access method interface. The DAM allows record passing between programs running under COS and MVS; customizing the DAM allows COS users flexibility in the way they process data.

SUPERLINK/ISP makes sequential access to MVS data both comfortable and easy. Dataset integrity is enhanced. Datasets can be accessed in transparent, blocking, or character mode, and can be shared between systems and between users. No changes to source programs are required.

SUPERLINK/ISP is easy to use, requiring only two Cray JCL statements. The hardware connection is made through Cray low-speed (6-Mbyte) I/O channels, Cray Front-end Interface (FEI) devices, and IBM channels. For more information on SUPERLINK/ISP, contact the nearest Cray Research sales office.

New station software services announced

Cray Research recently released several new and enhanced station software services, ensuring that Cray computers continue to fit a wide range of customer requirements. The latest releases include Release 3 of the IBM VM Station Software Service, an enhanced version of the DEC VAX/VMS Station Software Service, and the CDC NOS 1.15 Station Software Service.

IBM VM Station

Release 3 of Cray Research's VM Station Software Service continues to provide users with easy access to CRAY X-MP and CRAY-1 computer systems from mainframes running International Business Machine Corporation's Virtual Machine/System Product (VM/SP). The VM Station runs in its own virtual machine under Release 3 of VM/SP, supporting versions 1.14 and later of Cray Research's COS operating system. Release 3 of the VM Station contains the following additional features:

- NETDATA support. The Station now supports MVS Network data format for dataset and job submission. This support includes remote MVS systems in addition to local and remote VM systems.
- Graphics data handling. The Station now provides a communications path for graphics data between a user's COS job, the IBM mainframe, and graphics devices. Graphics data can be displayed directly from an interactive COS job, batch COS job (DISPOSE/CRGRAPH), or saved in a CMS minidisk file and displayed later with the Station CRDGRAPH (display graphics) command.
- Dispose to CMS minidisk. A new command, CRDISK, allows VM Station users to receive COS datasets disposed to their virtual machines directly to CMS minidisk files.
- Virtual machine autologging. In conjunction with dispose to CMS disk, the VM Station enables users to autolog other virtual machines via the COS DISPOSE statement.
- User-callable translation tables. The VM Station provides six translation tables for use during VM-COS data transfers. It also gives users the option of applying their own tables to these transfers.
- Help on error messages. Users can get on-line help on each message the Station issues.
- Machine-readable documentation. The VM Station manuals are produced with IBM's Document Composition Facility (DCF). The DCF files (both SCRIPT input and SCRIPT output) are shipped with the Station software, enabling sites to tailor the information and print their own versions.

In addition, the new Station release includes the following enhancements:

- Improved performance. The ratio of total CPU time to virtual CPU time used by the Station has been reduced from 4.0 under release 2 to 2.8 with this release due to tracing and I/O improvements. The Station's maximum average transfer rate is 2.25 Mbyte/sec (Cray disk to virtual memory).
- Expanded status display features. A new command allows users to submit jobs to COS while monitoring job status. Status displays can now include Station diagnostics and statistics on NSC HYPERchannel activity and COS resources.
- Improved NSC HYPERchannel support. The Station now provides error recovery for NSC HYPERchannel timeouts and enhances the performance of the HYPERchannel interface by eliminating an I/O interrupt and an SIO operation.
- Enhanced magnetic tape support. The VM Station dataset transfer facilities now process a wide range of IBM tape file parameters, including IBM standard label support.
- This release also includes new security features, automatic dataset routing, and expanded on-line help.

DEC VAX/VMS Station

The latest release of the VAX/VMS Station Software Service provides users of remote DECnet nodes with enhanced access to CRAY X-MP and CRAY-1 computer systems. Remote users of Digital Equipment Corporation's VAX-based DECnet networks can now submit, monitor, and manipulate Cray jobs and datasets as if directly linked on an attached station. Additional enhancements include a redesign of Cray context, COS master operator facilities, and a configuration management utility that makes the Station software easy to install, monitor, and modify. Version 3 of the VAX/VMS Station links version 1.13 or later of the COS operating system and version 4.0 or later of the VAX/VMS operating system.

CDC NOS Station

The CDC NOS 1.15 Station Software Service continues to provide users with easy access to CRAY X-MP and CRAY-1 computer systems running COS from a Control Data Corporation front-end system running on the CDC Network Operating System (NOS). The major feature of the 1.15 Station release is dual-state support, which provides immediate access to the Cray computer system from the CDC NOS/Virtual Environment (NOS/VE) using the NOS Station. This optional feature extends the capability of COS system access to installations running NOS/VE, while retaining the features of the NOS environment. Additional enhancements include improved trace facility allowing the internal trace to be written to a local file and operator commands that release held COS datasets, control streams, and control writing of the internal history trace.

CRAY X-MP I/O devices enhanced

In February 1986, Cray Research announced new advances in I/O hardware that complement the enhanced line of CRAY X-MP computer systems. A 32-million-word Solid-state Storage Device, the SSD-3I, can now be incorporated into the CRAY X-MP I/O Subsystem (IOS) chassis. All Cray SSDs provide marked improvement on I/O-bound applications and the SSD-3I is available at a lower cost than the stand-alone SSD model.

A new IOS, the IOS Model C, is also now available. It is included on all new CRAY X-MP Computer Systems, providing the necessary high-speed I/O for scientific processing. It includes parallel disk drive capabilities, I/O buffering of disk-resident datasets, high-performance on-line tape handling, efficient front-end system communication, and can accommodate the new integral SSD.

In addition, a new controller that supports the IBM data-streaming channel was announced. The new Cray BMC-5 block multiplexer controller connects with the IBM 3480 controller, allowing use of the 3480 Cartridge Tape System. Its 3-Mbyte/sec transfer rate is nearly double that of its predecessor. Besides being an option on the IOS Model C, the BMC-5 is an easy upgrade from a BMC-4 on older IOS models. For more information on Cray Research's enhanced CRAY X-MP I/O devices, contact the nearest Cray Research sales office.

DD-49 available on CRAY-2 systems

The DD-49 Disk Storage Unit is now available for the CRAY-2 computer system. The new DD-49 Disk Storage Unit provides CRAY-2 computer users with an opportunity for significant performance improvements. The DD-49 has a 1200-Mbyte (formatted) data capacity and its data transfer rate is over twice as fast as its predecessor, the DD-29. The drive is capable of reading or writing at a burst rate of 12 Mbyte/sec. The sustained rate exceeds 9.6 Mbyte/sec.

A software driver for the DD-49 is available under the Cray UNICOS operating System. UNICOS software enhances the performance of the DD-49 with zero latency data buffering and disk striping. The driver supports all current features of UNICOS including disk striping. Up to 32 units may be configured in a DD-49 striped group. In addition, a multilevel recovery algorithm provides effective and tolerant error recovery for the storage unit. For more information on the DD-49 Disk Storage Unit for the CRAY-2 computer system, contact the nearest Cray Research sales office.

Recent Cray system installations

In the past few months Cray Research has installed several CRAY X-MP supercomputers at both old and new customer sites. At the close of 1985, a system was installed at AT&T Bell Laboratories, located in Murray Hill, New Jersey, to augment existing computing facilities. Bell Laboratories first installed a CRAY-1 in 1979.

Two CRAY X-MP systems were installed for use by the National Aeronautics and Space Administration (NASA). A CRAY X-MP/24 computer system with a Solid-state Storage Device (SSD) was installed at NASA's Lewis Research Center in Cleveland, Ohio, earlier this year. In addition, a CRAY X-MP/48 computer system including a 128-million-word SSD was installed at the NASA/Ames Research Center, Moffett Field, California, during the first quarter. ZeroOne Systems, Inc., which provides NASA/Ames with supercomputing capability, ordered the system to replace its CRAY X-MP/22 system previously installed at NASA/Ames. The CRAY X-MP/48 system provides additional supercomputing capability for research in such areas as computational fluid dynamics, physics, chemistry, and aeronautics.

Among the newest Cray Research customers are the University of Texas (UT) system and Apple Computer. A CRAY X-MP/24 computer including an SSD is scheduled for installation at the UT Balcones Research Center in Austin, Texas during the second quarter of 1986. The supercomputer will be at the heart of a center for high-performance computing being established by the University of Texas system. The system will be used for basic and applied research by all 14 campuses in the UT system.

In February 1986, Cray Research announced that Apple Computer, Inc. ordered a CRAY X-MP/48 supercomputer. The system will be installed at Apple's headquarters in Cupertino, California, in the spring of 1986. It will be used to simulate hardware and software architectures and to accelerate new product development. "To remain a technology leader in the personal computer industry, we must be willing to make significant investments in research and development," said Delbert W. Yocam, Apple's executive vice president of product operations. "With the CRAY X-MP supercomputer and its powerful simulation capabilities, we will be able to evaluate a greater variety of future product directions."

1985 at a glance

For Cray Research, 1985 was a year of great change — major new product announcements, progress in engineering and development, a broadening marketplace, and facility expansion. It was also a year of great success and building for the future, as a brief look at the year's accomplishments indicates.

The marketplace continued to broaden as supercomputers became economically justifiable for a widening range of customers. In 1985, Cray Research installed 28 new systems and reinstalled six systems. The 28 new systems installed in 1985 included two CRAY-2 systems, 13 uniprocessor CRAY X-MP systems, nine X-MP dual-processor models, and four X-MP four-processor systems. These 28 systems represent 55 central processing units; by comparison, the 23 new systems installed in 1984 represented 38 central processing units. One of the new systems was installed internally: a CRAY X-MP/216 computer system at the company's computer center in Mendota Heights, Minnesota for software development and marketing support. A CRAY-1/S was installed for computer aided design work in Chippewa Falls, Wisconsin.

The company passed a milestone in March when it shipped its 100th system. By year-end, the cumulative number of systems installed stood at 115. These systems were installed at a total of 79 customer sites, also a record. In 1985 alone, Cray systems were installed at 19 new customer sites. This trend is expected to continue, as the company received 39 customer contracts in 1985, 19 of them with organizations that have never before done business with Cray Research.

The broadening diversity of Cray Research is apparent in its expanding technical development efforts. The company invested almost $50 million in a wide variety of engineering and development activities during 1985 — an amazing statistic considering that it took just $8.8 million to make the first CRAY-1. Among other things, this money was allocated to five independent hardware development projects: ongoing engineering efforts on the CRAY X-MP series under the direction of Dean Roush and the CRAY-2 computer system under Jerry Brost, plus advanced development projects headed by Les Davis, Seymour Cray, and Steve Chen.

The five computer development projects are supported by two separate in-house semiconductor efforts. The first is a silicon group currently producing ECL gate array circuits at the 2500-gate level and working to reach the 7500- to 10,000-gate level. The second group is concentrating on gallium arsenide circuits for Seymour Cray's CRAY-3 program and currently is fabricating circuits in the 300- to 400-gate range with switching times as low as 120 picoseconds.

In 1985, Cray Research demonstrated its ability to improve the performance of its existing products and to pioneer new technologies. In announcing enhancements to the CRAY X-MP Series of Computer Systems, the company set a new standard for supercomputer price/performance. And in unveiling the CRAY-2 computer, Cray Research moved forward the state of the art in overall supercomputing performance and capability. The company also announced several input/output products and enhancements in 1985, including new price/performance levels on the Solid-state Storage Device (SSD), new disk storage devices, and a variety of system interfaces.

In addition to these hardware efforts, many development teams are working on new software for all models of Cray supercomputers. During 1985 the company demonstrated and began customer installation of UNICOS, its operating system based on the UNIX System V operating system (see this issue's related article). At year-end, a new-generation FORTRAN compiler was also nearing completion. Other ongoing software projects are concerned with incorporating Cray systems into many customer network configurations.

As the marketplace expanded, Cray Research installed its first systems in several new industry areas. During 1985, the company installed at Fairchild Camera and Instrument (Gate Array Division) the first supercomputer dedicated to semiconductor development. At year-end the company was nearing completion on its first installation in the chemical industry at the DuPont Company. Cray Research also received its first order from the medical research industry when the National Cancer Institute signed a contract for a CRAY X-MP computer system to be installed in 1986.

Several U.S. universities ordered Cray systems during 1985. Cray Research products were chosen for national supercomputer centers at the University of Illinois and the University of California, San Diego — centers established with National Science Foundation funding to broaden research in the use and development of supercomputers. In addition, an affiliate of the University of Minnesota, Research Equipment, Inc., ordered a CRAY-2 system, providing the university's Supercomputer Institute and other researchers access to a new computational tool. And, as mentioned elsewhere in this section, the University of Texas system ordered a CRAY X-MP computer to be used for basic and applied research by all 14 campuses in the university system.

Facility expansion was extensive for Cray Research in 1985, as the company's manufacturing, engineering, and development square footage was increased 60 percent from approximately 314,000 square feet to 526,000 square feet. In mid-1985, a major portion of manufacturing operations was moved to a 100,000-square-foot facility in Chippewa Falls, and the previous manufacturing facility is now devoted to system test activities. An adjacent 26,000-square-foot building was added to house manufacturing support functions.

A 48,000-square-foot development and production facility for the CRAY-2 computer system was also completed in Chippewa Falls. The building vacated by the CRAY-2 group now houses Seymour Cray's team for CRAY-3 development work, where remodeling was underway at year-end to add a clean room and additional equipment.

Cray Research's printed circuit board facility in Chippewa Falls was doubled in size in 1985 to over 26,000 square feet to meet production needs. Construction was in progress at year-end on a 19,000-square-foot addition to be completed in July 1986. Additional checkout bays were added in the engineering building, and a machine shop was established. The module assembly facility in Rice Lake, Wisconsin was tripled in size to 16,000 square feet to handle increased module work and to take on some wiring jobs. In Mendota Heights, a new 34,000-square-foot marketing services building was added for applications, benchmarking, and other marketing support personnel. And a 44,000-square-foot technical operations facility was built in Chippewa Falls to house field support, hardware training, and diagnostics.

All in all, Cray Research is a very different company than it was at its founding in 1972. But one constant has held true since the beginning: the company still has the simple goal of building the fastest, most powerful computers there are. It's nice to know that some things never change.

Close, but no submarine

The clue was: "Nicknamed 'Bubbles' and looking like a hot-tub, the CRAY-2 is the world's fastest of these." Fans of the television game show "Jeopardy!" had their supercomputer knowledge tested recently when this clue appeared under "Business and Industry for $500." The contestant who chose the category apparently was unfamiliar with high-performance multiprocessor computer systems and guessed: "a submarine."

APPLICATIONS IN DEPTH

Boeing's Petroleum Gallery makes integration an art

In January 1986 Boeing Computer Services announced The Petroleum Gallery sm, an integrated computational service for the petroleum and geophysical contracting industries. Over the Boeing Telecommunications Network, users have access to Boeing Computer Services' wide range of computing equipment, including a CRAY X-MP/24 computer system.

"The Cray system is ideal for the type of large-scale problem solving commonly associated with oil and gas exploration and production, including seismic studies, contour mapping, and reservoir modeling," explained Joe Sturdevant, director of petroleum industry marketing at Boeing Computer Services.

The Petroleum Gallery's software is available for use with IBM personal computers, although it will soon be available for use with other personal computers, and with a wide variety of peripheral devices. The Cray system is available for analyses requiring much greater computational power.

The Petroleum Gallery supports exploration and production activities including:

- ☐ Seismic studies
- ☐ Contour mapping and fault location
- ☐ Exploration drill planning
- ☐ Development drill planning
- ☐ Production management
- ☐ Reservoir modeling and simulation
- ☐ Investment analysis.

"System transparency is a key concept behind the service," Sturdevant added. "In essence, the Petroleum Gallery user will drive a multimillion dollar computer system from a desktop personal computer."

The first Petroleum Gallery release includes Petroleum Information Corporation's Petroleum Information On-line data bases and analytical reservoir modeling software from Scientific Software-Intercomp (SSI), J.S. Nolen and Associates, and Exploration Consultants Limited. In addition, the Compagnie Generale de Geophysique (CGG) has made its one-pass three-dimensional depth migration technology available so that CGG's North American clients can process seismic data on Boeing Computer Services' CRAY X-MP/24.

For more information on Boeing's Petroleum Gallery, contact Donald G. Covert, 3320 East Shea Blvd, Suite 225, Phoenix, Arizona, 85028; telephone: (602) 996-5898.

New applications directory available

Revision E of the Cray Research Applications Software Directory is now available. The 362-page directory lists over 300 applications programs available for use on Cray computers. Previously and newly available programs are included for the following application areas:

- □ Structural and mechanical engineering
- □ Electronics and electrical engineering
- □ Nuclear engineering and power
- □ Computational fluid dynamics
- □ Petroleum exploration and production
- □ Chemistry and biotechnology
- □ Miscellaneous science and engineering
- □ Graphics and imaging
- □ Mathematical and statistical libraries, simulation and mathematical programming, languages, and tools.

Separate from the complete directory, individually bound sections are available for each of the preceding categories. A bound section containing only the index, supplier cross-reference appendix, new product forms, and order forms is also available.

Current plans call for the directory to be revised semiannually. Revised versions will include newly available programs and updated information on those programs previously available. For more information about revision E of the Cray Research Applications Software Directory, or to obtain a copy, contact your nearest Cray Research sales office. If you have an applications package you would like to see listed in future directories, or have questions about an available package, contact Trudy Sprague at Cray Research's applications department, 1333 Northland Drive, Mendota Heights, MN 55120; telephone: (612) 681-3668.

DISCO available on Cray computers

The recent conversion of Digicon Geophysical Corporation's DISCO seismic software package to CRAY-1 and CRAY X-MP computers makes available a popular seismic data processing package previously available only to users of conventional mainframes. DISCO is a modular interactive software system combining the DISCO basic package with other advanced packages.

The most outstanding feature of the system is the DISCO Seismic Monitor, a control program that coordinates the execution of the system's individual modules. (Each seismic processing function is performed by a separate module; modules are individually written and requested by the user.) The Seismic Monitor includes the DISCO MACRO. A MACRO is created when the modules for a combination of programs are named and entered into the system. A disk-stored MACRO saves the user the time it would take to rewrite lengthy input for frequently-used processsing sequences.

The DISCO Basic Package includes the following features:

- □ Data reduction — demultiplex, gain recovery, vibroseis correlation, editing, resampling, and vertical and horizontal stack
- □ CDP sort — capability of handling two-dimensional, crooked line, and three-dimensional geometry
- □ Input/output — users can read or write to tape or disk in DISCO internal format or any other standard or general format
- □ Data enhancement — T.V. filter, T.V. deconvolution, T.V. scaling, muting, spectra analysis, and F-K analysis
- □ Velocity analysis — constant velocity stacks or gathers and dip-corrected coherency contoured velocity analysis
- □ Residual statics — automatic residual statics, common surface point residual statics, and non surface consistent statics
- □ Plotting — flexible plotting package using high-speed hardware rasterizers and supporting many well-known plotters.

Advanced packages include programs for migration, advanced velocity analysis, wavelet processing, signal enhancement techniques, three-dimensional processing, refraction statics, and vertical seismic profiles.

DISCO is an easily transportable system requiring no modifications to the computer operating system. All of the applications software is written in FORTRAN 77. For more information on DISCO on Cray computers, contact Digicon Geophysical Corporation, 3701 Kirby Drive, Houston, TX 77089; telephone: (713) 526-5611.

Database management system converted to Cray systems

The database management system SIR/DBMS is now available on the CRAY-1/S and CRAY X-MP computer systems. SIR/DBMS is a proprietary system developed by SIR, Inc. of Skokie, Illinois. The code is used for database management primarily by large research-oriented organizations and universities.

The ease with which SIR/DBMS handles large, complex databases makes it an ideal database management system to run on Cray computers. The availability of SIR/DBMS on front-end machines commonly used with Cray systems, such as the DEC VAX 11/780, provides flexibility to organizations with an interest in distributed CAD/CAM and other technical applications requiring portability of databases.

SIR/DBMS capabilities available on Cray systems include:

- □ Active data dictionary for complete database integrity, security, documentation, and control
- □ Comprehensive procedural programming and database access language for programmer productivity and applications development
- □ Two report-writers for simple or customized requirements
- □ Camera-ready tabular display facility
- □ Basic statistics and plotting capabilities
- □ Interfaces to statistical packages for more advanced analyses
- □ Built-in text editor to facilitate program development
- □ User-created procedure library for maintaining and managing applications.

For more information on SIR/DBMS, contact the applications department, Cray Research, Inc., 1333 Northland Drive, Mendota Heights, MN 55120; telephone: (612) 681-3652.

USER NEWS

CRAY-2 computer renders architecture in 3-D

The classic tools architects use to display their ideas — two-dimensional drawings and three-dimensional miniatures — are valuable for the information they convey about a proposed structure or a change in landscaping. By studying drawings and physical models, a client can get some idea of the final appearance of an architectural project. But the idea conveyed by these models is often crude, and basing decisions on them is necessarily a gamble. Rather than play the odds, however, architects can combine computer graphics and supercomputer power to convey their ideas precisely.

Computer graphics applied to architectural rendering provides a way for architects and clients to view proposed structures from any distance, any angle, any elevation, and in any lighting conditions. Computer graphics can also reveal the appearance of a proposed structure in the context of what would be its surrounding environment. Although clearly a more versatile and accurate display tool than drawings and miniatures, computer graphics used for architectural rendering relies on the manipulation of very large databases — a task prohibitively time-consuming on conventional computers with small memories.

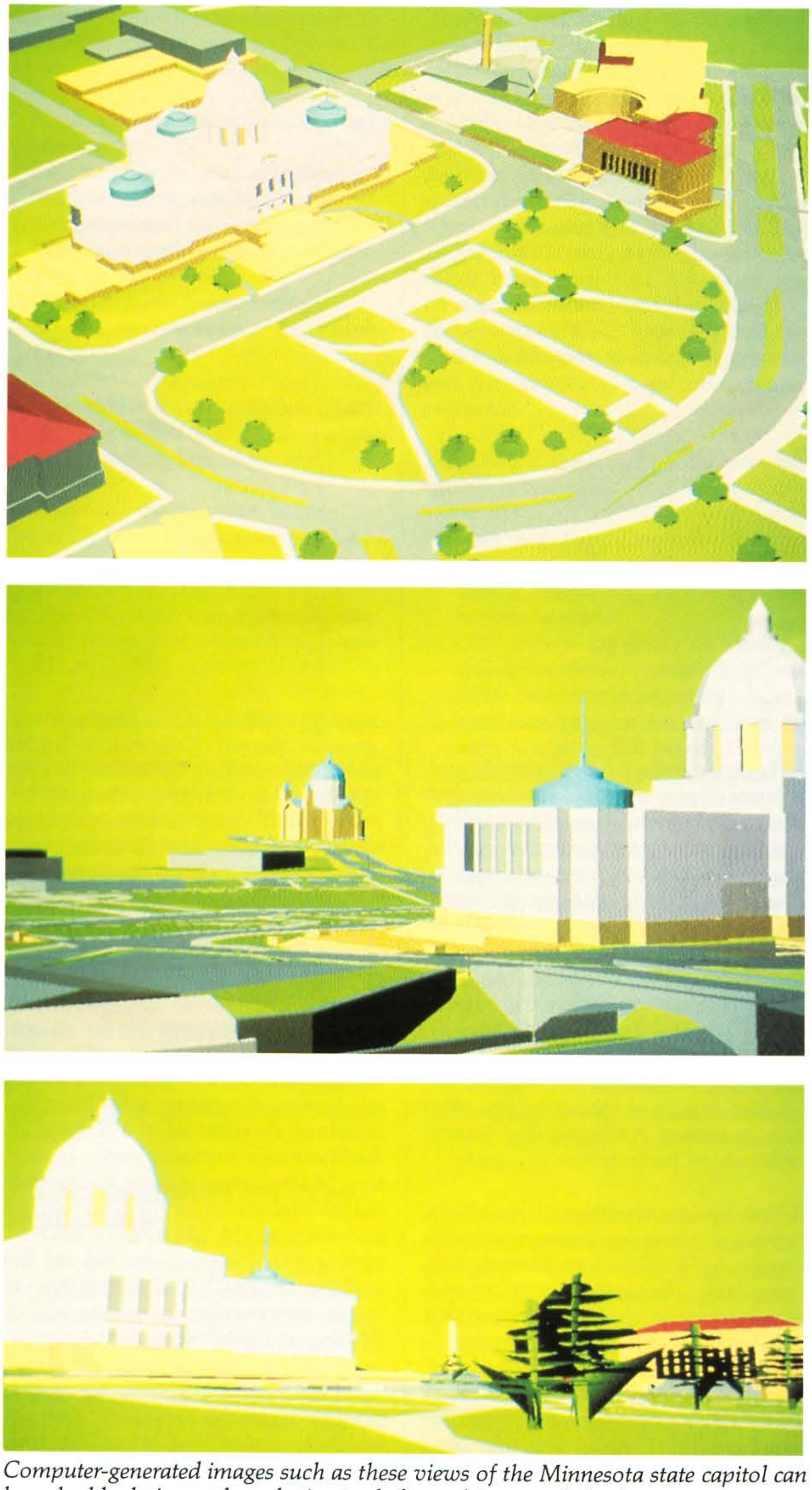
However, a group of researchers at the Univeristy of Minnesota's School of Architecture have successfully bypassed this problem. Using a single-processor CRAY-2 system at the university, they have generated renderings for several architectural projects.

"Once the data are digitized, an infinite number of views can be generated with a computer," explained graduate student Peter Hill. "Because no pre-defined parameters exist, the architect can ask a client, 'How many details do you want?' Computer graphics also provide a lot of feedback, enabling the architect to build a 'walk-through' model and view the structure from inside. The time and work required make it impractical to do this with drawings."

The group's most ambitious computerized project to date is a series of renderings of the Minnesota state capitol and surrounding area. The digitizing of the capitol grounds was commissioned by a state agency to promote a landscaping competition. The agency is soliciting ideas for developing the large empty parcels of land near the capitol.

"The original database for the capitol was developed for a PC-based system, which worked well for data input and low-level manipulation," said Oliver Ng, assistant director of the School of Architecture's microcomputer laboratory. "As the database grew, we transferred the problem to an Apollo computer and then to the university's CRAY-1 system. But it wasn't until we put the problem on the CRAY-2 system that we had enough memory to consider adding shading and shadows to the pictures."

The CRAY-2 system's speed also made it possible to generate two short movies in which the viewer "flies" around the area, viewing it from many angles and elevations. The images were created on the



Computer-generated images such as these views of the Minnesota state capitol can be valuable design and marketing tools for architects and developers.

CRAY-2 computer with the graphics package MOVIE.BYU. The program was modified to run on the CRAY-2 system by Scott Gaff, a programmer at the University of Minnesota's Supercomputer Institute.

The Institute granted the group eight hours of CPU time on the Cray computers, much of which will be used to refine the model and to render ideas from entrants in the landscaping competition. The versatility and detail computer graphics provide make it an ideal medium for presenting architectural ideas. And Cray computers are ideal for manipulating large data sets repeatedly to generate numerous detailed images.

The new improved universe

The large-scale structure of the universe, at least the largest structure so far observed, consists of long strands and flat "pancakes" of superclusters of galaxies. This cellular structure is believed to be the product of matter, originally sprayed out of the Big Bang, being shaped by gravity working over many billions of years. Joan Centrella, an astrophysicist working at Drexel University in Philadelphia, has been trying to account for the large-scale structure of the universe, according to what is known, or surmised, about the nature of matter and the way gravity acts on it.

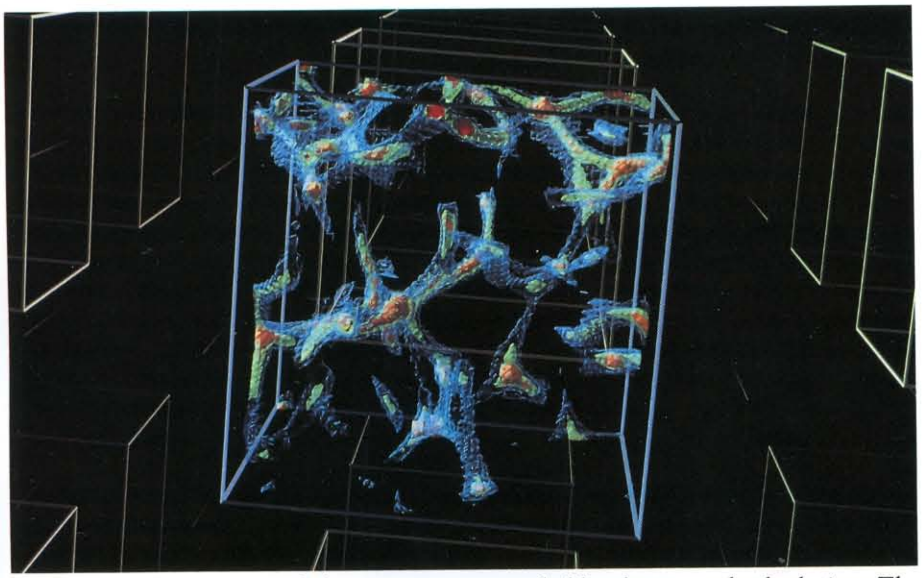
Her primary research tools are equations, algorithms, Cray computers, and computer graphics devices. By simulating the evolution of the universe's large-scale structure computationally, Centrella hopes to verify or refute theories about the types and distributions of matter in the universe. (The work has been ongoing for several years and was described in CRAY CHANNELS Vol. 6, No. 4.)

Crucial to interpreting the results of Centrella's work is the quality of the graphic displays. Centrella recently received a grant from the National Science Foundation to work with Digital Productions, Inc., a Los Angeles company specializing in computer graphics applications, to improve the graphic presentation of her results. Using Digital Productions' CRAY

X-MP computer system to generate the data and Digital Productions' expert staff to help process the results, Centrella was able to create a film, to be aired this year in the United Kingdom, that dramatically shows the coalescence of the universe's matter into the large-scale structures observable today. The results are significantly more understandable, and more dramatic, than previous displays. "It's fantastic," Centrella said. "I'm very pleased with the images."

A variety of graphics techniques were used to process the data, perhaps the most significant being the use of complex polygons to build the images, rather than the simple cubes used in previous work. Over 100,000 polygons were generated per frame of film. The supercluster shapes were color-coded to indicate increasing density towards the center of each filament, and many of the images used several layers of transparency.

"The ability to create these types of quality images will become increasingly important as scientists do more complicated, time-dependent simulations," Centrella commented. "The combination of supercomputers and the kind of technical and artistic expertise there is at Digital Productions can make significant contributions to scientific research."



The large-scale structure of the universe as revealed by theory and calculation. The contents of the box show a representative sample of the structure that extends in all directions. In fields such as astrophysics, where direct experimentation is not feasible, computer graphics can help researchers test and refine their theories.

Models track radiation for safety's sake

Government and industry operations that handle radioactive materials go to great lengths to ensure public safety. But to formulate policies that will best safeguard the public, it would be useful if the potential effects of radiation exposure levels were quantified. To derive such information, the Los Alamos National Laboratory in Los Alamos, New Mexico has initiated a computer modeling project. Laboratory researchers are developing and refining models that simulate the transport of radioactive particles and their uptake in bodily organs, the results of which can then be used to aid environmental impact studies. The researchers' models — BIOTRAN, HUMTRN, and EFFECTS — take into account thousands of pieces of information about ecology, human physiology, and population dynamics. Thanks to the speed and storage capacities of the laboratory's Cray computers, the researchers can extend simulations of individuals to cover entire populations.

BIOTRAN, the ecosystem model, was developed in the mid-1970s and incorporates knowledge about climate, soil, and plant and animal growth. Its initial use was to model the transport of radioactive particles under- and aboveground for waste management risk analysis studies. After several modifications it was coupled with a human physiology model, HUMTRN, in 1984.

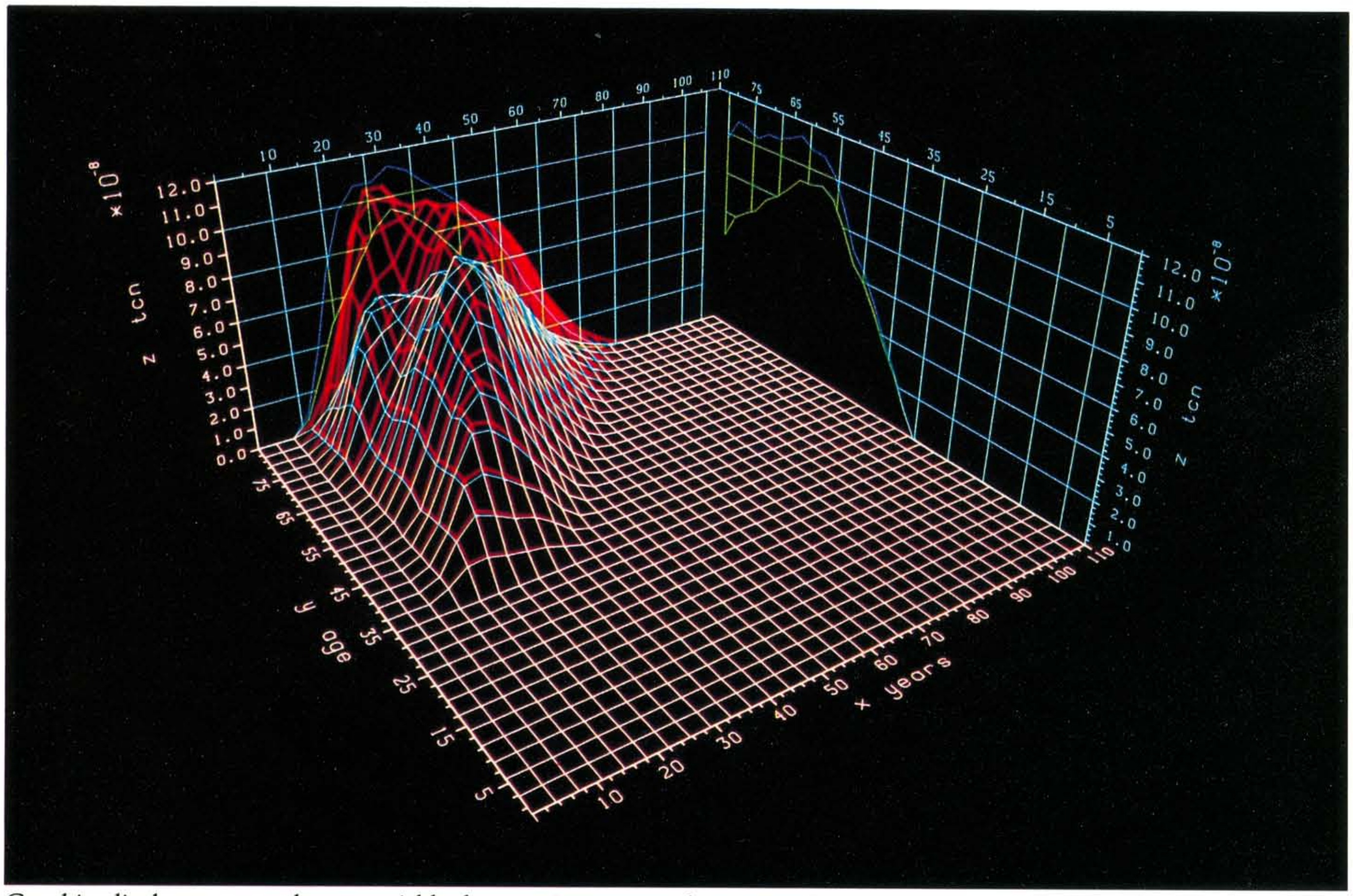
In HUMTRN, organs are represented as varying proportions of carbohydrates, proteins, and fats. The model receives its input in the form of simulated foodstuffs — programs that also specify various amounts of carbohydrates, proteins, and fats. HUMTRN "ages" individuals by varying their organ sizes, thus making the simulated population a dynamic one that changes over time. The model provides BIOTRAN with simulated air, water, and foodstuffs to be followed as intakes and uptakes in humans.

EFFECTS, a risk assessment code, was developed concurrently with BIOTRAN and HUMTRN to estimate the impact of simultaneous acute and chronic radiation exposures on a population. EFFECTS estimates cancer risks for the HUMTRN population based on burdens to the bodily organs of individuals of varying age and of either sex. Combined BIOTRAN-HUMTRN-EFFECTS simulations can be run in timesteps representing days or years. In certain cases, the combined model is used to assess environmental risks before the Los Alamos National Laboratory expands its facilities or implements new materials handling processes.

"About two years ago the model outgrew the capacities of our other mainframes, so we moved it to the Lab's CRAY-1 computers," explained Jay Wenzel, a health physicist at the laboratory and member of the team that developed the models. "Using the CRAY-1 systems enables us to study population dynamics by simulating a million individuals in one run. We can also run models covering hundreds of years, allowing us to do multi-generational studies."

Wenzel and colleague Anthony Gallegos, dean of graduate studies at New Mexico Highlands University in Las Vegas, New Mexico, have run simulations to validate the models. One validation test compared the simulated impacts of natural background radiation with those of an

USER NEWS



Graphics displays can turn large unwieldy data sets into images that are much easier to interpret. This graph from the EFFECTS model shows absolute risk for radiation-induced lung cancer deaths for a 20-year continuous uranium exposure plus lifetime-burden. Risk for males is shown in light blue, for females in red.

accidental release of airborne radioactive particles on a small population. For the long-term continuous exposure to natural background radiation the impact on the population was very small, with young adults receiving the largest bone doses and risks. For long-term intakes following the simulated air release, young teenagers received the highest bone dose while young adults received the largest risk. Wenzel says the differences show that the combined model has enough resolution to predict age- and sex-specific responses of populations to environmental contaminants. This ability enhances the precision with which environmental impact studies can be carried out.

Along with assessment applications, the model has potential uses in research and education. Wenzel points out that if residual radiation levels at a contaminated site can be measured then the model can be run "backwards" to determine initial exposure levels. "As for education," he adds, "the environmental simulations are used to teach principles of ecology. As part of the laboratory's outreach program we offer a semester-long course to gifted high school students with biology and math backgrounds that involves using the Cray computers to run ecological simulations. The Air Force Academy in Colorado Springs has also expressed an interest in using the model this way."

As is common with supercomputer applications, the model produces data sets so large that they must be displayed graphically to be interpreted. To meet this need, the researchers use the three-dimensional color plotting and charting package DISSPLA from Integrated Software Systems Corporation (ISSCO) of San Diego, California.

"Showing the output from these models graphically lets us rapidly understand and verify complex scenarios," Gallegos said. "The CRAY-1 computers enable us to plot 50,000 to 60,000 points on a grid."

Current efforts are aimed at revising the codes to allow multi-animal, -plant, and -atom simulations. "Running the model with several atoms may reveal synergistic effects that would otherwise be hard to predict," Gallegos explained. In addition, the models have spawned a family of related software. Among the members are PLTGRO, a plant model; RMNANT, a beef steer or cow model; and AQAT, a deep-water algae-producing lake model. All of the codes used in the research are publicly available. For information, contact National Technical Information Service, U.S. Department of Commerce, 5285 Port Royal Road, Springfield, VA 22161.

WORLDWIDE BUSINESS CENTERS

Domestic

Corporate Headquarters
608 Second Avenue South
Minneapolis, MN 55402
(612) 333-5889, telex 290789

**Government Marketing/
Government Relations Office**
National Place
1331 Pennsylvania Avenue, Suite 1331 N
Washington, DC 20004
(202) 638-6000

Central Region Office
5350 Manhattan Circle
Boulder, CO 80303
(303) 499-3055
- Albuquerque
- Chicago
- Detroit
- Minneapolis/St. Paul
- St. Louis

Eastern Region Office
11710 Beltsville Drive, Suite 500
Beltsville, MD 20705
(301) 595-5100
- Atlanta
- Boston
- Huntsville
- Laurel
- Pittsburgh
- Rochester
- Tampa

Petroleum Region Office
5847 San Felipe, Suite 3000
Houston, TX 77057
(713) 975-8998
- Dallas
- Tulsa

Western Region Office
5776 Stoneridge Mall Road, Suite 350
Pleasanton, CA 94566
(415) 463-2800
- Los Angeles
- Seattle

International

Cray Canada, Inc.
4211 Yonge Street, Suite 210
Willowdale, Ontario
Canada M2P 2A9
(416) 229-2729, telex 2296238

Cray Research France S.A.
7 Rue de Tilsitt
75017 Paris, France
(01)-766-01-55, telex 660568

Cray Research GmbH
Perhamerstrasse 31
8000 Munich 21, West Germany
(089)-56014-0, telex 5213211

Cray Research Japan Ltd
W 1154, Shin Aoyama Building
1-1 Minami-Aoyama 1-chome
Minato-Ku, Tokyo 107, Japan
(03)-403-3471, telex 28893

Cray Research S.R.L.
Via Vivaio, 15
20122 Milan, Italy
(02)-70-98-27, telex 315403

Cray Research (UK) Ltd.
Cray House
London Road, Bracknell
Berkshire RG12 2SY
United Kingdom
(0344)-485971, telex 848841

CRAY RESEARCH, INC.