

Experimental study of performance and QoS of multicast and unicast video traffic in WAN links

Higinio Facchini¹, Santiago Pérez¹, Luis Marrone², Fabián Hidalgo¹, Adrian Cardenas¹

¹ CeReCoN – Departamento de Electrónica – Facultad Regional Mendoza – UTN
Rodríguez 273, Ciudad Mendoza
CP (M5502AJE) República Argentina
{higiniofac, santiagocp}@frm.utn.edu.ar

² LINTI – Facultad de Informática – Universidad Nacional de La Plata
50 y 120 – La Plata, Argentina
CP (B1900ASD) República Argentina

Abstract. In recent years there has been an exponential increase in the growth in multimedia applications, and in particular in video applications. In this work, a quantitative analysis is performed by experimentation, in order to evaluate the behavior and impact of video traffic on WAN networks. We propose a WAN test bed with four scenarios that allows to inject unicast or multicast video traffic, compressed with several codecs. From capturing video traffic, we identified several interesting performance and QoS metrics. We include detailed contributions on the impact produced by several factors, when there are restrictions on the bandwidth, as in a corporate real WAN link of some few Mbps. The work oriented on the configurations to be taken into account for an adequate management of similar networks and an efficient use of the resources available, without compromising the performance and the expected QoS.

Keywords: Multicast traffic, Video codec, WAN test bed M, QoS

1 Introduction

In recent years there has been an exponential increase in the growth of multimedia applications and, in particular, in video applications. Today you can receive digital TV with high definition and a greater supply of channels. There are Internet sites that offer movies on-line, and it is common for users to upload and download videos from sites web. It has also been widespread recording and share videos with mobile phones. The video call through the Internet is common. Large companies and organizations use video conferencing applications for face-to-face collaboration in different geographical regions. And even within the same company video traffic grows.

The growing number of multimedia users also involves a significant increase in bandwidth requirements. Consumers are becoming more demanding in terms of quality and the performance of the products based on video, and therefore, there is a strong incentive for continuous improvement in multimedia technologies. Understand

the structure of the frames of video and the requirements that they impose on the network, will help network administrators to improve the flow of traffic in the same. Aruba Networks [1] presents an interesting classification of the video traffic.

A number of standards have been developed for video encoding, such as H.261, H.263, MPEG1, MPEG2, MPEG4 and H.264. H.264/MPEG4 represents a great leap in video compression technology with, typically, a reduction of 50% of the average bit rate for a given video quality compared to MPEG-2, and approximately a reduction of 30% compared with MPEG-4 Part 2.

There are different jobs on the subjects considered, such as the analysis of multicast traffic and multicast routing protocols [2], analysis of video codecs [3], analysis of video traffic under different contexts [4], analysis of multicast video traffic [5], comparative analysis of multicast vs unicast [6], analysis of QoS for video traffic [7-9] and work on IPTV [10]. In this work was raised to obtain quantitative conclusions from experimental studies, on the best options of video traffic in laboratory networks WAN using actual equipment.

To do this, the main contributions of this paper are: (i) The demonstration, with detailed quantitative results, that: a) For multicast video traffic, all the metrics of performance and QoS obtained and analyzed respond to the expectations, and depend exclusively on the video codec used, and not of the topology of the network or of the number of clients receiving video streaming, and b) For the unicast video traffic, in addition, it was observed that the number of bytes transmitted and the final speed on each link if they depend on the topology of the network and the number of clients receiving video streaming; ii) The specification of a test bed WAN and four scenarios in experimental and of a comparative methodology to establish mechanisms between the different trials; y iii) The experimental determination of the behavior of each codec with regard to the statistical distributions of the interframes times.

The remainder of this paper is structured as follows: Section II provides an overview of the IP Multicast; section III presents the scenario and resources of experimentation; Section IV presents the results of the experimentation; and finally, in Section VII sets out the most important conclusions.

2 IP Multicast

For years, the suppliers of ISP services have been trying to use multicast technology to offer the service of high quality video through the interconnection of networks. They see the multicast capacity to scale easily and use the bandwidth effectively as an important tool, which could help them to control the loads of traffic, save bandwidth, offer new services, generate additional revenue and strengthen their long-term profitability.

Unfortunately, there are problems with the multicast deployment model in the interconnection of networks. In the first place, the multicast protocols need to be deployed in each router in the path between sources and receivers. Service providers have also found problems in providing reliable multicast services through the best efforts environment of the interconnected networks. Multicast solutions, in general, have lacked the bit rate required for adaptive streaming video transmission.

There is a widespread belief that you can't use the multicast technology to deliver video over interconnected networks. Certainly, the IP multicast promised to offer a number of benefits. Unlike unicast delivery, in which a server replicates a data stream for transmission to all client devices on the network that request it, the delivery of IP multicast invokes the server to transmit a single stream of data in the routing infrastructure. The routing infrastructure then replica effectively the data stream so that no router interface transport more than a single copy of the sequence, regardless of how many clients receive (Figure 1).

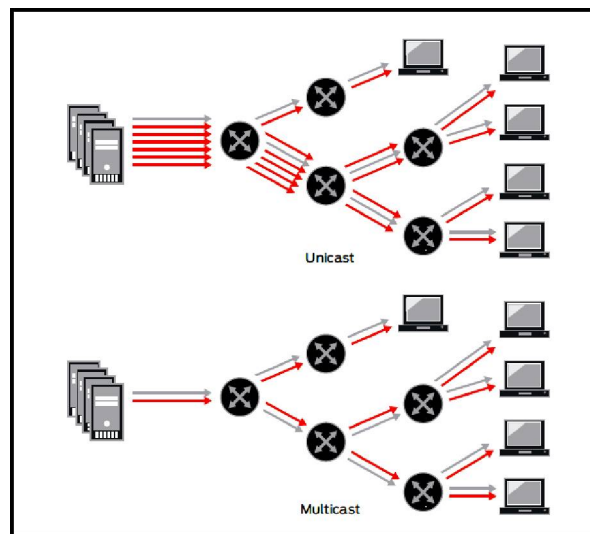


Fig. 1. Unicast and multicast

This difference is reflected in the preservation of bandwidth and data flows more efficient, which obviously adds up to a better network performance. But, in order to enjoy these benefits, you need to deploy IP multicast protocols on each router in the path. Therefore, the multicast IP address exists today mainly in LAN networks and in small areas of interconnected networks highly controlled. In fact, some companies have begun to offer customized solutions to try to resolve these limitations. The barriers that have so far prevented the widespread deployment of IP multicast are:

- Each routing device in the delivery route must support IP multicast, which means that if the IP multicast is omnipresent, each service provider must decide to offer multicast service.
- Is a great challenge to offer reliable multicast service in the best effort inherently environments that exist in the network today.
- Streaming video requires adaptive bit flow, what it lacks the multicast.
- The formats of video streaming of today, designed to use progressive download, mechanisms are not inherently capable of multicast.
- The interdomain multicast or between pairs of Internet providers ISP requires business agreements and complex network configurations.

In this experimental work has been quantified the impact and advantage on performance and QoS, of the use of the multicast video traffic, in environments of WAN interconnection, approaching the real situation of companies that link their LANs using WAN links of ISPs.

3 Scenario and experimental resources

In the present study we evaluated experimentally the performance of streaming video on a WAN test bed with the use of network computers, servers, and real customers.

The generic topology used includes a computer functioning as streaming server and up to 20 desktop computers (PCs) as clients connected at the ends of a network as if it were a WAN. This network is formed by WAN routers and switches with different types of links by linking to them. Figure 2 shows the general topology. In this topology, the links indicated with continuous line are of type FastEthernet with a transmission speed of 100 Mbps, while the links indicated by dashed line is serial interfaces connected in point-to-point format a transmission speed of 2Mbps. For the operation between routers R1 to R6 is configured the routing protocol OSPF v2. For the same routers are configured the routing protocol multicast PIM dense mode. The existing redundant links were raised as an option to simulate a WAN, but was used to configure the routing protocol for traffic between the server and each client always follow a single path.

For the topology is used the following hardware:

- Server: a desktop computer with the following features: CPU Intel Core i5 processor with 8GB of RAM and Windows 7 Professional 64 Bit.
- Clients: desktop computers with the following characteristics: AMD Athlon(tm) II X2 250 to 3GHz with 2GB of RAM and operating system Windows 7 Professional 32 Bit.
- Routers R1, R2, R3 and R4: Cisco model 2811, and Cisco routers R5 and R6 were resolved with switches Cisco WS-CS3750.
- Finally, for the connection of the routers with the PCs: Cisco switches Layer 2 Catalyst Model WS-2950-24.

And used the following software:

- Streaming server: The Unreal Media Server software, installed on the PC indicated how Video Server [11]. How to video player client software: the customer Unreal Media Player installed on all client computers. And for the capture Wireshark [12].

From the topology of experimentation, were derived 4 scenarios: is output from the server (a) is received in 3 clients (PC2-1, PC0-1 and PC3-1), (b) is received in 6 clients (PC1-1, PC2-1, PC0-1, PC0-2, PC3-1 and PC4-1), (c) is received in 9 clients (PC1-1, PC2-1, PC2-2, PC0-1, PC0-2, PC0-3, PC3-1, PC3-2 and PC4-1), and (d) is received on all clients. In all cases were injected with the video of [13].

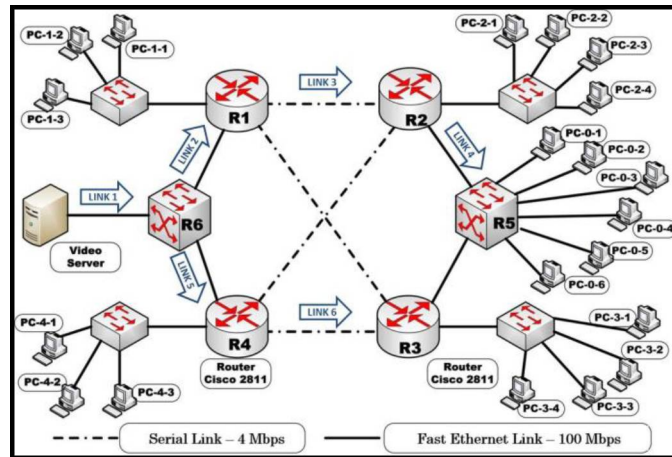


Fig. 2. Scenario of experimentation

4 Results of experimentation

4.1 Impact on bit rate of traffic

The PIM settings, routing and bandwidth based on the topology of work defined the paths for each of the traffic, shown in Figure 2, between the video server and remote stations.

This leaves in evidence a possible overload on the WAN links for the unicast traffic, which can compromise the bandwidth, performance and QoS, depending on the number of flows that pass through and the additional traffic to the video, typical of a business organization.

For the evaluation of the influence of the unicast traffic analyzes the bit rate, measured in each link individually for each scenario, and carried out a comparison between the same. In Figure 3 shows the bit rate for the scenario 4. This scenario implies an even greater overload on the WAN links for the unicast traffic, which can be compromised depending on the additional traffic to the video, typical of a business organization. In the link server output (1) there are 20 simultaneous flows, in the Link 2 there are 13 simultaneous flows, in the Link 3 there are 10 simultaneous flows, in the Link 4 there are 6 simultaneous flows, in the Link 5 there are 7 simultaneous flows and on the Link 6 there are 4 simultaneous flows. In this scenario it is clear that the overload, especially for the codecs MPEG-4 AVC and MPEG-2, is approaching the upper limit of capacity of the link 3.

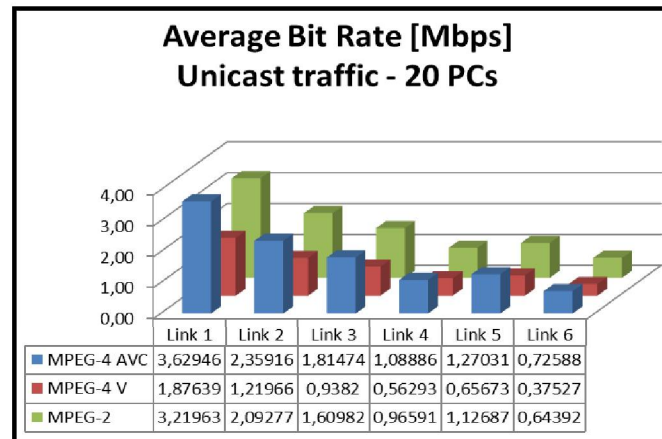


Fig. 3. Average rate per link for 20 PCs

4.2 Impact on the QoS

In addition, the transmission of video and multimedia over data networks facing specific problems in regard to the quality perceived by users. Several types of degradations usually occur in the video signals transmitted over packet networks.

The specific factors of IP networks that affect the quality of video, and on which we can draw conclusions from the tests carried out, are: packet loss, delay, jitter and transfer rate. In the tests carried out in the 4 scenarios, not detected packet loss, because we are using a network topology and configuration set to the application. According to the analyzed in Figure 4, the delays obtained in all cases do not exceed the 70 ms. In Figure 5, in the case of multicast the delay difference is not greater than 30 ms, while in the case of unicast, the jitter maximum is can generate some problems in video playback. In each video codec is identified the rate of transfer required, according to the resolution of the video. The bandwidth of each link used in the transfer must be sufficient for the transfer of all video streams participants, like all other traffic present. If this is not true, it may appear the above problems of packet loss and delays over the allowed. It was possible to observe the influence of the increase of occupation of the bandwidth of the links according to the amount of intervening for unicast traffic flows. For the multicast traffic, the increase of clients does not affect the bandwidth. In Figure 6 summarize this behavior, showing the total average transfer rate for each codec, in each of the scenarios. It is clear that the traffic is kept constant for multicast, regardless of the amount of clients; while, for the unicast traffic, at the exit of the server, the transfer rate increases proportionally to the number of clients connected, which can give an idea of the maximum amount of clients depending on the available bandwidth.

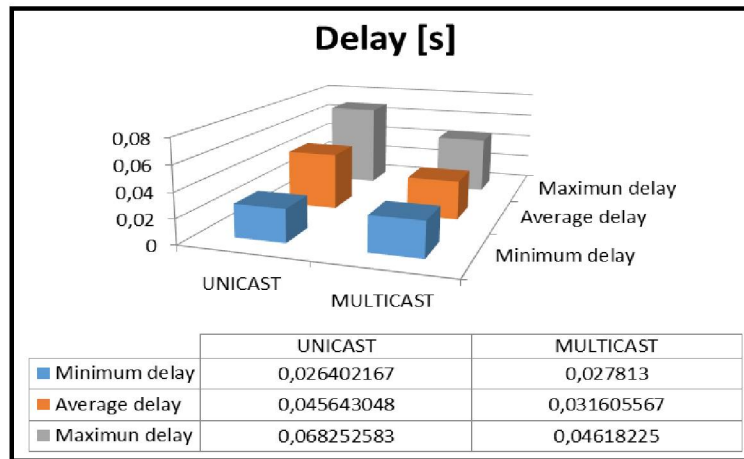


Fig. 4. Delay average, maximum, and minimum

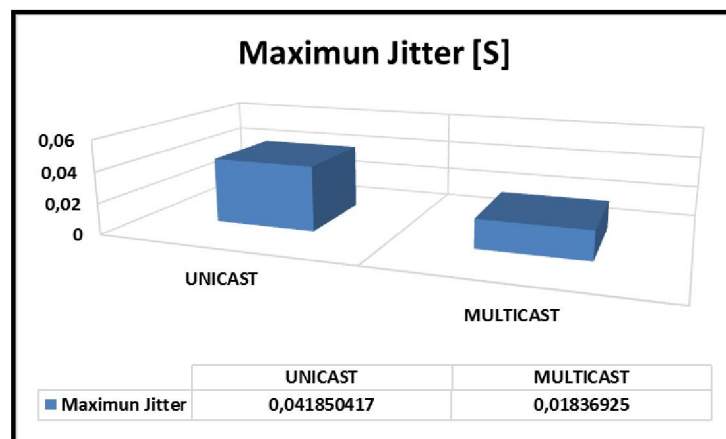


Fig. 5. Maximum jitter

4.3 Impact of the bursts of traffic

Most of the values of the metrics up to now have been averages. However, it must be considered that the video traffic introduces bursts of frames, that is to say, spikes in traffic. This may cause transient load commitments in the links and generate losses of frames. The best way to assess this behavior is to analyze the statistical distribution of the interframe time for the different codecs in one of the ends of the link.

In statistics, the Pareto distribution, formulated by the sociologist Vilfredo Pareto, is often used with said object. It is a continuous probability distribution with two parameters a and b whose density function for values $x \geq b$ is:

$$f(x) = \frac{ab^a}{x^{a+1}}$$

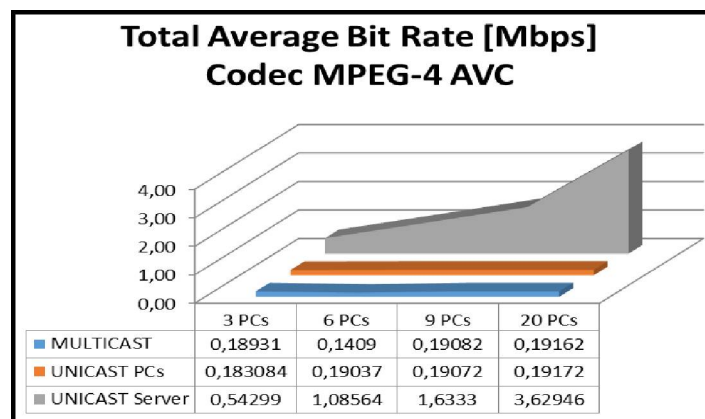


Fig. 6. Total average transfer rate – MPEG-4 AVC

- Behavior autosimilar codec MPEG-4 AVC, MPEG-4 VISUAL and MPEG-2

It analyzes the distribution of the 906, 913 and 608 frames of the video transfer, grouped by interframe spaces, for MPEG-4 AVC, MPEG-4 VISUAL and MPEG-3, respectively, compared with the expected distribution of Pareto. According to the previous point following vectors A comply with the Pareto distribution, and the type of traffic autosimilar has the parameters $a=0.75$ and $b=1.3$, $a=0.4$ and $b=1.75$, and $a=0.98$ and $b=1.15$, respectively.

	0.000	0.001	0.002	0.003	0.004	0.005	0.006	
0	451	200	56	43	30	24	23	...

	0.000	0.001	0.002	0.003	0.004	0.005	0.006	
0	274	124	101	87	43	36	28	...

	0.000	0.001	0.002	0.003	0.004	0.005	0.006	
0	344	98	57	25	22	17	12	...

- Comparison of the autosimilar distribution between codecs

In Figure 7 are analyzed comparatively standard distributions of the frames grouped by interframes spaces for the 3 codecs. It is noted that although all have a general behavior and typical traffic autosimilar have slight differences on the point of greatest concentration of burst of traffic (indicated in the various parameters a and b obtained from the Pareto function).

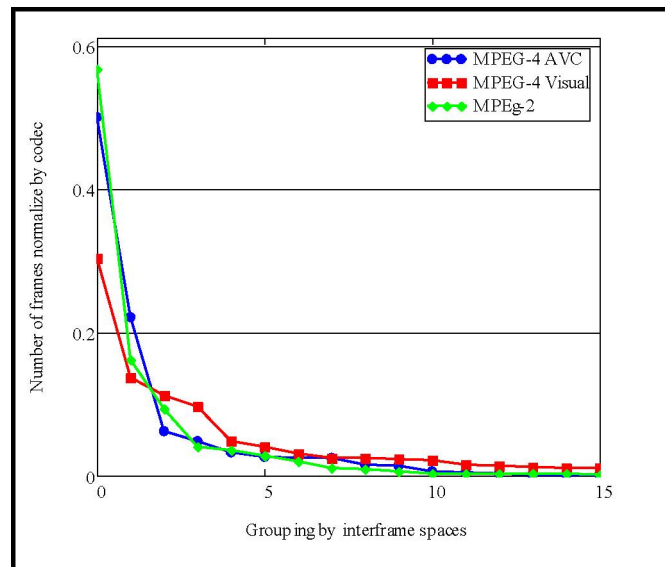


Fig. 7. Contrast of the number of frames 3 standard codecs

5 Conclusions and future work

This experimental work had as objective to make a contribution to the knowledge, updating and advancing the art state in the performance and QoS, multicast and unicast traffic for streamings of video in a real lab network as a testbed of a WAN.

It is presented in detail the experimental results obtained on various direct metric, average and statistics, according to a general topology and 4 scenarios, using a streaming video with 3 different encodings, with types of multicast and unicast traffic.

The pilot study shows that:

- The metrics collected on the laboratory pilot WAN show that multicast traffic provides the QoS and the expected return on each station, and on the server network segment, and the shared segments and links, using different types of codecs, to guide us about the behavior it can expect in a WAN real.
- Variations in the characteristics of the performance metrics and QoS observed in the multicast traffic video basically depend on the codec used. However, in the case of the unicast traffic of video it increases the dependency on the available bandwidth,

the interframe time, and the behavior of bursts on the various shared links (which vary directly according to the amount of connected clients simultaneously).

- The unicast traffic was more sensitive than the multicast for the number of end users, especially in the segment where is the unicast applications server and in the shared segments by the groups of customers unicast.
- The impact of the mixture of a global traffic link will depend on the topology of the codecs used, the number of customers to serve and the settings that are applied to the topology by multicast protocols of the network. And specifically, will depend on whether the video traffic is unicast or multicast.
- An adequate administrative management of the network can obtain an efficient use of the resources available, without compromise performance and the expected QoS.

Future work are oriented to the continuation and deepening of this study. In one of them follows a line of research by adding wireless clients to the topology and IPv6 and its new features for multicast traffic.

References

1. <http://www.arubanetworks.com/>, (Accessed March 16th, 2017)
2. Ian Brown, Jon Crowcroft, Mark Handley, Brad Cain "Internet Multicast Tomorrow" - Articles of interest Vol 1 N°6 - <http://www.isoc.org/pubs/int/cisco-1-6.html>
3. A. Abdalla, A. Mazhar, M. Salah and Sahar Khalaf, "Comparative Study of Compression Techniques for Synthetic Videos," *The International Journal of Multimedia & Its Applications (IJMA)*, vol. 6, no.2, April 2014.
4. D. Grois, D. Marpea, T. Nguyena and O. Hadarb, "Comparative Assessment of H.265/MPEG-HEVC, VP9, and H.264/MPEG-AVC Encoders for Low-Delay Video Applications," *SPIE Proceedings*, vol. 9217, Applications of Digital Image Processing XXXVII, California, USA, September, 2014.
5. V. Chandrasekar, K. Baskaran - Performance of Video Conferencing using Protocol Independent Multicast Routing with Core failure - *International Journal of Computer Applications* (0975 – 8887) Volume 41– No.10, March 2012
6. Sorin Cocorada - Multicast vs. Unicast Transmissions for wireless IP Camera Surveillance Suystems – *Acta Technica Napocensis – Vol 48 N°3 2007*
7. S. Tanwir, H. Perros and B. Anjum, "A QoS evaluation of video traffic models for H.264 AVC video," *Fifth International Conference on Next Generation Networks and Services (NGNS)*, pp. 313-320, Casablanca, Morocco, May, 2014.
8. S. Pérez, H. Facchini, A. Dantiacq, G. Cangemi and J. Campos, "An Evaluation of QoS for intensive video traffic over 802.11e WLANs-Quantitative analysis," *International Conference on Electronics, Communications and Computers (CONIELECOMP)*, pp. 8-15, Cholula, Mexico, february, 2015.
9. S. Pérez, H. Facchini, A. Dantiacq, G. Cangemi and J. Campos, "Behavior of Codecs for Multicast Video Traffic using WAN Test Bed - Experimental Study," *International Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON 2015)*, pp. 269-274, Santiago, Chile, October, 2015.
10. Acosta Escobar M, Treminio Henriquez J, Estudio de IPTV multicasting para la Universidad de Don Bosco – Trabajo final Ingeniería en Ciencias de la Computación.
11. Unreal Streaming Technologies, Unreal Media Server, Los Ángeles, USA, Available: <http://umediasever.net/>.
12. Available: <https://www.wireshark.org/>.
13. Available: https://www.youtube.com/watch?v=iLbnDWS3f_g

Modelado y simulación de arquitecturas de Cloud Computing con CloudSim: comunicación entre entidades.

Rosatto Daniel¹, Bond Román¹, Belizán Maximiliano¹, Morales Martín^{1,2}, Encinas Diego^{1,3}

¹Instituto de Ingeniería y Agronomía - Universidad Nacional Arturo Jauretche

²Unidad CodApli - Facultad Regional La Plata - UTN

³Instituto de Investigación en Informática LIDI (III-LIDI) - Facultad de Informática - UNLP

danielrosatto@gmail.com, roman.alejandra.b@gmail.com, maximiliano.h.belizan@gmail.com,
martin.morales@unaj.edu.ar, dencinas@unaj.edu.ar

Resumen. Las diferentes herramientas de simulación para las distintas áreas del conocimiento son excelentes sistemas proveedores de posibles comportamientos en cuanto a obtener conclusiones previas durante el diseño. En el caso de los sistemas de Cloud Computing, la diversidad de herramientas de simulación es escasa e insuficiente, comparando con otros sistemas de simulación como por ejemplo redes de computadoras. En este trabajo se abordará el desarrollo obtenido como una alternativa válida para la implementación de entidades de simulación capaces de soportar intercambios de datos simulados en el contexto de un framework de simulación de arquitecturas de Cloud Computing.

Palabras clave: Cloud Computing, Redes, Simulación.

1 Introducción.

Las ventajas de los sistemas Cloud se combinan con la centralización de la autenticación, la seguridad en el acceso y el licenciamiento de los sistemas virtualizados. Los grandes proveedores de servicios Cloud, desarrollaron productos comerciales para cubrir las necesidades TI de pequeñas, medianas y grandes empresas. En este marco, es posible desarrollar conceptos académicos pero siempre hay restricciones que se vinculan con las características del despliegue, o restricciones relacionadas con los servicios que brinda el proveedor de Cloud. Algunas son: los códigos fuente de los servidores Cloud generalmente no se publican, no es posible acceder a un paquete compilado, o nunca se sabe fehacientemente en donde se encuentran los datos o información.

Al comenzar con el desarrollo de un gran despliegue de un sistema Cloud, los posibles riesgos a abordar son los costos, la incertidumbre de la implementación del paradigma, el nivel de confianza y seguridad que se le proveerá al cliente, según lo expresado en el acuerdo del nivel de servicio. Una buena forma de previsualizar los

resultados es mediante la utilización de sistemas de simulación, configurando los posibles escenarios que reflejen situaciones reales y analizando los datos de salida para determinar si es viable o no realizar tal despliegue.

La herramienta de simulación utilizada es el framework CloudSim, que mediante entidades (datacenters, máquinas virtuales, etc), permite reflejar diferentes escenarios de infraestructura Cloud. Debido a las necesidades de este trabajo y ya que el core de CloudSim permite el agregado de nuevas funcionalidades, se desarrolla una expansión de CloudSim. En el siguiente apartado se explican los criterios de elección del framework CloudSim para realizar el aporte.

1.1 Simulación de arquitecturas de Cloud Computing.

CloudSim [1] [2] es un framework de simulación generalizado y extensible que permite el modelado, la simulación y la experimentación de diferentes infraestructuras y servicios de aplicaciones de Cloud Computing. Un ejemplo de utilización es la simulación de muchos centros de datos.

Su arquitectura consiste en entidades específicas que se representan como clases Java que pueden ser heredadas o variadas para simular experimentos. Estas clases representan centros de datos, hosts físicos, máquinas virtuales, servicios a ejecutar en los centros de datos, servicios en la nube de usuarios, redes internas, centro de datos y consumo de energía de los hosts físicos y elementos de los centros de datos. Además, CloudSim soporta la inserción dinámica de los elementos de simulación y proporciona aplicaciones de paso de mensajes y la topología de la red del centro de datos.

Una definición importante de CloudSim es la de entidad. Una entidad es una instancia de un componente, que es una clase o un conjunto de clases que representan un modelo CloudSim (data center, host). El motor de simulación es capaz de simular el tiempo de ejecución de las aplicaciones ingresadas como Cloudlets con información básica [3].

La versatilidad de CloudSim es la principal ventaja del sistema. La integración de nuevos parámetros y conceptos de la simulación es implementada desde abstracciones preestablecidas convenientemente en este trabajo. Las abstracciones principales son: SimEvent [4], SimEntity [4], DataCenterCharacteristics y Virtual Machine (Vm). En cada una de estas instancias es posible establecer el detalle del comportamiento y el estado de la simulación teniendo en cuenta el alcance del modelado que se planifica.

Se propone una simulación que provea métricas de entrada/salida de servicios Cloud simulados tales como SAAS, PAAS o IAAS desarrollada con objetos que heredan de SimEvent y de SimEntity pero con sus métodos y estados únicos en cada simulación. Las denominaciones de los servicios SAAS, PAAS e IAAS hacen referencia al objetivo del sistema Cloud, es decir cuál es el servicio que provee.

- Un sistema SAAS, (Software as a service) brinda servicios de software a través del acceso Cloud, como por ejemplo el reproductor de videos online de Youtube.
- Un sistema PAAS (Platform as a service) brinda una plataforma de hardware virtualizado a través del acceso Cloud, como por ejemplo los storages de archivos como SugarSync.

- Un sistema IAAS (Infrastructure as a service) brinda una infraestructura de hardware y software completa accesible a través de acceso Cloud, como por ejemplo, los servidores virtuales de Iplan.

Los grandes proveedores de servicios Cloud brindan los tres paradigmas juntos, de tal manera que proporcionan servicios integrales.

1.2 Simulación de procesamiento de datos.

CloudSim establece que el entorno de simulación es un sistema de Data Centers con la administración de un broker que asigna los recursos disponibles. En el modelo de simulación de CloudSim, el objetivo es procesar todas las instancias de Cloudlets, que se definen al inicio de la simulación.

Las Cloudlets son definiciones de unidades de información que cualquier Dashboard de un sistema Cloud genera por la interacción de los distintos clientes. En el modelo de CloudSim, las Cloudlets contienen los requisitos de memoria necesaria, de forma que el broker asigne a una Vm capaz de procesarlo, según su algoritmo de asignación.

2 Desarrollo.

Se logró crear un desarrollo basado en las clases SimEntity y SimEvent de CloudSim, utilizando metodologías de expansión de trabajos previos [5] y [6]. Estas clases no están vinculadas con los Cloudlets porque el objetivo del modelado del ambiente de simulación de CloudSim no es procesar algoritmos de entrada. Sólo obtener la cantidad de Cloudlets procesados por cada data center y cuál fue su costo temporal. En éste caso se establecen las modificaciones en el ámbito de CloudSim para configurar a la memoria Ram de cada Vm (Virtual Machine) como una entidad de simulación que procesa eventos de simulación. Es la condición inicial de la simulación para observar y medir procesos distribuidos en cada una de las Vm del broker. Los datos en las celdas de cada SimEntity Ram, (que se ha denominado RamEntity) serían sincronizados por los eventos de simulación autogenerados por la lógica del nuevo modelo. La medición de los procesos se realizaría en el contexto de un Benchmark precargado en el modelo.

En lo referido a Cloudsim, los servicios Cloud que implementa son capaces de procesar las cloudlets que definen los creadores de CloudSim. En este caso la opción es evaluar los resultados obtenidos en el Log de CloudSim al finalizar el proceso sintético de Benchmark, que interactúa con el Sistema de Archivos sintético de CloudSim. Se explica en el siguiente apartado los llamados y las delegaciones del control entre cada uno de los actores de la simulación.

2.1 Expansión de entidades.

Para crear una nueva entidad cloudsim se debe proceder de manera similar a los cloudsim-examples pero adecuándose a las necesidades y sin modificar los archivos originales de cloudsim.

Se incorporaron las siguientes entidades: primero, una RamEntity como componente de las máquinas virtuales en reemplazo de la variable ram.

En segundo lugar, se cambió la clase Vm por SyntheticVm con una composición de RamEntity.

Los procesos sintéticos surgen de la necesidad de respuesta de la simulación ante factores internos. La delegación de la ejecución de las líneas de procesos sintéticos logra que la respuesta de su algoritmo sea obtenida por los recursos simulados de CloudSim independiente del hardware real que está ejecutándose. En la Figura 1 se observa que toda la simulación depende exclusivamente de los recursos que provee la capa de simulación, debiendo reportar cada paso a la capa de control.

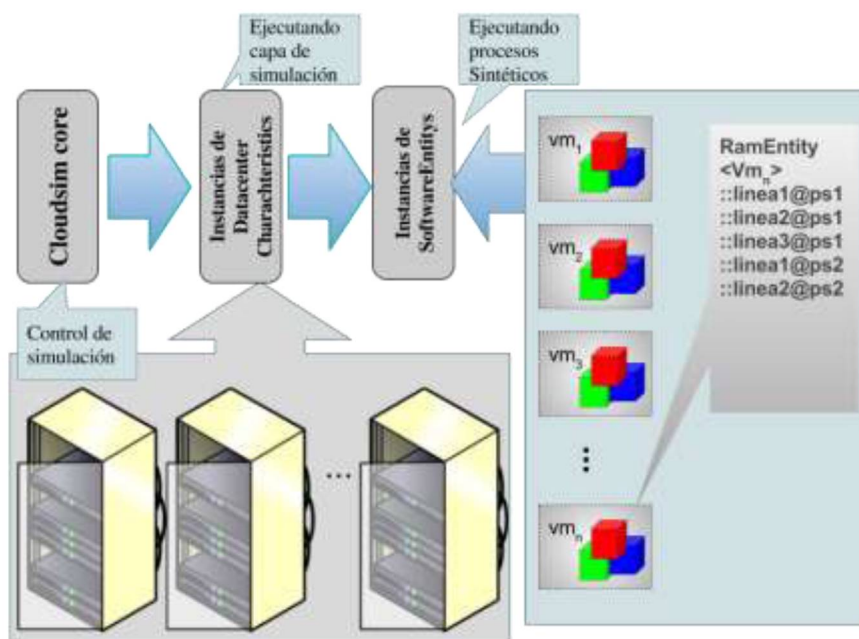


Figura 1: Diagrama de expansión de entidades de CloudSim

Se exponen a continuación el método de desarrollo de la entidad RamEntity, que proporciona el soporte para mantener la simulación de celdas accesibles mediante eventos de simulación.

Estudiando detalles de la clase abstracta SimEvent fue posible definir los requerimientos de software que la nueva entidad debe cumplir. Los requisitos son adecuados para la implementación de RamEntity a partir del cumplimiento de sus restricciones de software. Es por eso que se generan test JUnit RamEntity_test basado en cada caso de los requisitos de los métodos heredados de la superclase SimEntity.

Los Unit test producen una implementación de RamEntity que verifican el cumplimiento de los requerimientos de la superclase con el desarrollo aportado.

En la figura 2 se detallan las relaciones entre las nuevas entidades, el core del simulador y los eventos. Lo interesante en este modelo es que cada celda de RamEntity soporta la asignación de un Object de java. Es decir que se conserva la versatilidad original de CloudSim.

El evento inicial debe ser configurado antes de hacer un start simulation en el tiempo que se asigna en el método send del core de CloudSim. El evento al estar destinado al id (identificador) de la RamEntity2 y la fuente debe ser el id de RamEntity2. Será procesado por la simulación con la correspondiente etiqueta del evento, que en éste caso es una Ram_Access_Read. La versatilidad del simulador se refleja en la posibilidad de que RamEntity1 tiene la capacidad de generar un nuevo evento como respuesta.

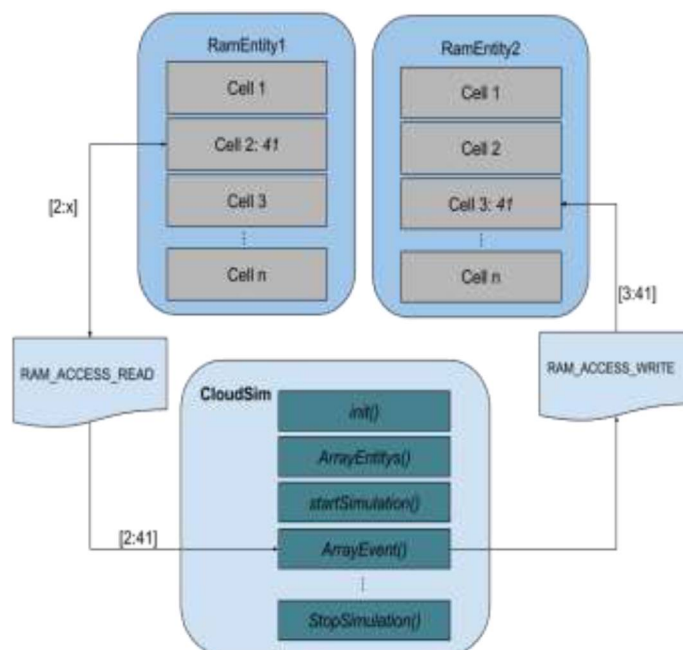


Figura 2: Diagrama de transferencia de información entre celdas RamEntity

El evento de respuesta de RamEntity1 contiene el valor de la celda leída, y el id de destino en el tiempo que el gestor de tiempos le asigna para la respuesta del hardware. El evento de respuesta se guarda en la cola de eventos futuros de CloudSim. Luego, se continúa recorriendo la lista de eventos; hasta que llega el evento en el que la respuesta obtenida en la lectura de RamEntity1 (en este caso 41) se debe enviar al destinatario, RamEntity2. Se procesa según lo etiquetado en el Ram_Access_Write, con el contenido proporcionado por RamEntity1 (celda de la RamEntity2 y el dato) y se escribe en la RamEntity2.

3 Resultados.

La simulación completa consta de un datacenter con un broker y dos máquinas virtuales vm_1 y vm_2 corriendo a un solo Cloudlet, como se ve en la Figura 3. El primer resaltado en cian, de la Tabla 1, es el momento en el que finalizan las inicializaciones de las entidades previo al comienzo de la simulación. Notar que las RamEntities inicializan las celdas durante la inicialización de las entidades. Luego el Broker asigna los recursos para establecer el proceso del Cloudlet.

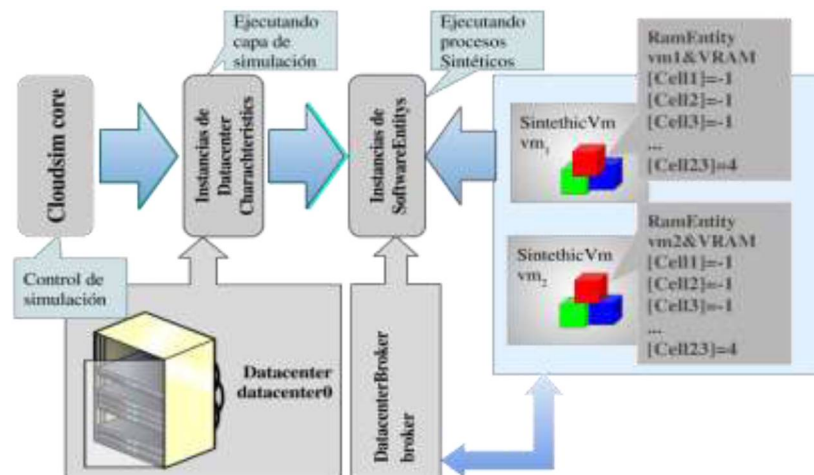


Figura 3: Diagrama de la simulación realizada en este trabajo.

En la Tabla 1 se exponen los resultados de la primera versión de la implementación de RamEntity y se explican a continuación. Cada instancia de RamEntity lleva el nombre VRAM, y a su vez son componentes de las entidades SinteticVm cuyas instancias tienen los nombres vm_1 y vm_2 .

Tabla 1: Salida LOG de la simulación de RamEntity.

```

Starting BenchmarkingCloudSimExample1...
Initialising...
Starting CloudSim version 3.0
Datacenter_0 is starting...
Broker is starting...
Entity >vm1&VRAM< : Cells init...
Entity >vm2&VRAM< : Cells init...
Entities started.
0.0: Broker: Cloud Resource List received with 1 resource(s)
0.0: Broker: Trying to Create VM #0 in Datacenter_0
0.0: Broker: Trying to Create VM #0 in Datacenter_0
0.1: Broker: VM #0 has been created in Datacenter #2, Host #0
0.1: Broker: Creation of VM #0 failed in Datacenter #2
0.1: Broker: Sending cloudlet 0 to VM #0
Entity >vm1&VRAM< :59.0 SUCCESS ACCESS READ: -1
Entity >vm1&VRAM< :79.0 SUCCESS ACCESS WRITE: -1 <- vm1&VRAM[23]=4
Entity >vm1&VRAM< :99.0 SUCCESS ACCESS READ: 4
Entity >vm2&VRAM< :101.0 SUCCESS ACCESS WRITE: 4 <- vm2&VRAM[23]=4
Entity >vm2&VRAM< :109.0 SUCCESS ACCESS READ: 4
Entity >vm1&VRAM< :: ENULL Ram event type, time: 118.91 ,data: -1
Entity >vm1&VRAM< :: ENULL Ram event type, time: 198.91 ,data: 4
Entity >vm2&VRAM< :: ENULL Ram event type, time: 218.91 ,data: 4
400.1: Broker: Cloudlet 0 received
400.1: Broker: All Cloudlets executed. Finishing...
400.1: Broker: Destroying VM #0
Broker is shutting down...
Simulation: No more future events
CloudInformationService: Notify all CloudSim entities for shutting down.
Datacenter_0 is shutting down...
Broker is shutting down...
Entity >vm1&VRAM< :Shutdown ...
Entity >vm2&VRAM< :Shutdown ...
Simulation completed.
Simulation completed.

===== OUTPUT =====
Cloudlet ID   STATUS   Data center ID   VM ID   Time   Start Time   Finish
Time
0            SUCCESS    2                0       400    0,1          400,1
Benchmarking CloudSim Example1 finished!

```

Luego, en el segundo resaltado cian, el evento que interactúa con vm1&VRAM produce una lectura de la celda 23, luego escribe 4 en la misma celda. Posteriormente se verifica en el tiempo 99 el valor 4 en la celda 23.

Seguidamente lo resaltado en amarillo verifica de manera similar la escritura de la celda 23 de vm2&VRAM. Los eventos ENULL posteriores representan a los eventos generados por las respuestas de las lecturas y escrituras de RamEntity. Los tiempos de las devoluciones ENULL son los sincronizados por el gestor de tiempos.



Figura 4: Comparación de los tiempos consumidos por cada respuesta de la RamEntity ante los eventos generados.

Se observa en la Figura 4 una comparación de los tiempos de respuesta obtenidos por las acciones de las SimEntity implementadas. En futuras versiones las respuestas de los tiempos se combinarán con un análisis estadístico para establecer las estimaciones de las variables temporales que se requieran.

Tabla 2: [9]

TIEMPO MEDIO DE INICIO DE MV		
Cloud	OS	Tiempo
EC2	Linux	96.9 segundos
EC2	Windows	810.2 segundos
Azure	WebRole	374.8 segundos
Azure	WorkerRole	406.2 segundos
Azure	VMRole	356.6 segundos

En la Tabla 2 se pueden observar los tiempos consumidos en ejecutar una máquina virtual, con distintos sistemas operativos, en los dos proveedores de Cloud Computing más conocidos. Con CloudSim el tiempo es mucho menor, para poder centrarse más en métricas de funcionamiento y de hardware simulado.

4 Conclusiones.

Se logró establecer un método de desarrollo adecuado para un framework de simulación de arquitecturas Cloud. El objetivo de lograr que las entidades de simulación soporten intercambios de datos fue cumplido exitosamente en la entidad RamEntity. Se destaca que el método es genérico para cualquier entidad que se quiera implementar a futuro de acuerdo a los objetivos del modelo.

La precisión del intercambio de los datos es susceptible al comportamiento implementado en el método que procesa los eventos. De manera que mientras más eventos se incluyan en la implementación de éste método, se incrementa la calidad de la transferencia de información sintética entre entidades.

Ampliar este framework ayuda a comprender no sólo los entornos Cloud Computing, sino que también se puede entender mejor este framework para poder seguir expandiéndolo y luego replicarlo en un espacio físico.

5 Trabajos futuros.

Esta línea de investigación es la base a una serie de extensiones de funcionalidades y características que continúan en la etapa de desarrollo. El modelo de simulación seguirá proporcionando la versatilidad original de CloudSim, y además se proporcionarán nuevos actores en el escenario de la simulación. Dichos actores serán entidades y eventos que se consideran relevantes según lo investigado en las experiencias actuales de desarrollo para complementar otras líneas de investigación [7] [8] obteniendo un modelo aplicable.

En los siguientes trabajos, los intercambios de datos en RamEntity darán la posibilidad de crear procesos simulados en una tabla de procesos, también simulada en cada SyntheticVm.

La tabla de procesos simulada estará incluida en la InitEntity que será una entidad administradora de procesos de cada SyntheticVm para dar una respuesta adecuada en cada VirtualMachine simulada.

6 Referencias

1. CloudSim - <http://www.cloudbus.org/cloudsim>, julio 2017
2. Rajkumar Buyya “CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms” Published online 24 August 2010 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/spe.995.
3. Kushang Parikh, Nagesh Hawanna, Haleema. P.K, Jayasubalakshmi.R and N.Ch.S.N.Iyengar. School of Computing Science and engineering Vellore Institute of Technology, Tamil Nadu, “Virtual Machine Allocation Policy in Cloud Computing Using CloudSim in Java.” 2015.
4. Fred Howell & Ross Mc Nab “SimJava: A Discrete Event Simulation Library For Java”. International Conference on Web-Based Modeling and Simulation. 1998.

5. Rodrigo N. Calheiros , Rajiv Ranjan, Anton Beloglazov, César A. F. De Rose and Rajkumar Weiwei Chen , Ewa Deelman - Information Sciences Institute-University of Southern California Marina del Rey, CA, USA - WorkflowSim: A Toolkit for Simulating Scientific Workflows in Distributed Environments. 2012
6. Hamza Ouarnoughi, Jalil Boukhobza, Frank Singhoff, Stephane Rubini, Erwann Kassis. "Considering I/O Processing in CloudSim for Performance and Energy Evaluation". OpenStack Cloud Software: Open source software for building private and public clouds. © Springer International Publishing AG 2016 M. Tafer et al. (Eds.): ISC High Performance Workshops 2016, LNCS 9945, pp. 591–603, 2016. DOI: 10.1007/978-3-319-46079-6 40.
7. Zaccardi, G.; Galarza, B.; Encinas, D.; Morales, M. "Implementación de Cloud Computing utilizando OpenStack". II Congreso Nacional de Ingeniería Informática y Sistemas de Información (CoNaIISI 2014). Ciudad de San Luis, Argentina.
8. Galarza, B.; Zaccardi, G.; Encinas, D.; Morales, M. "Análisis de despliegue de una IaaS utilizando Openstack". XXI Congreso Argentino de Ciencias de la Computación (CACIC 2015). Ciudad de Junín, Argentina.
9. Ming Mao, Marty Humphrey "A Performance Study on the VM Startup Time in the Cloud". IEEE 5th International Conference on Cloud Computing (CLOUD), 2012.