

## Time Warp: Stardate 1984

By Kelley M. Essoe

***Future Shock: Auricle ushers in a whole new age of synergistic operating systems for computers... and computer users. So get ready to "beam up" to the future.***

Remember the computer system aboard the Starship Enterprise? Whenever the venerable Captain Kirk found himself in a tight spot and in urgent need of specifics on some glob-like invading life form or immediate strategic advice to help thwart the Klingons' endless assault on the rest of the God-fearing universe, he would consult the ship's computer.

"Computer, specifics on the glob-like invading life form, please." And in the sexiest voice this side of Marilyn Monroe the computer would immediately respond with all the slimy details,

"How about a history of Klingon Commander Boy George?", Kirk asks, changing the subject. And without a blink of her mascaraed memory chips or the audacity to refer him to a "main menu" in order to access another database, the computer would promptly disclose the complete and kinky life chronicle of Commander Boy George. Whatta gal, that computer.

But then, of course, she is an extremely advanced cybernetic organism of the 25th century. Certainly by then our advancements in computer technology will, at the very least, allow us the kind of

quick and simple repartee that Captain Kirk enjoyed with his computer.

He could communicate with her in English, Wonderfully handy. He could go from one order of business to another at the spur of the moment without blowing her dainty little circuits. That's nice. He could concurrently talk to her while she was talking to him, interjecting comments, requests and even changing the subject without having to either politely wait for her to finish or rudely shutting her up by slapping her sharply on her run/stop key. Now that's nifty. Best of all, he could access all her capabilities and knowledge instantly, without even so much as a fleeting glance at one single menu. All he had to do was ask and, on the spot, she delivered. (Which is more than the Post Office does right now and they have legs.)

That's what I call random access. What wouldn't we give to have a whack at her?

Unfortunately, the bad news is we don't have a chance in Hades of living to meet her there in Stardate 43 point 6.

The good news is... we don't have to, Richard Grant, lawyer and computer hacker



Photo by Laurel Bledsoe

Richard Grant (r.), creator of Auricle, and his brother, television composer Ron Grant.



Screen photos by Richard Benjamin Grant

Auricle makes full use of the Commodore 64's graphics and color capabilities.

*par excellence*, introduced me to her great-grandmother the other day.

Her name is *Auricle* and she's every bit the liberated lady that her great-granddaughter on board the Enterprise will be. *Auricle* is, of course, quite young and has not yet matured to her full capacity. Nevertheless, her staggering potential is immediately obvious. It's like being introduced to a future queen. She's not wearing the jeweled crown yet, but she reeks of power and royalty. It's merely a matter of time.

At the moment she resides at the same address in which she was born: within a brand new, and in itself remarkable program for music composers called *Auricle: The Film Composer's Time Processor*, co-conceived by Richard Grant and his brother, composer Ron Grant, using a Commodore 64.

*Auricle* came into existence through the determination of the Grants to have their time processing program do exactly what they wanted it to do — as opposed to what the computer's operating system would or would not allow it to do.

Ron Grant, one of the alternating composers for the night-time television drama "Knotts Landing", grew weary of the painstaking and laborious task of figuring out the crucial split-second timing involved in film scoring. A visionary of sorts, he knew there had to be a better way.

To a composer, movies or television are an endless stream of fractional seconds, each needing to be complemented by the mood-setting accompaniment of music. Throughout a film the music, like a shadow, must build and break in total unity with the action taking place on screen. Ideally, the musical tempo should speed up here, slow down there, imperceptibly snaking its way around in order to coordinate exactly with specific "hits", or

dramatic points, within the storyline. But, until now, the composer had to find and settle for the most acceptable average within the rigid and uniform bounds of metronomic tempo. It's a risky business, for if not timed precisely, these hits quickly become misses, and the music becomes a premature clue or a tootlate redundancy, either way standing out like a wart on the end of one's nose.

Sound synchronization is calculated by counting the frames and sprockets on a piece of film. Movie film is simply a long ribbon of single frames bordered on two sides by a chain of little square holes, or sprockets. The film itself looks like a series of separate photographs, which is exactly what it is,

The speed at which the film races by in order to create the illusion of "real-time" motion is 24 frames per second. Accordingly, the traditional unit of time-frame measurement is 24 frames. Each frame has eight sprockets, providing the sub-units to mark off fractions of frames. Therefore, a film composer is actually juggling three separate modes of time measurement: "real" time, frame-sprocket time, and metronomic time.

Take, for example, a hit occurring at two minutes and 42.7 seconds in real time. The composer must first convert the musical tempo that he wants to use into a frame count — usually some fraction of the standard 24 such as 19-4, or 19 frames plus 4 sprockets. From this working rate of cadence, he then has to adjust the music, fine tuning the tempo so that at precisely two minutes and 42.7 seconds the steady pulse of music climaxes in synchrony with the action taking place on screen.

So, like David valiantly facing Goliath with his slingshot, the composer tackles this behemoth of time armed with his hand cal-

culator, "click track" book, digital metronome, paper, pencils and Excedrin. When he first attacks this monumental task, if he's lucky, the frames of film, the tempo, the mood and the "hits" he has to make will all line up symmetrically. Does this happen often? Sure. About as often as Mr. Spock giggles.

Charles Bernstein, who composed the score for Columbia's acclaimed motion picture "Sadat", struggled over one particularly difficult "hit" in the picture. The cue itself was a not overwhelming 30 bars in length — just slightly longer than one verse of "Yankee Doodle." Nevertheless, the timing was irregular, ranging from extremely rapid to abysmally slow in a matter of a beat or two.

Three hours later he finally had it worked out.

Earlier this year, Bernstein got a call from friend and colleague Ron Grant. Ron wanted him to come over and see a new computer program that he and his brother had been working on.

"I don't know anything about computers, Ron."

"You don't have to. Just come take a look."

So he did. And though he was treading the unfamiliar ground of computer country, it was obvious that Ron and Richard's time processing program was unmistakably a major breakthrough for music composers. Bernstein was taken with the possibilities.

He soon learned that the program allows composers to effortlessly adjust the tempo of their scores to fit the visuals. It gives them complete control over every beat within every bar of music. In effect, the program "microscopes" the musical score down to its most elemental component, a single beat, giving composers total creative power over the manipulation of their adversary — time.

Bernstein found the list of "Topicals", or system capabilities, immense and thorough. They include metering, re-metering, timings, re-timings, adding hits, tuning forks, clicks, bars, beats, splits, hit cues and tempos. What more could a composer want? Or need?

Well, for one thing, computers are intimidating. To a novice, the language they speak might as well be Sanskrit. Bernstein was concerned about being able to use the program without having to take a six-month night course in computers.

Herein lies one of the Time Processor's most unique features. Like the computer aboard the Enterprise, it speaks English. If the composer wants to re-meter bar ten to 3/8, he types in "re-meter bar 10 to 3/8". If he wants the program to break down bar 20, he types in "break down bar 20". If he doesn't like the color on the CRT, he types in "re-color crt light blue". If he wants to see the list of Topicals, he types "topicals." *Auricle* is astoundingly user-friendly,

*Auricle* takes the intimidation out of it.

A second notable feature is the system's complete lack of menus. The program will deliver what you want, whenever you want it, regardless of what it happens to be doing at the time. Ask and ye shall receive, (I'll have a side of fries please).

Bernstein was spellbound.

Then Ron gave him a pointed demonstration. Using the *Time Processor*, he attacked that dastardly "Sadat" cue that had given Bernstein three hours of agony.

It took Ron all of five minutes. Bingo. Bernstein became a total believer.

Now, of course, Bernstein is a composer not a computerist. As a composer what he saw that evening was the perfect composer's tool. That knocked his socks off.

But what I saw when I visited the Grants literally swept me off my feet. It was *Auricle*— the wave of the future, the precursor of Captain Kirk's science-fiction computer — the operating system behind the *Time Processor* program,

It is *Auricle* that provides the most unusual and the most powerful aspects of the *Time Processor*. *Auricle* ignites the imagination. A prototype for a whole new generation of computer operating systems, *Auricle* could conceivably catapult us into the realm of "where no man has gone before."

Richard Grant explained why they had named their program *Auricle*. "It has to do with the fact that auricle means 'ear'... essentially the ear of the computer. What it does is, it takes and breaks the paradox of the screen. Remember Broderick Crawford in 'Highway Patrol'? He would pick up his car microphone and say ten-four! Ten-four! He couldn't talk while the guy on the other end was talking and vice versa. Well, that's the way computers work — ten-four."

*Auricle* takes the ten-four out of it.

The screen is the communications link between the computer and you, the user. It acts as *both* the computer's mouth and ear — though not concurrently. While the computer is "talking" the screen is a mouth. When it has finished speaking, the screen then becomes the ear, usually prompting you with a "?" or some such signal to inform you that it is now your turn to say something. Pressing the RETURN key is your way of signing out — ten-four — you're done with your turn and the ball is back in the computer's court.

Taking turns, as such, is all well and good. Miss Berilla taught me that in kindergarten. We raise our hand before we speak, we wait our turn at the drinking fountain, and we never, ever interrupt while someone else is speaking.

I always wondered what happened to Miss Berilla. Now I know. She's the one who designed interactive computer protocol,

Personally, I see no need for all this polite behavior with a computer. Particularly since it's so bloody one-sided. When was the last time your computer apologized for rudely interrupting your input with a "syntax error"? Or begged your pardon after a "crash"?

Dear Miss Berilla, with all due respect, sit on it.

The *Auricle* system creates a whole new communications link with your computer. Instead of one channel that must act as both mouth and ear, *Auricle* provides a second dedicated channel of interchange ... the ear. The end result of this fundamentally logical, not to mention humane, concept of giving the computer a "face", so to speak, is utterly spectacular.

You can demand of the system virtually any task it is able to perform, at any time, in English, interrupting to your heart's content. All you have to do is ask. The computer will shut up, listen and deliver,

And what a blessing that *Auricle* speaks English,, (or French, Spanish, German, even Pig-Latin), The inexperienced user can access the computer without learning a whole new language. As Ron put it, "English you remember. An assemblage of escape-control/C's takes you forever to learn and then you forget a week later."

I couldn't agree more. Let the computer be the one to go to the Berlitz School of Languages.

Put quite simply, it's a matter of who — you or the computer — has to sit and flip through a manual in order to find a specific command so that you both can get on with it. And we know who can flip pages faster.

By referring to her dictionary of "Topicals" in the *Time Processor*, *Auride* understands any of the "natural language" commands associated with the functions within a composer's province.

The *Time Processor*, however, is only one application of the *Auride* operating system. Outside the *Time Processor*, *Auride* can contain any number of similar dedicated dictionaries for other types of programs. Or even one large reference that covers a considerable scope of divergent topics.

Menus become an unnecessary, arid, obsolete "tripping stone" of the past. You simply don't need them. With *Auride*, they'd only get in the way.

What's that, you ask? No menus? NO MENUS AT ALL?? That's right. It's an Emancipation Proclamation. Free the victims of menu tyranny! No more Main Menu. No more choosing what's behind curtain number three. No more thrillingly frustrating attempts at trying to find the sub-menu that will eventually take you to your desired task.

But how will we know how we're supposed to use a program without menus to guide us? Simple. Ask *Auride*. She'll show you her list of "Topicals" or program functions. And once you are familiar with a certain program's abilities you'll never have to ask again.

Except, of course, when you want to change the semantics to fit your own needs, conventions or personality quirks.

Yep, If you prefer calling a particular function by a name other than what the software author chose to call it, change it to suit yourself. After all, it is your software, you bought it and you're the one who's going to use it. Maybe you'd rather call the "merge" function "mix it up," instead. Maybe you don't like pedantic computer vocabulary and find it easier to deal with "fill-ins" as opposed to "variable

blocks". Or perhaps you're a computerese die-hard and long for escape-control/M over simple English. Go ahead, indulge yourself. You can always change it tomorrow. You're the boss.

Richard, a man with a sense of humor, has programmed his computer to call him "Bigshot". "Whadda you want, Bigshot?" They have a wonderful relationship.

Which brings us to the essence of *Auride*.

We are in the era of "personal computing". But, until now, users have had to adjust themselves to the software. Usher in what Ron calls "personal programs". *Auride* creates an environment wherein the program adjusts to fit the user.

As Richard puts it, "We're not computers... we have a right to be confused! We have a right to not know exactly what we want or in precisely what order we're going to want it!"

*Auride* forgives us our innately haphazard thought processes. She permits us the non-linear randomness that gives birth to our creativity. *Auride* yields to our fits and starts and allows us to be — forgive me, Mr. Spock — human.

On screen, *Auride* appears as a rectangular window. A funny shape for an ear, but then beauty's only skin deep. The user can slide this window all over the screen or, when it's not needed, have it disappear altogether. Whatever you type within the confines of this window is immediately "heard" by the computer. It's like whispering in its ear. And if you want to get someone's attention... whisper.

*Auride* can literally change the face of computing as we know it, Imagine these capabilities within a database program, or a word processor, spreadsheets or business applications, Picture what it could mean to creating graphics. Consider modem communications, games and adventures. And

how about programming utilities?

Richard found that at a certain point in the writing of *Auride: The Film Composer's Time Processor*, *Auride* actually became a major programming and debugging aid.

He discovered that he could ask the computer where a specific module was, such as "Click Cue", and the computer would come back with the proper hex number. While using *Auride* within a BASIC program he was writing, he could ask "What Is Next F", and the computer would tell him where in the program he had available space — where his next function slot would be. It occurred to him that without much ado, he could literally disassemble the program *white is was actually running*

"But I haven't even begun to elaborate on that process", says Richard. "The potential is limitless."

Pretty exciting stuff. And with speech synthesizing just around the corner...

*Auride* is not yet on the market. But, within the *The Film Composer's Time Processor*, she has made her film debut.

On April 5, 1984, at MGM studios in Los Angeles, *Auride* was involved in a television first. The responsible party was Lance Rubin, a composer for the hit show "Dallas", and one of the first *Time Processor* addicts. At his scoring session for this season's final "Dallas" episode, the variable "click-tracks" from which the orchestra kept time were provided by a Commodore 64 and the Grants' program, *Auride: The Film Composer's Time Processor*.

So ready or not, here it is, the future... brought to you by *Auride*.

I'm ready. Beam me up, Scotty.

\* \* \* \* \*