

Modelo de costes para el despliegue de pruebas E2E en entornos Cloud

Cristian Augusto ^[0000-0001-6140-1375], Jesús Morán ^[0000-0002-7544-3901], Claudio de la Riva ^[0000-0001-5592-9683], Javier Tuya ^[0000-0002-1091-934X]

Computer Science Department, University of Oviedo, Gijón,

{augustocristian, moranjesus, claudio, tuya}@uniovi.es

Resumen. La ejecución de pruebas de sistema puede requerir grandes cantidades de recursos físico-lógicos al necesitar todo el sistema para su ejecución. Ejecutarlas en la infraestructura potencialmente limitada y escalable del Cloud, permite obtener un mejor coste, eligiendo entre las distintas infraestructuras disponibles. No obstante, es habitual que la ejecución de las pruebas no use totalmente los objetos contratados, por lo que no todo el coste es usado en la ejecución de las pruebas. En este trabajo se detalla un modelo de coste de las pruebas en el Cloud que tiene en cuenta los recursos que requieren las pruebas de sistema. El modelo divide el coste de la infraestructura (coste contratado) en el coste del plan de ejecución (imputado a los recursos empleados en las pruebas) y sobresuscripción (coste de la infraestructura Cloud no utilizada). Con el modelo se pueden comparar diferentes combinaciones de objetos en el Cloud, y elegir la más efectiva en términos de estos tres costes.

Palabras clave: Pruebas de sistema, Cloud Computing, Modelo de Costes

1 Introducción

Las pruebas de sistema, también denominadas *End-to-End* (E2E), se caracterizan por probar el sistema desde el inicio hasta el final, incluyendo la interacción con el usuario, y suelen ser costosas al requerir el sistema completo (ej. una aplicación web completa con varios balanceadores de carga). Desplegar las pruebas E2E en el Cloud, permite beneficiarse de su flexibilidad y escalabilidad [1] pudiendo obtener un mejor coste. No obstante, presenta desafíos como seleccionar la combinación de infraestructura más adecuada en términos de coste entre el gran abanico disponible [2]. Elegir una combinación de infraestructura inadecuada supone que no sea utilizada totalmente por los recursos de las pruebas, lo que deriva en qué parte del coste no se use en estas [3].

En este artículo, se presenta un modelo de coste en el que se tiene en cuenta el coste de los recursos empleados en las pruebas E2E (coste del plan de ejecución) y el asociado a la infraestructura no usada (coste de sobresuscripción). El modelo parte de una planificación de la ejecución de los casos de prueba junto con sus recursos, para el que se valoran varias combinaciones de infraestructura. Las combinaciones no solo se comparan en términos del coste de la infraestructura (contratado) sino que la elección tiene

en cuenta el coste del plan de ejecución y de sobresuscripción, eligiendo aquella combinación más efectiva en términos de costes.

2 Background

En anteriores trabajos [4], [5], los autores presentaron RETORCH: **Resource aware End to end Test ORCH**estration, una técnica de orquestación de casos de prueba centrada en los recursos. En RETORCH los **recursos** son las entidades físicas, lógicas o computacionales requeridas por los casos de prueba durante la ejecución (ej. una base de datos, un navegador, un servidor web, entre otros) de los cuales pueden crearse instancias: entidades de recurso independientes usadas por los casos de prueba. Partiendo de los casos de prueba, se realiza una caracterización de los **recursos** empleados por los casos de prueba junto con sus **atributos** (posibilidad de compartición y tipo de acceso realizado, cantidad que se pueden instanciar, etc.). Esta información es anotada junto a los casos de prueba, y se emplea para agruparlos en *TJobs*, conjuntos de casos de prueba con un uso de recursos compatible (ej. solo leen o modifican partes del recurso que el otro caso no usa) que son desplegados junto con sus recursos de forma contenerizada. Los *TJobs* se ordenan dando lugar a un **plan de ejecución** que tiene como objetivo reducir tiempo y la cantidad de recursos necesarios en la ejecución de las pruebas, desplegando dicho plan en una infraestructura de integración continua (CI).

3 Modelo de costes

La ejecución y despliegue de los *TJobs* en el *Cloud* requiere de la contratación de objetos que suponen un coste (**coste de infraestructura**). Este **coste de infraestructura** no tiene en cuenta qué parte del coste se está empleando directamente en las pruebas y qué parte es asociado a la infraestructura no usada. Con objeto de calcular que parte del coste se emplea en las pruebas, se propone un modelo de costes que representa la configuración del *Cloud* con las siguientes entidades:

- Los **Objetos Cloud** son las diferentes instancias empleadas para desplegar las instancias de recursos en el *Cloud*, se contratan por un tiempo y con unas capacidades (i.e. memoria, procesadores, almacenamiento, entre otros). Consideramos Objetos Cloud a las máquinas virtuales, los contenedores y los servicios.
- Las **Opciones de Facturación** son las formas de tarificación de las distintas instancias de Objeto Cloud ofrecidas por los proveedores (ej. Microsoft, Google, Amazon). Las opciones contemplan un coste, periodo de facturación (por cuánto se contrata) y capacidades. Consideramos Opciones de Facturación el pago por uso (*pay-as-you-go*), adelantado (*pre-invoiced*) y subasta (*auctions*).

Las instancias de recurso pueden desplegarse en un abanico de Objetos y Opciones de Facturación en el *Cloud*. Equivocarse en la elección, deriva en que las instancias de recurso no usen completamente los Objetos, por lo que el coste que se paga (**coste de la infraestructura**) se divide en:

- El **coste del plan de ejecución** (usado) es el coste de las instancias de Objeto Cloud que se imputa a las instancias de recurso acorde al uso de *capacidades* y *tiempo* que hacen durante las pruebas. Este nos permite conocer cuánto costará una determinada combinación de Objetos Cloud y Opciones de Facturación.
- El **coste de sobresuscripción** (no usado) es la diferencia entre el coste de la infraestructura y del plan de ejecución. Permite conocer el coste de las capacidades/tiempo de instancia de Objeto Cloud que no son utilizadas durante las pruebas.

Ejemplo Práctico

Para ilustrar los costes, la Figura 1 muestra un el perfil de uso de recursos (BBDD y navegadores desplegados en servicios como Objeto Cloud) que muestra las capacidades contratadas (verde) y las usadas (rojo) por las instancias de **recurso**. La parte derecha de la Figura 1 muestra la estimación del acumulado de costes (contratado vs. usado) de dos combinaciones con diferente servicio para desplegar la BBDD (**I: A-C** y **II: B-C**):

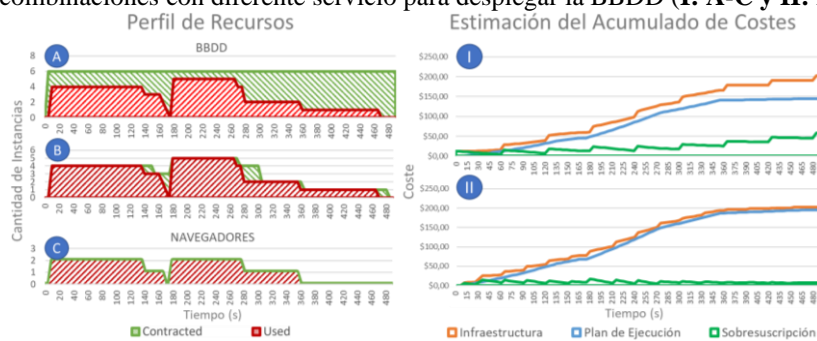


Figura 1 Perfil de uso de recursos y estimación del acumulado de costes

Los servicios **A**, **B** y **C** tienen diferentes **Opciones de Facturación** cuya *capacidad* es el número de instancias de recurso desplegables. El servicio **A** tiene un *periodo de facturación* de 1 minuto, el **B** 30 segundos y el **C** 1 segundo. El *coste* del servicio **A** es de 12 \$/6 instancias-BBDD, **B** 2 \$/instancia-BBDD y **C** 0,005 \$/instancia-navegador.

La parte derecha de la Figura 1 (**I-II**) muestra la estimación del acumulado de costes para una ejecución del plan en cada combinación. El **coste de infraestructura** (línea color naranja) es el resultado de sumar los costes de las instancias de Objeto Cloud contratadas en cada combinación (**A-C** y **B-C**). El **coste del plan de ejecución** corresponde con la línea azul, es el coste imputado a todas las instancias de recurso requeridas por las pruebas. El coste de cada instancia de recurso requerida por los *TJobs*, se calcula multiplicando el coste de la instancia de Objeto Cloud por la proporción de *capacidades/tiempo* usadas por el recurso (**utilización**). El **coste de sobresuscripción** en color verde se calcula con la diferencia de los otros dos costes.

En este ejemplo el **coste de infraestructura** es similar (108 \$) por lo que se elige en términos de los otros costes: **I** tiene un menor **coste del plan de ejecución**, pero mayor coste de **sobresuscripción**, ya que la **Opción de Facturación** del servicio **A** requiere contratar 6 instancias independientemente de las que se utilicen. La **combinación II**, tiene un menor **coste de sobresuscripción** gracias a su *periodo* (30 s) aunque, su **coste del plan de ejecución** es mayor debido al alto *coste* del servicio **B** (2 \$ instancia/30 s).

Basándose en los tres costes, se puede concluir que la mejor alternativa es la **combinación I**, ya que provee de un menor coste de las pruebas, además de Objetos que podrían emplearse para otras tareas como, por ejemplo, si se añadiesen nuevas pruebas.

4 Conclusiones

El modelo de coste de las pruebas en el *Cloud* permite diferenciar qué parte de coste es empleado en las pruebas y sobresuscripción. Ambos permiten comparar diferentes combinaciones de Objetos, atendiendo no solo al **coste de infraestructura**, dando soporte para la elección de una infraestructura *Cloud* más efectiva en términos de coste.

Como líneas de trabajo futuro se pretende aplicar el modelo en conjuntos de casos de prueba de aplicaciones reales y estudiar cómo incorporar el coste del plan de ejecución como un factor más a optimizar durante la orquestación de las pruebas [6].

Agradecimientos

A la Dra. Antonia Bertolino (ISTI-CNR, Pisa Italia) por sus contribuciones en la línea de investigación. Este trabajo ha sido financiado por MCIN/AEI/10.13039/501100011033, proyectos PID2019-105455GB-C32 y RED2018-102472-T.

Referencias

- [1] A. Bertolino *et al.*, “A systematic review on cloud testing,” *ACM Computing Surveys*, vol. 52, no. 5, pp. 1–42, Sep. 2019, doi: 10.1145/3331447.
- [2] L. Sun, H. Dong, F. K. Hussain, O. K. Hussain, and E. Chang, “Cloud service selection: State-of-the-art and future research directions,” *Journal of Network and Computer Applications*, vol. 45, pp. 134–150, Oct. 2014, doi: 10.1016/j.jnca.2014.07.019.
- [3] K. Inçki, I. Ari, and H. Sözer, “A survey of software testing in the cloud,” *Proceedings of the 2012 IEEE 6th Int. Conf. on Software Security and Reliability Companion, SERE-C 2012*, pp. 18–23, 2012, doi: 10.1109/SERE-C.2012.32.
- [4] C. Augusto, J. Morán, A. Bertolino, C. de la Riva, and J. Tuya, “RETORCH: Resource-Aware End-to-End Test Orchestration,” in *Communications in Computer and Information Science*, Sep. 2019, vol. 1010, pp. 297–310. doi: 10.1007/978-3-030-29238-6_22.
- [5] C. Augusto, J. Morán, A. Bertolino, C. de la Riva, and J. Tuya, “RETORCH: an approach for resource-aware orchestration of end-to-end test cases,” *Software Quality Journal*, no. SQJO-D-19-00197R1, 2020, doi: 10.1007/s11219-020-09505-2.
- [6] C. Augusto, “Optimización de Recursos en Pruebas de Sistema,” Univeristy of Oviedo, Gijón (Asturias), Spain, 2020. [Online]. Available: <http://digibuo.un-ovi.es/dspace/handle/10651/56205>