

PAPER • OPEN ACCESS

Web services selection a perspective of computational physics

To cite this article: M A Adarme *et al* 2020 *J. Phys.: Conf. Ser.* **1587** 012017

View the [article online](#) for updates and enhancements.

You may also like

- [Service Replacement Based on Service Cluster Based on Information Technology](#)
Xueqing Qin
- [Web Service Performance Analysis and Research of Distributed LAMP Architecture](#)
Yuan Yang
- [A review of attacks, objects, and mitigations on web services](#)
Rafif Masrur Rauf, Tubagus Eiffel Rivaldo and Amiruddin Amiruddin



ECS
The
Electrochemical
Society
Advancing solid state &
electrochemical science & technology

DISCOVER
how sustainability
intersects with
electrochemistry & solid
state science research

Web services selection a perspective of computational physics

M A Adarme¹, M A Jimeno², and E G Puerto¹

¹ Universidad Francisco de Paula Santander, San José de Cúcuta, Colombia

² Universidad del Norte, Barranquilla, Colombia

E-mail: madarme@ufps.edu.co, majimeno@uninorte.edu.co

Abstract. Concerning computational physics, web services are conceived as mathematical units that are experienced in different systems that offer service composition. Due to the exponential growth of web services and their deployment on cloud platforms, quality of service parameters have now become an essential factor when searching for and selecting services that must satisfy specific non-functional requirements of a user application. A variety of service components are highly configurable and are dynamic scenarios because a significant number of services can meet these requirements. This work analyzes the systemic perspective of approaches for the selecting and searching of web services that have specifications of optimization strategies based on the configurable quality of service parameters with test scenarios in cloud environments that have a considerable number of services as input. The study shows that policies based on artificial intelligence and related areas are the ones with the most significant convergence, and the approaches analyzed to give a perspective of future work aimed at strategies based on automatic learning.

1. Introduction

Service composition requires high levels of service combinations that satisfy a specific functional requirement. Every scenario generated through arrangement selects web services with an optimal service quality parameter; hence, computational physics offers an experimental scene leading to the characterization of the performance of a web service as a mathematical unit converted into a group of code, intending to provide scenarios that enable an ideal composition of the specific requirement. Service-oriented architecture (SOA) styles have become the most widely used paradigm in the development of applications that require high levels of service reuse [1] and use as fundamental elements for the construction of distributed applications the Web services (WS). WS is a way of providing communication between components and sophisticated software applications offering independence and transparency between computational platforms [2], and this advantage has taken to the development of distributed applications with multilanguage support. However, current information system requirements make it necessary to develop complex business systems that operate with a considerable number of WS, which are combined under an operation called composition.

Web services composition (WSC) refers to the combination of WS describing the internal logic of how they will interact; they converge in an application that deploys them in a logical and synchronized way to satisfy a particular functional requirement [3,4]. The composition is basically modeled in two phases in a typical environment: the first defines (search and select) the WS involved that depend on restrictions given by functional and non-functional requirements, whose result is placed in an abstract representation model (graph, network, among others) or in workflows where the order, conditions and restrictions of execution of the WS are specified. Second phase consists of the performance of the WSC



model that is delegated to a centralized or distributed execution engine depending on the deployment architecture [5].

WS providers use cloud resources for their deployment, so the interaction and consumption of resources are key analysis factors when carrying out a WSC, so, for example, many WS may be exposed to different levels of quality of service (QoS) on their service level agreement (SLAs) in a set of clouds. This factor gives the WSC its importance for resource optimization models. The aim of this paper is the analysis of search and selection mechanisms on cloud environments that offer a description of their heuristics of management for the search and selection of WS that takes into account parameters of quality of service.

The paper shows the main computational approaches of different middlewares and cloud platforms with strategies for the selection and search of WS. It also establishes a contextual background for the composition of web services in their dynamic behavior, and this framework is the basis for the analysis of the technical aspects of each approach. The study is based on four relevant issues for analysis:

1. Research references based on academic experiences with benchmark systems built under dynamic behaviors.
2. Description of test scenario allowing to specify the test models from the computational physics approach.
3. Restrictive search filter analysis by the criteria for middleware and framework, or any research proposal concerning algorithms with QoS conditions for selecting services.
4. Research describing the analysis, design and construction of the middleware solution or framework. The aim is to have a clear perspective of how the activities within the WS composition process were approached by the authors and to be able to identify characteristics of the computational strategies used and on which area they were based.

2. Overview of the selecting and searching problem of web services

The use of SOA with cloud computing allows the construction of distributed applications that can communicate and interact (invoke) services from their surroundings or calls that belong to the same context (domain) where the application is being executed. As a result of the dynamic environment of cloud resources, the tasks of composing services are not carried out statically as is usually the case in traditional SOA, since in the cloud the composition depends on parameters for the use of computational resources that, although they are elastic by nature, there are utilization considerations which involve hardware as a service (I/O, memory and CPU) [6], in this case, some services will consume more resources than others, causing an inconvenience of monetary costs (monetary increase) and hardware disposition (platforms that limit the use of their hardware and only with authorizations increase their resource) [7].

Conceptually, information technology (IT) users acquire infrastructures through the Cloud, and their applications are executed using and consuming the resources they offer. These resources are called services, and the Cloud model identifies three categories: infrastructure-as-a-service (IaaS), platform-as-a-service (PaaS), and software-as-a-service (SaaS) [8].

Although the growth of clients has been exponential in this model, the problem lies in the amount of WS that can be composed to satisfy a particular requirement [9,10], it is difficult for users to analyze the options that suit their needs, even more at the time of implementing a software under this model, tasks such as search, mediation and tracking a WS become very important [11]. Currently, the number of SWs is growing rapidly, so the search and selection process is no trivial task, especially if there are non-functional requirements that must be taken into account and that must support cloud computing environments with QoS parameters that ensure compliance with the requirements of applicants and providers, as well as the ability to establish agreements for the provision and acquisition of computational resources (static or dynamic negotiations) if the composition so requires [12,13]. The selection and SW search problems have been widely studied in different research [2,10-16], these can be grouped into:

- High algorithmic complexity in search processes due to the considerable number of WS with similar functionalities.
- WS selection heuristics to be inferred the functionality of service when ambiguity exists in the description and handling of incomplete information in the web context.
- Various forms of integration solution that allow the same result for different models of WSC.

3. Middleware and framework

Solution approaches are mostly based on middlewares and frameworks (FW), within which techniques and associated methods are specified in areas of emerging computing knowledge (ubiquitous computing, autonomic computing, reflective systems, context conscious systems, among others) [15]. Middlewares have been responsible for providing interoperability between services of heterogeneous platforms through the control of message passage, protocol management, security, and other aspects of non-functional requirements of the services; its main advantage lies in that they operate at the level of interoperability of existing systems [16]. FW integrates a platform based on components to standardize and facilitate system development operations. In general, middleware allows for coupling to already created systems, and FW offers a conceptual and technological architecture on which the solution is going to be implemented.

In [17], and the indexing technique is developed for the management of information on Cloud providers and their services. It implements algorithms for the searching and selection of services in a controlled environment; its solution strategy is based on the classification of providers by QoS parameters and categorization of services by their functionality. In its development, in general, it proposes the use of a dynamic information structure that has the advantage of modeling suppliers' data, as well as their non-functional requirements and coding them in such a way that the selection algorithm quickly finds similarities between services and suppliers, offering the advantage of performing the classification optimally. SMICloud [18] is a framework for the automatic searching and selecting of cloud providers based on preset user requirements. This framework dynamically implements negotiations between SLAs and develops a technique for classifying services using metrics for each QoS parameter, resulting in the categorization of services by each provider through a standard evaluation mechanism of cloud services for each type of QoS.

Gomes, *et al.* [19] introduce middleware for managing cloud resources based on non-functional requirements (NFR) of services pre-established by users. It is based on model-driven techniques that allow the non-functional requirements of service to be specified and selected according to its RNF, based on the choice of multiple clouds. Its result is a selection alternative under the premise of reduced consumption costs. In [20], a model is made for WSC on SaaS that takes into account QoS requirements and non-functional requirements presented by cloud providers. It uses genetic algorithms to perform WSC with an SLA negotiation module. The composition is deployed under a SaaS environment, in general, the user interacts directly with a SaaS interface that serves as an intermediary between the cloud providers and the exposed services, this solution strategy allows to centralize the requests of users, process them taking into account QoS attributes and look for correspondence between which service and provider is the most appropriate to meet the requirement given by the user.

Wakrime, *et al.* [21] offers an overview of WSC using heuristics based on "minimal unsatisfiability" that are used for the treatment of domains where NP type problems predominate. This system uses SAT encoding and the creation of minimal unsatisfiability sub-formulas to obtain the WS involved in the composition of a SaaS architecture. The use of dynamic QoS attributes is highlighted from this work, *i.e.*, the problem is continuously coded if there are changes in the QoS requirements, the system finds the minimum composition as an unsatisfactory variant of a formula obtained for the modeled composition problem. The advantage of this approach is to be able to minimize the overall cost of using the WS needed in the composition activity, as well as control general attributes (for the case of the WSC) and atomic for each WS.

A context conscious QoS and SLA work for WSC in cloud environments are developed in [10]. WSC is modeled as an optimization problem. Each WS is evaluated and selected, taking into account QoS constraints contained in the SLAs; an algorithm is used that decides the best WS that does not violate the limitations preset by the non-functional requirements of the WSC. The contribution of this research is a heuristic based on genetic algorithms that minimize the computational cost of searching and selecting a WS, as well as offering an evaluation method that detects within the WSC those WS that could eventually violate a condition of the SLAs. This research also highlights the problem of the similarity of WS functionalities together with different QoS attributes on a growing number of clouds that leads to a question of high computational complexity.

Yu, *et al.* [22] have developed a form of WSC using automatic learning methods, and his research highlights the advantage of using composition strategies based on a knowledge base that is fed every time a WSC is performed, in such a way that WS can be reused without going through evaluation stages again. The composition is addressed through an uncertainty planning problem, and execution flow patterns are determined. They feed the Q-learning-based system, which is a machine learning class by reinforcement.

In [23] it is created a system of dynamic selection of services in time of execution based on criteria of quality, the compositions is modeled as a problem of optimization and solved through entire programming, its system finds the optimal solution that satisfies these criteria for service and then the model is placed to external level where the behavior of the services is taken in the activity of composition. E3 [24] is also based on this work, where a framework is developed to optimize the search for services that meet a quality criterion, a problem considered NP when there are multiple services that satisfy a non-functional requirement, uses multi-objective genetic algorithms that considerably reduce the search time of the service, the results are placed in a set of Pareto that store optimal solutions that satisfy multiple quality conditions.

In the field of the semantic web is the work done by Zhou, *et al.* [25], which implements a system for cloud platforms under a Bayesian algorithm solution strategy. The work establishes ontologies to describe WS and providers together with a scheme of relationships that leads to enriching a semantic environment that operates on inferential processes in decision-making techniques based on Bayesian methods. With the modeling of the WSC metrics are established to identify the similarities between WS and having the system based on ontologies, the semantic composition of WS is easily implemented.

Table 1 shows the categorization of the projects studied by solution strategies; it can be seen that the techniques in the area of artificial intelligence are the greatest rise, as well as mechanisms based on machine learning and implementation of optimization algorithms based on genetic algorithms, linear and multimodal programming.

Table 1. Categorizing projects by solution strategy.

Researcher	PA ¹	ML ²	MD ³	GA ⁴	OA ⁵
Wang D, <i>et al.</i> [10]				X	
Sundareswaran S, <i>et al.</i> [17]	X				
Garg S, <i>et al.</i> [18]		X			
Gomes R C, <i>et al.</i> [19]		X	X		
Bentaleb A, <i>et al.</i> [20]				X	
Wakrime A, <i>et al.</i> [21]					X
Yu L, <i>et al.</i> [22]		X			
Alrifai M, <i>et al.</i> [22]					X
Wada H, <i>et al.</i> [24]				X	
Zhou X, <i>et al.</i> [25]		X		X	

¹ Planning algorithm; ² Machine learning; ³ Model driven;

⁴ Genetic algorithms; ⁵ Other optimization algorithms

WSC's research area and specifically WS selection and search processes have grown considerably, the literature is extensive and presents varied approaches, this work analyzed those investigations that offered a broad description at the architectural level of middlewares and frameworks that focused their activity in the selection of WS with QoS. In this analysis, the proposals interact with composition engines, and a small number create their own deployment platforms for their tests. Likewise, the analyzed works demonstrate that the mechanisms are implemented for heterogeneous information systems where diversity of systems, applications, technologies, and architectures can be used.

Also, the projects analyzed are independent at the time of data entry to their systems, some based their listings on a dataset and others manually record each SW with their suppliers and QoS statically, the reason that facilitates the entry of requirements to the system, allowing middlewares such as FW to obtain an entry that promotes the discovery of services and analysis of WSC models. On the other hand, they document the WS registration mechanisms, specifying their semantics and the way their search engines will operate it. Dynamic searches are usually performed at design time, some approaches to WS incorporation at run time, but limited to discovering a set of WS that are available. There are similarities between the approaches analysed in terms of the problems encountered in implementing its selection strategies, and these include:

1. Incorrect semantics in the presentation of the WS functionality, at the level of the identification of the input and return parameters, generating problems in the search for services in the diverse repositories.
2. Searching of service is done in runtime, it leads to problems of type NP when there are services with which they offer the same functionality.
3. Handling of static information by WS providers. Providers sometimes give descriptions that do not correspond to the service, thus preventing the successful invocation of a WS because their parameters do not match.
4. Web service standards based on XML, such as SOAP and WSDL, present a lack of explicit semantics so that two identical XML descriptions can mean totally different things, sometimes depending on the context in which they are used.

4. Conclusion

Emerging computing areas have been identified on which the middlewares and framework projects were based. It is emphasized that a significant number of approaches work under the discipline of artificial intelligence, and its direct applicability is seen in applications based on automatic learning. Similarly, the strategies implement their system, taking into account QoS constraints that offer a whole panorama in the field of compositions optimization. The analyzed projects present exhaustive documentation of some characteristics of the system. However, it could be evidenced by the lack of documentation of some in the testing stage. Even though there are many proposals and system approaches that offer selection and searches of WS, this work analyzed those investigations that offered a wide description at the architectural level of their solution strategies, it was detected that proposals interact with composition engines and a reduced number create their deployment platforms. No matter the case, its objective is to offer mechanisms so that heterogeneous information systems can use a diversity of systems, applications, technologies, and architectures, an intrinsic advantage of systems based on the cloud.

References

- [1] Rosen M, Lublinsky B, Smith K T, Balcer M J 2012 *Applied SOA: Service-Oriented Architecture and Design Strategies* (California: John Wiley & Sons)
- [2] Manouvrier M G V and Murat C 2014 Web services composition: Complexity and models *Discret. Appl. Math.* **196** 1
- [3] Erl T Puttini R and Mahmood Z 2013 *Cloud Computing: Concepts, Technology and Design* (USA: Prentice Hall PTR)

- [4] Khadka R and Sapkota B 2010 An evaluation of dynamic web service composition approaches *4th International Workshop on Architectures, Concepts and Technologies for Service Oriented Computing (ACT4SOC)* (Athens: University of Piraeus)
- [5] Agarwal V, Chafle G, Dasgupta K, Karnik N 2005 Synthy: A system for end to end composition of web services *Web Semant. Sci. Serv. Agents World Wide Web* **3** 311
- [6] Jula A, Sundararajan E, Othman Z 2014 Cloud computing service composition: A systematic literature review *Expert Systems with Applications* **41(8)** 3809
- [7] Goettelmann E, Fdhila W, Godart C 2013 Partitioning and cloud deployment of composite web services under security constraints in Cloud Engineering *IEEE International Conference on Cloud Engineering (IC2E)* (Redwood: IEEE)
- [8] Kavis M J 2014 *Architecting the Cloud: Design Decisions for Cloud Computing Service Models (SaaS, PaaS, AND IaaS)* (USA: John Wiley & Sons)
- [9] Guzmán Luna J A, Ovalle Carranza D A 2008 Composición de servicios: Una aplicación de la web semántica y las técnicas de planificación automática *Ingeniería e Investigación* **28(3)** 145
- [10] Wang D, Ding H, Yang Y, Mi Z, Liu L, Xiong Z 2016 QoS and SLA aware Web service composition in cloud environment *KSII Transactions on Internet & Information Systems* **10** 12
- [11] Sheng Q Z, Qiao X, Vasilakos A V, Szabo C, Bourne S, Xu X 2014 Web services composition: A decade's overview *Information Science* **280** 218
- [12] Zheng Z, Zhang Y, Lyu M R 2014 Investigating QoS of real-world web services *IEEE Trans. Serv. Comput.* **7(1)** 32
- [13] Wu C-S, Khoury I 2012 Tree-based search algorithm for web service composition in SaaS *Ninth International Conference on Information Technology: New Generations (ITNG)* (Las Vegas: IEEE)
- [14] Zou G, Gan Y, Chen Y, Zhang B 2014 Dynamic composition of web services using efficient planners in large-scale service repository *Knowledge-Based Syst.* **62** 98
- [15] Lemos A L, Daniel F, Benatallah B 2016 Web service composition: A survey of techniques and tools *Web Serv. Compos. a Surv. Tech. Tools* **48(3)** 33
- [16] Pérez H and Gutiérrez J 2014 A survey on standards for real-time distribution middleware *ACM Comput. Surv.* **46(4)** 1
- [17] Sundareshwaran S, Squicciarini A, Lin D 2012 A brokerage-based approach for cloud service selection *IEEE 5th International Conference on Cloud Computing (CLOUD)* (Honolulu: IEEE)
- [18] Garg S K, Versteeg S, Buyya R 2011 Smicloud: A framework for comparing and ranking cloud services *Fourth IEEE International Conference on Utility and Cloud Computing (UCC)* (Australia: IEEE)
- [19] Gomes R, Costa F, Da Rocha R, Georgantas N 2014 A middleware-based approach for QoS-aware deployment of service choreography in the cloud *Proceedings of the 11th Middleware Doctoral Symposium* (Delft: Association for Computing Machinery)
- [20] Bentaleb A, Ettalbi A 2016 Toward cloud SaaS for web service composition optimization based on genetic algorithm *2nd International Conference on Cloud Computing Technologies and Applications (CloudTech)* (Marrakech: IEEE)
- [21] Wakrime A A, Jabbour S 2015 Minimum Unsatisfiability based QoS Web service composition over the cloud computing *15th International Conference on Intelligent Systems Design and Applications (ISDA)* (Marrakech: IEEE)
- [22] Yu L, Zhili W, Lingli M, Jiang W, Meng L, Xue-song Q 2013 Adaptive web services composition using q-learning in cloud *IEEE Ninth World Congress on Services (SERVICES)* (Santa Clara: IEEE)
- [23] Alrifai M, Risse T, Nejdl W 2012 A hybrid approach for efficient Web service composition with end-to-end QoS constraints *ACM Trans. Web* **6(2)** 7
- [24] Wada H, Suzuki J, Yamano Y, Oba K 2012 A multiobjective optimization framework for SLA-aware service composition *Serv. Comput. IEEE Trans.* **5(3)** 358
- [25] Zhou X, Mao F 2012 A semantics web service composition approach based on cloud computing *Fourth International Conference on Computational and Information Sciences (ICCIS)* (China: IEEE)