Podcast: ACM ByteCast
Episode: Episode 85 - Eric Allman

Welcome to ACM ByteCast, a podcast from the Association for Computing Machinery, the world's largest education and scientific computing society. We talk to researchers and innovators who share experiences and lessons they've learned, and their visions for the future of computing. Today's conversation is hosted by Scott Hanselman of the Hanselminutes podcast. He talks with Eric Allman, the developer of sendmail and its precursor delivermail in the late 1970s at UC Berkeley. In 1998, he and Greg Olson co-founded Sendmail Inc.

Scott starts by asking Eric why he retired. Eric responds that during COVID it seemed that the timing was right. He felt like he wasn't doing anything of use at UC Berkeley, and his husband has a degenerative neural disease and wanted to travel before he had trouble walking, so it made sense to retire. He's been with his husband, also a well-known computer scientist, for more than 40 years. Eric doesn't do much computing anymore, but he has founded a 501c3 called the Berkeley Historic Building Fund, and he spends his time working on that.

Scott asks about truly letting go of computing once you retire. Eric shares about his work at UC Berkeley that spanned a variety of topics, from user interfaces to neural networks, and how much the industry has changed over that time period. He says there used to be the idea that each country would only need one mainframe and there was the concept that all this technology would bring about a utopian future where the ability to communicate would increase and there would be no need for wars and such. They both express disappointment at that lack of reality nowadays.

Both gentlemen speak at length about the initial concept of email technology being a tool to bring communities closer. The original sendmail was created before the dawn of the internet and Scott asks Eric to relate it to machine learning. Eric says that it was basically setting up something where you could use effectively an AI technique to route email, but it hadn't occurred to him to compare it to large language models. Eric says that one aspect of sendmail that aged well is the use of MPX files and later IPC or different implementations of Syslog. Scott also expresses appreciation for sendmail rules that are recursively applied, which Eric says is part of the genius and frustration with it.

The conversation shifts to AI, and Eric says that in the beginning, they were using neural networks but not really calling it AI. The goal was to get the technology to the point where they could run a relatively small neural network. The new models are much more sophisticated and Eric enjoys playing with Claude and testing its capabilities. He is clear that he doesn't believe that AI has consciousness though, and Scott agrees. For example, Claude can run a debugging program but then add back the original bug because it doesn't "think". Eric says AI is helpful as

long as you utilize it effectively. Scott says he goes back and forth between loving and hating AI for assistive coding.

Eric finds himself using code writing for things around the house like his wine cellar, or various other projects. They discuss the merits of using Python and C# programming and say that there's a benefit of both. They also discuss the back and forth between TypeScript and JavaScript, likening JavaScript to English, in that it doesn't always make sense but it's the language we use the most. They wrap up the conversation by discussing retirement and the joys of being able to pick up computer science and put it down as they see fit, and enjoying what you're doing at any given time. Scott finishes the conversation by thanking Eric for joining.

Learn more about ACM Bytecast at https://learning.acm.org/bytecast