CAMEO: Continuous Analytics for Massively Multiplayer Online Games on Cloud Resources

Alexandru Iosup

Electrical Eng., Mathematics and Computer Science Department
Delft University of Technology, Delft, The Netherlands
`A.Iosup@tudelft.nl`

Abstract. Massively Multiplayer Online Games (MMOGs) have grown to entertain tens of millions of players daily. Currently, the game operators and third-parties using gameplay information rely on pre-provisioned resources to analyze the current status of the player community and the evolution of this status over time. Instead, with the appearance of cloud computing it has become attractive to use on-demand resources to run automated MMOG data analytics tools. Thus, in this work we introduce CAMEO, an architecture for Continuous Analytics for Massively multiplayEr Online games on cloud resources. Our architecture provides various mechanisms for MMOG data collection and continuous analytics of a pre-determined accuracy in real settings. We assess the capabilities of our approach by taking and analyzing complete or partial snapshots from Runescape, one of the most popular MMOGs with a community of over 3,000,000 active players. Notably, we show evidence that CAMEO already supports simple continuous MMOG analytics, and give a first estimation of the costs of the analytic process.

1 Introduction

Massively Multiplayer Online Games (MMOGs) gather tens of millions of players into a fractioned online community. To serve the interests of these players, the game operators and the third-party entities such as community and fan-owned web sites need to collect, analyze, and then synthesize the status of the community components. While the final synthesis may differ from entity to entity, the data collection and analysis (collectively, the *game analytics*) can benefit from recent advances in the availability of on-demand resources through cloud computing services such as Amazon's Elastic Compute Cloud (EC2). In this work we present CAMEO, an architecture for continuous analytics of data taken from massively multiplayer online games on cloud resources.

Online data crawling has often been employed in the past to determine the stationary and dynamic characteristics of Internet-based communities. However, the focus of the research community has been either in making the crawling process more parallel [1–3], or analyzing the acquired data using more scalable parallel or distributed algorithms [4,5]; both these approaches assume that enough resources are available for the task. In contrast, in this work we focus on

---

[1] We thank the Delft ICT Talent Grant for the financial support.

a domain-specific application, MMOGs, and focus on a different problem which stems from a restricted resource availability (which in turn is the direct result of minimizing costs): continuous analytics of a pre-determined accuracy in real settings. Our contribution is threefold:

1. We present a first formulation of the problem of continuous analytics for MMOGs (Section 2);
2. We introduce CAMEO, an architecture for continuous analytics of data taken from massively multiplayer online games that uses cloud computing environments to dynamically obtain resources(Section 3);
3. We show that CAMEO can be used to acquire and track data from Runescape, a popular MMOG, and give a first cost estimation for this process (Section 4).

2 Continuous Analytics for MMOGs

In this section we present the problem of continuous analytics for MMOGs.

2.1 Definition

MMOGs generate data that need to be analyzed at various levels of detail and for various purposes, from high-level analysis of the number of players in a community for in-game reward allocation to the detailed analysis of the user mouse clicking behavior for audit and cheat detection. Usually, a replica of the data to be analyzed needs to be created, which raises the problem of maintaining consistency between the original and the replica(s). Similar to other cases of information replicas in distributed systems, creating exact copies of the data for analysis purposes may not be only expensive, but also unnecessary [6]. Instead of ensuring that the replicas are strongly consistent, our goal is to maintain information replicas whose difference is bounded and the bound is under the control of the analyst. This goal stems from traditional work on continuous consistency of information replicas with deviation in the staleness of information [6] and quasi-copying [7].

We can now define *continuous analytics for MMOGs* as the process through which relevant MMOG data are analyzed in such a way that prevents the loss of important events affecting the data. The relevance of the data is application-specific, as it depends on the target of the analysis. Similarly, the important events allow for the information replicas to be loosely consistent with the original, within application-specific bounds.

2.2 Challenges

Every data analysis process includes data collection, storage, processing, and presentation, each of which raises generic challenges in supporting continuous MMOG analytics. We focus here only on the MMOG-specific challenges, challenges due to data characteristics, and challenges due to data ownership, which we describe in turn.

MMOGs pose unique data scale and rate challenges. MMOGs generate and manage massive amounts of information; for example, the database logging user actions for Everquest 2, a popular MMOG, stores over 20 new terrabytes (TB) of data per year. Other projects such as CERN's Large Hedron Collider or the Sloan Digital Sky Survey produce data orders of magnitude larger than MMOGs, but these projects are using large and pre-provisioned (expensive) computational and data infrastructure that game companies cannot afford. Furthermore, the data production rate for these other projects is stable over time spans of days or even weeks, whereas for MMOGs the daily user activity has peaks and may even change hourly [8].

MMOGs pose unique data ownership challenges. MMOGs often involve multiple companies in their design-development-distribution-use process; each of these companies may have different commercial interest and thus compete for generating and managing game-related data. Moreover, there may be many types of data users, from audit companies who should access all data to fan communities that may only be allowed to access information open to everyone. Thus, MMOGs raise data access challenges. Other commercial applications, notably financial and government public relations services, face similar problems. However, in contrast to MMOGs these services produce data for entities that can afford expensive data collection and processing infrastructure, such as brokering agencies or news corporations.

2.3 Applications

There are many applications for continuous MMOG analytics, both for the gaming industry and for other domains. We describe the most important such applications in the following.

Within the gaming industry, the main applications are to audit the process of each company involved with the MMOG, to understand the play patterns of users and support future investment decisions, to detect cheating and prevent game exploits, to provide user communities with data for ranking players, to broadcast gaming events, and to produce data for advertisement companies and thus increase the revenue stream for the MMOG owners.

In other areas, by domain the applications may include studying emergent behavior in complex systems (systems theory), understanding the emergence and evolution of the contemporary society [9] (social sciences) and economy [10] (economics), uncovering the use of MMOGs as cures and coping mechanisms [11] (psychology), investigating disease spread models [12] (biology), etc.

3 The CAMEO Architecture

In this section we present the CAMEO architecture for continuous MMOG analytics. The CAMEO architecture is built around the idea of enabling continuous MMOG analytics while using resources only when needed. To achieve this goal, it acquires and releases computational and storage resources dynamically from cloud computing environments such as Amazon's EC2+S3.
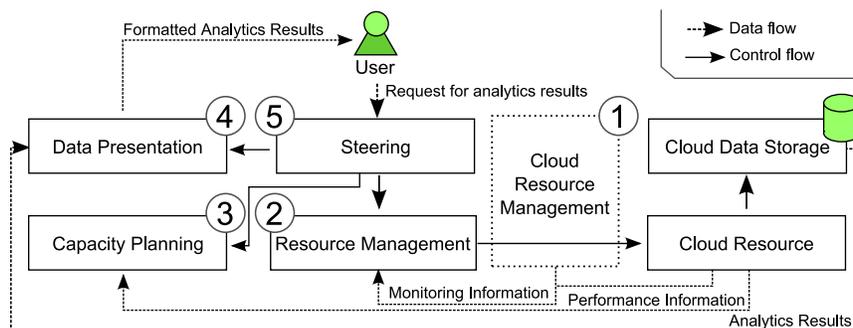
Fig. 1. The CAMEO architecture.

3.1 Overview

The five main components of the CAMEO architecture are depicted in Figure 1. The *Cloud Resource Management* component (component 1 in Figure 1) provides access to the computational and storage resources of the cloud computing environment, and is maintained by the cloud owner. The *Resource Management* component (#2) acquires and releases resources from the cloud and runs the analytics applications. It also uses the monitoring information provided by the cloud resource management and the resources as input for further management actions, such as transparent fault tolerance through application instance replication. The *Capacity Planning* component (#3) is responsible for deciding how many resources must be acquired for the analytics process. The decisions are based on the system's capability to produce results, analyzed during the course of the analytics process, and on the accuracy and cost goals of the process. The *Data Presentation* component (#4) formats and presents the results of the analytics process to the user. The *Steering* component (#5) is responsible for coordinating the analytics process. Towards this end, it takes high-level decisions, expressed through the configuration of each other's component process.

Except for the use of cloud computing resources, our architecture uses a traditional approach. However, the components have unique features specific to the targeted application. We describe in the remainder of this section three distinctive features of CAMEO.

3.2 Resource Management Mechanisms

The triggering of the analytics process depends on the nature of the application and on the system status. On the one hand, the nature of the application may allow the system analyst to design a stable analysis process such as a daily investigation of the whole community of players. On the other hand, special analysis may be required when the system is under unexpectedly heavy load, or when many players are located in the same area of the virtual world. To address this situation, we design the Resource Management component to provide two mechanisms for using cloud resources: one static and one dynamic. The *steady*

analytics[2] mechanism allows running a periodic analytics operation on cloud resources. The *dynamic analytics* mechanism allows running a burst of analytics operations on cloud resources. Optimizing the allocation of resources for static analytics or for mixed static-dynamic analytics is a target for this component, but beyond the scope of this work. Similarly, the case when the cost of data transfers is significant, that is, similar or higher to the cost of the computational resources, is left for future work.

3.3 Steering through Snapshots of Different Size

The analytics process includes collecting the necessary information from the data source. The collection results in a *snapshot*, that is, a read-only dataset which has been extracted from the original data. We further call *complete snapshot* a snapshot that includes data for all the players managed by the MMOG, and contrast it to a *partial snapshot*. Taking snapshots complies with the continuous analytics definition introduced in Section 2.1.

Depending on the goal of the analysis, it may be possible to obtain meaningful results through continuous analytics based on partial snapshots; for example, when the goal is to obtain statistical information about the player community it may suffice to continuously analyze a randomly chosen group of players of sufficient size. We design the Steering component to be able to perform a two-step analytics process in which first complete snapshots are taken from the system with low frequency, and partial snapshots are acquired often.

3.4 Controlling the Process

The taking of a snapshot has a certain duration, which depends on the performance of the cloud resources and also on the limitations set by the owners of the original data; to prevent denial-of-service attacks and to improve scalability with the number of requests, it is common for the data owners to limit the network bandwidth available for an individual resource (IP address).

Assume that a single machine can acquire a new snapshot every T time units (seconds). Then, we can achieve linear scaling (to a certain degree) in the number of acquired snapshots by installing new machines; K machines can acquire K snapshots every T time units. We can then control either how many snapshots we acquire every T time units, or the minimal performance that has to be delivered by each machine to acquire exactly one snapshot every T time units.

4 Experimental Results

In this section we show evidence that our approach (and CAMEO implementation) can be used for continuous MMOG analytics. (Analyzing the results of a continuous MMOG analytics process falls outside the scope of this work.)

---

[2] We do not use the term "static" to underline that this is a continuous process.

Table 1. The resource characteristics for the instance types offered by Amazon EC2.

Resource Type	Cores (ECUs)	RAM [GB]	Architecture [bit]	I/O Performance	Disk [GB]	Cost [$/h]
m1.small	1 (1)	1.7	32	Med	160	0.1
m1.large	2 (4)	7.5	64	High	850	0.4
m1.xlarge	4 (8)	15.0	64	High	1,690	0.8
c1.medium	2 (5)	1.7	32	Med	350	0.2
c1.xlarge	8 (20)	7.0	64	High	1,690	0.8

4.1 Experimental Setup

The Analyzed MMOG Using CAMEO, we have taken and analyzed several complete snapshots of the state of Runescape over a period of one and a half years. We have also also taken partial snapshots of the state of Runescape in quick succession, which enables us to study in the future the dynamics present in the Runescape community. We have written application-specific web crawlers for the data collection process.

The Platform We have used Amazon EC2 resources to acquire and process Runescape data. The EC2 user can use any of the five resource (*instance*) types currently available on offer, the characteristics of which are summarized in Table 1. An ECU is the equivalent CPU power of a 1.0-1.2 GHz 2007 Opteron or Xeon processor. The theoretical peak performance can be computed for different instances from the ECU definition: a 1.1 GHz 2007 Opteron can perform 4 flops per cycle at full pipeline, which means at peak performance one ECU equals 4.4 gigaflops per second (GFLOPS). Throughout the experiments conducted for this work we have used the `m1.small` instances; extending the Capacity Planning module with the ability to use multiple instance types is left as future work.

4.2 Analytics Results

Using CAMEO, we analyzed the skill level of millions of RuneScape players, which shows evidence that CAMEO can be used for measurements several orders of magnitude larger than the previous state-of-the-art [13]. CAMEO collected in August 2008 official skill level data for 2,899,407 players[3], of which 1,817,211 (over 60%) had a skill level above 100; the maximum skill level is 2280. The values for players with skill level below 100 include application-specific noise (mostly starting players) and are therefore polluted. Thus, we present here only data for all players with skill above 100, and for a single measurement. Figure 2 depicts the overall skill level of RuneScape players, with bins of 100 levels. The number of players per bin is well characterized by a skewed normal-like distribution; the majority of the players are of average skill or below, the most populated skill level bins are those corresponding to the middle skill level, and the number of high-level players is significant. We have explored the implications of this skill level distribution in our previous work on automatic content generation [14].

---

[3] The current population of Runescape has increased to over 3,000,000 active players.
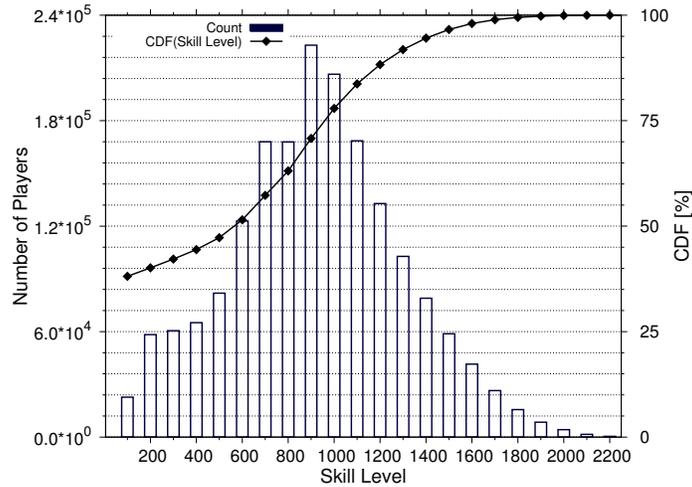
Fig. 2. Pareto graph, that is, combined PDF (left vertical axis) and CDF (right vertical axis) depiction of the skill level of the RuneScape player population. Each bar represents a range of 100 levels. CDF stands for cumulative distribution function; $CDF(x)$ is the total number of players with skill level up to and including x. Note that the left vertical axis is not linear. See text for why the CDF of the skill level does not start at 0%.
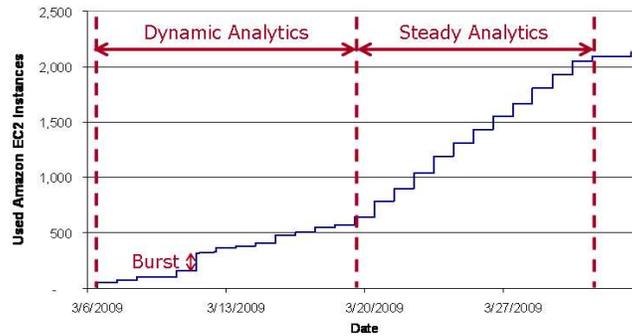
Fig. 3. Resource consumption in the two analytics modes: dynamic and static.

4.3 Resource Management

To demonstrate the capability of CAMEO to perform both dynamic and steady analytics, and to monitor the process, we show in Figure 3 the evolution of the cumulative number of consumed CPU hours over time. The dynamic analytics are based on uneven bursts of activity, of which the burst during March 10 is the most prominent. The steady analytics part of the experiments reveals an even use of resources over time, with the steps indicating a new work cycle.

One of the contributions of this work is getting a first estimation on the cost of continuous MMOG analytics. Figure 4 shows the total cost incurred by the continuous analytics process over the course of one month. For this simple analysis process, which acquired partial snapshots and only browsed the data in memory during the processing phase, the cost is below $500 per month. It is

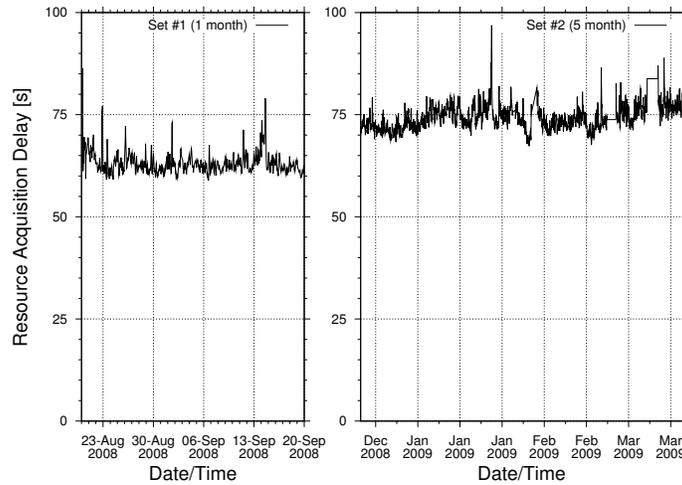**Fig. 4.** Putting a cost on continuous analytics for MMOGs.

Fig. 5. Evolution of the VM Install time in EC2 (hourly average) over six months (two data sets collected from `CloudStatus.com`).

not our intention to argue that the cost of continuous analytics for an MMOG can be this low; much more complex analytics taking many more computational hours are performed for any of the applications presented in Section 2.3.

4.4   Platform Capabilities

An important assumption in our work is that resources can be acquired on-time, that is, that whenever resources are requested by the Resource Management component of CAMEO the cloud will provide them within a reasonable time. We now show that this is indeed the case.

We have made initial install time measurements in August 2007, and found that the average install time was steady around 50s [15]. To understand the long-term evolution of the install time in EC2, we have obtained the measurement

results published online by the independent CloudStatus team [16]. We have
written web crawlers and parsing tools and taken samples every two minutes
from August until October 2008 (set #1, two months), and from December 2008
until May 2009 (set #2; only the first four months are depicted in Figure 5).
Figure 5 shows that the install time fluctuates by around 5 seconds within short
time intervals (days), but that the average install time has increased from 50s in
August 2007, to 64s in August 2008, and to 78s in April 2009. This indicates a
doubling of the rate of the increase in install time every half year. If this trend
continues, conservatively the install time will reach 80s in 2009 (confirmed), and
around two minutes by June 2010. We leave as future work a detailed study of
the time patterns that may occur in the install time, e.g., effects of the hour-of-
the-day and of the day-of-the-week. We conclude that the resource acquisition
time is steady within a short period of time (hours, days) and has a slow yearly
increase. An investigation of the storage capabilities of the Amazon cloud [17]
allows us to reach the conclusion that the use of cloud resources for continuous
MMOG analytics is possible even for bursts of user activity.

5   Related Work

We have already discussed the main differences between our work and generic
web crawling approaches [1–3] and parallel or distributed analytics [4, 5]. In
contrast with this body of previous research, ours focuses on a more restricted
application–albeit with millions of users– but focuses on using (and paying for)
the resources used in the analytics process only when they are needed.

Closest to our work, Provost and Kolluri [4] examine many basic techniques
for scaling up inductive algorithms. While data analytics (as a superset of induc-
tive algorithms) has evolved considerably in the decade passed since this survey,
the problem of continuous MMOG analytics raises new challenges, and our ap-
proach is based on using on-demand resources (cloud computing) instead of a
fixed computational platform.

6   Conclusion and Future Work

The growing world of Massively Multiplayer Online Games (MMOGs) raises
important derivative online applications and interesting new challenges to the
distributed computing community, including the problem of massive game data
analytics. Motivated by a subset of this problem, in this work we have introduced
CAMEO, an architecture for continuous analytics of data taken from massively
multiplayer online games on cloud resources. Using a reference implementation
of CAMEO and resources leased from the Amazon EC2 cloud, we have taken
complete and partial snapshots of the Runescape multi-million player commu-
nity for a period of over eighteen months. Our results give evidence that cloud
computing resources can be used for continuous data acquisition and analysis.
Furthermore, we have devised within CAMEO mechanisms for controlling the

analytics process, including taking partial snapshots of a given size, whose analysis leads to a pre-determined accuracy of the results. Last, we have provided a first cost estimation for the continuous analytics process.

For the future, we plan to investigate in more detail the trade-off between the amount of data acquired and the quality of the analysis results for MMOGs. We will also investigate the use of more heterogeneous resource types coming from one or more clouds, and the restricted use of cloud resources when local resources are available.

References

1. A. Arasu, J. Cho, H. Garcia-Molina, A. Paepcke, and S. Raghavan, "Searching the web," *ACM Trans. Internet Technol.*, vol. 1, no. 1, pp. 2–43, 2001.
2. J. Cho and H. Garcia-Molina, "Parallel crawlers," in *WWW*, 2002, pp. 124–135.
3. H.-T. Lee, D. Leonard, X. Wang, and D. Loguinov, "Irlbot: scaling to 6 billion pages and beyond," in *WWW*, 2008, pp. 427–436.
4. F. J. Provost and V. Kolluri, "A survey of methods for scaling up inductive algorithms," *Data Min. Knowl. Discov.*, vol. 3, no. 2, pp. 131–169, 1999.
5. R. J. Bayardo, Y. Ma, and R. Srikant, "Scaling up all pairs similarity search," in *WWW*, 2007, pp. 131–140.
6. H. Yu and A. Vahdat, "Efficient numerical error bounding for replicated network services," in *VLDB '00: Proceedings of the 26th International Conference on Very Large Data Bases.* San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2000, pp. 123–133.
7. R. Alonso, D. Barbará, and H. Garcia-Molina, "Data caching issues in an information retrieval system," *ACM Trans. Database Syst.*, vol. 15, no. 3, pp. 359–384, 1990.
8. V. Nae, A. Iosup, S. Podlipnig, R. Prodan, D. H. J. Epema, and T. Fahringer, "Efficient management of data center resources for massively multiplayer online games," in *ACM/IEEE SuperComputing.* IEEE/ACM, 2008.
9. C. Steinkuehler and D. Williams, "Where everybody knows your (screen) name: Online games as "third places"," in *DIGRA Conf.*, 2005.
10. E. Castronova, "On virtual economies," *Game Studies*, vol. 3, no. 2, 2003.
11. D. Williams, N. Yee, and S. Caplan, "Who plays, how much, and why? debunking the stereotypical gamer profile," *Journal of Computer-Mediated Communication*, vol. 13, no. 4, pp. 993–1018, September 2008.
12. BBC NEws, "Virtual game is a 'disease model'," News Item, Sep 2009, [Online] Avilable: http://news.bbc.co.uk/2/hi/6951918.stm.
13. N. Yee, "The demographics, motivations, and derived experiences of users of massively multi-user online graphical environments," *Presence*, vol. 15, no. 3, pp. 309–329, 2006.
14. A. Iosup, "POGGI: Puzzle-based Online Games on Grid Infrastructures," in *Euro-Par*, ser. LNCS, 2009, pp. 390–403.
15. A. Iosup, T. Tannenbaum, M. Farrellee, D. H. J. Epema, and M. Livny, "Interoperating grids through delegated matchmaking," *Scientific Programming*, vol. 16, no. 2-3, pp. 233–253, 2008.
16. The Cloud Status Team , "JSON report crawl," Jan. 2009, [Online]. Available: http://www.cloudstatus.com/.
17. M. R. Palankar, A. Iamnitchi, M. Ripeanu, and S. Garfinkel, "Amazon S3 for science grids: a viable solution?" in *DADC '08: Proceedings of the 2008 international workshop on Data-aware distributed computing.* ACM, 2008, pp. 55–64.