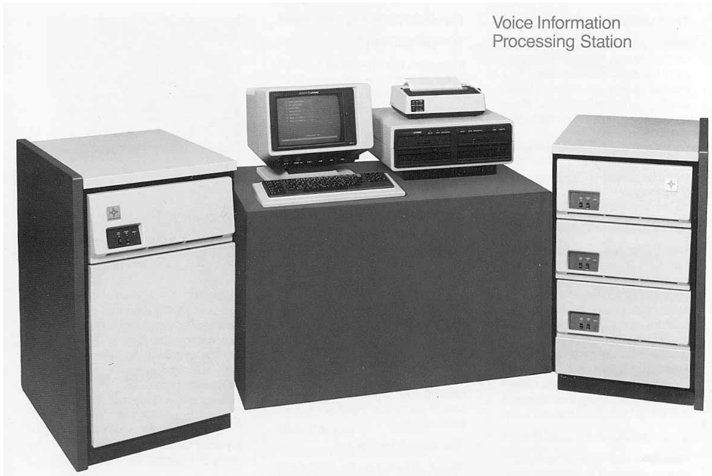
The Invention of Voice Mail©



In the early 1970s, Sperry UNIVAC Defense Systems in Eagan Minnesota, formerly a division of Remington Rand UNIVAC, formed a small research group whose primary mission was to develop a continuous speech recognition system for military applications. Such a system promised to improve operation of eyes-busy and hands-busy tasks such as inputting data to a computer while flying a helicopter or controlling a CRT display without taking one's eyes off the screen. The requirements were tough—telephone/radio bandwidth speech, no pause between words, and a noisy environment. During the 14 years of the Speech Research Group's existence, it had considerable success.

The initial leader of the Speech Research Group was Mark Medress, a PhD in linguistics and a master's in electrical engineering. Others were Dr. Wayne Lee, an expert at prosody--the stress and intonation aspects of speech, and computational linguistics. His nickname was Doctor Zero. He would later become a well-known author and consultant. I knew Hank Ordson in high school in the late 50s. He was an electronics genius even then. He was into ham radio and *designed* his own FM radio receiver for a high school science project. He saved his school a lot of money because he personally repaired the movie projectors. Hank liked to whistle tunes like J.S. Bach's *Jesu Joy Man's Desiring* while working at a computer terminal. Sometimes one of us would join in the counterpoint at another terminal. Soft-spoken Dr. Tim Diller, linguist and programmer, was a former missionary. He had served in the Philippines where he translated the Bible into Tagalog. Toby Skinner was a math major and statistics expert who wrote much of the speech analysis software (most signal processing is based on statistics). Karen Nieken, our technician, assembled and maintained most of our equipment. Her family was in the heavy construction business in St. Cloud so she had grown up around machines. She was the most mechanically inclined woman I ever met. For example, she changed the oil in the family cars except when she was 9 months pregnant and couldn't fit underneath. She was gregarious and talkative and always put everyone at ease, even big-shots. John Siebenand's college degree was in psychology plus he had trade school training in electronics. He was a former field engineer who could fix anything. He was equally at home with both hardware and software. Larry Lutton was another former field engineer with a degree in EE who loved to ski. Don Anderson was a methodical but inventive EE who designed most of our hybrid analog/digital signal processing equipment. Don and I took a course at a local community college when the first 4-bit microprocessors appeared. They were considered hobby stuff then, but the experience proved very valuable later. Dean Kloker was a quiet, studious programmer. He and Tim would sometimes play recorder duets in the anechoic chamber during lunch break. I joined them once for a trio. Clint Crosby was an earthy nuts-and-bolts hardware engineer who kept the parts coming together. My degree was in physics with previous experience in radar and sonar R&D.

There were others who were not permanent members of the group but contributed to our work. Art Olive was an organic farmer and mathematician who programmed the Fast Fourier Transform for us--it made possible real-time spectral analysis, not only in speech but sonar and other applications. Jens Peterson was an old-school technician whom Don Anderson called upon to help with circuit design. Jens learned his trade in the days of analog computing. I could tell which part was designed by Jens with one look at the logic prints—his work was loaded with op-amps. Dr. Pinky Strange was a red-haired linguist at the University of Minnesota psychology department with whom we traded data files. Jim Marver was a young and eccentric expert at Galois logic—a form of computer circuit design by algebra. He came and went at all hours, even keeping a sleeping bag under his desk for those times when he worked all night. I'm not sure what his role was--there was lots of classified work being done around us, so we didn't ask. Gene Chicoine was an extroverted former Navy officer who was a drum-beating marketeer for our stuff. Unlike me, he was a big guy with a big voice. Management called us The Overt and The Covert. Gene once claimed with great astonishment that I had improved the intelligibility of human speech with a new digital sampling technique. Actually, it just sounded better to him because I had removed a portion of the spectrum that was being distorted by his hearing damage caused by big deck guns during the Korean War. Lee Granberg helped out with our graphical displays. He invented the first computer CRT display for the Navy many years earlier from which all of today's computer CRT monitors are derived [Patents #3,466,645 and #4,081,799]. Bob Shutt was a marketing type who knew both the military and civilian customers.

Visitors to the Eagan plant were often given tours of the facilities. A talking computer was a real novelty then, so the Speech Lab was one of the most popular stops on the tour. A computer that seemed to understand what you were saying and respond by voice in a human-like way was spooky at the time. Gregarious Karen was often given the task of entertaining these "visiting firemen." She amazed visitors from third world countries in which women played no part in business or engineering. One wonders what they told their friends back home.

The Speech Group's early success led to a contract with a certain defense intelligence agency that awarded us a contract to develop an algorithm called *word spotting*. The task was to identify certain key words being spoken in an eavesdropped radio or telephone conversation. When combined with a context analysis program, it was possible to determine the gist or topic being discussed. In some ways this task is easier than understanding cooperative speech because the detection rate can be lower and still work. This program was the first Top Secret project at the Eagan plant and resulted in the construction of a special RF-shielded lab space underground which Larry and Don designed. This project, we were told, caused our plant to become a target in the USSR's ICBM system. This form of professional recognition was received with a bit of pride. The word-spotting program demonstrated the feasibility of automated real-time gisting in English and Russian on a large scale. Now, thirty years later, one wonders what our government is doing with this technology.

Our equipment was crude by modern standards. There were no microprocessors, no PCs and no Web. The minicomputer had just been developed but was used like a main frame. Too slow for high-level computer languages like FORTRAN, we coded all real-time programs in machine language, assembly code or CMS-2, the standard navy computer language. The Internet hadn't been invented yet, but we were a node on its predecessor, the ARPANET. We were the only defense contractor among the 12 sites connected at that time. All the others were universities. Hank, Dean, Larry, and John set this up, a major R&D project by itself. Data was transferred between sites by passing files from one node to another until it reached its destination—a technique known as token ring store-and-forward. Passing text files created a primitive form of email. If a node crashed, the ring stopped until it was fixed. Sometimes the US Mail was faster. But Mark's strategy was to be as cooperative as possible with other researchers in order to learn from them. It seemed to work, but I suspect they learned more from us than we learned from them.

A necessary tool in our speech-recognition research was the ability to store and play back digitally recorded speech and portions of speech sounds. This gave us the ability to break sentences into words, rearrange them and play back the result, thus creating a new sentence in the speaker's own voice. This led to another government contract called *voice deception*.

We designed and built a computer that recorded radio voice messages in real time, decomposed them into words and phrases by speaker and cataloged them. An operator could then quickly pick from a list and compose a new voice message. It worked very well--I witnessed a test of this system by the Air Force at a Red Flag exercise at Nellis AFB that fooled pilots into responding to the computer's voice instead of their own controller.

In order for speech recognition and voice deception to work well, we devised a noise-removal algorithm based upon a math process known as homomorphic deconvolution that was originally developed to clean up noisy 78-RPM phonograph records. We used it to separate pilot's speech from background sounds which are considerable in fighter aircraft. As a diversion, I applied some of our speech-recognition software to the background and found that it could identify the aircraft type or facility reasonably well. This allowed us to overlay a pilot's speech on another background to make the speech sound like it came from a different location. In addition, unintelligible speech of a second speaker in the background could sometimes be made intelligible by removing the foreground speech. When I presented these discoveries at a Navy briefing in San Diego, all of my data was immediately classified at a security level higher than I held.

An unclassified version of the noise removal algorithm resulted in an Air Force contract to help linguists. Although real-time noise removal didn't improve intelligibility much for human listeners, it reduced operator fatigue. Linguists loved it.

Not all of our projects were successful. We worked for nearly a year developing a large vocabulary text-to-speech voice response system for the National Weather Service (NWS). One entered an area code from a telephone and it responded with the three-day weather forecast. We measured its acceptance and performance with a special booth at the MSP airport. It worked very well. Later, the Weather Service granted a sole-source contract for production versions to a competitor using our system as a specification. We weren't even asked to bid. Management was so miffed that they forbid the entire division to do business with the NWS. I ran into the Weather Service contracting officer at a computer conference a year or so later. When I asked him why they didn't ask us to bid, he just smiled.

But we learned a lot about human factors from that project that served us well later. I learned, for example, that to say "You have made an error" or "Please try again" was intimidating. It was more tactful to reword the question or have the computer apologize for messing up.

The Minnesota Department of Transportation had a grant from the federal government for improving traffic flow using computers. They approached us with a traffic problem at a shopping center in a northern Minneapolis suburb. They asked us to find a solution using computer signage or speech via radio. I spent a day at the shopping center and saw for myself the nature of the problem. It had only one entrance. There was nothing a computer could tell drivers that they didn't already know. A second entrance on the opposite side of the shopping center was the obvious solution. But that's not what they wanted to hear.

Our unique ability to create high-quality computer-generated speech led to a contract from the FAA thru our air traffic control division in the same plant, just upstairs from the Speech Lab. Air traffic controllers routinely issue voice weather and warnings to pilots flying under Visual Flight Rules such as *"November three-seven-six. Traffic, 12 o'clock, 5 miles, southbound. Beacon reported altitude five thousand, six hundred."* Because the air traffic controller's computer knows all this information, all it needed was a voice to automate the warnings. We demonstrated this with live aircraft in Knoxville, TN in the mid seventies. In a John Henry style of man-versus-machine test, we found the computer's information to be slightly more accurate than the controllers' because the computer waited to calculate altitude until the last millisecond before speaking. For a fast-climbing military aircraft, this could be a difference of many hundreds of feet. But it sounded too good. It was an embarrassment to controllers because we used the voice of a professional radio announcer who had a voice much clearer than the average controller. Also, pilots were not aware that it was a computer so they tried to talk back to it. An FAA manager asked, "Can't you make it *sound* like a computer?"

A test of the Voice Response Unit (VRU) for automatic Minimum Safe Altitude Warnings (MSAW) was evaluated in Atlantic City, NJ. Preventing a plane from hitting the ground ("terrain avoidance" in FAA-speak) is serious stuff so we worked and reworked the messages until they met everyone's approval including a team of lawyers.

The MSAW VRU included a calibrated test message that we used for audio quality measurements in radio transmissions. It was activated by an unpublished keyboard entry from the controller's workstation. Somehow, one of the air traffic controllers triggered this test message while the system was online, and pilots on the eastern seaboard heard a 10-second excerpt of Nat King Cole singing *Straighten Up and Fly Right*. The FAA insisted we replace it with an audio tone.

About that time, Sperry developed a large new core memory for its Navy UYK-20 computers that it hoped to sell to the US Navy. The annual Navy computer show was coming up soon in Monterey, CA and they were looking for a way to demonstrate it there. Gene came up with the idea of storing a lot of speech and graphics in the big memory and having the computer put on its own show. It worked beautifully—graphics on a rugged shipboard workstation explaining how the memory worked with voice and musical accompaniment - all in the same room with dozens of Sperry's vendors and competitors.

But the crowded ballroom where the big computer show was being held was severely overloaded electrically. Every few minutes the power failed and the hotel staff ran around shouting to each other in Spanish until power was restored. Then the computer types hustled to re-boot their computers as quickly as possible. But the Sperry computer with its huge core memory recovered instantly and spoke its start-up message in a loud voice "The Sperry UNIVAC Expanded Memory System is Ready!" while the IBM guys in the next booth were still looking for their boot tapes. This happened many times and the Sperry marketing guys took advantage of it by turning up the volume knob whenever the lights went out.

The Knoxville and Monterey experiments made us famous around the corporation. It led to another contract, this time with Sperry's commercial division. They had a customer in Germany, Otto Versand, the largest catalog ordering company in Germany and the third largest in the world. Their offices were so busy that they had full-time forklift operators in Hamburg keeping the line printers filled with paper. Otto Versand was a huge customer of Sperry UNIVAC 1100 main frames built in Roseville, MN. But Otto Versand threatened to give their business to IBM unless Sperry came up with new technology to increase Otto's advantage in this highly competitive business.

I liked the German Sperry organization's corporate motto--*Wie lange als Computers gibt*—as long as there have been computers. It reflected the German respect for experience.

Someone in their organization read about the Knoxville test and had the idea to automate Otto's catalog ordering by phone. They came to us.

The catalog ordering business in Europe had big peaks and valleys throughout the day. It was not cost-effective to hire large numbers of telephone operators just for a couple of hours per day, so potential orders were being lost due to overload. A talking computer could take up the slack, it was reasoned.

Meanwhile, a Sperry plant in Winnipeg was looking for orders for its UYK-502 computer. It was a ruggedized, mil-spec computer for the Canadian Navy that was a low-cost version of the US Navy's standard minicomputer, the UYK-20. This Canadian computer had the same instruction repertoire as our U1616 Speech Lab computer which was a hardwired predecessor of the AN/UYK-20. It wasn't very fast, only 100K instructions per second. But it had a powerful input-output section—a perfect traffic cop for voice response data flow. A deal was struck that pleased everyone.

The only other catalog ordering voice-response computer in existence was a special purpose machine developed for Sears. But it was custom made for Sears and it couldn't be adapted to other applications so it was no serious competition for us. Nevertheless, it proved the concept of Touch-Tone input and computer speech output for automated catalog ordering.

We designed the VRU-100 Voice Response Unit to be a general-purpose programmable machine for a wide variety of applications. We couldn't call it a computer because that would violate some telephone industry standards so it was just a "unit." Fair enough because it had to be attached to a mainframe anyhow. Our old nemesis IBM, no doubt still fuming over the Monterey affair, quickly developed a voice response *unit* of its own but our VRU remained the first and only in Europe for a long time.

The voice part was easy, even in German. The hard part was the input. Counting clicks from rotary telephones was too unreliable. Touch-Tone telephones were just starting to be used in the US and were rare in Europe. The few European examples had their buttons arranged in a circle like a rotary dial telephone. They were designed only for dialing. After connection, the reversal of the TIP and RING line currents would disable the tone generators. In order to get them to generate tones after connection, it was necessary to install full-wave rectifiers in each telephone. In those days, the tone generators in Touch-Tone phones used two big copper coils, not an integrated circuit chip.

A better solution came from a then small company, Interstate Technology (IT), which made a small hand-held device that generated Touch Tone signals and connected to the mouthpiece with a rubber cup. We collaborated with them to make a version of their device for European telephones. It worked perfectly. They sold thousands of them. In the end, IT made more money from the project than we did.

We had some Otto Versand catalogs lying around the lab. When word got out that the bath section contained photos of nude women, which it did - visitors to the Speech Lab increased.

For the VRU's voice we used a well-known German TV news commentator, Werner Wegle. Otto's phone lines were swamped for a few days with people wanting to talk to Herr Wegle. Sometime later, I visited one of the twelve offices where our VRU-100s were installed. I was surprised to see all twelve of its line-activity lights glowing steadily (they flashed whenever a Touch-Tone character arrived). No one can push telephone buttons that fast. How could this be, I asked. I was told that Otto's operators had learned to take orders offline on memory dialers, call the VRU and press playback, spilling hundreds of characters in a stream at a rate of ten per second. Fortunately we had programmed the VRU for the worst-case data rate. It turned out to be the norm.

Years later I bumped into a Sperry Germany executive at a computer conference. I asked him how the 12 Otto Versand VRUs were working. They were so reliable, he replied, that his field engineers had no experience repairing them. Not surprising. They were made of our best milspec hardware with no moving parts.

Sperry kept Otto as a customer and Otto became the largest catalog company in the world in 1987. I like to think we were a part of that.

Marketing was always on our case to provide software to allow customers to digitize their own voice response vocabularies. We resisted this because we wanted to control the voice quality which we felt only we could do well because of our understanding of speech prosody from our speech recognition experience. To this day I have never heard concatenated computer speech as natural as our work.

It seems strange today, but convincing people that speech or any sound can be recorded as a string of binary numbers was tough. The math of Nyquist's Criteria and Shannon's Law was too much for non math types, so we made a recording of computerized speech in 12 languages including German, French, Urdu, Mandarin and even an excerpt from Homer's *Odyssey* spoken in ancient Greek plus Charlie Parker's 1942 bebop recording of *Yardbird Suite*. We workers in the Speech Lab heard that ancient Greek bit so often we had it memorized and used parts of it as a greeting.

The idea of voice mail was a logical extension of our voice response work. In fact, we had a crude form of it working in the early 70s. When Don Anderson or Mark Medress did remote demos of voice response on our Speech Lab computer, we could leave recorded messages for them that they could recall with a special Touch Tone code. The concept of voice mail and its potential usefulness for business had been around for a long time. A Bell Labs study had shown that 80% of all business telephone calls were a one-way transfer of information. But it wasn't until the early 80s that we had the hardware we needed—reliable disk drives, 8-bit microprocessors and the widespread availability of Touch Tone telephones. And so we were commissioned by the commercial division of Sperry to build a Voice Information Processing Station (VIPS) as a stand-alone component of SperryLink, a PC-like workstation for office applications to be built in Roseville Minnesota.

There were already two voice mail systems before us. VMX, a small company, developed an array of microprocessors that had no central control. It was complex and unversatile but VMX had a very broad patent. To our amazement, their patent even claimed the digital encoding of speech. A VMX employee told me later that they never sold any machines but made all of their profit from patent suits.

IBM's voice mail system was a business minicomputer adapted to this application. It had a non-real-time multitasking operating system. IBM wrote a single-channel voice mail program and then ran up 8 copies of it simultaneously. It was expensive and lacked important features. Worst of all, it had to be taken offline every day at midnight to defragment its disk drives.

Our design was much different than either the VMX array processor or IBM's time-shared architecture or maybe it was something in-between. It was an extension of our successful voice response system, still using our super-reliable Canadian Navy computer. It had no operating system, just a disk handler written by John Siebenand that we called the Siebsoft Operating System. But John's disk handler performed disk defragmentation as a background low priority task, so our system never needed to be taken offline. This proved to be especially important for international business where calls come in at all hours. That was the last straw for IBM. They couldn't compete with us so they went out of the voice mail business, leaving Sperry the only provider of a voice mail product for a while.

John and I developed a speech compression algorithm called Block Scaled Pulse Code Modulation that was a direct result of our speech recognition work. It effectively doubled the amount of speech we could store and simplified the Z80 signal processing. Speech rate, pause removal, telephone tone detection and volume could be controlled by manipulating blocks of speech data instead of individual samples. All signal processing was done with adds and subtracts. Time-consuming multiplies and divides were not needed.

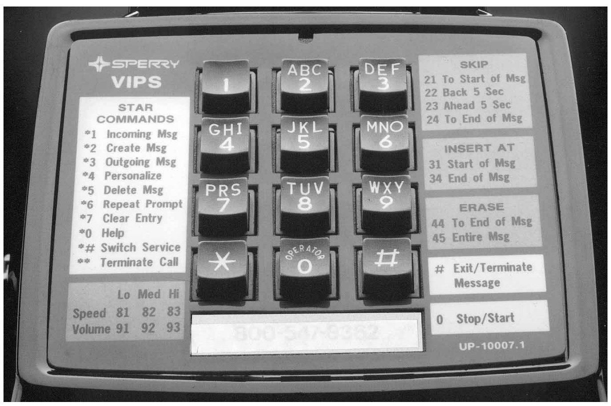
Ray Hedin designed an audio processing card for VIPS that had two Z80 8-bit microprocessors that Ray matched perfectly to our BSPCM coding scheme. One Z80 performed all of the telco interface and signal processing functions while the other Z80 interfaced to the computer's backplane. We used 10 of these cards in each machine to process up to 10 phone lines simultaneously. This gave our machine over ten times the processing ability of IBM's design with no operating system overhead. Wade Johnson wrote the application program for the CPU that was a command and control program unburdened by signal processing. This allowed enough excess processing to test every operation as it happened which was more typical of military computers rather than depending on off-line tests more characteristic of commercial computers. It tended to catch intermittent malfunctions and get them repaired before they affected system reliability.

I submitted a paper describing the audio processing card to the journal of the Institute of Electrical and Electronics Engineers without mentioning that machines were already in the field. But the reviewers rejected it, saying they didn't think the design would work.

Because no successful voice mail system preceded us, we didn't know what features were really needed or would be accepted by customers so we put in everything we could think of. Our design philosophy was to create a voice version of the then current word processing programs just starting to be in widespread use at the time. Wade included tools to edit voice messages such as deleting sentences or inserting speech into the middle of a message and forming distribution lists and forwards and replies much like a written office memo would be produced. Speech playback could be sped up or slowed down with a microcode version of an old analog technique used in tape recorders for the blind known as the Fairbanks Algorithm. It could play back speech at a blazing speech rate of up to three times the original input. The 3x rate was hard to listen to. Nobody used it except other speech scientists and the blind, who were used to listening to talking book tape players. A pause-removing scheme adapted from our speech recognition work sped up playback and saved storage. It allowed speakers to hesitate as much as they wanted while dictating, knowing the pause would be removed. Many of these features were rarely used. Others became commonplace in other voice mail systems and email.

The SperryLink CRT monitor listed all the messages a user had in his mailbox including the names of the senders and recipients. These could be manipulated from the keyboard without listening to them, a feature still not found in most modern voice mail systems. We wanted to add gisting in order to display the subject of the message but of course we couldn't.

User's telephone numbers were known to VIPS, so incoming messages could be forwarded without the caller knowing the phone number of the recipient. A traveler could inform VIPS of a temporary location, for example, and receive messages anywhere in the world. VIPS were linked together and informed each other via telephone line Touch Tone signals of the locations of users, a simple predecessor of Internet message routing. Voice messages were transferred from one VIPS to another via analog signals. This feature was used routinely by Speech Lab personnel as VIPS sales boomed internationally. We called it voice store-and-forward, borrowing an ARPANET term.



As sales were made around the world, we posted a map on the wall of the Speech Lab with pushpins indicating each VIPS location. When almost all time zones were filled, we proudly proclaimed that the sun never sets on Sperry speech products.

One of our crackerjack salesmen was a South African of German Jewish descent operating out of London named Stan Friedlein. His idea of being in the neighborhood was to be in the same continent. His baggage was a plastic shopping bag with a Camels ad in Afrikaans. He could be anywhere, "out of the office" as he would say—Singapore one day, Port Said the next, or Russia, or France. We never knew where. So we used VIPS to communicate with him and he with us. The VIPS network found him every time and faithfully delivered our messages and returned his replies. Even I was amazed.

In order for VIPS to place telephone calls, it had to recognize local telephone signaling tones—busy, ring, etc. Each country has its own set of tones: Britain, for example, had "tinkle tones" every few minutes that reminded callers that they were using a valuable national resource so get on with it. VIPS was programmed to recognize them. Again, our clandestine word-spotting experience with speech in the presence of non-speech was pressed into use, microcoding the detection of telephone signals into the Z80s of each audio channel. With Stan's international skill with government telephone bureaucracies, getting us specs and recordings, we could program the Z80 microcode for a new country and pass the country's signaling certification test in a couple of days at a time when telephone circuits typically accomplished this with custom analog circuits. This gave us a huge competitive advantage in Europe.

We were surprised by some of the applications that developed for VIPS. A school district in Phoenix used one for calling parents when their kid was AWOL. The Union Pacific and later some airlines used it for crew scheduling--it allowed crew members to negotiate times and locations. A gay group used one for messaging—a predecessor of a web chat room. Three Mile Island bought several for emergency evacuation notices. One VIPS served as a help line for a homeless shelter.

One day, the PR department received a letter asking what our company was doing for the handicapped community. It was forwarded to me, **thanks a lot!** I explained that we were a defense contractor so we worked only at the command of government agencies and sometimes for our commercial division. I explained voice response and voice mail and pointed out that blind people can use our voice information products as easily, perhaps even easier, than the sighted.

We saw the Internet coming and we warned our managers that it would eventually replace voice mail. We were wrong. But the designers of SperryLink were even more wrong. They bet that the Internet would not develop into anything useful, so they linked SperryLink stations (*stations* would later be known as *servers*) by private telephone lines without the ability to adapt to the Internet--a big mistake.

We also foresaw telephone companies incorporating voice mail into their networks on a large scale. This seemed like the ultimate and best place for voice mail—imbedding into the telco would simplify integration and increase economy of scale. And it seemed like a logical extension of basic telephone service. We told our managers that to continue in the voice mail business we would have to build much bigger systems and target the telephone companies as customers. They chose not to. That was probably a good decision because it would have taken us too far outside the charter of a defense contractor. We had gone as far as we could go and it was a great ride.

This article provided to the VIP Club and Lockheed Martin Corporation Legacy Project
by David P. Andersen on July 16, 2007.