

# **Political History of UNIX**

*Andrew Tannenbaum*

MASSCOMP  
Westford, MA 01886

19 January 1984

# Political History of UNIX

*Andrew Tannenbaum*

MASSCOMP  
Westford, MA 01886

## INTRO

This isn't a talk about how Ken Thompson and Dennis Ritchie hacked their beloved and renowned PDP-7 to pieces in a tower on high.

I will not mention the extraordinary Bell Labs research environment - crafty computer science gurus who took precious time off from their highly important research efforts to help beat UNIX into shape.

I'm going to talk about some of the more earthly factors that contributed to the UNIX that we know and love today. Perhaps after hearing my story of how we got here, you'll be more able to follow in their footsteps, bringing your own products similar fame and glory, or at least you'll be better prepared to understand the results of UNIX's unusual upbringing.

There is an incredible number of people riding on the UNIX bandwagon, the size of the UNIFORM audiences is testament to that fact.

## THE CABBAGE PATCH OPERATING SYSTEM

I sometimes think of UNIX as the Cabbage Patch Operating System.

UNIX wasn't a completely new idea, it was an amalgam of good ideas.

It's certainly the latest craze, with people foaming at the mouth lining up in crazed hordes to get a peek, sometimes paying ridiculous prices for an opportunity to use the product. Of course, they say, it's worth it, there are lesser pleasures in life which are far more expensive.

Like the Cabbage Patch Kids, UNIX has existed for a while, lying around in a relatively dormant state, waiting for the market to explode.

Like Cabbage Patch Kids, every UNIX is slightly different (I'm sorry to report) but they're all similar enough to be valuable as members of the group.

Cabbage Patch Kids are the product of a company which doesn't specialize in dolls: Lately Coleco (Connecticut Leather Company) has been concentrating on the Adam computer and Colecovision home video games.

Of course, UNIX is the product of another big toy company, AT&T.

## WHAT TOOK SO LONG?

You might wonder what happened forces have tugged at UNIX during the almost 15 years since UNIX was conceived.

Until recently, UNIX was never an AT&T OS product in the sense that VMS was a DEC OS product. UNIX was always, AT&T would claim, a telecommunications support tool, only because AT&T was a restricted monopoly and it was forced by legal consent decree to stay out of the computer business.

This important idea here is that AT&T wasn't allowed to compete in the computer marketplace, they weren't allowed to sell a product which would compete with IBM or DEC or any of your companies. This caused UNIX to grow up in unusual circumstances, with some benefits and some disadvantages.

In the case of most computer products, a computer company sees that the marketplace yearns for a certain product or service, and then it struggles to create that product or service before someone else does, so that it might make a killing before the rest of the market produces clones. This wasn't the case in the early days of UNIX, though it is certainly the case today.

When UNIX left the caring hands of Thompson and Ritchie, it was soon handed off to an entity called the UNIX Support Group (USG). The people who controlled the progress of USG controlled the future direction of UNIX through the 1970's.

## HOW DID BELL USE UNIX?

Within AT&T, remember that UNIX was a tool, used by the Bell System projects which dealt with research and development of different parts of the telephone network - like switching, network planning, service order processing, and directory production.

UNIX was the prize in a tug-o-war between these various projects, UNIX was controlled by the telco projects with the most bucks, projects like BANCS and 5ESS, and the BTL UNIX releases sometimes had special purpose software in them with just these projects in mind. These were not the needs of the programmer, or any other common class of user, they were the specific needs of telco projects.

## PROGRAMMER'S WORKBENCH

What was the Bell System going to do with UNIX?

Bell Laboratories had an enormous investment in expensive computer equipment: IBMs, UNIVACs, Honeywells, and such. It was decided that it would be efficient to have a coherent USER interface to all these systems. What I'm trying to say here is that all the drones who punched cards for the 360's got jealous of the UNIX users with their timesharing terminals who didn't have to wait overnight for job turnaround. Today it might strike you as strange that a Teletype Model 33 user was a subject of envy: those days are gone indeed.

Anyway, one of the major UNIX forces at Bell Labs devoted itself to producing PWB, the Programmer's workbench. Programmers would learn one editor (ed) one command interpreter (sh) and be able to submit their batch jobs over rje links from one UNIX terminal. UNIX was to have COBOL syntax checkers and dump analyzers. The idea was to leave the crunching to the big batch machines, and let UNIX front end handle the humans.

Interestingly enough, I heard Bill Joy give a speech in Massachusetts where he claimed that today's UNIX workstations should be used like yesterday's terminals, and that our workstations should be networked to large mainframe CPU's which can do the supercrunching. This philosophy is not at all unlike that of PWB.

Within the Bell System, programmers used UNIX to talk to their IBM, UNIVAC, and Honeywell mainframes.

The Bell System looked at UNIX as a small machine OS. ARPA had different ideas, UNIX would be the successor to TENEX.



some crazed grad student slave labor to whip up a product that the Bell System would have taken forever to produce. The incentive just wasn't there in the Bell System.

As long as UNIX ran on PDP/11's, UNIX was never taken seriously by people with "real work" to do. No one really believed that you could actually PORT an operating system, it just wasn't done. The PDP/11 was nice and cheap, but you couldn't run big processes on it.

### \*EXPLOSION\*

When the VAX came out, you could address memory out the wazoo. Not only that, but a bunch of Bell Labbies actually PORTED UNIX to it, in \*NATIVE MODE\* without a gargantuan effort. The VAX had virtual memory and number crunching capability, and Berkeley had hackers interested in exploiting it.

When the VAX 11/780 emerged as a UNIX workhorse, 4.1bsd emerged as the UNIX system of choice for the VAX. Outside the Bell System, practically no one ran Bell System UNIX on their VAXes. People would buy a 32v UNIX system from WECO just for the license, they'd never even read the tape, they'd then send their license to UCB and get a copy of the latest BSD release.

Why? The UCB system was more programmer friendly. Eventually, it had paging, job control, faster I/O and higher throughput, support for many non-DEC I/O devices, reasonable though not perfect support software like screen editors and mailers, and compilers for languages like lisp and pascal. UCB users were happy because they got the toys that they wanted to play with.

### BELL ATTITUDES

Eventually, smart users throughout the Bell System were running hacker-friendly 4bsd, and USG first responded with "it's not really faster, it only SEEMS faster." Well, the hackers wanted a system that seemed faster.

Why couldn't the Bell System respond? In a nutshell, BTL CS Research was too small and godly. USG was too mundane. There was nothing in between.

Yes, the Bell System does have an incredible record of innovation. They have UNIX, as well as some lesser accomplishments like the transistor and negative feedback, fundamental development of lasers and Nobel prize-winning work in radio-astronomy. They even do some telecommunications work.

Looking at it in another light, Bell Labs has more PhD's than any other organization in the world. If I remember my propaganda correctly, they have well over 7,000 PhD's, and another 20,000 engineers who can actually get the work done. While the Bell System has been a leader in innovation, they haven't done it by making efficient use of their resources, and indeed they were never under any pressure to. For this reason, a small and elite crew of hungry guerrillas like the UCB students who worked on BSD were able to have a major impact on a small piece of the Bell System's workspace.

What would have happened if there was a group of 20 hyperactive hackers developing UNIX over the years?

The BTL environment consists of mostly telephony engineers, EE's with communications backgrounds. They are ed users, 300 baud silent 700's, tek graphics scopes. Well educated, but with poor understanding of state of the art interfaces and very stubborn and arrogant. Ed and sh are just fine, thank you.

Bell was not the SAIL/ALTO/MIT/ARPA LISP/EMACS DEC10 AI environment, which is where much of the winning software from the 60's and 70's came from. Maybe not AI, but at least some good software.

Dennis Ritchie says that BTLCS was waiting for a DEC10 that never got funded. If they got it, UNIX

would never have gotten off the ground. What do we owe to the Bell System administrator who turned down the DEC10?