Chapter 13

BPM in the Cloud: The BOC Case

Alexander Gunka, Harald Kuehn and Stepan Seycek

13.1 Introduction

To move an existing application to a Cloud-based operating model is a challenging task. This chapter presents a real life case in this domain. It is based on a case study from BOC which uses MODAClouds technology to enact four major use cases for the Cloud deployment of the BPM tool ADONIS. The first use case describes the provider selection in a multi-Cloud environment based on the decision support system Venues 4Clouds. In the second use case CloudML4Clouds is used to implement a model-based Cloud deployment procedure. The third use case shows the usage of Tower 4Clouds for real-time monitoring spanning various system levels to enable DevOps engineers to gather their custom monitoring metrics. The fourth use case describes the Cloud-to-Cloud migration process including the implemented approach for data migration aspects.

13.2 Context and Motivation

BOC group is a medium-sized software and consultancy company providing products and services for Business Process Management (BPM), Enterprise Architecture Management (EAM) and Governance, Risk, Compliance (GRC). BOC originated

A. Gunka · H. Kuehn · S. Seycek (✉)
BOC Information Technologies Consulting GmbH, Operngasse 20b,
1040 Vienna, Austria
e-mail: stepan.seycek@boc-group.com

A. Gunka
e-mail: alexander.gunka@boc-group.com

H. Kuehn
e-mail: harald.kuehn@boc-group.com

© The Author(s) 2017
E. Di Nitto et al. (eds.), *Model-Driven Development and Operation
of Multi-Cloud Applications*, PoliMI SpringerBriefs,
DOI 10.1007/978-3-319-46031-4_13

1995 as a spin-off company from the University of Vienna, Department Knowledge Engineering. Since then, BOC has grown to one of the leading companies within these domains, established operations in various European countries and maintains a world-wide customer base. In all their activities, BOC follows a model-driven approach.

ADOxx is BOC's meta-modelling platform for implementing the modelling products of the BOC Management Office by defining domain specific meta-models, by configuring specific behavior and adding functionality to complement a given methodology. Business users of the products can manage their model and object repositories in a collaborative way leveraging highly adaptable versioning and release workflows. They can create analytical views, define custom queries, and generate various reports interacting via web-based graphical editors and dashboards.

While most enterprise software is still deployed on-premises, Software-as-a-Service is expected to grow rapidly over the next years. According to IDC the total Cloud software market will grow to surpass \$100 billion by 2018 at a compound annual growth rate of 21.3 %. In order to be able to benefit from these new opportunities and from the advantages in terms of resilience, agility and cost efficiency the Cloud promises, BOC committed to a strategy for providing their applications as SaaS in addition to their existing sales and operation models. In order to minimize risks BOC decided to apply an iterative process to achieve this target [1]. Technology developed within the MODAClouds project plays an important role in achieving this step of business model extension.

As one of the first steps implementing this strategy, a prototypical instantiation of a process modelling language using the ADOxx meta-modelling platform has been ported to the Cloud with the help of tools and methodologies developed within MODAClouds. Based on the results of this evaluation, BOC has recently moved the solution to production environment by launching ADONIS:cloud [2].

13.3 Application Scenario

During the process of defining requirements to be supported by MODAClouds, four main use cases (depicted in Fig. 13.1) were identified with regard to BOC's application scenario. First, the *selection of Cloud providers* should be simplified with the help of decision support tools and methodologies. After this step MODAClouds should provide assistance to *deploy* a given application stack to selected Clouds in an automated, Cloud provider independent way. Advanced *monitoring* techniques should then be used to track system health and quality of service. In case of detected violations another *cloud provider* would be *selected* and the application should be *re-deployed* to the new provider. In addition, data would be *migrated* and traffic would be switched to the new deployment.

Fig. 13.1 BOC's main use cases to be supported by MODAClouds to offer ADONIS as SaaS

13.3.1 Cloud Provider Selection

Soon after the decision to offer the BPM solution ADONIS as SaaS, BOC was facing the challenge of selecting an appropriate IaaS provider. They started the decision making process by collecting various decision criteria taking into account operations, legal, cost, and sales related aspects. They assigned relative weights to these criteria and marked some of them as must-have features. As a next step around 20 candidate providers were considered and rated first only considering the must-have criteria which helped reducing efforts in data collection by ruling out a large portion of the initial candidates list. For the remaining candidates the rest of the data was collected and the three providers with the highest rankings were presented to the decision team for discussion.

Retrospectively some of the weaknesses of this intuitive approach soon became obvious [3]:

- The cost of re-evaluating the providers on a regular basis in order to improve quality of service and cost efficiency would be very high.
- The goal of covering different markets will require addressing strict data location policies and therefore the ability to deploy to multiple Clouds. These aspects are hard to address using the initial approach since it would require a much larger number of providers to be analysed resulting in very high efforts in data gathering and analysis.
- The criteria chosen intuitively based on BOC's own experience were not comprehensive enough to cover all relevant aspects.
- In particular, the ease of migration from one provider to another, which should have been considered as one of the most important criteria, was ignored.

BOC contributed these experiences during the requirements elicitation process for Venues 4Clouds, MODAClouds Decision Making System for Cloud Service Offer Selection which is described in detail in Chap. 2 [add reference]. In particular, BOC contributed a number of functional requirements, an initial set of decision criteria, and general success criteria for a Decision Support System to be developed [4].

Based on the requirements gathered, a risk-driven framework for decision support [3], a methodology for eliciting risk, cost and quality aspects in multi-Cloud environments [5, 6], and a prototype of a Decision Support System (DSS) have

Table 13.1 Stakeholders and assets considered in the DSS

Stakeholders	Type of intangible asset	Assets considered
Business representatives	Business oriented intangible assets	Customer loyalty, legislation compliance, internal efficiency and performance, sales rate, market awareness, improve product innovation and quality
Technical (DevOps) representatives	Technology oriented intangible assets	Data privacy, data integrity, maintainability, end user performance, service availability, cost stability

been developed. BOC used this methodology and the prototypical DSS to perform a light weight risk analysis by first defining a set of relevant business-oriented and technology oriented assets, determining risks related to these assets, and treatments mitigating these risks [4]). During this assessment process both business and technical stakeholders were involved. Table 13.1 lists the assets considered by either of these stakeholders.

In that way, for example customer *loyalty*, which was identified as an important asset by business representatives, has been related to the technical oriented assets *data privacy*, *data integrity*, *end user performance*, and *service availability*. *Data privacy*, in turn, has been related to the risks *unauthorized access from IaaS provider*, *insufficient isolation*, *insufficient physical security*, and *data exposure to government authorities*. As possible mitigations for these risks the *availability of certificates guaranteeing information security* and the *possibility to select a specific data location* were selected. A complete mapping of all identified assets, risks and mitigations can be found in the appendix of MODAClouds public deliverable 2.3.2 [7].

The fact that the user is guided through a structured process, starting with high level assets helped to identify a larger set of risks and, consequently, treatments to be considered, that would not have been detected using the original intuitive approach. At the same time the process turned out to be simple enough and well-guided by the tool to be usable for a small or medium enterprise (SME) such as BOC.

Since the current version of the DSS (Venues 4Clouds) is at prototype stage only, both the data gathered about Cloud providers and the pool of assets and risks already predefined in the tool are in no way complete at the time of writing. Provided that the systems knowledgebase will grow over time and that the data available can be maintained up to date and accurate, e.g. by employing self-learning mechanisms and being able to extract data from multiple online sources, it will be able to assist SMEs such as BOC in continuously keeping track of a large number of potential providers and in analysing them in a cost efficient way.

13.3.2 Application Deployment to Multiple Clouds

Before the start of the MODAClouds project BOC already had gathered some experience with deployment automation by employing the configuration management system Puppet to automate software installation in their application hosting environment. However, when it came to deployment to the Cloud and, in particular, to multiple Clouds, they soon realized some shortcomings of their approach:

- Even though Puppet provides configuration modules for different Cloud stacks like e.g. OpenStack, provisioning of IaaS instances in a totally Cloud-stack-independent, transparent way was hard to accomplish.
- Deploying parts of the application (e.g. the database tier) to PaaS would be even harder since it would require the use of another configuration to deploy the application stack to PaaS.

As soon as a first version of MODAClouds design time component CloudML4Clouds which is responsible for deploying the modelled application was available, BOC started evaluating the component and found the following potential benefits:

- It would enable them to automatically deploy to any of the Cloud systems previously selected by Venues 4Clouds including the provisioning of Cloud instances in a transparent way, resulting in a higher degree of automation and consequently less manual operation efforts.
- The model-based nature of the approach was expected to enable BOC to document individual deployments in a traceable and comprehensible way and support them in explaining deployment decisions to their customers.

However, it also turned out that using CloudML4Clouds to deploy on Windows Virtual Machines—one of the application components to be deployed in BOCs case is a Windows application—had some disadvantages compared to the use of Linux based VMs. In particular, the tool is relying on the Windows Remote Management protocol to execute remote commands and to deploy to a Windows VM which in turn requires CloudML4Clouds to be executed on a Windows machine. BOC had already invested in developing scripts for Puppet deploying parts of their application to Windows and preferred to capitalize on these investments rather than having to re-implement these scripts using Windows PowerShell commands. Hence, they decided together with the partners involved in developing the component to start an initiative to integrate CloudML4Clouds with Puppet modules.

At the time of writing all required extensions needed for deploying software components through Puppet on Windows VMs are available and BOC currently evaluates the usability of the Puppet extension to CloudML [8].

Since Puppet and similar tools like Chef are widely used in the DevOps community BOC expects the possibility to integrate them with CloudML4Clouds to be beneficial for a large number of potential users.

13.3.3 *Cloud Application Monitoring*

Before deciding to push the SaaS business, BOC already had a basic monitoring solution in place for several hosting projects. It was based on infrastructure performance indicators like CPU load and RAM consumption as well as the basic availability of the customer facing frontends and central backend components. When going larger scale with a full-fledged SaaS platform, the system health and performance need to be tracked more in detail in order to be able to mitigate issues at an early stage. Such a monitoring solution needs to collect application specific data and in some cases combine metrics from various sources in order to provide the comprehensive view of the whole system required by the operations engineers.

This is where MODAClouds' monitoring technology Tower 4Clouds (Chap. 5) has been introduced with great success. Its Resource Description Framework (RDF) based streaming technology together with a flexible approach to configure data collectors and data analysers provide a solid framework for the challenges of a reliable and extensible monitoring platform.

Tower 4Clouds can be used as part of the complete MODAClouds toolset including design time quality constraint modelling, CloudML deployment and runtime self-adaptation. However, the MODAClouds monitoring components can as well become building blocks for tailored environments, even if the other parts of Energizer 4Clouds are not being used. In the case of BOC's SaaS platform it has been integrated with the existing solution based on the open source tool Icinga.¹ This solution allows the operations team to continue working with the well-established Icinga frontend and its service recovery mechanisms while obtaining more details about the state of the platform. Furthermore, MODAClouds' concept for data collectors makes it very easy for the DevOps engineers to extend the number and the types of metrics collected.

The integration of the stream based monitoring framework with the Icinga poll-based metrics acquisition has been implemented by leveraging the observer interface for Tower 4Clouds and providing a generic Icinga plugin for retrieving the collected data from the observer. With such a toolset adding a new metric that can be acquired with an existing data collector is just a matter of extending the data collector's configuration accordingly and defining a corresponding service within Icinga.

An additional benefit from the usage of MODAClouds' technology for monitoring is that all streams that represent numeric time-series data can be also directed into the Metrics Explorer (see Chap. 5), a web based graphing tool. This can be very useful for the operations engineers when a retrospective view at system performance indicators is needed and trends are to be interpolated.

The monitoring solution architecture for BOC's SaaS platform based on Tower 4Clouds, Icinga and the Metrics Explorer is depicted in Fig. 13.2.

¹<https://www.icinga.org/icinga-2/distributed-monitoring/>.

Fig. 13.2 Monitoring solution architecture for BOC's SaaS platform

13.3.4 Cloud to Cloud Migration

One of the key motivations for choosing IaaS technology over other models such as housing is the shift from capital expenditure (CAPEX) to operational expenditure (OPEX). This enables service providers to relocate their services among multiple infrastructures without losing investments. For BOC's Cloud services this capability is an essential asset as it allows for cost optimisation and it also can be the right approach for dealing with availability or performance issues encountered within a specific infrastructure. In addition there is another valid use case for Cloud-to-Cloud migration: customers especially in the public administration sector are confronted with changing regulatory policies related to location of services and data they use for their work. Such policies may at some point in time prohibit usage of services and storage of data outside the respective country.

BOC's SaaS platform has been designed with the objective to be extensible to sites located in different countries if customers have the need to have their data stored and services deployed in specific geographical locations. This is achieved by relying on basic IaaS for compute, storage and network resources managed through CloudML which in turn triggers Puppet [9] for deploying and configuring the BOC services. This Cloud vendor independent toolchain enables BOC to move their services among IaaS platforms from various providers.

The Cloud-to-Cloud migration approach chosen by BOC is one that relies on data replication mechanisms of the used database management system (MS SQL Server or Oracle Database), CloudML for deploying application stacks in multiple sites and the REST-enabled MODAClouds load balancer for switching the traffic from one site to the other. The complete procedure that needs to be executed to perform the switch from site A to site B consists of the following steps.

1. Create a deployment model for site B including a load balancer instance pointing at site A and B application stacks as well as a DBMS instance. Update the deployment model for site A adding the application stack of site B to the load balancer.
2. Enact the deployment one both sites with CloudML, then create a full database backup on site A and restore it on site B.
3. Update the deployment model of site A to remove the application stack. Enact the change of site A with CloudML (start of downtime). Create a differential database backup on site A and restore it on site B.
4. Update the deployment model for site B adding the application stack. Enact the deployment change on site B with CloudML (end of downtime). Trigger the DNS switch for the publicly accessible domain so that the user traffic is routed to the load balancer on site B.
5. Once all traffic is on site B update the deployment model on site A removing the DBMS and the load balancer as well as all underlying IaaS resources. Release all resources on site A by enacting the deployment model with CloudML.

The main steps of the relocation are depicted in Fig. 13.3.

As all of the involved steps are scriptable, automation is possible if the frequency of the migration use case justifies the effort for implementing the automated solution based on the procedure described above.

Fig. 13.3 Main steps of the Cloud-to-Cloud relocation

13.4 Conclusion and General Recommendations

Even though some of the experiences and observations made are specific to the case described in this chapter, the authors believe that some general recommendations can be derived for companies or, more specifically, SMEs that are either planning to cloudify some of their business critical software or, being a software provider such as BOC, to extend their business model with an SaaS offering.

As mentioned earlier there are several good reasons to think about Cloud application monitoring as well as a strategy to migrate from one Cloud provider to another from the very beginning of the Cloud migration process. This should encompass the following aspects:

- When selecting a particular Cloud service, the ease of migration to another equivalent service should be considered. This implies on one hand the existence of such services and on the other hand the ability to migrate software components and data to these other services easily.
- In order to increase cost efficiency and quality of service the Cloud service provider market should be analysed on a regular basis. The selection of Cloud services and Cloud providers might become a reoccurring task. A common knowledge base and a tool based approach for decision making as planned for MODAClouds Venues 4Clouds tool will help saving efforts for data acquisition and analysis and making decisions in a traceable, comprehensible way.
- In order to be able to easily deploy to different providers deployment automation or even self-adaptation should be considered to save operation efforts. Automated deployment should ideally work on different Cloud stacks (i.e. on different Cloud services) with as little adaptations as possible.
- Monitoring should be considered an integral part of the Cloud service as it is the only reliable way to track SLA adherence. The solution should be easy to use, maintain, extend and at best it shall be managed together with the business application by means of the configuration management system. The combination of an established product such as Icinga with the sophisticated Tower 4Clouds RDF stream processing toolkit is a good candidate for this challenge.

References

1. Alexander Gunka SS (2013). Moving an application to the cloud—an evolutionary approach. In: MultiCloud'13. Prague, Czech Republic
2. BOC (2014) BOC Group: ADONIS:cloud Landing page. Retrieved May 2015, from <http://www.boc-group.com/at/adoniscloud> (2015). Retrieved May 2015, from <https://www.icinga.org/>
3. Smrati Gupta VMM (2015, May 5–7). Risk-driven Framework for Decision Support in Cloud Service Selection. In: 15th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CC-GRID (2015) Shenzhen. Guangdong, China
4. MODAClouds (2013) Deliverable 2.3.1: decision making toolkit requirements and architecture, and update on business methodology. Retrieved May 2015, from MODAClouds: <http://www.modaclouds.eu/publications/public-deliverables/>

5. Omerovic A, Muntès MV (2013) Towards a method for decision support in multi-cloud environments. In: Proceedings Fourth International Conference on Cloud Computing, Grids, and Virtualization (CLOUD COMPUTING 2013) pp 162–180
6. Muntès Mulero VPM (n.d.) Eliciting risk, quality and cost aspects in multi-cloud environments. In: Proceedings Fourth International Conference on Cloud Computing, Grids, and Virtualization (CLOUD COMPUTING 2013) pp 238–243
7. MODAClouds (2014) Deliverable 2.3.2: decision making toolkit requirements and architecture, and update on business methodology. Appendix A. Retrieved May 2015, from MODAClouds: http://www.modaclouds.eu/wp-content/uploads/2012/09/MODAClouds/_D2.3.2/_-DecisionMakingToolkitRequirementsAndArchitectureAndUpdateOnBusinessMethodology.pdf
8. MODAClouds (2015) D4.3.3 MODACloudML IDE—final version. Retrieved May 2015, from http://www.modaclouds.eu/wp-content/uploads/2012/09/MODAClouds_D4.3.3_MODACloudMLIDEFinalVersion.pdf
9. Puppet (2015) Puppet. Retrieved May 2015, from <https://puppetlabs.com>

Open Access This chapter is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, duplication, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the work's Creative Commons license, unless indicated otherwise in the credit line; if such material is not included in the work's Creative Commons license and the respective action is not permitted by statutory regulation, users will need to obtain permission from the license holder to duplicate, adapt or reproduce the material.