

SeaClouds: Agile management of complex applications across multiple heterogeneous clouds

D. Athanasopoulos⁽¹⁾, M. Barrientos⁽²⁾, L. Bartoloni⁽³⁾, A. Brogi⁽³⁾,
M. Buccarella⁽³⁾, J. Carrasco⁽²⁾, J. Cubo⁽²⁾, F. D'Andria⁽⁴⁾,
E. Di Nitto⁽¹⁾, A. Nieto⁽²⁾, M. Oriol⁽³⁾, E. Pimentel⁽²⁾, S. Zenzaro⁽³⁾

⁽¹⁾ *Dipartimento di Elettronica, Informazione e Bioingegneria,
Politecnico di Milano, Italy*

⁽²⁾ *Departamento de Lenguajes y Ciencias de la Computación,
Universidad de Málaga, Spain*

⁽³⁾ *Department of Computer Science, University of Pisa, Italy*

⁽⁴⁾ *ATOS, Spain*

Abstract. SeaClouds is a European FP7 research project, whose goal is to develop a novel open solution to provide developers with the capabilities to design, configure, deploy, and manage complex applications across multiple heterogeneous clouds in an efficient and adaptive way.

1 General data of the project

Name	Seamless adaptive multi-cloud management of service-based applications
Acronym	SeaClouds
Project Id	EU-FP7-ICT-610531
Funding source	European Union
Duration	From October 1, 2013 to March 31, 2016
Total funding	Budget: 2.99 MEuros - EU contribution: 2.19 MEuros
Consortium	ATOS (Spain), University of Malaga (Spain), University of Pisa (Italy), Politecnico di Milano (Italy), Cloudsoft (UK), Nurogames (Germany)
Web site	http://www.seaclouds-project.eu/

2 Brief description of the project

How to deploy and manage, in an efficient and adaptive way, complex applications across multiple heterogeneous cloud platforms is one of the problems that have emerged with the cloud revolution. SeaClouds is a European FP7 research project which aims at enabling a seamless adaptive multi-cloud management of complex applications by supporting the distribution, monitoring and migration of application modules over multiple heterogeneous cloud platforms. In short:

- SeaClouds is a software platform which supports the “design, development, planning and management” of complex business applications distributed on multi-cloud environments.

- SeaClouds provides an enterprise capability with continuous software delivery that enables independent application vendors to mitigate risks and reduce time and cost to market.
- SeaClouds orchestrates services, platforms and infrastructures to ensure that they dynamically meet the needs of cloud applications.
- SeaClouds provides an integrated standards-based multi-cloud application management system that follows the DevOps approach with various basic capabilities delivered to the developer via an innovative Graphical User Interface (GUI).

3 Objectives of the project

The SeaClouds project aims to develop a new open source platform featuring a seamless adaptive multi-cloud management of service-based applications. The platform consists of an application management system over IaaS and PaaS clouds, and it implements a DevOps approach for continuous software delivery.

SeaClouds permits developers to design, deploy, manage and configure complex applications across multiple and heterogeneous clouds, something unfeasible hitherto.

The specific objectives of SeaClouds are:

- ***Orchestration and adaptation of services distributed over different cloud providers.*** SeaClouds aims at providing the assisted design, synthesis, and simulation of service orchestrations on different cloud providers, by distributing modules of cloud-based applications over multiple heterogeneous clouds.
- ***Unified application management of services distributed over different clouds.*** SeaClouds will be able to deploy, manage, scale and monitor services over technologically diverse cloud providers. Such operations will be performed by taking into account application requirements and by providing developers with support beyond the handling of single services.
- ***Monitoring and run-time reconfiguration operations of services distributed over multiple heterogeneous cloud providers.*** Monitoring will be in charge of detecting the possible need of redistributing services across cloud providers. Dynamic reconfiguration will let orchestrations evolve so as to realize all the required changes. Reconfiguration ranges from dynamically replacing malfunctioning services to migrating them to different cloud providers.
- ***Compliance with major standards for cloud interoperability.*** SeaClouds will manage applications deployed on technologically diverse cloud platforms, unifying operations such as monitoring and lifecycle management, promoting the adoption of OASIS standards for cloud interoperability, in particular TOSCA¹ and CAMP².

¹ <https://www.oasis-open.org/committees/tosca/>

² <https://www.oasis-open.org/committees/camp/>

4 Final expected outcomes of the project

In order to achieve the aforementioned objectives, the following capabilities are expected as outcomes of SeaClouds:

- **Matchmaking.** To allow querying or browsing available (IaaS and PaaS) cloud offerings and select them based on application requirements.
- **Deployment optimizer.** To optimize the deployment topology of an application across multiple clouds to address non-functional requirements.
- **Application management.** To support efficient deployment and multi-cloud governance of complex applications on various (IaaS and PaaS) cloud offerings by leveraging cloud harmonized APIs and platform-specific adapters.
- **Monitoring and SLA enforcement.** To feature monitoring and independent metrics to allow users to monitor the health and performance of applications.
- **Repairing.** To scale both horizontally and vertically cloud resources to maximize the performance of each module of an application.
- **Replanning and application migration.** To provide a seamless migration of different application modules between dissimilar clouds.
- **Database migration and data synchronization.** To enable the portability of data among databases on different clouds.

5 Current status of the project

We now describe the platform architecture, SeaClouds main functionalities, the case studies used to validate the platform, and relations and collaborations with other projects, initiatives and standards.

5.1 Platform architecture

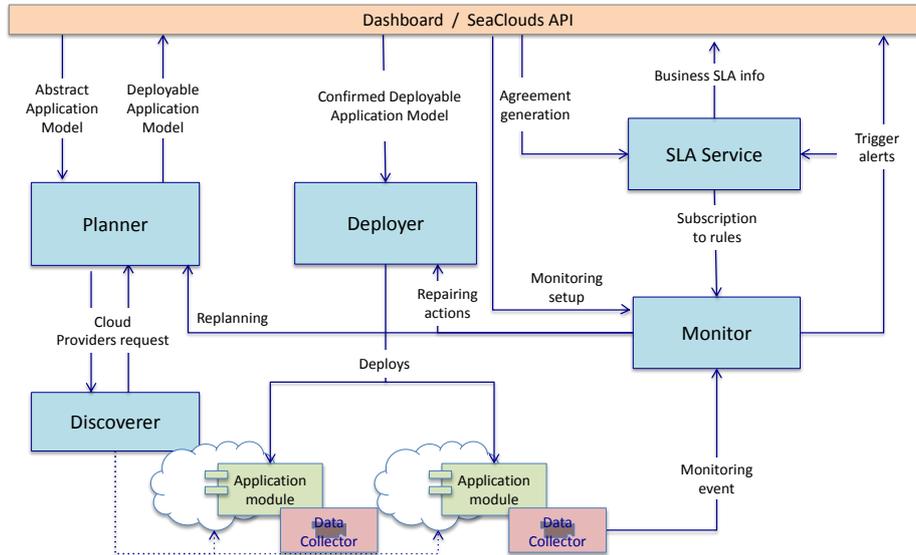


Figure 1. SeaClouds architecture.

Figure 1 illustrates the architecture and main components of SeaClouds. On top of the figure, there is the *Dashboard / SeaClouds API*, which provides a harmonized API and a dashboard to manage cloud applications with SeaClouds. This component orchestrates the different modules of the SeaClouds platform:

- *SeaClouds Discoverer*. It is in charge of discovering available capabilities and add-ons offered by available cloud providers.
- *SeaClouds Planner*. It is in charge of generating an orchestration plan taking into account the application topology and requirements.
- *SeaClouds Deployer*. It is in charge of executing deployment plans generated by the Planner.
- *SeaClouds Monitor*. It is in charge of monitoring that the Quality of Services (QoS) properties of the application are not violated by the clouds in which the application modules were deployed, and of determining -together with the Deployer- the reconfiguration strategies to trigger repairing or replanning actions.
- *SeaClouds SLA Service*. It is in charge of mapping low level information gathered from the Monitor into Quality of Business (QoB) information about the fulfilment of the SLA defined.

A distinguishing aspect of the SeaClouds architecture is that it builds on top of two OASIS standards initiatives: TOSCA, to specify the topologies and deployment plans, and CAMP, to interoperate with heterogeneous PaaS providers.

5.2 Design-time functionalities

The supported design time functionalities are:

- *Matchmaking*, implemented by the Planner component,
- *Deployment optimizer*, implemented by the Planner component,
- *Discovery*, implemented by the Discoverer component.

At design time, the designer specifies the Abstract Application Model (AAM), which is a TOSCA YAML model that defines the topology and requirements of the application to be deployed. Once the AAM has been specified, SeaClouds starts the Planning stage. The Planner acts with two sub-processes: Matchmaking and Optimizer (as described in D3.1³).

The matchmaking aims at finding, for each deployable module in the AAM, the list of candidate cloud offerings that fulfil the requirements for that module. To generate such list of cloud offerings, the planner interacts with the Discoverer. Then, the Optimizer selects one cloud offering from the list of candidates using metaheuristic algorithms. As a result, the Planner generates a Deployable Application Model (DAM), which specifies the concrete cloud services used to distribute the application.

³ SeaClouds Deliverable D3.1 - Discovery, design and orchestration functionalities: First specification and prototype. Available from <http://www.seaclouds-project.eu/deliverables.html>

5.3 Run-time functionalities

The supported run-time functionalities are:

- **Application management**, implemented by the Deployer component.
- **Monitoring and SLA enforcement**, jointly implemented by the Deployer, Monitoring and SLA Service components.
- **Repairing**, jointly implemented by the Deployer and Monitor components.
- **Replanning and application migration**, jointly implemented by the Deployer, Monitor, and Planner components.
- **Database migration and data synchronization**, integrated from the MODAClouds EU project⁴ and is integrated with the other elements of SeaClouds so that they can activate it.

The Deployer component, detailed in Deliverable D4.1⁵, executes the confirmed DAM, while the Monitor is configured with the monitoring rules taken from the user requirements. Also, the SLA service subscribes to rules or alerts and in connection with the Monitor, it enforces the policies of the agreements. The Deployer permits the deployment of the application's modules over heterogeneous IaaS and PaaS, and it tracks the dynamic evolution of the deployment and management of the application modules themselves, by means of a Live Application Model (LAM), as described in Deliverable D4.1.

Once the application is deployed, the Deployer manages it and instruments the Monitor. In particular, the Deployer installs DataCollectors in the cloud machine(s) in which the application modules have been deployed. A DataCollector component gathers raw monitoring data and pushes them to the Monitor. The latter component interacts with the SLA service to manage possible violations of QoS and QoB properties.

When a violation issue occurs, and it can be fixed without replanning, the Monitor and the Deployer interact to repair the issue. Otherwise, the Monitor interacts with the Planner to trigger the generation of a new plan to be executed by the Deployer.

5.4 Case studies

5.4.1 ATOS case study

The ATOS case study is a complex software application that aims at developing an innovative and integrated solution for the use of social inclusion tools by elderly people and for the general management (self-management included) of their medical problems. The currently existing application will be decomposed into a set of modules that will be deployed, monitored and administered in one or more cloud (IaaS and PaaS) cloud providers with the support featured by the SeaClouds platform.

From the re-engineering our health application, ready to be ported on the cloud through the SeaClouds platform, we expect three main benefits:

⁴ <http://www.modaclouds.eu>

⁵ SeaClouds Deliverable D4.1 - Definition of the multi-deployment and monitoring strategies. Available from <http://www.seaclouds-project.eu/deliverables.html>

- To reduce the vendor lock-in problem, by having the opportunity to choose and change competitive providers according to the changing resource requirements during the lifetime of the application,
- To execute the application on fully managed cloud infrastructure running in enterprise-class data centers with secure, reliable cloud services to support and deliver hosted software solutions, and
- To deliver flexible resources to accommodate fluctuations in customers' production workloads as well as on-demand capacity for training, demonstration, proof-of-concept or test/development.

5.4.2 NURO case study

NURO cloud game case study is based on a Nurogames Engine, which is responsible for data consistency and cheating protection. NURO will modify its monolithic server approach to get a cloud ready version.

The goal of the case study is to keep the functionalities of the currently available version, by taking advantage of cloud flexibility and of SeaClouds tools to optimize and to save administrative time and effort. Reliability, stability and security of the game server application should be improved by the SeaClouds solution. Saving resources in burst scenarios and a flexible pricing of real used resources will be possible.

The main benefits that we expect from porting NURO application to the cloud via SeaClouds are:

- Stable QoS. The flexible use of resources should guarantee the QoS on pick phases. Maintenance time should be optimized on rollout of new versions.
- Simpler administration and maintenance. Designers and administrators can focus on the application. They just have to configure the modules according to their needs and invoke deployment, installation and updates of the application. Administration of hardware and operating system is to be done by the SeaClouds system or the cloud provider.
- Scalability over time. If the game is accepted and running for a long time, increasing number of players will cause the server applications to consume more resources. A cloud solution should make it easy to scale the system to the users' needs. Moreover, on pick phases the QoS limits will be met and on burst phases wasting of resources will be avoided.
- Flexibility and control on payment. SeaClouds could considerably reduce costs, only paying for resources which are actually needed. Consuming and releasing resources depends on the needs to satisfy the QoS rules. We expect to set also cost limits to avoid exceeding budgets.

5.5 Relations and collaborations with other projects and standards

As discussed earlier, SeaClouds aims at covering different needs of application deployment and management over multiple IaaS and PaaS cloud providers.

As far as IaaS environments are concerned, the SeaClouds platform takes into account the main IaaS APIs and technologies, such as OpenStack⁶ and CloudStack⁷. To solve the problem of deploying multi-cloud in IaaS environments, SeaClouds is using Apache jclouds⁸ via Apache Brooklyn⁹. Recently, SeaClouds members have been contributing on adding the support for Docker and Microsoft Azure¹⁰ compute to Apache jclouds.

As far as PaaS environments are concerned, cloud developers are often locked into a specific cloud environment. To mitigate the vendor lock-in issues, several research projects try to wrap services offered by different providers in order to make them compatible. One of the most related projects in this regard is Cloud4SOA¹¹, which allows the management of PaaS services of different providers. Also, Cloud4SOA provides a set of cloud-independent monitoring mechanisms and a matchmaking methodology. SeaClouds takes advantage of Cloud4SOA and moves beyond it by defining its own matchmaking mechanisms for selecting target services that better adapt to the application components.

Another related project is CloudWave¹². With CloudWave, service providers will be able to rapidly design and deliver innovative, sustainable digital services for consumers at low cost and high quality. SeaClouds converges with CloudWave in the used distributed algorithms and data models that enable cloud infrastructures and applications to take actions in response to the dynamic changes in their environment. On the other hand, SeaClouds focuses on tackling the problem of deploying and managing complex multi-services applications over technologically heterogeneous clouds environments. It is worth mentioning SeaClouds and CloudWave organize a joint scientific workshop at ESOC 2015¹³.

Also, SeaClouds relates to MODAClouds¹⁴, which aims at providing quality assurance during the application's life-cycle, mitigating part of the portability vendor lock-in issues. To achieve its purposes, MODAClouds offers a monitoring methodology independent of any provider API. This monitoring methodology is going to be extended by SeaClouds project.

The peculiarity of SeaClouds is in the development of an almost market-ready product, adopting and extending Brooklyn (whose main developers are part of the

⁶ <https://www.openstack.org>

⁷ <https://cloudstack.apache.org>

⁸ <https://jclouds.apache.org>

⁹ <https://brooklyn.incubator.apache.org>

¹⁰ <http://azure.microsoft.com>

¹¹ <http://www.cloud4soa.com>

¹² <http://cloudwave-fp7.eu>

¹³ <http://esoc2015.unime.it>

¹⁴ <http://www.modaclouds.eu>

project) and employing OASIS TOSCA¹⁵ and CAMP¹⁶, as main standards for the deployment and management of cloud applications. SeaClouds members have been involved with CAMP and TOSCA technical committees, suggesting improvements to the specifications based on our activity. Also, SeaClouds members attend conference calls being held within the TOSCA monitoring subgroup, which focuses on extending TOSCA specification to describe monitoring aspects of an application.

Last, but not least, a cross-fertilization effort between Alien4Cloud¹⁷ and SeaClouds is currently ongoing to assess the possibility for Alien4Cloud to use SeaClouds deployer, and for SeaClouds to use Alien4Cloud TOSCA YAML parser.

A more thorough discussion of related projects is provided in deliverables D2.3.1¹⁸ and D2.3.2¹⁹.

6 Impact and Exploitation

SeaClouds adopted from the very beginning of the project an **open source strategy**, which includes transparent development, Apache 2.0 licensing, and developer engagement. This strategy also serves as enabler for three main impact paths:

- **Apache Brooklyn Integration and Extension for Application Deployment.** The Apache Brooklyn application management framework will not only serve as part of SeaClouds' Deployer component, but also as a vehicle for SeaClouds' post-project value proposition by supporting TOSCA and extending deployment capabilities to PaaS layer and multi-cloud scenarios. The SeaClouds-extended Apache Brooklyn will offer a more robust multi-cloud application deployment for European service providers to accompany their IaaS and PaaS service, strengthening a solution already supported in the market.
- **Contribution to Vendor-Driven Standards.** SeaClouds is actively contributing to two leading standards in cloud computing PaaS segment: OASIS CAMP and TOSCA. By extending these standards, SeaClouds is tapping a channel directly to Europe's cloud supply (vendors) to help empower multi-cloud scenarios for demand (their customers) via improved topology, orchestration, management and interoperability.
- **Consortium Exploitation Opportunities.** Project partners will be able to exploit SeaClouds' open-source solution to strengthen their commercial and research portfolios. Industry partners such as Cloudsoft and ATOS can integrate SeaClouds assets into their own selected commercial offerings, benefiting as primary contributors to development. Similarly, academic partners can propel their research in the multi-cloud domain.

¹⁵ <https://www.oasis-open.org/committees/tosca>

¹⁶ <https://www.oasis-open.org/committees/camp>

¹⁷ <http://alien4cloud.github.io>

¹⁸ SeaClouds Deliverable D2.3.1 - Periodic Standardization report.

Available from <http://www.seaclouds-project.eu/deliverables.html>

¹⁹ SeaClouds Deliverable D2.3.2 - Periodic Standardization report.

Available from <http://www.seaclouds-project.eu/deliverables.html>