

Media Brokerage: Agent-Based SLA Negotiation

Bruno Veloso^{1,3}, Benedita Malheiro^{2,3}, and Juan C. Burguillo¹

¹ School of Telecommunication Engineering, University of Vigo, Vigo, Spain

² School of Engineering, Polytechnic Institute of Porto, Porto, Portugal

³ INESC TEC, Porto, Portugal

Abstract. Media content personalisation is a major challenge involving viewers as well as media content producer and distributor businesses. The goal is to provide viewers with media items aligned with their interests. Producers and distributors engage in item negotiations to establish the corresponding service level agreements (SLA). In order to address automated partner lookup and item SLA negotiation, this paper proposes the MultiMedia Brokerage (MMB) platform, which is a multiagent system that negotiates SLA regarding media items on behalf of media content producer and distributor businesses. The MMB platform is structured in four service layers: interface, agreement management, business modelling and market. In this context, there are: (i) brokerage SLA (bSLA), which are established between individual businesses and the platform regarding the provision of brokerage services; and (ii) item SLA (iSLA), which are established between producer and distributor businesses about the provision of media items. In particular, this paper describes the negotiation, establishment and enforcement of bSLA and iSLA, which occurs at the agreement and negotiation layers, respectively. The platform adopts a pay-per-use business model where the bSLA define the general conditions that apply to the related iSLA. To illustrate this process, we present a case study describing the negotiation of a bSLA instance and several related iSLA instances. The latter correspond to the negotiation of the Electronic Program Guide (EPG) for a specific end viewer.

Keywords: Agent-based brokerage platform, service level agreements, negotiation.

1 Introduction

Media content personalisation poses a significant challenge to media content producers, distributors and viewers. The goal of media content producers is to place their media items at the highest price possible and the goal of a content distributor is to strengthen the relationship with existing customers and increase the number of customers by offering the most appropriate media contents from a multitude of sources, while taking into account the viewer profile. The viewers are unable to search, find and choose their preferred content in near real time due to the size of the search space both in terms of the number of offers and of the diversity of sources. This complex problem is inherently distributed, decentralised and requires a distributed system approach, where the actors can be

modelled as autonomous intelligent entities. The MMB platform addresses this challenge by acting as a broker on behalf of producer and distributor businesses and adopting a layered multiagent architecture. Although recommendation is out of the scope of this paper, producers rely on recommendation services to select items for negotiation.

Service provisioning is typically regulated by legal binding contracts called SLA. A SLA is an agreement between service providers and consumers. This agreement describes the service, the rights and the responsibilities of both parties [4]. The SLA life cycle is composed of six steps: (i) template definition; (ii) template discovery; (iii) negotiation; (iv) service provision; (v) monitoring and enforcement; and (vi) termination and decommission.

The main contributions of this paper occur at two different levels: at the SLA level and at the business model level. At the SLA level, it includes the definition of bSLA and iSLA and the adoption of distinct negotiation protocols: (i) the negotiation of bSLA instances adopts the WS-Agreement one shot negotiation protocol; and (ii) the negotiation of iSLA instances follows the WS-Agreement Negotiation specification and adopts the Foundation for Physical Intelligent Agents (FIPA) Iterated Contract Net Interaction Protocol (FICNIP). In terms of the business model, although all services are provided generically under the pay-per-use model, the celebrated iSLA remain latent till the items are actually provisioned, *i.e.*, when viewers click to watch the corresponding items. Only then, the corresponding iSLA are activated. Additionally, the bSLA defines the brokerage service provisioning terms, under which the individual iSLA are established. This SLA cascading effect is, as far as we know, a novel and relevant approach. Once the validity period of the established iSLA expires, they are removed from the system without costs for the parties. The provisioning of an item results in three payments: (i) the distributor pays the provider the established item provisioning fee; (ii) the provider pays to the platform the accorded brokerage fee; and (iii) the distributor pays the platform the negotiated brokerage fee. These fees are typically distinct.

This paper is organized in five sections. Section 1 presents the context, identifies the problems addressed and describes the approach adopted. Section 2 presents the related work on multiagent brokerage systems that establish Service Level Agreements. Section 3 describes the system architecture and the implemented platform. Section 4 describes a test case. Finally, Section 5 provides the conclusions and presents the future developments.

2 Related Work

Brokerage platforms frequently adopt the multiagent paradigm since resource, service or partner brokerage is an inherently distributed, decentralised complex problem. In this section we explore the state of the art regarding multiagent brokerage systems which include SLA negotiation.

The *Web Services SLA negotiation framework* (WS SLA) is an agent-based system composed of provider and consumer agents that negotiate SLA instances

[1]. The *mOSAIC* project relies on a multiagent system, which is composed of broker, vendor, meeter, tier, mediator and archiver agents, to negotiate cloud infrastructure resources. The brokering policy sets constraints and objectives on multiple parameters. The broker analyses a number of SLA proposals and chooses the SLA proposal that satisfies the applicable constraints and objectives rules [2]. *Cloud@Home* uses a SLA-based broker for cloud infrastructure resources. It has a modular architecture composed of resource management, SLA management and resource abstraction modules. The negotiation policies can only be configured by the administrator [3]. The *S-Cube* by [5] proposes a SLA negotiation for services composed of monitor, service listener, negotiation broker and runtime service discovery tool. The services are described using Business Process Execution Language (BPEL). This system is event-driven, *i.e.*, reacts to events created by providers or consumers. The broker module uses negotiation rules. The *Framework for automated service negotiation in cloud computing* (FASNCC) [6] is a multiagent system composed of autonomous provider and consumer agents that negotiates cloud computing resources. It includes a marketplace where resource offers and requests are registered and matched. The *SLA negotiation framework for cloud computing* (SLA NFCC) by [8] includes the customer agent, the SaaS broker coordinator agent, the SaaS provider agent and the IaaS layers. These layers are supported by the IaaS SLA generator, directory, policy database and knowledge base modules. The SLA negotiation considers time and market constraints as well as trade-offs by relying on counter offer generation strategies and decision making heuristics. The *Policy-based Web Services SLA negotiation system* (PbWS SLA NS) by [9] is composed of provider and consumer agents. The system provides a flexible framework for SLA negotiation by incorporating multiple strategy models based on the high level policies. The MMB platform, when compared with these platforms, has the following distinctive features: (*i*) implements two types of SLA (brokerage and item SLA); (*ii*) adopts two SLA specifications; (*iii*) relies on two standard negotiation protocols; and (*iv*) businesses define the negotiation policy of individual items. The bSLA defines the platform service provisioning terms, under which all individual iSLA established by the business are governed.

Table 1 summarises the SLA negotiation approaches adopted by the previous platforms. In terms of specifications, the WS SLA and the SLA NFCC implement the Web Service Policy Specification (WS-Policy) to describe negotiation policies, whereas Cloud@Home and the MMB platform implement the WS-Agreement Specification. In terms of negotiation protocols, WS SLA, mOSAIC, PbWS SLA NS and the MMB platform (iSLA instances) use FICNIP: WS SLA adopts an utility-based negotiation strategy, the SLA NFCC applies one to many bargaining protocol and Cloud@Home and MMB (bSLA instances) adopt the one shot negotiation protocol included in the WS-Agreement. The S-Cube project implements a reactive single phase and a reactive two phase SLA negotiation protocols, and finally, the FASNCC implements a bilateral multi-step monotonic concession negotiation protocol.

Table 1. Comparison of Agent-based SLA Negotiation Platforms

Platform	EndPoint	Web Service API Specification	SLA Negotiation Protocol
WS SLA	SOAP	WS-Policy	FICNIP
moSAIC	REST	-	FICNIP
Cloud@Home	SOAP	WS-Agreement	One shot
S-Cube	SOAP	-	Reactive single phase
FASNCC	REST	-	Proactive two phase
SLA NFCC	REST	WS-Policy	Bilateral multi-step monotonic concession
PbWS SLA NS	SOAP	-	One to many bargaining
MMB	REST	WS-Agreement	One shot (bSLA)
		WS-Agreement Negotiation	FCNIP (iSLA)

3 Media Brokerage Platform

The architecture of the implemented MMB platform, which displayed in Figure 1, is organized in (1) interface, (2) agreement, (3) enterprise and (4) market layers and contains five types of agents: (*i*) interface agents to interact with businesses; (*ii*) SLA agents to manage SLA instances; (*iii*) enterprise agents to model businesses; (*iv*) market delegate agents to negotiate specific items (producer) or the fulfilment of an EPG (distributor); and (*v*) layer manager agents (interface, agreement, enterprise and market layer agents). An earlier version of this platform is detailed in [7]. Each business (producer or distributor enterprise) is represented in the platform by the corresponding: (*i*) enterprise interface agent located in interface layer; (*ii*) SLA agent located in agreement layer; (*iii*) enterprise agent that models the enterprise within the platform in the enterprise layer; and (*iv*) an undetermined number of delegate agents involved in specific item or EPG negotiations in the market layer. These agents are identified by a trading code, preventing third parties from intervening in the undergoing negotiations.

3.1 Service Level Agreements

The MMB platform implements two Open Grid Forum (OGF) specifications: (*i*) the Web Services Agreement Specification (WS-Agreement) to represent all SLA templates used and to support the one shot bSLA negotiation; and (*ii*) WS-Agreement Negotiation Specification (WS-Agreement Negotiation) to support the multi-round iSLA negotiation. These specifications were, in our case, implemented on top of the FIPA Agent Communication Language (ACL) in order to minimise the communication latency between parties. Additionally, to implement a multi-round negotiation mechanism within the WS-Agreement Negotiation Protocol, we added FICNIP. In our case, we also contemplate the periodic or event-driven renegotiation of bSLA.

The life span (validity) of a bSLA is typically longer than that of an iSLA. In fact, multiple iSLA, involving a given business, can be celebrated, applied and terminated within the scope of the same bSLA. The arrows between the iSLA and the bSLA life cycles represent the feedback provided by the iSLA

instances to the distributor and producer bSLA monitoring modules. When an iSLA expires, the producer pays the platform for the brokerage service, and the distributor reimburses the platform and the producer for the brokerage and the item provisioning services, respectively.

Since the structure of a WS-Agreement template is highly extensible, both bSLA and iSLA are instances of WS-Agreement templates. The SLA template presented in Table 2 refers to an iSLA and includes context (validity, parties, *etc.*), terms (service terms and guarantee terms) and constraints fields. The service terms specify the service functionalities and the guarantee terms stipulate the service obligations and penalties. The constraints define the acceptable parameter negotiation range. All SLA instances are negotiated according to the negotiation strategy defined by the corresponding business.

Table 2. SLA Template

Context	Negotiation	Agreement initiator
		Agreement responder
		Service provider
		Template id
		Template name
	Negotiation offer	Counter offer to
		Creator
		Expiration time
		State
		Price
Terms	Service	Item
		Service id
		Service name
	Guarantee	Penalties
		Rewards
Constraints	Creation	Item
		Constraints

3.2 Brokerage SLA Negotiation

A business wishing to join and use the platform needs to establish a bSLA with the platform. The bSLA negotiation strategy is stored in the business SLA agent and the negotiation with the platform adopts the WS-Agreement one shot protocol. In the case of a distributor, the bSLA specifies that the distributor must invite for negotiation all producers with items matching the viewer profile. The producer bSLA states that a producer must accept all item negotiation invitations issue by potential partner distributors. The business SLA agent launches the two-step bSLA negotiation process: (*i*) obtains the SLA template; and (*ii*) negotiates the platform services fruition conditions, using the single proposal WS-Agreement Negotiation protocol. Upon success, the business completes its registration in the platform and is ready to take advantage of the platform brokerage services.

3.3 Item SLA Negotiation

Every time a producer uploads a new item, the corresponding SLA agent creates an iSLA template instance. Whenever the producer accepts a distributor invitation and selects this item for negotiation, the producer and distributor launch ephemeral delegate producer and distributor agents to negotiate the terms of the item iSLA. The goal of a producer delegate is to negotiate successfully the item. The goal of a distributor delegate is to negotiate the fulfilment of a given viewer EPG with personalised items. The fulfilment of an EPG is performed by a sequence of item negotiations, *i.e.*, each negotiation selects one item for the EPG, and it is achieved when the list of negotiated items fills the duration of the EPG. An established iSLA specifies that the distributor must share with the producer the corresponding viewer profile and feedback data and that the producer must provide the item when requested by the viewer. The provisioning of the item implies the payment by the distributor of the accorded value.

One business may participate in multiple trading rooms by launching multiple delegate agents, *i.e.*, a business can negotiate simultaneously n products or m times the same product. Each delegate agent is identified by a universally unique identifier (UUID) code generated when the producer is invited to negotiate. The negotiation protocol is the FICNIP, where the distributor delegate and producer delegates play the roles of the initiator and participant agents, respectively.

The distributor delegate agent uses the item utility function U_{dpi} to assess all producer proposals. Equation 1 presents U_{dpi} , which is the normalised utility, according to distributor d , of the item i proposed by producer p , where $price_{pi}$ is the normalise item price proposed by producer p , $ranking_{pi}$ is the normalise item-viewer similarity ranking determined by producer p and, finally, α_d is the weight attributed to the item ranking by distributor d .

$$U_{dpi} = \alpha_d \times ranking_{pi} + (1 - \alpha_d) \times (1 - price_{pi}) \quad (1)$$

The distributor sets α ($\alpha_d \in [0,1]$) according to the relevance it attributes to the item ranking versus the item price. In fact, the item negotiation between producer and distributor delegates corresponds to the negotiation of the iSLA, *i.e.*, the terms of the item provisioning service. Figure 1 represents the successful establishment of an iSLA. This process is conducted in three steps; *(i)* the distributor and producer delegates negotiate using FICNIP and the distributor delegate chooses the proposal with highest utility – represented by A and B in Figure 1; *(ii)* the outcome is communicated to the enterprise and SLA agents – represented by C and D in Figure 1; and *(iii)* the chosen producer and distributor SLA agents establish the iSLA according to the negotiated terms – represented by E and F in Figure 1. The validity of an established iSLA is equal to the validity of the generated EPG, which is by default 24 h. The outcome of each negotiation is an item, which will be added to the viewer’s EPG. Since the platform adopts the pay-per-use model, a successful negotiation between a producer and a distributor does not imply any immediate item transaction or payment. The actual transaction only takes place when the viewer clicks to watch the item, triggering the iSLA activation and the item provisioning by the producer

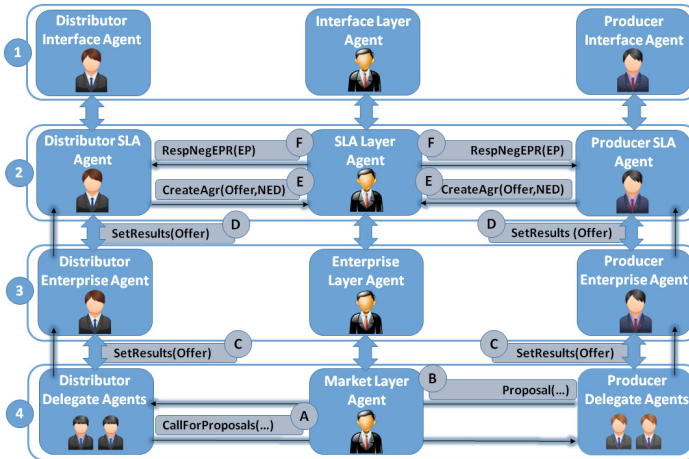


Fig. 1. Item Service Level Agreement establishment

(transaction). During the validity of the iSLA, the distributor reports all relevant viewer data (item watch time, rating and tag) to the producer. Billing and payment occurs when the iSLA validity expires.

4 Case Study

To illustrate the SLA processing within the agent-based media brokerage platform, the following subsections present the bSLA and iSLA life cycle and explain their role in the generation of a personalised EPG. The reported experiments used the HetRec 2011v. 2.0 data set¹, containing information on films and users.

4.1 Brokerage SLA

The business SLA agent is responsible for conducting the life cycle of the bSLA, which defines the terms of brokerage service between the business (producer or distributor) and the platform. In the SLA creation stage, the SLA agent creates a bSLA template instance, negotiates with the SLA layer agent and, upon success, establishes the bSLA between the business and the platform. To illustrate this process, let us assume that during the registration of distributor D01, the D01 SLA agent offers to pay a royalty of 0.9 % of its transactions to the platform and the SLA Layer agent counter offers, on behalf of the platform, a royalty of 1 % of the transactions. D01 accepts this counter offer and the bSLA is established.

During service provisioning, the business SLA agent monitors the brokerage service conditions, namely, the number of iSLA negotiation invitations, established iSLA, successful and violated iSLA as well as the payments to the platform and, in the case of the distributors, to the producer partners.

¹ <http://www.grouplens.org/datasets/hetrec-2011/>

The conclusion stage corresponds to the voluntary business de-registration or to the compulsory business eviction (penalty for the repeated violation of the bSLA or iSLA terms) from the platform. In the case of the voluntary de registration, the SLA agent checks for pending iSLA, bills and violations before terminating the bSLA.

4.2 Item SLA

When a producer uploads an item to the platform, the iSLA life cycle starts. During the creation stage, the producer SLA agent first generates an iSLA template instance. When the producer selects the item for negotiation in the market layer with a distributor, the producer and distributor delegates negotiate the final terms of the iSLA, taking into account item price and ranking, and, upon success, establish the iSLA between the distributor and the producer.

To generate a personal EPG, we selected randomly from the dataset a viewer of distributor D01. The distributor D01, based on the profile of this viewer, invites several (≥ 12) producers for negotiation. The distributor and the invited producer delegates are launched in the market layer and the EPG content negotiation begins. During the FICNIP negotiation rounds, the producer delegates adopt a linear incremental item price adaptation tactic. The distributor delegate evaluates and chooses the item proposal with highest item utility value and repeats the process until the EPG is filled. The item i utility (U_{dpi}) depends on the item price ($price_{pi}$) and item ranking ($ranking_{pi}$) provided by producer p .

Table 3 presents the outcome, including the EPG time line and the negotiated items together with the chosen producers, the accorded item price, the item ranking and the item utility (calculated by the distributor D01 delegate). The viewer accesses this personal EPG via a Web page. When he/she clicks to watch an item, the item provisioning service starts. During the service consumption, the SLA agent monitors service conditions, verifying the item genres and the fulfilment of the established terms.

Table 3. EPG Negotiation for the selected viewer

Time	Item	Producer	$price_{pi}$ ($\mu\text{€}/\text{min}$)	$ranking_{pi}$	U_{dpi}
00:00	(0049) The Usual Suspects	11	37	0.877	0.531
01:46	(1085) Star Wars: Episode V	02	44	0.821	0.481
03:50	(4569) Donnie Darko	13	45	0.875	0.489
05:43	(0205) Before Sunrise	04	41	0.775	0.485
07:28	(6490) Kill Bill: Vol. 2	25	35	0.813	0.525
09:45	(0823) Rear Window	16	37	0.818	0.516
11:37	(7925) The Incredibles	07	25	0.781	0.568
13:32	(1449) The Hunt for Red October	28	31	0.735	0.526
15:46	(0505) Rudy	19	37	0.833	0.520
17:40	(6760) The Lord of the Rings	10	28	0.800	0.557
21:10	(4337) Ghost World	01	43	0.750	0.468
22:52	(1036) Monty Python and the Holy Grail	14	49	0.856	0.464

The conclusion stage corresponds to the successful termination of iSLA or to the SLA resolution due to the violation of the iSLA terms. In the case of successful termination, the distributor SLA agent reimburses the producer and the platform for the item provisioning and brokerage services, whereas the producer pays the platform for the brokerage service. Table 4 shows the potential payments that the parties incur when the viewer consumes items 0049, 1085, 4569 and 0205 of the proposed EPG. For example, when Viewer watches item 0049 (The Usual Suspects), the producer P11 has to pay 0.74 $\mu\text{€}/\text{min}$ to the platform and the distributor D01 has to pay 37.00 $\mu\text{€}/\text{min}$ to the producer P11 and 0.37 $\mu\text{€}/\text{min}$ to the platform.

Table 4. EPG potential payments

	0049 Plat		1085 Plat		4569 Plat		0205 Plat
	P11 0.74		P02 0.88		P13 0.90		P04 0.82
D01	37.00 0.37	D01	44.00 0.44	D01	45.00 0.45	D01	41.00 0.41

5 Conclusions

In this paper we have identified and modelled two interdependent contractual relationships: (i) bSLA defines the platform service provision terms for each business; (ii) iSLA defines the item provisioning terms between businesses; and (iii) iSLA are celebrated within the scope of the involved business bSLA, *i.e.*, they must fulfil the agreed brokerage service provision terms. In terms of SLA negotiation, while the bSLA adopts and implements the WS Agreement single round negotiation, the iSLA implements the WS Agreement multi-round negotiation. In the case of the iSLA negotiation, the proposals are evaluated by an utility function that takes into account the item price and viewer similarity ranking. The subsumption of the individual iSLA terms, by the terms of the involved bSLA parties, together with the two types of SLA are up to our knowledge a novel and relevant approach. The described multi-agent SLA negotiation model has been implemented and tested using the HetRec data set.

To improve the overall quality of the services provided, we plan to design a distributed reputation model, considering the parties involved (platform, producers and distributors) and the SLA history, to be used in all future SLA negotiations. Businesses that have fulfilled their SLA obligations will reinforce their reputation when compared to those with repeated violations. In the case of the bSLA negotiation, the reputation model can be used to re-negotiate the bSLA terms. In the case of the iSLA negotiation, the distributor will use the producer reputation together with the item utility value to choose the item and the producer will consider the distributor reputation to decide whether to accept or reject a negotiation invitation.

Acknowledgements. This work was partially supported by the ERDF – European Regional Development Fund through the COMPETE Programme (operational programme for competitiveness) and by National Funds through the FCT – Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) within project «FCOMP – 01-0124-FEDER-022701».

References

1. Al-aaidroos, M., Jailani, N., Mukhtar, M.: Agent-based negotiation framework for web service's sla. In: 2011 7th International Conference on Information Technology in Asia, pp. 1–7. IEEE (2011)
2. Amato, A., Di Martino, B., Venticinque, S.: Evaluation and brokering of service level agreements for negotiation of cloud infrastructures. In: 2012 International Conference for Internet Technology and Secured Transactions, pp. 144–149. IEEE (2012)
3. Cuomo, A., Di Modica, G., Distefano, S., Puliafito, A., Rak, M., Tomarchio, O., Venticinque, S., Villano, U.: An sla-based broker for cloud infrastructures. *Journal of grid computing* 11(1), 1–25 (2013)
4. Hiles, A.: E-Business Service Level Agreements: Strategies for Service Providers, E-Commerce and Outsourcing. Service Level Management Series, Rothstein Catalog on Service Level Books (2002)
5. Mahbub, K., Spanoudakis, G.: Proactive sla negotiation for service based systems. In: 2010 6th World Congress on Services (SERVICES-1), pp. 519–526. IEEE (2010)
6. Pan, L.: Towards a framework for automated service negotiation in cloud computing. In: 2011 IEEE International Conference on Cloud Computing and Intelligence Systems, pp. 364–367. IEEE (2011)
7. Veloso, B., Sousa, L., Malheiro, B.: Personalised advertising supported by agents. In: Omatu, S., Neves, J., Rodriguez, J.M.C., Paz Santana, J.F., Gonzalez, S.R. (eds.) *Distrib. Computing & Artificial Intelligence*. AISC, vol. 217, pp. 473–481. Springer, Heidelberg (2013)
8. Wu, L., Garg, S.K., Buyya, R., Chen, C., Versteeg, S.: Automated sla negotiation framework for cloud computing. In: 2013 13th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, pp. 235–244. IEEE (2013)
9. Zulkernine, F., Martin, P., Craddock, C., Wilson, K.: A policy-based middleware for web services sla negotiation. In: IEEE International Conference on Web Services, pp. 1043–1050. IEEE (2009)