How to shoot yourself in the foot

Using various systems

(found on the net)

May 19, 2010

Contents

0.1 370 JCL

- You send your foot down to MIS and include a 300-page document explaining exactly how you want it to be shot. Two years later, your foot comes back deep-fried.
- You shoot yourself in the head just thinking about it.
- You find the first building you're in in the phone book, then find your office number in the corporate phone book. Then you have to write this down and describe, in cubits, your exact location in relation to the door (the right side thereof). Then you need to write down the location of the gun (loading it is a proprietary utility), then you load it, and the COBOL program, and run them, and with luck, it may be run tonight.

0.2 Ada

- If you are dumb enough to actually use this language, the United States Department of Defense will kidnap you, stand you up in front of a firing squad, and tell the soldiers, "Shoot at his feet."
- After correctly packaging your foot, you attempt to concurrently load the gun, pull the trigger, scream, and shoot yourself in the foot. When you try, however, you discover that your foot is of the wrong type.
- You scour all 156e54 pages of the manuals, looking for references to foot, leg, or toe; then you get hopelessly confused and give up. You sneak in when the boss isn't around and finally write the damn thing in C. You turn in 7,689 pages of source code to the review committee, knowing they'll never look at it, and when the program needs maintenance, you quit.

0.3 Algol

- You shoot yourself in the foot with a Civil War-era musket. The musket is aesthetically fascinating, and the wound baffles the adolescent medic in the emergency room.

0.4 Algol 60

- You spend hours trying to figure out how to fire the gun because it has no provisions for input or output.

0.5 Algol 68

- You mildly deprocedure the gun, the bullet gets firmly dereferenced, and your foot is strongly coerced to void.

0.6 APL

- You shoot yourself in the foot and then spend all day figuring out how to do it in fewer characters.

- You hear a gunshot and there's a hole in your foot, but you don't remember enough linear algebra to understand what happened.

0.7 Apple System 7

- Double click the gun icon and a window giving a selection for guns, target areas, plus ballon help with medical remedies, and assorted sound effects. Click shoot button and small bomb appears with note "Error of type 1 has occurred."

0.8 APT

- You cut a perfect bullethole in your foot and shoot through it.

0.9 ASP

- You try to shoot yourself in the foot, but the most advanced thing you can manage is to cut your wrist.

- Find a gun, it falls apart. Put it back together, it falls apart again. You try using the .GUN Framework, it falls apart. You stab yourself in the foot instead.

0.10 Assembly

- You try to shoot yourself in the foot only to discover you must first reinvent the gun, the bullet, and your foot. After that's done, you pull the trigger, the gun beeps several times, then crashes.

- You crash the OS and overwrite the root disk. The system administrator arrives and shoots you in the foot. After a moment of contemplation, the system administrator shoots himself in the foot and then hops around the room rapidly shooting at everyone in sight.

- By the time you've written the gun, you are dead, and don't have to worry about shooting your feet. Alternatively, you shoot and miss, but don't notice.

- Using only 7 bytes of code, you blow off your entire leg in only 2 CPU clock ticks.

0.11 BASIC

- Shoot self in foot with water pistol. On big systems, continue until entire lower body is waterlogged.

0.12 BCPL

- You shoot yourself somewhere in the leg; you can't get any finer resolution than that.

0.13 C

- You shoot yourself in the foot.
- You shoot yourself in the foot and then nobody else can figure out what you did.

0.14 C++

- You accidentally create a dozen instances of yourself and shoot them all in the foot. Providing emergency medical assistance is impossible since you can't tell which are bitwise copies and which are just pointing at others and saying, "That's me, over there."

0.15 C#

- You shoot yourself in the foot, but first have to switch to unsafe mode.
- You forget precisely how to use the .NET interface and shoot yourself in the foot. You sue Microsoft for damages.

0.16 Clipper

- You grab a bullet, get ready to insert it in the gun so that you can shoot yourself in the foot, and discover that the gun that the bullet fits has not yet been built, but should be arriving in the mail real soon now.

0.17 COBOL

- USEing a COLT.45 HANDGUN, AIM GUN at LEG.FOOT, THEN PLACE ARM.HAND.FINGER on HANDGUN.TRIGGER and SQUEEZE. THEN RETURN HANDGUN to HOLSTER. CHECK whether SHOELACE needs to be retied.
- Allocate $500,000 for the project. Define foot, bullet, gun. Run press_trigger. Go for coffee break. Return in time to put foot under bullet.

- You try to shoot yourself in the foot, but the gun won't fire unless it's aligned in column 8.

0.18 Concurrent Euclid

- You shoot yourself in somebody else's foot.

0.19 CP/M

- You remember when shooting yourself in the foot with a BB gun was a big deal.

0.20 CSS

- Everyone can now shoot themselves in the foot, but all their feet come out looking identical and attached to their ears.
- You shoot your right foot with one hand, then switch hands to shoot your left foot but you realize that the gun has turned into a banana.

0.21 dBase II

- You buy a gun. Bullets are only available from another company and are promised to work so you buy them. Then you find out that the next version of the gun is the one that is scheduled to actually shoot bullets.
- You squeeze the trigger, but someone corrupted the index and the bullet shoots you in the eye.
- You squeeze the trigger, but the bullet moves so slowly that by the time your foot feels the pain you've forgotten why you shot yourself anyway.

0.22 dBase IV

- You pull the trigger, but it turns out that the gun was a poorly-designed grenade and the whole building blows up.

0.23 DCL

- ```
  $ MOUNT/DENSITY=.45/LABEL=BULLET/MESSAGE="BYE" BULLET::BULLET$GUN
    SYS$BULLET
  $ SET GUN/LOAD/SAFETY=OFF/SIGHT=NONE/HAND=LEFT/CHAMBER=1
    /ACTION=AUTOMATIC/LOG/ALL/GULL
    SYS$GUN_3$DUA3:[000000] GUN.GUN
  $ SHOOT/LOG/AUTO SYS$GUN SYS$SYSTEM:[FOOT] FOOT.FOOT
  %DCL-W-ACTIMAGE, error activating image GUN
  -CLI-E-IMGNAME image file $3$DUA240:[GUN] GUN.EXE;1
  -IMGACT-F-NOTNATIVE, image is not an OpenVMS Alpha AXP image
  ```
  ```

0.24 Delphi

- You try to shoot yourself in the foot but discover that the bullets you already had are not compatible with the new gun version, but Borland promises a fix real soon now.
- When you press the button "shoot foot", the error message IDAPI ERROR cannot find shoot_foot database appears.

0.25 Eiffel

- You create a GUN object, two FOOT objects, and a BULLET object. The GUN passes both the FOOT objects as a reference to the BULLET. The FOOT objects increment their hole counts and forget about the BULLET. A little demon then drives a garbage truck over your feet and grabs the bullet (both of it) on the way.
- You take out a contract on your foot. The precondition is that there's a bullet in the gun; the postcondition is that there's a hole in your foot.

0.26 English

- You put your foot in your mouth, then bite it off.

0.27 Forth

- Foot in yourself shoot.
- Foot yourself in the shoot.
- First you decide to leave the number of toes lost on the stack and then implement the word foot-toes@ which takes three numbers from the stack: foot number, range, and projectile mass (in slugs) and changes the current vocabulary to blue. While testing this word you are arrested by the police for mooning (remember, this is a bottom-up language) who demonstrate the far better top-down approach to damaging yourself.
- BULLET DUP3 * GUN LOAD FOOT AIM TRIGGER PULL BANG EMIT DEAD IF DROP ROT THEN. This takes about five bytes of memory, executes in two to ten clock cycles on any processor, and can be used to replace any existing function of the language as well as in any future words. Welcome to bottom-up programming, where you too can perform compiler pre-processing instead of actually writing code.

0.28 FORTRAN

- You shoot yourself in each toe, iteratively, until you run out of toes, then you read in the next foot and repeat. If you run out of bullets or toes, you continue anyway because you have no exception-processing ability.

0.29 Genetic Algorithms

- You create 10,000 strings describing the best way to shoot yourself in the foot. By the time the program produces the optimal solution, humans have evolved wings and the problem is moot.

0.30 Haskell

- On a warm Saturday afternoon, sitting by the pool with a margarita, you casually sit up from your chaise lounge chair, reach over and pick up a gun, aim at your foot, and lazily pull the trigger.

- You shoot yourself in the foot very elegantly, and wonder why the whole world isn't shooting itself this way.

- You spend several hours creating a new copy of the Universe which is identical to the existing one except your foot has a hole in it. You then hear that it can be done more elegantly with Dyadic Functile Hyper-Arrows, but the very act of reading some of the included sample code causes one of your metatarsals to explode.

0.31 HTML

- You cut a bullethole in your foot with nothing more than a small penknife, but you realize that to make it look convincing, you need to be using Dreamweaver.

0.32 HyperTalk

- Put the first bullet of the gun into foot left of leg of you. Answer the result.

- You describe how to shoot yourself in the foot, which not only happens, but you also get cool visual effects.

- As of HyperTalk 2.2, you cannot shoot yourself in the foot from within the stack; you must write this functionality into an XCMD or XFCN. However, we anticipate this functionality to be incorporated into the next major release.

0.33 IDL

- You easily shoot yourself in the foot, complete with neat little graphs showing the trajectory of the bullet and the result of the impact. After twenty hours and ten thousand lines of code, your friend proudly announces that he has accomplished the same thing in an Excel spreadsheet.

0.34 INFORMIX

- The first gun doesn't work. Three months later INFORMIX's support desk send another gun which doesn't match the version number of the bullets. INFORMIX suggest you upgrade to INFORMIX-ONLINE. You pull the trigger and you shoe gets wet.

0.35 INGRES

- You pull the trigger, and your identical twin in San Franciso gets shot. You then turn off distributed query optimisation.

0.36 Java

- You write a program to shoot yourself in the foot and put it on the Internet. People all over the world shoot themselves in the foot, and everyone leaves your website hobbling and cursing.
- You amputate your foot at the ankle with a fourteen-pound hacksaw, but you can do it on any platform.
- After importing java.awt.right.foot.* and java.awt.gun.right.hand.*, and writing the classes and methods of those classes needed, you've forgotten what the hell you're doing.

0.37 JavaScript

- You find that Microsoft and Sun have released incompatible class libraries both implementing Gun objects. You then find that although there are plenty of Foot objects implemented in the past in many other languages, you cannot get access to one. But, seeing as JavaScript is so cool, you don't care and go around shooting anything else you can find.
- You've perfected a robust, rich user experience for shooting yourself in the foot. You then find that bullets are disabled on your gun.

0.38 LaTeX

- ```
  compy$ more foot_shooting.tex

  \documentclass[12pt}{article}
  \usepackage{latexgun,latexshoot}
  \begin{document}
  See how easy it is to shoot yourself in the foot? \\
  \gun[leftfoot}{shoot} \\
  \pain
  \end{document}

  compy$ latex foot_shooting
  ```
  ```

```
...
line 6: undefined control sequence \pain
```

0.39 Lisp

- You shoot yourself in the appendage which holds the gun with which you shoot yourself in the appendage which holds the gun with which you shoot yourself in the appendage which holds the gun with which you shoot...

- You attempt to shoot yourself in the foot, but the gun jams on a stray parenthesis.

0.40 Linux

- You shoot yourself in the foot with a Gnu.

0.41 Logo

- You can easily shoot the gun, but you have to work out the geometry to make sure the bullet goes into your foot.

0.42 Mac OS (System 7)

- Double-click the gun icon and a window appears, giving a selection for guns, target areas, and balloon help with medical remedies. Click the "shoot" button and a small bomb appears with a note "Bad F-line instruction."

0.43 Mac OS (System 7.1)

- Double-click the gun icon and a window appears, giving a selection for guns, target areas, and balloon help with medical remedies. Click the "shoot" button and a small bomb appears with a note "Error of type 1 has occurred."

0.44 Mac OS 9

- Double-click the gun icon and a window appears, giving a selection for guns, target areas, and balloon help with medical remedies. Click the "shoot" button and a window appears with the message "You need to install the latest version of CarbonLib. Should I get it for you?" You click "Yes" and your computer hangs.

0.45 Mac OS X

- You try to shoot yourself in the foot from the GUI but the gun has inexplicably turned into a bag of Skittles.
- You open up the Terminal, type sudo shoot -p /Library/BodyParts/Preferences/foot.plist, and your kernel panics.

0.46 Matlab

- You shoot yourself in the foot five times from the command prompt before you can put your foot in an m file.
- Once your foot is in an m file you shoot it fifty more ways effortlessly and then plot the results.
- Eventually you can't afford to continue shooting yourself in the foot this way, so you graduate to less elegant ways of shooting yourself in the foot with Excel.

0.47 Modula-2

- After realizing that you can't actually accomplish anything in the language, you shoot yourself in the head.

0.48 MOO

- You ask a wizard for a pair of hands. After lovingly hand-crafting the generic gun and generic bullet, you flag the objects as fertile and then tell everyone they can now shoot themselves in the foot.

0.49 Motif

- You spend days writing a UIL description of your foot, the trajectory, the bullet, and the intricate scrollwork on the ivory handles of the gun. When you finally get around to pulling the trigger, the gun jams.

0.50 MS-DOS

- You finally find the gun, but you can't find the file with the bullets for the life of you.
- You shoot yourself in the foot, but you can unshoot yourself with add-on software.

0.51 MPW

- Because you don't actually have a gun, you write an imitation UNIX shell and shoot yourself in the foot using Pascal.

0.52 .NET

- You can now shoot yourself in the foot with any of fourteen weapons, ranging from an antique medieval crossbow to a laser-guided Destructo-Beam. However, all these weapons must be manufactured by Microsoft and you must pay Microsoft royalties every time you shoot yourself in the foot.

0.53 Neural Networks

- You train the network in how to shoot your foot, after which it generalizes and keeps trying to locate some guy named Connor on the net.

0.54 Objective-C

- You write a protocol for shooting yourself in the foot so that all people can get shot in their feet.

0.55 Occam

- You shoot both your feet with several guns at once.

0.56 Oracle

- You decide to shoot yourself in the foot, so you go out and buy a gun, but the gun won't work without "deploying" a shoulder holster solution, relational titanium-alloy bullets, body armor infrastructure, a laser sight assistant, a retractable arm stock application, and an enterprise team of ballistic experts and a chiropodist.

- The menus for coding foot_shooting have not been implemented yet, and you can't do foot_shooting in SQL.

- ORACLE sell you a gun, a box of bullets, a holster, a cardboard mock-up of a wild-west town and a stetson. You find the trigger takes twenty seven people to pull it. ORACLE provide 26 consultants all with holsters, cardboard mock-ups and stetsons. The bullet doesn't leave the gun-barrel and you hire four more ORACLE consultants to optimise. The bullet bounces off your sandals.

0.57 Paradox

- Not only can you shoot yourself in the foot, your users can too.

0.58 Pascal

- The compiler won't let you shoot yourself in the foot.
- The gun is mounted such that it cannot point towards your feet, but you can swivel it round and shoot yourself in the head instead.

0.59 Perl

- You shoot yourself in the foot, but nobody can understand how you did it. Six months later, neither can you.
- You separate the bullet from the gun with a hyperoptimized regexp, and then you transport it to your foot using several typeglobs. However, the program fails to run and you can't correct it since you don't understand what the hell it is you've written.
- You stab yourself in the foot repeatedly with an incredibly large and very heavy Swiss Army knife.
- You shoot yourself in the foot and then decide it was so much fun that you invent another six completely different ways to do it.
- There are so many ways to shoot yourself in the foot that you post a query to comp.lang.perl.misc to determine the optimal approach. After sifting through 500 replies (which you accomplish with a short Perl script), not to mention the cross-posts to the perl5-porters mailing list for which you upgraded your first sifter into a package, which of course you uploaded to CPAN for others who might have a similar problem (which, of course, is the problem of sorting out e-mail and news, not the problem of shooting yourself in the foot), you set to the task of simply and elegantly shooting yourself in the foot, until you discover that, while it works fine in most cases, NT, VMS, and various flavors of Linux, AIX, and Irix all let you shoot you in the foot sooner than your Perl script could.

0.60 PHP

- Three thousand people line up on your apartment's welcome mat and demand to be shot in their feet. One by one, you oblige them, but halfway through, the http connection times out and the crowd lynches you.
- You shoot yourself in the foot with a gun made with pieces from 300 other guns.

0.61 PicoSpan

- You can't shoot yourself in the foot because you're not a host.
- Whenever you shoot yourself in the foot, someone opens a topic in policy about it.

0.62 PL/I

- After consuming all system resources including bullets, the data processing department doubles its size, acquires two new mainframes, and drops the original on your foot.

0.63 PostScript

- foot bullets 6 locate loadgun aimgun shoot showpage

0.64 Prolog

- You tell your program you wish to be shot in the foot. The program figures out how to do it, but the syntax doesn't allow it to explain.
- Your program tries to shoot you in the foot, but you die of old age before the bullet leaves the gun.

0.65 Python

- You shoot yourself in the foot and then brag for hours about how much more elegantly you did it than if you had been using C or (God forbid) Perl.
- You try to shoot yourself in the foot but you just keep hitting the whitespace between your toes.
- You create a gun module, a gun class, a foot module, and a foot class. After realizing you can't point the gun at the foot, you pass a reference to the gun to a foot object. After the foot is blown up, the gun object remains alive for eternity, ready to shoot all future feet that may happen to appear.

0.66 Revelation

- You'll be able to shoot yourself in the foot just as soon as you figure out what all these bullets are for.

0.67 Ruby

- You shoot yourself in the foot and then have to justify it to all your friends who are still naively using Perl.
- Your foot is ready to be shot in roughly five minutes, but you just can't find anywhere to shoot it.

0.68 SAS

- You spend three hours trying to cut your way through your foot with a rock flake, only to realize that the language was invented before guns allowed you to shoot yourself in the foot interactively in one easy step with no programming.
- You have no idea that the gun, the bullet, or your foot exists. The gun is locked in a safe in a bank vault on the other side of the galaxy, the bullet is locked in a safe in a bank vault in another galaxy, and the people who know the combinations for the safes and bank vaults died ten million years ago. Still, the gun goes off and fires the bullet through your foot.

0.69 Scheme

- You shoot yourself in the appendage which holds the gun with which you shoot yourself in the appendage which holds the gun with which you shoot yourself in the appendage which holds the gun with which you shoot...but none of the other appendages are aware this is happening.
- You vaguely remember something from your Comp Sci 101 class about shooting yourself in the foot, but why should you waste your time shooting yourself using a functional programming language?

0.70 sh, csh, etc.

- You can't remember the syntax for anything so you spend five hours reading man pages before giving up. You then shoot the computer and switch to C.

0.71 SmallTalk

- You spend so much time playing with the graphics and windowing system that your boss shoots you in the foot, takes away your workstation, and makes you develop in COBOL on a character terminal.
- You shoot yourself in the foot and your foot sends doesNotUnderstand: Pain to your brain.
- You daydream repeatedly about shooting yourself in the foot.

0.72 SML/NJ

- You program a structure for your foot, the gun, and the bullet, complete with associated signatures and function definitions. After two hours of laborious typing, forgetting of semicolons, and searching old Comp Sci textbooks for the definition of such phrases as "polymorphic dynamic objective typing system", as well as an additional hour for brushing up on the lambda calculus, you run the program and the interpreter tells you that the pattern-match between your foot and the bullet is nonexhaustive. You feel a slight tingling pain, but no bullethole appears in your foot because your program did not allow for side-effecting statements.

0.73 SNOBOL

- If you succeed, shoot yourself in the left foot. If you fail, shoot yourself in the right foot.
- You grab your foot with your hand, then rewrite your hand to be a bullet. The act of shooting the original foot then changes your hand/bullet into yet another foot (a left foot).

0.74 SQL

- You cut your foot off, send it out to a service bureau, and when it returns it has a hole in it, but it will no longer fit the attachment at the end of your leg.
- ```
  SELECT @ammo:=bullet FROM gun WHERE trigger = 'PULLED';
  INSERT INTO leg (foot) VALUES (@ammo);
  ```

0.75 UNIX

- ```
  % ls
  foot.c foot.h foot.o toe.c toe.o
  % rm * .o
  rm: .o: No such file or directory
  % ls
  %
  ```

0.76 Visual Basic

- You'll only appear to have shot yourself in the foot, but you'll have so much fun doing it you won't care.
- You do a Google search on how to shoot yourself in the foot using Visual Basic. You find seventeen completely different ways to do it, none of which are properly structured. You paste the first example into the IDE and compile. It brushes your teeth.

0.77 VMS

- %SYS-F-FTSHT, foot shot (fifty lines of traceback omitted)

0.78 Windows 3.1

- Double-click the gun icon and wait. Eventually a window opens giving a selection for guns and target areas. Click the "shoot" button and a small box appears with the note "Unable to open shoot.dll, check that path is correct."

0.79 Windows 95

- Your gun is not compatible with this OS and you must buy an upgrade and install it before you may continue. Then you will be informed that you don't have enough memory.

0.80 Windows ME

- There will be too many sudden reboots to allow the bullet to get through, so your foot hangs instead.

0.81 Windows XP

- Some teenage hacker shoots you in the foot with ActiveX. You develop gangrene and die.

0.82 XBase

- Shooting yourself is no problem, but if you want to shoot yourself in the foot, you'll have to use Clipper.

0.83 Xcode

- Your Objective-C and Java programs now have nifty little graphical interfaces and will run on both PowerPC and x86-based architectures, but you still can't shoot yourself in the foot unless you're the superuser.

0.84 XML

- You vaporize your entire lower half with a bazooka.

- You can't actually shoot yourself in the foot; all you can do is describe the gun in painful detail.

0.85 Z

- You write out all the specification of your foot, the bullet, the gun, and the relevant laws of physics, but all you can do is prove that you can shoot yourself in the foot.