Instituto Tecnológico de Costa Rica

Escuela de Ingeniería en Computación

Programa de Maestría en Computación

Modelado y simulación de *middleware* orientado a mensajes

Propuesta de Tesis sometida a consideración del Departamento de Computación, para optar por el grado de Magíster Scientiae en Computación, con énfasis en Ciencias de la Computación

Estudiante

Carlos Martín Flores González

Profesor Asesor

Ignacio Trejos Zelaya

Noviembre, 2018

Indice

Índice de figuras

Índice de cuadros

Capítulo 1

Introducción

Los métodos de predicción de rendimiento basados en modelos permiten a los arquitectos de software evaluar el rendimiento de los sistemas de software durante las primeras etapas de desarrollo. Estos modelos de predicción se centran en los aspectos relevantes de la arquitectura y de la lógica del negocio, dejando de lado detalles de la infraestructura subyacente. Sin embargo, estos detalles son esenciales para generar predicciones de rendimiento que sean precisas.

Para los ingenieros, es una práctica común simular el modelo de un artefacto antes de construirlo. Modelos de diseños de autos, circuitos electrónicos, puentes, entre otros, son simulados para entender el impacto de decisiones de diseño en varias atributos de calidad de interés como seguridad, consumo de energía o estabilidad. La habilidad de predecir las propiedades de un artefacto basado en su diseño y sin necesidad de construirlo, es una de las características centrales de una disciplina de ingeniería. A partir de esta visión, de lo que se considera una disciplina de ingeniería establecida, se podría decir entonces que la ingeniería de software es apenas una disciplina de ingeniería[1]. Esto porque frecuentemente los ingenieros de software carecen del entendimiento del im-

pacto de decisiones de diseño en atributos de calidad como rendimiento o confiabilidad. Como resultado, se intenta probar la calidad del software mediante costosos ciclos de prueba y error.

El no entender el impacto en las decisiones de diseño puede ser costoso y riesgoso. El probar software significa que ya se ha hecho un esfuerzo en su implementación. Por ejemplo, si las pruebas revelan problemas de rendimiento, es muy probable que la arquitectura necesita ser modificada, lo que puede conllevar a costos adicionales. Estos costos surgen debido a que en sistemas de software empresarial un bajo rendimiento es principalmente el efecto de una arquitectura inadecuada que efecto de código.

La ingeniería de rendimiento de software(SPE por sus siglas en inglés) es una disciplina que se centra en incorporar aspectos de rendimiento dentro del proceso de desarrollo de software, con el objetivo de entregar software confiable de acuerdo con propiedades de rendimiento particulares. Los modelos de rendimiento predictivos son una de las herramientas empleadas en SPE. Construidos en las fases tempranas del proceso de desarrollo de software, los modelos ayudan a predecir el rendimiento eventual del software y de esta forma guiar el desarrollo, para eso los modelos de predicción de rendimiento deben capturar todos los componentes relevantes del sistema.

Para aplicaciones de software modernas, esto puede implicar modelar complejas capas tales como máquinas virtuales o *middleware* de mensajería. Componer todos estos modelos puede resultar una tarea costosa e ineficiente. En su lugar, un modelo abstracto de la aplicación se puede construir primero y luego ir agregando los modelos de los componentes del sistema.

En este trabajo se propone la construcción de un modelo de rendimiento para un sistema que utilice *middleware* orientado a mensajes con el fin de evaluar la influencia en el rendimiento de dicho sistema. Se propone evaluar esta

influencia por medio de un ejemplo: tomar una aplicación de referencia con el fin de obtener sus métricas actuales de rendimiento, adaptarla para que utilice *middleware* orientado a mensajes y luego medir su rendimiento y generar un modelo a partir de esto.

Capítulo 2

Antecedentes

Capítulo 3

Definición del Problema

En los sistemas de software en los que se utiliza comunicación basada en mensajes, el rendimiento depende en gran medida del *middleware* orientado a mensajes (*Message Oriented Middleware* - MOM). Los arquitectos de software necesitan considerar su configuración y uso para obtener predicciones significativas sobre el comportamiento de la aplicación. Sin embargo, la inclusión de un MOM en un modelo de arquitectura de software requiere un esfuerzo adicional así también como de conocimiento detallado de la infraestructura utilizada. Los arquitectos podrían llegar a omitir la influencia del MOM y esto tendría como consecuencia la generación predicciones erróneas.

Las decisiones que han sido tomadas con poca información durante las etapas de diseño usualmente son muy difíciles de alterar y podrían hacer imposible el lograr los niveles requeridos de rendimiento de un sistema una vez que este haya sido puesto en producción. Es por esto que los arquitectos y diseñadores necesitan tener la habilidad de predecir el rendimiento del MOM utilizado, trabajando a partir de diseños abstractos sin tener acceso a la implemetación completa de la aplicación. Las aplicaciones que utilizan protocolos de mensajería proporcionados por un MOM realizan comunicación asíncrona y basada en

colas. Estas aplicaciones deben también hacer frente a consideraciones de arquitectura adicionales como la persistencia de los mensajes y flujos de control. Todos estos factores pueden llegar a influenciar en gran media el rendimiento de las aplicaciones.

Capítulo 4

Justificación

4.1. Innovación

4.2. Impacto

4.3. Profundidad

Capítulo 5

Objetivos

5.1. Objetivo General

Diseñar un método para modelar sistemas distribuidos de intercambio de mensajes (*message-oriented middleware*) para evaluar la influencia que tienen en el rendimiento de un sistema de software.

5.2. Objetivos Específicos

1. Revisión del estado del arte de trabajos relacionados con enfoques de predicción y medición del rendimiento en sistemas de software basados en componentes y en *middleware* orientado a mensajes

2. Adaptar un sistema de software de referencia para el cual ya exista un modelo de rendimiento y simulaciones para que se integre y comunique con *middleware* orientado a mensajes.

3. Comparar la solución implementada bajo diferentes cargas de trabajo

4. Crear un modelo del nuevo sistema y su rendimiento

5. Validar y analizar el modelo creado a través de experimentos

Capítulo 6

Alcance

Capítulo 7

Entregables

7.1. Revisión de literatura

Alineado con objetivo específico 1

Se pretende identificar los resultados de otros estudios relacionados con modelaje de rendimiento de software, así como retos y oportunidades de investigación que existan en esta área. Las preguntas de investigación inicialmente propuestas para conducir esta revisión son las siguientes:

PI.1 ¿Cuáles enfoques de predicción y medición del rendimiento en sistemas de software basados en componente se han propuesto?

PI.2 ¿Cuáles enfoques de predicción y medición de rendimiento de software se han utilizado para *middleware* orientado a mensajes?

PI.3 ¿Qué retos y oportunidades existen con estos enfoques en la actualidad?

PI.4 ¿Qué herramientas hay disponibles para modelaje de rendimiento de software?

7.2. Aplicación adaptada para que soporte comunicación basada en mensajes

Alineado con objetivo específico 2

Una vez que se haya identificado un sistema de software del cual ya exista un modelo de rendimiento y simulaciones, este se tomará como base para adaptarlo de tal forma que utilice *middleware* orientado a mensajes en algún punto de su ejecución.

7.3. Medición de rendimiento en aplicación adaptada

Alineado con objetivo específico 3

A la aplicación adaptada se le realizarán mediciones de su rendimiento con el fin de tener estos como entrada para las subsecuentes tareas de modelado propuestas. Serán estas mediciones las que ayudan a refinar y calibrar el modelo.

7.4. Modelo de rendimiento del nuevo sistema

Alineado con objetivo específico 4

A partir del sistema adaptado y sus mediciones, se modificará el modelo de rendimiento original del sistema de referencia para reflejar los cambios introducidos por el *middleware* orientado a mensajes.

7.5. Validación y análisis del nuevo modelo

Alineado con objetivo específico 5

El nuevo modelo se ejercitará con los perfiles de uso de la aplicación de referencia, con el fin de validar si este puede aproximar el comportamiento del sistema ahora que se le han introducido cambios.

Capítulo 8

Metodología

Capítulo 9

Cronograma de actividades

Bibliografía

[1] R. H. Reussner, S. Becker, J. Happe, R. Heinrich, A. Koziolek, H. Koziolek, M. Kramer, and K. Krogmann, *Modeling and Simulating Software Architectures: The Palladio Approach.* The MIT Press, 2016.