CloudDSF – The Cloud Decision Support Framework for Application Migration

Vasilios Andrikopoulos, Alexander Darsow, Dimka Karastoyanova, Frank Leymann

CloudDSF - The Cloud Decision Support Framework for Application Migration

Vasilios Andrikopoulos, Alexander Darsow, Dimka Karastoyanova, and Frank Leymann

IAAS, University of Stuttgart
Universitätsstr. 38, 70569 Stuttgart, Germany
{firstname.lastname}@iaas.uni-stuttgart.de

Abstract. Migrating existing applications to cloud solutions is a multi-dimensional problem that spans beyond technical issues and into the financial, security and organizational domains. The existing works in the field form a picture of a maturing but still incomplete research area, requiring the introduction of comprehensive solutions for the migration of enterprise systems and applications to cloud solutions. As part of this effort, in this work we focus on supporting decision makers in evaluating the need for migration, and guiding them along the decisions that need to be made before the actual migration process. For this purpose we build on existing work to provide an elaborated decision support framework that is available as a Web application. We discuss the evaluation of the framework by experts, identify its deficiencies and outline our future steps.

Keywords: Application migration; decision support; decision visualization

1 Introduction

Cloud computing has become increasingly popular over the last few years both with the industry and the academia. The main driving factors for this popularity as discussed e.g. in [16], are the ease of infrastructure provisioning, the cost savings due to the transfer from capital to operational expenses, and the potential for elastic resource utilization to cope with fluctuating demand. In this context, it is a key requirement for enterprises to migrate partially or completely their existing systems and applications to a cloud solution [6]. However, the migration of existing software to the cloud poses a number of challenges both technical as well as financial, legal and organizational [1]. In recent years a number of experience reports have started appearing discussing the migration of existing systems to cloud solutions, e.g. [7, 17, 22], illustrating in all cases the multi-dimensionality of the problem. In a recent publication, Jamshidi et al. [16] provide a systematic review of the State of the Art on methodologies, techniques, tooling support and research directions. Their conclusion is that the field is still at a formative stage, and that cross-cutting concerns like security and effort estimation are not being addressed sufficiently.

Along these lines, the work in [3] discusses the vision of a system that supports decision makers in deciding whether and how to migrate their applications to cloud solutions. Multiple decision points creating feedback loops with each other are identified and associated with tasks like cost analysis that not only depend on the decisions' outcome, but also affect these decisions in turn. However, the discussion in [3] stays on a high level and does not identify concrete decision outcomes that can be used in practice. This is a deficiency that we are addressing with this work.

More specifically, the contributions of this work can be summarized as follows:

– a publicly available Cloud Decision Support Framework (CloudDSF) which aims to assist decision makers during the migration of applications to the cloud, and
– an empirical evaluation of CloudDSF by a cohort of experts, together with a discussion on the steps required to realize a decision support system based on it.

The remaining of this paper is structured as follows: Section 2 summarizes the conceptual framework for cloud migration decision support discussed in [3]. Section 3 discusses the process of elaborating this framework into CloudDSF, the identified decisions and outcomes, and the visualization of CloudDSF as a Web application offering different interaction options to its users. Section 4 discusses the process of evaluating CloudDSF by means of a survey and summarizes the findings of the survey. Consequently, Section 5 provides a discussion on the aspects that need to be addressed in order to realize a decision support system based on CloudDSF. Finally Sections 6 and 7 close this paper with related work, and conclusions and future work, respectively.

2 Background

Figure 1 summarizes the vision for a Decision Support System for Cloud Migration as discussed in [3]. Two types of concepts are presented in the figure: *decision points* that identify high-level decisions that need to be made, and *tasks* that need to be performed in order to support these decisions. Four major decision points are identified:

1. *How to distribute the application* in logical and physical placement terms. This entails viewing the application as a set of components across different functional layers, and deciding which components are to be hosted in one or more cloud providers.
2. *Which is the elasticity strategy* that the application needs to implement in order to cope with current and future demand in the face of service level agreements (SLAs) and performance requirements of its users.
3. *What are the requirements of the application with respect to multi-tenancy*, i.e. to what extent the existing application is required to support resource sharing across different levels ranging from the bare metal to the application instance, to what degree it is designed for this purpose, and how it should be (re-)engineered to support multi-tenancy.
4. *How to select an appropriate (cloud) service provider and offering* that fits the application needs in terms of cost, expected performance, compliance and security requirements etc.

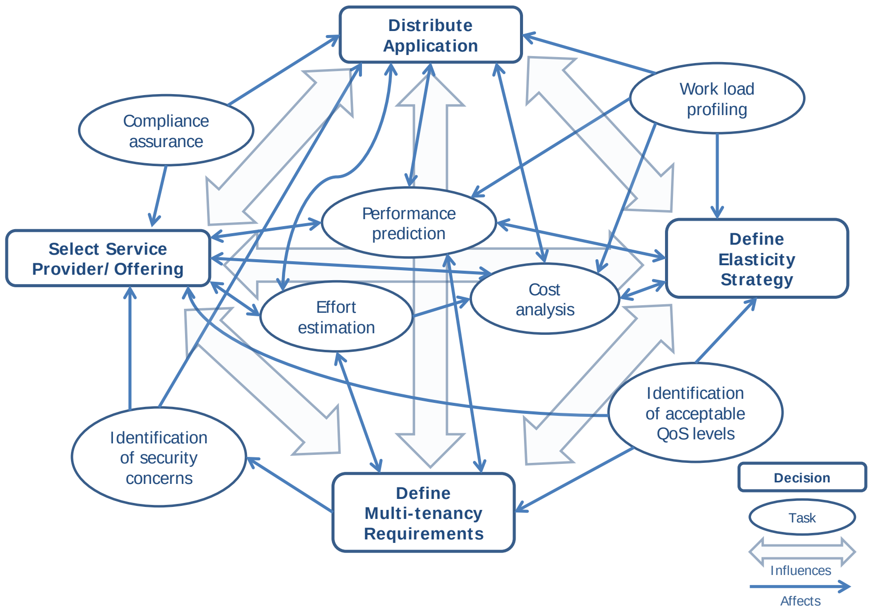

Fig. 1. The Decision Support Framework for Cloud Migration [3]

Relations between these decision points are illustrated with transparent block arrows in Fig. 1. In addition, the set of tasks that were identified in [3] to be related to these decisions points are:

a) *workload profiling* of the application,
b) *performance prediction* based on the workload profile of the application,
c) *effort estimation* for any necessary adaptations to the application during the migration process,
d) *cost analysis* that builds on the pay-per-use model of the cloud services, including the cost for the estimated adaptation effort,
e) *identification of acceptable QoS levels* in order to cope with existing or future SLAs,
f) *compliance assurance* with respect to organizational regulation, and national and international law, and
g) *identification of security concerns* with emphasis on critical data communication and storage.

Tasks and decision points therefore form a network of relationships with decisions made on one point, e.g. which elasticity strategy to use, affecting directly or indirectly all other decision points. However, and as mentioned in the introductory section, it can be easily observed that the level of granularity of the identified decision points in [3] is too coarse in order to be connectible to concrete decision outcomes that enable decision making in the field. Toward this goal, the remaining of this work presents our proposal for a cloud decision support framework for application migration based on elaborating the one discussed above.

3 CloudDSF

Following Power's classification of decision support systems (DSS) [23], the realization of the vision outlined in [3] means the development a *knowledge-driven DSS*. Such systems focus on suggesting and recommending actions to the decision maker referring to the gathered knowledge about the problem domain. As a first step towards such a system, in the following we present our proposal for a *Cloud Decision Support Framework (CloudDSF)* which gathers and visualizes knowledge from the migration domain.

For this purpose, and by using the work in [3] as the starting point, we conducted a thorough literature review focusing on the areas of decision support for application migration to the cloud e.g. [5, 8, 14, 18, 21], and application migration and cloud computing [4] in general, with the explicit goal of verifying and elaborating the already identified decision points. The gathered knowledge was initially captured in a spreadsheet which recorded potential (concrete) *decisions* affiliated with each *decision point*, their possible *outcomes*, as well as the relevant literature sources[1]. The result of this process serves as the *Knowledge Base* of CloudDSF, based on which a *Visualization* component was developed to enable interaction with users. Both of these components are presented in the following.

3.1 Knowledge Base

The Knowledge Base of CloudDSF capturing the results of the elaboration process is summarized in Table 1. A total of 17 decisions are identified for the existing 4 decision points (application distribution, elasticity strategy, multi-tenancy requirements and provider selection). For each decision, a set of outcomes is provided for a total of more than 50 outcomes, ranging from very specific, e.g. which pricing model is offered by the service provider (*Select Pricing Model*), to more coarse, e.g. whether a single or multiple components are to be hosted in the cloud (*Select Application Components)*. More specifically, the identified decisions are:

Application Distribution Four decisions are associated with application distribution. The first two follow Fowler et al.'s distinction between physical tiers and functional layers in the application architecture [9]. They are concerned with which application layer(s) and which application tier(s) are to be moved to the cloud (*Select Application Layer* and *Select Application Tier*, respectively). More than one layers or tiers at a time are of course also possible. The next decision (*Select Application Components*) becomes essential if finer granularity than a layer or tier is required

[1] The same process also resulted in the refinement of tasks, as well as the identification of additional tasks to be considered. However, for reasons of space we omit the discussion on task elaboration and postpone it for future work.

Table 1. The CloudDSF Knowledge Base

Decision Point	Decision	Outcomes
Application Distribution	Select Application Layer	– Presentation/Business/Data Layer – Multiple Layers
	Select Application Tier	– Client/Application/Data Tier – Multiple Tiers
	Select Application Components	– Single Component – Multiple Components
	Select Migration Type	– Type I, II, III or IV
Elasticity Strategy	Define Scalability Level	– Instance/Container/VM/Virtual Resource/ Hardware Level – Multiple Levels
	Select Scaling Type	– Vertical/Horizontal Scaling – Hybrid Scaling
	Select Elasticity Automation Degree	– Manual Scaling – Semi-automatic Scaling – Automatic Scaling
	Select Scaling Trigger	– Event-driven – Proactive
Multi-Tenancy Requirements	Select Kind of Multi-Tenancy	– Multiple Instances Multi-Tenancy – Native Multi-Tenancy
	Select Multi-Tenancy Architecture	– Any of the Possible Combinations
Provider/ Offering Selection	Select Cloud Deployment Model	–Private/Community/Public/Hybrid Cloud
	Select Cloud Service Model	– S/P/IaaS
	Define Cloud Hosting	– On Premise/Off Premise – Hybrid Hosting
	Define Roles of Responsibility	– Ownership/Operation/Management Role – Any Combination of Roles
	Select Pricing Model	– Free/ Pay-per-Use/-Unit/Subscription – Combined Model
	Select Cloud Vendor	– Evaluated Vendor
	Define Resource Location	– Evaluated Physical Resource Location

and refers to deciding which specific application component(s) are to be migrated. Finally, the last decision (*Select Migration Type*) refers to the type of migration to be used for the migration using the classification of Andrikopoulos et al. [1]: Type I — component(s) replacement, Type II — partial migration of functionality, Type III — full application stack migration (virtual machine-based), or Type IV — cloudification of the application. As such, the outcome of this decision may have an effect on the possible outcomes of all other decisions concerning the distribution of the application.

Elasticity Strategy The first decision concerning elasticity strategy refers to the scalability level required for the application (*Define Scalability Level*). For this purpose we discern between three different system levels: the physical *hardware* one, the *virtualization* level built on top of it, and the *application* level on top of that.

The virtualization level itself can be distinguished into *virtual resources*, e.g. the hypervisor used, and *virtual machines* allocated to these virtual resources. Similarly, the application level discerns between the application *instances* and the middleware *container* hosting these instances (e.g. application servers or database management systems). The scalability options for each level increase as we traverse the levels: while on the physical level there are only few choices (i.e. bringing in another server or updating the existing one with more powerful hardware), on application level potentially unlimited application instances can be added to cope with additional demand. The remaining decisions refer to the type of scaling that can be used i.e. vertical (adding/removing computational resources), horizontal (adding/removing instances or replicas) or hybrid combinations [28] (*Select Scaling Type*), how much automation is achievable by existing solutions [25] (*Select Elasticity Automation Degree*), and which type of trigger is used (*Select Scaling Trigger*): the more common event-driven type based on monitoring rules, or a proactive one which combines log files with real-time data to dynamically predict scaling actions [27].

Multi-Tenancy Requirements Following Guo et al.'s [13] classification of multi-tenancy, there are two kinds of multi-tenancy in cloud applications (*Select Kind of Multi-tenancy*): *multiple instances*, where each application tenant (application user) is working on a dedicated application instance, or *native* multi-tenancy where tenants share a single application instance and its underlying resources. Furthermore, and looking at the different system levels previously discussed (hardware, virtualization, application), different possibilities appear on which resources are to be shared between tenants. An application for example may rely on a single hardware and virtualization level for all tenants, but provide multiple instances for each tenant on the level of application or database management server (middleware). One database schema with separate tenant data spaces could be used, or different tenants may even share the same tables in the database. As a result, multiple combinations are available as outcomes of this decision (*Select Multi-Tenancy Architecture*).

Provider/Offering Selection Deciding on which cloud provider and which particular offering to be used depends fundamentally on the type of cloud solution that is appropriate for the enterprise. Towards identifying this solution we consider a series of decisions based on the definition of the different cloud solution types as provided by NIST (as updated and extended in [4]). In particular, it needs to be decided which of the private, public, community or hybrid deployment models is more suitable (*Select Cloud Deployment Model*), which of the **aaS*, that is Software, Platform or Infrastructure as a Service delivery models fits the application migration (*Select Cloud Service Model*), as well as where the application is to be hosted: on premise (in-house on e.g. a private cloud solution) or off premise (*Define Cloud Hosting*).

Depending on the deployment, delivery and hosting decisions, different responsibility roles are available to be distributed between application stakeholders and cloud providers (*Define Roles of Responsibility*). In principle there are three fundamental roles to be performed [4]: resource owner (i.e. to whom the infrastructure belongs to), resource operator (i.e. who hosts and makes sure that the application is running) and resource manager (i.e. who is responsible for managing the resources, rolls out

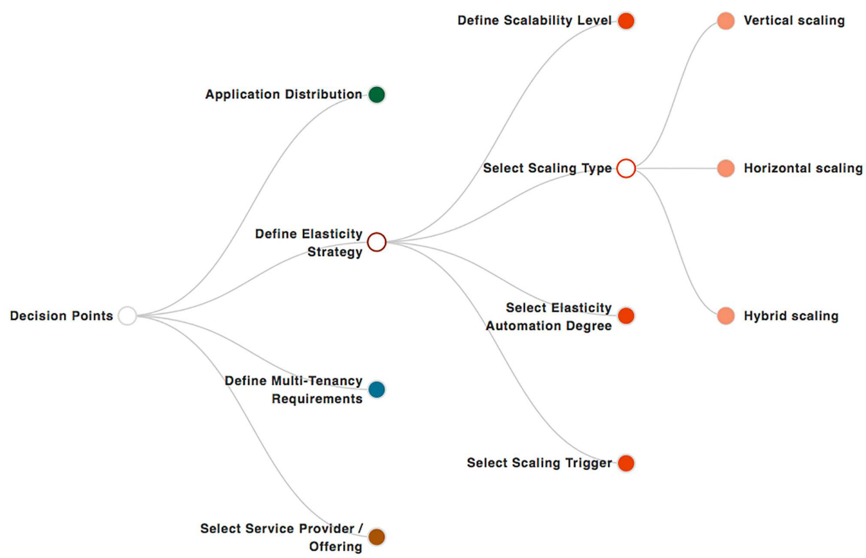

Fig. 2. CloudDSF Visualization Options: Tree View

updates etc.) Any combination of these roles is possible; for example in an off premise private cloud scenario, the enterprise may be the resource owner but not the resource operator or manager (roles to be performed by the cloud provider).

While cloud computing is usually associated with the "pay as you go" pricing model of utility-style charging for resource consumption [4], in practice many cloud offerings use alternative pricing models [27]. For this purpose we adapt the pricing model classification discussed by Suleiman et al. [27] to distinguish between free, pay-per-use, pay-per-unit (of time), or subscription-based charging, e.g. for reserved instances offered by the Amazon EC2 service[2] (*Select Pricing Model*). Finally, the remaining two decisions (*Select Cloud Vendor* and *Define Resource Location*) are concerned with the evaluation of the reputation of the cloud provider in the manner discussed e.g. in [12], and of the physical location of the migrated data for regulatory compliance purposes [10] (*Select Cloud Vendor* and *Define Resource Location*, respectively).

3.2 Visualization

Aiming for a modern, user friendly, cross-platform visualization of CloudDSF we opted for a Web-based solution which we made publicly available at `http://www.clouddsf.com`. The resulting Web application uses basic Web technologies like HTML, CSS and SVG in order to provide an interface to the users through their browsers. The decision points, decisions and possible outcomes, together with the identified tasks and their

[2] Amazon Elastic Cloud Computing (EC2): `http://aws.amazon.com/ec2/`

Fig. 3. CloudDSF Visualization Options: Partition View

relations, as discussed in the previous section are stored in a JSON[3] file, which is used as input for the visualization logic. The latter is written in Javascript, based on the D3[4] and jQuery[5] libraries that offer out of the box a number of features like network graph visualization and dynamic graph manipulation. The Bootstrap framework[6] was used for layout rendering purposes.

As a result, a series of hierarchical and network-based visualization options are available at `http://www.clouddsf.com`. Figures 2 and 3 show an example for each one of these options: a (hierarchical) tree view of the possible outcomes of the "Select Scaling Type" decision (Fig. 2), and a (network) partition view of all the decisions (Fig. 3) which reconfigures the ring to zoom to a decision point or to a decision and its outcomes when selected. Cluster, treemap and partition layouts are also offered to the user, allowing for different types of interaction with the application.

4 Evaluation

Given the fact that CloudDSF is meant to be part of a knowledge-driven DSS, evaluating the framework focuses on the *understandability*, *suitability* and *completeness* of

[3] JavaScript Object Notation (JSON): `http://www.json.org/`
[4] D3.js — Data Driven Documents: `http://d3js.org/`
[5] The jQuery library: `http://jquery.com/`
[6] Bootstrap: `http://getbootstrap.com/`

the decision points, decisions and outcomes in its knowledge base. In the following we present the evaluation procedure towards these objectives and report on our findings.

4.1 Procedure

The instrument of a Web-based survey by means of questionnaire was used for the evaluation of CloudDSF. More specifically, a questionnaire of 86 questions was designed following the guidelines discussed in [19], combining open (free text responses) and closed (choice from an ascending rating scale) questions towards the identified evaluation objectives (understandability, suitability and completeness). The questionnaire was then realized using the open source survey application LimeSurvey[7] and made available online for a period of two weeks in February 2014. A group of academic researchers, system developers, operations managers and IT consultants with expertise in cloud computing were invited by email to participate and provided with the credentials to access the survey. A completion rate of 42,9% was achieved among the participants of the survey for a total of 6 completed surveys (consisting of 4 academic and 2 industrial participants). The results of the survey including the posed questions — after discarding incomplete (unfinalized) questionnaires and anonymizing the participants — are also available online[8]. Only completed questionnaires are considered in the following discussions on the findings of the survey.

4.2 Findings

Figures 4, 5 and 6 summarize the results of the conducted survey with respect to the objectives of understandability and relevance (suitability)[9]. With respect to the former, it can be seen from Fig. 4 that the understandability of the decision points ('Overall' in Fig. 4) is perceived as good from the survey participants, while the (average) understandability of individual decisions per decision point is actually ranked higher. This is expected since the finer granularity of the decisions in the CloudDSF allows for better understanding of the decisions involved. In terms of suitability, Fig. 5 shows that the vast majority of decisions are deemed as relevant to the decision point that they are associated with, with only the application distribution decisions rated lower than the rest. Furthermore, in Fig. 6 all decision points with the exception of elasticity strategy are viewed as highly relevant for migration to the cloud, with elasticity strategy perceived as simply relevant. In terms of completeness, an average of around two additional decisions was pointed out by the survey participants as relevant for each decision point. The provided suggestions will be used in the future for further improving and elaborating CloudDSF.

5 Discussion

The evaluation of CloudDSF as discussed in the previous section showed that the framework contains a suitable and representative amount of knowledge for guiding

[7] LimeSurvey: `http://www.limesurvey.org`

[8] `http://www.clouddsf.com/survey/questionnaire_results.zip`

[9] *Note:* In all figures the scale is from worse (0) to best (4).

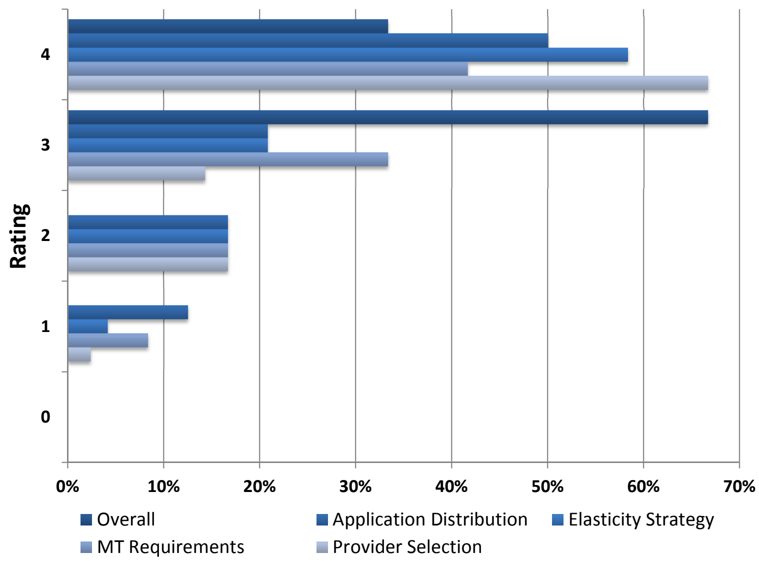

Fig. 4. Understandability of Decisions per Decision Point (Average)

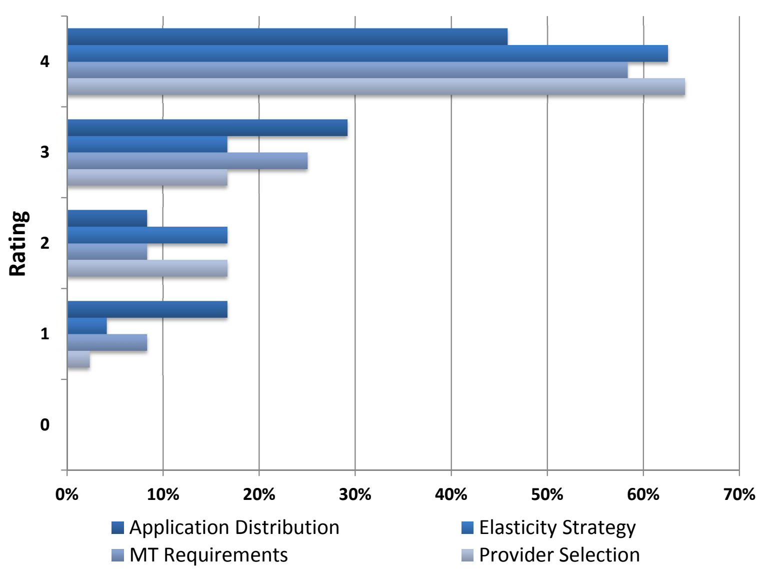

Fig. 5. Relevance of Decisions per Decision Point (Average)

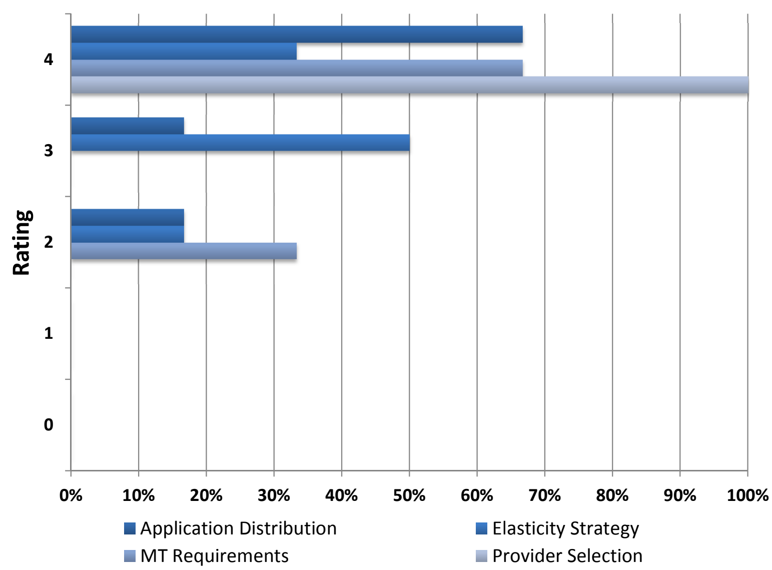

Fig. 6. Relevance of Decision Points

decision makers during the migration of applications to the cloud. One aspect of the accumulated knowledge that was not discussed sufficiently however in the previous section is that of the relationships between decisions, and between decisions and tasks. A number of relationships are already identified in Fig. 1 distinguished between two types: what is the *influence* of a decision type to another, and how a task *affects* the decisions at a decision point (and vice versa). The elaboration of the decision points of [3] into concrete decisions in CloudDSF results naturally into a significant increase in the number of these relationships, as shown by the cluster diagram of Fig. 7 (using one of the available visualization options in CloudDSF).

As shown in Fig. 7, there is a strong interplay between decisions (and decisions and tasks) forming a dense network of relationships with each other. In turn, this means that selecting any of the available outcomes for each decision has a direct or indirect impact to the possible outcomes in other decisions. Furthermore, the result of any of the identified tasks may end up constraining significantly the available outcomes across many of the related decisions, as well as potentially contradicting already taken decisions. A formal representation of these relationships that allows this type of reasoning on them is therefore required for the realization of the knowledge-driven decision support system discussed in [3]. The other major task is the linking between the decisions to be taken and the representation of the enterprise applications to be migrated to the cloud in order to translate decision outcomes into concrete actions. This linking is anyway required for most of the tasks (e.g. cost calculation) associated with the decisions. These two tasks are part of our future steps.

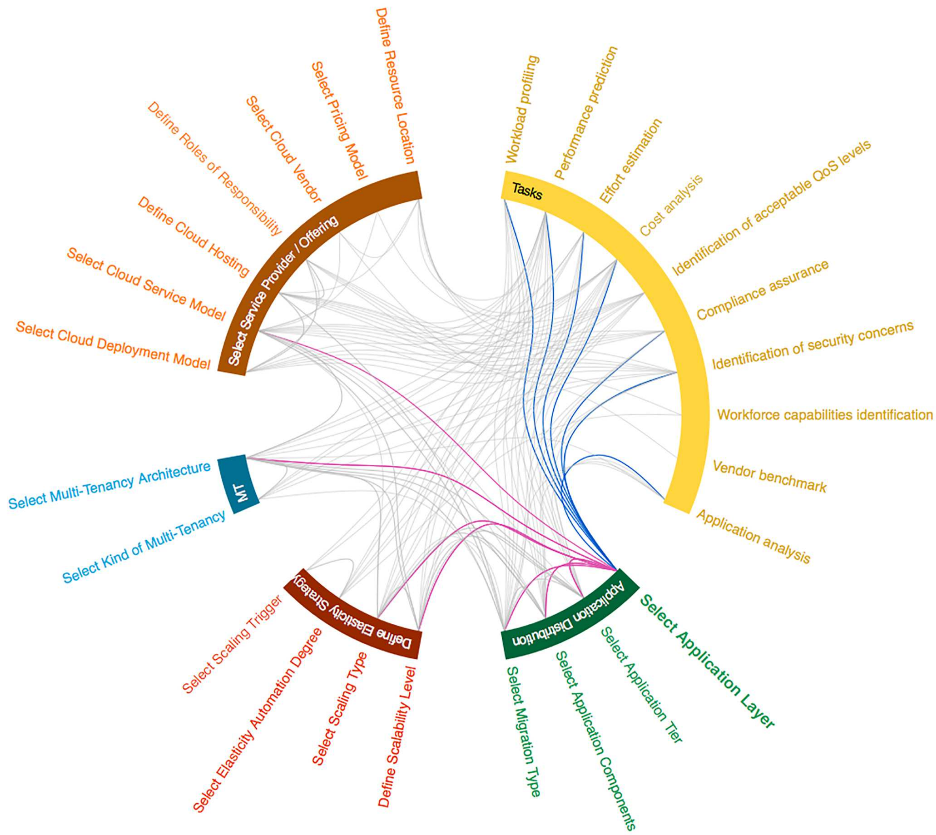

Fig. 7. Cluster View of the Decisions and Tasks in CloudDSF

6 Related Works

A series of works on decision support for the migration of applications to the cloud have appeared in the literature. The Cloudward framework [14], for example, was developed in collaboration between academic and industrial partners with the goal of migrating applications to hybrid cloud solutions. The framework takes into account cost savings, communication costs, transaction delays and constraints like security. CloudGenius [21] provides an automated decision making process towards identifying the optimal selection of a combination of IaaS offerings and VM image types for single-tier web application migration. The focus of the system is however on technical requirements and does not address organizational or other enterprise concerns. The Cloud Adoption Toolkit [18] provides a framework specifically aimed at enterprise stakeholders. For this purpose it provides the means for tasks like technology suitability analysis based on the profile of the enterprise, cost modeling and energy consumption analysis for the "to be" model of the migrated systems, as well as responsibility modeling distinguishing between operation, maintenance and management roles for migrated and non-migrated system components. These tasks are meant to be performed in a sequential manner, forming a decision making process.

In a similar fashion, the Cloudstep [5] approach provides a decision process consisting of nine activities including enterprise, legacy application and cloud provider profiling, constraint identification analysis and alternative migration scenarios evaluation and ranking. Constraints that are taken into consideration are categorized in seven areas: financial, organizational, security, communication, performance, availability and suitability. Finally, Chauhan and Babar present in [8] a high level seven step process built on best practices and lessons learned from the migration of legacy application to service-oriented architectures. As discussed in Section 3, the process of elaborating the decision points identified in [3] was based on these works.

A related research field to decision support for application migration is cloud service selection based on multi-criteria decision making (MCDM) techniques. Approaches like [15, 24, 26] and [11] provide the means to users to provide a set of requirements against which existing cloud service offerings are evaluated and ranked. Similarly, in [20], a resolution engine is presented that matches user-provided criteria with available offerings from a cloud service marketplace in order to identify suitable business-level offerings. In [12], the authors discuss a feature-based model for the description of service offerings that can be used for decision making towards offering selection. These works can be used as the basis for the implementation of a decision support system around CloudDSF.

7 Conclusion

While cloud computing has been increasingly successful with adoption by the industry, the migration of existing systems to cloud solutions has proven to be a more complicated problem than cloud vendors and proponents would admit. A big part of the problem is the multi-dimensionality required to deal with a series of technical, financial, legal and organizational issues. Towards supporting enterprise stakeholders in deciding whether and how to migrate their applications, in previous work we outlined the vision for a decision support system that incorporates different aspects of migration. The current work expanded on this vision to propose CloudDSF, a framework in which knowledge about the problem domain (i.e. migration to the cloud) was gathered, organized, visualized and offered as a publicly available Web application. The empirical evaluation conducted in collaboration with experts on cloud computing, while limited in scope, showed that the resulting framework is sound and suitable for migration-oriented decision making. However, in order to translate it into a full blown decision support system there are still important issues to be addressed.

In terms of future work the focus is on the tasks already identified in the previous sections, i.e. the incorporation of the evaluation results, the formalization of the relationships between decisions, and decisions and tasks, as well as the linking from the application model to the decisions that need to be taken. Part of our plans is also a larger scale evaluation of CloudDSF with a wider profile of participants, as well as additional evaluation by means of case studies in industrial migration projects. To this purpose, we need also to work on the elaboration of the relationships between tasks and decisions, and the connection between decisions outcomes and inputs and outputs of tasks. Cost calculation facilities provided e.g. by the Nefolog system [2] will be used as the pilot for this effort.

Acknowledgment

This work is partially funded by the FP7 EU-FET project 600792 ALLOW Ensembles.

References

1. Andrikopoulos, V., Binz, T., Leymann, F., Strauch, S.: How to Adapt Applications for the Cloud Environment. Computing 95(6), 493–535 (2013)
2. Andrikopoulos, V., Reuter, A., Mingzhu, X., Leymann, F.: Design Support for Cost-efficient Application Distribution in the Cloud. In: Proceedings of the 7th IEEE International Conference on Cloud Computing (CLOUD 2014) (to appear). IEEE Computer Society (2014)
3. Andrikopoulos, V., Strauch, S., Leymann, F.: Decision support for application migration to the cloud: Challenges and vision. In: 3rd International Conference on Cloud Computing and Service Science (CLOSER'13). pp. 149–155. SciTePress (2013)
4. Badger, L., Grance, T., R., P.C., Voas, J.: Cloud computing synopsis and recommendations - recommendations of the national institute of standards and technology. NIST Special Publication 800-146 (2012)
5. Beserra, P.V., Camara, A., Ximenes, R., Albuquerque, A.B., Mendonca, N.C.: Cloudstep: A step-by-step decision process to support legacy application migration to the cloud. In: Maintenance and Evolution of Service-Oriented and Cloud-Based Systems (MESOCA 2012) Workshop. pp. 7–16. IEEE (2012)
6. Buyya, R., Broberg, J., Gosćinśki, A.: Cloud computing: principles and paradigms. Wiley (2011)
7. Chauhan, M.A., Babar, M.A.: Migrating service-oriented system to cloud computing: An experience report. In: International Conference on Cloud Computing (CLOUD 2011). pp. 404–411. IEEE (2011)
8. Chauhan, M.A., Babar, M.A.: Towards process support for migrating applications to cloud computing. In: 2012 International Conference on Cloud and Service Computing. pp. 80–87. IEEE Computer Society (2012)
9. Fowler, M., et al.: Patterns of Enterprise Application Architecture. Addison-Wesley Professional (November 2002)
10. Gagliardi, F., Muscella, S.: Cloud Computing–Data Confidentiality and Interoperability Challenges, pp. 257–270. Springer (2010)
11. Garg, S., Versteeg, S., Buyya, R.: Smicloud: A framework for comparing and ranking cloud services. In: Utility and Cloud Computing (UCC), 2011 Fourth IEEE International Conference on. pp. 210–218. IEEE (2011)
12. Gudenkauf, S., Josefiok, M., Goring, A., Norkus, O.: A reference architecture for cloud service offers. In: Enterprise Distributed Object Computing Conference (EDOC), 2013 17th IEEE International. pp. 227–236. IEEE (2013)
13. Guo, C., Sun, W., Huang, Y., Wang, Z., Gao, B.: A Framework for Native Multi-Tenancy Application Development and Management. In: Proceedings of CEC/EEE'07. pp. 551–558. IEEE (2007)

14. Hajjat, M., Sun, X., Sung, Y., Maltz, D., Rao, S., Sripanidkulchai, K., Tawarmalani, M.: Cloudward bound: planning for beneficial migration of enterprise applications to the cloud. In: ACM SIGCOMM Computer Communication Review. vol. 40, pp. 243–254. ACM (2010)
15. Hussain, O.K., Hussain, F.K., et al.: IaaS cloud selection using MCDM methods. In: e-Business Engineering (ICEBE), 2012 IEEE Ninth International Conference on. pp. 246–251. IEEE (2012)
16. Jamshidi, P., Ahmad, A., Pahl, C.: Cloud Migration Research: A Systematic Review. IEEE Transactions on Cloud Computing 1(2) (2013)
17. Khajeh-Hosseini, A., Greenwood, D., Sommerville, I.: Cloud migration: A case study of migrating an enterprise it system to iaas. In: International Conference on Cloud Computing (CLOUD 2010). pp. 450–457. IEEE (2010)
18. Khajeh-Hosseini, A., Greenwood, D., Smith, J.W., Sommerville, I.: The cloud adoption toolkit: supporting cloud adoption decisions in the enterprise. Software: Practice and Experience 42(4), 447–465 (2012)
19. Krosnick, J.A., Presser, S.: Question and Questionnaire Design, pp. 263–314. Emerald (2010)
20. Menychtas, A., Gatzioura, A., Varvarigou, T.: A business resolution engine for cloud marketplaces. In: Cloud Computing Technology and Science (CloudCom), 2011 IEEE Third International Conference on. pp. 462–469. IEEE (2011)
21. Menzel, M., Ranjan, R.: Cloudgenius: decision support for web server cloud migration. In: Proceedings of WWW '12. pp. 979–988. ACM (2012)
22. Pahl, C., Xiong, H., Walshe, R.: A comparison of on-premise to cloud migration approaches. In: Service-Oriented and Cloud Computing (ESOCC 2013), pp. 212–226. Springer (2013)
23. Power, D.J.: Decision Support Systems: A Historical Overview, pp. 121–140. Springer Berlin Heidelberg (2008)
24. ur Rehman, Z., Hussain, F.K., Hussain, O.K., et al.: Towards multi-criteria cloud service selection. In: Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2011 Fifth International Conference on. pp. 44–48. IEEE (2011)
25. Sallam, A., Li, K.: Virtual machine proactive scaling in cloud systems. In: Cluster Computing Workshops (CLUSTER WORKSHOPS), 2012 IEEE International Conference on. pp. 97–105. IEEE (2012)
26. Saripalli, P., Pingali, G.: Madmac: Multiple attribute decision methodology for adoption of clouds. In: Cloud Computing (CLOUD), 2011 IEEE International Conference on. pp. 316–323. IEEE (2011)
27. Suleiman, B., Sakr, S., Jeffery, R., Liu, A.: On understanding the economics and elasticity challenges of deploying business applications on public cloud infrastructure. Journal of Internet Services and Applications 3(2), 173–193 (2012)
28. Vaquero, L., Rodero-Merino, L., Buyya, R.: Dynamically scaling applications in the cloud. ACM SIGCOMM Computer Communication Review 41(1), 45–52 (2011)