Rust - A programming language created from a broken elevator, can 'surpass' both C and C++

Rust, a programming language was born from an elevator failure in an apartment building. It was quickly accepted by users and widely applied.

A new programming language was born from an elevator failure in an apartment building. It was quickly accepted by users and widely applied. It is the Rust programming language, even chosen to improve or create new software instead of C , C++ or Java.

In 2006, Graydon Hoare, a programmer at Mozilla, the company that develops the open-source web browser Firefox, returned home from work to find his apartment building elevator broken due to a software bug. This was not the first time such an incident had occurred.

As a programmer, Hoare knew that the cause of the problem was a memory error. Elevator operating software is often written in C or C++, which is prone to memory overflow errors.

Frustrated by the elevator breakdown, Hoare decided to create a programming language that was more flexible, friendly, and less buggy. And so the programming language Rust — named after a plant disease caused by an extremely hardy fungus — was born.

After 17 years, Rust has become one of the most popular programming languages ??with about 2.8 million programmers using it. It is considered by companies like Amazon or Microsoft as the programming language of the future because it solves some common problems in languages ??like C and C++, especially memory

overflow errors.

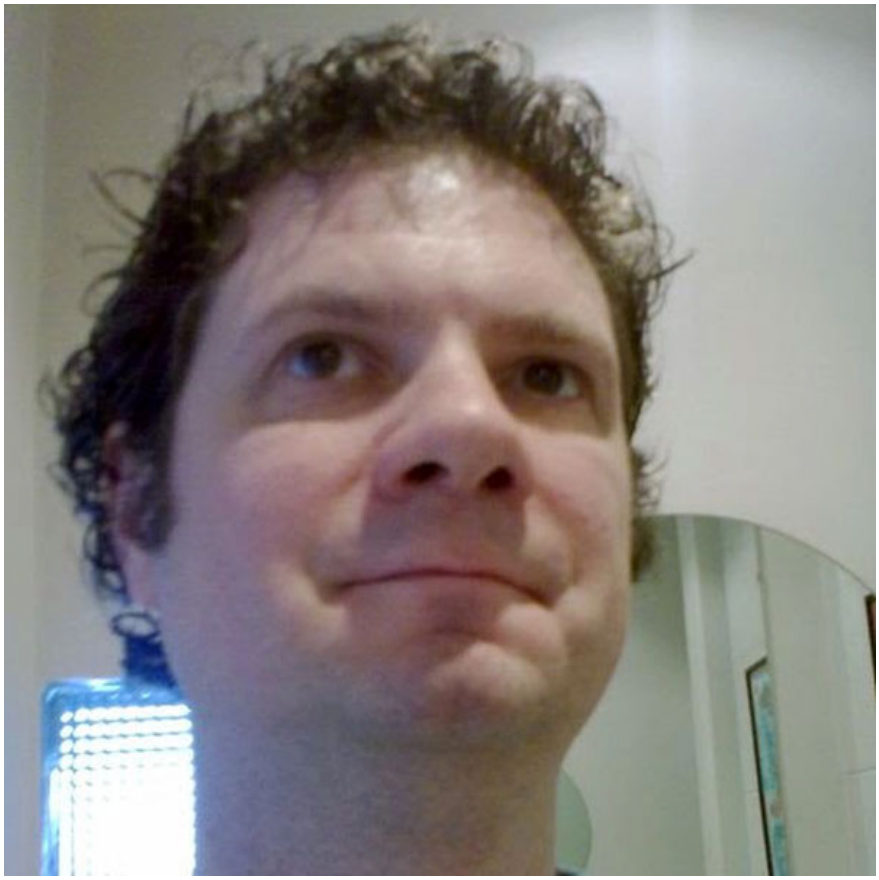
Computer memory is like a chalkboard. When the software is running, data is written to the board for checking, and if it is no longer needed, it is erased.

Each programming language has its own way of managing memory. C and C++ allow programmers to have more control over how and when the software writes data, but this requires them to carefully monitor where the data is written and when to clean up the data. The software can write over an area that already contains data if the programmer forgets to clean up.

Programming languages ??like Java, JavaScript, or Python come with a component that can periodically search for and clean up unused components. This reduces the risk of memory leaks, but requires a lot of resources to run.

Hoare designed Rust to strike a balance between resource consumption and memory cleanliness. Rust automatically finds where data needs to be cleaned up, but requires programmers to follow strict coding rules about how data is used and copied in the software.

Or simply put, Rust will be harder to code but will not have memory errors.



The Rust design team, in addition to Hoare, who has 10 years of experience in the software field, includes other members such as software engineers Niko Matsakis and Felix Klock, who have experience in the fields of memory research and programming languages.

In 2009, Rust was funded by Mozilla as an open source project. By the early 2010s, Rust received support from the worldwide programming community.

On May 15, 2015, the first stable version of Rust was released, and a year later Mozilla launched Servo, a browser engine written in Rust.

In 2017, Rust was used in Firefox's CSS renderer, making it faster. Since then, Rust has been used more widely, such as in Dropbox, Discord, Amazon Web Services.

Research shows that code written in Rust is as efficient as Java but consumes half the energy.

After 17 years of existence, Rust has become a highly regarded programming language, even being proposed by Microsoft leaders for widespread use in writing new software instead of C and C++.

Rust, with its advantages of being fast, simple, friendly and safe, will be used more and more to create new software in the future. Of course, C and C++ will not disappear for decades to come.