# New Cloud Services for product placement in television *

†

André Regado
MOG Technologies
Porto, Portugal
andre.regado@mog-technologies.com

Alexandre Ulisses
MOG Technologies
Porto, Portugal
alexandre.ulisses@mog-technologies.com

Miguel Poeira
MOG Technologies
Porto, Portugal
miguel.poeira@mog-technologies.com

Pedro Santos
MOG Technologies
Porto, Portugal
pedro.santos@mog-technologies.com

## ABSTRACT

**Over the last years, video consumption under digital format is increasing. There are more people watching television through their tablet's, smartphones and computers. This allows that adverts maybe targeted according the final user preferences. The goal of this work is to present a cloud based platform that allows an insertion of personalized and contextualized ads, in real time, based on the end user network location.**

## CCS CONCEPTS

•**Information systems → Multimedia streaming;** *Computational advertising;*

## KEYWORDS

Native Advertising, Advertising, Location Based Services, Cloud Computing, Adaptive Bitrate Streaming

**ACM Reference format:**
André Regado, Alexandre Ulisses, Miguel Poeira, and Pedro Santos. 2017. New Cloud Services for product placement in television . In *Proceedings of ACM International Conference On Interactive Experiences for Television and Online Video, Hilversumlands, March 2017 (TVX'17),* 4 pages.
DOI: 10.475/123_4

## 1 INTRODUCTION

The increase of video consumption under digital format is evident. Nowadays everyone has a smartphone or tablet, where they can watch television. Usually, these devices have Internet access which has lead to a substantial growth of the online video marketing market in the last few years.

Native Advertising is the new way to display advertising, where the ads will be embedded in the video content so that the viewer can't reject the advertisements' messages.

The registration of the user in a specific wifi network, for watching videos, defines a context that allows personalized adverts streaming.

In this paper, we present a high-level architecture for contextualize product placement in online television using a cloud-based platform.

## 2 ADVERTISING IN VIDEO

The way to do advertising in video is changing. Traditionally, adverts appeared on Television in ad breaks between TV program segments, where the use of pre-roll, mid-roll and post-roll are well known. However, the traditional ad breaks became ineffective and inefficient, so it's crucial to change the way Tv ads are displayed. Native Advertising is a concept whose main goal is to improve the way we display ads, putting an end to the concept of advertisement breaks, by embedding them in the content of the video itself (IAB 2016; Pulizzi 2014).

We can apply Native Advertisement in video. However, this poses a complex problem as each part of the program has an unique way to display its content. A possible way to do this is by replacing the objects embedded in the video according the final user.The incorporation of these objects in the video must be done in a seamless way in order to avoid any kind of interference with the main content. To achieve this naturalness, computer vision techniques are needed to match the object into the content namely by controlling the texture and lightning of the new object.

## 3 PERSONALISED ADVERTISEMENTS IN VIDEO

In this work, two approaches for personalized advertising were combined: Contextual Match Advertising and Location Based Advertising (Shatnawi and Mohamed 2012). The proposed architecture will fuse different objects into the video content that, somehow, will be related with it. The choice of objects to be embedded will be based on Location-based services (LBS), which allow us to know where the end user device is (Kushwaha and Kushwaha 2011; Roxin et al. 2008).

The usage of location-based advertising is possible due to the fact that any place where the user goes, he will usually carry a smart device with him. Location-Based Advertising is efficient because, first of all is contextualized as ads could be personalized according to user location. Besides that, it is timely because location

*Produces the permission block, and copyright information
†The full version of the author's guide is available as `acmart.pdf` document


*TVX'17, Hilversumlands*
 123-4567-24-567/08/06. . . $15.00
DOI: 10.475/123_4

data is recorded in real time and it allows brands to reach people on a specific precise moment (Dang and Chang 2015). Nowadays, location based services are already used in various applications (Buczkowski 2012; Shu Wang and Yi 2008).

- Marketing: companies use user's location to send messages to them.
- Information Services: when a user wants to know the closest restaurant or cinema.
- Navigation: navigation services allows the user to pinpoint its exact geographical coordinates and get the directions to the required location.
- Hobby: there are games that take advantage of the user's location for gamification purposes (e.g. Pokemón GO).

## 4 HIGH-LEVEL ARCHITECTURE

In this section, we provide a high-level architecture for product placement in television using a cloud-based platform. We will present how the system works and how its components interact.

### 4.1 MPEG-DASH

MPEG-DASH is an Adaptive Bitrate Streaming technique, it works by breaking the content into a sequence of small HTTP-based file segments. The server will provide the video content in multiple different bit rates. In the beginning of the session, a client requests a Media Presentation Description (MPD) to the Server. While the DASH client is buffering and playing the video content, it will also analyse the variation of the network bandwidth and, depending on this analysis, it will decide which segments to download to maintain the appropriate buffering.

In summary, DASH utilizes an algorithm that uses the maximum streaming bit rate that the network allows in order to deliver the content with the expected Quality of Experience (QoE) to the user (Swaminathan 2013; Timmerer and Griwodz 2012).

By taking advantage of cloud's benefits, we are proposing an architecture that is (Almubaddel and Elmogy 2016; Alzahrani et al. 2014; Mell and Grance 2011):

- Built on modular components and blocks, that can be re-utilized to build new functionalities.
- Expansible, which means that it can be adapted to new scenarios.
- Scalable, that is, operation's needs are not bound by hardware stock availability or ease of deployment.
- Collaborative, because it enables people to easily collaborate on the same project, wherever they are located.
- Enables fast, transparent updates and bug fixes, which lets the production team abstract on technical details.

### 4.2 Metadata

Video Multiple Ad Playlist (VMAP) is used to express the structure of the ad inventory as a set of timed ad breaks within a publisher's video content (Bureau 2014).

In the following image 2, we can see an example of one AdBreak in VMAP. The most important information is the time offset because it indicates the exact time that this segment needs to be inserted in the video content. One other important fact is that combined with the VMAP structure we receive a VAST response which has more information about the advertisement, such as its duration and location.

```xml
<vmap:AdBreak breakType="linear" breakId="myid" timeOffset="00:00:04">
    <vmap:AdSource allowMultipleAds="false" followRedirects="true" id="1">
        <vmap:VASTAdData>
            ...
        </vmap:VASTAdData>
    </vmap:AdSource>
    <vmap:TrackingEvents>
        <vmap:Tracking event="breakStart">
            http://MyServer.com/breakstart.gif
        </vmap:Tracking>
    </vmap:TrackingEvents>
</vmap:AdBreak>
```

**Figure 2:** *VMAP example*

Video Ad Serving Template (VAST) is a video ad-serving template that provides a uniform way for advertising data to be transferred from ad servers to video players independent of any technology. In others words, VAST provides a common protocol that enables ad servers to use a single ad format across multiple video players (Laytor et al. 2012).

In the following image 3, we can see an example of VAST. Through VAST we have information about the location and the duration of the ad:

```xml
<Ad>
    <InLine>
        <AdSystem>
            My Ad Server
        </AdSystem>
        <AdTitle>
            Car Company
        </AdTitle>
        <Description>
            Brand
        </Description>
        <Impression id="myid">
            ...
        </Impression>
        <Creatives>
            <Creative>
                <Linear>
                    <Duration>
                        00:05:23.125
                    </Duration>
                    <TrackingEvents>
                        <Tracking event="start">
                            http://MyServer.com/breakstart.gif
                        </Tracking>
                    </TrackingEvents>
                    <MediaFiles>
                        ...
                    </MediaFiles>
                </Linear>
            </Creative>
        </Creatives>
    </InLine>
<Ad>
```

**Figure 3:** *VAST example*

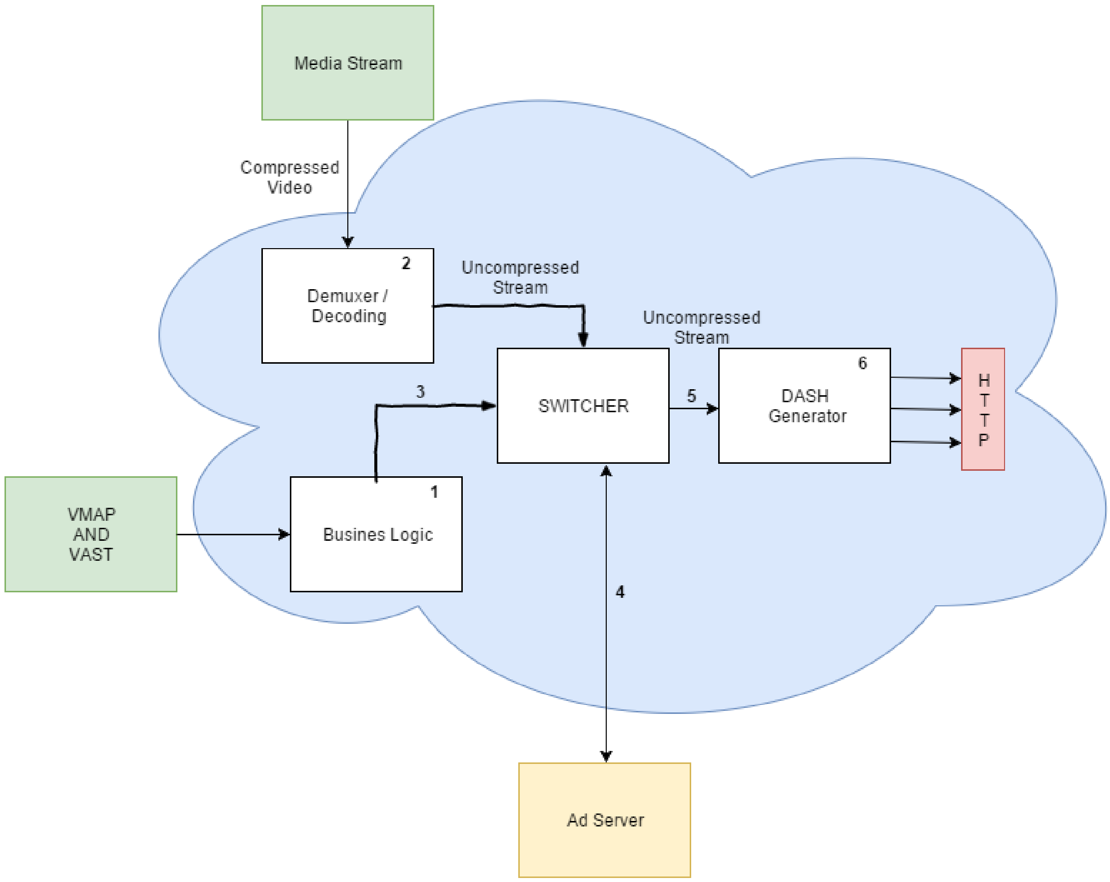


**Figure 1:** *Architecture. Components and its interactions*

*4.2.1 XML Schema (XSD) will be used to express a set of rules to which VMAP and VAST must obey in order to be considered valid according to that schema.*

## 4.3 Components

In this section, we present four components that compose our architecture 1. it will be demonstrated how the architecture will switch the original video with the ads according to the final user profile.

Native Advertisement is too computational intensive to be delivered live, so we see the initial systems having multiple parallel sequences of a program, each with different objects/items placed in them. So it is not a real time process because, for realism, it is necessary to consider shadows, reflections, among others, which need to be pre-processed.

The Ad Server will store the program segments in multiple versions, where the same short piece of program material has a different product placed in each version. This server needs to be connection wise, close to the program servers or CDN's. The goal of a CDN is to provide content to end-users focusing on improving its availability and performance by forming a system of cache servers that use geographical proximity as a criteria of how to deliver that content (Shafiq et al. 2012).

*4.3.1 Business Logic receives as an input two XML's (VAST and VMAP). It's responsible for parsing both XML's and for validating these XML's against their XSD (XML Schema). Then, Business Logic transfers the VMAP and VAST information to Switcher.*

*4.3.2 Demuxer receives as an input the compressed original video and deliveries as its output the uncompressed video.*

*4.3.3 Switcher receives as an input the uncompressed stream from the demuxer module and the information of VMAP and VAST from the Business Logic module. The main goal of this component is to switch between the main program and the correct segments.*

*4.3.4 Dash Generator receives as an input an uncompressed stream and as the name depicts, it is responsible for generating different representations (segments) and providing them.*

## 4.4 WorkFlow

In this section, we provide more details about the components that compose the proposed architecture 1 and how they interact each other.

(1) Business Logic processes VMAP and VAST. To do this, it was used the Microsoft XML Core Services (MSXML), a set of services that allows xml-based applications. MSXML uses Document Object Model (DOM), to handle a XML

document as a tree structure where each node is an object representing a part of the document. XPath was used to identify and navigate through nodes by using path syntax. It was also used to validate XML documents using XSD.

(2) Demuxer receives the original stream. A demuxer is a module responsible for splitting individual elements of a media file, e.g., video, audio or subtitles and sending them to their respective decoders. To do this, FFmpeg was used as an open source project for handling multimedia data like encoding and decoding.

(3) Through VMAP, Business Logic informs the Switcher about the number of segments that need to be changed, the exact time that it is needed to insert a segment embedded with ads and its duration and location by VAST.

(4) Switcher is the brain of the system. It is responsible for receiving information from other nodes and for switching streams between the original video and the segments with the most appropriate item placed in it and getting back to the original program stream. Switcher downloads the replacements segments from Ad Server. Our approach is to have a library of many short sections of the program, each with a different product placement in it. These will be stored in an Ad Server and the user's location identity will cause the recall of the most contextually appropriate sequence of placement.

(5) From Business Logic information and the original video from demuxer module, Switch outputs the correct video with the ads.

(6) DASH Generator breaks the original video content into HTTP-based segments/chunks (2 seconds) and encodes it at multiple bit rate, streaming them to the associated CDN.

## 5 CONCLUSION

This paper has described a high-level architecture for product placement in television using a cloud-based infrastructure. Traditional advertising breaks are dying. The new form of advertising is Native Advertising, where the item to be promoted appears placed within the program. This solution combines Native Advertising, personalized advertisements and location based services.

The viewer will see the program in real time, but the architecture will be switching streams seamlessly between main program scenes, selected stream of program with most appropriate item placed in it and back to the main program stream. When watching a video, for example, all viewers will see the same video most of the time, but the placement scenes will be served differently depending on the viewer's location.

## 6 ACKNOWLEDGMENT

This article is a result of the project Remote Prod 7819, supported by Norte Portugal Regional Operational Programme (NORTE 2020), under the PORTUGAL 2020 Partnership Agreement, through the European Regional Development Fund (ERDF).

## REFERENCES

Majed Almubaddel and Ahmed M Elmogy. 2016. Cloud Computing Antecedents, Challenges, and Directions. *Proceedings of the International Conference on Internet of Things and Cloud Computing* (2016), 16:1–16:5. DOI: http://dx.doi.org/10.1145/2896387.2896401

Ahmed Alzahrani, Nasser Alalwan, and Mohamed Sarrab. 2014. Mobile cloud computing: advantage, disadvantage and open challenge. *Proceedings of the 7th Euro American Conference on Telematics and Information Systems - EATIS '14* (2014), 1–4. DOI: http://dx.doi.org/10.1145/2590651.2590670

Aleksander Buczkowski. 2012. Location-Based Services - Applications. (2012). http://geoawesomeness.com/knowledge-base/location-based-services/location-based-services-applications/

Interactive Advertising Bureau. 2014. Video Multiple Ad Playlist (VMAP). (2014). http://www.iab.com/wp-content/uploads/2015/06/VMAP.pdf

Hung Dang and Ee-Chien Chang. 2015. PrAd: Enabling Privacy-Aware Location Based Advertising. *Proceedings of the 2Nd Workshop on Privacy in Geographic Information Collection and Analysis* c (2015), 1:1–1:8. DOI: http://dx.doi.org/10.1145/2830834.2830839

IAB. 2016. Digital Video In-Stream Ad Format Guidelines and Best Practices [Report]. May (2016), 4–23. http://www.iab.com/wp-content/uploads/2016/01/DVAFG

Amit Kushwaha and Vineet Kushwaha. 2011. Location Based Services using Android mobile Operating System. *International Journal of Advances in Engineering & Technology* 1, 1 (2011), 14–20. http://www.ijaet.org/media/Location-Based-Services-using-Android-mobile-Operating-System-Copyright-IJAET.pdf

R E V W Illiam R C Laytor, Betsy Currie, Rhonda Stubbs, Nancy Smith, Jenna Robinson, A M Jerry Broome, Mark Mitchell, Steve Thrailkill, and Ken Troutman. 2012. Video Ad Serving Template (VAST). (2012), 71. https://www.iab.com/wp-content/uploads/2015/06/VASTv3

Peter Mell and Timothy Grance. 2011. The NIST definition of cloud computing. *NIST Special Publication* 145 (2011), 7. DOI: http://dx.doi.org/10.1136/emj.2010.096966

Joe Pulizzi. 2014. The Ultimate Guide to Native Advertising. (2014). https://www.linkedin.com/pulse/20140107180859-5853751-the-ultimate-guide-to-native-advertising

Ana-Maria Roxin, Christophe Dumez, Maxime Wack, and Jafaar Gaber. 2008. Middleware models for location-based services. *Proceedings of the 2nd international workshop on Agent-oriented software engineering challenges for ubiquitous and pervasive computing - AUPC '08* (2008), 35–40. DOI: http://dx.doi.org/10.1145/1387249.1387255

M Zubair Shafiq, Alex X Liu, and Amir R Khakpour. 2012. Revisiting Caching in Content Delivery Networks. *Sigmetrics* (2012), 567–568. DOI: http://dx.doi.org/10.1145/2637364.2592021

Maad Shatnawi and Nader Mohamed. 2012. Statistical techniques for online personalized advertising. *Proceedings of the 27th Annual ACM Symposium on Applied Computing* (2012), 680–687. DOI: http://dx.doi.org/10.1145/2245276.2245406

Jungwon Min Shu Wang and Byung K Yi. 2008. Location Based Services for Mobiles: Techonogies and Stardards. January 2008 (2008). https://www.researchgate.net/publication/265758564

Viswanathan Swaminathan. 2013. Are we in the middle of a video streaming revolution? *ACM Transactions on Multimedia Computing, Communications, and Applications* 9, 1s (2013), 1–6. DOI: http://dx.doi.org/10.1145/2490826

Christian Timmerer and C Griwodz. 2012. Dynamic adaptive streaming over HTTP: from content creation to consumption. *Proceedings of the 20th ACM international* ... (2012), 1533–1534. DOI: http://dx.doi.org/10.1145/2393347.2396553