So You Wanna Be a Cowboy

Tex Curtis

"Mommas don’t let your babies grow up to be cowboys,
Don’t let ‘em pick keyboards and drive an old VAX,
Make ‘em be doctors and lawyers, not hacks."
— with apologies to Willie Nelson

Every morning during my youth a banner atop the Fort Worth Star-Telegram would remind me that I was growing up “Where the West Begins.” A strong wind blowing through north Fort Worth would remind the rest of Cowtown that the old Chisholm Trail now ended in the cattle yards just across the Trinity River from downtown. Every year during the Southwestern Exposition and Fat Stock Show the public schools would host a Ranch Day, and all the kids could come to school in their cowboy or cowgirl getup. Many of the kids had ridden horses since they were toddlers, and some harbored thoughts of owning a ranch and being a cowboy when they grew up.

Few, probably none, became real cowboys. Even so, those that became programmers found an outlet for their childhood fantasies. They traded a million acres for a million pixels and rued the cold damp. Yet childhood dreams die hard, and kids who rarely roughed it in the backyard still had an image of themselves to fulfill. It was an image born from watching John Wayne (“the Duke”) and interminable TV Westerns. It was an image of the lone hero struggling through the night against impossible obstacles to save the (pick one(town, herd, girl)). The first time you delivered on a project from hell, you knew you had filled the boots in your mind. The Duke would be proud.

Few images seem quite so ingrained in the American psyche as that of the heroic cowboy. Hollywood raised the middle generations of the 20th century on images of a life that faded with the 19th. Madison Avenue has used nostalgia for cowboys to sell every manner of product. And too many software pundits, having become enamored with the rare wizard hacker, have venerated the “cowboy programmer.”

Even urban cowboys feel that this metaphor degrades their professionalism. So how did the pundits concoct this similitude? Should they be gored by their own oxymoron?

The image of the wild, unruly cowboy evolved from occasional barroom brawls started by trail-riders in towns along the drive. A minority of cowboys engaged in these outbursts, and it was a miniscule part of the lives of those who did. Although most cowboys owned a revolver, few had the marksmanship portrayed in TV Westerns. Nevertheless, this image became popular with people, especially easterners, who had never seen a ranch or been on a trail drive. However, as Theodore Roosevelt observed in Ranch Life and the Hunting Trail, “Few of the outrages quoted in east-

Continued on p.110
Rollercoasters and the possible death Marches.

A programmer is no different from a cowboy. They both suffer from compiler errors, never admit uncertainty about the success of their code, and do not have much time for a family. Yet when evaluated on the trait of machismo, the programmer clearly emerges as the programmer ever left a complaint unvoiced. Yet when they sign on for the typical death march project, programmers acquiesce to inhuman hours with a stoicism that can only emerge from the brand of professional pride that made cowboys react to heat, dust, loneliness, mosquitoes, weariness, thirst, and injuries with silence. Programmers exude a new, more complex stoicism—one with an attitude.

Programmers suffer the same social stigma that plagued cowboys. Although cowpuncher and nerd seem antipodes, they share a common perception by “more refined” folk of being introverted and just a bit odd. In appearance and manner, neither the programmer nor the cowboy fits into most crowds. Both are in some way weird, and the rush becomes resentment. A new, more complex stoicism—one with an attitude.

Programmers suffer the same social stigma that plagued cowboys. Although cowpuncher and nerd seem antipodes, they share a common perception by “more refined” folk of being introverted and just a bit odd. In appearance and manner, neither the programmer nor the cowboy fits into most crowds. Both are in some way weird, and the rush becomes resentment. A new, more complex stoicism—one with an attitude.

Programmers suffer the same social stigma that plagued cowboys. Although cowpuncher and nerd seem antipodes, they share a common perception by “more refined” folk of being introverted and just a bit odd. In appearance and manner, neither the programmer nor the cowboy fits into most crowds. Both are in some way weird, and the rush becomes resentment. A new, more complex stoicism—one with an attitude.

Whether cutting cattle or code, weakness in the profession’s core competence cannot be substituted by capability in any other.
competence cannot be substituted by capability in any other.

Roping dominated when cowboys had to separate specific cows from the herd (cutting) and capture them with a lariat for branding (a cowboy version of copyrighting one’s property). However, the cowboy could never accomplish these tasks without the close cooperation of an exquisitely skilled cutting horse. In fact, the collaboration achieved between the cowboy and his cutting horse was as tightly attuned as that achieved by even the most aligned practitioners of pairs programming. The pinnacle of this alliance relied on little more than pressure from the cowboy’s knees to guide the horse’s reactions. When their relationship achieved this maturity, the cowboy could dispense with the reins, leaving both hands free for roping.

Much as for programmers, a cowboy’s professional life was organized into projects—the annual cattle drives. The stages of these projects included the roundup, cutting and branding, and the trail drive. Long before fences enclosed the West, cattle grazed on open ranges. As with any open system, cattle from many ranches mingled freely. Annually they had to be rounded up and calves separated for branding. The brands were cryptic—much like programming notation—and perhaps provided an early inspiration for APL. Some calves became separated from their mothers and survived on handfed food, medicine, and parental advice from the chuck wagon. Cow-boys that were unreliable or caused problems were usually dismissed at towns along the trail.

Certain occasions demanded disciplined coordination under great stress—stampedes. Cattle could be spurred by anything from lightening to stray roosters. During stampedes, cowboys had to spring into action already knowing their responsibilities. The best riders would gallop to the front of the herd and rein back in an attempt to slow it. Other cow-boys would ride to the point and try to turn the herd, thus changing its direction and forcing it to slow. Stampedes were not stopped by heroes; they were stopped when all cowboys maintained their position and worked together to slow, turn, and control the herd. When cowboys broke discipline, someone died.

The fencing of the range and the coming of the railroad were as threatening to many cowboys as the slow decline of technologies, such as Cobol, is to programmers who are “fit for little but [their] own special work.” Cowboys, unfortunately, got no reprieve with the year 1900. Yet some adapted. Modern cowboys use helicopters and ranch management software, while dreaming of laser-guided lassos and Stetson-mounted displays. Yet, for cutting in tight corridors, there is still no substitute for skill with a lariat and a quarter horse. Similarly for cutting tight code, fourth-generation languages and automatic code generators have not replaced talented programmers. Cowboys and programmers are both aware that they are critical for driving valuable stock.

Maybe true cowboy programmers would be good after all. Cow-boys and programmers both tend to be introverts who fiercely guard their independence. Yet others must be able to work with them, especially under stress. Both have uncommon skills. Yet if they do not accept the disciplined procedures of their profession, the stock will suffer. So let’s call disorderly programmers what they are—not cowboys, but mavericks.

Mavericks may have a role in research labs where they can roam about unfettered, but they do not belong on project teams whose members must depend on each other to perform reliable work on competitive schedules. Maybe mommas should encourage their budding prodigies to be more like cowboys. The Duke would be proud.

Bill Curtis is one of the head honchos at TeraQuest, an outfit that works with software organizations to corral their maverick projects. He began riding horses at two and has chased mavericks on his best friend’s ranch. Nevertheless, his ineptness with a lariat prioritized math over mustangs. Before signing on with TeraQuest, he rode for GE, ITT, MCI, and SCI, where he helped develop the CMM brand. He rides out of Austin, a town now more dominated by dudes than longhorns. He bunks at curtis@acm.org.