Dennis Ritchie - The Computer Science Pioneer without whom there would be no Jobs

Outline

1. Dennis Ritchie - Tributes

2. Dennis Ritchie - The Man

3. Dennis Ritchie - The Founder of C
   - Success of C
   - Critique of C

4. Dennis Ritchie - The Founder of Unix
   - Portability of Unix
   - History of Unix
   - The Fellowship of Unix
   - On Success of Unix
   - What Unix did not do

5. Some Photos and Fun

One Language and Operating System to Rule All [1]

One C to Program all Operating Systems
One Unix to Rule all Architectures
and Deploy in the Internet to Bind Them All

[1] http://en.wikiquote.org/wiki/The_Lord_of_the_Rings

“It has always seemed strange to me . . . The things we admire in men, kindness and generosity, openness, honesty, understanding and feeling, are the concomitants of failure in our system. And those traits we detest, sharpness, greed, acquisitiveness, meanness, egotism and self-interest, are the traits of success. And while men admire the quality of the first they love the produce of the second.” **John Steinbeck, Cannery Row**

Calling for October 30 to be celebrated as Dennis Ritchie Day by Tim O’Reilly of O’Reilly Media.

Dennis Ritchie - Tributes I

“Pretty much everything on the web uses those two things: C and UNIX. The browsers are written in C. The UNIX kernel - that pretty much the entire Internet runs on - is written in C. Web servers are written in C, and if they’re not, they’re written in Java or C++, which are C derivatives, or Python or Ruby, which are implemented in C. And all of the network hardware running these programs I can almost guarantee were written in C. It’s really hard to overstate how much of the modern information economy is built on the work Dennis did.” **Rob Pike.**

“There’s that line from Newton about standing on the shoulders of giants. We’re all standing on Dennis’ shoulders.” **Brian Kernighan.**

Dennis Ritchie - Tributes II

“Jobs was the king of the visible, and Ritchie is the king of what is largely invisible. Jobs’ genius is that he builds these products that people really like to use because he has taste and can build things that people really find compelling. Ritchie built things that technologists were able to use to build core infrastructure that people don’t necessarily see much anymore, but they use everyday.” **Martin Rinard, Professor of Electrical Engineering and Computer Science at MIT and a member of the Computer Science and Artificial Intelligence Laboratory.**[2]

[2] http://www.wired.com/wiredenterprise/2011/10/thedennisritchieeffect/

Dennis Ritchie - The Man

Dennis MacAlistair Ritchie was born on September 9, 1941 and found dead on October 12, 2011. He graduated from Harvard with degrees in Physics and Applied Mathematics. He started working for Bell Labs in 1967 and was awarded PhD in 1968 by Harvard University under the supervision of Patrick C Fischer.

He created the *C* programming language and along with Ken Thompson, the *Unix* Operating System, without which there is no Internet.

Ritchie was awarded the Turing Award along with Ken Thompson in 1983, Hamming Medal from IEEE in 1990 and the National Medal of Technology in 1999

His homepage can be seen at http://cm.bell-labs.com/who/dmr/.

Dennis Ritchie - The Founder of C

The *C* programming language evolved from the language *B* developed by Ken Thompson which was, in turn, based on *BCPL* during the years 1969-73.
Dennis Ritchie on *C*:

“C has the power of assembly language and the convenience of . . . assembly language.”

“C is peculiar in a lot of ways, but it, like many other successful things, has a certain unity of approach that stems from development in a small group.”

“I can’t recall any difficulty in making the C language definition completely open - any discussion on the matter tended to mention languages whose inventors tried to keep tight control, and consequent ill fate.”

Success of C I

Ritchie identifies the success of *C* to the following factors: [3]

1. Being a part of *Unix* operating system and thus available to thousands of users.
2. On the other hand, *Unix*'s use of *C* made *Unix* itself popular as it was highly portable.
3. A simple and small language with simple and small compilers making it easy to port it to different architectures and processors.
4. Generating time- and space-efficient programs is not difficult.
5. Abstracted sufficiently from machine details to achieve program portability.
6. It was developed as a tool to write programs that did useful things.

Success of C II

7. A parsimonious, pragmatic approach where the essential needs of the programmer are covered but it does not try to supply too much.
8. *C* has remained remarkably stable and unified.
9. While portability was not one of its original design goals, it has done remarkably well in this area by allowing even operating systems to be written across all types of machines - from the personal computer to the supercomputers.

[3] http://cm.bell-labs.com/cm/cs/who/dmr/chist.pdf

Critique of C I

Ritchie critiques *C* as follows: [4]

1. Relationship between arrays and pointers has been one stumbling block for beginners.
2. Declaration syntax mimics expression syntax.
3. *C* evolved from typeless languages and so is not strongly typed. The current compilers help a lot in identifying errors across declaration and usage but earlier compilers would not do it.
4. The indirection operator, *, is a unary operator that works well in simple cases but in more complex cases requires parentheses to parse.

Critique of C II

5. Has considerable power to describe important concepts such as vectors whose length varies at run time. Unlike Algol and Pascal, its contemporaneous languages, it does not fix the length of the arrays. It treats strings and arrays the same and distinguishes strings by introduction of a null character.
6. The array treatment, however, has implications for optimization. It also makes it hard to treat arrays as more fundamental data types on which operations can be defined as a whole.

[4] http://cm.bell-labs.com/cm/cs/who/dmr/chist.pdf

Dennis Ritchie - The Founder of Unix I

Dennis Ritchie on *Unix*:
UNIX is simple. But it just needs a genius to understand its simplicity.

In a paper presented at the Tenth Hawaii International Conference on the System Sciences, Honolulu, January, 1977, Ritchie gave a “retrospective” on Unix.
Systems should be written in a highlevel language that encourages portability. Manufacturers who build more than one line of machines and also build more than one operating system and set of utilities are wasting money.

Dennis Ritchie - The Founder of Unix II

The greatest care should be taken to ensure that there is only one format for files. This is essential for making programs work smoothly together.

He adds on his Home Page that this has not been done by any OS and his complaint today is “now I am harried by MS Word documents that cannot be read by MS Word”.

Portability of Unix

Ritchie said in an interview that his greatest contribution was to come up with the idea of porting Unix from PDP-11 to Interdata system . This lead to one of the greatest advances in Operating Systems - operating systems that were not tied down to a particular hardware.

Here is what Tim O'Reilly of O'Reilly Media has to say:[5]

"it was Unix's culture of collaborative development and architecture of participation that was the deepest tap root of what became the open source software movement, and not coincidentally, much of the architecture of the Internet as well."

[5] http://radar.oreilly.com/2011/10/dennis-ritchie-day.html

History of Unix I

- In 1960s, MIT, GE and Bell Labs worked on an operating system called *Multics*. Bell Labs pulled out of the project when the cost and deadlines were overrun.
- The fellowship were the last to move out of Multics project. Ken Thompson developed a file system and also a paging system on Multics as also a game called **Space Travel** which he then ported to PDP-7.
- The fellowship then built a *hierarchical file system*, *computer processes*, *device files*, *command interpreter (shell)*, *time-sharing from Multics* and some utility programs.
- With a promise of providing text editing facilities, Thompson and Ritchie managed to get Bell Labs to provide them with a PDP-11/20.

History of Unix II

- For the first time, in 1970, Unix ran on a PDP-11/20 with a text formatting program called *roff* and a text editor, all of which were written in the assembly language.
- In 1972, Unix was re-written in C, changing forever how Operating Systems were developed!
- Unix was then distributed along with the source code to many universities which showed interest and thus, a de facto standard of operating systems was born.
- At Berkeley, Bill Joy worked on the Unix system and the Berkeley Software Distribution (BSD) was born. This included the *C Shell*, *vi*, *TCP/IP stack* and other notable features for the first time and the Internetworking was born

and the rest, as they say, is history!

The Fellowship of Unix I

Ken Thompson, Dennis Ritchie, M.D.McIlroy and J.F.Osanna formed the fellowship around the Unix system. This is what Ritchie says about why they wanted to continue the project of Multics:

“What we wanted to preserve was not just a good environment in which to do programming, but a system around which a fellowship could form. We knew from experience that the essence of communal computing, as supplied by remote-access, time-shared machines, is not just to type programs into a terminal instead of a keypunch, but to encourage close communication. ”

The Fellowship of Unix II

Dr Pike says that the thing he misses most from the 1970s at Bell Labs was the terminal room. Because computers were rare at the time, people did not have them on their desks, but rather went to the room, one side of which was covered with whiteboards, and sat down at a random computer to work. The technical hub of the system became the social hub.

On Success of Unix

1. Simple enough to comprehend but complex enough to do most of what its users need.
2. User Interface is clean and terse to the point of being cryptic.
3. Unix file system is simple in structure but more powerful and general than those found in any other OS of that or the recent times. Every file and device is considered a sequence of bytes, concealing the properties of the physical devices from the user.
4. Written in a high level language making it portable.
5. Security was better than most in terms of permissions allowed to access and modify files and directories.

What Unix did not do

Unix was not built to handle real-time responses as in true real-time systems as it does not allow one to change the scheduling priority or property of a process. This is true even today.

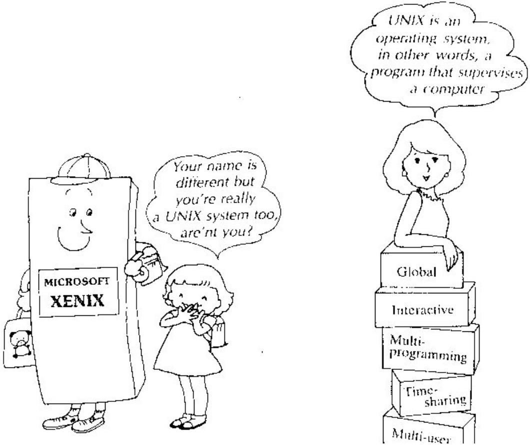
In the days of its implementation, Unix did not have

1. A proper IPC - neither shared memory nor any other IPC other than pipes were present.
2. Asynchronous I/O was absent.
3. A hierarchical file system as compared to the file systems of its time.

Ken Thompson and Dennis Ritchie at a PDP-11[6]

[6] Courtesy: Dennis Ritchie Home Page

An Advertisement on Unix[7]


MICROSOFT
XENIX
Your name is different but you're really a UNIX system too, are'nt you?
UNIX is an operating system, in other words, a program that supervises a computer
Global
Interactive
Multi-programming
Time-sharing
Multi-user

[7] Courtesy: Dennis Ritchie Home Page

References I

- Dennis Ritchie.
  http://en.wikipedia.org/wiki/Dennis_Ritchie

- Without Dennis Ritchie, there would be no Jobs,
  http://www.zdnet.com/blog/perlow/without-dennis-ritchie-there-would-be-no-jobs/19020

- Dennis Ritchie Day. http://radar.oreilly.com/2011/10/dennis-ritchie-day.html

- Computers : History : Pioneers : Ritchie, Dennis
  http://www.dmoz.org/Computers/History/Pioneers/Ritchie,_Dennis/

- Dennis Ritchie Home Page at Bell Labs.
  http://cm.bell-labs.com/who/dmr/

References II

- Dennis Ritchie: The Shoulders Steve Jobs Stood On. http://www.wired.com/wiredenterprise/2011/10/thedennisritchieeffect/
- History of Unix. http://en.wikipedia.org/wiki/Unix#History
- Unix History: An early video from AT&T advertising Unix http://www.youtube.com/watch?v=7FjX7r5icV8
- The One Ring http://en.wikiquote.org/wiki/The_Lord_of_the_Rings