

## A Multi Criteria Recommendation Engine Model for Cloud Renderfarm Services

Ruby Annette J<sup>1</sup>, Aisha Banu W<sup>2</sup>, Subash Chandran P<sup>3</sup>

<sup>1,3</sup>Department of Computer Science, SSE, Saveetha University, India

<sup>2</sup>Department of Computer Science and Engineering, B.S Abdur Rahman University, India

---

### Article Info

#### Article history:

Received Dec 21, 2017

Revised Mar 15, 2018

Accepted Aug 10, 2018

---

#### Keyword:

Cloud renderfarm services

Engine model

Quality of Service (QoS)

Simple Additive Weighting

---

### ABSTRACT

Cloud services that provide a complete platform for rendering the animation files using the resources in the cloud are known as cloud renderfarm services. This work proposes a multi criteria recommendation engine model for recommending these Cloud renderfarm services to animators. The services are recommended based on the functional requirements of the animation file to be rendered like the rendering software, plug-in required etc and the non functional Quality of Service (QoS) requirements like render node cost, time taken to upload animation files etc. The proposed recommendation engine model uses a domain specific ontology of renderfarm services to identify the right services that could satisfy the functional requirements of the user and ranks the identified services using the popular Multi Criteria Decision Analysis method like Simple Additive Weighting (SAW). The ranked list of services is provided as recommended services to the animators in the ranking order. The Recommendation model was tested to rank and recommend the cloud renderfarm services in multi criteria requirements by assigning different QoS criteria weight for each scenario. The ranking based recommendations were generated for six different scenarios and the results were analyzed. The results show that the services recommended for each scenario were different and were highly dependent on the weights assigned to each criterion.

*Copyright © 2018 Institute of Advanced Engineering and Science.  
All rights reserved.*

---

### Corresponding Author:

Ruby Annette. J,

Department of Information Technology,

B.S. Abdur Rahman University,

Chennai, Tamilnadu, India.

Email: rubyannette\_it\_phd\_2013@bsauniv.ac.in

---

## 1. INTRODUCTION

Artists in 3D studios describe their animations in scene files. In order to view the animation described in the scene file, rendering software is used to read the scene file and draw images that are representative of each frame described in the scene file. The process of rendering a scene file is computationally expensive. Typically, a scene file will contain many frames. Fortunately, the individual frames of a scene file can be processed in parallel. Accordingly, many 3D studios use computer clusters, commonly referred to as render farms, to compute multiple frames simultaneously.

The individual processors in the render farms are often referred to as render nodes. Many 3D studios have their own in-house render farms and they render their scene files on their own render farms as much as possible. However, in some cases, due to the sheer volume of scene files that must be rendered and/or in order to meet a particular deadline, the 3D studios will out source at least some of their scene file rendering to a cloud based rendering provider. Furthermore, many smaller 3D studios do not have their own render farms due to the expense of establishing and maintaining a render farm and these smaller 3D studios usually rely solely on third party rendering providers such as the cloud-based rendering providers. The Cloud based

render farm services [ 1-6 ] are basically delivered either as an Infrastructure-as-a-Service and Platform-as-a-Service. The IaaS delivery model enables the user to rent the required computing power like the compute unit, bandwidth, storage space etc on an hourly basis and pay only for the resources that had been used. Leading players include Amazon EC2, Rackspace. Amazon EC2 supports Cloud based rendering through Axceleon's Cloud Fusion.

The PaaS type of cloud renderfarm offers a complete infrastructure to the animators which include the compute unit, software licenses of the software required like the rendering software , the rendering job management tools etc. The PaaS based cloud render farm services also takes care of the license required for the software and plug-ins are used during the rendering process and the charges are usually included in the render node charges like in case of the Rebus farm®, RevUpRender. Some work have been done to rank the cloud IaaS services in general [7-15], however they are not based on the perspective of a cloud based render farm which may focus on features specific to animation file rendering like the GPU based IaaS services etc. No work have been done on recommending the PaaS type of cloud render farm services as per our knowledge except few [16,17] and this invention is focused on recommending the IaaS and PaaS type of Cloud render farm services.

A method to recommend the cloud render farm services is very essential as the number of cloud render farm services are mushrooming at a higher speed and the rates are becoming competitive. As the service has to meet multiple criteria like the functional requirements and the non functional requirements, the animators find it difficult to identify the right service provider in less time. A recommendation engine that is specific to cloud renderfarm services which could recommend the services that satisfy both the functional and the multi criteria non functional QoS requirements of the users in minimum duration will be very beneficial to the animators. It is an object of this work to provide a method of recommending cloud based render farm services that satisfies multiple criteria and overcomes at least some of these problems.

## **2. MULTI CRITERIA RECOMMENDATION ENGINE**

### **2.1. Multi Criteria Recommender Systems**

The systems that are based on the Multi Criteria Decision Making methods for generating the recommendations are called the Multi Criteria Recommender Systems. In these types of Multi Criteria Recommender Systems, the recommendation itself is coined as a Multi Criteria Decision Making problem and an appropriate MCDM method is chosen to rank all the alternatives from the best one to the worst. The user is provided this list of ranked items as the recommendations. Many works have focused on the detailed analysis and classification of these Multi Criteria Recommender Systems [ 18-22]. Many others have applied these MCDM methods to develop recommender systems specific to their domain and have been successful.

Some of the popular MCDM methods include Analytical Hierarchical Processing (AHP), Simple Additive Weighted Method (SAW), and Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS), ELimination and Choice Expressing Reality (ELECTRE) etc and this work proposes to use the MCDM method called SAW for ranking the cloud renderfarm services. The cloud renderfarm services are ranked using the SAW method based on the multiple criteria requirements of the animators in the form of a ranked list and the service with the highest rank is recommended as the best service that could satisfy both the functional and the non functional requirements of the animators.

### **2.2. Multi Criteria Recommendation Engine Architecture**

The Cloud Renderfarm Recommendation Engine (CRR) is made up of two main modules namely the search engine module and the recommendation module. The search engine module enables the animators to enter their functional requirements that the cloud render farm services should satisfy like the render engine supported, plug-ins required etc. Once the services that satisfy these conditions are filtered, then these list of services are further ranked and filtered based on the multiple QoS criteria selected by the animator in the recommendation module and the ranked list of services is provided as the recommended services. A detailed explanation of the process involved in both these modules is given below.

### **2.3. Search Engine Module**

The search engine contains the cloud renderfarm services ontology which is created by filtering the information obtained from the web portals of the services providers. The Service Profile Interface in the search engine collects the details about the services provided like the functional requirements offered by them and the non functional Quality of Service (QoS) values promised by the service provider. The service details obtained are stored in the services registry. Whereas the QoS values promised are stored in the QoS Registry. The Requirements Gathering Interface collects the functional and the non functional Quality of Service (QoS) requirements of the animators and stores them in the Requirements Registry. The match maker

component of the search engine matches the animators functional requirements with the service details provided by the service provider and identifies the service providers who could satisfy the functional requirements of the animators.

The functional requirements of the PaaS cloud renderfarm service providers are very different from the traditional IaaS type of services and are very domain specific. The examples of the functional requirements of PaaS cloud renderfarm services are like the Render Engine Software supported, Supported software license required, supported animation modeling Software, Plug-in supported etc. Similarly the components involved in calculating the price of a render node is also very different from the IaaS type of services as they are calculated as an aggregate of the following components cost like the Render node unit, outgoing data, Job management application, hyper threading, incoming data, S/w license, Technical support etc.

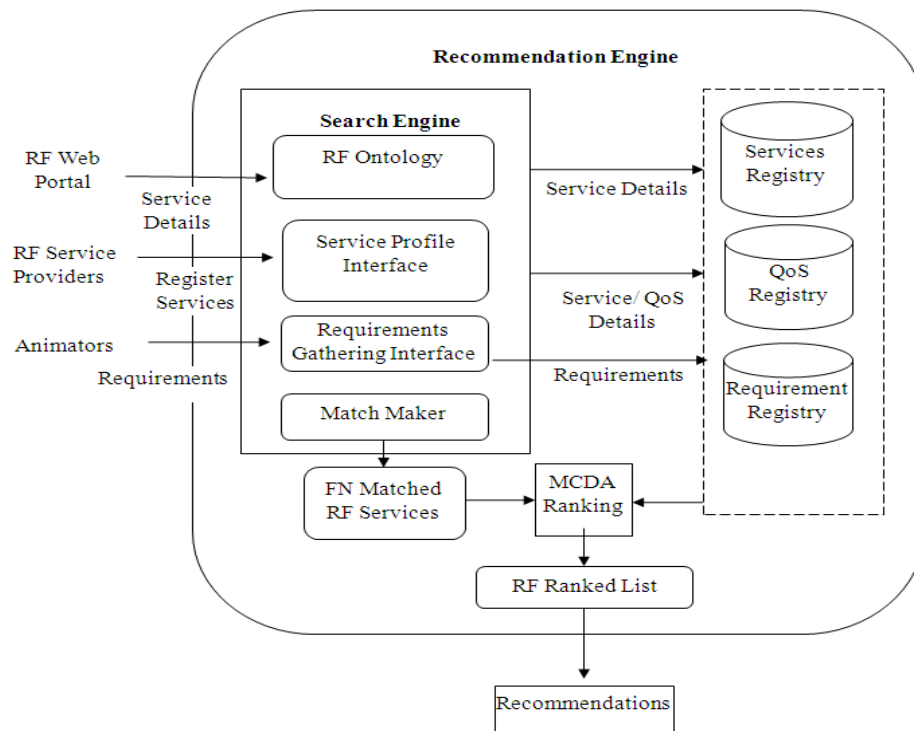


Figure 1. Multi criteria recommendation engine architecture

### 2.3. Multi Criteria Recommendation Module

The second main component of the proposed recommendation engine is the recommendation module. The main objective of this multi criteria recommendation engine is to compare the services based on multiple QoS criteria and recommend the right PaaS cloud renderfarm services to the animators. To achieve this objective the Multi Criteria Decision Analysis (MCDA) method named SAW (Simple Additive Weighting) has been used to rank the services based on the promised QoS values and the ranked list is provided as the recommendations to the animators.

### 2.4. SAW Method of Ranking

The Simple Additive Weighting (SAW) Method is considered the simplest and the clearest method as it is not computationally expensive and has been used widely in many domains. In this method, as a first step, a Decision Matrix (D) is constructed to quantify the values of the attributes or the criteria for each alternative service selected for ranking. Then a Normalized Decision Matrix (D') is derived by calculating the normalized rating ( $r_{ij}$ ) for both the benefit criteria and the worst criteria based on the Equations 2 and 3 respectively as given below. A criteria or attribute is considered as a benefit criteria, only if higher the value of the attribute higher is the benefit to the user. For example, .Performance, availability etc. Whereas, a criteria or attribute is considered as a worst criteria, if lower the value of the attribute is considered the best.

For example, Cost, Service Response Time etc. Finally the overall score (Si) is calculated by assigning the importance weights (wj) to the attributes as given in Equation 1 for alternative (N) with (M) number of QoS attributes.

$$S_i = \sum_{j=0}^M w_j r_{ij} \tag{1}$$

For, i=1, 2, 3... N

Where: r<sub>ij</sub> - normalized rating,

'i' - ith alternative

'j' - jth criterion

w<sub>j</sub> - jth criterion weight.

The benefit criteria value of r<sub>ij</sub> is calculated using the equation 2 as given below.

$$r_{ij} = x_{ij} / \max_i(x_{ij}) \tag{2}$$

Similarly, the worst criteria value of rij is calculated using the equation 3 as given below.

$$r_{ij} = \min_i(x_{ij}) / x_{ij} \tag{3}$$

Where: x<sub>ij</sub> - original value of jth criterion

**2.5. Numerical Example for Applying SAW to Rank Cloud renderfarm services**

The Simple Additive Weighting (SAW) Method of Ranking first calculates the weighted sum average of all the QoS attributes selected to compare the renderfarm services. The four QoS attributes (X j) selected for ranking the cloud renderfarm services in this work are the Service Response Time (X1), File Upload Time (X2), render node cost (X3) and Elasticity (x4). A decision matrix D has been constructed using the attribute values provided by each service provider as given in Table 1. Next, the normalized decision matrix D' is constructed based on the equation 2 and equation 3 given above depending on the nature of the attribute that is whether it is a benefit criteria or a worst criteria as given in Table 2. Finally, the evaluation score for every renderfarm service which has been filtered by the match maker is determined as the product of the relative importance weights (wj) and the normalized value of the criteria for each service using the Equation 1 given above. The product value is then aggregated and the renderfarm services are recommended based on the aggregated score. The cloud renderfarm service with the highest score is recommended as the first best cloud renderfarm service recommended and so forth.

Table 1. Decision Matrix D for ranking Renderfarms

	X1	X2	X3	X4
RF1	0.62	8	9	0.4
RF2	0.63	1.5	8	0.7
RF3	0.62	15	12	0.05
RF4	0.63	7	6	0.9
RF5	0.62	11	10	0.1

Where: X1 - Service Response Time

X2 - File Upload Time

X3 – Render node Cost

X4 - Elasticity

RF – Renderfarm

The normalized decision matrix D' evaluated using the Equation 3 is given below. Equation 3 has been used for calculation since all the attributes considered in this work like cost are of worst criteria category for which, the lower the value is preferred. The weight (W) assigned by the user for each attributes in six different scenarios is given below. W (S1) represents the weights assigned by the animator in Scenario 1.

$$W(S1) = [ 0.3 \ 0.2 \ 0.2 \ 0.3 ]$$

$$W(S2) = [ 0.1 \ 0.1 \ 0.7 \ 0.1 ]$$

$$W(S3) = [ 0.7 \ 0.1 \ 0.1 \ 0.1 ]$$

$$W(S4) = [ 0.1 \ 0.7 \ 0.1 \ 0.1 ]$$

$$W(S5) = [ 0.1 \ 0.1 \ 0.1 \ 0.7 ]$$

$$W(S6) = [ 0.2 \ 0.3 \ 0.3 \ 0.2 ]$$

Where: W - Weight assigned to each criteria or attribute  
 S(n) - Scenario analyzed

Table 2. Normalized Matrix D' for Ranking Renderfarms

	X1	X2	X3	X4
RF1	1	0.19	0.67	0.25
RF2	0.984	1	0.75	0.142
RF3	1	0.1	0.5	2
RF4	0.98	0.214	1	0.111
RF5	1	0.136	0.6	1

In the next step, the evaluation score for all six scenarios is calculated using the Equation 1 as given above and the values are aggregated to get the overall rank of the cloud based renderfarms. The ranking obtained is then sorted in the descending order and the cloud renderfarm service with the highest score is recommended as the first best cloud renderfarm service recommended and so forth as given in Table 3.

Table 3. Ranking of Cloud Renderfarm Services

	S1	S2	S3	S4	S5	S6
RF1	0.547	0.613	0.811	0.325	0.358	0.508
	Rank #5	Rank #4	Rank #4	Rank #5	Rank #4	Rank #5
RF2	0.6878	0.737	0.878	0.8876	0.3728	0.7502
	Rank #3	Rank #2	Rank #1	Rank #1	Rank #3	Rank #2
RF3	1.02	0.48	0.78	0.42	1.56	0.78
	Rank #1	Rank #5	Rank #5	Rank #2	Rank #1	Rank #1
RF4	0.5701	0.8305	0.8205	0.3589	0.2971	0.5824
	Rank #4	Rank #1	Rank #3	Rank #3	Rank #5	Rank #4
RF5	0.7472	0.6338	0.8736	0.3552	0.8736	0.6208
	Rank #2	Rank #3	Rank #2	Rank #4	Rank #2	Rank #3

### 3. RESULTS AND DISCUSSION

From the ranking details of cloud renderfarms given in Table 3 and the individual ranking for each scenario given in Figures 1-6, it could clearly be seen that the recommended cloud renderfarm services is different in each scenario. From the ranking details given above, it could be inferred that the RF3 is an all round performer as it has been Ranked #1 three times for scenarios S1, S5 and S6. Moreover, the RF2 service provider is ranked best, if the main criteria are SRT and File upload time. However, RF4 is considered best if the main criterion is file upload time. Though RF5 has not been ranked first for any scenario, it could still be considered as an overall performer as it has been ranked #2 for three times for S1, S3 and S5. The overall evaluation score and rank of RF1 is very less and it has been mostly ranked in fourth and fifth position for all scenarios.

As this multi criteria recommendation model uses the ranked list to recommend the cloud renderfarm services, for an animator with multiple criteria requirements as in case of scenario1, the proposed recommendation engine would recommend the cloud renderfarm services in the order of RF3, RF5, RF2, RF4, RF1, where RF3 is the best recommended service that could satisfy both the functional and the non functional QoS requirements of the animators. Similarly, for an animator who is looking for a cost effective. Renderfarm services, the recommended services in their respective order are RF4, RF2, RF5, RF1, RF3, Where RF4 is recommended as the most cost effective service that could satisfy the requirements of the animator. Recommending the services based on a single criteria with relatively higher importance than others is easier than recommending services where all criteria have near to similar weights as in the case of Scenario 1 and 6, the proposed recommendation engine model effectively deals with these scenarios and provides better recommendations in less time as given in table 3. Thus the proposed recommendation engine effectively reduces the time taken to discover a cloud renderfarm service that could satisfy both the functional and the non functional requirements of the animators.

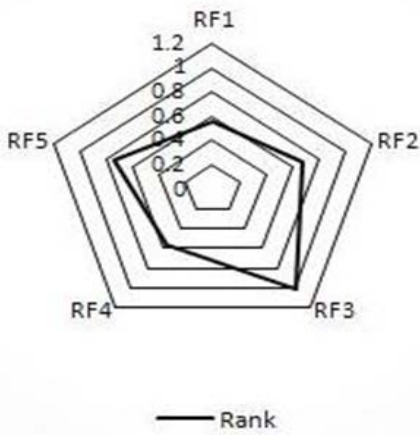


Figure 1. Ranking for Scenario 1

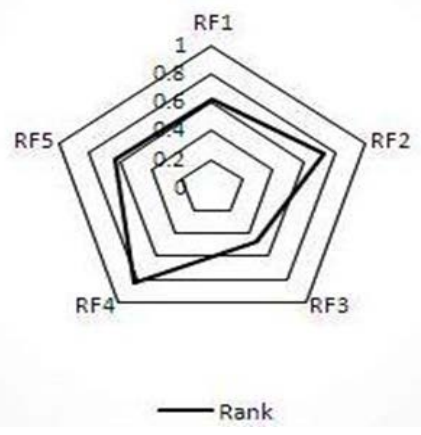


Figure 2. Ranking for Scenario 2

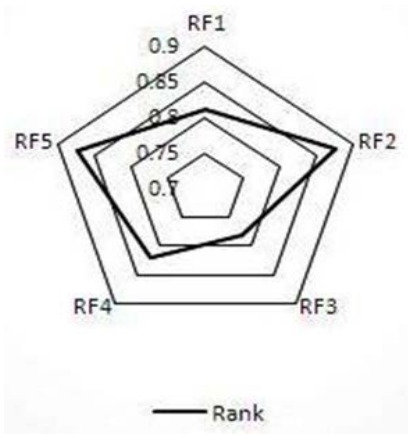


Figure 3. Ranking for Scenario 3

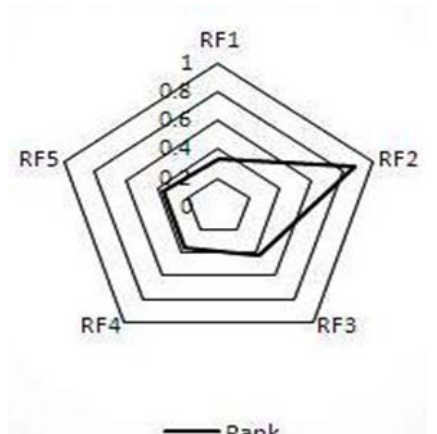


Figure 4. Ranking for Scenario 4

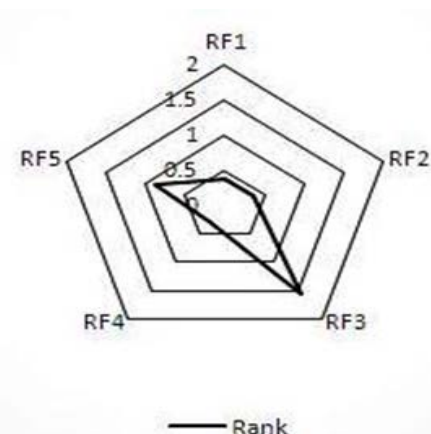


Figure 5. Ranking for Scenario 5

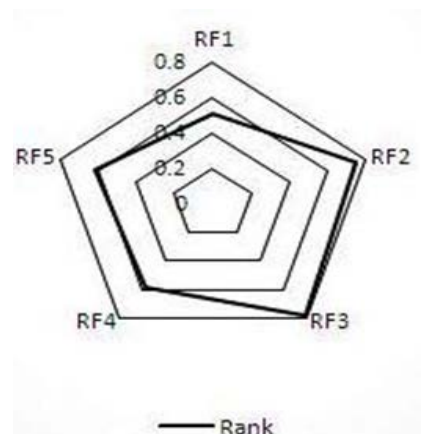


Figure 6. Ranking for Scenario 6

#### 4. CONCLUSION

The proposed work has identified the functional characteristics and the non functional characteristics of the cloud renderfarm services for recommending the right service to the animators. The services that suit the functional requirements of the animators are filtered using ontology of renderfarm services and multi criteria decision making algorithm called SAW has been used to rank the services based on the promised Qos details provided by the service provider. The ranking method has been applied to

recommend services in six different scenario and the results are quite promising as these MCDA methods are proven mathematical models used to rank items based on multiple criteria. As a future work, more QoS criteria would be identified or cut down in order to increase the efficiency and the accuracy of cloud render farm services.

## REFERENCES

- [1] James Kennedy, Philip Healy, A method of provisioning a Cloud-based renderfarm, EP2538328 A1 [Patent]
- [2] James Kennedy, David Healy, A method of rendering a scene file in a Cloud-based render farm, EP2538330 A1 [Patent].
- [3] Carroll, Martin D., Ilija Hadzic and William A. Katsak, "3D Rendering in the Cloud, Bell Labs Technical Journal 17.2 (2012): 55-66.
- [4] Srinivasa, Ganapati, et al, Runtime prediction framework for CPU intensive applications, U.S. Patent No. 7,168,074. 23 Jan. 2007.
- [5] Cho, Kyungwoon, et al, Render Verse: Hybrid Render Farm for Cluster and Cloud Environments, Control and Automation (CA), 2014 7th Conference on. IEEE, 2014.
- [6] Weini, Zhou, et al, A New Software Architecture for Ultra-large-scale Rendering Cloud, Distributed Computing and Applications to Business, Engineering & Science (DCABES), 2012 11th International Symposium on. IEEE, 2012.
- [7] S. K. Garg, et al., "Smicloud: A framework for comparing and ranking cloud services," In proc. of the 2011 Fourth IEEE International Conference on Utility and Cloud Computing (UCC), pp. 210-218, 2011.
- [8] S. WANG, et al., "Cloud model for service selection," In Proc. of the 2011 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHP), pp. 666-671, 2011.
- [9] F. K. Hussain and O. K. Hussain, "Towards Multi-Criteria Cloud Service Selection," In Proc. of the 2011 Fifth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), pp. 44-48, 2011.
- [10] A. Li, et al., "CloudCmp: comparing public cloud providers," In Proc. of the 10th annual conference on Internet measurement, pp. 1-14, 2010.
- [11] A. Li, et al., "CloudCmp: shopping for a cloud made easy," USENIX HotCloud, 2010.
- [12] A. Li, et al., "Comparing public-cloud providers," Internet Computing, IEEE, Vol. 15, pp. 50-53, 2011
- [13] TANI, Hicham GIBET, and Chaker EL AMRANI. "Cloud Computing CPU Allocation and Scheduling Algorithms using CloudSim Simulator." International Journal of Electrical and Computer Engineering (IJECE) 6.4 (2016).
- [14] Geerthik, S., S. Venkatraman, and Rajiv Gandhi. "AnswerRank: Identifying Right Answers in QA system." International Journal of Electrical and Computer Engineering (IJECE) 6.4 (2016).
- [15] Jena, Tamanna. "Disaster Recovery Services in Intercloud using Genetic Algorithm Load Balancer." International Journal of Electrical and Computer Engineering (IJECE) 6.4 (2016).
- [16] Ruby Annette and Aisha Banu. W. Article: A Service Broker Model for Cloud based Render Farm Selection. International Journal of Computer Applications 96(24):11-14, June 2014.
- [17] Annette, J. Ruby, W. Aisha Banu, and P. Subash Chandran. "Rendering-as-a-Service: Taxonomy and Comparison." Procedia Computer Science 50 (2015): 276-281.
- [18] Liu, Duen-Ren, and Ya-Yueh Shih. "Integrating AHP and data mining for product recommendation based on customer lifetime value." Information & Management 42.3, pp. 387-400, 2005.
- [19] Manouselis, Nikos, and Constantina Costopoulou. "Analysis and classification of multi-criteria recommender systems." World Wide Web 10.4, pp. 415-441, 2007.
- [20] Nguyen, H., Haddawy, P.: DIVA: applying decision theory to collaborative filtering. In Proc. AAAI Worksh. Recomm. Syst., Madison, WI, July 1998.
- [21] Schaefer, R.: Rules for Using Multi-Attribute Utility Theory for Estimating a User's Interests. In Proc. ABIS Worksh. 'Adaptivität und Benutzermodellierung in interaktiven Softwaresystemen', Dortmund, Germany, October 2001.
- [22] Adomavicius, Gediminas, Nikos Manouselis, and YoungOk Kwon. "Multi-criteria recommender systems." Recommender systems handbook. Springer US, pp.769-803, 2011.