

Modelado y Verificación de Patrones de Diseño de Arquitectura de Software para Entornos de Computación en la Nube

Brayan Solano Fonseca¹, Esteban Morales Ureña²

I. INTRODUCCIÓN

Este trabajo propone una herramienta basada en un meta-modelo de patrones de arquitecturas de software en la nube que facilita la tarea de modelado y, además, posibilita la verificación semiautomática de los patrones aplicados a fin de garantizar su correcta instanciación. Complementariamente, se sientan las bases necesarias para la definición de un entorno de diseño integrado que incluya guías de diseño basadas en el conjunto de componentes y conexiones propuestos junto con sugerencias inteligentes derivadas de las decisiones arquitectónicas adoptadas por el usuario a lo largo de la construcción del diseño.

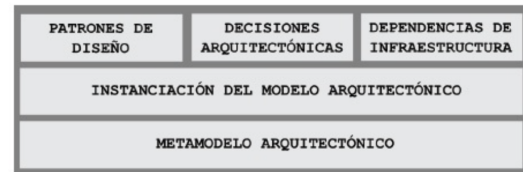
II. PATRONES DE DISEÑO ARQUITECTÓNICO

Un patrón de diseño arquitectónico define una familia de sistemas en base a un esquema de organización estructural, bajo esta perspectiva los patrones arquitectónicos pueden ser vistos como los bloques constructivos básicos de las arquitecturas de software. Las arquitecturas de sistemas de software complejos suelen basarse en la aplicación de más de un patrón de diseño. Estos patrones ayudan a determinar la forma en la cual deben ser distribuidos los distintos tipos de componentes de una aplicación web. Luego, la identificación de los patrones adecuados es una tarea importante que debe ser llevada a cabo por el arquitecto a cargo de la construcción del diseño. No todos los patrones de CC son nuevos. Muchos de los patrones existentes para las arquitecturas de software tradicionales han sido adaptados para su aplicación en contextos de aplicaciones web basadas en entornos de CC. Sin embargo, no todos los patrones de diseño existentes abstraen correctamente los elementos que conforman cada uno de los niveles arquitectónicos requeridos (esto es, software e infraestructura).

III. ENTORNO DE DISEÑO INTEGRADO PARA ARQUITECTURAS WEB

La base de trabajo consiste en un módulo conceptual denominado metamodelo arquitectónico que define el vocabulario de elementos disponibles para la creación de arquitecturas de CC. En este caso, el metamodelo arquitectónico posibilita la instanciación de modelos de arquitecturas en base a un conjunto predefinido de componentes y conectores arquitectónicos. La siguiente figura esquematiza el conjunto de módulos identificados como parte del entorno de diseño propuesto.

En base a las instanciaciones que se producen como resultado de la utilización de los componentes y conectores arquitectónicos identificados, se definen tres nuevos módulos que actúan como complementos en la especificación de arquitecturas de CC, a saber:



- **Módulo patrones de diseño:** Verifica la correcta aplicación del conjunto de patrones de diseño propuesto en (Fehling, Leymann, Retter, Schupeck Arbitter, 2014) sobre el modelo de arquitectura instanciado, dando lugar a una evaluación del diseño.
- **Módulo decisiones arquitectónicas:** Aplica un modelo de representación del conocimiento durante la evolución del diseño basado en la trazabilidad de los elementos arquitectónicos involucrados en las decisiones tomadas por el arquitecto.
- **Módulo dependencias de infraestructura:** Utiliza la información provista por los proveedores de infraestructura a fin de asistir al arquitecto en el proceso de selección y contratación de los servicios requeridos para el despliegue de los componentes de software (incluidos en el modelo) sobre el entorno de CC.

IV. CONCLUSIONES

Teniendo en cuenta que el paradigma de CC se ha convertido rápidamente en una de las estrategias de solución tecnológica más populares e influyentes del mundo actual, en este trabajo se han especificado los lineamientos requeridos para la construcción de un entorno de trabajo que facilite la tarea de diseño arquitectónico. Sobre este metamodelo, se han implementado los módulos de instanciación y verificación de patrones de diseño, tendientes a componer el entorno final. Luego, al restringir los componentes disponibles, el arquitecto debe tomar una menor cantidad de decisiones que, posteriormente, pueden ser verificadas conforme patrones de diseño de CC preestablecidos.

REFERENCIAS

- [1] M. Blas, H. Leone and S. Gonnet, "Modelado y Verificación de Patrones de Diseño de Arquitectura de Software para Entornos de Computación en la Nube", *SciELO*, 2019. <http://www.scielo.mec.pt/scielo.php?script=sciarttextpid=S1646-98952019000500002lang=es>. [Accessed: 20- Mar- 2020].