

Resumen

DevOps (*Development & Operations*) es un nuevo movimiento que fomenta la colaboración entre los desarrolladores y el personal de operaciones a través de un conjunto de principios, prácticas y herramientas para optimizar el tiempo de entrega del software. En particular, la práctica del *despliegue continuo* de software es una gran fuente de problemas y genera mucha atención cuando los artefactos de software se entregan tarde o cuando un defecto crítico llega a producción, y los clientes lo notan. Al mismo tiempo, la práctica del despliegue continuo es la frontera entre los desarrolladores y el personal de operaciones en el ciclo de entrega del software. En consecuencia, se recomienda iniciar la implementación de DevOps con la práctica del despliegue de software. Para enfrentar este desafío, los profesionales e investigadores están utilizando la Infraestructura como Código (*Infrastructure as Code*, IaC), que es un enfoque para la automatización de la infraestructura basado en prácticas de desarrollo de software. El objetivo de IaC es definir en un script todas las instrucciones para crear, actualizar y ejecutar recursos de infraestructura. En este escenario, la automatización del aprovisionamiento de la infraestructura acelera la práctica del despliegue continuo en el ciclo de entrega del software.

La computación en la nube se ha convertido en el principal modelo de pago por uso utilizado por profesionales e investigadores para conseguir servicios en la nube en un corto período de tiempo. En este escenario, las compañías están pasando de generar potencia informática *in-house* hacia la obtención de recursos informáticos provistos en la nube a través de internet como servicios web. Al mismo tiempo, IaC y la computación en la nube están promoviendo algunos cambios en la industria. Por ejemplo, los equipos de operaciones pasan todo su tiempo trabajando en software, en lugar de configurar servidores y conectar cables de red. Existe una variedad de herramientas IaC que utilizan scripts para definir y ejecutar los recursos de infraestructura en diferentes proveedores de servicios IaaS (*Infrastructure as a Service*). Sin embargo, la diversidad de lenguajes de scripting de las herramientas IaC junto con la heterogeneidad del tipo de infraestructura que ofrece cada proveedor de servicios IaaS, han ocasionado que utilizar scripts IaC para el aprovisionamiento de infraestructura sea una actividad lenta y propensa a errores.

El objetivo del proyecto de doctorado es proponer una solución a la diversidad de los lenguajes de scripting —de las herramientas IaC— y a la heterogeneidad

del tipo de infraestructura que ofrece cada proveedor de servicios IaaS respecto al aprovisionamiento de infraestructura en la nube. Para afrontar estos desafíos se propone MoCIP (*A Model-Driven Approach to Cloud Infrastructure Provisioning*).

MoCIP es un enfoque dirigido por modelos para el aprovisionamiento de la infraestructura en la nube que soporta IaC mediante la Ingeniería de Software Dirigida por Modelos (*Model-Driven Engineering*, MDE). MoCIP utiliza los dos principios fundamentales de MDE: *abstracción* y *automatización*. En primer lugar, se desarrolló el lenguaje específico de dominio ArgonML para *abstraer* las capacidades de la computación en la nube, tales como cómputo, elasticidad, almacenamiento y redes. El lenguaje ArgonML permite modelar los recursos de infraestructura de la nube. En segundo lugar, se desarrolló la herramienta ARGON para *automatizar* el aprovisionamiento de infraestructura en la nube. ARGON realiza transformaciones de modelo a modelo para generar modelos que representan la infraestructura de diferentes proveedores de servicios IaaS. Además, ARGON realiza transformaciones de modelo a texto para generar scripts IaC con la información subyacente de cada proveedor de servicios IaaS y de cada herramienta de aprovisionamiento de infraestructura.

Con el fin de investigar la interacción de MoCIP en un contexto de aplicación se diseñaron y ejecutaron un conjunto de experimentos controlados. En primer lugar, se evaluó la percepción de los usuarios al utilizar MoCIP. Se encontró evidencia empírica para afirmar que los participantes percibieron que MoCIP es fácil de usar y útil para realizar el aprovisionamiento de infraestructura en Amazon Web Services. Además, los participantes tienen la intención de utilizar MoCIP en el futuro. En segundo lugar, se evaluó el rendimiento y la percepción de los usuarios al comparar la herramienta ARGON versus la herramienta Ansible respecto a la definición de la infraestructura de Amazon Web Services. Por un lado, se encontró evidencia empírica para afirmar que ARGON es más efectivo y eficiente que Ansible para especificar los recursos de infraestructura. Por otro lado, se encontró evidencia empírica para afirmar que los participantes percibieron que ARGON es más fácil de usar y útil que Ansible para definir la infraestructura. Además, los participantes tienen la intención de usar ARGON en el futuro.