5. Case Studies

In the following sections we present two case studies where the Split & Merge architecture has been successfully used to increase the efficiency of video processing, with focus on reducing the total processing times.

5.1 The Split&Merge for Globo.com Internet Video Compression

Globo.com[1] is the Internet branch of Globo Organizations, the greatest media group of Latin America, and the leader in the broadcasting media segment for the Brazilian internet audiences. As Internet branch, Globo.com is responsible to support all companies from the group in their Internet initiatives. One of the most important of them is making available all content produced for the Globo TV and PayTV channels online. This means that Globo.com needs to produce more than 500 minutes of video content for the Internet every day. Moreover, all content needs to be produced in several different formats, to be consumed by various electronic devices, such computers, laptops, tablets and mobile phones.

For this task, Globo.com developed a video encoding platform, which runs on all content producing sites, such as sports, entertainment and journalism studios. However, with the popularization of the high definition content, the encoding task became much more complex, and the video producing process became a bottleneck, in particular for sports and journalism.

On the sports side, it is Globo.com's responsibility to put on the web all full matches of national soccer tournament, with approximately 2 hours of duration each. The problem is that on given Sundays there are 10 simultaneous matches being played and the encoding for one single format takes around 3 hours (considering a standard definition input). If the input is in HD, the process is increased to 8 hours.

On the journalism side, the problem is with breaking news. With the increase in video quality the time required to make the content available also increases, which is definitely not acceptable for this type of content, that needs to be made available as soon as possible.

1 – http://www.globo.com

Because an increase in production times, both in sports and journalism, was not an option, the solution was to optimize process (aka. encoding) performance. However, since entire video encoding was done in a server, in a straightforward way, the first option to reduce the processing times was to perform a hardware upgrade, replacing the existing infrastructure by most powerful machines. The problem is that even if Globo.com bought the best servers, total encoding times would not be significantly reduced. The demand for peak production times continued to be a problem.

Furthermore, replacing of all encoding hardware would have an outrageous cost, more so if we take into account that the servers would be idle for a great portion of the time. Sports events typically occur during the weekends and there is no way to predict the needs for breaking news. Although feasible, the infrastructure replacement was not a good solution.

Thus, to address this problem efficiently, we implemented a solution using the Split&Merge approach proposed in this work. We focused on reducing the total processing time, and the overall cost of the video encoding process, allowing high definition video production without significant increases in the production times. The solution is detailed by the following steps, as shown on Figure 13 as folows:

1. Content Upload to Amazon EC2: The original content, a DV format video with 25Mbps of date rate is transferred from Globo.com to Amazon EC2 Master Instance, in a shared filesystem that is mounted by each of EC2 nodes.
2. Content Encoding in EC2 using Split&Merge Approach: After content is transferred from Globo.com to EC2, a message is posted in the EC2 Master Instance, which starts the S&M video encoding, distributing the chunks across several EC2 nodes, which are started or stopped by the Master, according to processing demands.
3. Original and Encoded Video Transfer from EC2 to S3: When the encoding process is finished, the compressed video, with 500kbps of data rate, and the original video, are transferred from the EC2 Master to S3, for permanent storage.

4. S3 Storage of Original and Encoded Content: The content stored on S3 will remain stored as long as needed, or desired, by Globo.com.

5. Copy of Encoded Content from S3 to Globo.com: Finally, the encoded video is copied to Globo.com, allowing Internet video distribution using the existing infrastructure.

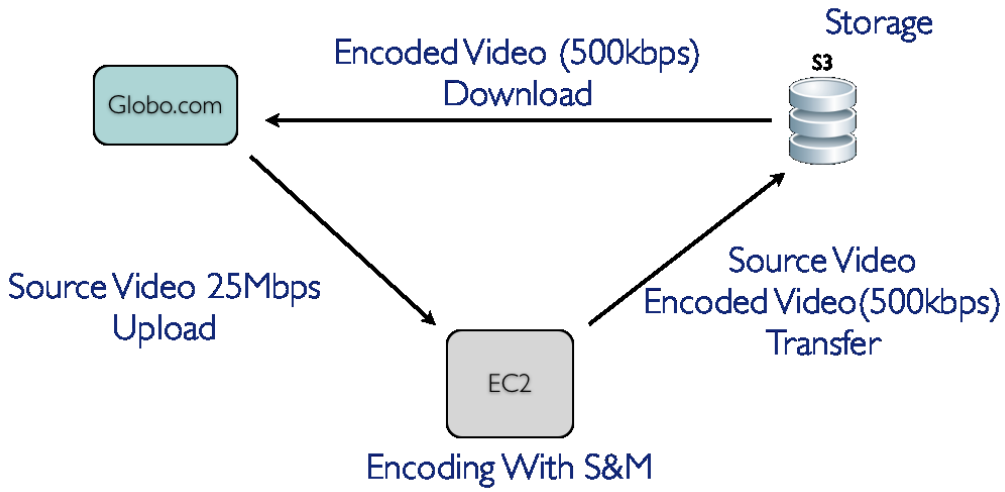

Figure 13. The Video Production approach using S&M deployed on Amazon AWS

It is possible to calculate to total cost of each step, as well as for the whole process, in order to compare it to the alternative of doing an infrastructure upgrade. For this calculation, we considered an average production of 500 minutes of video per day, and, one video encoding per video produced, which means that only one format is generated in the output. Table 4 bellow shows a detailed analysis:

Table 4. Cost of S&M approach deployed in Amazon AWS for Globo.com's case

Content Upload to Amazon EC2 (DV 25Mbps)	$9.15
Content Encoding in EC2 using Split&Merge Approach	$7.58
Original and Encoded Video transfer from EC2 to S3	$25.23
S3 Storage of Original and Encoded Content	$14.00
Copy of Encoded Content from S3 to Globo.com	$0.30
Total Cost per Day (for 500 minutes of video)	***$56.26***
Total Cost per Year	***$20,534.90***

It is interesting to note that the total processing cost using this approach in an entire year is less than the cost of a single server, without considering the costs associated to power, cooling, maintenance, operations, security and other. This result means that the proposed approach is 100 times, or more, cheaper than the infrastructure upgrade, since the encoding farm has more than 100 servers.

Furthermore, if a new format needs to be produced, for the same original content, the total cost will increase only by $1.00 per day for each additional format, and without increasing production times, as more EC2 nodes for chunk encoding could be engaged as needed.

These results show us that the proposed Split&Merge approach, deployed in Amazon AWS platform, works towards reducing costs as well as processing times for Internet video encoding. However, as a general video processing architecture, it could be used for several different video processing applications, as we exemplify in the next section.

5.2 The Split&Merge for Video Event Extraction using OCR

The ability to automatically extract information about events in sports videos, e.g. the moment when one team scores a goal, faults, player exchange, as well as additional information such as team, player and stadium names, is extremely interesting in situations where no additional data or is associated with the video. This information can be employed to make useful annotations that can help improve indexing, storing, retrieval and correlating video files for future use.

Usually, the process of video data extraction is done manually, where each video is watched and annotated according to the data seen by a person. However, data extraction and annotation of large amounts of video makes this process slow and costly, and, in many cases, it may turn it unfeasible. Therefore, developing algorithms that can (partly) reproduce the human capacity to extract and classify information is extremely relevant.

Sports videos are very good candidates for automatic information extraction because there is often several bits of relevant information on the video itself. Figure 14 exemplifies a video frame that contains information that can be identified and extracted for content annotation. However, to obtain satisfactory results, it is necessary to apply several image-processing techniques for character recognition, to each individual frame, making this task a very complex one.

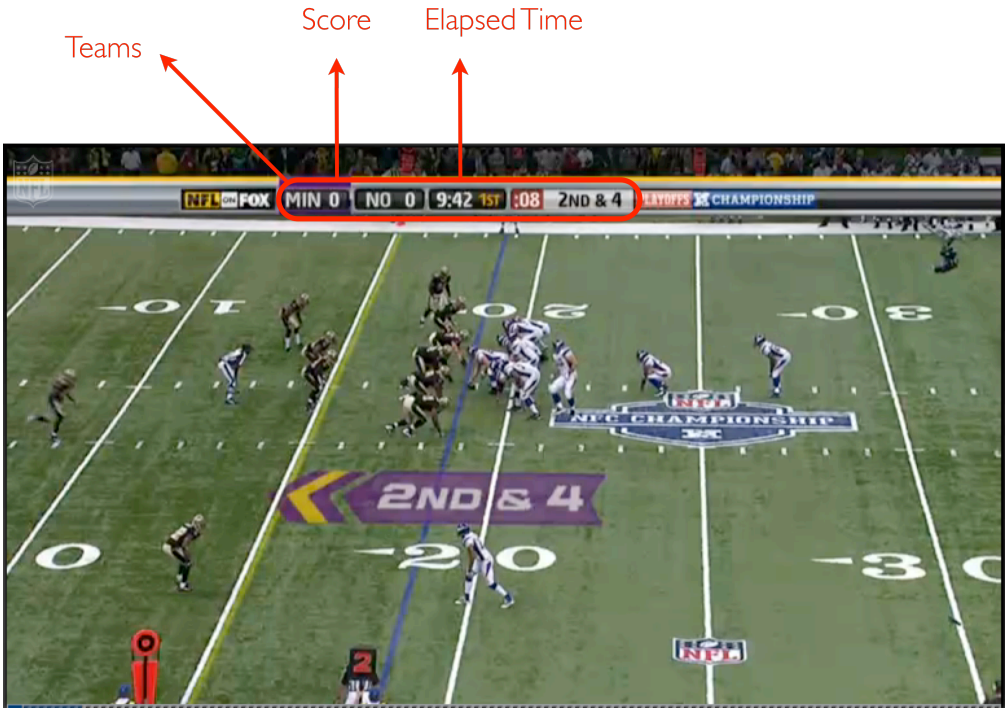

Figure 14. Information inside a Sports Video

We would like to call attention to the fact that video and image processing are computationally very expensive processes, i.e., they require a great amount of processing power, as well as large storage resources. Solutions to reduce processing times, and improving the rational use of available resources are thus very important. Motivated by this scenario, we experimented with the Split&Merge approach to extract events from sports videos automatically. Our goal is to make it a timely and efficient process, by using dynamic resource provisioning provided by Cloud services. This represents a huge competitive advantage because, unlike the traditional process, it is possible to process very large videos, that would typically take several hours to process, within minutes (a fixed amount of minutes, for that matter).

To implement this project we made use of different supporting technologies, that had to be combined and extended to obtain the desired goal. Because development and improvement of OCR algorithms is not the focus of this study, we used a combination of existing tools to perform the character recognition tasks. We chose a well-known, free optical character recognition engine, Tesseract [26], which provides character accuracy greater than 97% [27], and is currently developed by Google. We combined it to ImageMagick [28], a tool that pre

processes video frames, cropping out irrelevant information, and transforming the video frame to monochrome with white background, to increase OCR efficiency.

In addition, to allow for the parallel and distributed processing of video frames across multiple nodes, we used CloudCrowd [29], a Ruby framework that implements a scheduler that delegates tasks to different nodes, and obtains the status from them. This tool was used to implement a queue controller responsible for scheduling tasks in a distributed environment.

Finally, to allow for the implementation of an elastic architecture, capable of scaling up according to demand, and being deployed in a public Cloud service, we used the Spit&Merge architecture on to of Amazon Web Services platform (AWS) [8, 9] that served as the Cloud infrastructure provider. It is important to note that, in addition to providing processing resources on demand, the AWS platform also provided distributed and fault-tolerant storage as well as a relational database service.

We begin by detailing the sampling technique used for reducing processing times, implemented using the Split&Merge architecture [30, 31, 32], as illustrated in Figure 15.

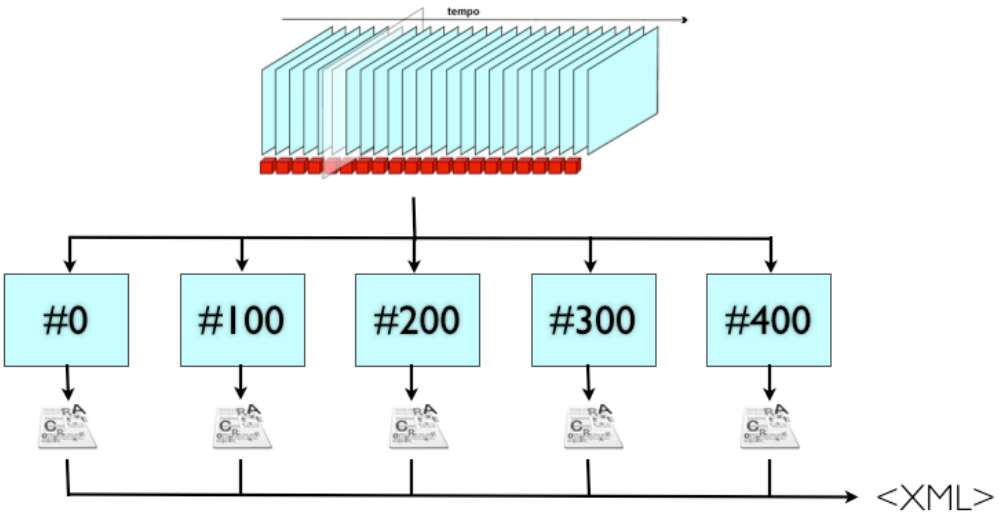

Figure 15. The Split&Merge for Video Event Extraction using OCR

In order to eliminate redundant information, and significantly reduce the amount of information to be processed, we sampled some video frames. Sampling is only possible in situations where the difference of information between

subsequent frames is minimal. In the case of sports videos, at 30 frames per second, we extracted only one frame per second, which proved enough to identify the desired information. Then, each sampled video frame was processed. When relevant information was identified, we applied OCR techniques. As a result, for each individual video frame, we extracted a certain amount of information (e.g the teams, score, elapsed time, among others), as shown in **Error! Reference source not found.**.

Figure 16. Data extraction process

For the initial tests we selected 5 different sequences of high-definition 720p soccer videos, from different matches and with different duration, encoded with MJPEG 30Mbps, 29.97fps, and audio PCM/16 Stereo 48kHz. In these videos, we wanted to identify which were the teams involved in the match, what the final score was, and when a goal was scored. We also wanted to discover which was the sample frequency that offered the best cost-benefit in terms of precision and time required for identification.

For a more realistic scenario, the videos were chosen from different matches, and with short (2 minutes or less) and long (10 or more minutes) durations. In addition, the score did not appear throughout the content duration, being displayed only at certain moments. About the OCR engine, we use Tesseract 3 without any training, only with a specific dictionary containing each possible acronym for a team.

To validate the impact of sampling on processing time and accuracy, each video was processed several times, each time with a different sampling rate: one with 1 frame every 1 second, 1 frame every 2 seconds, every 5 seconds, 10

seconds, 15 seconds, 20 seconds and 30 seconds, so that, the smaller is the sampling interval, the greater is the amount of frames to be analyzed.

Figure 17, as follows, shows how the sampling rate influences the efficiency of the extraction, i.e., the algorithm's capacity to identify and extract a score in a frame where the score is displayed. Frames without a score were not taken into account.

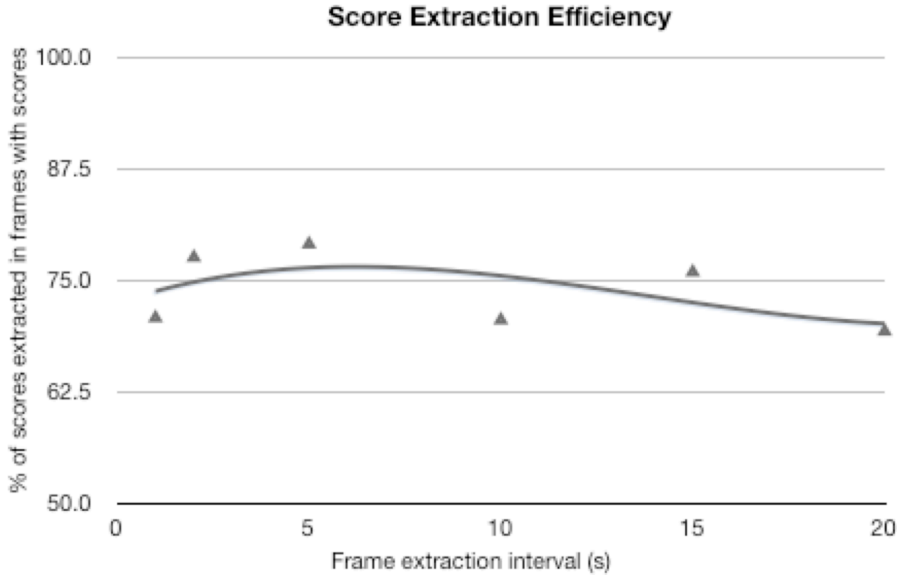

Figure 17. Efficiency in the extraction process for different sampling rates

Note that the extraction efficiency is around 75%, independently of the sampling rate. That is to say that three scores are successfully extracted on each four occurrences. For example, in a two minute video, sampling 1 frame every second, there are 120 frames to analyze. If 100 frames of the 120 have some score displayed, the proposed algorithm can extract 75 scores from these frames, i.e., in 75 frames the OCR process returns relevant information. Also note that increasing the sampling interval, the extraction efficiency suffers a slight reduction, and it could be a consequence of the small number of frames to analyze. In fact, the extraction efficiency should not present great variations when using different sampling rates, since each frame is isolated processed.

On the other hand, the algorithm's capacity to extract the information correctly is directly related to the number of frames analyzed. With more frames, the greater will be the information available to make a better decision. With lower

information volume, the extraction becomes guessing, as the algorithm chooses the highest probability option.

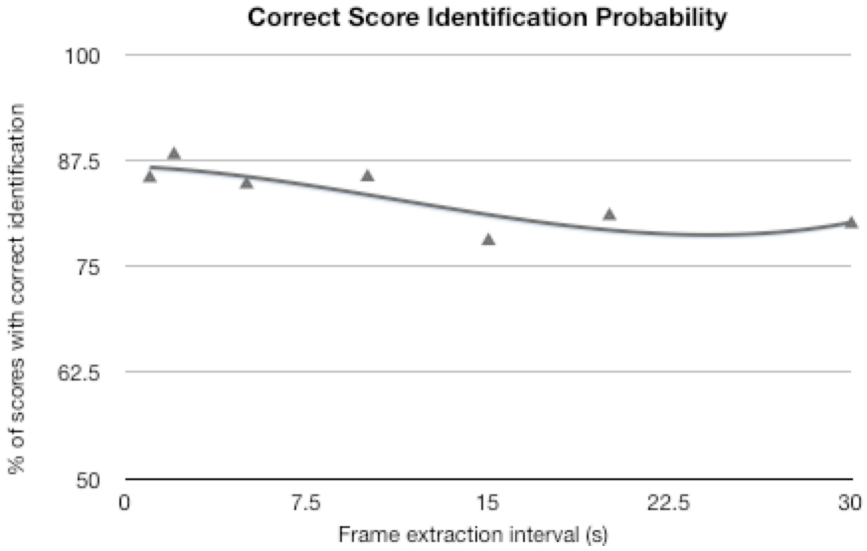

Figure 18. The Probability of a Correct Identification by the OCR Engine

As expected, Figure 18, above, shows that with lower frame sampling intervals the probability of correct score identification increases, and stabilizes around 87%. This result means that in 87% of extracted scores, the output of OCR process is correct, which is directly consistent with Tesseract's engine efficiency. This value, in fact, is obtained for one isolated frame, an not for the full video analysis. This means that the efficiency of a video data extraction could be greater if we consider a hole set of frames and use them to perform a correction in eventual identification errors. It is also important to remind that some OCR tools, such as Tesseract, could be trained to increase the efficiency in the OCR process, and, in this first prototype, we didn't perform such training.

One important remark is that, since the desired extraction output is the name of teams in the match, the final score, and the name of the team that scored the goal, obtained through a composition of all OCR outputs, for this test set and for all tested sampling rates the returned information is correct.

To demonstrate the advantages brought forth by the proposed architecture we compare the cloud implementation to results using the traditional process (process all frames in a single server). Figure 19 shows the times, measured in seconds, required for the processing of different number of frames, with always

one worker per frame (in the cloud environment), using the proposed Split&Merge implementation (darker line), and the ones using the traditional process (gray line).

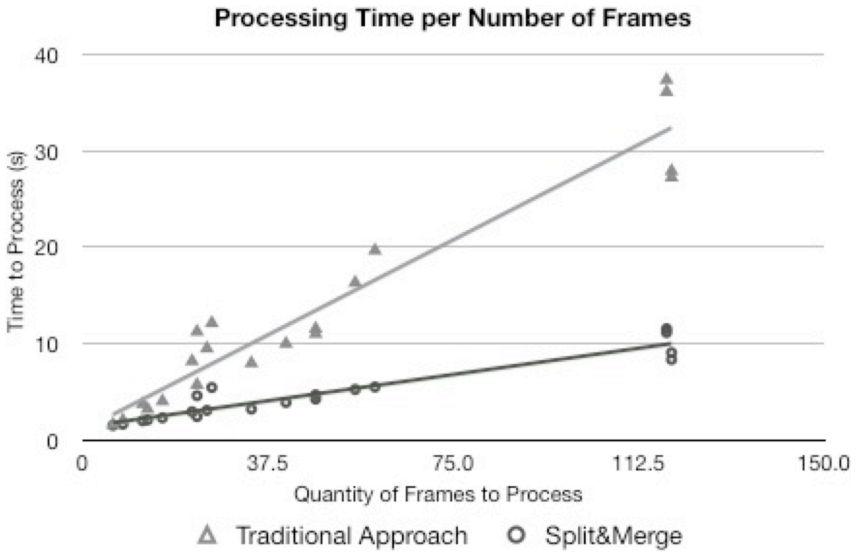

Figure 19. Total Processing Times for Different Number of Workers

Note that the Split&Merge implementation requires only 30% of the total time spent using the traditional process, which is extremely interesting for applications where time-to-market is vital. Indeed, as neither the split nor the merge steps can be parallelized, its execution time will be independent of the amount of active workers, and will be responsible for the increasing in processing time for the Split&Merge approach.

To provide an idea of cost savings, we contrast the Split&Merge approach when deployed in the public Cloud, against the costs of having a private infrastructure dedicated to this task. Taking into account the results presented in Figure 19, we have an approximate cost of $0.0007 per minute of video, considering a 1 second sampling interval, to get the results 3 times faster, using the Amazon AWS platform, with the additional advantage that it is possible to process very large videos in a few seconds. The total cost shows that the architecture of Split&Merge deployed in the public Cloud is not only efficient in terms of processing time, but also in deployment and operation costs.

Considering an optimal situation where there are unlimited resources available, it is possible to use the experimental results to predict the total cost and number of nodes needed to process videos of different durations. Table 5, bellow,

compares the traditional process with the proposed Split&Merge(S&M) approach. In this example the goal is to get the total process time 5 times faster using S&M. We are also using the cost per minute, although Amazon's minimum timeframe is one full hour, considering scenarios where there are a great number of videos to be processed, so, that machines are not shut down after a single process.

Table 5. Comparison between the Traditional Process and the Split&Merge approach, for 1 second of sampling interval

Input Video Duration	*Traditional Process Duration*	*S&M Process Duration*	*Number of S&M Nodes*	*Normalized S&M Cost Using EC2 (in US dollar)*
30 sec.	9 sec.	3 sec.	30	$0.0003
5 min.	9 min.	2 min.	300	$0.005
30 min.	53 min.	13 min.	1800	$0.029
2 hour	3.5 hour	41 min.	7200	$0.103

Note that the Split&Merge approach, deployed in a public Cloud, reduces the total processing time for a 2-hour video from 3.5 hours to 41 minutes, with the total processing cost of $0.103. However, it is possible to instantiate more workers, reducing the total processing time even more, to just a few seconds, and for the exact same price. In this case, the total reduction in processing time will be limited to the necessary time to perform the split and the merge steps.