Times Have Changed

Robert L. Patrick

Editor: David Walden

The following little stories from the past may seem familiar to some readers and less familiar to others. Many things have changed, and others not so much.

Today computer input typically comes in over the network, from DVDs or thumb drives, or off hard drives built into the computer.

1952. My first computer experience involved a computer that read each instruction and all its data from manually punched cards. It was rated at 100 cards per minute, but it slowed down during heavy calculations. The hopper held 500 cards, and the computer expected an operator to be in attendance at all times. However, when we worked late to run priority work, we got hungry. So we devised an extension for the hopper out of cardboard and duct tape so it could hold 2,000 cards. This gave us time for a quick trip to a nearby hamburger joint for takeout.

Today computers are either small enough to use in an office setting or large systems in rooms with all sorts of built-in safety measures.

1954. The computer room at General Motors Research was designed to be part of a foundry. So when the IBM 701 computer was installed, we got a giant CO2 fire bottle to protect it. The bottle was 5 feet tall and 1.5 feet in diameter and was mounted on a cart. The wheels of the cart were about 3 feet in diameter. I was chosen to demonstrate its use for fire protection. The whole staff gathered for the demo, about 25 in all. After a short talk, I grabbed the nozzle and pulled the trigger. I hadn't done my homework, and instead of the spring-loaded release trigger common to smaller units, this fire bottle had a latching trigger. I got it started okay, but it wouldn't shut off. I'd built a large cone-shaped pile of dry ice before I figured out how to release the trigger. All my colleagues were overwhelmed with hysterical laughter.

Today, with wireless connections, most people don't get a chance to deal with any real wiring any more.

1954. At GM Research the computer offices we inherited from the foundry still had a klaxon that blared for hourly union workers. When programming, noise tends to break your focus and concentration. My immediate boss, two of my peers, and I decided to disconnect the wires from the offensive device. We weren't allowed to move furniture or use ladders, as the union would be offended, so I clambered up on a wall partition with a pair of side cutters to clip a wire.

At the precise time my colleagues were holding my feet to keep me from falling, T.C. Van der Grift, the big boss (and an old associate of Charles Kettering) came in unannounced and said, "What cha doin' boys?" From 8 feet above, I stammered that we were getting rid of an offensive noise that interrupted our work. He said, "Carry on," and left.

I bet many demos still have a "spontaneous" component today. Maybe some things haven't changed so much.

1955. During an open house at the GM Tech Center where I worked, we had to prepare a demonstration of our giant new (unreliable) IBM 701 computer. We programmed it using its simplest instructions because these were the least likely to fail during the open house. The computer had two console lights that could be programmed to flash and keep the operator informed of what part of the program was being executed. We made a clown's face out of poster board that, when lowered by a string, covered the console and the computer blinked its eyes at the audience. The little kids loved it. We used $1 million of electronics to blink a clown's eyes.

Today debugging can often be done across the network with the diagnostician never leaving his or her home base.

1955. When I was at GM, we had the only big computer in the corporation. I was working on a hot job for a division in Milwaukee. Once Milwaukee very badly needed me and a case of 10,000 punched cards that I had produced. The corporate motor pool sent a limo to my home, and my wife gave them the overnight case she had packed. Then the limo picked up me and my case of cards at the Tech Center and took me to Detroit airport. There I boarded a corporate DC-3 (sole occupant) that flew "little me" to Milwaukee.

Until recently, big users produced their own applications programs. In the early days, they also built their own control software or modified the manufacturer's product.

1962. We had put computer usage accounting in the original GM I/O system when we developed it at GM for the 704 back in 1956. The system was efficient but did not handle priority work well. When we designed the Direct Couple System (an extension to basic 7090 IBSYS from which sprang HASP, ASP, and JES3) at the Aerospace Corporation in 1962, we needed additional detail about the quantities and overlap between I/O transactions and computing activities (I/O balance) to configure the system properly.

Dick Van Vranken, the chief system programmer at Aerospace, put patches in the IBSYS operating system to record (and time stamp) all input and output transactions and how they overlapped with segments of heavy

1058-6180/14/$31.00 © 2014 IEEE

As a personal computer user, I wonder about the regression testing that is done these days by computer software vendors.

computing on the IBM 7090 we were using. The project goals were throughput and the ability to respond to priorities without degrading other work.

Our design goal moved all scheduling and I/O services to an IBM 7040 acting as a frontend processor and let the 7090 compute without ever waiting for I/O service. As a result of a fluke in IBM pricing, we also saved hardware rental dollars. Furthermore, it was to be IBSYS compatible so the customers would never know.

About this time, IBM brought us a footlocker of electronics complete with customer engineering instructions telling how to connect it up. This box had been traveling through their customer set measuring I/O and overlap. (We didn't know it at the time, but they were designing the System/360 hardware and needed detail customer usage profiles.)

When we compared the measurements from their footlocker of electronics with our software patches, they didn't match. Upon investigation, they found a wiring error in the footlocker. Thus, we made a contribution to the design of the System/360.

1963. After the Direct Couple (control) software was running, we physically reconfigured the machine room. For security and visitor segregation, we provided view windows in a hall. The tape vault fed two nearby rows of tapes. A remote console printer at the vault told what tape to mount and where. Because three 1,000 line per minute printers used a lot of paper when running full out, we made a clear path for dollies transporting boxes of paper going to the printers. An analysis of printer output delays found that operator fatigue caused queues of completed work to build up behind the printers. We manufactured slides for printouts behind each printer and these slides fed completed print jobs to a conveyor belt that delivered them to a security station. The unclassified printouts went to open cubby holes for pickup, and the classified work was held for formal sign out by the submitter. We also rotated the machine room staff through their assignments hourly because some operational jobs were more fatiguing than others.

Rockwell was one example of a customer leading a field: they built the first database management system of which I know.

1968. Pete Nordyke and Uri Berman had built the first database software. They had an online parts and configuration management application running on the IBM 7010. In 1968, we generalized and remanufactured the code for the System/360. IBM participated and sold it as a product. It was called IMS/360. At Rockwell, we needed to give the best service possible because they were in the middle of developing the Apollo space capsule and badly needed computing capacity. We had two IBM 360-65s installed in a large complex about the size of a basketball court. There were several communications control units, banks of disk drives, rows of tape drives flanking a tape vault, and two CPUs.

In a test with IMS running on one machine, the other Mod 65 fed it data transactions in a realistic manner. Then, the number of transactions was increased until saturation was reached. The tuned system could handle 4,000 transactions per hour at the time of initial release.

As a personal computer user, I wonder about the regression testing that is done these days by computer software vendors.

1969. If you are running a big system with frequent changes, you could collect troublesome jobs to use for release verification. Jack Van Kinsbergen, then at Hughes, had a tape of “bad” jobs that he ran to verify that each new release of OS/360 was compatible with the previous one. He had captured a bunch of troublesome jobs, their input, and a tape copy of their output he knew to be correct. After each new OS release was ready to install, he ran this batch and compared output tapes.

There is constant talk about computer security these days, and highly structured, if not necessarily completely effective, methods are used for protection. For instance, it is said passwords should be greater than eight characters and contain upper and lower case letters, at least one number, and at least one nonalphanumeric character.

1967. I worked for a government agency on some highly classified work. They used random code words for naming projects. Fred Ruffing was chosen to build and maintain the master list of new code words because he knew more dirty words than anyone else.

After 32 years as a freelance consultant, I retired. It took a couple of years to decompress.

My career ladder was coder, programmer, project leader, technical manager, company founder, systems analyst, and consultant. I retired in good health and needed a project, so I focused my systems analysis skills on earthquake preparations.

The US Geological Survey (USGS) has long forecasted a big quake for California. They can't say exactly where or when the quake will hit, but statistically it is due. It will probably hit Southern California the hardest (http://pubs.usgs.gov/circ/1324/c1324.pdf), and the last comparable quake opened up a rent in the earth 220 miles long.

Robert L. Patrick *was an independent consultant for more than 30 years. Most of this time he was involved with either applications development or operations management. Contact him at bobpatrick@mac.com.*

cn *Selected CS articles and columns are also available for free at http://ComputingNow.computer.org.*

ADVERTISER INFORMATION • JANUARY-MARCH 2014

Advertising Personnel

Marian Anderson: Sr. Advertising Coordinator
Email: manderson@computer.org
Phone: +1 714 816 2139 | Fax: +1 714 821 4010

Sandy Brown: Sr. Business Development Mgr.
Email sbrown@computer.org
Phone: +1 714 816 2144 | Fax: +1 714 821 4010

Advertising Sales Representatives (display)

Central, Northwest, Far East:
Eric Kincaid
Email: e.kincaid@computer.org
Phone: +1 214 673 3742
Fax: +1 888 886 8599

Northeast, Midwest, Europe, Middle East:
Ann & David Schissler
Email: a.schissler@computer.org, d.schissler@computer.org
Phone: +1 508 394 4026
Fax: +1 508 394 1707

Southwest, California:
Mike Hughes
Email: mikehughes@computer.org
Phone: +1 805 529 6790

Southeast:
Heather Buonadies
Email: h.buonadies@computer.org
Phone: +1 973 304 4123
Fax: +1 973 585 7071

Advertising Sales Representatives (Classified Line)

Heather Buonadies
Email: h.buonadies@computer.org
Phone: +1 973 304 4123
Fax: +1 973 585 7071

Advertising Sales Representatives (Jobs Board)

Heather Buonadies
Email: h.buonadies@computer.org
Phone: +1 973 304 4123
Fax: +1 973 585 7071