

Walrus

Manuel I. Capel
School of Computer Engineering
and Telecommunications
University of Granada
Granada-18071, Spain
<http://lsi.ugr.es/~mcapel.html>
Email: manuelcapel@ugr.es

Resumen—

I. INTRODUCCIÓN

Walrus es un servicio de almacenamiento de datos que hace uso de las tecnologías de servicios Web actuales y posee una interfaz que es compatible con “Simple Storage Service”(S3) de Amazon [1]. Walrus implementa REST¹ a través del protocolo HTTP, denominado la interfaz de “consultas”², así como todas las interfaces que son compatibles con S3.

The paper is organized as follows. In the following section the formal background behind BPMN formalization and verification method is presented. And thus, the proposal of temporal semantics for BPMN constructs is detailed together with a set of transformation rules into CSP+T. Then, we present the application of our proposal to an example of a BPM related to Customer Relationship Management (CRM) business. The last section discusses the conclusions and suggestions for future work.

II. FUNCIONALIDAD DE WALRUS

Los usuarios que tienen acceso a Eucalyptus suelen utilizar Walrus para transmitir datos hacia/desde la nube así como también desde las instancias que han iniciado en los nodos.

Walrus actúa como un servicio de almacenamiento para las imágenes³ de máquinas virtuales (VM) que se cargan en Walrus y que son accedidas desde los nodos de un *cloud*. Los usuarios pueden utilizar las herramientas de S3 de Amazon para transmitir datos desde/hacia Walrus. El sistema comparte credenciales con la base de datos de usuarios de los controladores del cloud. Walrus soporta transferencias de datos en serie y concurrentes, tal como hace S3. Walrus no proporciona construcciones de tipo cerrojo (“locks”) para realizar escrituras sobre objetos que estén sujetas a problemas de concurrencia de accesos.

A los usuarios se les garantiza que se guardará una copia consistente del objeto si existen escrituras concurrentes sobre el mismo objeto.

Una vez que una petición ha sido verificada, el usuario se ha autenticado y se ha comprobado con las listas de control de accesos asociadas a los objetos demandados, se procederá a transmitir las lecturas y las escrituras a través del protocolo HTTP.

Walrus actúa realmente como un servicio de gestión y almacenamiento de una imagen de máquina virtual, de tal forma que el sistema de ficheros raíz de la máquina virtual, el *kernel* y la imagen del *ramdisk* se empaquetan y se cargan utilizando herramientas EC2 de Amazon⁴.

Las herramientas de EC2 permiten comprimir y encriptar imágenes utilizando las credenciales de los usuarios y dividir estas imágenes en múltiples partes que son detalladas un archivo de descripción de imágenes, conocido como “*the manifest*” según la jerga introducida por EC2.

Walrus normalmente sirve las peticiones del *controlador de nodos* NC, de tal manera que antes de instanciar una imagen en un nodo, Walrus envía una petición de descarga de imagen que es autenticada utilizando un conjunto interno de credenciales, posteriormente las imágenes serán verificadas, descriptadas y finalmente transferidas.

Walrus mantiene una cache de imágenes que han sido ya descriptadas y las actualizaciones de cache se producirán cuando se sobrescriba un *manifiesto de imagen*, de acuerdo con la notación EC2.

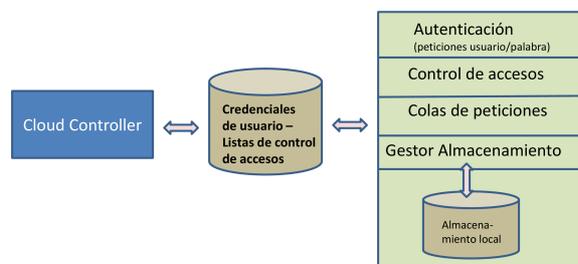


Fig. 1. Estructura de Walrus.

¹REST es un término acuñado por Roy Fielding en su tesis doctoral para describir un estilo arquitectónico de utilidad para redes. REST es un acrónimo que significa “Representational State Transfer”

²“Query Interface”

³Sistema de archivos de *root* e imágenes del *kernel* y del *ramdisk* que se utiliza para para crear instancias en los nodos de un cloud

⁴Elastic Compute Cloud (EC2) es un servicio web que proporciona capacidad informática con tamaño modificable en la nube. Está diseñado para facilitar a los desarrolladores recursos informáticos escalables basados en web

Walrus es totalmente modular y sus componentes principales son los siguientes:

- Autenticación
- Transmisión
- Almacenamiento en segundo plano⁵

III. ARQUITECTURA DE WALRUS

IV. ESTUDIO DE UN CASO

V. CONCLUSION

REFERENCES

- [1] D. Nurmi, R. Wolski, et al. *The Eucalyptus Open-source Cloud-computing System*. Proceedings of the 2009 IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGRID),124:131,IEEE Computer Society,2009.
- [2] *PMDB: An Approach to Improve the Quality of Service for Multi-tenant RDBMS in the Cloud*. Barcelona, 3–5 April, 2014.
- [3] *A Template Description Framework for Services as a Utility for Cloud Brokerage*. Barcelona, 3–5 April, 2014.
- [4] J. Spillner, J. Mueller, A. Schill. *Creating Optimal Cloud Storage Systems*.Future Generation Computer Systems,1062:1071, 2012.
- [5] Hui-Shyong Yeo and Hyotaek Lim. *Leveraging Client-side Storage Techniques for Enhanced Use of Multiple Consumer Cloud Storage Services on Resource-Constrained Mobile Devices*. Journal of Network and Computer Applications, 2014.

VI. APPENDIX

⁵backend storage