Cloudifier: an ecosystem for the migration of distributed applications to the cloud

Umberto Villano*, Stefano Cagnoni†, Maria Carla Calzarossa‡, Antonella di Stefano§ and Beniamino di Martino¶
*Dipartimento di Ingegneria
Università degli Studi del Sannio
Email: villano@unisannio.it
†Dipartimento di Ingegneria dell'Informazione
Università degli Studi di Parma
Email: stefano.cagnoni@unipr.it
‡Dipartimento di Ingegneria Industriale e Informazione
Università degli Studi di Pavia
Email: mcc@unipv.it
§Dipartimento di Ingegneria Elettrica, Elettronica e Informatica
Università degli Studi di Catania
Email: adistefa@diit.unict.it
¶Dipartimento di Ingegneria dell'Informazione
Seconda Università degli Studi di Napoli
Email: beniamino.dimartino@unina2.it

***Keywords*-Cloud Computing; Migration; Performance evaluation; Security; Semantics**

***Abstract*—The Cloudifier project deals with the design and development of an ecosystem of tools for the assisted migration to a cloud or multi-cloud environment of scientific and business-oriented distributed applications. Recent surveys show that many of the organizations that are not yet running their applications in the cloud are experimenting with infrastructure-as-a-service mode. The Cloudifier tools will provide capabilities to evaluate the characteristics of a legacy distributed application by profiling its behavior, to collect information about performance, cost and security on commercial cloud service providers, as well as to assess the quality of the interconnections between providers. On the basis of this data, and interacting with the customers that expose their requirements in an informal way, a "smart" brokering system will find an optimal service composition and evaluate its execution cost, by taking into account the cost plans of the providers, and possibly exploiting multi-cloud configurations.**

I. Introduction

Recent surveys show that the great majority of organizations that are not yet running applications in the cloud are experimenting with infrastructure-as-a-service [1]. However, many enterprises are still reluctant to the idea of placing their core, mission-critical IT systems out in the cloud. This is particularly true for deeply integrated applications, whose migration to the cloud could be rather challenging. Additional factors inducing to keep IT systems on-premise are interoperability with existing systems, security, data sovereignty and regulatory compliance [2].

This paper reports a three-years project proposal submitted to the Italian Ministry of University and Research (MIUR) in response to the PRIN 2015 call[1]. The objective of the Cloudifier project is the design and implementation of an ecosystem of tools for the assisted migration to a cloud or multi-cloud environment of distributed applications, either scientific or business-oriented. The suite of Cloudifier tools will allow the evaluation of the characteristics of a legacy distributed application by profiling its behavior. In addition, through these tools it will be possible to collect information (about performance, cost, security, etc.) on commercial cloud service providers (CSP) and assess the quality of the interconnections between providers. On the basis of this data, and of the interaction with the customers, who expose their requirements in an informal way, a "smart" brokering system will find an optimal service composition and evaluate its execution cost, by taking into account the cost plans of commercial providers, and possibly exploiting multi-cloud configurations.

The distributed application to be migrated is decomposed into a number of elementary components, and the execution model to be adopted in the cloud is chosen, by also taking into account possible additional requirements about reliability (e.g., high availability, disaster recovery) and trustworthiness. Then, an estimate of the cloud resources required for executing the application is projected on the cost models of CSP offerings, thus obtaining a forecast of the execution cost (as a function of the target application, of the number of runs required, or on a monthly/yearly basis in the case of services constantly exposed on the net). Interaction with the customer will make it possible to choose a provider

[1] http://prin.miur.it

configuration and to find out an expected cost. A provisioning and deployment system allows all the components of the cloudified application to be launched automatically, without human intervention, even in the case of multi-cloud configurations. The cloud resources actually used (and their corresponding costs) are constantly monitored during the application execution, and alerts are generated whenever a significant deviation from the predicted behavior is detected or even suspected. This may or may not entail a modification of the application execution plan.

The rest of this paper is structured as follows. In Section II, the state-of-the-art of existing tools and techniques for the migration and effective exploitation of clouds is reviewed. Then, in Section III the architecture of the Cloudfier ecosystem is explained in detail. The paper closes (Section IV) with our conclusions.

II. Related Work

Currently economic convenience factors are driving a large number of organizations to plan the deployment of their applications to the cloud in IaaS mode [3]. In this context, providing support for the migration of legacy applications to clouds and multi-clouds (multiple clouds made available by different providers) is definitely a hot topic. Even if works addressing the problem of the migration to the cloud have recently started to appear in the literature [4], [5], [6], a fully integrated solution as the one proposed in this paper is still missing.

The identification of distributed applications or workflows has been insufficiently explored in the literature, and most papers are focused on application partitioning, a similar but different problem. Existing products and tools ([7], [8], [9], [10]) have significant limitations, such as neglecting geographical distribution of tasks or network distances, or requiring the user to specify "by hand" tasks, physical machines, network status. These problems limit the effectiveness of such systems when they operate in heterogeneous and dynamic scenarios like multi-clouds. Even the papers that focus on the subsequent deployment of tasks on clouds [11] consider only single-cloud scenarios. Moreover, none of these papers deal with unpredictable and highly variable network traffic, or identify complex application patterns. Conversely, the approach adopted in this paper makes it possible to identify the replica patterns, so as to devise optimized deployments.

Dynamic resource provisioning in cloud environments has been mainly investigated in the framework of fluctuating workload patterns [12]. More specifically, predictive and reactive provisioning mechanisms that enable on-the-fly resource allocation and handle both long-term variations and short-term fluctuations in the workload have been studied [13], [14]. Predictive provisioning is used to estimate workload on a long-term scale, while reactive provisioning is used to correct errors or react to unanticipated conditions, such as flash crowds. The time-varying characteristics of the workload have been described by means of Markovian processes that can be used with analytical techniques for performance prediction [15].

Commercial cloud brokers, such as Smart Cloud Broker[2], do not consider security aspects. Ongoing research projects, such as SPECS [16], [17], take into account the fact that many providers are increasingly adopting cloud-specific security control frameworks as the CSA Cloud Control Matrix [18] and NIST SP 800-53 Rev. 4 [19]. Thus, security monitoring and probing can be used to support the brokering process. Only a few limited approaches [20], [21], [22] have been proposed to optimize the selection of service providers for multi-cloud collaborative computing, based on trust that can be automatically computed from objective measures and user feedbacks. Moreover, to the best of our knowledge, no effective solution has been defined to further improve the trust model with automatic learning of service utilization patterns.

Very few results on cloud cost forecast are currently available in the literature. The approach proposed by CloudProphet [23] is the most similar to the one we are proposing in this paper. It aims at providing forecasts of legacy applications executed on clouds. Predictions are based on application instrumentation and tracing. Also similar to our proposal are the Kingfisher [24] and the CloudGuide [25] systems, which try to reduce customer costs by applying optimization techniques. A Rightscale service, PlanForCloud[3], provides long-term cost forecasts of cloud applications, enabling the user to choose the most economic solution. However, it computes costs solely on the basis of the resource usage expected by the user. Other cost monitoring and management tools with no prediction capabilities are Cloudyn[4], CloudCruiser[5], Cloudability[6]. The use of benchmarks based on custom workloads to predict the provider behavior is proposed by CloudCMP [26].

Cloud service semantic discovery is appealing, and usually exploits WSDL annotations (OWL-S). With few exceptions, literature lacks machine-readable approaches to service description and semantic matchmaking [27], [28], [29], [30]. The mOSAIC project[7] has applied semantics to the description, discovery and composition of cloud services. In presence of multiple prices for services, smart brokers can obtain significant savings for users. While multi-cloud environments might ease the brokering of applications, considering non-functional requirements is still an open issue [31], [32], [33].

Virtualization and infrastructure fast changes become

[2] http://www.smartcloudbroker.com
[3] http://www.planforcloud.com
[4] http://www.cloudyn.com
[5] http://www.cloudcruiser.com/product/
[6] http://www.cloudability.com
[7] http://www.mosaic-cloud.eu

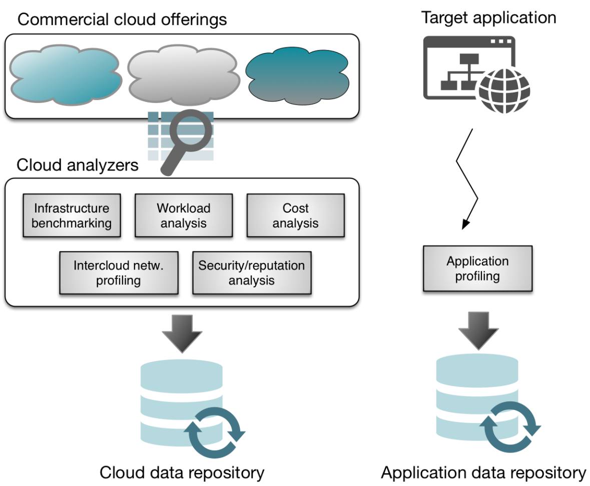

Figure 1. Step 1: Analysis

challenging issues for resource monitoring [34], [35], [36]. An OCCI extension proposes monitoring at IaaS level. High and low-level monitoring is provided by several platforms (CloudWatch, Monitis, Host sFlow, Ganglia). Big vendors promote patterns to develop orchestrated services. Recent efforts focus on QoS improvements by optimal brokering and service composition. Orchestrating middlewares [37] exist, but there are few studies on QoS of composite services [38], [39].

III. Architecture of the Cloudifier Ecosystem

Even if the Cloudifier suite of tools will be closely integrated, for the sake of clarity it is useful to decompose the migration process of the target distributed application into three main steps, as follows:

1) Analysis;
2) Interaction with customer;
3) Brokering, deployment and execution.

In the Analysis step (Fig. 1), the commercial cloud offerings and the application to be migrated are analyzed. In particular, the commercial cloud offerings are evaluated in terms of the cost, performance, workload and (perceived) security of the leased infrastructures. Moreover, the intercloud connections (i.e., the network connections between infrastructures leased from distinct CSPs) are benchmarked, in order to assess the usefulness of a multi-cloud configuration. On the other hand, the target application is profiled in order to capture and understand its behavior and resource usage. At the end of the analysis step, the collected information about CSP offerings and application profiling is stored in a cloud data repository and in an application data repository, respectively.

The Analysis step is performed by the joint operation of a set of tools that are in charge of:

- Infrastructure benchmarking. Benchmarks will be executed to measure the performance parameters that make it possible to "tailor" a simulation model of the cloud to the set of actually available resources. The results will feed a simulation engine of the leased infrastructure, whose objective is to predict performance and resource usage during the application execution.
- Workload analysis. The application of statistical techniques will allow the analysis of the dynamic behavior of the workload executed on the clouds and the construction of the corresponding models, as well as the investigation of the presence of trends and cyclic behavior, related for example to temporal usage patterns. The predictive models of the workload will identify and highlight increased usage of some specific resources, thus making it possible to implement corrective actions on their provisioning.
- Cost analysis. A tool will examine the offers of commercial cloud providers, extracting information on available resources and pricing plans. The cost models will be continuously updated.
- Intercloud network profiling. A network profiler tool will monitor the underlying intercloud networks to estimate the best deployment schema among the available multi-cloud resources, under different costs and network performance constraints. It will also take into account the degree of isolation, a very important aspect to be considered when deciding about the deployment of network-dependent applications into multi-cloud environments.
- Security/reputation analysis. This component will extract information about security and reliability of commercial providers, by also considering customer feedbacks. It will adopt an approach based on the notion of "security statements" included in cloud SLAs. As for reliability, the component will collect and analyze – using advanced techniques such as machine learning algorithms – statistics about availability and outages of commercial cloud providers. Customer feedbacks will be also collected, in order to produce an almost complete profile of cloud providers.
- Application profiling. A profiler will provide information about the distributed applications running on a set of machines, their relationships and their performance under real production workloads. The profiler will operate at two different levels. By operating locally, it will analyze and monitor all the processes executed in the specific machine (either virtual or not) to identify all the information needed for their correct execution, in terms of both software libraries and hardware resources. On the contrary, by operating at cluster level, the profiler will combine the local information collected on each machine with global information provided by a network monitor to find relevant interactions among

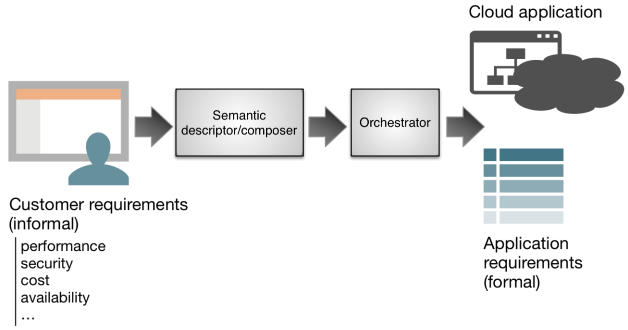

Figure 2. Step 2: Interaction with customer

the processes running on different servers. The profiler will operate either in autonomous or supervised mode. In both cases, the identification of the distributed applications will be performed by means of stochastic-based machine learning algorithms.

The second step of the migration process deals with the Interaction with customer (Fig. 2). In what follows, by "customer" we mean the cloud user of the CSP(s), who in fact plays the role of the manager/developer of the distributed application to be cloudified. In this step, customers provide the requirements in an informal way (by graphical composition, or even by application description documents). The requirements being collected refer to performance, security, cost and availability. They are processed by:

- Semantic descriptor/composer. A semantic-based description of the application will be obtained by means of customer interaction, helping the selection and management of services and patterns from multiple providers. The corresponding tool will produce, by graphical composition, a semantic-based Application Descriptor, from which a (set of) resources and service configurations and associated SLA parameters will be inferred. The tool will also consider the (semi-)automatic derivation of the semantic descriptor from available structured or semi-structured application descriptions (e.g., UML diagrams, specification requirements, design documents).
- Orchestrator. The semantic-based description will drive orchestration, which will exploit formal methods and matching algorithms in order to analyze properties and QoS of the orchestrated services. This will enable the verification of user requirements for composite services and their summaries in terms of SLAs. The Orchestrator will also analyze workflow patterns in order to evaluate the QoS of composite services. In addition, it will manage cloud services composition at run-time.

The outputs of this step are the cloudified application and the requirements for its execution, expressed in a formal way by an SLA.

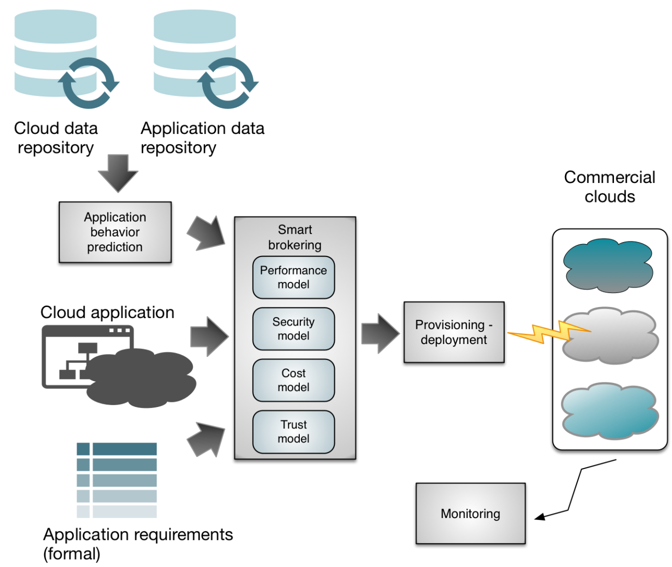

Figure 3. Step 3: Brokering, deployment and execution

In the third step of the migration process, that is, Brokering, deployment and execution (Fig. 3), the outputs of the previous steps (data repositories, cloud application, formal application execution requirements) are processed by an additional set of tools that are in charge of:

- Application behavior prediction. Simulation-based performance prediction of cloud applications will be used to assess the effect on performance of alternative deployments. The entire cloud ecosystem, that includes application components, OS, hypervisor and actual hardware, will be simulated at a balanced degree of detail, with a reasonable trade-off between accuracy and prediction speed. Predictive and pro-active techniques for just-in-time resource provisioning will take into account uncertainties of workload intensity and demands, and reactive techniques will cope with potential errors in resource estimation due to unforeseen effects.
- Smart brokering. The smart broker will exploit optimization techniques to select an optimal configuration of providers among the supported ones. The component-based architecture of the broker will allow the integration of multiple plugins to handle different types of optimization criteria (performance, security, cost, trust, etc). The plugins will be fed with the output of the previous phases, along with the predicted application behavior.
- Provisioning-deployment. The provisioner component will allow the definition of an optimized deployment plan in terms of IaaS cloud services and security mechanisms to be installed. Semantic description and workflow information will help in the definition of an internal representation of the SLA template for

matching service levels, terms of services and service bindings against requirements. Automatic deployment of the cloud application will be performed, in that cloud resources are leased from the selected providers and the application is deployed with no human intervention.
- Monitoring. After the selection of provider and resources, a customized monitoring profile is built. The feedback obtained from the real cloud will be useful for assessing the solutions, and for discovering possible SLA violations, generating alerts for the customer in the case of substantial deviations from the predicted behavior (e.g., additional resource usage, performance bugs, security breaches).

IV. Conclusions

This paper has presented the architecture of Cloudifier, an ecosystem of tools for assisting the migration of distributed applications to the cloud.

The technologies proposed by the Cloudifier project will have a significant impact in accelerating the adoption of cloud computing both in public and private sectors. In particular, the proposed solutions will allow cloud users to improve efficiency and lower costs by identifying the most cost-effective infrastructure for deploying a given workload. More specifically, any actor (be it SME, public operator or individual) interested in running an application in the cloud, will be able to make educated choices and select the most appropriate infrastructure for the applications at hand.

Cloudifier is perfectly in line with the objectives of the Digital Agenda for Europe. It will help to bridge the gap existing between cloud research and market and enable European citizens and businesses to get the most out of cloud technologies and contribute to overcome the well-known "vendor lock-in" problem, typical of cloud environments.

Ackowledgments

The authors wish to thank the Cloudifier team for the constructive support in the preparation of the project proposal.

References

[1] RightScale, Inc., "State of the cloud report," 2015. [Online]. Available: http://assets.rightscale.com/uploads/pdfs/RightScale-2015-State-of-the-Cloud-Report.pdf

[2] "The cloud migration survey: What will move, what will stay, and why?" September 2014. [Online]. Available: http://www.businesscloudnews.com/2014/09/26/the-cloud-migration-survey-what-will-move-what-will-stay-and-why/

[3] Verizon Harvard Business Review, "Business agility in the cloud," June 2014. [Online]. Available: https://hbr.org/resources/pdfs/tools/Verizon_Report_June2014.pdf

[4] P. Jamshidi, A. Ahmad, and C. Pahl, "Cloud migration research: A systematic review," *IEEE Transactions on Cloud Computing*, vol. 1, no. 2, pp. 142–157, 2013.

[5] J. Jermyn, J. Hwang, K. Bai, M. Vukovic, N. Anerousis, and S. Stolfo, "Improving readiness for enterprise migration to the cloud," in *Proceedings of the Middleware Industry Track*, ser. Industry papers. ACM, 2014, pp. 5:1–5:7.

[6] S. Islam, E. R. Weippl, and K. Krombholz, "A decision framework model for migration into cloud: Business, application, security and privacy perspectives," in *Proceedings of the 16th International Conference on Information Integration and Web-based Applications & Services*, ser. iiWAS '14. ACM, 2014, pp. 185–189.

[7] W. Chen and E. Deelman, "Partitioning and scheduling workflows across multiple sites with storage constraints," in *Parallel Processing and Applied Mathematics*. Springer, 2012, pp. 11–20.

[8] W. Jaradat, A. Dearle, and A. Barker, "Workflow partitioning and deployment on the cloud using orchestra," in *Proceedings of the 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing*. IEEE Computer Society, 2014, pp. 251–260.

[9] D.-K. Kang, S.-H. Kim, C.-H. Youn, and M. Chen, "Cost adaptive workflow scheduling in cloud computing," in *Proceedings of the 8th International Conference on Ubiquitous Information Management and Communication*, ser. ICUIMC '14. ACM, 2014, pp. 65:1–65:8.

[10] F. C. Chua and B. A. Huberman, "A bayesian approach to the partitioning of workflows," *arXiv preprint arXiv:1511.00613*, 2015.

[11] W. Jaradat, A. Dearle, and A. Barker, "Towards an autonomous decentralized orchestration system," *Concurrency Computation*, 2015, article in Press.

[12] M. Calzarossa, M. Della Vedova, L. Massari, D. Petcu, M. Tabash, and D. Tessera, "Workloads in the Clouds," in *Principles of Performance and Reliability Modeling and Evaluation*, ser. Springer Series in Reliability Engineering, L. Fiondella and A. Puliafito, Eds. Springer, 2016.

[13] D. Huang, B. He, and C. Miao, "A Survey of Management in Multi-Tier Web Applications," *IEEE Communications Surveys Tutorials*, vol. 16, no. 3, pp. 1574–1590, 2014.

[14] Jennings, B. and Stadler, R., "Resource Management in Clouds: Survey and Research Challenges," *Journal of Network and Systems Management*, vol. 23, no. 3, pp. 567–619, 2015.

[15] S. Pacheco-Sanchez, G. Casale, B. Scotney, S. McClean, G. Parr, and S. Dawson, "Markovian Workload Characterization for QoS Prediction in the Cloud," in *Proceedings of the 4th International Conference on Cloud Computing*, ser. CLOUD'11. IEEE, 2011, pp. 147–154.

[16] M. Rak, N. Suri, J. Luna, D. Petcu, V. Casola, and U. Villano, "Security as a service using an SLA-based approach via SPECS," in *Proceedings of 5th IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*, vol. 2, 2013, pp. 1–6.

[17] J. Luna, A. Taha, R. Trapero, and N. Suri, "Quantitative reasoning about cloud security using service level agreements," *IEEE Transactions on Cloud Computing*, 2015.

[18] "Cloud Security Alliance's Cloud Control Matrix (CSA CCM)," https://cloudsecurityalliance.org/download/cloud-controls-matrix-v3-0-1/, accessed: 2015-12-22.

[19] "National Institute of Standards and Technology (NIST) Special Publication (SP) 800-53 Revision 4 (R4)," http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-53r4.pdf, accessed: 2015-12-22.

[20] T. H. Noor, Q. Z. Sheng, S. Zeadally, and J. Yu, "Trust management of services in cloud environments: Obstacles and solutions," *ACM Comput. Surv.*, vol. 46, no. 1, pp. 12:1–12:30, 2013.

[21] S. Habib, V. Varadharajan, and M. Muhlhauser, "A trust-aware framework for evaluating security controls of service providers in cloud marketplaces," in *Proceedings of the 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, 2013, pp. 459–468.

[22] X. Li, H. Ma, F. Zhou, and W. Yao, "T-broker: A trust-aware service brokering scheme for multiple cloud collaborative services," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 7, pp. 1402–1415, 2015.

[23] A. Li, X. Zong, S. Kandula, X. Yang, and M. Zhang, "Cloudprophet: towards application performance prediction in cloud," *SIGCOMM-Computer Communication Review*, vol. 41, no. 4, p. 426, 2011.

[24] U. Sharma, P. Shenoy, S. Sahu, and A. Shaikh, "A Cost-Aware Elasticity Provisioning System for the Cloud," in *Proceedings of the 31st International Conference on Distributed Computing Systems (ICDCS)*, 2011, pp. 559–570.

[25] S. H. Liew and Y.-Y. Su, "CloudGuide: Helping users estimate cloud deployment cost and performance for legacy web applications," in *Proceedings of the 4th IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*, 2012, pp. 90–98.

[26] A. Li, X. Yang, S. Kandula, and M. Zhang, "Cloudcmp: comparing public cloud providers," in *Proceedings of the 10th International Conference on Internet Measurement*. ACM, 2010, pp. 1–14.

[27] R. Aversa, L. Tasquier, and S. Venticinque, "Management of cloud infrastructures through agents," in *Proceedings of the 3rd International Conference on Emerging Intelligent Data and Web Technologies*, September 19 - 21 2012, pp. 46–53.

[28] S. Venticinque, V. Negru, V. Munteanu, C. Sandru, R. Aversa, and M. Rak, "Negotiation policies for provisioning of cloud resources," in *Proceedings of the 4th International Conference on Agents and Artificial Intelligence (ICAART 2012)*, vol. 2, 2012, pp. 347–350.

[29] T. Binz, U. Breitenbücher, O. Kopp, and F. Leymann, "Tosca: Portable automated deployment and management of cloud applications," in *Advanced Web Services*. Springer, 2014, pp. 527–549.

[30] G. Cretella and B. Di Martino, "A semantic engine for porting applications to the cloud and among clouds," *Software: Practice and Experience*, vol. 45, no. 12, pp. 1619–1637, 2015.

[31] W. Wang, D. Niu, B. Li, and B. Liang, "Dynamic cloud resource reservation via cloud brokerage," in *Proceedings of the 33rd IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2013, pp. 400–409.

[32] N. Grozev and R. Buyya, "Inter-cloud architectures and application brokering: taxonomy and survey," *Software: Practice and Experience*, vol. 44, no. 3, pp. 369–390, 2014.

[33] R. Aversa, B. Di Martino, M. Rak, and S. Venticinque, "Cloud agency: A mobile agent based cloud system," in *Proceedings of the 4th International Conference on Complex, Intelligent and Software Intensive Systems (CISIS 2010)*, 2010, pp. 132–137.

[34] G. Katsaros, R. Kušbert, and G. Gallizo, "Building a service-oriented monitoring framework with REST and Nagios," in *Proceedings of the IEEE International Conference on Services Computing (SCC)*, 2011, pp. 426–431.

[35] S. Ferretti, V. Ghini, F. Panzieri, M. Pellegrini, and E. Turrini, "QoS-aware clouds," in *Proceedings of the 3rd IEEE International Conference on Cloud Computing (CLOUD)*, July 2010, pp. 321–328.

[36] J. Shao, H. Wei, Q. Wang, and H. Mei, "A runtime model based monitoring approach for cloud," in *Proceedings of the 3rd IEEE International Conference on Cloud Computing (CLOUD)*, 2010, pp. 313–320.

[37] "US government cloud computing technology roadmap release 1.0 (draft)," in *Special Publication 500-293*. NIST, 2011, vol. 2, pp. 1–85.

[38] A. Jula, E. Sundararajan, and Z. Othman, "Cloud computing service composition: A systematic literature review," *Expert Systems with Applications*, vol. 41, no. 8, pp. 3809–3824, 2014.

[39] Y. Liu, M. Li, and Q. Wang, "A novel user-preference-driven service selection strategy in cloud computing." *International Journal of Advancements in Computing Technology*, vol. 4, no. 21, 2012.