

---

---

# Compiladores

- ¿Porqué hacen falta los nuevos lenguajes de programación? -

---

---

3CM7

Vargas Romero Erick Efraín  
Prof. Tecla Parra Roberto

Instituto Politécnico Nacional  
Escuela Superior de Cómputo  
Juan de Dios Bátiz, nueva industrial Vallejo  
07738 ciudad de México



# Chapter 1

## ¿Porqué hacen falta los nuevos lenguajes de programación?

### 1.1 Introducción

Los lenguajes humanos luchan por sobrevivir en un mundo cada vez más diverso, donde algunas lenguas mueren, mientras otras, las utilizadas para la programación, están más vivas que nunca. Constantemente nacen nuevos lenguajes y parece que cada gigante tecnológico quiere tener el suyo propio. Go, el lenguaje de programación de Google, apareció en 2009; Facebook presentó el suyo en marzo de este año bajo el nombre de Hack; y Apple dio a conocer Swift en junio. No hace muchos años, los usuarios esperaban pacientemente ante una pantalla en blanco el paso de una información o funcionalidad a otra. Ahora, los consumidores se han vuelto mucho más exigentes y los lenguajes de programación han respondido con novedades para ofrecer experiencias mucho más ágiles en los dispositivos móviles. Según el informe Sociedad de la Información, que presenta cada año la Fundación Telefónica, el móvil superó por primera vez en 2016 al ordenador como principal dispositivo por el que los españoles se conectan a la red. Los dos principales sistemas operativos principales, Android e iOS, han desarrollado sus propios lenguajes, para ofrecer a sus usuarios experiencias rápidas, visuales y muy sencillas.

### 1.2 ¿Perjudica o beneficia a los programadores?

Si la tendencia se mantiene, los desarrolladores tendrán que determinar si realmente les perjudica o beneficia, aunque de momento, hay opiniones para todos los gustos. Ángel Castillo, desarrollador e ingeniero de Android, explica que todos los lenguajes de programación tienen elementos en común y una misma estructura básica. A pesar de ello, cada código tiene sus especificidades, herramientas y reglas, que hacen “muy tedioso estar programando para varios lenguajes y para diferentes sistemas operativos”. Para Miguel Ángel López, desarrollador web y Android, esto no es un inconveniente. Considera que al dominar un lenguaje de programación, pasar a otro es algo relativamente sencillo. Esto es así no solo porque hay piezas en común, sino también porque algunos derivan de otros. ¿Un ejemplo? Java y C# provienen ambos de C++ que a su vez sale de C. “Esto permite que cualquiera que conozca uno de ellos pueda entender en gran medida el resto con un solo vistazo, aunque algunos aspectos tenga que deducirlos o no pueda escribirlos correctamente”.

## 1.3 Un lenguaje para cada empresa

En principio, para una empresa no sería necesario tener un lenguaje de programación propio, pues es costumbre entre los desarrolladores aprender varios (cuántos más, mejor). No obstante, llega un momento en que las compañías son “tan grandes, importantes o prestigiosas” que, según Castillo, tienen capacidad para forzar a la gente a utilizar su propio código.

### 1.3.1 Kotlin, el rey del móvil

Se calcula que 9 de cada 10 usuarios de ‘smartphones’ utilizan el sistema operativo Android. Esto ha convertido, con el permiso de Java, a Kotlin en el nuevo rey de la programación de las ‘apps’. Un lenguaje de programación moderno, robusto, simple y potente que, tras el apoyo público de Google en la primavera de 2017, se convirtió en un lenguaje oficial de Android, al mismo nivel que Java. Al igual que ocurre con Android, Kotlin, es un proyecto de código abierto bajo Apache 2.0. El espaldarazo por parte del gigante norteamericano permitirá ir olvidando gradualmente las limitaciones del uso obligatorio de Java en Android, aunque con una ventaja clara: ambos lenguajes podrán convivir sin perder el soporte en Android. Este nuevo lenguaje, desarrollado por la empresa IntelliJ de Jet Brains, reduce el código a utilizar, lo que se traduce en menos errores y un código más fácil de entender.

### 1.3.2 Swift, la última incorporación de Apple

Swift es, al igual que Kotlin, un lenguaje relativamente reciente. La compañía de la manzana lo presentó en el World Wide Developers Conference del año 2014 y ha ido creando versiones posteriores. BBVA utiliza ya la versión 4 de Swift para el desarrollo de su ‘app’ corporativa iPhone sobre el sistema operativo iOS. Swift está creado por el propio Apple y permite diseñar ‘apps’ para iOS, Mac, Apple TV y Apple Watch. Sus creadores destacan de este código que es eficaz y rápido, 2,6 veces más que Objective-C y, a la vez, se integra a la perfección con este lenguaje. Con respecto a su predecesor, Objective-C, Swift es más seguro y, además, incluye importantes mejoras en el rendimiento. Una gran novedad es que Swift es una solución de código abierto, liberada por Apple bajo licencia Apache 2.0 en diciembre de 2015, por lo que cualquier desarrollador puede contribuir a su código fuente.

### 1.3.3 Experiencias más nativas en la web móvil

Para el desarrollo de su web móvil, BBVA ha decidido implementarla como una Single Page Application (SPA) con HTML5, CSS y Javascript. La gran ventaja de una SPA, un tipo de aplicación web donde todas las pantallas las muestra en la misma página (sin recargar el navegador), es que las comunicaciones entre cliente y servidor son muy ágiles, con el propósito de dar una experiencia más fluida a los usuarios. SPA se caracteriza por tener un solo punto de entrada -una página-, con varias vistas. Es decir, en la misma página se intercambian las diferentes pantallas de las navegaciones del usuario, lo que produce el efecto de que se encuentra ante varias páginas, aunque eso no suceda realmente.

## 1.4 En busca de un solo lenguaje de programación

Sea como sea, hay a quien tanto lenguaje de programación le causa dolor de cabeza, por eso la NSA ya está trabajando –junto con la Universidad Carnegie Mellon– para adoptar uno políglota, el Wyvern. La idea es que los programadores de aplicaciones no tengan que depender de tantos lenguajes ni aprender a utilizar numerosas herramientas. Está especialmente pensado para aplicaciones web que, según los investigadores, hoy en día se escriben como una mezcla mal organizada de diferentes lenguajes y formatos. Que un proyecto como este se haga realidad puede ser el sueño de muchos programadores (o no, porque es obra de la NSA y eso, por razones obvias, levanta suspicacias).

## 1.5 Lenguajes concurrentes, paralelos y distribuidos

La necesidad de ofrecer concurrencia en el acceso a los recursos computacionales se remonta a los primeros sistemas operativos. Mientras que un programa realizaba una operación de entrada o salida otro podría gozar del tiempo del procesador para sumar dos números, por ejemplo. Aprovechar al máximo los recursos computacionales fue una necesidad apremiante, sobre todo en la época en que las computadoras eran caras y escasas; el sistema operativo tenía que ofrecer la ejecución concurrente y segura de programas de varios usuarios, que desde distintas terminales utilizaban un solo procesador, y así surgió la necesidad de introducir algunos conceptos de programación concurrente para programar los sistemas operativos.

## 1.6 Referencias

- Varela, Mar. "Los nuevos lenguajes de programación optimizan la experiencia del usuario en el móvil". Marzo, 2018. Disponible en línea en: <https://www.bbva.com/es/nuevos-lenguajes-programacion-servicio-experiencia-usuario-movil>
- Oktaba, Hannah. "Lenguajes de programación. ¿Por qué hay tantos y aparecen nuevos?". Febrero, 2015. Disponible en línea en: [https://www.researchgate.net/publication/267253724\\_LENGUAJES\\_DE\\_PROGRAMACION\\_POR\\_QUE\\_HAY\\_TANTOS\\_Y\\_APARECEN\\_NUEVOS](https://www.researchgate.net/publication/267253724_LENGUAJES_DE_PROGRAMACION_POR_QUE_HAY_TANTOS_Y_APARECEN_NUEVOS)
- El Asri, Lucia. "La pesadilla políglota del desarrollador: ¿hay demasiados lenguajes de programación?". Diciembre, 2014. Disponible en línea en: <https://blogthinkbig.com/hay-demasiados-lenguajes-de-programacion>