A Service Level Agreement Driven Framework to Customise Cloud Service Billing*

José María García, Octavio Martín-Díaz, Pablo Fernandez, Carlos Müller, and Antonio Ruiz-Cortés

Universidad de Sevilla, Seville, Spain
{josemgarcia,omartindiaz,pablofm,cmuller,aruiz}@us.es

Abstract. Cloud service providers offer to their customers a variety of pricing policies, which range from the simple, yet widely used pay-as-you-go schema to complex discounted models. When executing the billing process, stakeholders have to consider usage metrics and service level objectives in order to obtain the correct billing and conform to the service level agreement in place. The more metrics, discount and compensations rules are added to the pricing schema, the more complex the billing generation results. In this paper we present a monitoring-based solution that enables the dynamically definition of both service level objectives and discount rules, so that providers can customise the billing generation process in terms of the service level agreement they offer. We validate our proposal in a real-world scenario, introducing a micro-service based software solution deployed in a Kubernetes cluster.

Keywords: Billing · Pricing · Cloud Services · Monitoring · Service Level Agreements.

1 Introduction

The cloud computing market is evolving towards highly customisable pricing schemata, where customers can choose among many different options when purchasing services, including discount policies depending on actual usage [7]. Furthermore, service level agreements usually introduce additional conditions to take into account within the billing process. Thus, once the services are provisioned, the monitoring data is used to generate appropriate bills that are charged according to a set of rules described in the agreement. Figure 1 shows a simplification of this billing process.

Both providers and customers are interested in the monitoring and billing process from different perspectives. On the one hand, providers clearly need to bill their customer according to the actual usage of the relevant services. On the other hand, customers can use the same billing rules to check that the billing

* This work has been partially supported by the EU Commission (FEDER), Spanish and Andalusian R&D&I programmes under grants TIN2015-70560-R, RTI2018-101204-B-C21, TIN2016-81978-REDT, and P12-TIC-1867.

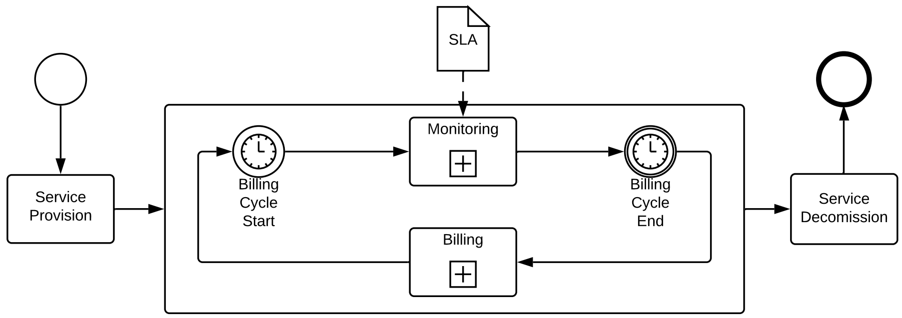

Fig. 1. Simplified service billing process.

carried out by the provider conforms to their agreement, or even to estimate or simulate the billing for subsequent periods [4]. In this paper, we mainly focus on the billing generation from the providers perspective. Our proposal is based on the definition of service level agreements (SLAs) augmented with pricing and billing policies, so that the latter can be based on metrics computed to monitor the former. Thus, our solution can dynamically generate billing information, allowing the provider to define and change pricing and discount rules depending on specific application scenarios.

The rest of the paper is structured as follows. First, in Sec. 2 we present the real-world scenario that motivates this work. Then, Sec. 3 presents our proposal to dynamically generate bills with respect to service level and discount rules. We validate our solution in Sec. 4, applying it to the motivating scenario. We discuss in Sec. 5 related work regarding cloud pricing models and billing mechanisms. Finally, Sec. 6 concludes the paper, discussing our results and identifying future work.

2 Motivating scenario

The contributions presented in this work are motivated in the context of a real scenario. Specifically, our scenario corresponds to a distributed provision of Software as a Service (SaaS) of educational platforms in the public administration of Andalusia (the biggest region of Spain in terms of size and population). In this context, the administration contracts a SaaS provider to deliver the platforms for a network of over 2000 schools distributed geographically across the region and the provider should set and maintain a flexible infrastructure that can scale to deploy the appropriate number of platform instances. Moreover, from a regulatory perspective, in order to satisfy the privacy requirements the provider should maintain a private cloud infrastructure in their data centers and shared amongst their customers.

In order to address the requirements, the initial infrastructure model established in this project was to settle a fixed reserved physical infrastructure for the scenario that dynamically deploys platforms as they are required by the different centers. However, this initial model proved to be unsatisfactory for both the administration and providers due to the billing model established. On the one hand, from the perspective of the administration, while there was a dynamic pricing for the administration based on the number of platforms used on every day, there was a minimum base cost derived from the maintenance and operation of the fixed reserved infrastructure. On the other hand, from the provider perspective, there was a continuous under-usage of the infrastructure that can not be used for other customers in order to generate new business possibilities.

To overcome this limitations, both administration and provider acknowledge to explore an alternative pricing model to migrate the reserved physical infrastructure to a data-center managed by the provider as a private cloud for its customers. By using this shared infrastructure model, it is still feasible to reserve parts of the infrastructure, but during the periods where usage load by the administration are low (e.g. on vacations) the provider can allocate more computing resources to other customers with the subsequent gain, while the administration gets a discount for the low usage load. In particular, in this scenario, an agreement was settled defining a set of rewards for the administration in case the reserved resources (in terms of computing, memory or disk) are not used during a certain period.

However, it is important to highlight that the specific requirements of scalability and privacy impose also significant challenges in terms of the billing process. Specifically, due to the dynamic nature of a cloud infrastructure, it is required to have a flexible billing management that: (i) can be customised to express the rules agreed by the different parties and (ii) provides an analytic platform to develop the actual calculation of the bills in an automated fashion based on metrics gathered from the live system.

3 Customisable Billing Generation Process

In order to allow the level of customization required for the dynamic billing scenario described before, we propose a billing generation framework based on SLAs descriptions, which are augmented with pricing model specifications. For a concrete service, the augmented SLA description contains not only the usual service level objectives, possibly including compensations, but also pricing and discount rules, which serve the basis for the billing generation process. In the following sections, we exemplify our proposal using the results of the *POETISA* project, which was conducted to provide a solution to the challenges identified in Sec. 2.

3.1 Pricing model specification

Our proposed pricing model is shown in Fig. 2 by means of an excerpt of the UML model of iAgree SLA language[1]. In the model, *Pricing* appears as a first-

[1] http://iagree.specs.governify.io/Specification/#iagree-specification

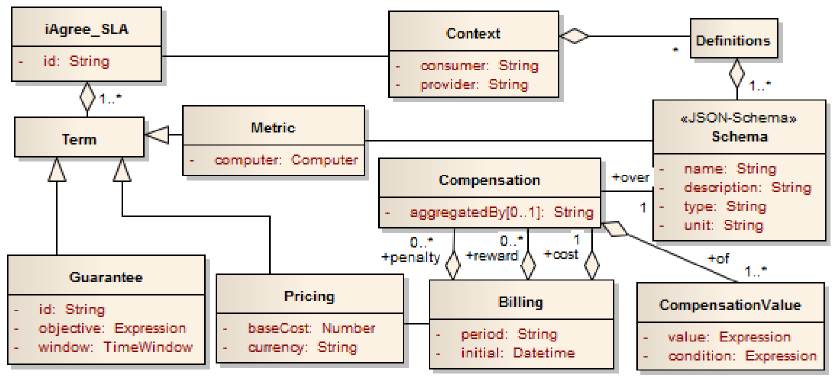

Fig. 2. UML model excerpt of iAgree elements including Pricing.

class iAgree term as others like *Guarantees* and *Metrics*. A Metric term includes its description, datatype, unit, the value thresholds, and a computer URL that references an API-REST interface that is required to monitor the current metric value (e.g. the average memory consumption is computed by the URL provided at line 38 in the POETISA SLA excerpt of Fig. 3). A Guarantee term specifies the service level objective that must be considered in a time window (e.g. `Uptime` $\geq$ `99.00` at line 43 in Fig. 3). A Pricing term informs about the referred currency and a base cost. The base cost can be modified in a specific billing period by considering a set of compensation values over the cost schema in a similar way as proposed for guarantees in [10]. For instance, in Fig. 3, the Daily Platform Cost (DPC) of `0.78` (line 19) refers to the base cost per educational Moodle-based platform, while the final cost is calculated depending on the Number of Days of the Month (NDM) and the Number of Moodle platforms (NM) (line 24). In case NM is less than `150` (line 22), the minimum number of Moodle platforms to be considered is that same value (line 21). Furthermore, within a billing period the cost can be also modified either by rewards or penalties. In the example of Fig. 3, a monthly billing discount of €10 (line 30) is applied if the Average Memory Consumption (AMC) is less than 30 Gb (line 31).

3.2 Billing Generation

From the pricing and billing rules specified using the format described above, we derive a set of rules that are evaluated using a rule engine [12]. Thus, our proposal first analyses the iAgree specification in order to extract the metrics information by using each computer URLs declared. We query those computers components specifying the relevant time window to generate the corresponding bill. The values obtained are injected into the conditions and values of the pricing and billing rules (see Fig. 3).

Together with the metrics information, our proposal transforms the iAgree conditions and values into a rule with the format `condition` $\rightarrow$ `action`. Then,

```
id: POETISA_SLA:
context:
  provider: https://www.isa.us.es/
  consumer: http://www.juntadeandalucia.es/
  ...
  definitions:
    schemas:
      BillingDiscount:
        description: Percent to be discounted at the next monthly bill
        type: double; unit: '%'
terms:
    pricing:
        currency: EUR
        billing:
            period: monthly
            initial: '2018-04-28T10:35:36.000Z'
            cost:
                over: MonthlyCost
                    baseCost: DPC: 0.78
                of:
                    value: DPC*150*NDM
                    condition: NM < 150
                of:
                    value: DPC*NM*NDM
                    condition: NM ≥ 150
        rewards:
            over: BillingDiscount
            aggregatedBy: sum
            of:
                value: 10
                condition: AMC < 30
            ...
    metrics:
    AMC:
      schema:
        description: Average memory consumption
        type: double; unit: 'GB'; minimum: 0; maximum: 100
      computer: http://10.96.0.100/api/v2/AMC
    ...
    guarantees:
      id: G_Uptime
        of:
            objective: Uptime ≥ 99.00
            window: ...
```

Fig. 3. Pricing and Guarantee terms excerpt of POETISA SLA in *iAgree* syntax.

making use of a rule engine (e.g. json-rules-engine[2]), we can compute discounts and final price according to the iAgree specification. By using this approach, our solution can take advantage of the expressiveness of rule engines to describe complex conditions and generate bills using arbitrary formulas for computing discounts and service costs, in terms of the `action` part of the rules obtained from the analysis and transformation of the iAgree document.

[2] https://www.npmjs.com/package/json-rules-engine

4 Validation

In order to evaluate the suitability of our conceptual solution, we developed a software prototype that was deployed in the context of the motivational scenario as described in Sec. 2.

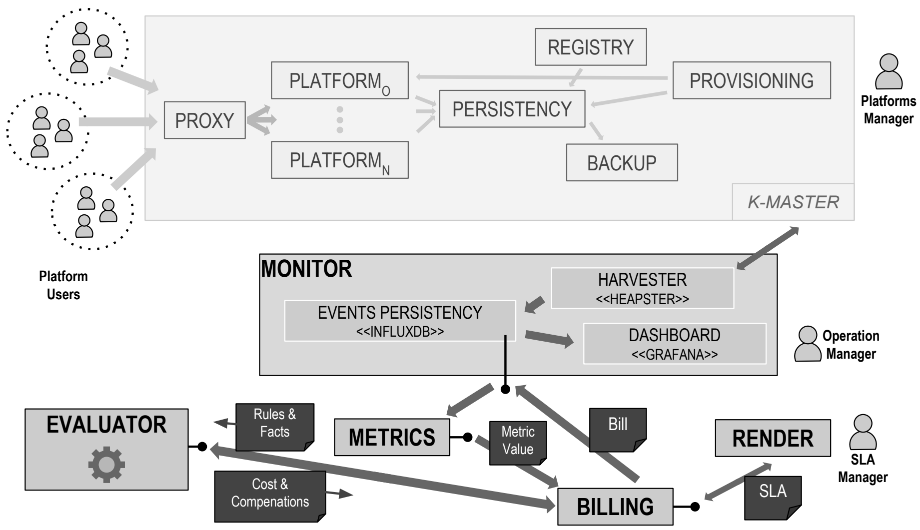

Fig. 4. Prototype Architecture

From an architectural point of view, Fig. 4 depicts the different elements that conform the validation scenario; specifically, this architecture includes an ecosystem of components, that are deployed as part of a Kubernetes cluster and integrated with the preexisting provisioning infrastructure (in the top part of the figure) for educational platforms. Specifically, the prototype consists of the following elements:

- Monitor component integrates Heapster (responsible for gathering events monitored from the cluster), InfluxDB (a persistence store with query capabilities specialised in time-series and events), and Grafana (providing a visual dashboard for service operation). Fig. 5 shows a screenshot of the Grafana dashboard we created.
- Metrics manages the calculation of an specific element that is significant for the SLA, based on the event data harvested from the cluster.
- Billing component includes the parsing and analysis of the SLA and the gathering of metrics in order to generate a set of rules and facts that can be computed by the evaluator component to calculate a bill.
- Evaluator component encapsulates the rule engine to compute an actual cost and compensation for a given set of rules an facts.

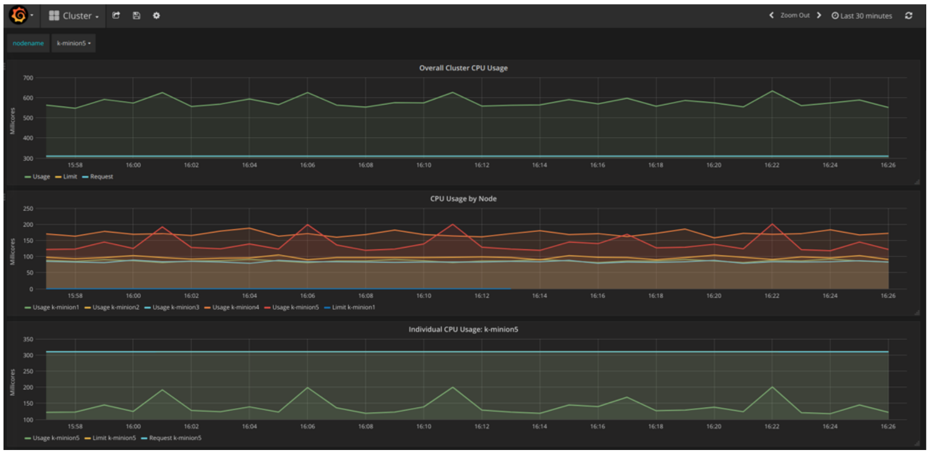

Fig. 5. Screenshot of the Dashboard component.

– Finally, the Render component provides a user interface (as shown in Fig 6) for the management of SLA and billing. Note that in the screenshot presented, a billing has been generated so that objectives and condition rewards are marked in green if met, red otherwise. All of them can be changed in order to simulate variations in the billing process.

5 Related Work

Cloud providers have been adopting different pricing models, which are evolving towards much more dynamic, complex schemes. Only recently there have been some comparative analysis, such as [1, 6, 8, 14, 2]. In general, most providers apply similar variants of pay-as-you-go, subscription, and auction-based pricing schemes. However, these works do not include billing discounts in their study, in spite of their variability when reviewing SLAs offered by cloud providers.

Focusing on billing process, one of most relevant works is CYCLOPS [11], a framework with a modular, micro-services based architecture aimed at developing a dynamic rating, charging and billing for cloud service providers. Rules are a key element for tackling the variability in accounting and billing. Besides CYCLOPS, there are other interesting works: [3, 5, 13] introduce these topics in the context of federated clouds, and [9] introduces some guidelines for providers to make easier billing estimation to customers, in order to avoid discrepancies. Furthermore, there exist commercial tools supporting billing related processes and their optimization, such as jBilling[3], CloudAbility[4], and RightScale[5].

[3] https://www.jbilling.com/
[4] https://www.cloudability.com
[5] https://www.rightscale.com/

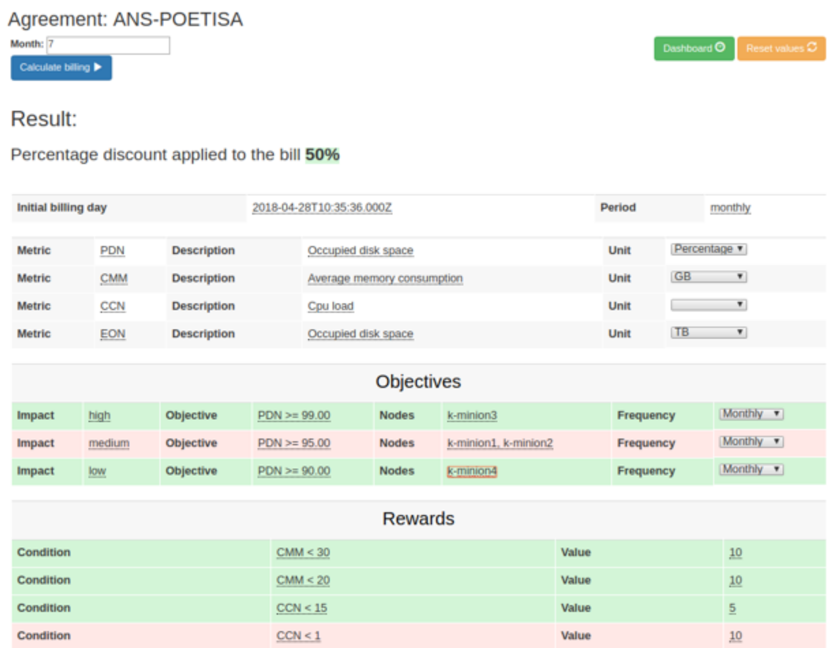

Fig. 6. Screenshot of the Render component.

6 Conclusions

Service billing customisation is a challenging task for cloud providers that need to adapt their pricing and discount rules dynamically, according to business goals and SLAs. Furthermore, service consumers can also benefit from customised billing policies, adapted to their actual needs. Our proposal focus on the definition of said policies, which results in an integrated monitoring and billing process. From a set of customisable metrics and billing rules, our solution generates the corresponding bills, while also considering SLAs enforcement.

As shown in our validation scenario, by means of a customised service billing we can adjust generated bills to achieve both consumers and providers goals. Thus, our solutions allows them to define complex SLAs including discounts based on a set of metrics, providing a monitoring dashboard to analyse the suitability of the policies and obtain evidences to sustain the generated billing information. As future work, we will also support billing validation and estimation, which are related processes mainly relevant to service consumers.

References

1. Al-Roomi, M., Al-Ebrahim, S., Buqrais, S., Ahmad, I.: Cloud Computing Pricing Models: A Survey. International Journal of Grid and Distributed Computing **6**(5), 93–106 (2013). https://doi.org/10.14257/ijgdc.2013.6.5.09

2. Cots Salleras, G.: The impact of Cloud Computing adoption on IT Service Accounting approaches-A Customer Perspective on IaaS Pricing Models. Tech. rep., Universitat Politecnica de Catalunya (2012)
3. Elmroth, E., Márquez, F.G., Henriksson, D., Ferrera, D.P.: Accounting and billing for federated cloud infrastructures. In: 8th International Conference on Grid and Cooperative Computing, GCC 2009 (2009). https://doi.org/10.1109/GCC.2009.37
4. García, J.M., Martín-Díaz, O., Fernandez, P., Ruiz-Cortés, A., Toro, M.: Automated analysis of cloud offerings for optimal service provisioning. In: Service-Oriented Computing: 15th International Conference, ICSOC 2017. vol. 10601 LNCS, pp. 331–339. Springer (11 2017). https://doi.org/10.1007/978-3-319-69035-3_23
5. Harsh, P., Jegou, Y., Cascella, R.G., Morin, C.: Contrail Virtual Execution Platform Challenges in Being Part of a Cloud Federation (Invited Paper). In: Towards a Service-Based Internet. ServiceWave 2011 (2011). https://doi.org/10.1007/978-3-642-24755-2_5
6. Kansal, S., Singh, G., Kumar, H., Kaushal, S.: Pricing Models in Cloud Computing. In: Proceedings of the 2014 International Conference on Information and Communication Technology for Competitive Strategies - ICTCS '14. pp. 1–5. ACM Press, New York, New York, USA (2014). https://doi.org/10.1145/2677855.2677888
7. Martín-Díaz, O., García, J.M., Fernandez, P., Ruiz-Cortés, A.: Automatic Generation of Purchasing Plans for Cloud Services. In: Jornadas de Ciencia e Ingeniería de Servicios (JCIS) 2016. SISTEDES, Salamanca (2016), http://hdl.handle.net/11705/JCIS/2016/015
8. Mazrekaj, A., Shabani, I.: Pricing Schemes in Cloud Computing : An Overview. International Journal of Advanced Computer Science and Applications (IJACSA) **7**(2) (2016). https://doi.org/10.14569/IJACSA.2016.070211
9. Mihoob, A., Molina-Jimenez, C., Shrivastava, S.: Consumer Side Resource Accounting in the Cloud. In: Building the e-World Ecosystem. I3E 2011 (2011). https://doi.org/10.1007/978-3-642-27260-8_5
10. Müller, C., Fernandez, P., Gutierrez, A.M., Martín-Díaz, O., Resinas, M., Ruiz-Cortés, A.: Automated Validation of Compensable SLAs. IEEE Transactions on Services Computing (2018). https://doi.org/10.1109/tsc.2018.2885766
11. Patanjali, S., Truninger, B., Harsh, P., Bohnert, T.M.: CYCLOPS: A micro service based approach for dynamic rating, charging & billing for cloud. In: 2015 13th International Conference on Telecommunications (ConTEL). pp. 1–8. IEEE (7 2015). https://doi.org/10.1109/ConTEL.2015.7231226
12. Proctor, M.: Drools: A rule engine for complex event processing. In: Proceedings of the 4th International Conference on Applications of Graph Transformations with Industrial Relevance. pp. 2–2. AGTIVE'11, Springer-Verlag, Berlin, Heidelberg (2012). https://doi.org/10.1007/978-3-642-34176-2_2
13. Rochwerger, B., Breitgand, D., Levy, E., Galis, A., Nagin, K., Llorente, I.M., Montero, R., Wolfsthal, Y., Elmroth, E., Caceres, J., Ben-Yehuda, M., Emmerich, W., Galan, F.: The Reservoir model and architecture for open federated cloud computing. IBM Journal of Research and Development **53**(4), 1–4 (4 2010). https://doi.org/10.1147/jrd.2009.5429058
14. Toosi, A.N.: On the Economics of Infrastructure as a Service Cloud Providers: Pricing, Markets, and Profit Maximization. Tech. rep., University of Melbourne (2014)