# Towards a Formal Specification of SLAs with Compensations*

Carlos Müller, Antonio M. Gutiérrez, Octavio Martín-Díaz, Manuel Resinas, Pablo Fernández, and Antonio Ruiz-Cortés

University of Sevilla, Sevilla, Spain

**Abstract.** In Cooperative Information Systems, service level agreements (SLA) can be used to describe the rights and obligations of parties involved in the transactions (typically the service consumer and the service provider); amongst other information, SLA could define guarantees associated with the idea of service level objectives (SLOs) that normally represent key performance indicators of either the consumer or the provider. In case the guarantee is under-fulfilled or over-fulfilled SLAs could also define some compensations (i.e. penalties or rewards). In such a context, during the last years there have been important steps towards the automation of the management of SLAs, however the formalization of compensations in SLAs still remains as an important challenge.

In this paper we aim to provide a characterization model to create SLAs with compensations; specifically, the main contributions are twofold: (i) the conceptualization of the Compensation Function to express consistently penalties and rewards and (ii) a model for Compensable Guarantees that associate SLOs with Compensation Functions. This formalization models aim to establish a foundation to elaborate tools that could provide an automated support to the modeling and analysis of SLAs with compensations. Additionally, in order to validate our approach, we model and analyze a set of guarantee terms from three real world examples of SLAs and our formalization proves to be useful for detecting mistakes that are typically derived from the manual specification of SLAs in natural language.

## 1 Introduction

The shift from product to services in the industry is a major trend for developed countries. In such a context this evolution implies the creation of a network of dependable organizations that exchange services and create cooperative information systems (CIS) to gain business value. For instance, in a cloud scenario a Software as a Service (SaaS) provider may use several Infrastructure as a Service (IaaS) providers such as Amazon Elastic cloud (EC2)[1] and the Google Cloud

* This work was partially supported by the European Commission (FEDER), the Spanish and the Andalusian R&D&I programmes (grants P12-TIC-1867, TIN2012-32273, TIC-5906 and IPT-2013-0890-3)

[1] Available at `http://aws.amazon.com/ec2/`

R. Meersman et al. (Eds.): OTM 2014, LNCS 8841, pp. 295–312, 2014.
© Springer-Verlag Berlin Heidelberg 2014

Storage service[2], to offer a service combination of a virtual machine with support for data persistence.

In this cooperative environment there is a craving for guarantees that support a reliable service consumption and cooperation, and service level agreements (SLAs) represent a first-class citizen to describe the parties rights and obligations. Specifically, SLAs are composed by different terms that typically define guarantees associated with a certain service level objective (SLOs) and they should be enforced by one party (the guarantor) to another party (the beneficiary); in most cases the former correspond to the service provider, and the latter to the service consumer. Additionally, real world SLAs usually include a set of compensations that represent the consequences of underfulfilling (penalties) or overfulfilling (rewards) the SLOs. We coin the concept of Compensable SLAs referring to such SLAs that include at least a compensation action, either a penalty or a reward. An example of the importance of this kind of SLAs is the cost of cloud service unavailability amounting to more than 70 million USDs based on hourly costs, by providers such as Amazon and Microsoft from 2007 to 2012 [8].

In this scenario it is important to note that in spite intraorganization modeling has been extensively studied with concepts of KPI[9] or PPI[5], there is a lack of a formal model to specify SLAs with compensation mechanisms. In fact, there is a strong relationship between the KPI of an organization and its commitments in terms of SLOs identified within an SLA.

Specifically, this work is focused on the modeling of SLAs with compensations with two main research goals: (i) a formal definition of different kinds of compensations, either penalties or rewards, and (ii) the checking of some desirable properties to automate the analysis of compensations. This analysis would represent important benefits for both consumers and providers in CIS: On the one hand, service providers could automate the optimization of the provision of services based on the compensations involved; on the other hand, service consumer could automate the analysis of guarantees in the SLA to understand its risk.

Our approach is grounded on the novel definition of Compensation Function (CF) that is inspired in the concept of penalty function introduced by Leitner et al. [11] and it has been extended to include the notion of rewards and to be aligned with the current most prominent SLA specification (WS-Agreement [1]). In addition, we also extend the notion of consistence and validity proposed in [14] in order to include compensations by means of a formal definition of properties that could be automatically checked. Our proposal has been successfully applied to model three real world SLAs that define different types of compensations and to detect some potential inconsistences.

This paper is structured as follows: Section 2 introduces the motivating scenarios of three real world SLAs; in Section 3 we present the conceptualization of the Compensation Function. In Section 4 we present the relationships between the SLO and Compensation Function to formalize the concept of Compensable Guarantees and model the different examples presented in the motivating scenarios.

[2] Available at `http://cloud.google.com/products/cloud-storage`

In Section 5 we analyze the literature to identify related approaches to deal with compensations. Finally, in Section 6 we outline some conclusions and future work.

## 2 Motivating Scenarios

SLAs are widely used in the industry in situations where consumers and providers need or desire to explicitly express certain guarantees over the service transactions. These guarantees are typically tied to certain consequences in terms of penalties and rewards depending whether the guarantee is underfulfilled or overfulfilled; we commonly refer to these consequences as compensations.

In this section we motivate the need for formal compensable agreements with three real world scenarios that include both computing services and human-driven services. In all cases, there is a strong need to express compensations related to the guarantees defined. In the rest of the section we introduce the scenario and present an example of compensation identified in its SLA.

### 2.1 AWS EC2 SLA

Amazon Web Services (AWS) is a service catalogue that has boosted the idea of cloud computing in the industry; amongst them, the Elastic Computing Cloud (EC2) represents a widely used Infrastructure as a Service. The aim of EC2 is to provide a scalable infrastructure to organizations that have variable needs or they need to grow seamlessly without the investment for an internal data center. In this context, the reliability of a virtualized infrastructure represents a key point for IaaS consumers in order to choose a service like AWS EC2.

As a consequence, Amazon has explicitly published an SLA for EC2[3] that is based on the idea of Monthly Uptime Percentage (MUP); this element, characterizes a guarantee over the availability of the virtual resources requested. Specifically, the consequences of failing a certain MUP is defined by Amazon in two levels: in case the MUP drops below 99.95 percent and in case the MUP drops below 99 percent. Figure 1 depicts the actual penalty function [10] of this scenario that is defined as a percent of discount in the next billing cycle a.k.a Service Credit Percentage (SCP). Note that in such Figure a dark point denotes the inclusion of the service property value in the interval, gray points means the value exclusion.

### 2.2 Telecomm SLA

The regional Government of Andalusia in Spain outsources the installation and management of telecommunication networks. Specifically, the demanded services include issues such as: project managing, interventions, network maintenance, installations, and logistics. An SLA[4] is specified by the regional government including some penalties for the services provider.

[3] Available at `http://aws.amazon.com/es/ec2/sla/`
[4] Available at `http://goo.gl/WIke8y`

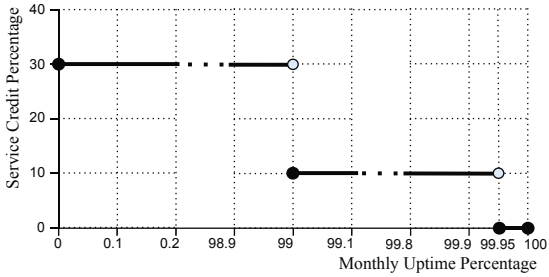



**Fig. 1.** Example AWS EC2: Penalties for Amazon as provider

Two examples of terms with penalties have been selected from the agreement. On the one hand, the term shown in the example ARG-1 of Figure 2 demands that 90% of interventions must be solved (cf. solid vertical line in the graph). However, as depicted in the table some penalties apply for a range of values that fulfill such a demand. Specifically, the table establishes that if the service provider solve more than 95% of interventions no penalties apply, but some bill penalties apply from 90% to 95% of interventions solved. This situation could imply a definition error that must be tackled in our proposal. On the other hand, the term shown in the example ARG-2 (cf. Figure 3) has not the problem of ARG-1 because it demands solving 95% of urgent interventions and any underfulfillment involve penalties.

| SLA: Interventions > 90 | |
|---|---|
| Measurement | % Penalty |
| 94 =< x < 95 | 1 |
| 93 =< x < 94 | 2 |
| 92 =< x < 93 | 3 |
| 91 =< x < 92 | 4 |
| 90 =< x < 91 | 5 |
| x < 90 | 10 |

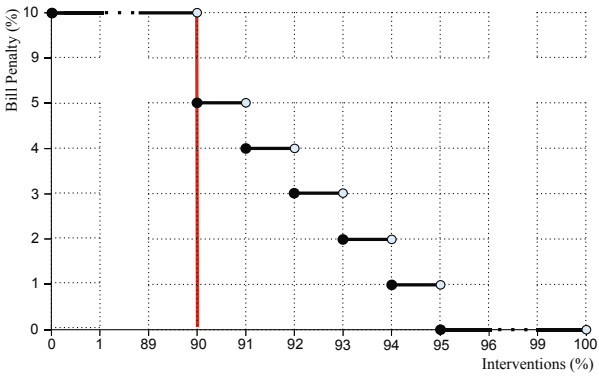



**Fig. 2.** Example ARG-1: Penalties for the Andalusian Regional Government

## 2.3 GNWT SLA

The Government of the Northwest Territories (GNWT) of Canada outsources the IT support. Specifically, the demanded services include issues related to: reporting, user support, problem correction, application enhancement, process

| SLA: Urgent_Interventions > 95 | |
|---|---|
| Measurement | % Penalty |
| 94 =< x < 95 | 1 |
| 93 =< x < 94 | 2 |
| 92 =< x < 93 | 3 |
| 91 =< x < 92 | 4 |
| 90 =< x < 91 | 5 |
| x < 90 | 10 |

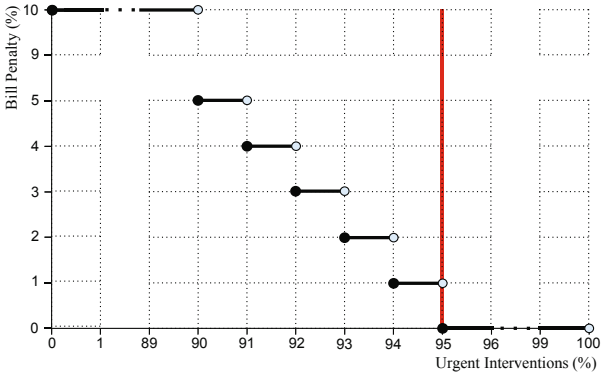



**Fig. 3.** Example ARG-2: Penalties for the Andalusian Regional Government

and application improvement, and other services. They provide a template for establishing an SLA with an external vendor providing the mentioned kind of IT support with the desired service levels and penalties and rewards for the parties.

Four examples of terms with at least a penalty or a reward have been extracted from its SLA template[5]. In the example GNWT-1 included in Figure 4 the GNWT demands the delivery of quarterly reports. The government must receive such reports not less than five days before scheduled review meetings under a penalty of 5% of monthly invoice for the IT support provider.

Example GNWT-2 of Figure 4 depicts specific times for different milestones that take place in the resolution of problems that have made a critical application function unusable or unavailable and no workaround exists (severity 1 code). Specifically, an initial response should be received within 15 minutes, an estimation response should be ready in 2 hours, subsequent responses are expected every 30 minutes, and the problem must be resolved within 4 hours. In this case, a reward for the provider applies if all problems are resolved in less than 2 hours, and a penalty for the provider applies if any of them is resolved in more than 4 hours. An additional clause rewarding than no problem is older than 60 days is included as example GNWT-3 of Figure 4 shows. Note that in previous examples of Figures 1, 2, and 3 just a penalty for a party is established and thus we consider them as *half-compensated* terms. However, in this term and other such as the example GNWT-4 of Figure 4 not only penalties for the provider are established (cf. Figure 5a), but also rewards (cf. Figure 5b). In our proposal we tackle a *full-compensated* joint modeling for penalties and rewards.

This SLA also includes a term relating the scheduled project delivery and the real project delivery that is shown in example GNWT-4 of Figure 4. This term includes a reward for the provider if the elapsed days until delivery is less than 20% lesser than planned but also a penalty for the provider if the elapsed days until delivery is exactly 20% greater than planned (cf. 120% value in Figure 6).

[5] Available at `http://www.fin.gov.nt.ca/ocio/sim/sdlc/3/resources/sla.htm`

| Type | Measurement | Penalty |
|---|---|---|
| **Quarterly Status Report** | Delivered at quarterly intervals and not less than five business days before scheduled review meeting | 5% of monthly invoice |

Example GNWT-1

| Severity Code | Initial Response | Estimation Response | Subsequent Responses | Resolution |
|---|---|---|---|---|
| **1** | 15 minutes | 2 hours | Every 30 min. | 4 hours |

| Type | Measurement | Reward | Penalty |
|---|---|---|---|
| **Severity 1 Resolution** | All Severity 1 problems are resolved in less than 2 hours. | 10% of monthly fees | NA |
| | One or more Severity 1 problems are resolved in over 4 hours. | NA | 10% of monthly fees |

Example GNWT-2

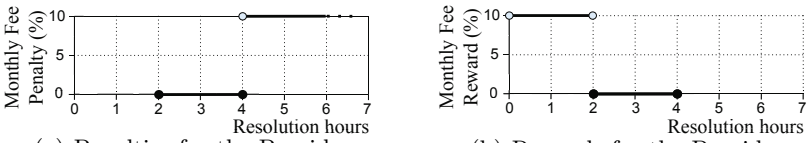
| Type | Measurement | Reward | Penalty |
|---|---|---|---|
| **Maximum Problem Aging** | No problem is older than 60 days. | 5% of monthly fees | NA |

Example GNWT-3

| Type | Measurement | Reward* | Penalty |
|---|---|---|---|
| **Project Delivery** | Total elapsed days until delivery is more than 20% greater than planned. | NA | 10% of the amount invoiced for the project. |
| | Total elapsed days until delivery is 20% less than planned. | 5% of the amount invoiced for the project. | NA |

Example GNWT-4

**Fig. 4.** Compensations actions extracted from the SLA of GNWT




(a) Penalties for the Provider. (b) Rewards for the Provider.

**Fig. 5.** Penalties and rewards extracted from the example GNWT-2

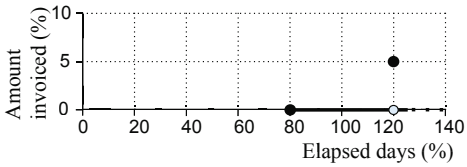



**Fig. 6.** Penalties extracted from the example GNWT-4

In this case the penalty could be wrong, because it would apply for any value more than 20% greater than planned and not only such exact value. Such a problem would be tackled in our proposal.

## 3 Compensation Functions

Compensation functions are defined over services properties in the context of a guarantee satisfied by a guarantor to a beneficiary. Specifically, they associate two types of compensations depending on the subject and recipient of the compensations: on the one hand, a penalty represents a compensation from the guarantor to the beneficiary and, on the other hand a reward represents a compensation from the beneficiary to the guarantor.

In this section we formalize the concept of Compensation Function in order to analyse some interesting properties that would support a consistency check of compensable guarantees. Following we present the formalization by means of a set of supporting core definitions:

**Definition 1 (Service Property Values).** *The set $SP_{sp}$ denotes the set of all possible values of a service property sp ($SP_{sp} = \{v_1, ..., v_n\}$).*

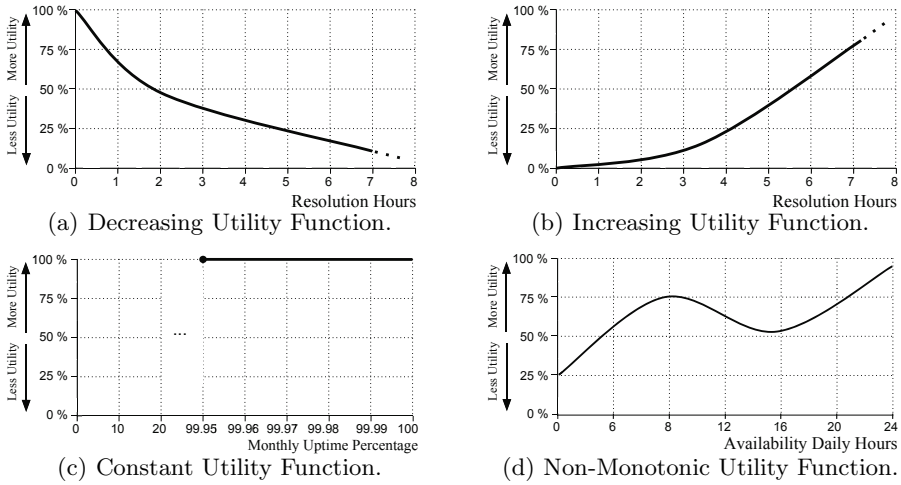
In the examples of Section 2 we find several service properties such as, MUP, interventions, and urgent interventions, with the following finite set of service property values $SP_{sp} = \{0, ..., 100\}$; and others with an infinite set of service property values such as, resolution hours, or elapsed days $SP_{sp} = \{0, ..., \infty\}$.

**Definition 2 (Utility Function).** *An Utility Function for a certain service property sp, denoted by $UF_{sp}$, is a function from SP to $\mathbb{R}$ that associates a utility to each of the values; i.e. it defines which service properties values $SP_{sp}$ are more interesting for a given party.*

Utility functions are typically complementary in case of guarantor and beneficiary. Since they are normally private, we should guess the utility function behind the examples of Section 2. For instance, Figure 7 includes two utility functions for the resolution hours service property of GNWT-2 example (described in Figure 4), an utility function for Availability warranty from Amazon EC2 and an utility function for availability daily hours which aims optimizing different customer goals, such as a common office time or fully 24 hours per day availability (from horizontal demand in [3]). As shown in the Figure, the parties may define different kinds of utility functions, namely, decreasing, increasing, constant, or non-monotonic.

**Definition 3 (Utility Precedence).** *Let $v_1$ and $v_2$ be values of the service property values $SP_{sp}$ of a service property sp, and $UF_{sp}$ a utility function defined on the same service property; a precedence relation called utility precedence is defined on $SP_{sp}$ by $UF_{sp}$. Thus, we denote that $v_1$ is less interesting than $v_2$ by $v_1 \prec v_2$.*

For instance, the example of Figure 7(a) may represent the utility of the beneficiary of the GNWT-2 (described in Figure 4). In the utility precedence defined for such a utility function, the higher value of resolution hours, the less interesting for the beneficiary (e.g. $4 \prec 2$). On the other hand, the example of Figure 7(b) may represent the utility of the guarantor in the same scenario.




(a) Decreasing Utility Function. (b) Increasing Utility Function.

(c) Constant Utility Function. (d) Non-Monotonic Utility Function.

**Fig. 7.** kinds of utility functions

In this case, the utility precedence defined for such a utility function establishes that a lesser value of resolution hours is less interesting for the guarantor (e.g. $2 \prec 4$).

**Definition 4 (Compensation Function).** *A compensation function for a given service property sp, denoted by $CF_{sp}$, is a function from SP to $\mathbb{R}$ that associates a compensation to each of the values. Similarly to utility functions, the compensation functions can be either decreasing, or increasing, or constant, or non-monotonic.*

As a normalized convention that is aligned with related works [10,11] we establish a positive compensation as penalties (that should be compensated from the guarantor to the beneficiary) and negative compensations as rewards (i.e. beneficiary should compensate guarantor).

Figure 8 shows an example of increasing compensation function taken from the example GNWT-2. The function denotes: (1) a penalty for the guarantor if the problems are solved in more than 4 hours; (2) a reward for the guarantor if problems are solved in less than 2 hours; and (3) no compensation applies in problems are solved from 2 to 4 hours, inclusive.

It is important to note that compensation functions could only include penalties or rewards. For instance, Examples ARG-1 and ARG-2 of Figures 2 and 3 just include penalties but not rewards.

**Definition 5 (Compensation Regions).** *A compensation function for a given service property sp $CF_{sp}$ defines up to three compensation regions, namely: penalized, rewarded, and neutral.*

$$CF(x) = \begin{cases} 10\% & \text{if } x > 4 \\ 0\% & \text{if } x \in [2..4] \\ -10\% & \text{if } x \in (0..2) \end{cases}$$

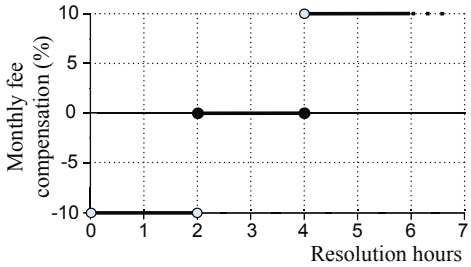



**Fig. 8.** Compensation function corresponding to example GNWT-2

$$\begin{aligned} Penalized(CF_{sp}) &= \{v_i \in SP_{sp} \cdot CF_{sp}(v_i) > 0\} \\ Neutral(CF_{sp}) &= \{v_i \in SP_{sp} \cdot CF_{sp}(v_i) = 0\} \\ Rewarded(CF_{sp}) &= \{v_i \in SP_{sp} \cdot CF_{sp}(v_i) < 0\} \end{aligned}$$

Figure 9 shows these three potential subsets. Thus, $\forall v_i < a$ in Figure 9 $v_i$ is a rewarded value. $\forall v_i > b$ in Figure 9, $v_i$ is a penalized value. And $\forall v_i, a \leq v_i \leq b$ in Figure 9, $v_i$ is a neutral value. Since omission means a lack of compensation in natural language, when there is not explicit definition of a penalty or a reward, we consider a unique subset of neutral values.

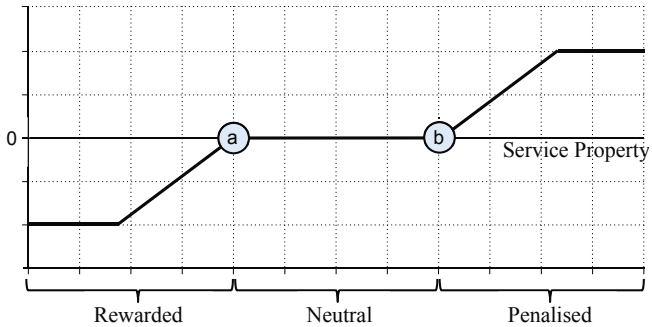



**Fig. 9.** A generic example of increasing compensation function

### 3.1 Validity of Compensation Functions

We formalize the validity of compensation functions as a property related with the consistence in terms of utility and the saturability of compensations. Consequently, for the sake of the clarity we divide this validity in two separate subproperties (Consistent and Saturated) that are defined next.

*Property 1 (Consistent$_{CF}$). A compensation function $CF_{sp}$ is said to be consistent if the compensation for a less interesting value of service property is less*

*or equal than the compensation for a more interesting value according with the utility precedence defined by the Utility function of the beneficiary. Thus, compensation and utility functions could be covariant if both are increasing or decreasing functions; and contravariant otherwise.*

$$Consistent_{CF}(CF_{sp}) \iff \forall v_1, v_2 \in SP \cdot v_1 \preceq v_2 \Rightarrow CF_{sp}(v_1) \geq CF_{sp}(v_2)$$

Figures 1 and 8 depict a decreasing and an increasing consistent compensation function, respectively. However, example GNWT-4 depicted in Figure 10 is not consistent with the probable utility precedence derived from a monotonically increasing utility function of the beneficiary.

*Property 2 (Saturated). Let $CF_{sp}$ a compensation function, it is said to be saturated if there exist two values ($v_{min}$ and $v_{max}$) for the service property, that delimit the higher compensation, either penalty or reward.*

$$\begin{aligned} Saturated(CF_{sp}) \iff \quad & \forall v_i \in SP, \exists v_{max}, v_{min} \in SP \cdot \\ & CF_{sp}(v_i) \leq CF_{sp}(v_{max}) \wedge CF_{sp}(v_i) \geq CF_{sp}(v_{min}) \end{aligned}$$

This property prevents the definition of infinite compensations that should be avoided in real scenarios; consequently (and as it was expected) all the different examples of compensation functions described in the paper are saturated since they correspond with real world examples.

At this point, based on the previous properties we can develop a further formalization of the validity:

*Property 3 ($Valid_{CF}$). Let $CF_{sp}$ a compensation function, it is said to be valid if it is both consistent and saturated.*

$$Valid_{CF}(CF_{sp}) \iff Consistent_{CF}(CF_{sp}) \wedge Saturated(CF_{sp})$$

The compensation functions of Figures 1 and 8 are valid. On the contrary, Figure 10 shows the compensation function of GNWT-4 example that is not consistent and therefore, not valid.

## 4 Compensable SLA

SLAs specify the rights and responsibilities of the different parties involved in a service consumption. Amongst others, a key element part of SLAs are the guarantee terms [1] that are typically defined over a service level objective. Based on this conceptualization, we coin the concept of Compensable Guarantees to those which include a compensation function and subsequently, Compensable SLAs represent a type of SLA that includes at least one Compensable Guarantees.

In this context, while a guarantee term is defined around the idea of a SLO that should be guaranteed by a guarantor to a beneficiary (e.g. *Response Time* $<$ *100ms* or *Monthly Uptime Percentage* $\geq$ *99.95%*); a compensable guarantee term

$$CF(x) = \begin{cases} 5\% & \text{if } x = 120 \\ 0\% & \text{if } x \in [80..120) \\ 0\% & \text{if } x \in (120..\infty) \\ -10\% & \text{if } x \in (0..80) \end{cases}$$

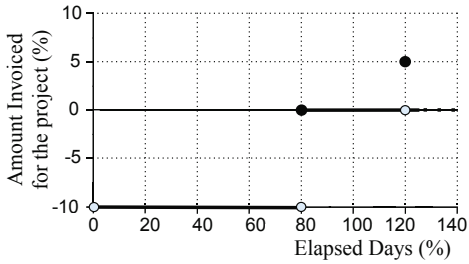



**Fig. 10.** Not valid Compensation function of example GNWT-4

includes the concept of penalties and rewards to compensate the underfulfillment or overfulfillment of the SLO, respectively.

Based on the formalization of the Compensation Function presented in Section 3, next we provide the formal definitions of Compensable Guarantees and SLAs. In addition, we introduce a set of properties to analyse Compensable Guarantees to extend the notion of consistence and validity in SLAs (presented in [14]) in order to ground the development of supporting tooling to help in the modeling of valid Compensable SLAs.

**Definition 6 (Service Level Objective).** *An $SLO_{sp}$ is a valid*[6] *assertion defined over a service property sp.*

Examples of SLOs include *Response Time < 100ms* or *Monthly Uptime Percentage ≥ 99.95%.*

**Definition 7 (Fulfillment Regions).** *The assertion defined by an $SLO_{sp}$ determines two regions over the values of the service properties they dealt with, namely fulfilled and unfulfilled. This regions are delimited by the threshold $Th_{SLO}$.*

$$\begin{aligned} Fulfilled(SLO_{sp}) &= \{v_i \in SP_{sp} \cdot SLO_{sp}\} \\ Unfulfilled(SLO_{sp}) &= \{v_i \in SP_{sp} \cdot \neg SLO_{sp}\} \end{aligned}$$

**Definition 8 (Compensable Guarantee).** *A compensable guarantee $CG_{sp}$ is a two-tuple of the form $(SLO_{sp}, CF_{sp})$ in which $SLO_{sp}$ is a service level objective and $CF_{sp}$ is a compensation function that are defined over the same service property sp.*

$$CG_{sp} = < CF_{sp}, SLO_{sp} >$$

Figure 11 shows a typical compensation function (defined over an increasing utility precedence) that depicts the relationships between the fulfillment regions delimited by the SLO and the compensation regions defined by the CF (c.f. Section 3). Moreover, as shown in this figure, it is important to highlight that

[6] A formal validity criteria for SLOs is presented in [14].

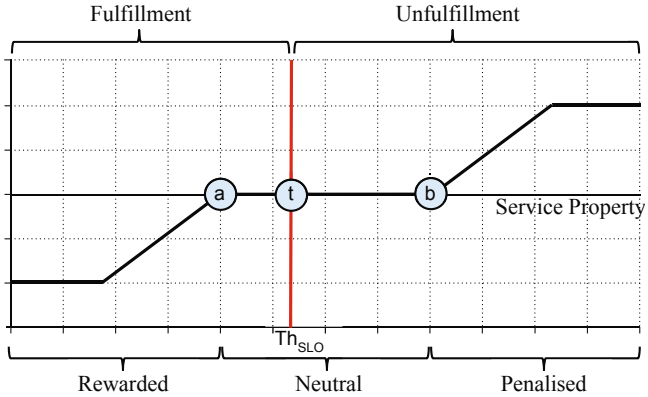



**Fig. 11.** A generic example of compensable guarantee showing the fulfillment and compensable regions

fulfillment regions are not necessary coupled with compensation regions; specifically, figure exemplifies a case having neutral service properties values between $a$ and $b$ without any compensation: service properties values between $t$ and $b$ are unfulfilled but not penalized, and similarly service properties values between $a$ and $t$ are fulfilled but not rewarded. In addition, figure shows how threshold $Th_{SLO}$ delimits the fulfillment from the unfulfillment values.

Following our formalization of compensable guarantees, Figures 12, 13, and 14 present our modeling of the compensable guarantee terms identified in the three real world SLAs found in the scenarios presented in Section 2. Each example, corresponds with a guarantee term showing the specific compensation function (as a black line) along with the $Th_{SLO}$ (as a solid vertical line) derived from the SLO; in case there is no SLO explicit, we have inferred a threshold ($Th_{Gtor}$ or $Th_{Ben}$) depending on whether the SLA was specified by the guarantor (AWS EC2 scenario) or the beneficiary (GNWT and Telecomm. SLA scenarios); these inferred thresholds are depicted as discontinuous lines.

### 4.1 Validity of Compensable Guarantees

The validity a compensable guarantee is derived from the coherence between its SLO and its CF. Consequently, in this section we formalize the validity criteria that is established upon the concept of validity for compensation functions and the consistency between its SLO and CF. This formalization extends the notion of validity and consistence presented in [14].

*Property 4 (Consistent). A compensable Guarantee $CG_{sp}$ is said to be consistent if there is at least one fulfilled and neutral value (that would be $TH_{SLO}$) and the fulfillment regions are coherent with compensation regions: the fulfilled values are either neutral or rewarded and, complementary, the unfulfilled values are either neutral or penalized.*

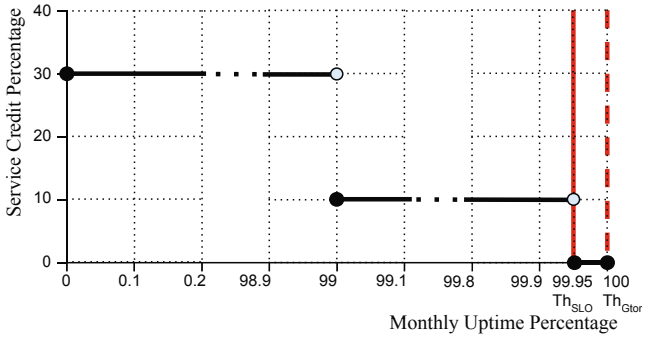
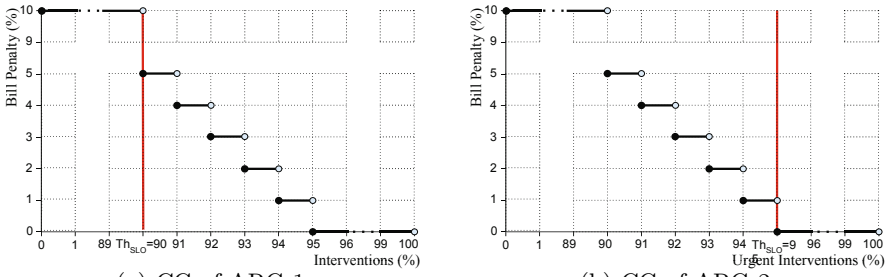
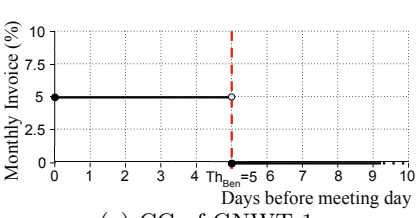



**Fig. 12.** Example from AWS EC2 SLA

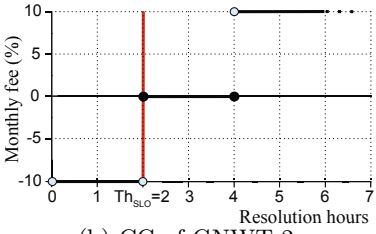



(a) CG of ARG-1. (b) CG of ARG-2.

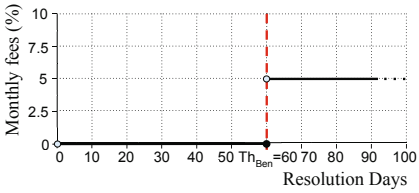
**Fig. 13.** Compensable Guarantees of the ARG examples



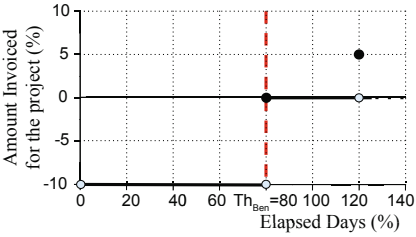

(a) CG of GNWT-1.




(b) CG of GNWT-2.




(c) CG of GNWT-3.




(d) CG of GNWT-4.

**Fig. 14.** Compensable Guarantees of GNWT examples

$$Consistent_{CG}(CG_{sp}) \iff Fulfilled(CGsp.SLO) \cap Neutral(CGsp.CF) \neq \varnothing \wedge$$
$$Fulfilled(CGsp.SLO) \subset Neutral(CGsp.CF) \cup Rewarded(CGsp.CF) \wedge$$
$$Unfulfilled(CGsp.SLO) \subset Neutral(CGsp.CF) \cup Penalized(CGsp.CF)$$

In this point, based on the previous properties we can develop a further formalization of the validity for a compensable guarantee:

*Property 5 (Valid). Let $CG_{sp}$ a compensable guarantee, it is said to be valid if it is consistent and it contains a valid CF.*

$$Valid_{CG}(CG_{sp}) \iff Valid_{CF}(CG_{sp}.CF) \wedge Consistent_{CG}(CG_{sp})$$

**Table 1.** Comparative analysis of examples

| | $Consistent_{CF}$ | $Saturated_{CF}$ | $Valid_{CF}$ | $Consistent_{CG}$ | $Valid_{CG}$ |
|---|---|---|---|---|---|
| **AWS EC2** | ✓ | ✓ | ✓ | ✓ | ✓ |
| **ARG-1** | ✓ | ✓ | ✓ | ✗ | ✗ |
| **ARG-2** | ✓ | ✓ | ✓ | ✓ | ✓ |
| **GNWT-1** | ✓ | ✓ | ✓ | ✓ | ✓ |
| **GNWT-2** | ✓ | ✓ | ✓ | ✓ | ✓ |
| **GNWT-3** | ✓ | ✓ | ✓ | ✓ | ✓ |
| **GNWT-4** | ✗ | ✓ | ✗ | ✓ | ✗ |

Table 1 presents a comparative study of the different examples (Figures 14-12) over the different properties defined. Based on this analysis we highlight the following insights:

- Example ARG-1 of Figure 13 includes a SLO that represents a $Th$ with a penalty; consequently, this example does not satisfy the $Consistent_{CG}$ property with an invalid threshold and therefore the guarantee is not $Valid_{CG}$.
- GNWT-4 of Figure 14 represents a case (elapsed days) where we can induce a monotonic increasing Utility Function for the beneficiary; based on this utility precedence, the compensation function is not consistent so we can infer that this case represents a human mistake in the SLA derived from the usage of natural language.

## 5 Related Work

In this section we extend our initial analysis with a further exploration over relevant research papers that include the idea of SLAs with penalties and/or rewards. Table 2 shows a comparative study of the different properties identified over the examples of the papers.

The proposal of Leitner et al. in [10] formalizes the problem of finding the optimal set of adaptations, which minimizes the total costs arising from SLA violations and the adaptations to prevent them. In this work, a model for penalty functions is presented; this formalizations has been the starting point of our motivational scenario description presented in Section 2 and consequently, our approach represents an extension to this model in order to develop a complete formalization for Compensable Guarantees and SLAs. Based on our model, we have studied 4 examples (page 2) included presented in [10] relating to the cost of violations of one service property, namely: time to offer, order fulfillment time, process lead time, and cost compliance. In [11] the same authors present an approach for optimally scheduling incoming requests to virtual computing resources in the cloud, so that the sum of payments for resources and loss incurred by SLA violations is minimized. The studied example (page 3) includes a linear penalty function with two point of discontinuities. The example relates the penalty with a service property representing the duration of requests to virtual computing resources in the cloud.

**Table 2.** Properties analysis in Examples found in the papers

| | | $Consistent_{CF}$ | $Saturated_{CF}$ | $Valid_{CF}$ | $Consistent_{CG}$ | $Valid_{CG}$ |
|---|---|---|---|---|---|---|
| Leitner et al. [10] | TimeToOffer<=2 (Pag.2) | ✓ | ✓ | ✓ | ✓ | ✓ |
| | OrderFulfillment<=5 (Pag.2) | ✓ | ✓ | ✓ | ✓ | ✓ |
| | ProcessLeadTime<=6 (Pag.2) | ✓ | ✓ | ✓ | ✓ | ✓ |
| | CostCompliance<=5 (Pag.2) | ✓ | ✓ | ✓ | ✓ | ✓ |
| Leitner et al. [11] | SLA Cost = Penalties (Pag.3) | ✓ | ✓ | ✓ | ✓ | ✓ |
| Buco et al. [4] | Penalties (Pag.12) | ✓ | ✓ | ✓ | n/a | n/a |
| | Penalties & Reward (Pag.18) | ✗ | ✗ | ✗ | n/a | n/a |
| Grabarnik et al. [7] | Penalties & Reward (Pag.7) | ✓ | ✓ | ✓ | n/a | n/a |
| Rana et al. [15] | ExecutionTime (Pages 6-7) | ✓ | ✗ | ✗ | n/a | n/a |

Other examples taken from relevant related works are the following: Buco et al. propose in [4] an SLA management system, called SAM that provides penalties in a Service Level Management process. In the first studied example (page 12) the compensation function relates some penalties with a service property denoting the alert time for SLA managers. In the second example some penalties and rewards are specified depending on the service level fulfillment of the SLAs; these last example define overlapping values for the rewards (i.e. same value can have different compensations) and therefore, the compensation function cannot be defined. Grabarnik et al. propose in [7] a model that can be used to reduce total service costs of IT service providers using alternative delivery teams and external service providers. The studied example (page 7) includes penalties and rewards for a service property that represents the process execution time. Rana et al. identifies in [15] how SLOs may be impacted by the choice of specific

penalty clauses. From such a work we have studied an example (pages 6-7) that relates the penalty of different levels of service execution time. It is important to note that the compensation function does not meet the saturation property and therefore it does not fulfill the validity property we defined in Section 4. In case there is no SLO explicit (Buco et al., Grabarnik et al. and Rana et al.) the validity and consistency of the compensable guarantee check is non applicable (n/a).

In our approach we leverage utility functions of service properties to analyze the consistency of compensations. In business studies, utility function models are also analyzed as they are strongly dependent on customer preferences and behaviour. [3] describes a business scenario with cost, customer expectations and reputation variables where reward function follows a non-monotonic behaviour (based on satisfying preferences from different customers). Similarly, Fenghui Ren et al. analyze in [16] how utility function is obtained from customer objective function (i.e., customers timetable preferences affect how transactions distribute through commercial opening hours). In temporal-aware web services matchmaking where the QoS varies on specific time periods, utility function models are also leveraged to find the optimal service offer for a given demand with validity periods [12]. In [13] we proposed how to specify utility functions and time periods within the WS-Agreement specification [1]. Finally, utility functions are also composed and integrated within comprehensive preference models enabling the combination of several ranking mechanisms [6].

Angelov et al. propose in [2] a formal representation for contracts to detect and solve different kinds of conflicts. Although the proposed contracts representation supports penalties and rewards by means of reparation clauses, they are not validated against utility functions as proposed in the current paper.

## 6 Conclusions and Future Work

In this paper we characterize the concept of Compensable SLA that includes a definition of guarantees with penalties and rewards for involved parties when the Service Level Objective (SLO) is underfulfilled or overfulfilled. We motivate our proposal upon the study of three real world SLAs ranging from human-driven services to computing services and define a model to formally express Compensable SLAs by conceptualizing the Compensation Function as the appropriate artifact to consistently combine penalties and rewards in the context of a SLO. In order to support the modeling and analysis of Compensable SLAs, we extend the notion of validity and consistence presented in [14] with a formal definition of properties.

Moreover, we validate our formalization by modeling our motivational examples and check the different properties defined in order to analyse them. It is important to highlight that based on this study we have identified some inconsistences that could be derived from typical human mistakes when using natural language in the definition of compensable SLAs.

Since a thorough study of the different kinds of real-world SLAs has not been performed, as future work we will analyse other real-world examples that

incorporate more complex situations such as guarantees defined over multiple service properties and non-monotonic utility functions. Examples of this kind of utility functions can be found in other business domains such as logistics, where delivery normally refers to a just-in-time situation so the penalty would be defined not only for a late delivery of the product, but also for an early delivery. Another line of future work will be to extend our analysis tooling framework IDEAS[7] to incorporate to automatically check the different properties studied in this work, in order to detect singular situations and common pitfalls.

## References


1. Andrieux, A., Czakowski, K., Dan, A., Keahey, K., Ludwig, H., Nakata, T., Pruyne, J., Rofrano, J., Tuecke, S., Xu, M.: Web Services Agreement Specification (WS-Agreement) gfd-r.192 (October 2011)
2. Angelov, K., Camilleri, J.J., Schneider, G.: A framework for conflict analysis of normative texts written in controlled natural language. The Journal of Logic and Algebraic Programming 82(5-7), 216–240 (2013)
3. Bar-Isaac, H., Deb, J.: What is a good reputation? career concerns with heterogeneous audiences. International Journal of Industrial Organization 34(0), 44–50 (2014)
4. Buco, M.J., Chang, R.N., Luan, L.Z., Ward, C., Wolf, J.L., Yu, P.S.: Utility computing sla management based upon business objectives. IBM Systems Journal 43(1), 159–178 (2004)
5. del Río-Ortega, A., Resinas, M., Cabanillas, C., Ruiz-Cortés, A.: On the Definition and Design-time Analysis of Process Performance Indicators. Information Systems 38(4), 470–490 (2012)
6. García, J.M., Junghans, M., Ruiz, D., Agarwal, S., Cortés, A.R.: Integrating semantic web services ranking mechanisms using a common preference model. Knowl.-Based Syst. 49, 22–36 (2013)
7. Grabarnik, G., Ludwig, H., Shwartz, L.: Management of service process qos in a service provider - service supplier environment. In: The 9th IEEE Int. Conf. on Ent. Comp., E-Commerce, and E-Services (CEC/EEE), pp. 543–550 (July 2007)
8. International Working Group on Cloud Computing Resiliency (IWGCR). Downtime statistics of current cloud solutions (2012)
9. Kronz, A.: Managing of process key performance indicators as part of the aris methodology. In: Corporate Performance Management: Aris in Practice, pp. 31–44. Springer, Heidelberg (2006)
10. Leitner, P., Hummer, W., Dustdar, S.: Cost-based optimization of service compositions. IEEE Transactions on Services Computing 6(2), 239–251 (2013)
11. Leitner, P., Hummer, W., Satzger, B., Inzinger, C., Dustdar, S.: Cost-efficient and application sla-aware client side request scheduling in an infrastructure-as-a-service cloud. In: 2012 IEEE 5th International Conference on Cloud Computing (CLOUD), pp. 213–220 (June 2012)
12. Martín-Díaz, O., Ruiz-Cortés, A., Durán, A., Müller, C.: An approach to temporal-aware procurement of web services. In: Benatallah, B., Casati, F., Traverso, P. (eds.) ICSOC 2005. LNCS, vol. 3826, pp. 170–184. Springer, Heidelberg (2005)


[7] Available at `http://www.isa.us.es/IDEAS/`

13. Müller, C., Martín-Díaz, O., Ruiz-Cortés, A., Resinas, M., Fernández, P.: Improving temporal-awareness of WS-agreement. In: Krämer, B.J., Lin, K.-J., Narasimhan, P. (eds.) ICSOC 2007. LNCS, vol. 4749, pp. 193–206. Springer, Heidelberg (2007)
14. Müller, C., Resinas, M., Ruiz–Cortés, A.: Automated analysis of conflicts in ws–agreement. IEEE Transactions on Services Computing, TSC (2013)
15. Rana, O.F., Warnier, M., Quillinan, T.B., Brazier, F., Cojocarasu, D.: Managing violations in service level agreements. In: Grid Middleware and Services Chapter Title - Managing Violations in Service Level Agreements, pp. 349–358 (2008)
16. Ren, F., Zhang, M.: Bilateral single-issue negotiation model considering nonlinear utility and time constraint. Decision Support Systems 60(0), 29–38 (2014)