Title:	*MODAClouds evaluation report – initial version*
Authors:	*Arnor Solberg (SINTEF), Franck Chauvel (SINTEF), Alexander Gunka (BOC), Stepan Seycek (BOC), Marcos Almeida (SOFTEAM), Antonin Abherve (SOFTEAM), Elisabetta Di Nitto (Polimi), Francesco Di Forenza (Polimi), Lorenzo Cianciaruso (Polimi), Giovanni Paolo Gibilisco (Polimi), Marco Scavuzzo (Polimi), Roi Sucasas Font (ATOS), Florin Picioroaga (Siemens), Jacek Dominiak (CA), Juan Perez-Bernal (Imperial), Weikun Wang (Imperial), Gabriel Iuhasz (IeAT), Ciprian Craciun (IeAT), Silviu Panica (IeAT)*
Editor:	*Daniel Pop (IeAT)*
Reviewers:	*Aida Omerovic (SINTEF), Francesco D'Andria (ATOS)*
Identifier:	*Deliverable # D3.7.1*
Nature:	*Report*
Version:	*1*
Date:	*15 October 2014*
Status:	*Final*
Diss. level:	*Public*

Executive Summary

This report describes how well the tools and methods developed in the MODAClouds project fulfil the requirements of case studies and identified requirements at WP level. The current version of the deliverable is based on the outcomes of the evaluation methodology developed in the Evaluation plan (D3.6) that was applied on the first prototypes developed by the four case study providers in MODAClouds. The evaluation plan is based on Basili's Goal Question Metric (GQM) method and software testing techniques. The outcome of this evaluation is aimed to be used as an input for the revision of the concepts in WP2 to WP6. The evaluation itself assesses one by one the goals defined in the Evaluation plan for each MODAClouds artefact. Next, the MODAClouds main objectives defined in the DoW and the Key Performance Indicators (KPI) are evaluated. This is an intermediate report that will be updated in M36.

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

Chapter 1

1 Introduction

This deliverable presents the initial evaluation of the MODAClouds software-based solution with respect to i) the main objectives of the project, ii) the Key Performance Indicators (KPIs) iii) requirements that were identified in all WPs and collected in a coordinated way by T3.1, and iv) the specific objectives of the technical work packages. All these requirements of the evaluation and validation phase of the MODAClouds solution were refined and fully described in the D3.6 Evaluation Plan [2], which defines the goals and metrics of the evaluation process, the evaluated KPIs, as well as the schedule for executing the evaluation plan. Basically, the entire evaluation process is derived from Goal-Question-Metric (GQM) [3] methodology.

This is an intermediate report that takes a snapshot of all MODAClouds artefacts at M24 and the report will be updated in M36. It assesses the status of the goals defined in the evaluation plan [2] for each of the MODAClouds artefacts. Next, the MODAClouds objectives defined in the DoW [1] and the Key Performance Indicators (KPI) are evaluated using the results obtained from the evaluation of goals.

An overview of the MODAClouds artefacts, which are subject of current evaluation, is illustrated in **Error! Reference source not found.**, extracted from D3.6. There two major 'super-artefacts', MODAClouds IDE and MODAClouds Runtime platform (in light blue in table), each being decomposed in measurable artefacts evaluated accordingly.

Artefact	Short description
MODAClouds IDE	The MODAClouds IDE integrates a set of tools supporting the design-time activities of the MODAClouds approach
Decision Support System **[DSS]**	This tool 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.
MODACloudML Functional Modelling Environment **[Modelio4MODAClouds]**	This tool supports the modelling of cloud applications and of the data they manipulate. It supports the CCIM, CPIM and CPSM models and the generation of the deployment artefacts from them.
QoS Modelling and analysis [**SPACE4Cloud+Line**]	This tool allows the QoS Engineer to perform the following operations: – Modelling of QoS requirements – Quality prediction and analysis of QoS requirements fulfilment – Generation of monitoring rules – Definition of the goals and constraints driving the execution of self-adaptation
Cloud Modelling Language **[CloudML]**	This tool supports the model specification of deployment and resource provisioning including deployment and provisioning constraints, and interfaces with the MODAClouds runtime components in order to effectively deploy the cloud application
Data modelling & migration **[DMM]**	This tool is able to analyse the data that are going to be manipulated by the applications and the constraints on them (and on their storage and retrieval).
Filling the Gap Design-Time Manager	This component is responsible for retrieving information from the runtime platform where the cloud application is deployed. This

[FG-DTM]	information is used for two main purposes: first, to update the parameters of the design-time models, allowing these to better capture the actual behaviour of the application; and second, to provide the Cloud App Admin and the QoS Engineer with information about the behaviour of the application at runtime.
MODAClouds Runtime platform	The MODAClouds runtime platform provides a set of modules for managing execution and adaptation of cloud applications in a multi-cloud environment
Multi-cloud deployment-models at runtime **[MDM-Models @runtime]**	The MDM-Models@runtime component stores the complete deployment model of the application and keeps it alive at run time, that is, any change in the system is reflected in the model, and, vice versa, any modification performed to the model through the exposed MDM-Models@Runtime API is reflected in the running system.
Monitoring platform **[monitoring platform]**	The Monitoring Platform is responsible of monitoring the status of the application.
Self-adaptation platform **[Self-adaptation platform]**	The self-adaptation platform is responsible of controlling the application so that SLAs are respected. It uses monitoring data pushed to it by the monitoring platform and it controls the system by acting on the MDM-Models@Runtime component.
Execution platform **[Execution platform]**	The Execution Platform acts as a middleware layer (between the MODAClouds run-time environment and the targeted IaaS / PaaS where the application is running). During the application execution, all management operations on the execution platforms are accessed through this component. This component is composed by different libraries: **- jCloud** **- Flexiant** **- mOSAIC** **- PaaS Unified Library**
Support platform **[Support platform]**	The Support platform contains all those services (load balancer, object database, DNS, Artefact Repository, etc.) necessary to support the execution of the cloud application in a multi-cloud environment.

In the following sections, each artefact is evaluated one by one with respect to the goals defined for it. The content of each sub-section is a synthesis of the experiments performed by 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 specified goal
- Deviations – delays versus the schedule included in the evaluation plan
- Recommendations – action points and suggestions for the coming period w.r.t evaluated artefact.

The reader is referred to D3.6 Evaluation plan [2] for a detailed description of the goals and related questions. It is not in the scope of this document to duplicate that information. Here we will refer each goal by its title, e.g. “Evaluate DSS with respect to the risk analysis methodologies”. For a selection of domain-specific key terms (e.g. CCIM, DSS, VM) used throughout this document please see Glossary section at the end of the document. For a description of the MODAClouds components we mention in this document the reader can refer to deliverables D3.2.2 MODAClouds architecture – Final version [4] and D3.4.1 MODAClouds Integration Report – Initial version [5].

Mind that that the main target of this report is the software-based MODAClouds solution and its components, thus, it is not a general evaluation of all activities and progress in the project as a whole.

The remainder of the document is structured as follows. The next two sections (MODAClouds IDE evaluation and MODAClouds Runtime platform evaluation) present the assessment of each artefact included in IDE and Runtime platform, respectively. Section **Error! Reference source not found.** discusses the status of MODAClouds objectives, while section **Error! Reference source not found.** presents the assessment of the MODAClouds KPIs. Last section of the document provides concluding remarks.

Chapter 2

2 MODAClouds IDE evaluation

2.1 Decision Support System [DSS]

Who:	WP2 (CA) Health-care application (ATOS)
Overall:	
At this stage, the evaluation of Decision Support System (DSS) has been performed by CA. With respect to the three goals defined in the evaluation plan for this tool in WP2, the outcome of evaluation shows fulfilment of all three goals. Each goal is evaluated by two questions with one metric per question. Out of six questions, all have been answered, except for Q2 of third goal whose evaluation was postponed to M36.	
Goal 1	Evaluate DSS with respect to addressing different types of multi-cloud deployment architectures
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
A 5-step approach for guiding the user though the process of selecting the most suitable Cloud Service Provider was implemented and extensively described in D2.1.2.	
Goal 3	Evaluate DSS with respect to simplification of decision making process
The DSS is structured around the idea of guiding the user though the process of selection using wizard. This allows the user to address each of the selection criteria at the time as well as allows different actors participating in the selection to describe different requirements and assets needed to fulfil the needs of the deployment.	
Deviations	
Health-care case study postponed the evaluation of DSS for M36.	
Recommendations	
ATOS to proceed with the evaluation of DSS and to provide feedback to WP2 in timely manner so that improvements and fixes can be integrated smoothly. To be revalued in M36.	

2.2 MODACloudML Functional Modelling Environment [Modelio4MODAClouds]

Who:	BPM System (BOC) MODELIO Project Management Server (SOFTEAM)

	Smart City Urban Safety Planner (SIEMENS) Health-care application (ATOS) WP4 (SOFTEAM)
Overall:	
MODELIO has been extensively evaluated in the first two years of the project by all CSP. Their evaluation has been done with respect to modelling power of the IDE in supporting the three abstraction levels (CCIM, CPIM and CPSM), and with respect to cloud provider selection and code-generation from models. The results are documented in a set of eight deliverables, D8.{2, 3, 4, 5}.{1, 2} delivered at M18 and M24. All models were implemented on the Functional Modelling Tool. No deviations from Evaluation plan were observed. The one and only 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
(BOC) Models have been created using the IDE describing all relevant aspects of the case study. The case study models are stored in the MODAClouds model repository and are detailed in the final case study design deliverable D8.3.2. 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). Regarding the requirement to perform this for 5 different deployment cases: 1. Deployment alternative for CCIM service assembly has been tested for the only relevant scenario in this context (the case study application) 2. Instance level deployment from type level deployment has so far been tested for two different scenarios (single VM vs. two VMs) 3. CPSM deployment alternative from CPIM deployment has so far been tested for two different scenarios (single VM vs. two VMs) (SOFTEAM) A detailed evaluation of MODELIO Project Management Server case study has been performed together with POLIMI and IMPERIAL and it is extensively documented in D8.2.2. CPIM models involving PaaS and IaaS support were built by utilizing the IDE and fed to SPACE4Cloud for preliminary experiments. CPSM models of deployment targeting Amazon and Flexiant, involving PaaS DBs provided by both of them, were successfully designed as well. (ATOS) Overall, the IDE tool provides enough modelling support on all three levels as far as this case study is concerned. Models are documented in D8.4.2. (SIEMENS) Joining other CSPs, SIEMENS evaluates the modelling support offered by IDE at all abstraction layers (CCIM, CPIM and CPSM) as good. Deployment models were created for Amazon EC2 and Flexiscale, and are documented in D8.5.2. (WP4) All models were implemented on the Functional Modelling Tool (except for the CPIM and CPSM level data models, which upon decision taken at M12 will not be supported by the Functional Modelling Tool but by the Data Mapping Component itself). With respect to code generation and transformations, this will be evaluated by CSP in the forthcoming period. According to these, an adequate level of transformations and code generation is already provided: (1) SPACE4Cloud input files are successfully generated from Service Definition Model and Service Orchestration Model and (2) refinement transformations were implemented and are available to CSPs evaluation. While code generation targeting SPACE4Cloud/LINE models 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.
Deviations
No deviations. Full evaluation of MODELIO4MODAClouds will only be available at M36.
Recommendations
ATOS: To evaluate usage models at CCIM layer for M36. To evaluate data models, data migration and self-adaptation for M36. During the evaluation of the MODELIO4MODAClouds IDE, CSP provided suggestions and recommendations shaping the evolution of the tool. In summary, - As manual tweaks are necessary to get the desired representation in the diagram (e.g. needed to set representation of "port" elements to "Image", needed to set representation of internal structure of "Node" types to diagram), there is still room for improvement in the last year of the project. - Cloud resources have to be added manually, hence automation in direction will be benefical.

2.3 QoS Modelling and analysis [SPACE4Cloud and LINE]

Who:	BPM System (BOC) MODELIO Project Management Server (SOFTEAM) Smart City Urban Safety Planner (SIEMENS) WP5 (POLIMI, IMPERIAL)
Overall:	
QoS modelling and analysis tools were assessed both internally by WP5 partners, and externally by CSPs (BOC and SOFTEAM). Out of the five goals in this category, three (2, 3 and 5) were only partially achieved. Extensive experiments were executed to evaluate LINE features. CSPs had a good involvement in the evaluation of this MODAClouds artefact.	
Goal 1	Evaluate MODAClouds monitoring approach and tools (BOC)
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, MODAClouds IDE provides adequate transformations from CCIM QoS constraints to CPSM monitoring rules. Monitoring rules can be delivered by the use of the IDE, but they need to be completed manually w.r.t the trigged actions in case the corresponding monitoring rules are not respected.	
Goal 2	Evaluate MODAClouds QoS and Cost analysis tools (SOFTEAM)
SOFTEAM setup an experiment in Flexiant and compared the QoS and costs measured on Flexiant, and on the machines themselves, to those produced by SPACE4Cloud/LINE. We created two virtual machines in two different Flexiant networks, one with a HTTP server serving MODELIO models and another one generating traffic following the same pattern than in a SPACE4cloud/LINE model. Projected CPU usage and response time were 100% and 30s, while actual values were between 8-15% and 200-300ms. The difference between both numbers was deemed to outdated resource database information. Cost predictions were off by 7%. We noticed that some cost items are were not taken into consideration by SPACE4clouds/LINE. Flexiant charges for things like disk IO, network IO, and network and IP utilisation that are not taken into consideration by the SPACE4Cloud/LINE. In our example these parameters did not represent a large percentage of the cost, and therefore the difference	

between the expected and computed costs was negligible, however in a more complex application (with multiple instances) or with a more IO bound behaviour the estimations may differ considerably from the predicted actual value.	
Goal 3	Evaluate the QoS Modelling and Analysis Tool installation
The QoS Modelling and Analysis Tool is easy to install based on the provided documentation. The installation experiment included Palladio Bench, together with Eclipse Kepler, and SPACE4Clouds. At this stage, the QoS Modelling and Analysis Tool is not directly integrated with the WP2 cost estimation web-based service. If the results of the WP2 tool can be imported in MODELIO then the QoS Modelling and Analysis tool can use its integration within MODELIO to run analysis.	
Goal 4	Evaluate QoS requirements, constraints definition and performance evaluation
The QoS requirements and constraints supported by the tool satisfy the needs of the MODELIO Project Management Server case study.	
Goal 5	Evaluate LINE with respect to the generation and solution of QoS models
Basically, LINE supports the solution of QoS models from the MODAClouds application models, but additional transformations are required for this purpose. 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.	
Deviations	
SIEMENS delayed the generation of monitoring rules for M36. CSP delayed the evaluation of the goals defined within WP5 for M36.	
Recommendations	
Supporting additional types of constraints should be considered. Additional constraints defined for the case study – related to users, REST calls, Database performance – have to be manually transformed to monitoring rules (see Annex for details) (BOC) Predictions produced by SPACE4Cloud/LINE can be improved by fiddling with the input resource demand specifications and other parameters to get results that are closer to the actual system. It is not clear which process should be followed to make sure predictions will match more complex systems. (SOFTEAM) The evaluation of SPACE4Cloud/LINE by CSP is going to be performed during the third year of the project. To further investigate the integration between QoS modelling and analysis tools and WP2 cost estimation services. To continue the investigations on applying QoS and cost analysis tools on more complex applications. To improve the integration between LINE and generated QoS models. To continue the development of LINE.	

2.4 Cloud Modelling Language [CloudML]

Who:	Smart City Urban Safety Planner (SIEMENS) MODELIO Project Management Server (SOFTEAM) WP4 (SOFTEAM, SINTEF)
Overall:	
CSPs assessed CloudML with respect to two main goals: multi-cloud support and deployment of complex applications (Constellation). The evaluation process is still in progress for multi-cloud support and need to be completed in Y3. Some limitations of CloudML surfaced during the deployment of a complex application.	
Goal 1	Evaluate MODAClouds IDE with respect to multi-cloud support
This goal evaluates the support offered by CloudML w.r.t. multi-cloud deployment. In the experiments performed by SIEMENS they managed to generate the model from MODAClouds IDE, but the generated model need to be further matched against the manual one. Nevertheless, some components of the Storm cluster (SIEMENS's case study), such as Nimbus VM, Workers VM, Zookeeper VM etc., were successfully deployed, but there are problems in creating and sharing security groups for limiting the access to the ports. SOFTEAM and SINTEF jointly evaluated CloudML's support for multi-cloud in three experiments, as follows: 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) The results are: 1) Successful deployment and execution for Constellation and SensApp 2) Successful deployment and execution for Constellation 3) Successful deployment and execution for SensApp On top of this, SINTEF also evaluated 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) and reported successful deployment and execution for SensApp. The following three experiments by SINTEF were postponed for M36: 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) 6) Management replication of data on different cloud providers at both IaaS and PaaS level 7) 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)	
Goal 2	Evaluate MODAClouds IDE with respect to generated scripts (SOFTEAM)

Although deployment scripts were generated based on the models, successful, full-fledged deployment was hindered by the following limitations of CloudML: (1) CloudML has still poor support for synchronization of events happening in different instances during deployment (i.e. when initialization of instance A needs to happen after initialization of instance B). This slightly hindered the quality of execution as sometimes manual intervention were necessary. (2) Our deployment depends on an external service that does need to be configured or initialized (our license server). Since CloudML does not support this kind of services yet, some manual intervention on deployed instances is necessarily.
Deviations
SIEMENS: Evaluation of the generated models has not been completed at M24. In the multi-cloud deployment scenario, some modules of the system (e.g. Reconciliator VM and Tomcat cluster deployment cf. D8.5.2) will be evaluated for M36. SOFTEAM, SINTEF: The evaluation of the last three steps of the scenario were postponed to M36.
Recommendations
Fix the deviations of the first evaluation period and continue with the execution of Evaluation plan. Investigate the ways to extend CloudML towards the overstepping the blocking points identified by CSP up to now.

2.5 Data modelling & migration [DMM]

Who:	WP6 (POLIMI)
Overall:	
At this stage, the evaluation of Data Modelling and Migration (DMM) artefact has been performed by POLIMI w.r.t. the three goals defined in the D3.6 [2]. All questions due in M24 were evaluated and all three goals are considered achieved.	
Goal 1	Evaluate data migration overhead
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 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.6.6 for details.	
Goal 3	Evaluate scalability of data migration system
The data migration system is considered scalable w.r.t. to the number of entities to be migrated.	
Goal 4	Evaluate lossless data migration
Experiments performed have achieved 100% data migrated.	
Deviations	
According to Evaluation Plan, some of the CSPs should have been running the evaluation of this component to M24 (ATOS), while others (BOC) scheduled the evaluation for M36. The decision at	

project level is that all evaluations of this component will be performed by M36 involving selected CSPs and their feedback shall be provided to all WP6 partners.
Recommendations
The scarce assessment with regards to this component does not allow us to draw consistent recommendations for the evolution of this component in the third year of the project. It is paramount that CSPs invest more effort in testing this component and work closely with WP6 partners in providing feedback and running conclusive experiments in the coming period.

2.6 Filling the Gap Design-Time Manager [FG-DTM]

According to Evaluation Plan (D3.6) [2] evaluation activities related to this artefact are scheduled in Y3 of the project, thus the results will only be available for D3.7.2 in M36.

Chapter 3

3 MODAClouds Runtime platform evaluation

3.1 Multi-cloud deployment- models at runtime [MDM-Models @runtime]

Who:	WP6 (SINTEF)
Overall:	
With respect to the two goals defined in the evaluation plan for MDM-Models@runtime platform, they are only partially achieved at this stage. Out of four questions related to the first goal, one is scheduled for M36, all others were evaluated. Results and comments show that basic functionalities are supported, but the evaluation of full implementation will be repeated for M36.	
Goal 1	Evaluate MDM-Models@Runtime with respect to its causal connection functionality on multi-cloud
The dynamic adaptation of the deployment of a cloud application deployed on IaaS was successfully executed with SensApp. Migration of a cloud application from one cloud IaaS provider to another IaaS cloud provider was successfully executed for SensApp, but without data migration / synchronization. For the multi-cloud-related experiments, Amazon and SINTEF's private cloud were used. Migration between different PaaS has not been tested so far. With respect to synchronization between the deployment model at runtime and the running system, the status of VMs and PaaS services are kept synchronized, but the status of software components if not yet monitored. MDM-Models@Runtime is integrated with the MODAClouds IDE, but high level commands have not been tested so far. The integration with Self-adaptation reasoner is not yet available and it will be evaluated in Y3.	
Goal 2	Evaluating enactment of a specified bursting scenario for the MDM-Models@Runtime platform
The evaluations shown that **simple** bursting scenarios are enacted, but the bursting is achieved by providing as input a deployment model and not through a burst command.	
Deviations	
No deviations.	
Recommendations	
Tests performed so far, although proved a functional component, need to be re-run in Y3 in order to fully cover the expectations specified in the Evaluation Plan (e.g. migration between PaaS providers, data migration, bursting etc.). The evaluation of this component by CSPs need to be clarified and the selected CSPs should engage in experiments to validate the MDM-Models@runtime artefact by M36.	

3.2 Monitoring platform [monitoring platform]

Who:	WP6 (POLIMI, IMPERIAL)

	BPM System Case Study (BOC) MODELIO Project Management Server (SOFTEAM)
Overall:	
This artefact was only partially evaluated at M24, full evaluation of all the goals defined for it is scheduled for M36. Installation of Monitoring Platform has been seen as acceptable by WP6 evaluators and CSPs. The initial tests performed w.r.t. installation yield very good results that mostly match the expectations at this stage. SOFTEAM integrated three components of the Monitoring Platform in their architecture and developed two new ones based on the API provided by the platform.	
Goal 1	Evaluate the Monitoring Platform installation
Partners who evaluated (POLIMI, BOC) the installation of Monitoring Platform (using RPM packages) concluded that the installation is easy enough providing the documentation, but human involvement is required due to missing scripts (e.g. init.d service scripts). PaaS deployment: The Deterministic Data Analyser (DDA), Monitoring Manager (version 1.2+) and Knowledge base have been successfully deployed on Heroku PaaS. The infrastructure DC cannot be deployed on PaaS because it is for IaaS level monitoring (CPU, Memory etc.). However, being a library, the application level Data Collector (DC) can be deployed on PaaS, as successfully done by ATOS. (POLIMI and IMPERIAL) SOFTEAM integrated in Constellation three components of the Monitoring Platform (the Deterministic Data Analyser, Knowledge Base and the MODAClouds Monitoring Manager) and developed two new ones (Constellation Data Collector and Constellation Data Analyser) using the API provided by the platform in order to support the communication between Constellation components and MP.	
Goal 2	Evaluate the Monitoring Platform with respect to the monitoring rules
The tests done by POLIMI have shown that the configuration of the DC is correctly stored in the knowledge base. After installing a rule and querying the knowledge base, the fields collectedMetric and monitoredTarget are correctly set. Also, the required CSPARQL queries are correctly installed in the DDA based on the specification of the monitoring rule. The installation of a new rule and the registration of observers happen in few instances. The query has been installed in less than 1 seconds, i.e. 10 times faster than required in the plan. Thus, we can summarize that the Monitoring Platform satisfies the monitoring rules.	
Goal 3	Evaluate the Monitoring Platform with respect to the monitoring data collection
There is an instance of a DC running for each DC instance enabled in the knowledge base. A test has been made to collect CPU utilization with the Data Collector. The Data Collector is able to retrieve the configurations such as the samplingProbability and the samplingTime from the knowledge base and send back the CPU utilization to the DDA.	
Goal 4	Evaluate the Monitoring Platform with respect to the statistical data analysers
There is an instance of a statistical data analyser (SDA) running for each SDA instance in the knowledge base. The SDA is able to retrieve the configurations such as the timeStep from the knowledge base and send back the estimated demand to the DDA. 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%.	
Goal 5	Evaluate the Monitoring Platform overhead
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.	
Deviations	
Evaluation of SDAs configuration management in the knowledge base based on the monitoring rule specification has been postponed to M36. Nevertheless, there are other metrics scheduled for evaluation at M36 in order to fully assess the goal related to monitoring rules (Goal 2). Except for Monitoring Platform installation the other goals under Monitoring Platform component were not evaluated by CSPs.	
Recommendations	
The configuration of Monitoring Platform should be fully automatized, maybe with some scripts that permits to the user just to start the components. Another suggested improvement for installation of Monitoring Platform is the implementation of a tighter interaction between the MODAClouds Deployer and the knowledge base so that the deployment of the application will also configure the Monitoring Platform. Continue the evaluation of the Monitoring Platform by CSPs (BOC, SOFTEAM, others) and by WP6 partners according to the Evaluation Plan. More experiments for deployment and monitoring are needed to validate the integration of MP in Constellation, plus testing the migration to other Cloud providers (SOFTEAM).	

3.3 Self-adaptation platform [Self-adaptation platform]

Who:	WP6 (IMPERIAL)
Overall:	
The initial analysis of Self-adaptation platform performed on a real test bed for the Ofbiz application revealed promising results. All three questions assessing the single goal defined for this MODAClouds artefact were considered to be accomplished at a level labelled “to some extent”. Further efforts will be put in the development and validation of this component in the third year of the project.	
Goal 1	Evaluate the Self-adaptation mechanism
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 provider without introducing significant overhead and QoS violations. Thus, we can conclude that scaling policies are effective. Load balancing is supported inside a single Cloud provider among heterogeneous VMs. The algorithms implemented set out to maximize the system throughput to achieve optimal resource usage. They also feature different classes of users, like gold, silver and bronze users which may have differentiated SLAs. W.r.t. effectiveness of the load balancing policies implemented within the Self-adaptation reasoner, comparison has been done between naive weight assigning method and algorithms implemented in the	

load balancing reasoner to determine the weights for weighted round robin load balancing policy.
Deviations
No deviations from the plan were observed.
Recommendations
To continue the development and evaluation of this component during the third year of the project. To implement and evaluate the Load Balancing between different clouds (target M30). To perform comparisons against other tools and scenarios (target M30).

3.4 Execution platform [Execution platform]

Who:	BPM System (BOC) MODELIO Project Management Server (SOFTEAM) Health-care Application (ATOS)
Overall:	
To this date, the Execution platform has been evaluated by three CSP. All rated the human involvement in deploying tools and applications being at an acceptable level. Some experiments were delayed for M36 and shall be performed alongside with those already scheduled for the Y3 of the project.	
Goal 1	Evaluate MODAClouds deployment tools
BPM System case study (BOC) has been successfully deployed to a single Windows-based VM. Although human intervention was required, it is considered at an acceptable level. At this stage, overall deployment support is considered as adequate based on BPM case study. Constellation was successfully deployed and executed. All deployment scripts were generated using the IDE (SOFTEAM). Health-care application and its database were successfully deployed and executed in CloudBees (ATOS).	
Deviations	
Puppet integration is a work in progress and its evaluation was postponed to M36 (BOC). Evaluation of integration of PaaS services with own application delayed for M36 (ATOS).	
Recommendations	
CSPs to continue evaluation of the platform according to the Evaluation plan. Address all the deviations observed at M24.	

3.5 Support platform [Support platform]

Who:	WP6 (IEAT)
Overall:	

The evaluation of support platform at M24 targeted three goals, related to Load Balancing API, Object store and Artefact repository. Two of them have been evaluated as achieved, while the Artefact repository doesn't yet support search feature.	
Goal 1	Evaluate Load Balancing RESTful API
The LB RESTful API is suitable for exposing the load-balancer to the self-adaptive reasoned (SAR); Haproxy API allows external clients to change the weight of each server with a REST call.	
Goal 2	Evaluate the Object Store module with respect to its functionality
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.	
Goal 3	Evaluate Artefact Repository with respect to its functionality
Artefact Repository is able to store and retrieve artefacts. Search is not yet supported.	
Deviations	
No deviation for Support platform.	
Recommendations	
Extend Artefact repository to support search feature and re-evaluate for M36. Investigate the integration with the Execution platform and deployment on multi-cloud environment.	

Chapter 4

4 MODAClouds objectives evaluation

This chapter evaluates the MODAClouds main objective, its sub-objectives as they were defined in DoW [1]. This evaluation is performed both at the MODAClouds Milestone 4 (First evaluation at M24) and it will be repeated at MODAClouds Milestone 6 (Finalization at M36). The following sub-sections of the chapter present the results of this evaluation at M24 using a table-based, as defined in D3.6 [2]. Each table has has four columns:

- 'Is it delivered?' – contains a very short description of the objective
- M24 – the evaluation at M24
- M36 – the evaluation at M36 (to be set in D3.7.2)
- Summary of collected data and comments – comments backing up the evaluation in column M24.

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 in seven components, one for methods, one for the decision support system and five addressing the open-source IDE and runtime platform, and their evaluations at M24:

Is it Delivered?	M24 Yes/No/TSE	M36 Yes/No/TSE	Summary of collected data and comments
Methods	Yes		The MODAClouds methods are mainly offered by the design time tools. More specifically, the evaluation of the functional modelling tool has highlighted that the model driven approach offered by the project is supported almost completely and the evaluation of SPACE4Clouds has shown that even though there is space for improvement, we are able to support analysis of QoS properties and optimization.
Decision Support System	Yes		The DSS has just been released at M24 and it has been evaluated only internally by CA. An analysis of the fulfilment of

			its general objectives will be performed by M36.
Open Source IDE and run time environment for:			
- high level design	Yes		Preliminary evaluation results give an indication that the MODAClouds toolset supports high level design and analysis of multi-cloud applications.
- early prototyping	TSE		There is no specific indication in the evaluation about the actual capability of the platform to support early prototyping. This aspect will be considered at M36.
- semiautomatic code generation	Yes		At the moment the MODAClouds IDE supports semiautomatic generation of deployment and monitoring scripts.
- automatic deployment on multi-clouds	Yes		Multi-cloud deployment experiments successfully performed, but multi-cloud migration will be addressed in Y3.
- Guaranteed QoS	TSE		MODAClouds guarantees QoS through its SLA framework and its self-adaptation platform. Both components have been delivered at M24 and will be evaluated at M36.

4.2 MODAClouds sub-objectives

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		Currently we support Amazon, Flexiant, all IaaS supported by jCloud.

Support/Coverage of PaaS	TSE		Currently, CloudBees, Azure, Google App Engine.
Support/Coverage of SaaS	TSE		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		We do support public clouds as indicated above, as for hybrid clouds we focus on supporting cloud bursting. We do have a prototype that will be evaluated at M36.
Support for migration of legacy applications to the Cloud	Yes		SensApp as well as 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		Being part of DSS that has been released at M24, this feature is available but not evaluated by case studies at this stage.
Guidelines for risk analysis and decision support	Yes		Being part of DSS that has been released at M24, this feature is available but not evaluated by case studies at this stage.
New business models	Yes		Being part of DSS that has been released at M24, this feature is available but not evaluated by case studies at this stage.

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	Yes		This feature has been evaluated at part of SPACE4Clouds.

aspects			
Support for optimized matching of cloud environment with application characteristics	Yes		This feature has been evaluated at part of SPACE4Clouds.

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		This feature has been released for the first time at M24 and will be extensively evaluated at M36
Support for data and application replication on multiple providers.	TSE		This feature is currently supported in terms of ability to migrate data from one cloud to another and has been evaluated using a large dataset extracted from Twitter public data.

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		This feature is under development and will be evaluated at M36.

Chapter 5

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).	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.	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, Mohagheghi 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.	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.	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.	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 developed for the MODAClouds project.	The integration test should be successful including all tools and requirements (M36).
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.	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.	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. 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.	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.	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	1. Provide MODACLOUDS software solutions as open source.	1. At least 2 software packages released as open source (M36). 2. Number of communities on the

applications life-cycle.	2. Meta-model extensions available as open source.	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 due in M24:

<table>
<tr><td colspan="3">KPI Definition / Remarks</td><td>Actual value</td></tr>
<tr><td colspan="3">SO1: Number of supported IaaS >= 3 (>= 1 at M18; >=3 at M30)</td><td>>=3</td></tr>
<tr><td colspan="4">Amazon, Flexiant, all those supported by jCloud</td></tr>
<tr><td colspan="3">SO1: Number of supported PaaS >= 2 (>=1 at M18; >=2 at M30)</td><td>3</td></tr>
<tr><td colspan="4">CloudBees, Azure, Google App Engine (not all parts of the project support all the three at the moment)</td></tr>
<tr><td colspan="3">SO1: Number of Cloud design patterns >=5 (>=2 at M18; >=5 at M30)</td><td>4</td></tr>
<tr><td colspan="4">Provider adapter, Runtime Reconfiguration, External Configuration Store, Leader-Followers</td></tr>
<tr><td colspan="3">SO2: Definition of reference QoS metrics and their monitoring methods >= 3 (M12).</td><td>>=3</td></tr>
<tr><td colspan="4">Supported metrics are: CPU utilization, Throughput, Response time.</td></tr>
<tr><td colspan="3">SO2: Median quality prediction accuracy (evaluated in terms of the mean value of the metrics) at design time (30%, M24).</td><td>~0.75%</td></tr>
<tr><td colspan="4"><table>
<tr><th>Metric</th><th>Median Error</th><th>Application</th><th>Experiment</th></tr>
<tr><td>CPU Utilization</td><td>0.75%</td><td>OFBiz</td><td rowspan="2">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%</td></tr>
<tr><td>Throughput</td><td>0.76%</td><td>OFBiz</td></tr>
</table></td></tr>
<tr><td colspan="3">SO3: Median quality prediction accuracy (evaluated in terms of the mean value of the metrics) at run-time (30%, M24).</td><td>Between 9.19% and 18.40%</td></tr>
<tr><td colspan="4"><table>
<tr><th>Metric</th><th>Median Error</th><th>Application</th><th>Experiment</th></tr>
<tr><td rowspan="2">CPU Utilization</td><td>9.19%</td><td>OFBiz</td><td>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.</td></tr>
<tr><td>18.40%</td><td>OFBiz</td><td>Comparison between predicted CPU utilization from SDA and real runtime values. CPU utilization is taken from OFBiz server</td></tr>
</table></td></tr>
</table>

			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
Monitoring platform was tested on: Flexiant, Amazon EC2, Microsoft Azure, Heroku, OpenNebula, OpenStack, Eucalyptus.	
S03: All the business models of Cloud providers identified in SO1 are analysed (M24).	Yes
S06: 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 M36).	2
1 while paper describing general approach (paper at MISE) in M6 1 white paper on domain-specific (on-line withe paper prepared by ATOS) in M12	
SO7: The number of satellite workshops organised at international conferences >= 3 (=1 at M12; >=2 at M24; >=3 at M36)	3
MultiCloud April 2013 / Prague, MICAS September 2013 / Timisoara and MICAS September 2014 / Timisoara	
S07: Participation in at least one collaboration working group (M18).	2
1. CloudML collaboration group led by SINTEF. (involving PaaSage, Artist and MODAClouds projects)	

2. TOSCA standardization working group	
S07: Organising at least 3 industrial seminars (=1 at M24; >=3 at M36).	1
CA seminar in Feb 2014 "Supporting the Development and Operation of Multi-Cloud Applications"	

Chapter 6

6 Conclusions

This deliverable presents the initial outcomes of MODAClouds evaluation process. This report will be updated at M36 (D3.7.2). The report emphasis the status of each MODAClouds artefact versus the expected goals. A set of recommendations is given for each artefact in order to guide the development and validation during the last year of the project.

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).

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)

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