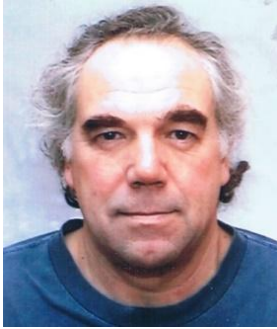


60 Seconds with ...

Les Hatton



Les Hatton Ph.D. is a mathematician notable for his work on failures and vulnerabilities in software-controlled systems. Originally an award-winning geophysicist, he switched careers in the early 1990s to study software and systems failure.

He has published widely and his theoretical and experimental work on software systems failure can be found in "Nature", "IEEE Transactions on Software Engineering", and numerous other journals. His 1995 book "Safer C" helped promote the use of safer language subsets in commercial embedded control systems.

In 2012, he took early retirement to try to understand recurring patterns he was observing in the many millions of lines of code he had then studied. In 2014 he proved that all software systems asymptote to the same length and alphabet distribution. In 2015, with the biochemist Greg Warr, he extended this to the distribution of known proteins and has since demonstrated similar behaviour in software, biology, music, literature and many other systems. Their findings have been recently published in the 2022 book: "Exposing Nature's Bias: the hidden clockwork behind society, life and the universe".

He is an emeritus Professor of Forensic Software Engineering at Kingston University, London and spends his spare time doing computational biology in his shed!

What first attracted you to working in the field of software reliability?

I was an earth scientist for the first 20 years of my career. In meteorology and later in seismology I found glaring examples of software failures frequently derailing the science. In 1992, I changed careers to study software reliability to try and find answers.

What aspect of your career are you most proud of?

I don't look back so taking early retirement in 2012 to work full-time on a problem in information theory with an old college friend, a biochemist, which eventually led to the discovery of a new Law of Nature – the Fundamental Law of Inequality. This has been hidden in plain sight for 5,000 years, but manifests itself in literature, language, wealth distribution, software and notably, genomics. The book just came out after some 15 years of work. Curiously, although unintentional, this mostly resolved my original questions about software reliability, but its implications for evolutionary biology are particularly exciting.

What advice would you give to yourself age 12?

Practice the guitar more.

What future changes would you like to see in the field of System Safety?

The primacy of engineering professionalism and judgement over management, media and indeed politics.

What's your most favourite quote or motto?

Albert Einstein. When he heard that a book titled 100 Authors against Einstein was published in Germany, he said:

"If I were wrong, then one would have been enough."

If you could learn to do anything, what would it be?

Fly.

If you could be any fictional character, who would you choose?

None of them really. Fictional characters are defined by somebody else. I'm far happier with my own path. If I could play any fictional character, it would probably be "the man with no name", although I'm not very good with horses.

What's the best piece of advice you've ever been given?

In spite of his unfailing belief in the opportunities of education, my dad's career advice was:

"It's not what you know, it's who you know".

The younger idealistic me disagreed, but boy was I wrong.

Which work of art or fiction best sums up your experiences with Covid-19?

It would have to be a work of fiction – the government's Covid-19 figures. I volunteered to work in a Covid modelling group through the pandemic and the data quality was truly awful – some of the worst I've ever had to work with.

"Taking early retirement in 2012 to work full-time on a problem in information theory with an old college friend ... eventually led to the discovery of a new Law of Nature – the Fundamental Law of Inequality"