



UNIVERSIDAD DE CÓRDOBA

ESCUELA POLITÉCNICA
SUPERIOR DE CÓRDOBA
Universidad de Córdoba



Métodos Formales en Ingeniería del Software

Práctica 1

Infer

Juan José Méndez Torrero
i42metoj@uco.es

Universidad de Córdoba

25 de febrero de 2019

Resumen

In the present document we will talk about a static code analyzer called Infer. We will say which is its license and some features that the analyzer has. Furthermore, we will talk about its usability and how it can be incorporated into another IDEs.

Índice

1. Introducción	3
2. Tipo de licencia	3
3. Usabilidad	3
4. Eficiencia	4
5. Extensibilidad	4
6. Técnica de análisis usado	4
7. Integraciones en IDEs	5

1. Introducción

En el presente documento vamos a hablar sobre el analizador estático de programas llamado *Infer*[1]. Este analizador está integrado dentro de la plataforma *Facebook*, y se encarga de comprobar que todas las plataformas de *Facebook* funcionen correctamente, es decir, Instagram, Facebook App, Messenger, etc. Este analizador está diseñado para los lenguajes Java, C, C++ y Objective-C.

En este documento hablaremos sobre los siguientes temas en relación a este analizador estático:

- Tipo de licencia.
- Usabilidad.
- Eficiencia.
- Extensibilidad.
- Técnica de análisis usado.
- Integraciones en IDEs.

2. Tipo de licencia

Infer es un analizador estático Open Source, es decir, que cualquier persona puede descargarlo de manera sencilla en su página web[2]. Además, han creado una página web en la que te permiten probar el analizador sin necesidad de instalarlo, pudiendo añadir cualquier archivo que esté en uno de los lenguajes comentados en la *Introducción*.

3. Usabilidad

Con respecto a la usabilidad de este programa, como hemos comentado en el apartado anterior, es muy amplia, es decir, que está a manos de todo el mundo, ya sea a través de su ordenador o incluso a través de su página web.

Para su instalación, lo más recomendable sería instalarlo en un sistema basado en UNIX, ya que, el programa *Homebrew*, nos permite instalar de manera muy sencilla este analizador, solamente poniendo en la terminal el siguiente comando:

```
brew install infer
```

Para analizar algún programa, basta con, a la hora de compilar el programa, poner delante del comando para compilar *infer* junto con las opciones adicionales del programa. Un ejemplo sería el siguiente:

```
infer analyze --[options] javac Holamundo.java
```

4. Eficiencia

En este apartado hablaremos sobre la eficiencia del analizador con respecto al tiempo y el consumo computacional. Para hacerse una idea de lo eficaz que es, tenemos que saber que este analizador no ejecuta el código, sino que lo analiza conforme se está compilando, ahorrando así mucho tiempo y consumo computacional.

Como ejemplo de su eficacia, en su página web, tienen colgado, como ya hemos comentado, un analizador on-line, siendo esto un claro ejemplo de lo ligero que debe de ser un analizador, ya que el objetivo es gastar el menor tiempo posible pero consiguiendo los mejores resultados.

5. Extensibilidad

Con respecto a la extensibilidad de este analizador estático, hay que decir que trabaja sobre muchos de los Sistemas Operativos existentes, con lo que podríamos decir que este analizador es muy extensible.

Como nota, es cierto que se puede utilizar en muchos Sistemas Operativos distintos, aunque hay que comentar que la instalación en cada uno se diferencian bastante, llegando a ser más complicada, por ejemplo, la instalación en Windows que en Mac OS.

6. Técnica de análisis usado

En esta sección hablaremos sobre cómo funciona **Infer**. Para ello, vamos a utilizar una imagen recogida de una conferencia de DevTools[3] donde explica cómo está configurado y cómo funciona su analizador. En la Figura 1 podremos observar cómo funciona **Infer**.

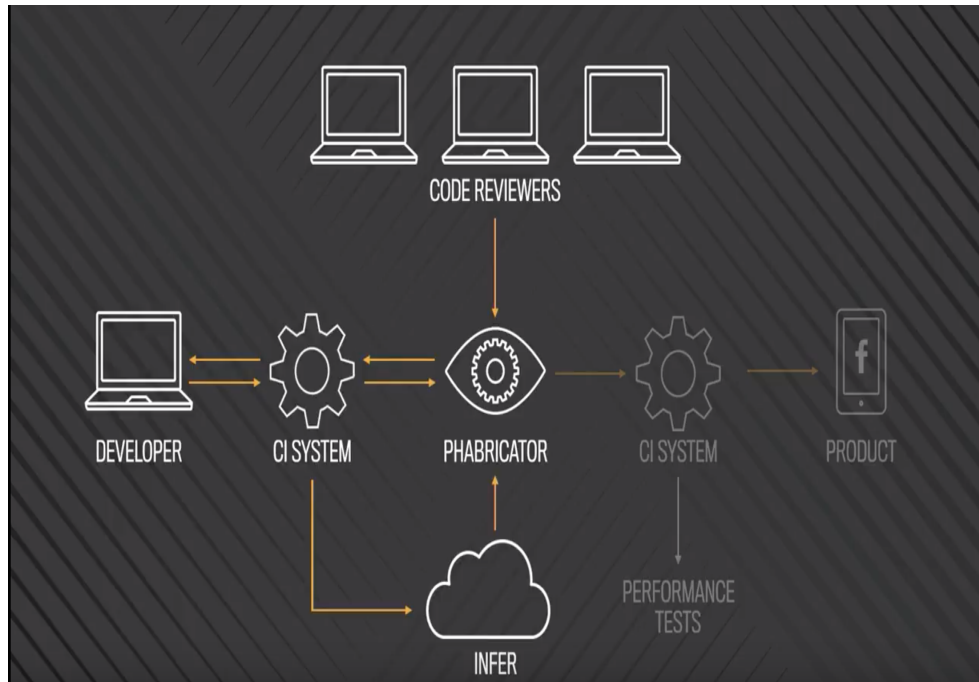


Figura 1: Método de trabajo de Infer

Como vemos, el desarrollador escribe el código, y al mismo tiempo se lo está pasando al fabricante y a **Infer**. El primero se lo envía a los inspectores de código y éstos le devuelven la información al fabricante, y al mismo tiempo, **Infer** le envía información sobre posibles fallos al fabricante. Una vez que todo está correcto, el fabricante sube el código a su aplicación una vez éste ha pasado el *Performance test*.

7. Integraciones en IDEs

Infer, al ser un analizador que trabaja ya dentro de *Facebook*, hace que la integración en otros IDEs sea una tarea más sencilla para el desarrollador. Como ejemplo, el IDE llamado Visual Studio, permite instalar el analizador como una extensión del mismo de manera sencilla. Además, al poder trabajar con los lenguajes Java, C, C++ y Objective-C, tiene una gran comunidad detrás que se encarga de mejorar este analizador para poder hacer más extensible junto con más eficaz.

Referencias

- [1] [Infer Main Page](#)
- [2] [Infer Browse App](#)
- [3] [Youtube Conference](#)