



Title: MODAClouds evaluation report – Final version

Authors: Nicolas Ferry, Arnor Solberg (SINTEF), Pooyan Jamshidi, Rasha Osman, Weikun Wang (Imperial), Stepqn Seycek (BOC), Vanessa Gligor (Siemens), Roi Sucasa (ATOS), Antonin Abhervé (SOFTEAM)

Editor: Nicolas Ferry (SINTEF)

Reviewers: Daniel Pop (IeAT), Roman Sosa (ATOS)

Identifier: Deliverable # D3.7.2

Nature: Report

Version: 2

Date: 04 November 2015

Status: Final

Diss. level: Public

Executive Summary

This report describes to which extent the MODAClouds tools and methods fulfil the requirements identified by the case studies and at WP level. The deliverable is based on the outcomes of the evaluation methodology developed in the Evaluation plan (D3.6), which is based on the Basili's Goal Question Metric (GQM) method and software testing techniques. The evaluation assesses one by one the goals defined in the Evaluation plan for each MODAClouds artefact. Compared to the initial Evaluation report, new goals and questions have been identified and evaluated by the case study partners and at the WP level. Finally, the MODAClouds main objectives and the Key Performance Indicators (KPI) as defined in the DoW are evaluated.

Copyright © 2015 by the MODAClouds consortium – All rights reserved.

The research leading to these results has received funding from the European Community's Seventh Framework Programme [FP7/2007-2013] under grant agreement n° 318484 (MODAClouds).

Members of the MODAClouds consortium:

Politecnico di Milano	Italy
Stiftelsen Sintef	Norway
Institutul E-Austria Timisoara	Romania
Imperial College of Science, Technology and Medicine	United Kingdom
SOFTEAM	France
Siemens SRL	Romania
BOC Information Systems GMBH	Austria
Flexiant Limited	United Kingdom
ATOS Spain S.A.	Spain
CA Technologies Development Spain S.A.	Spain

Published MODAClouds documents

These documents are all available from the project website located at <http://www.modaclouds.eu/>

Contents

CONTENTS	3
1 INTRODUCTION	5
1.1 CONTEXT AND OBJECTIVES.....	5
1.2 STRUCTURE OF THE DOCUMENT.....	5
2 MODACLOUDS SOFTWARE ARTEFACTS EVALUATION	5
2.1 VENUES 4CLOUDS.....	6
2.2 CREATOR 4CLOUDS	7
2.2.1 MODAClouds IDE	7
2.2.2 SPACE ^{Dev} 4Cloud and LINE	9
2.2.3 Cloud Modelling Language and deployment engine.....	13
2.3 ENERGIZER 4CLOUDS.....	15
2.3.1 Tower 4Clouds.....	15
2.3.2 Space ^{Ops} 4Clouds.....	17
2.3.3 ADDapter 4Clouds.....	21
3 SUMMARY OF THE EVALUATION OF MODACLOUDS RESULTS FROM THE CASE STUDIES PERSPECTIVE	24
3.1 BUSINESS PROCESS MODELLING SYSTEM	24
3.1.1 Exploiting MODAClouds results.....	25
3.1.2 Final evaluation summary.....	26
3.2 SMART CITY URBAN SAFETY PLANNER.....	26
3.2.1 Exploiting MODAClouds results.....	26
3.2.2 Final evaluation summary.....	27
3.3 HEALTH-CARE APPLICATION	27
3.3.1 Exploiting MODAClouds results.....	28
3.3.2 Final evaluation summary.....	28
3.4 MODELIO PROJECT MANAGEMENT SERVER	29
3.4.1 Exploiting MODAClouds results.....	29
3.4.2 Final evaluation summary.....	30
4 MODACLOUDS OBJECTIVES EVALUATION	32
4.1 MODACLOUDS MAIN OBJECTIVE	32
4.2 MODACLOUDS SUB-OBJECTIVES	33
5 MODACLOUDS KPI EVALUATION	37
6 CONCLUSIONS	44
BIBLIOGRAPHY	45
GLOSSARY	46
7 ANNEX A - MODELLING OF REPLICATION USING LINE RANDOM ENVIRONMENTS	47
7.1 MOTIVATION AND PROBLEM STATEMENT.....	47
7.2 METHODOLOGY	47
7.3 IMPACT OF REPLICATION	48
7.4 LINE MODEL.....	48
7.5 RESULTS.....	48

8	ANNEX B – MULTI-CLOUD LOAD BALANCER	49
8.1	MOTIVATION AND PROBLEM STATEMENT	49
8.2	DATA AND TESTBED	49
8.3	METHODOLOGY	50
8.4	CASE STUDY RESULTS	50
8.5	MORE EXPERIMENTAL RESULTS	52
8.6	CONCLUSIONS	54
9	ANNEX C – NEW GOALS, QUERY AND METRICS FOR TOWER 4CLOUDS.....	55

1 Introduction

1.1 Context and objectives

This deliverable follows the initial evaluation report D3.7.1 [12] and presents the final evaluation of the MODAClouds software solution on the basis of the requirements expressed by the case study partners and at the WP level. These requirements have been defined in the Evaluation plan deliverable D3.6 following the Goal-Question-Metric (GQM) [3] methodology. Moreover, since the release of D3.6, the MODAClouds partners have introduced new requirements. Their evaluation is also presented in this deliverable. In particular, each MODAClouds software artefact is assessed with respect to the goal defined in the evaluation plan. Table 1 provides an overview of these artefacts.

Table 1 List of the MODAClouds artefacts evaluated

Artefact	Module	Description
Venues 4Clouds	Decision Support System (DSS)	Allows the users to express their requirements, processes these requirements via risk-based analysis and presents a list of sets of cloud services that are most suitable to meet those requirements
	Creator 4Clouds	
	MODAClouds IDE	Allows developers to specify MODACloudML models at all the abstraction levels (CCIM, CPIM and CPSM).
	Space ^{Dev} 4Clouds	Offers support for the specification, assessment and optimisation of QoS characteristics for Cloud applications
	LINE	Allows users to evaluate the performance of the candidate architecture and deployment of a multi-cloud application.
	CloudML	A domain specific language to specify the provisioning and deployment topology of multi-cloud applications. In addition it can be used to trigger the initial deployment of the application.
Energizer 4Clouds	Tower 4Clouds	Is responsible for collecting, analysing, visualizing and storing monitoring information at runtime.
	ADDapter 4Clouds	Offers a set of provider independent cloud services that can be exploited at runtime by multi-cloud application thus supporting their execution
	Space ^{Ops} 4Clouds	<i>SpaceOps 4Clouds</i> is a multi-cloud self-adaptation and policy reconfiguration engine providing QoS prediction and topology optimization.

Finally, this deliverable report the status of the MODAClouds software solution with respect to the Key Performance Indicators (KPIs) and main objectives of the project as defined in the DoW.

1.2 Structure of the document

The remainder of the document is structured as follows. The next section presents the result of the evaluation of each artefact listed in Table 1. Section 3 discusses the status of the MODAClouds objectives, while section 4 presents the assessment of the MODAClouds KPIs. Finally, the last section of the document provides concluding remarks.

2 MODAClouds software artefacts evaluation

In the following sections, each MODAClouds software artefact is evaluated with respect to the goals defined for it. The content of each sub-section is a synthesis of the experiments performed by the MODAClouds partners, laid down in a table format with the following fields:

- Who – list of partners performing experiments with respect to the artefact
- Overall – overall assessment of the evaluation process status w.r.t. evaluated artefact, plus main outcomes
- Goal – the outcome of the evaluation of the specified goal
- Deviations – delays versus the schedule included in the evaluation plan
- Recommendations – action points and suggestions for future improvements.
- Evolution – main evolutions in the evaluation since M24

The reader is referred to D3.6 Evaluation plan [2] for a detailed description of the goals and related questions.

2.1 Venues 4Clouds

Who:	WP2 (CA) Health-care application (ATOS) BPM System (BOC)
Overall:	Venues 4Clouds has been evaluated by CA at the WP level and by ATOS and BOC as case study partners. With respect to the three goals defined in the evaluation plan for this tool in WP2, the outcome of evaluation shows their fulfilment. Each goal is evaluated by two questions with one metric per question. All questions have been evaluated
Goal 1	Evaluate DSS with respect to addressing different types of multi-cloud deployment architectures
	This goal is considered as achieved. Two main deployment architectures were defined, multi-cloud deployment and replication, respectively, which were also implemented in the user interface of the tool.
Goal 2	Evaluate DSS with respect to the risk analysis methodologies
	This goal is considered as achieved. A 5-step approach for guiding the user through the process of selecting the most suitable Cloud Service Provider was implemented and extensively described in D2.1.2. (BOC) From the case studies point of view, a sufficient set of assets, risks, and mitigation is covered.
Goal 3	Evaluate DSS with respect to simplification of decision making process
	(CA) Venues 4Clouds adapts the methods of TOSCA cloud selection mechanism and adapts it to the risk based modelling used to define the characteristics selection. Venues 4Clouds is structured around the idea of guiding the user through the process of selecting cloud providers with the help of a wizard. This allows the user to address each of the selection criteria as well as allows different actors participating in the selection to describe the different requirements and

assets needed to fulfil the needs of the deployment.	
Deviations	
One of the metrics defined by ATOS concerned the ability of the tool to measure the difference between actual costs of deploying their case study on a specific PaaS provider with respect to the predicted one. However, such feature is out of the scope of the tool.	
Recommendations	
(ATOS) suggests extending the list of PaaS services and providers supported, in particular for application containers.	
(BOC) suggests that it would be beneficial for a general use of the tool to provide mechanisms to extend on a regular basis the number of assets, risks, and mitigations.	
Evolution	
Q2 from the third goal was postponed to M36 and has been positively evaluated.	
In addition, since the ATOS case study relies on PaaS services, the evaluation of Venues 4Clouds was postponed to M36. This evaluation has now been performed and shows the fulfilment of the goals.	
Similarly, at M24 the evaluation of this tool by BOC was postponed. The evaluation has been performed and the tool applied to the BPM System case study. Overall it shows that the tool fulfil the expected goals.	

2.2 Creator 4Clouds

2.2.1 MODAClouds IDE

Who:	BPM System (BOC) MODELIO Project Management Server (SOFTEAM) Smart City Urban Safety Planner (SIEMENS) Health-care application (ATOS) WP4 (SOFTEAM, SINTEF)
Overall:	
The MODAClouds IDE has been used and evaluated by all case study partners by M24. In particular, details on the specified models can be found in a set of eight deliverables, D8.{2, 3, 4, 5}.{1, 2} delivered at M18 and M24. Nevertheless, all new features have been evaluated. In addition, the IDE has been evaluated at the WP level.	
No deviations from Evaluation plan were observed. The goal defined for this MODAClouds artefact is evaluated as fully achieved by all involved partners.	
Goal 1	Evaluate MODAClouds IDE with respect to cloud provider selection, modelling

	and code-generation
	<p>(BOC) Models describing all relevant aspects of the case study have been created using the IDE. The case study models are stored in the MODAClouds model repository and are detailed in the final case study design deliverable D8.3.2 as well as in D9.2.2. Overall, the IDE provides enough modelling support on all the levels (CCIM, CPIM, CPSM) as far as BPM System case study is concerned. The support for automated transformation and refinement is considered mostly adequate, and few suggestions for possible minor improvements are provided (see Recommendations section).</p> <p>(SOFTEAM) A detailed evaluation of Constellation Project Management Server case study has been performed and is extensively documented in D8.2.2. CPIM models involving PaaS and IaaS solutions were built by utilizing the IDE and successfully fed to QoS evaluation components. CPSM deployment models targeting Amazon and Flexiant, involving PaaS DBs provided by both of them, were successfully designed as well and successfully fed to runtime components (Tower 4Clouds, SLA manager and Models@Runtime engine).</p> <p>(ATOS) The IDE has been successfully used to specify all types of MODACloudML models except data model (not in the scope of the CS). Overall the IDE tool provides sufficient modelling support at all the levels as far as this case study is concerned. Models are documented in D8.4.2. The support for automated transformation and generation is considered adequate.</p> <p>(SIEMENS) Joining other CSPs, SIEMENS evaluates the modelling support offered by IDE at all abstraction layers (CCIM, CPIM and CPSM) as good. Deployment models are documented in D8.5.2. Deployment models were successfully fed to the runtime environment without manual intervention.</p> <p>(WP4) All models were successfully implemented using the MODAClouds IDE.</p> <p>Regarding the integration of the IDE with the other MODAClouds tools, SPACE^{Dev} 4Cloud input files are generated from Service Definition Model and Service Orchestration Model and refinement transformations are performed successfully. Monitoring rules are also successfully generated from QoS constraints and fed to Tower 4Clouds. Finally, models are also successfully exchanged between the IDE and Venues 4Clouds, the SLA tools and the Models@Runtime engine. As specified in the evaluation plan, generations and refinement mechanisms have been tested at least against 5 different cases.</p> <p>While these transformations are fully automated, model transformations refining deployment models are only semi-automated. Information that cannot be derived from the model (i.e. the choice of cloud providers and services, the reuse of external services) needs to be manually entered by end-users. Case studies providers consider these human interventions as adequate.</p> <p>Since the initial evaluation, the generation and refinement mechanisms have been improved; several partners have reported bugs that have been fixed incrementally. As a result, the robustness of the tool has been improved and the generation, transformation and refinement mechanisms are considered as working properly.</p>
Deviations	
	No deviations were observed.
Recommendations	
	(BOC) After applying manual changes, although model validation had produced no errors, generation of Instance level deployment from type level failed silently in some cases. Validation mechanisms could be improved in that regard.
Evolution	

(BOC) Based on the outcome of the M24 evaluation and based on new features implemented in the IDE, the evaluation has focussed on the following topics: CCIM interface descriptions and CPIM type level deployment model. Regarding interface descriptions, for each operation resource requirements have been successfully modelled as input for the QoS modelling and analysis tools. Type level deployment models have been successfully modelled. In particular the new concept *PuppetResource* allows for integration of the CloudML deployment approach with the configuration management tool Puppet. This integration has been successfully tested with the case study application.

Generation of models has been re-evaluated based on the results of the initial evaluation. Deployment has been tested for different scenarios, partly with the use of the new concept *PuppetResource*. The models created as a result of the transformations and manual steps can be found in the MODAClouds model repository and are described in deliverable 9.2.2.

Transformation from one level to the next worked well as long as no manual extensions and changes have been done on the source model. Although model validation had produced no errors, generation of Instance level deployment from type level failed silently in some cases.

Human involvement is required in particular to add additional middleware components like Apache Tomcat to be deployed in a VM, which are not actually part of the application and therefore not described at CCIM level. This kind of flexibility is actually considered as an important feature rather than a weakness.

(SOFTEAM) Models involving PaaS and IaaS support were constructed by means of the IDE. They were successfully fed to design and runtime tools for evaluation. In particular, monitoring rules can be delivered to the runtime platform by the use of the IDE and deployment model to the Models@Runtime engine.

(ATOS) The evaluation of the generation and transformation feature was postponed to M36. A particular focus has been given to the generation of monitoring rules and self-adaptation policies. This feature has been successfully tested as far as the case study is concerned.

(SIEMENS) As a first evaluation round, deployment models were edited and pushed manually to the deployment engine. For the final evaluation, these models have been successfully generated and fed to the runtime tools from the IDE.

2.2.2 SPACE^{Dev} 4Cloud and LINE

Who:	BPM System (BOC) MODELIO Project Management Server (SOFTEAM) Smart City Urban Safety Planner (SIEMENS) WP5 (POLIMI, IMPERIAL, IeAT)
Overall:	
The QoS modelling and analysis tools were assessed both internally by WP5 partners, and externally by CSPs (BOC and SOFTEAM). The goals and questions defined for this MODAClouds artefact are evaluated as fully achieved by all involved partners.	
Goal 1	Evaluate MODAClouds monitoring approach and tools (BOC)
The MODAClouds IDE provides enough modelling support for quality constraints modelling and it supports their assignment to services at CCIM level and nodes at CPIM and CPSM level. Moreover,	

<p>the IDE provides adequate transformations from CCIM QoS constraints to CPSM monitoring rules.</p> <p>Monitoring rules can be delivered by the use of the IDE, but they need to be completed manually w.r.t the triggered actions in case the corresponding monitoring rules are not respected.</p> <p>Data collectors are successfully and automatically deployed when modelled as part of the deployable components in the system deployment model.</p>	
Goal 2	Evaluate MODAClouds QoS and Cost analysis tools (SOFTEAM)
<p>LINE has been evaluated against an open source web application OFBiz deployed on Amazon EC2 and the application and database server are deployed on the same VM. The input parameters of LINE are generated from the techniques developed in the Feedback Loop, which have been validated against operational data and shows excellent accuracy, as we published in a recent IEEE Transactions on Software Engineering paper. For comparison, we build a simulation model generated from the queueing network model created by LINE, which reflects the exact performance.</p> <p>We compare the response time CDF generated from the simulation model and LINE together with the trace CDF. Under 50 users, we a clear match between LINE and simulation, both of which closely follow the trace CDF. For 70 and 90 users, both LINE and simulation are also very close to the trace CDF and the errors are still less than 15%. We have also applied LINE to the BOC case study for a data migration case study to Amazon RDS, obtaining equally low errors.</p>	
Goal 3	Evaluate the QoS Modelling and Analysis Tool installation
<p>The installation experiments included the installation of the Palladio Bench together with Eclipse Kepler, and SPACE^{Dev} 4Clouds. The two case study partners (BOC, SOFTEAM) that have been using the QoS Modelling and Analysis Tool as well as the WP5 partners have been able to install the tools. However, SOFTEAM reported that it required help from developers.</p>	
Goal 4	Evaluate QoS requirements, constraints definition and performance evaluation
<p>The QoS requirements and constraints supported by the tool satisfy the needs of the case studies.</p> <p>(BOC) Additional constraints have been considered specific to our case study. Manual transformation for such kind of constraints is necessary which is acceptable from the case study's point of view.</p>	
Goal 5	Evaluate LINE with respect to the generation and solution of QoS models
<p>This goal has been fully evaluated at M24. We report below the result of this evaluation.</p> <p>Basically, LINE supports the solution of QoS models from the MODAClouds application models, but additional transformations are required for this purpose.</p> <p>In experiments ran up to now, LINE proved to be a robust solver. It reports results similar to those obtained by LQNS when solving the same model. However, LINE provides additional features comparing to LQNS (e.g. Random Environments, Percentiles). On top of this, LINE also supports random environments that are used to model uncertainty. With respect to percentile support in LINE, the development version of LINE supports percentiles of response time at workload and functionality levels. Support for percentiles in functionalities with probabilistic branches is under development.</p>	
Goal 6	Evaluate the Feedback loop module with respect to its functionality
<p>This goal is assessed as fully achieved.</p>	

It is considered that the Feedback loop (FL) module support the estimation of parameters relevant for the QoS models built from the MODAClouds models. The FL implements a set of resource demand estimation algorithms, which depend on different types of monitoring data. A comparison between different algorithms with different sizes of data is made in D5.3.2.

In addition, the FL module is successfully integrated with Tower 4Clouds and extracts monitoring data from the RDF History DB. This allows the FL module to analyse the service resource demand and to update the corresponding value in the PCM and LQN models, which are later used by the QoS Modelling Tool for QoS analysis.

Finally, the FL module is able to generate reports on the application performance, such as response time and throughput overtime as well as the estimated resource demand. An example report is available in D5.3.2.

(SOFTEAM) The FL module has been successfully exploited to improve the QoS and cost analysis of the Constellation case study.

Goal 7	Evaluate the Metric Explorer with respect to its functionality
---------------	--

The metric explorer is evaluated as scalable and supports data aggregation and visualization. WP5 partners and CSPs successfully tested the integration between Tower 4Clouds and the Metric Explorer. The Metric Explorer supports the visualization of the metrics it is attached observing. Graphite¹ is exploited in order to provide scalability and data aggregation feature.

Goal 8	Evaluate the Batch Engine with respect to its functionality
---------------	---

The goal is assessed as fully achieved.

The batch engine is built on top of HTCondor. The two following APIs were already provided at M24 according to the users or use case requirements: (i) the queue management API that manages job submission and execution and is able both to resuming and deleting jobs from the job queue; and (ii) the infrastructure management API that provides the means for extending the backend cluster size of the HTCondor deployment. Since M24, the batch engine APIs have been updated to fix minor bugs and for some little refactoring. The features have not changed and are considered as adequate.

Goal 9	Evaluate SLA tool Component (ATOS)
---------------	------------------------------------

According to the two levels of SLA identified in MODAClouds, the case study was interested in the Customer - Application Provider layer, and all SLOs belonging to this layer were defined. The SLOs specified in the contract follow the QoS requirements specified using Creator 4Clouds.

Goal 10	Evaluate LINE with respect to database replication (BOC)
----------------	--

(BOC) This goal has been evaluated by comparing the actual deployment, and thus actual runtime results, of BOC data on Amazon RDS with LINE model predictions. LINE provides sufficient support for models handling data replication and to measure mean relative difference in response times with respect to results from actual experiment runs.

(IC) The cloud database replication scenario is a more complex modelling scenario in that it involves dynamic changes in system resources, service demands and expected performance. Further, hosting the database on Amazon RDS and initiating replica creation during execution of queries requires more flexible modelling features within QoS tools. The LINE random environment feature is suitable for such scenarios in its ability to represent variability and dynamic changes in modelled scenarios.

¹ <http://graphite.readthedocs.org/en/latest/>

<p>When comparing the measured response times of experiments conducted on Amazon RDS and EC2 using BOC data with the results of the LINE models, the LINE predictions had an error of less than 10% at most. The random environments feature proved to be robust and able to model the variability experienced by queries during replica creation.</p>	
Deviations	
<p>Goal 9 has been added and evaluated within period M24-M36. Details of the Goal and evaluation are provided in Annex A</p> <p>(ATOS) Although compliant with WS-Agreement, the generated SLA agreement is not usable outside of MODAClouds</p>	
Recommendations	
<p>(SOFTEAM) Installation documentation should be improved and aggregated in one location. Connection errors to the WP2 cost estimating services should be shown when performing evaluations.</p> <p>To further investigate how to better integrate between QoS modelling and analysis tools and WP2 cost estimation services.</p>	
Evolution	
<p>The evaluation of the goals defined within WP5 by CSPs was delayed to M36. These goals have been evaluated and overall are considered as achieved.</p> <p>Because the Feedback loop module was released at M30, its evaluation (Goal 6) has been conducted within the M30-M36 period. Overall the evaluation shows that this goal is fully achieved.</p> <p>Goal 7 was postponed to M36 and has now been evaluated. Overall the evaluation shows that this goal is considered as achieved.</p> <p>Goal 8 was postponed to M36 and has now been evaluated. The evaluation shows that this goal is achieved. Also, it is noted that the definition of the SLOs is constrained by the actual metrics supported by Tower 4Clouds and that even if the generated agreements are compliant with WS-Agreement, they are not usable outside of MODAClouds.</p> <p>Goal 9 has been added and evaluated within the M24-M36 period. Overall the evaluation shows that this goal is considered as achieved.</p> <p>(BOC) Additional constraints have been considered specific to the case study. Manual transformation for such kind of constraints is necessary, which is acceptable from the case study's point of view. BOC also reports that the availability of the web application for the monitoring manager made experimenting with monitoring rules and observers easier than it was before when there was just a REST API available.</p> <p>(SOFTEAM) SOFTEAM extended their M24 experiments with the use of the Feedback loop component with helped in fixing the following recommendation made at M24 for SPACE^{Dev} 4Clouds: "how to get real resource demands and speeds to feed the model". Since M34, SOFTEAM has a working demonstrator that uses Tower 4Clouds and Feedback Loop to feed the model with values from the actual deployed application and then improving the obtained results.</p>	

2.2.3 Cloud Modelling Language and deployment engine

Who:	Smart City Urban Safety Planner (SIEMENS) MODELIO Project Management Server (SOFTEAM) BPM System (BOC) Health-care application (ATOS) WP4 (SOFTEAM, SINTEF)
Overall:	
<p>WP4 partners and all case study providers assessed CloudML with respect to two main goals: multi-cloud support and deployment of complex applications. The outcome of the evaluation shows the fulfilment of the two goals defined in the evaluation plan.</p>	
Goal 1	Evaluate MODAClouds IDE deployment engine with respect to multi-cloud support
<p>All partners reports successful deployment of their multi-cloud applications.</p> <p>(BOC) Creator 4Clouds and CloudML now fully supports the deployment of the case study application. The deployment was successfully modelled in Creator 4Clouds, exported, and deployed using the CloudML deployer. Deployment models are generic enough to model different deployment topologies. The BPM System requires deployment on Windows-based system and the application has been successfully deployed on Windows-based VMs. In addition, the BPM system has been successfully deployed using CloudML integrated with Puppet. However, some manual steps are still necessary to configure the Puppet master infrastructure. Deployment models are presented in D9.2.2 and D8.2.2.</p> <p>(SOFTEAM) All deployment models related to the Constellation server have been successfully enacted. CloudML support for multi-cloud deployment is considered as sufficient as far as this case study is concerned. In particular, the Constellation server has been successfully deployed on different combination of cloud provider including on multiple IaaS and on a combinations of IaaS and PaaS.</p> <p>(ATOS) The ATOS case study focuses on the deployment of multi-cloud application over PaaS services. Overall, CloudML provides sufficient modelling and deployment support as far as this case study is concerned.</p> <p>(SIEMENS) The Smart City Urban Safety Planner requires a complex infrastructure to be deployed in order to be properly executed. Nevertheless, it has been successfully deployed and modelled using CloudML including in a multi-cloud context (AWS EC2 and Flexiant). SIEMENS suggests the addition of new feature related to security aspects (see recommendation).</p> <p>At the WP level, SOFTEAM and SINTEF jointly evaluated CloudML's support for multi-cloud in three experiments, as follows:</p> <ol style="list-style-type: none"> 1) Same application deployed distributed on two IaaS, Flexiant and Amazon (e.g., <comp1> on Flexiant, <comp2, comp3> on Amazon) 2) Same application deployed on IaaS and PaaS, e.g., <comp1> on Amazon, <comp2, comp3> on CloudBees 3) Same application deployed on public and private cloud (e.g., <comp1> on Flexiant, <comp2> on SINTEF private mini cloud) <p>The results are:</p> <ol style="list-style-type: none"> 1) Successful deployment and execution for Constellation and SensApp 2) Successful deployment and execution for Constellation 	

<p>3) Successful deployment and execution for SensApp</p> <p>On top of this, SINTEF also evaluated:</p> <p>4) Migration of cloud application from one cloud IaaS provider to another IaaS cloud provider (e.g., migration of cloud application from Flexiant to Amazon)</p> <p>5) Migration of cloud application from one cloud PaaS provider to another PaaS cloud provider (e.g., migration of cloud application from CloudBees to Google App engine)</p> <p>6) Migration of cloud application from one cloud provider to another cloud provider showing that features not available on the first cloud provider are exploited at the second cloud provider applying the same CloudML model (i.e., demonstrating exploitation of peculiar features of a particular cloud)</p> <p>and reported successful deployment and execution for SensApp of 4, 6. And together with ATOS, the successful deployment and execution of 5 for the Health-care application.</p> <p>Finally, Polimi has evaluated and reported successful execution of 7. Details on the Evaluation of the Data migration and replication tool are reported in section 3.3.3.3.</p> <p>7) Management replication of data on different cloud providers at both IaaS and PaaS level</p>	
Goal 2	Evaluate MODAClouds IDE with respect to generated deployment models and scripts (SOFTEAM)
<p>(SOFTEAM) All deployment models related to the Constellation server have been generated using Creator 4Clouds and successfully enacted.</p> <p>All CSPs have been able to model the deployment of their multi-cloud applications using Creator 4Clouds and to successfully exploit the CloudML deployment engine to enact these deployments.</p>	
Deviations	
<p>At the PaaS level, CloudML offers support for Cloud Foundry instead of Heroku as initially planned. This is due to ATOS decision to deploy the Health-care case study on a private Cloud Foundry in combination with the Pivotal public PaaS.</p>	
Recommendations	
<p>(SIEMENS) Currently the security group should be manually created with the cloud provider functionality</p>	
Evolution	
<p>(BOC) At the time of the first evaluation, the Integration with Puppet was “work in progress” and thus could not be evaluated. This integration is now finalized and considered as successful. CloudML now fully supports the deployment of the case study application. The deployment was successfully modelled in Creator 4Clouds, exported, and deployed using the CloudML deployer.</p> <p>(SOFTEAM) One of the recommendations made after M24 evaluation was to provide mechanism to synchronize and control deployment actions happening in different cloud instances during deployment. CloudML offer now a full support of synchronizations mechanisms between cloud instances. All deployment process of our case study can now be performed automatically. In addition, integration of CloudML with the Models@Runtime engine provide now an API that allow to monitor and control the status of the deployment of Constellation.</p>	

(ATOS) The first evaluation focused on single cloud deployment. In addition, the support for configuration of various PaaS services in order to enable their proper interaction was postponed to M36. For the final evaluation, multi-cloud deployment have been successfully modelled and performed and the various PaaS services involved were properly configured. In particular, hybrid deployments have been performed involving ATOS private cloud running Cloud Foundry and Pivotal public cloud.

(SIEMENS) For the final evaluation the deployment models deployed by CloudML were fully generated using Creator 4Clouds. In addition, the deployment process was fully automated.

(SINTEF) The evaluation of [4-7] was postponed to M36. These operations have now been successfully performed.

2.3 Energizer 4Clouds

2.3.1 Tower 4Clouds

Who:	WP6 (POLIMI, IMPERIAL) Smart City Urban Safety Planner (SIEMENS) Health-care application (ATOS) BPM System Case Study (BOC) MODELIO Project Management Server (SOFTEAM)
Overall:	
Tower 4Clouds has been evaluated by WP6 partners and by all case study providers. Overall, the monitoring platform is considered as easy to install and to extend with new data collectors. In addition, the documentation is considered as adequate by CSPs. Finally, the evaluation shows the fulfilment of the goals related to the enactment and evaluation of monitoring rules as well as to the data collection and analysis.	
Goal 1	Evaluate the Monitoring Platform installation
Overall the installation of Tower 4Clouds is considered as simple and the procedure is easy to follow. Both at the WP and CSPs level the documentation is considered as adequate. (POLIMI, IMPERIAL, and ATOS) Being a library, the application level Data Collector (DC) can be deployed on PaaS. This has been successfully evaluated by ATOS. (SOFTEAM) SOFTEAM successfully developed and integrated two new components (Constellation Data Collector and Constellation Data Analyser) using the API provided by the platform in order to support the communication between Constellation components and Tower 4Clouds. (BOC) Overall, the installation of Tower4Clouds is easy. When the DCs are modelled as deployable artefact in the deployment model, they are successfully instantiated. Also the DDAs are properly configured. In addition, custom data collectors have been successfully integrated in the platform and the documentation for performing such development and integration is considered as sufficient.	
Goal 2	Evaluate the Monitoring Platform with respect to the monitoring rules
Overall, Tower 4Clouds is able to satisfy monitoring rules. Tower 4Clouds is able to monitor and	

<p>evaluated the data specified in the rule and to trigger the relevant call when the specified condition is verified. This includes calls to high level commands of the Models@Runtime engine such as the scaling and bursting commands.</p> <p>When installing an erroneous monitoring rule, an error message is thrown. Evaluation shows that these messages are explanatory enough to fix the rule accordingly.</p>	
Goal 3	Evaluate the Monitoring Platform with respect to the monitoring data collection
<p>This goal is considered as achieved at both WP and CSP levels. Once the data collector are properly deployed and configured, the data is successfully gathered and visualized. The set of data collector can be easily extended and data collectors using the data-collector-library are properly registered in the manager, their API properly reflect their capabilities. In addition, when the communication between the collector and the manager is broken for longer than the keep alive property, the collector and its resources are unregistered.</p> <p>(SOFTEAM) Tower 4Cloud has been evaluated and fully integrated to our case study. We can now monitor the activity of all of our application components.</p> <p>(ATOS) Tower 4Clouds has been evaluated and integrated to our case study. For PaaS level applications data collectors can successfully be integrated in the applications and the latter can thus be monitored.</p> <p>(SIEMENS) Case study components can be successfully monitored and the resulting data can be visualized. As far as this case study is concerned, this goal is considered as achieved.</p>	
Goal 4	Evaluate the Monitoring Platform with respect to the statistical data analysers
<p>This evaluation has not changed compared to M24.</p> <p>The extensive experiments ran for different scenarios surfaced good results for estimation, prediction and correlation. For example, the average error rate is below 20%. An experiment with an artificial peak load for predicting CPU utilization reports error rate below 5%. Test for CPU utilization on a web server with 20 users sending requests has below 15% error rate. Another test with user fluctuating behaviour shows that the error rate is below 20%. Another experiment, with an artificial peak load between 2 CPU utilizations on two different VMs, reports error rate below 5%. Test on two web servers with 20 users sending requests with round robin policy has below 10% error rate. Another test with user fluctuating behaviour shows that the error rate is below 15%.</p>	
Goal 5	Evaluate the Monitoring Platform overhead
<p>This evaluation has not changed compared to M24.</p> <p>In D6.3.1, we have shown that the CPU utilization for the Data Collector (DC) is lower than 5%. We have tested the DC to get the CPU utilization on a VM with 2 GB memory. The memory overhead is 3.4% utilization.</p>	
Deviations	
<p>During last year of the project, the architecture of Tower 4Clouds has dramatically evolved. As a result, some of the questions and metrics defined at the beginning of the project were not applicable anymore. In particular, questions related to Goal 3 were updated accordingly. The new metrics and questions are summarized in Annex C.</p>	
Recommendations	

(BOC) Monitoring rules could be deployed with CloudML 4Clouds instead of being installed from the IDE. Also, when installing the platform, init scripts should be provided for standard service handling.

Evolution

The architecture of Tower 4Clouds has dramatically changed during Y3. As a result, some of the evaluation questions and metrics were not relevant anymore. New questions and metrics have thus been defined and successfully evaluated.

(BOC) As stated in D9.2.2, compared to the status at the end of the second year of the MODAClouds project, the deployment of the Tower 4Clouds components has been significantly simplified with the introduction of the integrated runtime platform. This approach has replaced the Puppet-based installation of the monitoring platform components, which in turn reduces maintenance efforts. Furthermore, the availability of the web application for the monitoring manager made experimenting with monitoring rules and observers easier than it was before when there was just a REST API available.

(SOFTEAM) Tower 4Clouds has been fully integrated into the constellation server case study.

(ATOS) The evaluation of Tower 4Clouds was postponed to M36. The evaluation has now been performed and the goals are perceived as achieved.

2.3.2 Space^{Ops} 4Clouds

The evaluation of Space^{Ops} 4Clouds has been divided according to three sub-components: the Self-adaptation platform that includes the auto-scaling reasoner and the self-adaptation policies, the Models@Runtime engine and its support for cloud bursting, and the multi-cloud load balancing reasoner.

2.3.2.1 Auto-scaling reasoner

Who:	WP6 (POLIMI, IMPERIAL) MODELIO Project Management Server (SOFTEAM)
Overall:	Since the final version of the Self-adaptation platform has been released at M30, this tool has been extensively evaluated during the last year of the project at the WP level and using the Softeam case study. All three questions assessing the single goal defined for this artefact were considered to be accomplished. No deviations were observed.
Goal 1	Evaluate the Self-adaptation mechanism
	The system demonstrated to scale following the incoming workload. As for M24, Analyses have been performed on a real test bed for the OFbiz applications. Analysis of simulation results has shown how the self-adaptive reasoner is able to significantly reduce costs with respect to threshold-based policies implemented by cloud providers without introducing significant overhead and QoS violations. Thus, we can conclude that scaling policies are effective. Detailed results are available in Ardagna Ciavotta Lancellotti presented at SYNASC-MICAS 2014

The load-balancing reasoner demonstrated that it is capable of increasing the throughput and balancing the resource usage among different VMs in single and in multiple clouds scenarios (multi-cloud load balancer) and also that it is capable of maximizing the revenue for different classes of users. The results demonstrating these claims can be found in D6.5.3.

In addition, the validation experiments tested the capability of the MODAClouds autoscaling reasoner to react to workload fluctuations considering the SOFTEAM HTTPAgent component deployed on an Amazon m3.large VMs. The workload generator (jMeter) injected a request rate in [6, 32] req/s. An average response time QoS constraint equal to 560ms was set for the *readOnlyModelFragment* service and the autoscaling reasoner used a five step ahead control with a time period of 5 min. Workload prediction was obtained by using an ARIMA model, while service demand estimates were obtained through the ERPS SDA acting at 10s time scale. The autoscaling reasoner started up to 12 VM instances. The percentage violations of the average response time measured at runtime were 1.9% at 10s time scale and 0% at 5 min time scale.

Deviations	
No deviations from the plan were observed.	
Recommendations	
Evolution	
<p>The evaluation of the support offered by the tool for scaling policies was postponed to M36. This evaluation has been performed and is considered as successful. Detailed results have been published at the QUDOS 2015 workshop.</p> <p>Load balancing between different cloud providers was delivered at M30; the evaluation of this feature as well as the comparison with other reasoner has thus been evaluated within the M30-M36 period. A more detailed evaluation of the multi-cloud load balancing feature is provided in Annex B.</p>	

2.3.2.2 Multi-Cloud Load Balancing reasoner

Who:	WP6 (IMPERIAL)
Overall:	
<p>The MODAClouds multi-cloud load-balancing reasoner has been evaluated using BOC ADONIS data. The goals were assessed by measuring throughput and response time in experiments where the load balancing policy is set by naïve weights and using the weights recommended by MODAClouds load balancing reasoner. Details of the methodology and experimental results are in Annex B.</p>	
Goal 1	Is the multi-cloud load balancing reasoner effective in real-world scenarios?
<p>This goal has been evaluated by measuring and comparing the experimental results obtained with the multi-cloud load-balancing reasoner under different scenarios of the BOC ADONIS application.</p> <p>Based on the results of the case study, the throughput has been increased, when we applied the weights according to the recommendation from the MODAClouds load-balancing reasoner, by</p>	

more than 100% and we also observed an overall of 75% improvements in terms of the “response time over throughput” metric.	
Goal 2	Mean relative difference in key QoS metrics with respect to results from other load balancing policies, using a broad range of scenarios for the test application.
<p>The load-balancing reasoner set the weights according to the demand it measures for different classes of users at runtime. The benefits of this reasoner is that, comparing with naive weight setting that only consider the size of VMs, the reasoner tries to maximize the revenue for the end users.</p> <p>Based on our experimental results, the revenue of the servers in our multi-class setting has been increased by 27% in three different experiments conducted internally in a more controlled environment but with real cloud data.</p>	
Deviations	
No deviations were observed.	
Recommendations	
Evolution	
These goals, questions and metrics have been added to the evaluation plan during Y3 while the multi-cloud load balancer has been developed. These goals have been evaluated by Imperial and BOC and are considered as achieved.	

2.3.2.3 Models@Runtime engine and cloud bursting

Who:	WP6 (SINTEF) BPM System (BOC) MODELIO Project Management Server (SOFTEAM) Health-care application (ATOS)
Overall:	
This component has been evaluated at the WP level by SINTEF and by several CSPs. Overall, the results show that the two defined goals are considered as fulfilled and all questions are reported as successfully accomplished. No deviations from the plan are observed.	
Goal 1	Evaluate Models@Runtime with respect to its causal connection functionality on multi-cloud
SINTEF at the WP level and CSPs evaluated Models@Runtime engine with respect to the following scenarios:	
1) Dynamic adaptation of the deployment of a cloud application deployed on IaaS (e.g., allocating new	

<p>type of Virtual Machines, adding/removing VMs)</p> <p>2) Migration of a cloud application (or part of) from one cloud IaaS provider to another IaaS cloud provider (e.g., migration of cloud application from Flexiant to Amazon)</p> <p>3) Migration of a cloud application (or part of) from one cloud PaaS provider to another PaaS cloud provider (e.g., migration of cloud application from CloudBees to Beanstalk)</p> <p>The results are:</p> <ol style="list-style-type: none"> 1) Successful adaptation for SensApp and the BPMN System 2) Successful migration of SensApp and Constellation, from AWS EC2 to Openstack and from AWS EC2 to Flexiant 3) Successful migration of the Health-care application from Pivotal to a private Cloud Foundry instance. <p>Regarding the causal connection that keeps the runtime model synchronized with the running system, the status of the IaaS and PaaS services is monitored as well as the status of the software components during the deployment. The integration between the Models@Runtime engine and Tower 4Clouds can provide support for monitoring the status of the software components once deployed if desired.</p> <p>The Models@Runtime engine has been successfully integrated with Creator 4Clouds and CloudML models are successfully enacted.</p> <p>(SOFTEAM) The Models@Runtime engine provides now an API that allows us to monitor and control the deployment of Constellation.</p> <p>(BOC) The deployment model is successfully updated within the Models@Runtime engine.</p>	
Goal 2	Evaluating enactment of a specified bursting scenario for the Models@Runtime platform
<p>The scaling and bursting commands have been created within the M24-M30 period and successfully applied by the CSPs and to SensApp at both IaaS and PaaS levels.</p> <p>(BOC) As stated in D9.2.2, performing Cloud-to-Cloud migration has been successfully performed with the support of the Models@Runtime engine provided by MODAClouds. The fact that the Models@Runtime engine builds on the CloudML deployment technology already tested during the deployment use case was very beneficial. We could simply incrementally update the models and the deployments worked as expected.</p> <p>(ATOS) The Health-care application has been successfully burst from one PaaS to another and in particular from a private Cloud Foundry instance hosted on ATOS premises to Pivotal. All PaaS services were created and configured to properly interact with each other.</p>	
Deviations	
No deviations were observed.	
Recommendations	
(BOC) The documentation of the high level command and of the Web Socket interface could be enhanced.	
Evolution	
For the first evaluation the scaling and bursting feature had to be triggered by editing and providing a	

new deployment model to the Models@Runtime engine. Since M24, the scaling and bursting commands have been created, enabling the evaluation of these bursting and migration features by the CSPs (first evaluation was performed by SINTEF). The evaluation of these features has thus been completed and overall the goals are considered as fulfilled

Similarly, support for PaaS migration was added within the M24-M30 period and tested by ATOS on the Health-care application.

2.3.3 ADDapter 4Clouds

2.3.3.1 Models@Runtime Deployment engine

Who:	BPM System (BOC) MODELIO Project Management Server (SOFTEAM) Health-care Application (ATOS) Smart City Urban Safety Planner (SIEMENS) WP6 (SINTEF)
Overall:	
The deployment tools have been extensively used and evaluated by all CSPs. Overall, the evaluation shows that the defined goal is achieved. No deviations have been observed.	
Goal 1	Evaluate MODAClouds deployment tools
<p>(BOC) BPM System case study has been successfully deployed on Windows-based VMs using the deployment tool together with Puppet. At this stage, overall deployment support is considered as adequate based on BPM case study.</p> <p>(SOFTEAM) Constellation was successfully deployed and executed. All deployment scripts were generated using the IDE.</p> <p>(ATOS) Health-care application and its database were successfully deployed, configured and executed on multiple clouds at the PaaS level.</p> <p>(SIEMENS) The Smart City Urban Safety Planner and its database were successfully deployed and executed.</p>	
Deviations	
No deviations were observed.	
Recommendations	
(BOC) Some manual set up of Puppet is required in order to properly integrate the Models@Runtime engine and Puppet. Minimizing these manual interventions could be beneficial.	

Evolution	
<p>(BOC) First evaluation with the BPM system, focused on deployment and installation on a single Windows-based VM. The final evaluation considered the whole systems including several VMs. The deployment and provisioning involved both the Models@Runtime engine and Puppet. The goal is considered as achieved.</p> <p>(ATOS) First evaluation focused on a single cloud deployment. For this final evaluation, multi-cloud deployments were considered. The goal is considered as achieved.</p>	

2.3.3.2 Support services

Who:	WP6 (IEAT, ATOS) Smart City Urban Safety Planner (SIEMENS)
Overall:	
<p>The support services have been widely used by the other MODAClouds components. Overall, the evaluation shows that all goals are fulfilled without deviation.</p>	
Goal 1	Evaluate Load Balancing RESTful API
<p>The load balancing RESTful API is suitable for exposing the load-balancer to the self-adaptive reasoner; Haproxy API allows external clients to change the weight of each server with a REST call. In addition, RESTful API can accept security certificates for certain scenarios requiring server authentication.</p> <p>(SIEMENS) The load balancer and its API have been successfully used in a multi-cloud context.</p>	
Goal 2	Evaluate the Object Store module with respect to its functionality
<p>The evaluation of the Object store has not changed since M24. The metrics show that the goal is fulfilled.</p> <p>The object store uses an internal key-value database to store the data, plus additional indices, links and annotations. Thus, configuration parameters can be stored and retrieved easily from the Object Store. Storing and retrieving of state data is achieved using the same API as for configuration parameters.</p>	
Goal 3	Evaluate Artefact Repository with respect to its functionality
<p>Artefact Repository is able to store and retrieve artefacts. It can be deployed in a multi-cloud scenario and uses RSYNC type synchronization across multiple clouds and provide the necessary authorization and authentication mechanisms.</p>	
Goal 4	Evaluate SLA evaluation component
<p>The evaluation shows that the tools correctly evaluate the SLOs. The evaluation of SLOs is assessed by Tower 4Clouds while SLA Tool itself correctly assesses QoB rules.</p>	
Deviations	
<p>No deviations were observed.</p>	

Recommendations	
Evolution	Multi-cloud support has been added to the artefact repository and to the load balancer. Both have been evaluated. Similarly, authorization and authentication support for the artefact repository have been added and evaluated. Compared to M24, the artefact repository enable search of particular artefact.

2.3.3.3 Hegira 4Clouds

Who:	Smart City Urban Safety Planner (SIEMENS) BPM System (BOC) WP6 (POLIMI)
Overall:	Most of the goals for this tool were due and already considered as achieved at M24. The remaining goals are now also considered as fulfilled. The tool has been evaluated at both the WP level and by SIEMENS.
Goal 1	Evaluate data migration overhead
	The result of this evaluation has not changed since M24. The outcome of experiments performed on Windows Azure Tables and Google Datastore proved that the overall migration system overhead is acceptable.
Goal 2	Evaluate data migration and synchronization correctness
	The result of this evaluation has not changed since M24. The data was migrated correctly. In cases where data types of the source database are not supported by the target database, data is saved in a serialized form into the target database. See D.6.7 for details.
Goal 3	Evaluate scalability of data migration system
	The result of this evaluation has not changed since M24. The data migration system is considered scalable w.r.t. the number of entities to be migrated.
Goal 4	Evaluate lossless data migration
	The result of this evaluation has not changed since M24. Experiments performed have achieved 100% data migrated.
Goal 5	Evaluate migration system supported databases
	Hegira 4Couds has been tested against 4 different types of databases.

Deviations	
(BOC) Replication based on backup and restore functionality of MS SQL server is being used in the migration use case instead.	
Recommendations	
Evolution	
<p>Goals 1 to 4 were due in M24 and thus their evaluations have not changed. Regarding the number of supported databases, for this final evaluation it is reported that Hegira 4Clouds support 4 different types of databases.</p> <p>One of the recommendations from M24 was to extend the evaluation of the tool to other partners. It has been considered and Hegira 4Clouds has been evaluated in the context of the Smart City Urban Safety Planner case study.</p> <p>(SIEMENS) Within the context of the case study, the data has been successfully synchronized between an instance of HBase database and an instance of Cassandra database.</p>	

3 Summary of the evaluation of MODAClouds results from the case studies perspective

In this section all case studies presents a summary of their evaluation of the MODAClouds tools. First, the case study is introduced, then an overview of tools that have been evaluated by the case study together with a description on how they have been used is provided. Finally, a summary of the evaluation is presented.

3.1 Business Process Modelling System

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 behaviour 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 a 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 Gartner the total SaaS market increased by around 15% per year from more than \$14 billion in 2012 to \$22 billion through 2015. 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 target². Technology developed within the MODAClouds project plays an important role in achieving this step of business model extension.

² Alexander Gunka, Moving an Application to the Cloud – an Evolutionary Approach. MultiCloud'13.

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:cloud³.

3.1.1 Exploiting MODAClouds results

The ADOxx platform has a three tier architecture, including technologies such as JEE web containers, Web-Services implemented in C++ within a MS Windows service application and relational databases MS SQL Server and Oracle. Taking into account these technologies and the fact that BOC has already had some experience with hosting projects, the evaluation of MODAClouds methodology and tools focused on the support provided when moving a production application to the cloud.

MODAClouds tools evaluated by year

Venues 4Clouds	Creator 4Clouds	CloudML 4Clouds	Tower 4Clouds	Energizer 4Clouds	Space ^{dev} 4Clouds
Y2	Y2-Y3	Y2-Y3	Y2	Y3	Y2

The experiments conducted by BOC initially focused on using Venues 4Clouds and Space^{dev} 4Clouds to select a number of suitable cloud providers which can be used to host the SaaS solution taking into account costs and quality requirements like for example the need for the data to be located in a specific region. Using the MODAClouds decision support system for the *Provider Selection* use case, especially its risk assessment driven process extended the approach BOC would have followed for provider selection with valuable aspects. After that the case study showed in the *Provisioning & Deployment* use case how Creator4 Clouds and CloudML 4Clouds can be used together to model different deployment scenarios and to deploy to different cloud providers and demonstrated how this approach can be extended to integrate the widely-used configuration management tool Puppet. The fact that the already existing Puppet modules for the ADOxx application components could be used to deploy and configure those artefacts proved that CloudML 4Clouds integrates well in existing deployment toolchains. In their *Monitoring* use case BOC showed how Tower 4Clouds can be integrated with an existing monitoring solution based on Nagios and how Tower 4Clouds allowed them to benefit from the possibility to collect new application specific metrics simply by configuring rules for extracting the relevant information from the application log files and from its graphing solution for numerical time- series data. The integration with Nagios has shown that Tower 4Clouds provides the right means (APIs and documentation) to use it in environments that do not build up a complete monitoring solution from scratch. Finally, the *Cloud to Cloud Migration* use case showed how Energizer 4Clouds, in particular the Models@Runtime engine with its continuous deployment capabilities can be used to move an application from one cloud provider to another. This helps reducing the risks associated with vendor lock-in and allows to react to outages or changes in regulatory requirements and to easily replicate the deployment to other locations. In addition to the four use cases mentioned above, parts of the MODAClouds load balancing features have been evaluated in the scope of the *Static Load Balancing* use. Here the load balancing reasoner was used to optimize the load balancing configuration of multi-cloud deployment of the ADONIS:cloud 1-Click Try-Out on heterogeneous cloud platforms with different IaaS capabilities.

MODAClouds tools have been used to address the following needs:

- The *selection of cloud providers* should be simplified with the help of decision support tools and methodologies.
- *Deployment* of a given application stack to the selected clouds should be supported in an automated, cloud provider independent way.

³ BOC. (2014). BOC Group: ADONIS:cloud Landing page. <http://www.boc-group.com/at/adoniscloud>

- Advanced *monitoring* techniques should be used to track system health and quality of service.
- In case of detected violations or changed business needs another *cloud provider* should be *selected* and the application should be *re-deployed* to the new provider including migration of data and management of traffic routing.

3.1.2 Final evaluation summary

The usage of the MODAClouds methodology and components supporting these cases has provided good results and has helped in improving the components themselves to make them more suitable to be used in a professional context. Notable examples of such improvement are the extension of MODACloudsML to deploy and use a Puppet infrastructure to support the management of applications and the integration of Tower 4Clouds with the Nagios platform already in use at BOC. Some concerns with respect to ease-of-use, raised after the first evaluation period at month 24 have been very well addressed by the WP3 efforts in providing an integrated platform installer.

3.2 Smart City Urban Safety Planner

The objective of this case study is to build a system collecting and analyzing data from various sources (public utility sensors, emergency calls, ...) in order to validate and to react to a city emergency event (e.g., gas leak, fire incident) in near real time.

Obviously, such a system has specific requirements in terms of high availability (it should never happen that it is not available when a reaction is needed), fault tolerance and high responsiveness. To cope with this last requirement considering, at the same time, the possibility to manage fast processing of data arriving from the various information sources, Siemens has chosen to adopt Apache Storm as enabling technology and a NoSQL database to store data concerning emergency events and real time data such as air pollution status or citizen reported incidents. Moreover, in order to increase the availability of the system, Siemens has exploited a partially replicated approach with the main application components replicated into two clouds (i.e., Flexiant and Amazon).

3.2.1 Exploiting MODAClouds results

MODAClouds tools evaluated by year

Creator 4Clouds	CloudML 4Clouds	Tower 4Clouds	Energizer 4Clouds	Hegira 4Clouds
Y2	Y2	Y3	Y3	Y3

The adoption of the MODAClouds tools has helped the case study in reaching its objectives. In turn, this case study has challenged specifically the MODAClouds deployment language and environment, that is, CloudML, as well as the runtime platform, Energizer 4Clouds because of its highly distributed architecture and the usage of Storm as complex event processing technique. This has led to the identification of a specific pattern for the application configuration, the Leader and Follower pattern, and to an extensive experimentation of runtime mechanisms that support multi-cloud replicated architectures.

CloudML has been used in order to deploy the necessary infrastructure on multiple cloud providers such as Flexiant and Amazon. This multiple clouds approach provides high availability and low

latency for the critical services that demand such a requirement. The model used for this tool, was created using Creator 4Clouds in a graphical interface and exported in a JSON format.

In order to have a stable system, we have monitored it using Tower 4Cloud, which provided us with a clear image on the status of the used resources. In addition, we have exploited the MODAClouds load balancer controller in order to improve the availability of the web data collector component. In this way, we distribute the oncoming traffic between two servers hosting the same application content.

Regarding the data storage, SCUSP relies on two NoSQL databases: Cassandra and HBase. The data between these two databases has been synchronized using Hegira 4Clouds.

3.2.2 Final evaluation summary

Using CloudML 4Clouds proved to be helpful tool for deploying a complex distributed framework in a multi-cloud environment. On the open source community there are not so many tools that provide such flexibility. Also, Hegira4Clouds provides an abstraction layer that supports a general approach for data synchronization between two NoSQL column oriented databases (such as Cassandra and HBase). This feature let us the opportunity to avoid technology lock-in.

3.3 Health-care Application

The Atos e-Health solution is a software application that aims at developing an innovative and integrated solution for the general management of patients suffering from dementia and their caregivers. It provides an integrated online clinical, educational, and social network to support all of them. Based on a set of monitoring parameters and measuring scales, this solution aims to early detect symptoms that predict decline, avoid emergencies and secondary effects and, ultimately, prolong the period that patients can remain safely cared at home, no matter where it is located. There are various stakeholders involved in this scenario that would benefit from the system capabilities offered by the e-Health application:

- **Patients and caregivers:**
 - Access to services, like videos or games, recommended by clinicians or experts.
 - Collect and register data and measurements (blood pressure, weight, activity levels, questionnaires, etc.).
 - Management of warnings or requests sent to the clinicians.
 - Improve awareness on the use of their sensitive data, like the patient's monitoring parameters and the patient's medication follow-up and drug adverse events.
- **Clinicians and Health System** (organization of people, institutions and resources to deliver health care)
 - Continuous monitoring and follow-up of the patients
 - Management and assignment of tasks and questionnaires
 - Services to meet the health needs of target populations
 - Improve workload of assistance teams:
 - Institutions/specialists dynamically added and removed on demand
 - Allocation/De-allocation of cloud resources depending on the workload.
 - Rapid elasticity, i.e., the network can respond rapidly and automatically to changes in demand from particular doctor/specialist
 - Help monitoring of risks like: data breaches/inappropriate access, disruption of service and data)

This software solution consists of two main software blocks: a multi-cloud server-side block and a client-side block. The server-side block is composed of a database and two server applications. All of them can be deployed in different multi-cloud scenarios alternatives, like private clouds, public clouds or hybrid scenarios. The client-side block consists of a desktop application (used by the patients and their caregivers) that connects to one of the server applications.

The main requirements posed by this case to the adoption of the MODAClouds tools are the following:

- Explore different deployment alternatives in multi-cloud (PaaS) environments, with hybrid configuration where public and private cloud services coexist, in order to support so-called cloud bursting when the application load increases.
- Monitor the different application components deployed in different PaaS.
- Manage the governance of all the deployed components.
- Exploit 3rd party services to create added value for the application.
- Assess the benefit of the Cloud in terms of flexibility and agility offered by resource scaling and application migration.

3.3.1 Exploiting MODAClouds results

MODAClouds tools evaluated by year

Venues4Clouds	Creator 4Clouds	CloudML 4Clouds	Tower 4Clouds	Energizer 4Clouds	SLA tool
Y3	Y2-Y3	Y2-Y3	Y2-Y3	Y3	Y3

ATOS exploited Venues4Clouds in order to get a list of the public PaaS providers that best matched the requirements of the case study. The experimentation with Creator4Clouds has allowed starting from a high level model of the application to define QoS constraints associated to the services offered by the interface components and to derive, in a semi-automatic way, a deployable model of the application, a set of corresponding monitoring rules and SLA agreements. This tool also offered us the capability of modelling different cloud architecture deployment alternatives for the same application.

After the deployment, the usage of Tower 4Clouds, SLA tool and Models@Runtime has allowed us to move the web GUI application dedicated to clinicians from a private to a public cloud (the PaaS provider with the best score from Venues4Clouds) and thus to demonstrate that such automatic cloud bursting approach is feasible and can be triggered when SLA agreements are violated.

3.3.2 Final evaluation summary

A multi-cloud approach offers several new possibilities and advantages for telemedicine applications, such as the Atos e-Health solution, but at the same time it also presents some risks and disadvantages. By using the MODAClouds ecosystem, we have been able to benefit from the advantages of such approach, and at the same time, we could avoid most of these risks and disadvantages associated to a multi-cloud deployment.

We could design, model, deploy and do the governance of the different e-Health application components in a multi-cloud environment as expected. The design-time tools also allowed us to define the QoS constraints and SLAs needed to define the MODAClouds auto-adaptation capabilities. At the end, they allowed us to test a cloud bursting scenario, where one application was migrated to a public cloud after being stressed. In general, we are very satisfied with the results obtained after using MODAClouds.

3.4 MODELIO Project Management Server

Modelio⁴ is a desktop based modelling tool available in commercial and open source versions. However, when we started the MODAClouds project, Softeam decided to extend Modelio features and architecture to enable sharing of models among multiple stakeholders and to provide other core functionalities, like version control and access policies. The result of this work is a cloud-based extension of Modelio called *Constellation*, which has been recently delivered as a service.

Constellation is an advanced repository that stores the models defined using the Modelio case tool. Models developed by users are decomposed into model fragments that are managed by an Administration Server. The concerns targeted by Constellation are not only related to project/fragment portfolio management, but also to governance issues. This is where the Constellation functionalities come in: organizing and managing collaborative (distributed) World Wide Modeling projects, both with regard to the content of these projects and to the management of the teams and individuals who work on them (rights, viewpoints on the project, and so on.).

In a traditional setup, users would need to provision their own infrastructure to support and secure the SVN server. The main drawback of this setting is that customers would need to buy and maintain the hardware necessary to run the server and maintain and update the servers by themselves. With the development of Constellation, Softeam extended its services offers with the ability to provide modelling features as a service. The technical architecture is based on an Administration Server component and on an agent-based architecture. Agents are opportunistically deployed in the cloud in order to offer either one shot modelling services or long term read only or read-write model storage.

During the process of migrating from a desktop-based to a cloud-based paradigm, the following requirements were to be considered:

- The adoption of the cloud by the company should be as seamless as possible. The Modelio team is composed by young and determined people that, however, have never been trained to work in a cloud context.
- The company should be able to study the feasibility of selecting different cloud services depending on various variables, both related to QoS requirements and also to the specific needs of customers.
- The cost of the adoption of the cloud should be kept under control in order to ensure that the company will achieve the return on investment in a relatively short time.
- Softeam wants to be able to set specific QoS constraints on the components of the Constellation architecture and wants to exploit the elasticity offered by the Cloud environment to cope with the changes in the number of users and in the workload they produce.

3.4.1 Exploiting MODAClouds results

MODAClouds tools evaluated by year

Venues 4Clouds	Creator 4Clouds	CloudML 4Clouds	Tower 4Clouds	Energizer 4Clouds	Space ^{dev} 4Clouds	Space ^{ops} 4Clouds
Y3	Y2	Y2	Y3	Y3	Y2	Y3

At design-time we followed the typical MODAClouds workflow, by using Creator4Clouds to define the architecture of our case study demonstrator, Venues 4Clouds to weight the differences between

⁴ <https://www.modelio.org/>

cloud providers as well as to analyse our non-functional requirements to determine the cloud providers that better correspond to our needs.

Concerning SPACE^{Dev} 4Clouds, we focused on the optimization features provided along with it, which provided us with an estimation of the number and size of VMs needed to deploy and run our services. Finally, we used Creator 4Clouds' integration with CloudML to deploy the application to the runtime platform, and to configure the Models@runtime engine, Tower4Clouds and the Feedback loop components automatically.

We exploited the new version of Constellation to test four runtime components: the Models@Runtime engine, SPACE^{Ops} 4Clouds, Tower 4Clouds and the Feedback loop. The Models@runtime engine allowed us to achieve advanced deployment automation, enabling not only to deploy Constellation but also the runtime adaptation of the deployed application. SPACE^{Ops} 4Clouds has successfully complemented the Models@Runtime engine by automatically triggering the adaptation of the deployment. We used this component to test the ability to adapt the deployment to usage peaks using its cloud bursting capabilities.

In addition, we exploited the Feedback loop component in order to improve our design models (specifically, the usage model and resource demand information needed for SPACE^{Dev} 4Clouds analysis) with data from real deployments. Finally, the Tower 4Clouds component was used to provide data to the other listed runtime components.

3.4.2 Final evaluation summary

The main benefit we get from the use of MODAClouds is having a single model that can be easily adapted to many different cloud providers. It also allow our engineers in defining quality targets and monitoring them live in a cloud transparent way. In fact, the whole application life cycle can now be controlled in a cloud independent way.

Tool	Specific evaluation
Venues 4Clouds	Easy to use, since the software does not need any installation. It effectively allows us to reduce time to select the most suitable cloud provider offer.
Creator 4Clouds	Our multi-cloud deployment scenarios showed that cloud migration costs much less now. "Model once, deploy to multi-clouds "
CloudML 4Clouds	No cost to adapt application for several providers. It allowed us to explore a large set of deployment configuration.
Tower 4Clouds	Integration to our application was easy, and independent from cloud provider. Easy to integrate and highly configurable
Energizer 4Clouds	Allow us to automate the deployment of the Constellation server on cloud environment and monitor the deployment process Very useful to observe the current state of the deployment and <i>interact</i> with it.
Space ^{dev} 4Clouds	Automated fine analysis of costs before deployment. Allowed us to choose initial deployment architecture.
Space ^{ops} 4Clouds	The main innovation is in the auto adaptation of deployment based on actual resources consumption and workload forecast. Cloud bursting was successfully tested in lab conditions.

Feedback Loop	Allow us to collect data from Runtime and exploit it to improve design accuracy Increased accuracy of analysis model
---------------	---

4 MODAClouds objectives evaluation

This chapter evaluates the MODAClouds main objective, its sub-objectives as they were defined in DoW [1]. The following sub-sections of the chapter present the results of this evaluation at M36 using a table-based approach as defined in D3.6 [2]. In particular, these results are an evolution of the tables initiated in D3.7.1 at M24. Each table has four columns:

- ‘Is it delivered?’ – contains a very short description of the objective
- M24 – the evaluation at M24
- M36 – the evaluation at M36
- Summary of collected data and comments – comments backing up the evaluation in column M36.

At each milestone, an objective is evaluated as Yes, No or To Some Extent (TSE), where

- Yes - means the objective is delivered as one or more features of MODAClouds solution, which doesn’t necessarily means that it is satisfactory for all its users; Summary/comments cell explains whether it needs further improvements or not;
- No – means the objective has not yet been delivered;
- To Some Extent – means only partially addressed in the reporting period.

4.1 MODAClouds main objective

The main objective of MODACLOUDS as stated in the DoW is to *deliver methods, a decision support system (DDS) and an open source IDE and run-time environment for the high-level design, early prototyping, semiautomatic code generation, and automatic deployment of applications on multi-Clouds with guaranteed QoS.*

The following table presents the main MODAClouds goal decomposed into seven sub-goals, one for the methods, one for the decision support system and five addressing the open-source IDE and runtime platform, and their evaluations at M36:

Is it Delivered?	M24 Yes/No/TSE	M36 Yes/No/TSE	Summary of collected data and comments
Methods	Yes	Yes	The MODAClouds methods are mainly offered by the design-time tools (i.e., Creator 4Clouds, Space ^{Dev} 4Clouds, CloudML). More specifically, the evaluation of the functional modelling tool has highlighted that the model-driven approach offered by the project is supported.
Decision Support System	Yes	Yes	Venues 4Clouds has been evaluated internally by WP2 and by some of the CSPs. Overall, the evaluation shows that it provide sufficient support for users to (1) express their requirements and processes them via risk-based analysis, (2) ease their selection of relevant sets of cloud providers.
Open Source IDE and run time environment for:			
- high level design	Yes	Yes	Evaluation results at both M24 and M36

			give an indication that the MODAClouds toolset supports high level design and analysis of multi-cloud applications.
- early prototyping	TSE	Yes	Several of the MODAClouds tools have been built with the idea to reduce the gap of Development and Operations teams and in particular the Feedback loop and the Models@Runtime tools. Their evaluation shows that they facilitate the exchange of useful information between design and runtime activities.
- semi-automatic code generation	Yes	Yes	Since M24, Creator 4Clouds supports the semi-automatic generation and refinement of models, in particular between the various levels of abstraction. Several of the generated models are fed automatically into the runtime platform (e.g., Monitoring rules, Deployment models)
- automatic deployment on multi-clouds	Yes	Yes	Multi-cloud deployment, scaling and migration experiments were successfully performed.
- Guaranteed QoS	TSE	Yes	MODAClouds guarantees QoS through: Space ^{Dev} 4Clouds for developers by offering evaluation and optimization mechanisms, Space ^{Ops} 4Clouds for operators via self-adaptation mechanisms and the SLA framework. All these components have been evaluated as achieving their goals.

4.2 MODAClouds sub-objectives

As for the overall MODAClouds objectives, this section is an evolution of the similar section presented in D3.7.1 at M24.

The evaluation of the more detailed sub-objectives (as stated in the DoW) is performed similarly as for the main MODAClouds objective and is presented in the following. The sub objectives are naturally overlapping with the main objective, thus, in this evaluation we emphasise only the additional or the more detailed aspects that are specified by the particular sub objective. Again the data needed to judge and reply to the questions are collected in the more detailed evaluation tasks related to the case study implementation and execution as well as the specific evaluation of WP.

The first sub-objective as described in the DoW is: *Model-Driven Development for Clouds and Multi-Clouds. The MODACLOUDS integrated development environment will feature advanced engineering MDD design-time methodologies and tools enabling the high-level design of Future Internet service-based applications, which will be semi-automatically translated into code able to run on multi-Cloud platforms. The code will be automatically deployable on multiple Cloud providers hiding the proprietary technology stack. Target environments for the MODACLOUDS framework will cover IaaS, PaaS and SaaS solutions spanning across all abstraction layers, supporting both public and hybrid Clouds. The framework will also support the migration of legacy applications to the Cloud.*

For this sub-objective the additional or more detailed aspects are coverage of the IaaS, PaaS and SaaS, public and hybrid clouds, and migration of legacy applications. All of these aspects relates to features of the MODAClouds IDE.

Is it Delivered?	M24 Yes/No/TSE	M36 Yes/No/TSE	Summary of collected data and comments
Support/Coverage of IaaS	Yes	Yes	Currently we support Amazon, Flexiant, CloudSigma, Openstack, Azure plus all IaaS supported by jCloud.
Support/Coverage of PaaS	TSE	Yes	Currently, CloudML and the Models@Runtime engine support Pivotal, Cloud Foundry, CloudBees, AWS RDS, AWS SQS, AWS Beanstalk. In addition, the CPIM library offers support for Azure, Google App Engine.
Support/Coverage of SaaS	TSE	Yes	Currently, we support some specific SaaS such as the database services offered by Azure and Google App Engine.
Support/Coverage of public and hybrid clouds	TSE	Yes	Several CSPs and WPs have evaluated the coverage of public and hybrid clouds. Hybrid clouds have been tested at both IaaS and PaaS levels. At the PaaS level, ATOS tested deployment involving their private Cloud Foundry instance together with the Pivotal public cloud. At the IaaS level, SINTEF and IeAT have performed evaluations involving their private clouds in combination with several public clouds. Moreover, several case study applications (e.g., BOC, SOFTEAM) have been exploiting the Flexiant testbed in combination with public clouds (e.g., CloudSigma, Amazon)
Support for migration of legacy applications to the Cloud	Yes	Yes	The BOC case study successfully migrated to Cloud environment.

The second sub-objective as described in the DoW is: *Multi-Cloud Economics. Developing applications for multi-Clouds may impact established enterprise procedures and business models. Metrics are needed to quantify the notion of risk for a particular choice relatively to the ecosystem in which it will evolve. However, decision models are hard to develop due to variability in Cloud resource prices across time (e.g., Amazon spot instances), geographic location, performance, legal aspects, etc. MODACLOUDS will develop decision support systems, risk analysis methods, provide proper guidelines, and identify new business models suitable for Cloud providers to address the needs of application providers and improve their trust in Clouds.*

For this sub-objective the additional or more detailed aspects are delivery of risk analysis methods according guidelines for the decision support system, and new business models. These aspects relate to features of the Decision support system.

Is it Delivered?	M24 Yes/No/TSE	M36 Yes/No/TSE	Summary of collected data and comments
Risk analysis methods	Yes	Yes	Venues 4Clouds integrate risk analysis

			methods that have been evaluated by CSPs. Overall this evaluation indicates that these methods have been successfully delivered.
Guidelines for risk analysis and decision support	Yes	Yes	This feature has been delivered at M24 and successfully evaluated at M36 by CSPs and at the WP level.
New business models	Yes	Yes	This feature has been delivered at M24 and successfully evaluated at M36 by CSPs and at the WP level.

The third sub-objective as described in the DoW is: *Quality-Driven Cloud Development. The MODACLOUDS integrated development environment will support the early analysis and reasoning on non-functional requirements and quality aspects of the final applications, and will optimise the matching between the target Cloud environments and application characteristics.*

For this sub-objective the additional or more detailed aspects are support for early analysis and reasoning of non-functional and quality aspects and matching of application cloud environment with application characteristics

Is it Delivered?	M24 Yes/No/TSE	M36 Yes/No/TSE	Summary of collected data and comments
Support for early analysis reasoning of non-functional and quality aspects	Yes	Yes	This feature has been successfully evaluated in close collaboration with the CSPs as part of Space ^{Dev} 4Clouds.
Support for optimized matching of cloud environment with application characteristics	Yes	Yes	This feature has been successfully evaluated in close collaboration with the CSPs as part of Space ^{Dev} 4Clouds.

The fourth sub-objective as described in the DoW is: *Run-Time Quality Monitoring and Assurance. Run-time techniques, independent from the Cloud providers management API, will be developed in order to afford switching the application, or part of it, from a Cloud provider to the other, providing performance and availability guarantees, and minimising application execution costs according to the run-time Cloud systems performance, failures, and resource prices. Data and application replication on multiple providers will be explicitly addressed in order to guarantee high availability and business continuity.*

For this sub-objective the additional or more detailed aspects are adaptation and migration of cloud applications according to performance, availability and execution cost and data and application replication on multiple providers.

Is it Delivered?	M24 Yes/No/TSE	M36 Yes/No/TSE	Summary of collected data and comments
Support for adaptation and migration of cloud applications according to performance, availability and execution cost	TSE	Yes	The final version of this feature has been released at M30 and has been evaluated as delivered in collaboration between the technical partners and the CSPs.
Support for data and application replication on multiple providers.	TSE	Yes	This feature is supported in terms of ability to migrate and synchronize data from one cloud to another and has been

			evaluated using a large dataset extracted from Twitter public data and using the Smart City Urban Safety Planner case study.
--	--	--	--

The fifth sub-objective as described in the DoW is: *Rapid software Evolution. A closed-loop between the run-time and design-time environments will be implemented in order to trigger the dynamic re-deployment of the final application or of its components to react to long-term failures of the Cloud providers or to exploit Cloud additional services or improved performance (e.g., new virtual machine instances or reduced prices), providing adaptation to changing contexts and requirements.*

For this sub-objective the additional or more detailed aspects are a closed loop between run time and design time to trigger adaptation according to dynamic contexts and user requirements.

Is it Delivered?	M24 Yes/No/TSE	M36 Yes/No/TSE	Summary of collected data and comments
A closed loop between run-time and design-time to trigger adaptation according to dynamic contexts and user requirements	TSE	Yes	This feature has been delivered at M30 and has been successfully evaluated in close collaboration with CSPs.

5 MODAClouds KPI evaluation

The MODAClouds KPIs and the methodology to measure them is already described in the DoW [1], thus, we will follow the evaluation plan as outlined there. The objective summary, operational goal and measurable success criteria and time of achievement for the KPIs as specified in the DoW [1] are presented below.

Objective summary	Operational goal	Measurable success criteria and time of achievement
SO1: Deliver an advanced software engineering model-based approach and an IDE to support systems developers in building and deploying applications, with related data, to multi-Clouds spanning across the full Cloud stack (IaaS/PaaS/SaaS).	<ol style="list-style-type: none"> 1. Develop and validate tools to support the design of application on multiple-Clouds. 2. Provide model-to-model transformations to address functional and non-functional concerns in application design. 	<ol style="list-style-type: none"> 1. The tool set must be perceived as usable in the industry cases as verified by performing a systematic study (e.g., using the Technology Acceptance Model (Davis 1989, Mohaghedhi 2012), M30). 2. Number of supported IaaS ≥ 3 (≥ 1 at M18; ≥ 3 at M30). 3. Number of supported PaaS ≥ 2 (≥ 1 at M18; ≥ 2 at M30). 4. Documentation on MODAClouds patterns and transformations must be perceived as complete (M30). 5. Number of Cloud design patterns ≥ 5 (≥ 2 at M18; ≥ 5 at M30).
SO2: Define quality measures, monitoring mechanisms, prediction models, and adaptive policies to provide quality assurance in Clouds and multi-Clouds.	<ol style="list-style-type: none"> 1. Define and formalise the QoS metrics and the corresponding models for their evaluation (both at design and run-time). 2. Implement automatic and provider independent deployment solutions and monitoring interfaces. 3. Implement run-time management policies to guarantee QoS constraints. 	<ol style="list-style-type: none"> 1. Definition of reference QoS metrics and their monitoring methods ≥ 3 (M12). 2. Median quality prediction accuracy (evaluated in terms of the mean value of the metrics) at design time (30%, M24). 3. Median quality prediction accuracy (evaluated in terms of the mean value of the metrics) at run-time (30%, M24). 4. Number of Cloud providers supported by the deployment and monitoring solutions ≥ 5 (≥ 2 at M12; 5 at M24). 5. Percentage of time that the QoS constraints are violated at run-time 5 (M30).
SO3: Provide support to costs and risks assessment to increase trust in Clouds.	Analyse several Cloud business models in order to define the criteria for reducing risks for Cloud migration and for reducing the costs of taking informed decisions.	<ol style="list-style-type: none"> 1. All the business models of Cloud providers identified in SO1 are analysed (M24). 2. The Decision Support System should include all Cloud parameters identified by case studies (M30).
SO4: Develop an integration framework between design tools and run-time.	Define and formalise the implementation of the interfaces among the software components	The integration test should be successful including all tools and requirements (M36).

	developed for the MODAClouds project.	
SO5: Create relevant and complex case studies for the entire risks assessment and software engineering methodologies based on practical industrial scenarios.	Develop validated and working applications for the four case studies.	Each tool should be evaluated by at least 50% of industrial case studies (M36).
SO6: Analyse and validate project outcomes through case studies.	<ol style="list-style-type: none"> 1. Show the effectiveness of the tool-set and of the run-time environment for the case studies. 2. Validate risks and assessment of costs for the case studies. 	<p>The requirements identified by the case studies and concerning the MODAClouds solution should be fulfilled completely at the end of the project (M36). As for low-priority requirements, a non-complete coverage will be tolerated, but it will have to be of at least 80% over the whole set of low-priority ones.</p> <ol style="list-style-type: none"> 2. The tool-set is perceived as effective by applying a systematic analysis (M12). 3. At least 1 white paper describing the general approach and 2 white papers on domain-specific guidelines for applying MODAClouds (M6, M18, and M36).
SO7: Ensure distribution of project results via dissemination activities on relevant publication channels, training, and standardisation.	<ol style="list-style-type: none"> 1. Contribute the results of MODAClouds to standards, forums and discussion groups. 2. Publish the results both on- and off-line to reach large audience. 	<ol style="list-style-type: none"> 1. At least 1 contribution to the standardisation bodies (M36). 2. The number of satellite workshops organised at international conferences ≥ 3 (=1 at M12; ≥ 2 at M24; ≥ 3 at M36). 3. The number of articles in scientific and /or business publications 20 (at least 15 at IEEE/ACM conferences and 5 in top ranked journals, M36). 4. At least 15 presentations in conferences and workshops (M36). 5. Participation in at least one collaboration working group (M18). 6. Organising at least 3 industrial seminars (=1 at M24; ≥ 3 at M36). 7. Number of hits on the MODAClouds public Web site 20,000 (M36).
SO8: Provide community based open source solutions supporting the full applications life-cycle.	<ol style="list-style-type: none"> 1. Provide MODACLOUDS software solutions as open source. 2. Meta-model extensions available as open source. 	<ol style="list-style-type: none"> 1. At least 2 software packages released as open source (M36). 2. Number of communities on the topics related to MODAClouds (M36). 3. All meta-model extensions are released as open models (e.g., Cloud abstractions, risk and quality assessment M36).

The table below summarizes the evaluation of metrics at M36:

KPI Definition / Remarks			Actual value
SO1: The tool set must be perceived as usable in the industry cases as verified by performing a systematic study (e.g., using the Technology Acceptance Model (Davis 1989, Mohagheghi 2012), M30).			Yes
SO1: Number of supported IaaS ≥ 3 (≥ 1 at M18; ≥ 3 at M30)			≥ 5
Amazon EC2, Flexiant, CloudSigma, Openstack, Azure plus all IaaS supported by jCloud.			
SO1: Number of supported PaaS ≥ 2 (≥ 1 at M18; ≥ 2 at M30)			4
CloudML and the Models@Runtime engine support Cloud Foundry (public and private instances), AWS RDS, AWS SQS, AWS Beanstalk. In addition, the CPIM library offers support for Azure, Google App Engine.			
SO1: Documentation on MODAClouds patterns and transformations must be perceived as complete (M30).			Yes
As shown by the final evaluation, all CSPs and WP4 partners have been able to successfully use and exploits the MODAClouds transformations and patterns through Creator 4Clouds.			
SO1: Number of Cloud design patterns ≥ 5 (≥ 2 at M18; ≥ 5 at M30)			7
4 cloud patterns extended to the multi-cloud context: Provider adapter, Runtime Reconfiguration, External Configuration Store, Leader-Followers; and 3 patterns presented as MODAClouds guidances.			
SO2: Definition of reference QoS metrics and their monitoring methods ≥ 3 (M12).			≥ 3
Supported metrics are: CPU utilization, Throughput, Response time.			
SO2: Median quality prediction accuracy (evaluated in terms of the mean value of the metrics) at design time (30%, M24).			$\sim 0.75\%$
Metric	Median Error	Application	Experiment
CPU Utilization	0.75%	OFBiz	Comparison between LINE and LQNS based on the model OFBizLB model, which has 2 user types and assumes deployment of application and DB server on the same VM. 9 different number of servers are considering, covering utilizations between 10% and 99%
Throughput	0.76%	OFBiz	
SO2: Median quality prediction accuracy (evaluated in terms of the mean value of the metrics) at run-time (30%, M24).			Between 9.19% and 18.40%
Metric	Median	Application	Experiment

	Error		
CPU Utilization	9.19%	OFBiz	Comparison between predicted CPU utilization from SDA and real runtime values. CPU utilization is taken from OFBiz server with work load generator OFBench periodically sending requests.
	18.40%	OFBiz	Comparison between predicted CPU utilization from SDA and real runtime values. CPU utilization is taken from OFBiz server with workload generator OFBench sending fluctuating requests.
Throughput	16.35%	OFBiz	Comparison scenario is using MODAClouds Load Balancer to maximize system throughput across 4 VMs with 4 OFBiz deployed on them. The workload generator OFBench sends 2 classes of different user sessions. The measured runtime throughput is compared with the theoretical result from the algorithm implemented in MODAClouds Load Balancer.
	16.76%	OFBiz	Comparison scenario is using MODAClouds Load Balancer to maximize system throughput across 4 VMs with 4 OFBiz deployed on them. The workload generator OFBench sends 4 different 4 classes of user sessions. The measured runtime throughput is compared with the theoretical result from the algorithm implemented in MODAClouds Load Balancer.
Demands	10.50%	OFBiz	Comparison of the demands estimated with the FMLPS and ERPS methods, against the ones obtained with the CI method. Covers the cases with 4 and 8 request classes.
SO2: Number of Cloud providers supported by the deployment and monitoring solutions ≥ 5 (≥ 2 at M12; 5 at M24).			7
<p>Monitoring platform was tested on: Flexiant, Amazon EC2, Microsoft Azure, Heroku, OpenNebula, OpenStack, Eucalyptus.</p> <p>The Models@Runtime engine was tested against: EC2, Flexiant, OpenStack, CloudSigma, Azure, Pivotal, Cloud Foundry, CloudBees, AWS RDS, AWS SQS, AWS Beanstalk</p>			
SO2: Percentage of time that the QoS constraints are violated at run-time (M30).			$\leq 5\%$ (M36).
<p>The validation experiments tested the capability of the MODAClouds autoscaling reasoner to react to workload fluctuations considering the SOFTEAM HTTPAgent component deployed on an Amazon m3.large VMs. The workload generator (jMeter) injected a request rate in [6, 32] req/s. An average response time QoS constraint equal to 560ms was set for the <i>readOnlyModelFragment</i> service and the autoscaling resoner used a five step ahead control with a time period of 5 min. Workload prediction was obtained by using an ARIMA model, while service demand estimates were obtained through the ERPS SDA acting at 10s time scale. The autoscaling reasoner started up to 12 VM instances. The percentage violations of the average response time measured at runtime were 1.9% at 10s time scale and 0% at 5 min time scale.</p>			
SO3: All the business models of Cloud providers identified in SO1 are analysed (M24).			Yes

SO3: The Decision Support System should include all Cloud parameters identified by case studies (M30).					Yes
SO4: The integration test should be successful including all tools and requirements (M36).					Yes
Integration and integration tests are described in D3.4.2					
SO5: Each tool should be evaluated by at least 50% of industrial case studies (M36).					Yes
Tool	ATOS	BOC	Siemens	Softeam	
Venues 4Clouds	X	X		X	
Creator 4Clouds	X	X	X	X	
MODAClouds IDE	X	X	X	X	
Space ^{Dev} 4Clouds		X		X	
CloudML	X	X	X	X	
Energizer 4Clouds	X	X	X	X	
Tower 4Clouds	X	X	X	X	
ADDapter 4Clouds	X	X	X	X	
Space ^{Ops} 4Clouds	X	X		X	
SO6: The requirements identified by the case studies and concerning the MODAClouds solution should be fulfilled completely at the end of the project (M36). As for low-priority requirements, a non-complete coverage will be tolerated, but it will have to be of at least 80% over the whole set of low-priority ones.					Yes
SO6: The tool-set is perceived as effective by applying a systematic analysis (M12)					Yes
SO6: At least 1 white paper describing the general approach and 2 white papers on domain-specific guidelines for applying MODAClouds (M6, M18, and					3

M36).	
2 white paper describing general approach (paper at MISE and MICAS) 1 white paper on domain-specific guidelines http://www.modaclouds.eu/publications/white-paper/	
SO7: At least 1 contribution to the standardisation bodies (M36).	Yes
Contributions to TOSCA, NIST, OMG	
SO7: The number of satellite workshops organised at international conferences ≥ 3 (=1 at M12; ≥ 2 at M24; ≥ 3 at M36)	4
MultiCloud April 2013 / Prague, MICAS September 2013 / Timisoara and MICAS September 2014 / Timisoara, QUDOS September 2015	
SO7: At least 15 presentations in conferences and workshops (M36).	104
SO7: The number of articles in scientific and /or business publications 20 (at least 15 at IEEE/ACM conferences and 5 in top ranked journals, M36).	75
Peer reviewed articles – conferences: 63 conference and workshop papers including 17 IEEE/ACM conferences. Peer reviewed articles – journals top ranked: 12	
SO7: Participation in at least one collaboration working group (M18).	3
1. CloudML collaboration group led by SINTEF (involving PaaSage, Artist and MODAClouds projects) 2. CloudML and deployment robustness collaboration group (involving MODAClouds and Diversify) 3. TOSCA standardization working group	
SO7: Organising at least 3 industrial seminars (=1 at M24; ≥ 3 at M36).	14
SO7: Number of hits on the MODAClouds public Web site 20,000 (M36).	177,725
SO8: At least 2 software packages released as open source (M36).	Yes
48 repositories 2000+ commits (8 main ones)	

SO8: Number of communities on the topics related to MODAClouds ≥ 1 (M36).	Yes
Contributions to TOSCA, Modelio, CloudML	
SO8: All meta-model extensions are released as open models (e.g., Cloud abstractions, risk and quality assessment M36).	Yes
https://github.com/SINTEF-9012/modaclouds-metamodels	

6 Conclusions

This deliverable presented the final result of the evaluation of the MODAClouds software solution on the basis of the evaluation plan defined in D3.6. Overall, the evaluation involved all MODAClouds partners and the results shows that all goals are considered as fulfilled.

In addition, this deliverable reported the result of the evaluation of the MODAClouds solutions with respect to the project's objectives and KPIs defined in the DoW. All the objectives are considered as achieved.

Bibliography

1. MODAClouds Consortium. (2012). MODAClouds Description of Work (DoW)
2. Solberg A. et al. (2014). D3.6 MODAClouds Evaluation Plan
3. Van Solingen, Rini, Egon Berghout. (1999). The Goal/Question/Metric Method. McGraw-Hill, Education. ISBN 0-07-709553-7
4. Ciavotta M. et al. (2014). D3.2.2. MODAClouds Architecture – Final Version
5. D’Andria F. et al. (2014). D3.4.1 MODAClouds Integration Report – Initial Version
6. Weikun Wang, Giuliano Casale, “Evaluating Weighted Round Robin Load Balancing for Cloud Web Services”. SYNASC 2014: 393-400
7. Jonatha Anselmi, Giuliano Casale, “Heavy-traffic revenue maximization in parallel multiclass queues”. Perform. Eval. 70(10): 806-821 (2013)
8. Andreas Brunnert, et al., “Performance-oriented DevOps: A Research Agenda”. CoRR abs/1508.04752 (2015)
9. Prototype of Business Process Modeling System – Final Release, MODAClouds Deliverable # D9.2.2
10. Runtime Environment Final Release, MODAClouds Deliverable # D6.5.3
11. Giuliano Casale, Juan F. Pérez, Weikun Wang, QD-AMVA: Evaluating systems with queue-dependent service requirements. Perform. Eval. 91: 80-98 (2015)
12. Daniel Pop et al (2014). D3.7.1 MODAClouds evaluation report – initial version

Glossary

CCIM (Cloud-enabled Computation Independent Model)

Top-most abstraction layer used to describe the application and its data.

CPIM (Cloud-Provider Independent Model)

Middle layer where the cloud concerns related to the application are described in a cloud-agnostic way.

CPSM (Cloud-Provider Specific Model)

Bottom-most abstraction layer used to describe the cloud concerns needed to deploy and provision the application on a specific cloud.

CSP (Case Study Provider)

A partner in MODAClouds consortium using and evaluation the MODAClouds solution on concrete real-life applications. There are four partners running case studies: BOC (Business Process Modelling System), SOFTEAM (MODELIO Project Management Server), SIEMENS (Smart City Urban Safety Planner) and ATOS (Health-care application).

DSS (Decision Support System)

This is a tool of the MODAClouds IDE solution that supports the Feasibility Study Engineer in identifying the main risks and advantages in adopting specific cloud solutions and in determining a first estimate of costs associated to these solutions.

DC (Data Collector)

In MODAClouds runtime environment, Data Collector is a component of the monitoring platform gathering the raw data for various metrics (e.g. CPU utilization, I/O throughput etc).

KPI (Key Performance Indicators)

These are indicators used to determine the factors to be considered in order to evaluate a solution.

VM (Virtual Machine)

A Virtual Machine an emulation of a particular computer system. Virtual machines operate based on the computer architecture and functions of a real or hypothetical computer, and their implementations may involve specialized hardware, software, or a combination of both (source: Wikipedia)

7 Annex A - Modelling of replication using LINE random environments

7.1 Motivation and Problem Statement

For cloud applications, databases reside on cloud platforms. Some DBaaS providers allow the database to be replicated to cater to a surge in incoming traffic from the application layer. In the other scenarios, when changing availability zones, the database must be migrated seamlessly to the new zone without disrupting service to incoming requests, as in the BOC cloud migration case study. In both cases, a full backup/snapshot of the database is created and then migrated to the new replica/host. During the migration process the performance of queries on the database will be affected by the backup and migration operations, which may lead to SLA violations.

The goal of this evaluation is to assess the ability of the LINE solver to model the effect of replication on query response times. LINE is integrated within Space4CloudsDev and is an output of WP5. LINE provides the feature of random environments that allow the representation of variability in the modelled system. This work is summarized within the WP9.2.2 deliverable.

Data and Testbed

To be able to evaluate the accuracy of the LINE performance predictions we migrated the BOC business modelling database to a MySQL database instance on Amazon RDS. MySQL was used as Amazon does not currently support replication for MS SQL Server. Using an Amazon EC2 instance with emulated clients connecting to the database, we ran queries against the database and then initiated a replica creation from the master DB. Figure 1 depicts the testbed used in this evaluation.

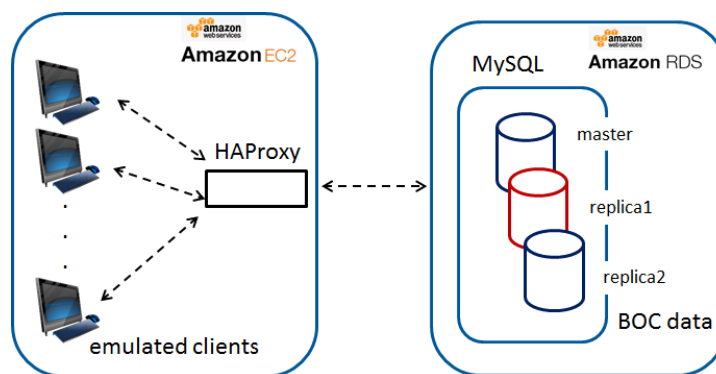


Figure 1. Amazon RDS testing environment.

7.2 Methodology

To be able to evaluate the effect of replication using LINE random environments, the following was performed:

- determine the phases in which performance is effected by replication
- collect runtime logs of response times for each phase

- fit the response times and calculate the service demands based on the logs
- parameterize the model: i.e., specify the service demands for each random environment
- compare LINE predictions to actual measurements

In the following we describe each of the above points.

7.3 Impact of replication

To determine the different phases during the replication process, we ran BOC queries against the master database and then created a replica as the queries were executing. We logged response times throughout. Figure 2 shows the actual query response times before the replica creation, during replica creation and data migration and finally after replica creation. During the replica creation phase the mean query execution time increases by over 20%. More importantly, the standard deviation of the execution time increases more than 20 times.

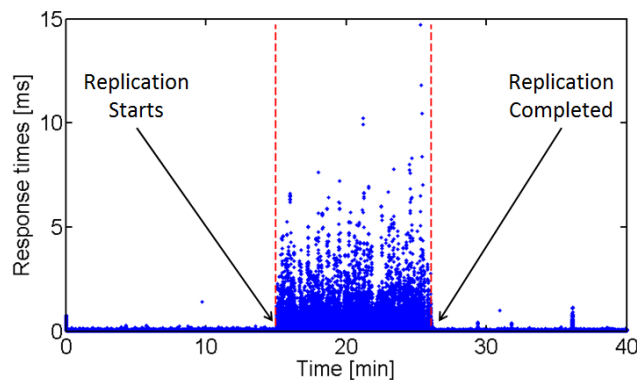


Figure 2. The impact of replication on query response time.

7.4 LINE Model

The LINE solver represents each phase during system execution as a random environment. Each random environment represents the service demands during that phase and the number of active DB servers. From Figure 3, when a new replica is created the model switches to the replication phase in which service demands are affected by data migration, after replication is completed the model switches to the phase in which one master and one replica are processing requests concurrently.

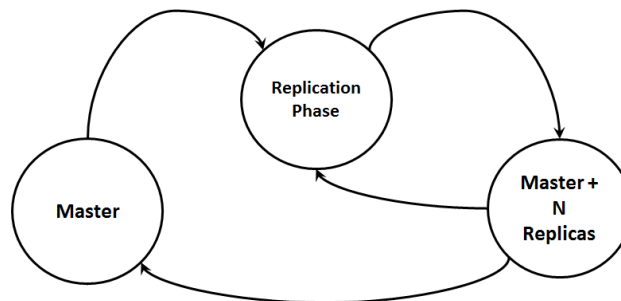


Figure 3. LINE random environment phases.

7.5 Results

Comparing the results of the LINE model to actual mean response times from the Amazon RDS instances, from Table 1, the mean error between LINE response time prediction and actual runs on Amazon RDS was less than 5% for pre and during replication phases and less than 10% for after replication.

Table 1. LINE accuracy in comparison to actual traces.

	before	during	after
trace	0.062	0.063	0.034
LINE	0.059	0.060	0.031
error (%)	4.150	4.361	8.907

8 Annex B – Multi-Cloud Load Balancer

8.1 Motivation and Problem Statement

The MODAClouds multi-cloud load-balancing reasoner determines the appropriate weights based on the actual demands for the requests on the backend servers. This determines the amount of load that will be routed to the backend in each individual cloud.

One of the deployment variants of BOC's BPM SaaS is the so-called 1-Click Try-Out. This demo version allows the user to access the tool without any registration, just by navigating to the service URL <https://try-out.boc-cloud.com> and entering a captcha. When marketing campaigns promote this low barrier access to ADONIS:cloud the load balanced deployment needs to be prepared for the traffic peaks. This has been done by adding additional application stacks deployed at CloudSigma to the load balanced pool with uniformly weighted round-robin policy. Having used Nagios⁵ and Icinga⁶ for infrastructure monitoring at BOC for some years the DevOps [8] team has built up know-how related to these tools. BOC plans to improve this approach by considering differently sized deployments in multiple clouds. The challenge then becomes the weight computation as it is by no means a linear function of the VM sizes. For this purpose, an experiment has been performed to use real monitoring data from a stress test and let the MODAClouds's multi-cloud load-balancer reasoner calculates the optimal weights.

The goal of this evaluation is to maximizing the throughput by minimizing the utilization imbalance between different servers, located globally distributed in different clouds, to improve system throughput or reduce response time. This work is summarized within the Deliverable # D6.5.3 (technical description) and D9.2.2 (BOC case study) deliverable.

8.2 Data and Testbed

⁵ <https://www.nagios.org/>

⁶ <https://www.icinga.org/>

By the multi-cloud load balancing, it is possible to assign different weights for the round robin load balancing policy set for the load balancer. With this, it is possible to evaluate the proportional load balancing according to the processing capacity of the target cloud, lead to different load balancing policies. For instance, it is possible to define the weights for each cloud: (1) Proportional to CPU,(2) Proportional to memory, (3) Proportional to price/hour or other possible quantifiable values.

The experiment setup included following components:

- an application stack deployed on an 8 core machine at Flexiant
- an application stack deployed on a 4 core machine at CloudSigma
- an application stack deployed on a 2 core machine at CloudSigma
- a JMeter⁷ based stress test client deployed in BOC's premises
- a MODAClouds load balancer deployed at CloudSigma

The initial setup was done with the weights assigned proportional to the CPU cores (8, 4 and 2). The stress test tool was configured to simulate 30 concurrent users, each logging in and opening 10 business process diagrams one after the other and logging out again. This sequence has been repeated for 30 minutes by each of the simulated users. The data collected with the stress test tool as well as the load balancer logs were handed over to WP5 for the calculation of optimal weights with the help of the MODAClouds multi-cloud load-balancing reasoner. The reasoner calculated the weights for the backend servers as follows: 25,45,30 respectively for the servers with 8,4, and 2 cores.

8.3 Methodology

In the 2 experiments, we looked at the application-level performance in terms of response time and throughput. In the first experiment we set weights according to the CPU cores, while in the second experiment we set them according to the demand calculated by the MODAClouds load-balancing reasoner. The results are presented in Table 2 (per second average response time and throughput as well as their respective confidence intervals (CI) for all VMs). In order to provide a clear comparison between the two settings, we also needed a metric to rank the policies for multi-cloud load-balancing. For this purpose, we used the ratio of mean response time to mean throughput as reported in Table 2. According to this metric, a lower value demonstrates a better policy as it shows a lower response time and a higher throughput as a result of that multi-cloud load balancing policy that is in place as the controlled variable in the experiments.

8.4 Case Study Results

From the boxplot in Figure 1 the throughput has been increased considerably for servers 2 and 3 but remained somehow unchanged in the first server (Flexiant server) even after the load balancing weights is adjusted according to the reasoner. The boxplot in Figure 2 demonstrate that the application level throughput as a result of load balancing policy change has been increased considerably.

The results in Figure 3 shows that the response time becomes more stabilized when the load balancing weights is set according to the demand estimated by the load balancing reasoner. Figure 4 demonstrates the scale by which the response time is decreased after the load balancing weights has been changed.

⁷ <http://jmeter.apache.org/index.html>

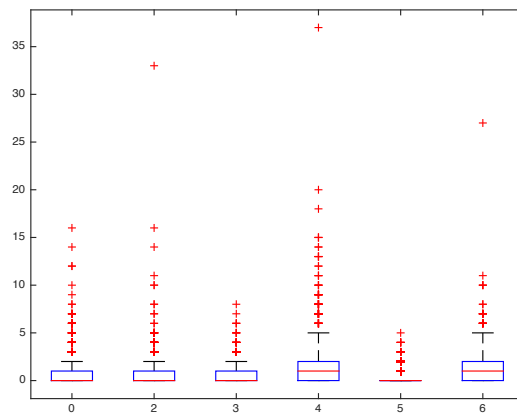


Figure 1. Throughput (request/s) per backend server (odds number servers with load balancing weights assigned with CPU cores while the even numbers are servers with load balancing policy with the weights assigned with MODAClouds load balancing reasoner).

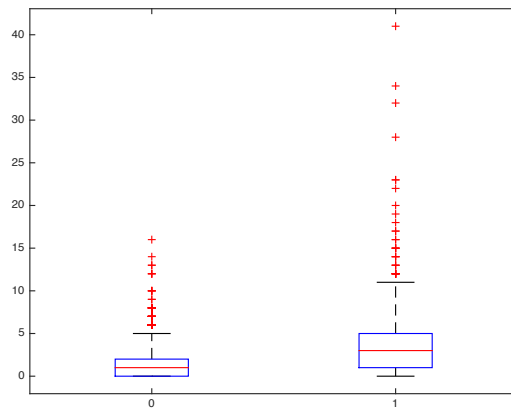


Figure 2. Average throughput (request/s) for ADONIS application (0: weights are assigned naively according to CPU cores, 1: weights are assigned according to MODACloud load balancing reasoner).

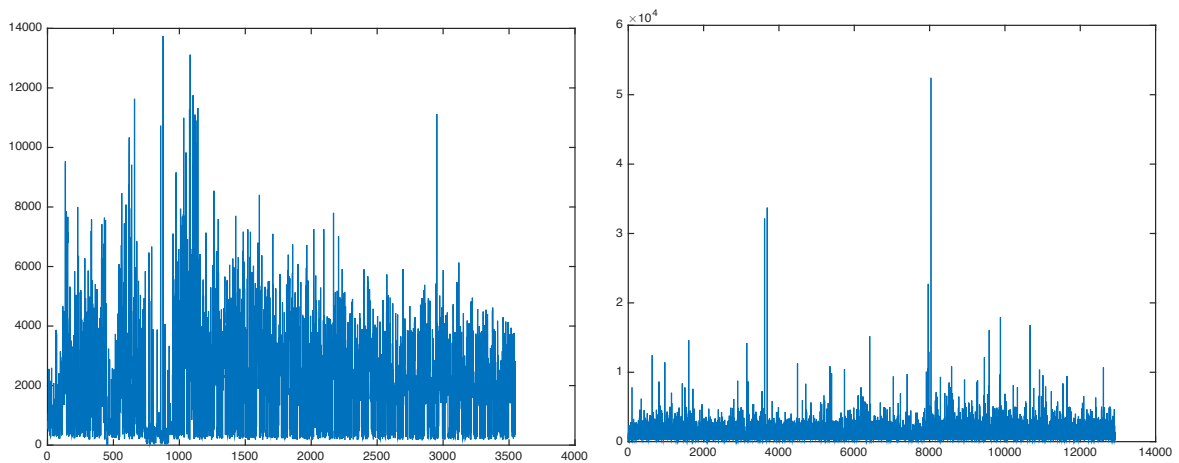


Figure 3. Reponse time (ms) before and after policy setting with MODAClouds load balancing reasoner.

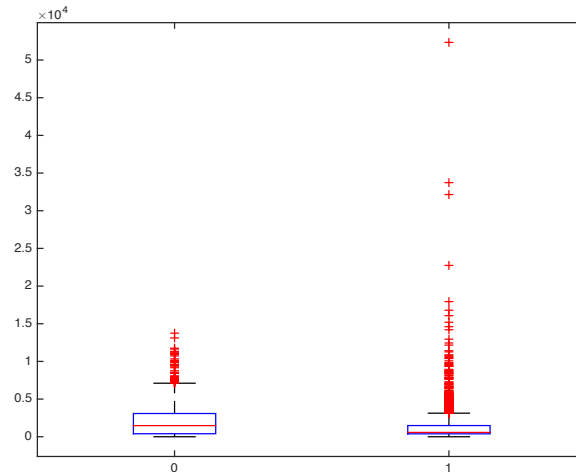


Figure 4. Reponse time (ms) characteristics before and after policy setting with MODAClouds load balancing reasoner.

We now discuss the summary of the results. The results Table 2 demonstrates that the average latency has been decrease and the average throughput of the application is increased. This also reflected in the ratio of mean latency to the mean throughput reported in the last column of the table. Therefore, we conclude that the setting of weights according to the MODAClouds load-balancing reasoner, which is based on demand estimation, is beneficial for multi-cloud applications. More detailed analysis of the results is presented in [10]. Note that a similar observation has been reported in an internal evaluation with a multi-cloud setting between Microsoft Azure and EC2 for OFBiz application in [11].

Table 2. Mean response time and throughput.

Experiment (weights)	Load Balancing Policy	Mean RT (mrt)	CI for mrt	Mean th. (mth/s)	CI for mth	mrt/mth
1 (20,10,5)	Proportional to CPU	1.938	[1.877, 1.999]	1.513	[1.399, 1.626]	1.2810
2 (25,45,30)	Proportional to demand	1.091	[1.069, 1.114]	3.382	[3.131, 3.633]	0.3228

8.5 More experimental results

We also performed some internal experimental study [6] with the MODAClouds load-balancing reasoner with OFBiz application as backend. The results in Figure 5, Figure 6 shows that the revenue (the weighted sum of the throughputs of each class of users) of the servers has been increased by 27% after the load balancing weights is set by the recommendation of the reasoner. In a similar multi-class setting [7], we observed a similar gain in the revenue by 30% as shown in Figure 7. In a multi-cloud setting, we setup a hierarchical load balancing in order to test the suitability of our load-balancing reasoner for multi-cloud setting. The results shown in Figure 8 demonstrate that when the load balancing is set according to a consolidated metric comprises of multiple factors, CPU size, Memory size and the price of VM, the throughput has been increased by 60%. In another study [12] we observed that some of the application requests show load dependent service times. We have extended balancing reasoner with a novel approximate mean value analysis (AMVA) solver. With the AMVA method, we are able to obtain the routing probabilities for each class of user to be dispatched to each VM given the fitted service rate from history data. The result is shown in Figure 9, where QD AMVA is the proposed method and LI AMVA is the one introduced in [7]. By relying on the proposed load

dependent AMVA, the throughput- optimization problem achieves a marked improvement (around 30%) in overall throughput compared to the load-independent AMVA.

Experiments are based on OFBiz application deployed on EC2. Different sessions of users are generated by the load generator. Heur and Optim are the proposed algorithms while “1112” and ”1113.5” refers to naïve weight assigning methods. Both 2 and 4 classes of sessions case shows that the weights returned by the proposed algorithms produce a higher throughput than the naïve ones. More details about these experiments can be found in [6,7].

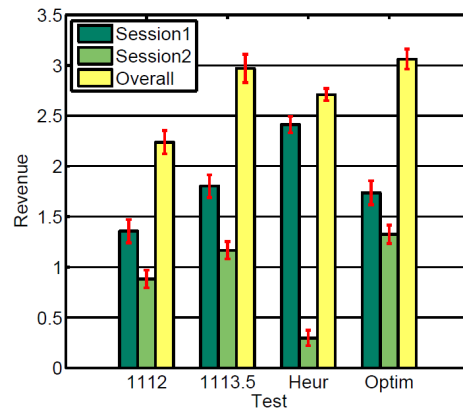


Figure 5. Revenue comparison of load balancing experiment with 2 classes of sessions of users [6].

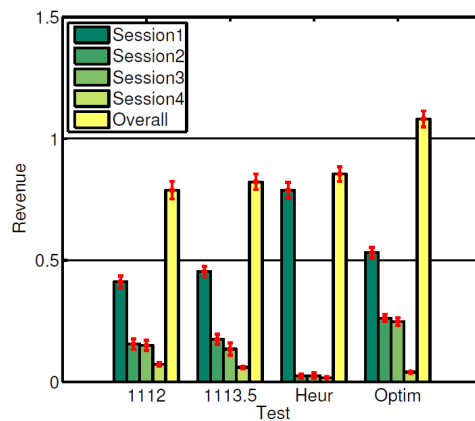


Figure 6. Revenue comparison of load balancing experiment with 4 classes of sessions of users [7].

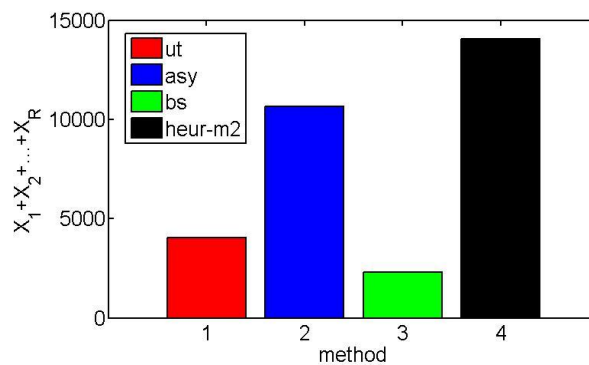


Figure 7. Revenue comparison of the load balancing methods for OFBiz application [2].

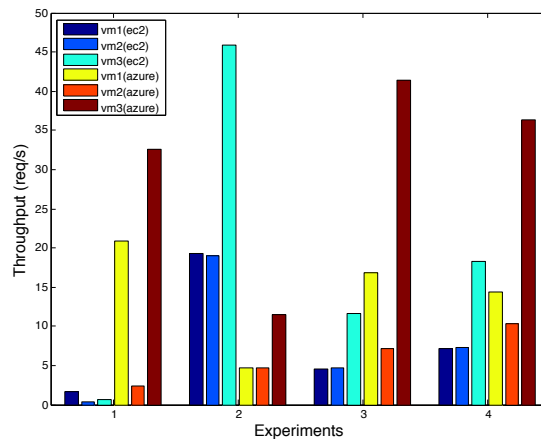


Figure 8. Mean throughput of the multi-cloud load balancing for OFBiz application [11].

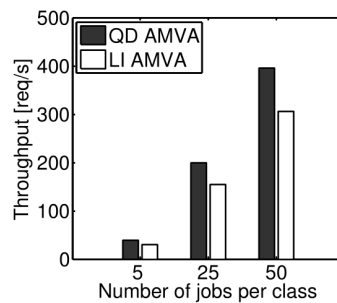


Figure 9. Comparison between load dependent and load independent approximation method.

8.6 Conclusions

Based on the summary of the results presented in Table 2, the observation with BOC multi-cloud load-balancing experimental study can be summarized as the following:

Improved performance (average response-time, throughput) - multi-cloud load balancing made the distributed ADONIS:cloud deployment more responsive and improved content delivery times by directing access to application backend processes according to the weights that were calculated by the MODAClouds load balancing reasoner. We observed a considerable increase in throughput for the application when the weights have been set proportionally according to the demand instead of VM sizes. We also observed a considerable decrease in the average latency for processing the generated requests.

Flexible load balancing. Multi-cloud local balancing enabled adaptive changes in the weights according to the heterogeneity of the resources in each cloud.

Reduce application downtime. Multi-cloud load balancing improved the availability of cloud-based applications by automatically directing user access to a new location anytime there is congestion in a cloud. MODAClouds multi-cloud load balancing facilitate this but simply changing the weights of the haproxy load balancing via the pyharapi APIs.

9 Annex C – New goals, query and metrics for Tower 4Clouds

Goal: Evaluate Tower 4Clouds with respect to the monitoring rules

Field	Description
MODAClouds artefacts of study	Tower 4Clouds
Purpose	Characterize the actual artefact
Focus	Characteristics: Functionality Feature(s): <ul style="list-style-type: none"> Transformation of QoS Constraints to monitoring rules Deployment of monitoring rules
Stakeholders	Application provider
Case study and requirements	CSPs requires the ability to support incremental deployment
KPI	SO1, SO2 , SO4, SO5, SO6
Context factors	Monitoring implementation and integration with state-of-the-art monitoring tools

Questions and metrics	Description
Q1	Is information exposed by the Manager API coherent with the installed monitoring rules?
M1.1	Every successfully installed monitoring rule is visible both in the webapp and using the GET /monitoring-rules api
M1.2	Installed rules required metrics are either available through the /required-metrics api or provided as output from an other rule (/metrics api or visible under metrics tab in the webapp)
M1.3	The /metrics api and the metrics tab in the webapp show all and only the metrics defined in the outputMetric actions of the installed rules
DCP	Who: WP6 When: within M36 for the final evaluation. How: Install monitoring rules through either the webapp or the REST api and check the results on both the webapp and through the REST api http://deib-polimi.github.io/tower4clouds/docs/manager/rest-api/ To Whom: MODAClouds project, Commission
Q2	Is the Tower 4Clouds able to satisfy monitoring rules?
M2.1	RestCall action perform the requested rest call when condition is verified
M2.2	CloudMLCall action perform the requested call to cloudML when the condition in the rule is verified
DCP	Who: WP6 When: within M36 for the final evaluation. How: Install rules and verify that actions are executed correctly

	To Whom: MODAClouds project, Commission
Q3	When installing wrong monitoring rules, is the error message informative enough?
M3.1	The error message allows to fix the rule when the rule is not a valid XML file
M3.2	The error message allows to fix the rule when the rule does not adhere to the schema in https://raw.githubusercontent.com/deib-polimi/tower4clouds/master/rules/metamodels/monitoring_rules_schema.xsd
DCP	Who: WP6 When: Within M36 for the final evaluation. How: Install wrong rules through the webapp and check that the error allow to fix the rule To Whom: MODAClouds project, Commission

Goal: Evaluate Tower 4Clouds with respect to the monitoring data collection

Field	Description
MODAClouds artefacts of study	Tower 4Clouds
Purpose	Characterize the actual artefact
Focus	Characteristics: Functionality Feature(s): <ul style="list-style-type: none"> • Standard data collectors for VM metrics • Integration of custom data collectors
Stakeholders	Application provider
Case study and requirements	BOC case study requires <ul style="list-style-type: none"> - support for the implementation of custom monitoring data analysis modules - support for the implementation of custom data collectors - the ability to leverage IaaS data collectors for both Linux and Windows - the ability to have the monitoring configuration adapted on changes the ability to support incremental deployment
KPI	SO1, SO2, SO4, SO5, SO6
Context factors	Monitoring implementation and integration with state-of-the-art monitoring tools

Questions and metrics	Description
Q1	When a data collector using the data-collector-library is started, configured with the correct manager IP and port, is it correctly registered in the manager?
M1.1	The manager /data-collectors API contains a new entry reflecting the data collector specification
M1.2	The /resources API or the model tab in the webapp contains the new instances of the resources provided to the data collector
M1.3	When the communication between the data collector and the manager is broken for longer

	than the configured keep alive period (default value is 30 seconds), the data collector and its resources are unregistered by
DCP	Who: WP6 When: within M36 for the final evaluation. How: Start a data collector that uses the data-collector-library, providing information about the manager IP and port and about the resources it is responsible for To Whom: MODAClouds project, Commission