

Lenguaje de programación Rust

Fernando Ugarte Ugalde - 201048339, Annia Gómez Blanco - 201108349 *Estudiantes* Instituto Tecnológico de Costa Rica, Escuela de Ingeniería en computación *Sede Cartago Profesor* Esteban Arias Méndez, Curso de Compiladores e Intérpretes IC-5701

Abstract—Rust is a systems programming language that runs blazingly fast, prevents segfaults, and guarantees thread safety. Many modern languages provide this memory safety, but they do it at runtime with references and garbage collection. JavaScript, Java, Ruby, Python, and Perl all fall into this camp. Rust is different. Rust is a statically typed compiled language meant to target the same tasks that you might use C or C++ for today, but it's whole purpose in life is to promote memory safety. By design, Rust code can't have dangling pointers, buffer overflows, or a whole host of other memory errors. Any code which would cause this literally can't be compiled. The language doesn't allow it.

Keywords—*Compilador, Interprete, \LaTeX , Lenguaje de Programación, Mozilla.*

I. INTRODUCCIÓN

Este lenguaje surgió de un proyecto personal desarrollado por Graydon Hoere (trabajador de Mozilla). Empezó a trabajar en el en el año 2006, Mozilla se involucró en el proyecto en el año 2009 y lo dio a conocer en el año 2010. En ese mismo año pasó del compilador inicial escrito en Ocaml al compilador auto-contenido, escrito en sí mismo en Rust. Se compilo así mismo 2011. La primera versión apareció en enero del 2012. La forma más fácil de probar Rust es a través del corralito, una aplicación en línea para escribir y ejecutar código Rust. Si desea probar Rust en su sistema, instálelo y vaya a través del tutorial del libro Guessing Game.

A. El uso de Rust

Así que para los programadores de sistemas Rust parece una gran opción sin embargo hay que recordar que es un lenguaje muy similar a C por lo tanto es de bajo nivel, probablemente diferente a lo que estamos acostumbrados hoy día. En otras Rust ofrece el rendimiento normalmente visto sólo en los idiomas de sistemas de bajo nivel pero la mayoría de las veces, ¡sin duda se siente como un lenguaje de alto nivel!

II. DESARROLLO

Rust es un nuevo lenguaje de programación que se centra en el rendimiento, la paralelización, y la seguridad de la memoria. Con la construcción de un lenguaje a partir de cero y la incorporación de elementos de diseño del lenguaje de programación moderno, los creadores de Rust evitaron una gran cantidad de “legado”

(requisitos de compatibilidad con versiones anteriores) que los lenguajes tradicionales tienen que tratar. En cambio, Rust

es capaz de fusionar la sintaxis expresiva y flexibilidad de lenguajes de alto nivel con el control sin precedentes y el rendimiento de un lenguaje de bajo nivel.

La elección de un lenguaje de programación por lo general implica ventajas y desventajas. Aunque la mayoría de los lenguajes modernos de alto nivel proporcionan herramientas para la seguridad de concurrencia y la seguridad de la memoria, lo hacen con una sobrecarga añadida (por ejemplo, mediante el uso de un GC (recolector de basura)), y tienden a carecer de rendimiento y control refinado.

Para hacer frente a estas limitaciones, se tiene que recurrir a lenguajes de bajo nivel. Sin las redes de seguridad de la mayoría de los lenguajes de alto nivel esto puede ser frágil y propenso a errores. Uno tiene repentinamente para hacer frente a la gestión manual de memoria, asignación de recursos, los punteros colgantes, etc. Crear software que puede aprovechar el creciente número de núcleos presentes en los dispositivos de hoy en día es difícil – asegurándose de que dicho código funciona correctamente es aún más difícil.

A. Primeros Pasos

¡El primer paso para usar Rust es instalarlo! Existen un número de maneras para instalar Rust, pero la más fácil es usar el script rustup. Recordamos que Rust funciona para Linux o Mac y Windows.

Si decide que es el momento de dejar de usar Rust, simplemente ejecuta el script de desinstalación: `sudo /usr/local/lib/rustlib/uninstall.sh` Si usaste el instalador de Windows simplemente vuelve a ejecutar el .msi y este te dará la opción de desinstalar.

Rust es orientado a la concurrencia y a la eficiencia con el control de la memoria explícita pues tiene control de localización y de etiquetas. Tareas de peso muy ligero del tipo de co-rutinas. Cuenta con una compilación nativa y estática. Su modo es estar orientado a la práctica pues es Multi-paradigmático, totalmente funcional, concurrente, Orientado a Objetos con funciones de primera clase con vínculos. Permite romper normas en la práctica, si es explícito dónde y cómo en sentido que se sale del esquema al que se esta acostumbrado.

III. CONCLUSIONES

Existen diferentes proyectos de Mozilla Research con Rust, uno de ellos es Servo que está construido desde cero con el lenguaje de programación Rust, es un motor de renderizado súper rápido. Con la promesa de ser más seguro, modular, con mejor rendimiento y capaz de ejecutar varios componentes



al mismo tiempo. Esto será clave para convertirlo en una alternativa súper rápida al resto de navegadores.

Rust fue desarrollado de forma totalmente abierta, donde se solicita a la comunidad contribuir en su desarrollo. Aunque es patrocinado por Mozilla y Samsung es un proyecto comunitario.

El propósito más fuerte en que se centra el lenguaje de programación Rust es en el rendimiento, la paralelización y la seguridad de la memoria.

Rust no es simplemente un lenguaje de programación – es también una comunidad.

REFERENCES

- [1] Rust Documentation,
<https://www.rust-lang.org/en-US/documentation.html>
- [2] El Lenguaje de Programación Rust,
<https://goyox86.github.io/elpr/README.html>
- [3] Roman, Andres Rust, el nuevo lenguaje de programación de Mozilla, 2010,
<http://www.abc.es/20101201/tecnologia/rww-rust.html>