

**Quarter of a Century  
of Inspiration  
and Friendship**  
**#RememberingJoe**

# Remembering Joe, a Quarter of a Century of Inspiration and Friendship

2019-04-23 by Francesco Cesarini

I first came across the name Joe Armstrong in 1994, when I bought the first edition of Concurrent Programming in Erlang, a book he co-authored. Our first interaction happened in 1995, when I was looking for a company interested in sponsoring my Master's thesis. I dialled the number for Ellemtel Utvecklings AB, home of the Ericsson computer science laboratory, asking to be connected to Joe Armstrong. Getting a Hello, my unprepared opening line was Skall jag ta det på Engelska, or would you prefer if I took it in Swedish? A silent pause was followed by laughter, the same laughter many of us have come to associate with Joe.

## Shouting at each other through the walls

Internships at the computer science lab were an immersion in enthusiasm and creativity. The lab had an academic, non conformist, almost anti-establishment, feel to it. Segregated to the corner of the building, pipe and cigarette smoke coming from some of the rooms. I am sure the cleaning staff were asked to “forget” the department existed, or maybe, they did not dare venture in there after hours. ISO 9000 reviews (and office moves) were a conspiracy to get Joe (and Robert) to clean their desks. But what mattered

was not how neat your desk was, but the drive to innovate with an aim to further our understanding in computer science. You did not use an existing web server, you wrote your own so as to understand how HTTP worked and suggest improvements and extensions. You did not take Ericsson's high security for granted, you found your way around firewalls and tried to outsmart the highly paid system administrators. When I got distributed Erlang working between my University and Ericsson UNIX accounts, I asked Joe if this was right. He walked in the hall and laughed loudly. It was his call for a show and tell, sharing knowledge with his curious colleagues who quickly make their way to his office. Sometimes, there was no need to make it there in person. Mike and Robert were always given the office next to his, so they could shout at each other through the walls.

## Ahead of his time

Brainstorming sessions with Joe were the best. We once spent a good part of an afternoon finding a solution to a problem. I ended up working into the early hours of the morning on the implementation, and indeed, it worked. When Joe walked into the office the following day, he pops his head into my room and says You know the solution we came up with yesterday? Forget about it, it will not work. My jumping up and down saying it does work, it works, here it is, was fruitless. He replied, no, we were wrong, and walked off. He was often ahead of his time, in both thoughts and problems he was solving, and had probably discovered some edge and borderline cases to our solution which was too frivolous to share. This is where you learnt to filter his ideas, take those you understood or believed in and parked those you did not like or understand.

## Solution to a problem

Sometimes, you would be walking outside his office, he would wave you in and share one of his daily epiphanies, hacks or articles and academic papers on a one-on-one basis. One of these hacks happened in conjunction with a major Erlang release in 1995. They had snuck in List Comprehensions and Funs in the language (product management had prioritized other features) and Joe showed me how they allowed you to implement clean and easy to maintain code, hiding recursive patterns and isolating side effects in a single location. Showing me the shortest implementation of QuickSort I had ever seen, he mentions you can solve the Eight Queens problem in six lines of code. I spent two nights trying to figure out a solution in my head, to no avail. Weary of a third sleepless night, I ask Joe for a solution only to be told: I have no idea, I never solved it myself. Look on the internet. WebCrawler and Lycos, whom had been around for a year, failed to provide a solution. Whilst I love Joe to bits, that particular day, I could have strangled him.

I think he was used to it (mentee students wanting to strangle him), as he claimed that we have always mistreated geniuses ahead of their time. Guttenberg was bankrupted, Galileo

escaped the death penalty when put on trial, whilst Darwin's theories were mocked by many. Fast forward to ten - fifteen years ago, governments and corporations alike were, instead of embracing peer to peer, persecuting those who built services on top of them. Ok, governments and corporations might not have strangled anyone, but they sure did sue them or try to put them in jail. Joe was disgusted by their treatment (he was the one who showed me how bittorrent worked), but he did smile when I told him one of the Pirate Bay founders claimed in an interview he was going to use his time in jail to learn Erlang. (I've looked everywhere for the article where I read about it, if anyone has a link, please send it to me [francesco.cesarini@erlang-solutions.com](mailto:francesco.cesarini@erlang-solutions.com))

## Leaving Ericsson

In December 1998, a few days after Erlang was released as open source, Joe and some of his Computer Science Lab colleagues left to found Bluetail, later acquired by Alteon Websystems, who in turn got acquired by Nortel. We were both very busy building products these years, and met mainly at conferences and workshops. I do recall being told of the buzz at Nortel when Erlang Solutions launched its first website, and when the dotcom bubble burst, receiving a call from Pekka, a colleague of his, saying he, Joe, Robert, Jane and many others in the team had just been fired! It did not take long for Nortel, in 2003, to start advertising for Erlang developers with ten years of experience, not realising they had recently let go seven of the ten people who at the time fit the bill. Nortel would not have hired me, at the time I only had 8 years of Erlang. You've got to love random acts of management.

Seeing his redundancy as an opportunity, he decided to make a stint in academia, working on his PhD through SICS, the Swedish Institute of Computer Science. Making reliable distributed systems in the presence of software errors is a must read for anyone trying to understand the rationale behind systems you write once and run forever. Folklore (Joe after a few beers) has it that after a couple of years at SICS, he walks into his supervisor's office and submits the full thesis. His supervisor looks up surprised and says, This is not how it is supposed to work. I should be giving you feedback as you write it. Oh, goes Joe, Let me know what you think. After SICS, Joe returns to Ericsson, completing the full circle. In 2014, becoming an Adjunct Professor at KTH, continuing to inspire students through his magical ability to pique their curiosity. Just like he inspired me back in 1995.

## Quest to learn

Joe knew that programming languages were not about popularity or beauty contests. They were all about solving problems and progressing the industry. He was just as excited about Rich Hickey, Don Syme or Simon Peyton-Jones's success stories, and wished Haskell, F# and Clojure to do as well. More recently, he got all excited about

Sylvan Clebsch's Ponylang. When we had dinner with the Go team, he enthusiastically explained Erlang's concurrency error handling mechanisms to Ken Thompson. He was encouraging Ken to integrate similar semantics in Go. Erlang will not be around forever, he once told me in the 90s. Something better will come along. I don't think we realised back then, that whatever will come along, is going to be heavily influenced by his work!

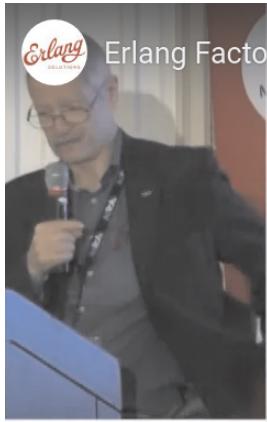


There were obviously languages he did not like, C++, Java and JavaScript (read Node.js), backed by well motivated reasons. But despite not liking them, he wanted to meet the inventors behind these languages in his quest to learn more, understand their motivation and share his ideas. He asked me to try and get Brendan Eich to speak at the Erlang Factory in San Francisco and was hoping to meet Bjarne Stroustrup at Code Mesh in London. I once had to drag him out of Google in Mountain View when James

Gosling was in his office and we happened to walk outside. Joe asked me if I thought it would be impolite for him to just go in and introduce himself, to which I suggested (for the sake of our host) that it might be better not to, and instead, get a proper introduction.

## The Trio, or even Quartet!

We are talking about Joe, but let's not forget that for a good part of his career, he was part of a team together with Robert and Mike. They were led by a patient Bjarne, who gave them free rein in solving telecom-related and being interested in what, and not how. It was the space they needed to innovate. I am not sure any of them on their own have been able to create Erlang, but together, they were able to leverage each other's strengths and succeeded at creating both Erlang and OTP. I tried multiple times to get them on stage



Erlang Factory SF 2015 Keynote - From WhatsApp to Outer Space



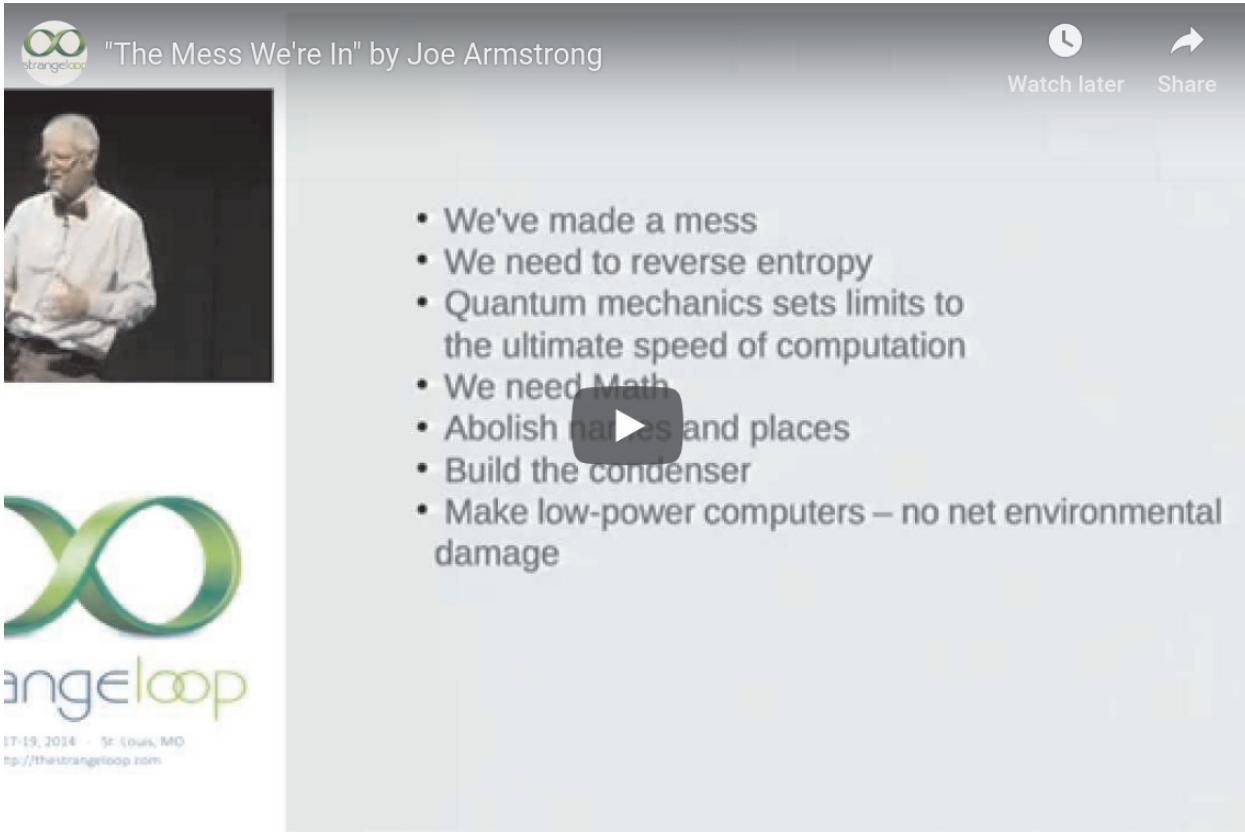
## The New Hardware Landscape

- 50 Billion connected devices
- 5G (I have to say this :-)
- 1000+ core CPUs
- 5 Gb/s radio access
- CPU power is free
- Memory is free
- Communication costs
- Energy costs

together to show the dynamics of the trio, but it was always scripted (even Joe's parts!). I succeeded the third time, at the 2015 San Francisco Erlang Factory Keynote, From WhatsApp to Outer Space, where the last 15 minutes of the talk, they go off script and start asking each other questions, to the roaring laughter of the audience. It shows Joe to be the inventor, Robert to be the one interested in aesthetics and Mike the finisher and end user with industry experience. It was wonderful to see their team dynamics, friendship and bonds still in place and going strong, twenty years on.

## Beyond Erlang

In more recent years, Joe started talking about topics beyond Erlang. His Strangeloop keynote, The Mess We're In, focused on one of his pet peeves in the software industry. As a result of computers becoming faster, software seems to become more complex, and hence, slower. He refers to the laws of physics, something the software industry has tried to defy for decades. Joe (a physicist by training) applied the laws of physics to computer science and distributed systems. Synchronously passing messages, shared memory or attempting to share data faster than the speed of light. If you hear a programmer say you can't do this, it defies the laws of physics, you now know where their quote comes from.



The screenshot shows a video player interface. In the top left corner is the Strangeloop logo, which consists of two interlocking green loops. Next to it is the title "The Mess We're In" by Joe Armstrong. In the top right corner are two small icons: a circular one with a play button and another with a share symbol. Below these are the text "Watch later" and "Share". On the left side of the video frame, there is a small thumbnail image of Joe Armstrong, an older man with glasses and a white shirt, standing and speaking. The main video area contains a list of bullet points and a play button icon. The list is as follows:

- We've made a mess
- We need to reverse entropy
- Quantum mechanics sets limits to the ultimate speed of computation
- We need Math
- Abolish names and places
- Build the condenser
- Make low-power computers – no net environmental damage

On the left side of the page, there is a large Strangeloop logo. It features a stylized green infinity symbol above the word "strangeloop" in a lowercase sans-serif font. Below the logo, the text "17-19. 2014 - St. Louis, MO" and the URL "http://thestrangeloop.com" are visible.

## The Mess We're In

One of the many projects he was planning on doing after retirement was to interview his heroes, and if he got enough interviews, publish the results in a book. He suggested interviewing Alan Kay on stage at Code Mesh in London in 2016. I have never seen a conference audience so mesmerised. A similar reaction happened in the Let's #TalkConcurrency panel discussion in November 2018, where we were able to get Sir Tony Hoare, Carl Hewitt and Joe Armstrong to discuss the past, present and future of concurrency models. Joe was originally supposed to run the interview, but as many of us felt he had just as much to say as Tony and Carl, we got him on the panel instead. I am glad we did, as we covered three different, but overlapping approaches to concurrency, each created to solve a different problem. Travelling to Cambridge with Joe for the recording, it was obvious he was not well. His lungs were at 60% capacity, and he often ran out of breath. The pulmonary fibrosis was evident. But we all hoped they would be able to keep it under control.

## Lung Research

On Saturday April 20th, 2019, I get the dreaded WhatsApp message that Joe had just passed away. Just the day before, the news was more positive, they had narrowed down the diagnosis and had adapted the treatment accordingly. Unfortunately, it was too late.

Joe leaves behind his wife, Helen, his children Claire and Thomas and two cats, Zorro and Daisy, who figure in various programs. Joe had named a previous generation of cats Wittgenstein and Schopenhauer but reality, in the form of Helen, renamed them; they became known as Tubby and Tiger. He also leaves behind many friends, colleagues, students and followers who will continue his work, spreading his ideas and ensuring they evolve and keep on getting embedded in mainstream programming practices.

We are all glad Joe got to see how his work has impacted the world around him, and how Erlang Style Concurrency and OTP are being adopted in the realm of distributed, fault tolerant systems which have to scale on multi-core architectures. Basically, the immediate future.

Thank you Joe for being yourself. Thank you Helen for supporting him in doing what he loved the most. And thank you Claire and Thomas for helping bring up an older brother called Erlang. As the old saying goes, no one truly understands concurrency until they have their second child (or cat).

Helen and Claire kindly requested donations to be made to lung research, with Claire starting a fundraiser on Facebook:

<https://www.facebook.com/donate/312270252802913/312270286136243?sfns=mo>

In the US there's the American Lung Association:

<https://www.lung.org/get-involved/ways-to-give/>

In the UK British Lung Foundation:

<https://www.blf.org.uk/donate>

In Sweden Hjärt-lungfonden:

<https://www.hjart-lungfonden.se/Stod-oss/>

## Remembering Joe <3

#RememberingJoe with credits to the wonderful community - many thanks for sharing  
<3

Many thanks and photo credits to: @samaaron; @puredanger; @rolodato;  
@bentanweihao; @MachinesAreUs; @bltroutwine; @danielozcps; @christian\_fei;

@bryan\_hunter; @EuenLopez; @cacorriere; @t\_sloughter; @acscherp;  
@strangeloop\_stl; @janjiss; @zhanghu; @MarioAquino; @MakisOtman; @perlhack;  
@colrack; @aodanne; @gsarwate; @scrogson; @gar1t; @RogerSuffling; @gturitto;  
@lelf Special thanks to @michaelbridge for the feature photograph.

