SLA-based Operation of Massively Multiplayer Online Games in Competition-based Environments[*]

Vlad Nae, Radu Prodan
Institute of Computer Science
University of Innsbruck
Technikerstr. 21a, A-6020 Innsbruck, Austria
radu@dps.uibk.ac.at

Alexandru Iosup
Dept. of Software and Computer Technology
Delft University of Technology
Mekelweg 4, 2628 CD, Delft, Netherlands
A.Iosup@tudelft.nl

ABSTRACT

To sustain the variable load of Massively Multiplayer Online Games (MMOGs) with guaranteed Quality of Service (QoS), game operators over-provision a static infrastructure capable of sustaining the peak load, even though a large portion of the resources is unused most of the time. This inefficient way of provisioning resources has negative impacts, leading to inefficient resource utilisation, high service prices, and limited market participation accessible only to the large companies. We propose a new ecosystem and model for hosting and operating MMOGs based on cloud computing principles involving four smaller and better focused business actors whose interaction is regulated through Service Level Agreements (SLAs): resource provider, game operator, game provider, and client. In our model, game providers lease operation SLAs from the game operators to satisfy all client requests and manage multiple distributed MMOG sessions. In turn, game operators efficiently lease on-demand cloud resources based on the dynamic MMOG load and ensure proper game operation that maintains QoS to all clients. In this paper, we focus on the business interaction between the game provider and the game operator by defining the SLA terms and the underlying negotiation protocol, including a model for compensations for QoS violations. We propose a method for ranking operational offers based on price, compensation and resource fitness, and study its impact on game provider's profit in an environment with several providers competing for SLAs from multiple game operators.

Categories and Subject Descriptors

I.6.8 [**Simulation and Modeling**]: Types of Simulation—*Gaming*; K.6.2 [**Management of Computing and Information Systems**]: Installation Management—*pricing and resource allocation, performance and usage measurement*

[*] Austrian Science Fund project TRP 72-N23 funded this research.

C^3S^2E-13 2013, July 10-12, Porto [Portugal]
Copyright 2013 ACM 978-1-4503-1976-8/12/06 ...$15.00.

General Terms

Economics, Management

Keywords

MMOG, Business services, SLA, QoS, Cloud computing

1. INTRODUCTION

Online entertainment including gaming is a strongly growing sector world-wide. Massively Multiplayer Online Games (MMOG) grew from ten thousand subscribers in 1997 to eight million in 2005 and the rate is accelerating estimated to 60 million people by 2015. Today, most companies still take a double role in the MMOG life cycle: game providers by investing in the development of the creative part of the game, and game operators by purchasing and managing a large data centre required for hosting it (using up to 40% of the total game revenues in an annual market of over 24 billion dollars). The companies providing the top-five most popular MMOGs in the western market are concrete examples of such in-house MMOG development, publishing and operation (see Figure 1). For example, Blizzard (developer of World of Warcraft) and Jagex Ltd. (RuneScape) own and operate tens of thousands of cores in hundreds of physical locations across all continents (World of Warcraft has operational costs of over $50 million per year). An important reason for this approach is the lack of business models and supporting middleware to enable outsourcing the operation of MMOGs, which should include methods for specifying and negotiating operational terms, responsibilities, and risk-related compensations. This approach has an important downside, the high initial investment in purchasing and running the data centres required to join the MMOG market.

Cloud computing promises eliminate the burden of permanent over-provisioning through on-demand resource leasing under cost and sometimes performance-driven *Service Level Agreements (SLAs)*. By leveraging this new cloud infrastructure model, companies may avoid the large costs of buying and maintaining depreciable hardware, and can enter the MMOG operation market with nearly-zero initial investment. We have tackled in [8, 13, 15] many of the technical challenges of on-demand provisioning and allocation of cloud resources to MMOGs under Quality of Service (QoS) constraints [18], however, commercial clouds still cannot readily be used for MMOG operation, as their SLAs mostly focus on hardware characteristics and lack support for negotiating MMOG-friendly SLAs. Furthermore, although focused

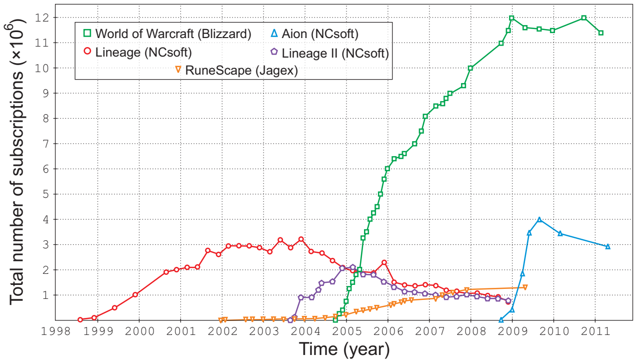


Figure 1: Top-five MMOGs in number of subscribed and concurrent users.

on infrastructure hardware, even large commercial cloud providers such as Amazon and Microsoft have experienced black-outs and variable performance over the past years. Thus, game operators cannot outsource their infrastructure services without a clear specification of responsibility and of compensations associated with risks of unavailability, lower performance, and other QoS violations.

To enable small and medium enterprises join the MMOG market, we introduce a new *MMOG ecosystem* that extends our previous work [15] with a comprehensive SLA specification and negotiation mechanism between two actors: (1) *game providers* who negotiate operation SLAs with game operators to satisfy all *clients* requests, and (2) *game operators* who efficiently lease and provision resources to MMOGs from *cloud providers* based on their dynamic load so that the required QoS parameters are maintained to all clients. Since we covered interaction between the game operator and the cloud provider in [8], we focus in this paper on the game SLA-based business interaction between the game provider and the game operator. Our proposed business model includes a comprehensive specification of the SLA terms underneath the negotiation protocol, and of compensations for temporary QoS violations. On top of this, we propose a method for ranking operational offers based on price, compensation and resource fitness, and study its impact on game provider's profit in an environment with several providers competing for SLAs from multiple game operators.

The paper is organised as follows. Section 3 presents the MMOG, business ecosystem, and QoS models underneath our approach. Section 4 describes the operational SLAs between the game provider and the game operator, followed by the negotiation protocol (including ranking of operational offers and compensations for QoS violations) in Section 5. We study in Section 6 the impact of ranking operational offers on game provider's profit in an environment with several providers competing for SLAs from multiple game operators. The paper ends with a related work summary in Section 2 and concluding remarks in Section 7.

2. RELATED WORK

We survey in this section three large bodies of related work: traditional SLA stacks for large-scale systems, SLA-based operational models for MMOGs, and cloud-based operation of services with millions of customers.

2.1 SLA stacks for large-scale systems

Much work has focused on this SLA stacks since at least the early 1980s [19], wizh two recent surveys focusing on SLA stacks for grids [20] and clouds [22]. Complete SLA stacks have been implemented, for example, by Globus [5], NextGrid [6], and SLA@SOI [4]. In contrast to these approaches, we focus on a (popular) domain-specific application, for which we extend traditional approaches with MMOG-specific considerations, propose a comprehensive SLA formalism, and investigate specific operational policies.

2.2 SLA-based operation of MMOGs

Recent work focuses on (soft) QoS guarantees for MMOG operation [3, 10, 21]. Wong [21] proposes a resource provisioning algorithm with QoS guarantees, but considers only networking aspects, whereas we consider here more resource types and a more realistic cloud model [15]. Briceno et al. [3] study resource allocation for MMOGs but, unlike our work, consider only computational requirements and use a simplified workload model (not traces from a real MMOG). Lee and Chen [10] investigate MMOG server consolidation techniques, focusing on the energy consumption; our study complements theirs. We also extend our previous work [15] with the formulation, negotiation, and use of SLAs for MMOGs. Several market-based operational models for MMOGs have been recently investigated. Our business model for MMOGs is closest in concept to the four-actor business model proposed in [12], but our work also studies the connection between the business and hosting models, and methods for controlling the provided QoS. Complex business models are proposed in [1, 2], but focus only on higher-level business interactions and goals for MMOG operation; in contrast, we study a novel operation model and its effects on the profits of both game and resource operators. Complementing our work, Nojima [16] studies the relationship between pricing models and MMOG player motivation, and Oh and Ryu [17] analyse different pricing models for the gaming service.

2.3 Cloud operation of multi-customer services

The entertainment industry has already started to migrate from the in-house to cloud-based infrastructure. Zynga, which in 2011 operated online gaming services for over 250 million users, uses Amazon EC2 resources for operating games up to several months after their launch; the games supported by Zynga require much less computational and network resources than MMOGs. On-demand gaming, which offloads gaming computation to the cloud and streams back to remote clients the video output of the game, is provided by companies such as Geelix [7], OnLive, Gaikai, and OTOY; we do not consider this game operation model in this work, because major MMOG operators have yet to switch to this model, in part because of the high network requirements imposed on the players. Since late- 2011, Amazon WS has been used for video-streaming by Netflix, and for off-loading web browsing for mobile devices with Android OS. In contrast with these approaches, our work adapts this model to the specifics of MMOGs, and proposes an in-depth study of a variety of scenarios with application to other branches of the entertainment industry.

3. MODEL

We present in this section the computational, business ecosystem, and QoS models underneath our approach.

3.1 MMOG Computational Model

Online games can be seen as a collection of networked *game servers* that are concurrently accessed by a number of players (or clients). Clients connect directly to one game server and are mapped to one avatar in the game world to whom they send their play actions and receive appropriate responses. Based on the actions sent, the avatar dynamically interacts with other avatars within a *game session*, influencing each others' state. The state update responses must be delivered within a given time frequency to ensure a smooth and responsive experience. The load of the game server is proportional to the number of interactions between entities. An overloaded game server delivers state updates to its clients at a lower frequency than the players expect which makes the overall environment fragmented and unplayable.

To concurrently support millions of active players and many more other server-driven entities (non-playing characters and other game objects) with guaranteed QoS, MMOG operators provision a large static infrastructure with hundreds to thousands of computers hosting a single distributed game session. The most common game session distribution technique is "zoning", which is based on spatial partitioning of the game world into geographical zones to be handled independently by separate machines. Other techniques, such as "instancing" and "replication", divide the entities contained in a zone across several machines.

3.2 MMOG Business Ecosystem

We propose a new ecosystem for MMOG operation and provisioning consisting of four actors: clients, game providers, game operators, and resource (cloud) providers (see Figure 2). The interaction between them is negotiated and regulated through bipartite SLAs, representing wrappers around QoS parameters which they agree to deliver (see Section 3.3). Since we introduce no change to the client and its interaction with the provider, we no longer mention it in this paper.

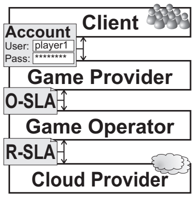


Figure 2: MMOG ecosystem.

Game providers offer a selection of MMOGs by contracting new games from development companies (we do not cover this offline interaction). Based on clients' requests, game providers assign clients to game zones delegated to game operators for QoS-based execution. The quality of game play (see Section 3.3) is monitored by and, in case of *SLA faults* (e.g. state update rate below minimum threshold), the client is compensated.

Game operators receive requests from the game providers for operating zones of different MMOG sessions with guaranteed QoS. Based on resource utilisation estimations (covered by us in [13]), the game operators construct *Operation SLA (O-SLA)* offers, negotiate SLAs with the game providers, and allocate resources accordingly (i.e. start new zones, allow client connections). We detail this interaction in Sections 4 and 5. To fulfill their agreements with game providers, game operators acquire the correct amount of resources by establishing *Resource SLA (R-SLA)* with cloud providers. We covered this interaction in [8, 15] and therefore no longer address it here. At predefined *measurement timesteps* during the game play, the game provider analyse the QoS information from the MMOG servers is analysed (see Section 3.3) and, whenever SLA faults are detected, they are compensated by the operators.

Resource providers are cloud data centres from which game operators lease computing resources and use them to run game servers with guaranteed QoS. We studied in [8] the opportunity of employing cloud infrastructures for MMOG hosting and no longer discuss it here.

3.3 MMOG QoS Metrics

A challenge in our architecture is mapping of the MMOG QoS requirements [18] to SLA terms that must be honoured at all times by the business actors. Unfortunately, this can only be enforced through best-effort resource allocation mechanisms in today's distributed computing infrastructures, which requires the introduction of compensation mechanisms in case of QoS violations. We define two important QoS metrics that characterise the quality of game play.

First, *instantaneous non-interruption ratio* represents the ratio between the measured state update frequency within one measurement timestep and the required minimal frequency. For example, if the minimal update frequency given by the game developer is 40 Hz and the measured update frequency is above 36 Hz in this measurement step, the instantaneous non-interruption ratio is: $\frac{36}{40} = 90\%$;

Second, *total non-interruption ratio* (similar to the general ITIL availability definition [11]) is the percentage of time over a given time interval (e.g. an SLA's validity time) the MMOG session has been accessible, and the state update frequency has been equal or greater than the required frequency. For example, if the game operator provided, over the last 24 hours, only 23.98 hours of game play during which the MMOG session was accessible and the state update rate was above the minimal frequency, the total non-interruption ratio is: $\frac{23.98}{24} = 99.9\%$.

4. OPERATIONAL SLAS

From the ecosystem presented in Section 3.2 and Figure 2, we focus in this paper on the interaction between the game provider and the game operator which is automatic and requires no human intervention. Based on the total number of accounts and its service policy, the game provider computes the maximum number of clients for each game zone. We define the provider's *service policy* as a quintuple:

$$\left(s_{ini}, s_{tni}, s_{time}, P^{(T)}, C^{(T)}\right) \tag{1}$$

with five terms: (1) target instantaneous non-interruption ratio s_{ini} that should be minimised, (2) total non-interruption ratio s_{tni} that should also be minimised, (3) interval of acceptable SLA validity periods s_{time} to avoid excessively long operation agreements, (4) target hourly price per client $P^{(T)}$, and (5) target compensation per client per minute $C^{(T)}$.

Using the estimated client requests (see Figure 1), the game provider negotiates the most appropriate terms for hosting each zone by establishing O-SLAs with different operators (see Section 5). Based on its policies and available cloud resources, each operator publishes an *O-SLA template*:

$$O\text{-}SLA = \left(G^{(O)}_{type}, t^{(O)}_{cli}, t^{(O)}_{ini}, t^{(O)}_{tni}, t^{(O)}_{time}, \sigma^{(O)}, P^{(O)}, C^{(O)}\right), \tag{2}$$

consisting of eight terms with either scalar or range values:

- *MMOG name* and *version* $G^{(O)}_{type}$;

- *client number* $t_{cli}^{(O)}$ (range) the game operator is ready to service;
- *instantaneous non-interruption ratio* $t_{ini}^{(O)}$ (range) representing the minimum percentage from the state update frequency the operator guarantees to maintain for all clients during the O-SLA validity period;
- *total non-interruption ratio* $t_{tni}^{(O)}$ (range) representing the percentage of QoS fulfilment from the entire O-SLA validity time and evaluated only at the end of the O-SLA validity period;
- *validity period* $t_{time}^{(O)}$ (range) representing the SLA lifetime offered by the game operator (with variable granularity from daily to semestrial);
- *geographical area* $\sigma^{(O)}$ in which the game operator will service the clients;
- *base price* $P^{(O)}$ for accepting an SLA utilising the lowest values of all terms in the given ranges;
- *compensation* $C^{(O)}$ for O-SLA faults, defined as the aggregate penalty:

$$C^{(O)} = \mathcal{P}\left(C_{cli}, C_{ini}, C_{tni}\right), \tag{3}$$

where $\mathcal{P}$ is a polynomial function the operator has to pay in case of O-SLA faults, consisting of three QoS terms: (1) compensation for unserved clients C_{cli}, (2) compensation for instantaneous non-interruption ratio C_{ini}, and (3) compensation for total non-interruption ratio C_{tni}. Although the aggregation function may carry a high weight in the operators' policies (e.g. make an expensive O-SLA template more attractive if it favours important compensation terms like C_{ini}), we consider for clarity reasons an additive function:

$$\mathcal{P}\left(C_{cli}, C_{ini}, C_{tni}\right) = C_{cli} + C_{ini} + C_{tni}\,. \tag{4}$$

We define these three terms using a *compensation function*:

$$C_x : \left[0; b_x^{(\max)}\right] \to \mathbb{R}^+,\ C_x\left(b_x\right) = \frac{c_x^{(u)} \cdot b_x}{u_x} \cdot f_x\left(\frac{b_x}{b_x^{(\max)}}\right), \tag{5}$$

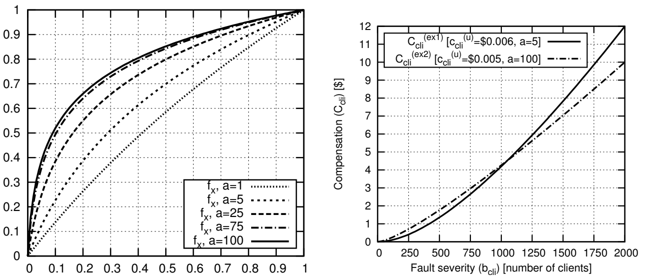
$\forall x \in \{cli, ini, tni\}$, where $\mathbb{R}^+$ denotes the set of positive real numbers, $c_x^{(u)}$ is the compensation for an O-SLA fault of one *term unit* u_x (i.e. one client or 0.1% of the instantaneous/total non-interruption ratio), b_x is the O-SLA fault severity for term x, $b_x^{(\max)}$ is its maximum severity, and f_x is a *shape function* with the signature:

$$f_x : [0; 1] \to \mathbb{R}^+, \tag{6}$$

employed for changing the importance of different fault classes. A game operator could make its offer more appealing by employing a shape function offering higher compensations for most frequent, low-severity faults rather than for the infrequent, higher-severity ones. For example, we can define a class of logarithmic parameterised shape functions:

$$f_x\left(b_x\right) = \frac{\log\left(a \cdot b_x + 1\right)}{\log\left(a + 1\right)}, \tag{7}$$

where $a \in \mathbb{R}^+$ is a coefficient shaping the distribution of compensations for different fault severities attempting to make the O-SLAs more appealing to the game providers.



(a) Logarithmic shape function f_x with five sample shape coefficients a.

(b) Compensation functions for client number term f_{cli} with a=5 and a=100.

Figure 3: Sample compensation functions employing logarithmic shape functions.

For example, Figure 3a plots the parameterised logarithmic function for a few values of a, while Figure 3b shows two sample compensation functions for the client number O-SLA term, both considering a term unit of one client ($u_{cli} = 1$) and a maximum fault severity of 2000 clients ($b_{cli}^{(max)} = 2000$). $C_{cli}^{(\texttt{ex1})}$ employs a 0.6 ¢ compensation per term unit ($c_{cli}^{(u)} = \$0.006$) and f_{cli} with a=5, while $C_{cli}^{(\texttt{ex2})}$ offers $c_{cli}^{(u)} = \$0.005$ and f_{cli} with a=100:

$$C_{cli}^{(\texttt{ex1})}\left(b_{cli}\right) = \frac{0.006 \cdot b_{cli}}{1} \cdot \frac{\log(5 \cdot \frac{b_{cli}}{2000} + 1)}{\log(6)};$$

$$C_{cli}^{(\texttt{ex2})}\left(b_{cli}\right) = \frac{0.005 \cdot b_{cli}}{1} \cdot \frac{\log(100 \cdot \frac{b_{cli}}{2000} + 1)}{\log(101)}.$$

Figure 3b shows that $C_{cli}^{(\texttt{ex2})}$ offers higher compensations for low severity faults ($b_{cli} < 1050$ clients), while $C_{cli}^{(\texttt{ex1})}$ offers higher compensations for the high severity faults ($b_{cli} > 1050$ clients). Thus, a game provider which expects more low severity faults might prefer $C_{cli}^{(\texttt{ex2})}$. Analogously, a game provider predicting high severity faults might choose $C_{cli}^{(\texttt{ex1})}$.

Non-negotiable O-SLA terms such as the issuer (i.e. game operator) and the measurement timestep describing the time interval between consecutive QoS evaluations are not represented for simplicity reasons.

5. O-SLA NEGOTIATION

We define the O-SLA negotiation between the game operator and the game provider as an decision process in which two parties interact with each other for mutual gain (i.e. maximise income and keep expenditures low). The game provider's income comprises the MMOG subscription sales and the compensations paid by the game operator in case of O-SLA faults, while its expenditures consist of the O-SLA acquisitions and the compensations to the clients for low QoS. The game operator's income results from the O-SLAs provisioned to the game provider, and its expenditures comprise the acquisition of resources from the cloud providers and the O-SLA compensations to the game providers. The accounting, billing and auditing aspects of SLAs fall outside the scope of this work (but solutions exist).

The three negotiation phases depicted in Figure 4 cover the game operators generating *O-SLA templates* based on cloud resource pricing and availability (phase one), the game providers instantiating and ranking O-SLA offers (phase two)

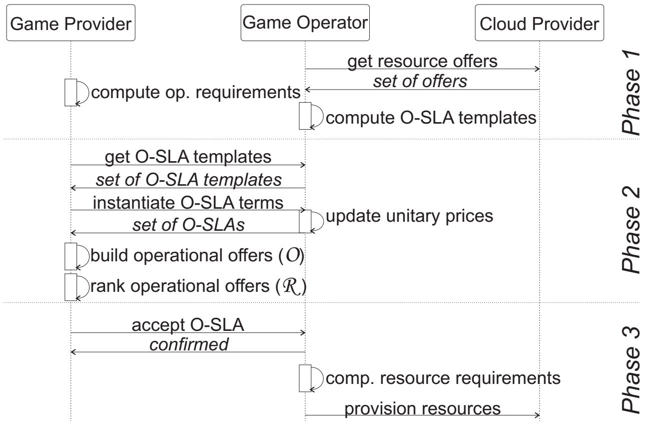


Figure 4: O-SLA negotiation protocol.

and finally, the binding agreement (phase three). A simpler one-phase request-offer matching algorithm, although desirable, cannot be employed because it would not be fair towards the game operators. The resource providers' pricing policies can change between the time the operator published an offer until the game provider accepted it, which would enable a game provider profit from delaying the answer. Introducing a validity deadline for offers to prevent this unfair behaviour could have negative effects on both game provider and game operator, as one might not have enough time for the ranking process, while the other would have to assume the risk of changes in cloud resource prices (within the offer validity time). Thus, the proposed negotiation involves dynamic offers (determined by the available cloud resources) and a possibility for game operators to propose small changes in price during the final agreement phase.

5.1 First Phase

In the first phase, the game operator checks the resources offered by different providers and publishes an *O-SLA template*, as defined in Equation 2. The game provider computes the *operational requirements*:

$$R = \left(G_{type}^{(R)}, t_{cli}^{(R)}, t_{ini}^{(R)}, t_{tni}^{(R)}, t_{time}^{(R)}, \sigma^{(R)}, P^{(T)}, C^{(T)}\right) \quad (8)$$

based on the current state of its provisioned O-SLAs and the estimated number of clients for the next time frame, where:

- $G_{type}^{(R)}$ is the required MMOG type;
- $t_{cli}^{(R)}$ is the estimated number of active accounts;
- $t_{ini}^{(R)}$ is the required instantaneous total non interruption ratio, initially equal to the minimum instantaneous non-interruption ratio s_{ini} of the game provider's service policy defined in Equation 1;
- $t_{tni}^{(R)}$ is the total non-interruption ratio, initially equal to the minimum total non-interruption ratio s_{tni} from the provider's service policy;
- $t_{time}^{(R)}$ is the estimated time period for these requirements expressed in hours;
- $\sigma^{(R)}$ is the geographical area;
- $P^{(T)}$ is the target hourly price per client defined by the game provider's service policy;
- $C^{(T)}$ is the target compensation per client per minute defined by the game provider's service policy.

5.2 Second Phase

In the second phase, the game provider gathers the O-SLA templates from all the game operators and instantiates them with the "best" values permitted by the O-SLA template for the operational requirements. When instantiating an O-SLA template, it also calculates the price increase for the client number P_{cli}, the instantaneous non-interruption ratio P_{ini}, and the total non-interruption ratio P_{tni}:

$$P_x = \frac{t_x^{(R)} - t_x^{(O_{\min})}}{u_x} \cdot p_x^{(u)} \cdot f_x\left(t_x^{(O)}\right), \forall x \in \{cli, ini, tni\}, \quad (9)$$

where $t_x^{(R)}$ is the operational requirement for the term $x \in \{cli, ini, tni\}$ (see Equation 8), $t_x^{(\min)}$ is the minimum value of the term x allowed by the operator through an O-SLA, $p_x^{(u)}$ represents the price per term unit u_x, and $f_x\left(t_x^{(O)}\right)$ is a shape function defined as in Equation 6. The final price the game provider is charged when accepting the O-SLA is:

$$P^{(O)} = P_{base} + \left(P_{cli} + P_{ini} \cdot t_{cli}^{(R)} + P_{tni} \cdot t_{cli}^{(R)}\right) \cdot T_{coeff}, \quad (10)$$

where P_{base} is the base price and T_{coeff} is the *validity period coefficient* that adjusts the price in case of changes in validity time requested by the provider:

$$T_{coeff} = \left\lceil \frac{t_{time}^{(R)}}{t_{time}^{(O_{\min})}} \right\rceil \cdot f_{time}\left(t_{time}^{(R)}\right), \quad (11)$$

where $\lceil\cdot\rceil$ is the ceiling function, $t_{time}^{(O_{\min})}$ represents the lowest O-SLA validity period allowed by the operator, and f_{time} is a shape function defined as in Equation 6.

Next, the O-SLA instances are grouped by the game provider into a set of M feasible *operational offers*:

$$O = \bigcup_{i=1}^{M} O\text{-}SLA_i, \quad (12)$$

Consider for example the operational requirements of 50 thousand clients and three O-SLAs (*O-SLA[1;3]*) with the maximum of 25, 20 and 30 thousand clients. The resulting operational offers are {*O-SLA1, O-SLA3*} and {*O-SLA2, O-SLA3*} ($M = 2$ in both cases). The combination {*O-SLA1, O-SLA2*} is not feasible because it does not meet the minimum requirements of 50 thousand players ($25 + 20 < 50$).

The game provider assigns to each operational offer an *operational rank* based on the weighted sum of three individual ranks: the pricing rank $\mathcal{P}_{O\text{-}SLA}$ (directly proportional), the compensation rank $\mathcal{C}_{O\text{-}SLA}$ (inversely proportional) and the resource fitness rank $\mathcal{F}_{O\text{-}SLA}$ (inversely proportional):

$$\mathcal{R} = \lambda_p \cdot \mathcal{P}_{O\text{-}SLA} - \lambda_c \cdot \mathcal{C}_{O\text{-}SLA} - \lambda_f \cdot \mathcal{F}_{O\text{-}SLA}, \quad (13)$$

where $\lambda_p, \lambda_c, \lambda_f \in [0; 1]$ and $\lambda_p + \lambda_c + \lambda_f = 1$. The main goal of this paper is to determine best practices for a game provider for computing these compensation and fitness weights in an environment with multiple competing providers and game operators (we studied the pricing weight in [15]). We define in the following the computation of the pricing, compensation and resource fitness ranks by the game provider.

The *pricing rank* $\mathcal{P}_{O\text{-}SLA}$ of an operational offer is a quantification how expensive a resource is, determined as the ratio between the aggregated hourly price $\frac{P_i^{(O)}}{t_{time_i}^{(O)}}$ of all M O-SLAs of an operational offer and the target price $P^{(T)} \cdot t_{cli_i}^{(O)}$ for servicing all clients in all M O-SLAs (see Equation 2):

$$\mathcal{P}_{O\text{-}SLA} = \frac{\sum_{i=1}^{M} \frac{P_i^{(O)}}{t_{time_i}^{(O)}}}{P^{(T)} \cdot \sum_{i=1}^{M} t_{cli_i}^{(O)}}. \tag{14}$$

The *compensation rank* quantifies the penalties the operator pays for O-SLA faults based on a *compensation gain* metric representing the area of the compensation function C_x within its definition interval $\left[0; b_x^{(\max)}\right]$ (see Equation 5):

$$\mathcal{A}_x = \int_0^{b_x^{(\max)}} C_x\left(b_x\right) \cdot db_x = \frac{c_x^{(u)}}{u_x} \cdot \int_0^{b_x^{(\max)}} b_x \cdot f_x\left(\frac{b_x}{b_x^{(\max)}}\right) \cdot db_x\,. \tag{15}$$

By substituting $y = \frac{b_x}{b_x^{(\max)}}$ in Equation 15, we obtain:

$$\mathcal{A}_x = \frac{c_x^{(u)} \cdot \left(b_x^{(\max)}\right)^2}{u_x} \cdot \int_0^1 y \cdot f_x(y) \cdot dy\,. \tag{16}$$

While the compensation gain completely characterises the compensation function for uniformly distributed SLA faults, it does not accurately do it in a real system with a non-uniform SLA fault distribution. To compensate for this drawback, we introduce an *SLA fault distribution function*:

$$\delta_x : \left[0; b_x^{(\max)}\right] \to \left[0; \Delta_{\max}\right], \tag{17}$$

where $\Delta_{\max}$ represents the maximum value of the SLA fault distribution function. We dynamically compute the SLA fault distribution for each MMOG zone by continuously monitoring the game play and recording each SLA fault. By superimposing δ_x to the compensation gain, we compute an adjusted metric called *characteristic compensation gain* which defines the compensation function for a specific MMOG:

$$\mathcal{A}_x^{(ch)} = \frac{c_x^{(u)} \cdot \left(b_x^{(\max)}\right)^2}{u_x} \cdot \int_0^1 y \cdot \delta_x\left(b_x^{(\max)} \cdot y\right) \cdot f_x(y) \cdot dy\,. \tag{18}$$

Using $\mathcal{A}_{C_x}^{(ch)}$, we can finally compute the compensation rank of an operational offer as the sum as the weighted sum of the normalised characteristic compensation gains for all O-SLA terms $x \in \{cli, ini, tni\}$:

$$\mathcal{C}_{O\text{-}SLA} = \sum_{i=1}^{M} \sum_{x \in \{cli,ini,tni\}} \psi_x \cdot \frac{\mathcal{A}_{x_i}^{(ch)}}{\mathcal{A}_x^{(REF)}}, \tag{19}$$

where $\mathcal{A}_x^{(REF)}$ represents a reference compensation gain considered ideal by the game provider (e.g. minimum compensation function from all operators), and $\psi_{cli}, \psi_{ini}, \psi_{tni} \in [0; 1]$ indicate the provider's preference for each O-SLA term, where $\psi_{cli} + \psi_{ini} + \psi_{tni} = 1$.

The *fitness rank* reflects how the operational offer matches the operational requirements, computed as a weighted sum of the ratio between the offered $t_x^{(O)}$ and the requested $t_x^{(R)}$ O-SLA terms (i.e. t_{cli}, t_{ini}, t_{tni}, and t_{time} – see Equation 2):

$$\mathcal{F}_{O\text{-}SLA} = \sum_{x \in \{cli,ini,tni,time\}} \phi_x \cdot \frac{S_x\left(t_{x_i}^{(O)}\right)}{t_x^{(R)}}, \tag{20}$$

$$S_x\left(t_{x_i}^{(O)}\right) = \begin{cases} \sum_{i=1}^{M} t_{x_i}^{(O)}, & x = cli; \\ \frac{\sum_{i=1}^{M} t_{x_i}^{(O)}}{M}, & x \in \{ini, tni\}; \\ \min\limits_{i \in [1;M]} \left\{t_{x_i}^{(O)}\right\}, & x = time, \end{cases} \tag{21}$$

$\phi_{cli}, \phi_{ini}, \phi_{tni}, \phi_{time} \in [0; 1]$ indicate the provider's preference for each O-SLA term ($\phi_{cli} + \phi_{ini} + \phi_{tni} + \phi_{time} = 1$) and S_x is an aggregation function (i.e. sum for number of clients, average for instantaneous and total non-interruption ratios, and minimum for validity period). The offer is unfit if the fitness rank is lower than one, is a perfect match if equal to one, or contains too many resources if higher than one.

As a final step, the operational offers are sorted in ascending order by their rank. It is worth mentioning that, although the price ranking is relatively static between successive negotiations (provided that the operators do not adjust their offers dynamically), the fitness and compensation rankings constantly vary based on the current operational demands and the operators' SLA fault history (see characteristic compensation gain function in Equation 18). This ensures that game providers do not constantly reach the same apparently-optimal operator, but are able to discover those whose offers most accurately match their needs.

5.3 Third Phase

In the third phase, the game provider attempts to accept an operational offer starting with the best ranked one, and continues through the list in case other competing providers already provisioned it. At this stage, the operators are allowed to propose small updates in the O-SLA terms to compensate for changes in the cloud providers' R-SLAs. In turn, the game providers will either recompute the rank for the O-SLA in question, or will simply skip to the next best offer according to their internal policy. After the negotiation, the provider tries to enforce the accepted O-SLA for the entire interaction with the clients and the game operator. To achieve this, the game provider collects and aggregates data from two sources: the game operator's QoS data collected from MMOG servers and the client that regularly reports (in the background) on the quality of game play. The game provider enforces the O-SLAs by compensating the clients according to their contractual terms (not covered here) and by penalising the game operators in case of QoS violations.

6. EXPERIMENTS

In the limited space available, we focus our experiments on the impact of the operational offer ranking on the game providers' gross profit in an environment with multiple game providers competing for O-SLAs from multiple game operators (see Equation 13). We performed further evaluation of our method in [14]. Our aim is to determine best practices for a provider in selecting the compensation and fitness weights for ranking operational offers that maximise the profit. We ignore the pricing weight studied in [15].

6.1 Experimental Setup

Our evaluation is based on simulation using traces from RuneScape, a real MMOG ranked second after World of

Table 1: Service policies of game providers.

Policy	s_{ini}	s_{tni}	s_{time} *[hours]*	$P^{(T)}$[\$]	$C^{(T)}$[\$]
`PP1-PP5`	$(\mathbf{0.86}; \mathbf{0.98}; \mathbf{0.03})$	0.992	$[168; 336]$	0.002	0.05
`PP6-PP10`	0.92	$(\mathbf{0.986}; \mathbf{0.998}; \mathbf{0.003})$	$[168; 336]$	0.002	0.05
`PP11-PP15`	0.92	0.992	$[(\mathbf{24}; \mathbf{312}; \mathbf{72}); (\mathbf{336}; \mathbf{624}; \mathbf{72})]$	0.002	0.05

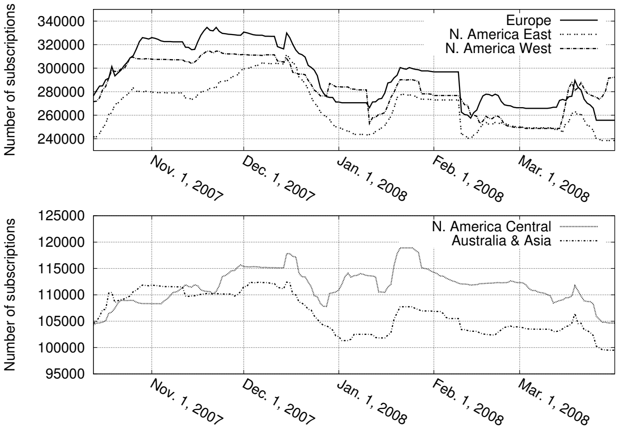


Figure 5: Concurrent RuneScape clients aggregated for all servers in each region.

Table 2: Commercial cloud R-SLAs.

Cloud provider	*VM types*	*Locations*	*Price [\$/ RSU/h]*	*GB/h]*	*Valid. [h]*	*Resource [seconds]*
Amazon	6	4	1.21	0.81	1	[65; 105]
CloudCentral	5	1	11.07	35.25	1	[50; 120]
ElasticHosts	4	1	1.22	2.73	1	[45; 120]
FlexiScale	4	1	0.72	1.46	1	[40; 50]
GoGrid	4	1	2.07	7.15	1	[60; 120]
Linode	5	1	0.67	2.37	24	[45; 120]
NewServers	5	1	0.38	0.71	1	[30; 120]
OpSource	6	1	0.09	0.15	1	[300; 540]
RackSpace	4	2	1.54	5.56	1	[100; 300]
ReliaCloud	3	1	0.96	1.04	1	[45; 60]
SoftLayer	4	3	0.70	1.75	1	[180; 300]
SpeedyRails	3	1	1.76	8.43	24	[80; 120]
Storm	6	2	0.99	1.54	1	[600; 900]
Terremark	5	1	1.40	6.14	1	[40; 60]
Voxel	4	3	0.83	0.94	1	[300; 600]
Zerigo	2	1	1.96	3.16	1	[60; 120]

Table 3: RuneScape-related O-SLA template.

Template	$t^{(O)}_{cli}$	$t^{(O)}_{ini}$	$t^{(O)}_{tni}$	$t^{(O)}_{time}$	C_x	
					f_x	a
O-SLA	$[2000; 20000]$	$[0.85; 0.95]$	$[0.99; 0.999]$	$[24; 168]$	log	$[1; 100]$

Table 4: Operational ranking configuration parameters.

Ranking	*Fitness rank* ($\mathcal{F}_{O\text{-}SLA}$)				*Operational rank*($\mathcal{R}$)	
acronym	ϕ_{cli}	ϕ_{ini}	ϕ_{tni}	ϕ_{time}	λ_f	λ_b
`or-[1;8]`	0.1	0.3	0.3	0.3	$(\mathbf{0.1}; \mathbf{0.8}; \mathbf{0.1})$	$(\mathbf{0.8}; \mathbf{0.1}; -\mathbf{0.1})$

Warcraft by the number of active paying customers in the US and European markets. We collected execution traces for a period of six months from 150 servers on four continents by sampling the number of players every two minutes (ranging between 0 and 2000, the maximum capacity of one RuneScape server [13]) (see Figure 5). For the client – game provider interaction, we use the real monthly subscription model of RuneScape (\$5.95 as of August 2010).

We employed 115 R-SLAs based on the resources offered by 16 commercial cloud providers summarised in Table 2. The hourly-based prices are first presented relative to the processing power and second relative to the memory availability. The prices include the upstream and downstream network traffic which may have an important impact on the final R-SLA prices, as in the case of CloudCentral. The geographical location, memory size, and price are clearly specified by all providers. We express the resource processing power using an MMOG-specific metric called *RS unit*, representing the equivalent requirements of one RuneScape server servicing 2000 clients. We compute this metric based on benchmarking and analysis data from existing investigations [13, 9]. We consider 100% resource uptime because most cloud providers have very high resource availabilities, only a few specifying concrete compensation terms.

We simulated 15 game providers that compete for operational offers based on the number of RuneScape active clients (see Figure 5) by employing one of the service policies `PP1-PP15` defined in Table 1 (see Equation 1). Each group of five providers, namely the groups employing the policies `PP1-PP5`, `PP6-PP10`, and `PP11-PP15`, runs the complete set of RuneScape MMOG traces. We run eight simulations labelled `or-1` to `or-8` in Table 4, each changing the operational ranking configuration employed by all the game providers by varying the weights for the compensation (λ_c) and fitness (λ_f) ranks (see Equation 13). We set the weight of the price rank to a fixed value of $\lambda_p = 0.1$ based on our previous study [15]. We start with a high emphasis on the importance of the compensation ranking (and implicitly a low importance on the fitness ranking) and continue by gradually changing the weights until reaching the opposite scenario, i.e. from $\lambda_b = 0.8$ and $\lambda_f = 0.1$, to $\lambda_b = 0.1$ and $\lambda_f = 0.8$ (see Equation 13 and Table 4).

We further simulated 74 game operators employing O-SLAs based on the template in Table 3, offering game providers an operation market with high diversity of operational and compensation terms. For the compensation function (see Equation 5), we used the logarithmic class of parameterised shape functions defined in Equation 7. We imposed an uniform distribution of the serviced geographical areas.

We analysed the financial aspect of the MMOG operation using two metrics: (1) *gross profit* representing the difference between the business actor's revenue and the cost of providing its services (excluding taxation and other overheads), and (2) *total compensation* (a fraction of gross profit) representing the total cost a business actor pays as a compensation for any SLA fault for the entire simulation period.

6.2 Results

Figure 6 shows the variation of three gross profit fractions: the MMOG operation expenses, client compensation expenses, and the income from O-SLA fault compensations. The top chart presents the trend of the aggregated profit

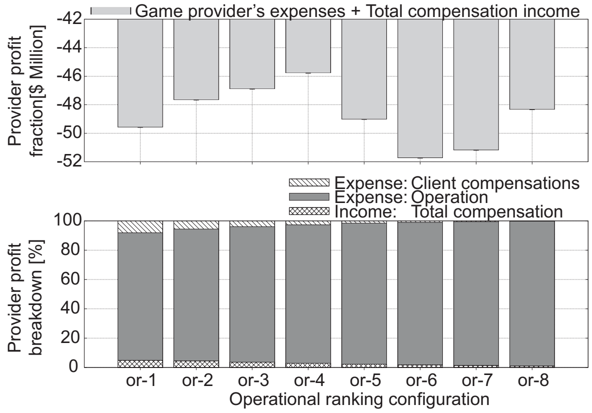


Figure 6: Variation of expenses and income from O-SLA compensations (top); profit breakdown (bottom).

fractions for all game providers, while the bottom chart shows their proportional composition. The income from client subscriptions of all game providers is constant throughout all runs at around $113.67 million (not shown in the graphs). The increasing trend of the game providers' gross profits from `or-1` to `or-4` is due to a decrease in the expenses caused by client compensations and resulting from a better selection of operational offers, a consequence of the increased weight of fitness ranking. The descending trend from `or-4` to `or-6` is due to increased expenses with operational offers, which is a consequence of further increasing the weight of fitness ranking. These expenses are slowly being compensated by further decreases in client compensations, which eventually lead to another increasing trend for `or-6` to `or-8`. Overall, the best gross profit value is reached when employing `or-4`. We observe that increasing the weight of fitness ranking leads to an increase of the operational offer expenses and to a decrease in the client compensations. Conversely, increasing the weight of the compensation rank leads to higher operational expenses, but also to an increase of the income from O-SLA fault compensations.

To analyse the impact of the operational rank weights on the game providers employing different service policies, we group them in two classes: (1) the *Low* s_{ini} class, offering the clients low QoS, in terms of the targeted instantaneous non-interruption ratio (employing `PP1` and `PP2` with low s_{ini}), and (2) the *High* s_{ini} class, targeting a high QoS by employing `PP4` and `PP5` with high values for the same term. The *Low* s_{tni} and *High* s_{tni} classes and respectively the *Low* s_{time} and *High* s_{time} classes are constructed similarly for the other QoS term and the targeted SLA validity. Figure 7 shows the gross profit variation of the game provider classes reported to an average of the `or-4` and `or-5` runs. We observe that the *Low* s_{ini} class favours the lower fitness rank and higher compensation rank weights, while *High* s_{tni} is only marginally affected by changes in the operational ranking weights. The *Low* s_{tni} slightly favours lower fitness rank weights, while the *High* s_{tni} is positively influenced by the increased fitness rank and lower compensation rank weights. The strongest impact of the operational ranking weights is observed for the s_{time} term: *Low* s_{time} favours lower fitness rank weights and has a strong negative reaction to higher compensation ranking weights, while *High* s_{time} performs best when the two weights are balanced and significantly worse otherwise.

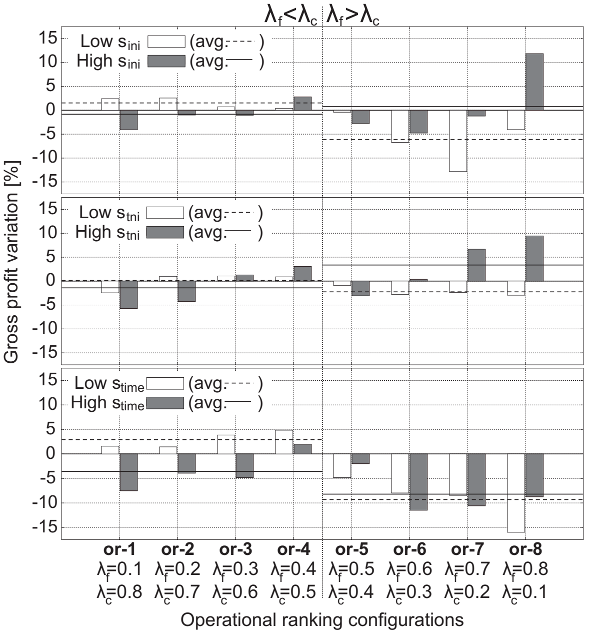


Figure 7: Gross profit variation in percentage to the average of `or-4` and `or-5`

.

Service policy term	*Low*		*High*	
s_{ini}	$\lambda_f \downarrow$	$\lambda_c \uparrow$	$\lambda_f =$	$\lambda_c =$
s_{tni}	$\lambda_f =$	$\lambda_c =$	$\lambda_f \uparrow$	$\lambda_c \downarrow$
s_{time}	$\lambda_f \downarrow$	$\lambda_c \uparrow$	$\lambda_f =$	$\lambda_c =$

Table 5: Best practice rank weight configurations for maximising game providers' profit: '↑' represents a high value, '↓' a low one, and '=' balanced.

We summarise these best-practice guidelines in Table 5, which defines the ranking weight configurations the game providers should use for different service policies. We conclude that the game providers' gross profit can be maximised in a competitive environment by balancing the fitness and compensation operational ranking weights. However, the game providers' service policies strongly impact the way in which the operational rank weights influence their gross profit, as summarised in Table 5.

7. CONCLUSION

The current MMOG ecosystem comprising tens of millions of players across hundreds of games forces game providers to also become game and infrastructure operators, leading to inefficient resource utilisation, high service prices, and limits market participation to only the largest game providers. We proposed a new ecosystem and business model for hosting and operating MMOGs which effectively splits the traditional monolithic MMOG companies into three main service providers: game providers, game operators, and resource providers, whose interactions are regulated through SLAs. In our model, game operators efficiently provision cloud resources for MMOGs based on their dynamic load and ensure proper game operation that maintains the required QoS to all clients. Game providers lease operation SLAs from the game operators to satisfy all client requests and manage multiple distributed MMOG sessions. These three self-standing, smaller, more agile service providers enable access to the MMOG market for the small and medium enterprises, and to the current commercial cloud providers. We focused in this paper on the business interaction between the game operator and the game provider by defining the negotiation protocol and the underlying O-SLA terms. Our model proposes a comprehensive MMOG operational ranking mechanism that considers and balances among three criteria: pricing, compensation, and resource fitness. We studied the impact of ranking operational offers on the game provider's gross profit in an environment with several providers competing for SLAs from multiple game operators through realistic simulations using traces from real MMOGs and SLAs from over ten commercial cloud providers We provide guidelines for balancing the three criteria and find that their impact depends on the service terms used in the game provider – client relation. In the future, we plan to investigate the impact of resource failures on QoS and compensation.

8. REFERENCES

[1] T. Alves and L. Roque. Using value nets to map emerging business models in massively multiplayer online games. In *Ninth Pacific Asia Conference on Information Systems*, PACIS, pages 1356–1367, 2005.

[2] B. Andersson, P. Johannesson, and J. Zdravkovic. Aligning goals and services through goal and business modelling. *Information Systems and e-Business Management*, 7(2):143–169, March 2009.

[3] L. D. Briceño, H. J. Siegel, A. A. Maciejewski, Y. Hong, B. Lock, M. N. Teli, F. Wedyan, C. Panaccione, C. Klumph, K. Willman, and C. Zhang. Robust resource allocation in a massive multiplayer online gaming environment. In *4th International Conference on Foundations of Digital Games*, FDG '09, pages 232–239. ACM, 2009.

[4] P. Chronz and P. Wieder. Integrating WS-Agreement with a framework for service-oriented infrastructures. In *11th International Conference on Grid Computing*, Grid, pages 225–232. IEEE Computer Society, 2010.

[5] K. Czajkowski, I. T. Foster, C. Kesselman, V. Sander, and S. Tuecke. SNAP: A protocol for negotiating service level agreements and coordinating resource management in distributed systems. In *8th International Workshop on Job Scheduling Strategies for Parallel Processing*, JSSPP, pages 153–183. Springer, 2002.

[6] P. Hasselmeyer, H. Mersch, B. Koller, H.-N. Quyen, L. Schubert, and P. Wieder. Implementing an SLA negotiation framework. In *eChallenges*, pages 154–161. IOS Press, October 2007.

[7] O.-I. Holthe, O. Mogstad, and L. A. Ronningen. Geelix livegames: Remote playing of video games. In *6th IEEE Consumer Communications and Networking Conference*, CCNC, 2009. ISBN 978-1-4244-2308-8.

[8] A. Iosup, V. Nae, and R. Prodan. The impact of virtualization on the performance and operational costs of massively multiplayer online games. *International Journal of Advanced Media and Communication*, 4(4):364–386, September 2011.

[9] A. Iosup, S. Ostermann, N. Yigitbasi, R. Prodan, T. Fahringer, and D. Epema. Performance analysis of Cloud computing services for many-tasks scientific computing. *IEEE Transactions on Parallel and Distributed Systems*, 22(6):931–945, June 2011.

[10] Y.-T. Lee and K.-T. Chen. Is server consolidation beneficial to MMORPG? A case study of World of Warcraft. In *IEEE International Conference on Cloud Computing*, CLOUD, pages 435–442. IEEE Computer Society, 2010.

[11] A. G. Ltd. ITIL glossary and abbreviations. `http://www.itil-officialsite.com/nmsruntime/saveasdialog.aspx?lID=1180\&sID=242`, July 2011.

[12] S. E. Middleton, M. Surridge, B. I. Nasser, and X. Yang. Bipartite electronic SLA as a business framework to support cross-organization load management of real-time online applications. In *Euro-Par 2009 – Parallel Processing Workshops*, number 6043 in LNCS, pages 245–254. Springer, 2010.

[13] V. Nae, A. Iosup, and R. Prodan. Dynamic resource provisioning in massively multiplayer online games. *IEEE Transactions on Parallel and Distributed Systems*, 22(3):380–395, 2011.

[14] V. Nae, R. Prodan, and A. Iosup. Autonomic cloud-based operation of massively multiplayer online games. In *The ACM Cloud and Autonomic Computing Conference*. ACM, August 2013.

[15] V. Nae, R. Prodan, A. Iosup, and T. Fahringer. A new business model for massively multiplayer online games. In *Second Joint WOSP/SIPEW International Conference on Performance Engineering*, ICPE 2012, pages 271–282. ACM, March 2011.

[16] M. Nojima. Pricing models and motivations for MMO play. In *Situated Play: Proceedings of the 2007 Digital Games Research Association Conference*, pages 672–681, September 2007.

[17] G. Oh and T. Ryu. Game design on item-selling based payment model in Korean online games. In B. Akira, editor, *Situated Play: Proceedings of the 2007 Digital Games Research Association Conference*, pages 650–657, Tokyo, September 2007.

[18] A. Parasuraman, V. A. ZelthamI, and L. L. Berry. A conceptual model of service quality and its implication. *Journal of Marketing*, 49:41–50, Fall 1985.

[19] R. Smith. The contract net protocol: High-level communication and control in a distributed problem solver. *IEEE Transactions on Computers*, 29(12):1104, December 1980.

[20] D. Talia, R. Yahyapour, W. Ziegler, P. Wieder, J. Seidel, O. Waldrich, W. Ziegler, and R. Yahyapour. Using sla for resource management and scheduling - a survey. In *Grid Middleware and Services*, pages 335–347. Springer, 2008.

[21] K. W. Wong. Resource allocation for massively multiplayer online games using fuzzy linear assignment technique. In *5th IEEE Consumer Communications and Networking Conference*, CCNC 2008, pages 1035–1039. IEEE, 2008.

[22] L. Wu and R. Buyya. Service level agreement (SLA) in utility computing systems. In *Performance and Dependability in Service Computing: Concepts, Techniques and Research Directions*, Advances in Web Technologies and Engineering, chapter 1, pages 1–25. IGI Global, July 2011.