Performance Predictions for Adaptive Cloud-Based Systems using FMC-QE

Stephan Kluth
FOM University of Applied Sciences
Berliner Allee 7, 30175 Hannover
Email: mail@stephankluth.de

***Abstract*—This paper presents an application of an analytical performance prediction approach to (self-)adaptive cloud-based systems. The methodology called Fundamental Modeling Concepts for Quantitative Evaluation (FMC-QE) uses three perspectives to hierarchically describe the performance behavior of a system. This methodology is now applied to cloud-based systems with infinite and parallel server capacities and further, the ideas are ported to the world of (self-)adaptive systems. Furthermore, a case study is shown as an example of such models.**

Keywords-FMC-QE; cloud; adaptive.

I. INTRODUCTION

Cloud technology is more and more used to process parts of every kind of business process [1] [2]. This paper shows a way to performance modeling and performance predictions in these distributed scenarios. Especially the performance predictions of the Fundamental Modeling Concepts for Quantitative Evaluation (FMC-QE) [3] [4] called FMC-QE Tableau could be used to implement algorithms for (self-)adaptive cloud-based workflow handlers. In this paper, this methodology is applied to cloud-based systems with infinite and parallel server capacities and further, the ideas are ported to the world of (self-)adaptive systems. These ideas are furthermore shown in a case study as an example of such models.

In the following, the methodology FMC-QE is shortly described as a repetition in Section II. Afterward, some related work is described in Section III. Then the linkage to (self-)adaptive systems is given in Section IV along with an example in Section V. Finally, some conclusions and future work are described in Section VI.

II. BACKGROUND: FMC-QE

FMC-QE [3]–[7] is a performance modeling and analysis methodology in which the systems are modeled from the perspective of the hierarchical service requests based in FMC [8]. In FMC-QE, these hierarchies are the key to complexity in the modeling and evaluation of complex systems. Furthermore, the complexity is reduced through three different modeling perspectives, the service request structures, the server structures, and the dynamic behavior including the control flow. Furthermore, the service requests are modeled as a tuple of value and unit like physical units. This enables hierarchical service request transformations through the Forced Traffic Flow Law [9].

The hierarchical service request structures are the entry point of the modeling in FMC-QE. An exemplary service request structure, modeled in FMC-QE in Entity-Relationship-Diagrams, is shown in Figure 1. Here one service request is decomposed into two sub-requests, an initialization, and an execution.

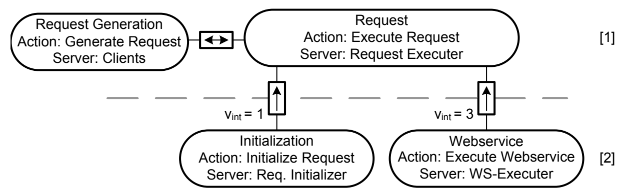

Figure 1. Service Request Structures [7].

In the modeling of the server structures, there is a distinction between logical and real servers with parallelism on every hierarchical layer. This enables the modeling of complex software systems running on distributed and shared hardware. The server structures in FMC-QE are modeled in Block Diagrams, as shown in Figure 2. In the example, there is one application server and an infinite number of web servers available (e.g., in a cloud-based approach).

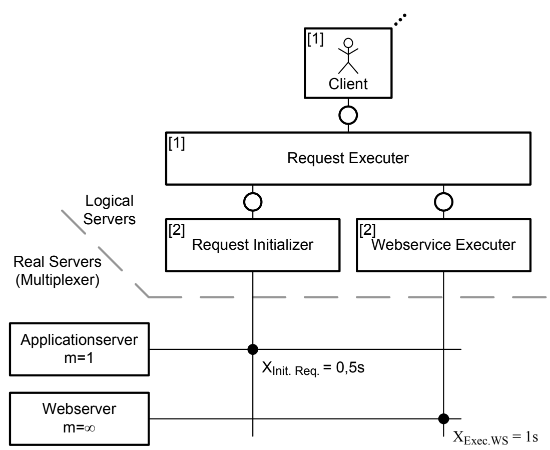

Figure 2. Server Structures [7].

The dynamic behavior and the control flows are modeled in Petri Nets [10]. This allows the modeling of parallel,

 ISBN: 978-1-61208-781-8

TABLE I. FMC-QE TABLEAU EXAMPLE [7].

Experimental Parameters:	
n_{ges}	30
λ_{bott}	2,0000
f	0,8000
λ	1,6000

Service Request Section							Server Section					Dynamic Evaluation Section						
[bb]	$SRq_i^{[bb]}$	$p_{[bb-1],i}$	$v_{i,ext}^{[bb-1]}$	$v_{i,int}^{[bb]}$	$v_i^{[bb]}$	$\lambda_i^{[bb]}$	$Server_i$	$X_{i,measured}^{[bb]}$	$m_{i,ext}^{[bb-1]}$	$m_{i,int}^{[bb]}$	$m_i^{[bb]}$	$X_{i,mpxed}^{[bb]}$	$\mu_i^{[bb]}$	$\rho_i^{[bb]}$	$n_{i,q}^{[bb]}$	$n_{i,s}^{[bb]}$	$n_i^{[bb]}$	$R_i^{[bb]}$
2	Webservice	1	1	3	3	4,8000	Webserver	1,0000	1	1	1	1,0000	1,0000		0,0000	4,8000	4,8000	1,0000
2	Initialization	1	1	1	1	1,6000	App. Server	0,2000	1	1	1	0,5000	2,0000	0,8000	3,2000	0,8000	4,0000	2,5000
1	Request	1	1	1	1	1,6000			1	1	1		2,0000		3,2000	5,6000	8,8000	5,5000
1	Request Generation	1	1	1	1	1,6000			1	1	1	13,2500	0,0755		0,0000	21,2000	21,2000	13,2500

Multiplexer Section				
$Multiplexer_j$	m_j	$X_j^{[1]}$	$\mu_j^{[1]}$	$\mu_j^{[1]} * m_j$
App. Server	1	0,5000	2,0000	2,0000
Webserver	∞	1,0000		

serial, branch, loop, and synchronization structures. The complexity of the state-space is further reduced through the distinction of operational and control states [11]. An exemplary FMC-QE Petri Net is illustrated in Figure 3.

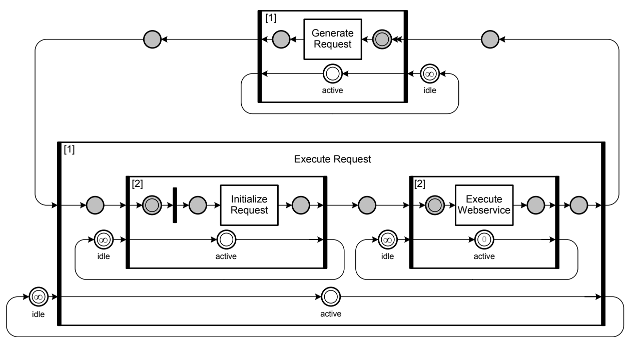

Figure 3. Dynamic Behavior and Control Flow [7].

The performance values of the modeled system are predicted in the FMC-QE Tableau. This hierarchical balance sheet is based on Little's Law [12] for relations within a hierarchical layer (horizontal) and the Forced Traffic Flow Law [9] for relations among layers (vertical). It extracts the performance parameters from the model and calculates predictions. The different system- and load-parameters are easy to change to compute a broad range of possible configurations. Table I shows an exemplary FMC-QE Tableau.

III. RELATED WORK

This work relates to the Palladio approach [13]. While the focus in Palladio is on simulation, here the focus is on numerical analysis. This approach could be integrated into Palladio to provide another numerical analysis beside Queuing Petri Nets (QPN) and Layered Queuing Networks (LQN) as described in [14] and shown in Figure 4. FMC-QE also uses ideas from LQN [3] and therefore cooperation in this field would be interesting to benefit from each other.

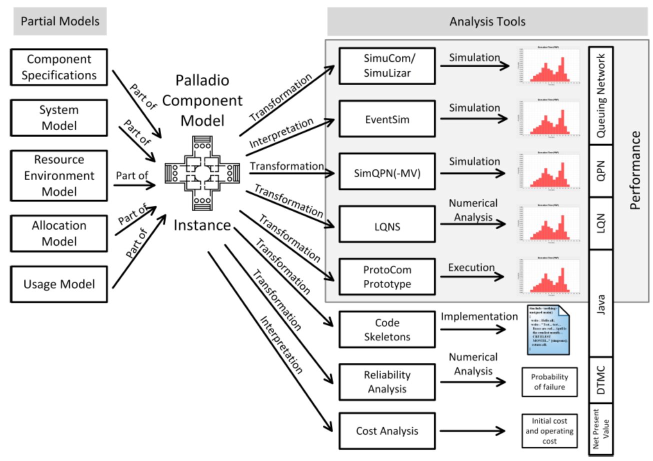

Figure 4. Palladio overview [13], [14].

As in earlier papers described [7], FMC-QE also uses model transformations to transform the non-hierarchical system models into the strictly hierarchical FMC-QE approach. In very new contributions of Palladio, similar transformations are described [15] and shown in figure 5.

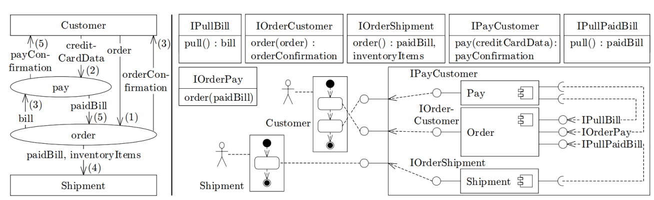

Figure 5. Palladio Transformations [15].

In contrast to this representation, FMC-QE uses the hierarchical service request as the central perspective and precise or approximated calculations as described in [16] and in section IV. Nevertheless, there are similarities in the hierarchical decomposition of the service request in FMC-QE [7] and the data flow transformations from Data Flow Models (DFM) into Palladio Component Models (PCM) in Palladio [15].

Furthermore, there are numerous simulation approaches

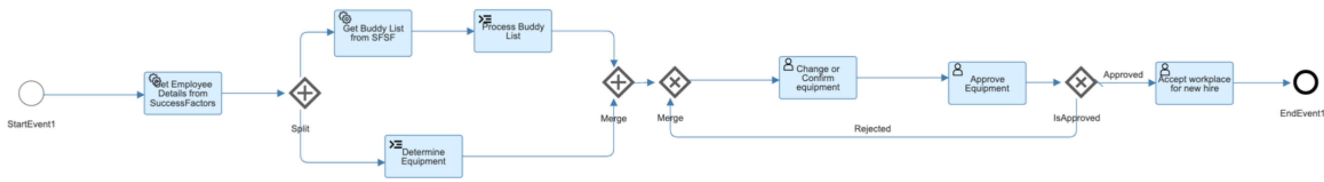

Figure 6. Example SAP Cloud Platform Workflow [17]

like [18]. FMC-QE delimits from these approaches, as it is based on numerical analysis. Nevertheless, comparison case studies of simulations and numerical analysis (in special FMC-QE) are very interesting as already done in [7].

IV. Applying FMC-QE to (Self-)Adaptive Cloud-based Systems

FMC-QE could be used to predict the performance behavior of the system processing the modeled business process. Therefore, the business process is transformed into the hierarchical request structure of FMC-QE [7] and if there are no inter-server control flows (and therefore, the underlying system is of type Product-Form-Network (PFN) [19] [16]) FMC-QE could predict the performance in a very fast (no simulation) and precise (if PFN) way. If the business process and the underlying system is not of type PFN, FMC-QE could provide approximations [16].

Business processes processed on cloud-based systems or systems, where parts of the whole business process are processed on cloud-based systems are the ideal use case for FMC-QE, because the processing unit could often be assumed as an infinite server as shown in Figure 7. This is the case if the closed Service-Level-Agreement is of type: *No matter how many requests will come, we will respond in x sec per request.*

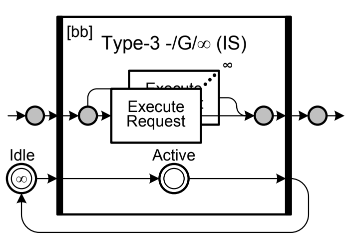

Figure 7. Infinite Server [3]

If the closed Service-Level-Agreement is of type: *We will provide x compute units each with a speed of y.*, the underlying system (multiplexer) could be assumed as a parallel server with exponentially distributed service times as shown in Figure 8 or with other service time distributions such as deterministic [3].

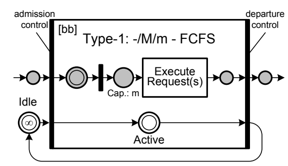

Figure 8. Infinite Server [3]

Another advantage of FMC-QE is, that the methodology is based on hierarchical service requests as shown in Figure 1. This reduces the complexity of the whole calculations and could therefore be implemented quite easily.

The performance predictions of FMC-QE could then be taken to adaptively (self-)adjust the allocated cloud computation performance based on the actual number of service requests or to compute a broad range of possible load scenarios to be aware of performance adaptations in the future.

V. Case Study

The approach was applied to an SAP case study [17], shown in Figure 6 with an example of ordering a notebook for a new employee.

In this section, the workflow of Figure 6 is transformed into the three perspectives of FMC-QE and the computations of the performance predictions in the FMC-QE Tableau. For other implementations of (self-)adaptive systems, this is not essential. The algorithms of the self-adaptive systems could just use the ideas of FMC-QE in terms of hierarchical service requests and the corresponding performance predictions [3] [16].

A. Dynamic Behavior and Control Flow (Petri Net)

In Figure 9, the dynamic behavior of the workflow is shown. The main difference between the original workflow [17] designed as BPMN-Diagram [20] [21] is the transformation to four hierarchical levels.

B. Service Request Structure and Static Structure

The corresponding service request structures of the example are shown in Figure 10. The service request is partitioned into the same 4 hierarchical levels. In this diagram the increased traffic flow realizing the retransmission of the *Equipment Negotiation Request* ($v_{int} = 1, 2 \frac{[OrderNotebookfornewEmployee-Request]}{[EquipmentNegotiationRequest]}$) is visualized.

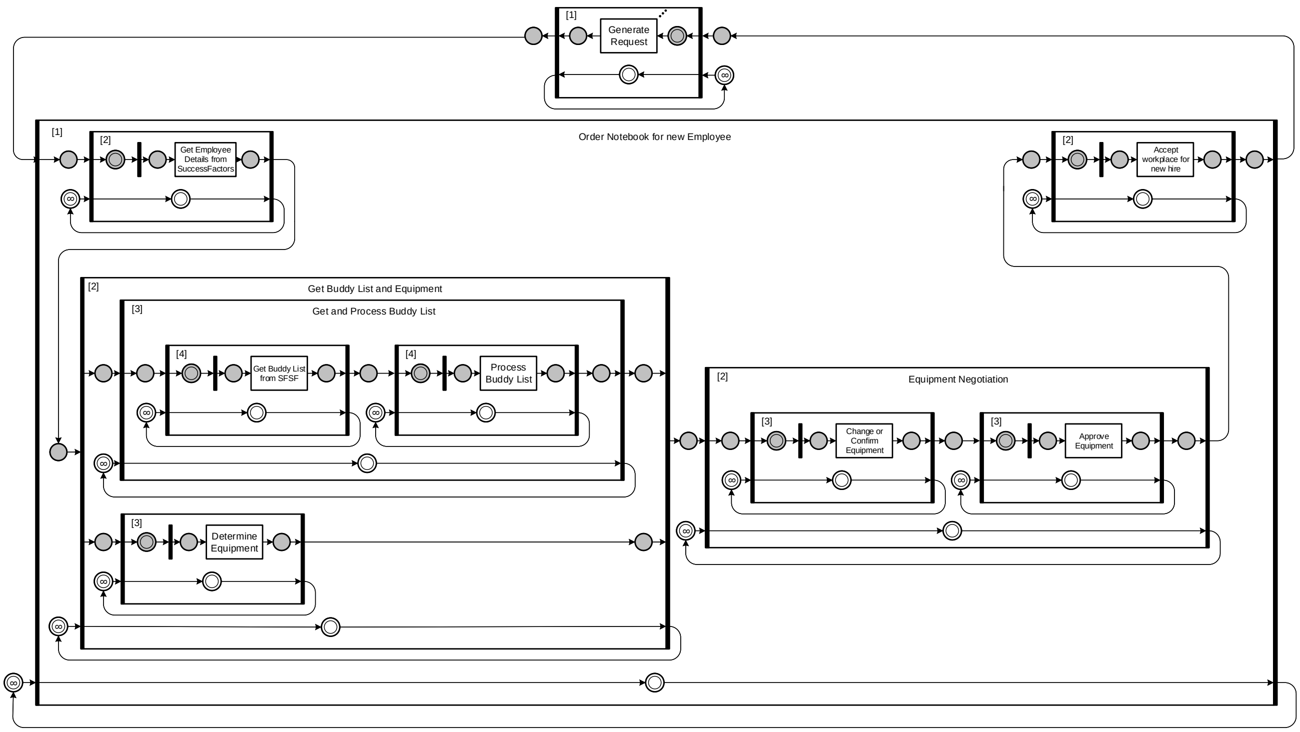

Figure 9. Case Study - Petri Net.

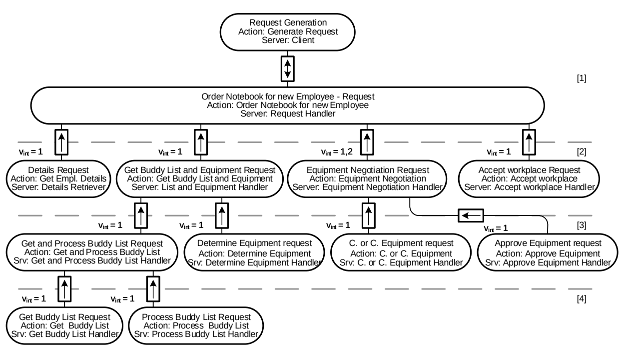

Figure 10. Case Study - Service Request Structures.

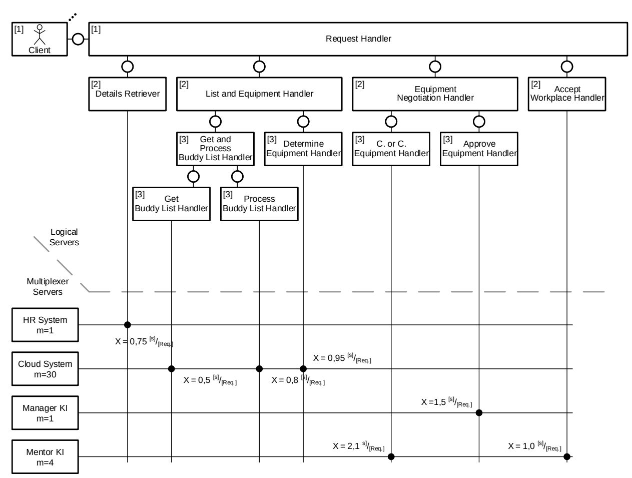

Figure 11. Case Study - Server Structures.

C. Static Structure

In the third diagram of the model, the server structures are represented in the Block Diagram, as shown in Figure 11. In this model, the four hierarchies of the logical server structures and the mappings to the multiplexer servers are defined. The different service times for the basic servers are also defined in this diagram. In contrast to the original use case [17] the human actors are replaced by AI to have more widely performance predictions (otherwise the humans would be the bottleneck) - this is only done for these performance calculations. In the real workflow, human actors are not replaced by AI.

D. Tableau

FMC-QE delivers exact solutions for open Product From Queueing Networks. Through transformations, the flat example could be transformed into a hierarchical model. After this transformation, a broad range of performance values could be calculated in the FMC-QE Tableau, as shown in Table II. On every hierarchical level $[bb]$, this includes values like queue lengths $n_{i,q}^{[bb]}$, waiting times $W_i^{[bb]}$, service durations

TABLE II. CASE STUDY - TABLEAU.

Experimental Parameters	
$n_{ges}^{[1]}$	80
$\lambda_{bott}^{[1]}$	0,5556
f	0,9500
$\lambda^{[1]}$	0,5278

Service Request Section								Server Section						Dynamic Evaluation Section								
[bb]	i	$SRq_i^{[bb]}$	$p_{p(i),i}$	$v_{p(i)}^{[bb-1]}$	$v_{i,int}^{[bb]}$	$v_i^{[bb]}$	$\lambda_i^{[bb]}$	$Server_i^{[bb]}$	$m_{p(i)}^{[bb-1]}$	$m_{i,int}^{[bb]}$	$m_i^{[bb]}$	Mpx_i	$X_i^{[bb]}$	$m_{i,mpx}^{[bb]}$	$\mu_i^{[bb]}$	$\rho_i^{[bb]}$	$n_{i,q}^{[bb]}$	$W_i^{[bb]}$	$n_{i,s}^{[bb]}$	$Y_i^{[bb]}$	$n_i^{[bb]}$	$R_i^{[bb]}$
2	1	Get Employee Details from SuccessFactors	1,00	1,00	1,00	1,00	0,528	Details Retriever	1	1	1	1	0,750	1,000	1,333	0,396	0,259	0,491	0,396	0,750	0,655	1,241
4	2	Get Buddy List from SFSF	1,00	1,00	1,00	1,00	0,528	Get Buddy List Handler	1	1	1	2	0,500	30,000	60,000	0,009	0,000	0,000	0,009	0,017	0,009	0,017
4	3	Process Buddy List	1,00	1,00	1,00	1,00	0,528	Process Buddy List Handler	1	1	1	2	0,800	30,000	37,500	0,014	0,000	0,000	0,014	0,027	0,014	0,027
3	4	Get and Process Buddy List	1,00	1,00	1,00	1,00	0,528	Get and Process Buddy List Handler	1	1	1				37,500		0,000	0,001	0,023	0,043	0,023	0,044
3	5	Determine Equipment	1,00	1,00	1,00	1,00	0,528	Determine Equipment Handler	1	1	1	2	0,950	30,000	31,579	0,017	0,000	0,001	0,017	0,032	0,017	0,032
2	6	Get Buddy List and Equipment	1,00	1,00	1,00	1,00	0,528	List and Equipment Handler	1	1	1				31,579		0,001	0,001	0,040	0,075	0,040	0,076
3	7	Change or Confirm Equipment	1,00	1,20	1,00	1,20	0,633	C. or C. Equipment Handler	1	1	1	4	2,100	4,000	1,905	0,333	0,166	0,262	0,333	0,525	0,498	0,787
3	8	Approve Equipment	1,00	1,20	1,00	1,20	0,633	Approve Equipment Handler	1	1	1	3	1,500	1,000	0,667	0,950	18,050	28,500	0,950	1,500	19,000	30,000
2	5	Equipment Negotiation	1,00	1,00	1,20	1,20	0,633	Equipment Negotiation Handler	1	1	1				0,800		18,216	28,762	1,283	2,025	19,498	30,787
2	6	Accept workplace for new hire	1,00	1,00	1,00	1,00	0,528	Accept Workplace Handler	1	1	1	4	1,000	4,000	4,000	0,132	0,020	0,038	0,132	0,250	0,152	0,288
1	7	Order Notebook for new Employe	1,00	1,00	1,00	1,00	0,528	Request Handler	1	1	1				0,667		18,496	35,044	1,850	3,505	20,345	38,549
1	8	Request Generation	1,00	1,00	1,00	1,00	0,528	Client	1	1	1		113,030		0,009		0,000	0,000	59,655	113,030	59,655	113,030

Multiplexer Section			
j	$Name_j$	m_j	$X_j^{[1]}$
1	HR System	1	0,750
2	Cloud System	30	1,300
3	Manager KI	1	1,800
4	Mentor KI	4	2,250

$Y_i^{[bb]}$ and response times $R_i^{[bb]}$.

Through the dependencies in the FMC-QE Tableau, some parameters, such as the service times $X_j^{[1]}$ or multiplicities m_j of the multiplexers or the overall arrival rate $\lambda^{[1]}$, could be adjusted to predict the described values. An example of such a calculation is shown in Figure 12.

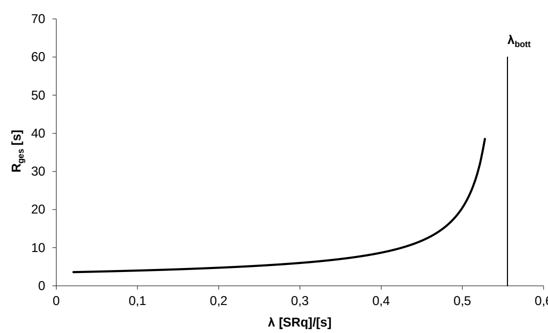

Figure 12. Case Study - Performance Prediction

This shows the dependency on the response time R from the arrival rate λ, as the arrival rate is increased towards the maximum bottleneck arrival rate λ_{bott}. In a possible real use case, a threshold for R could be defined from which for example further virtual servers would be allocated.

VI. CONCLUSION AND FUTURE WORK

With the help of FMC-QE including its hierarchical modeling and the underlying hierarchical performance calculations performance values, such as response times or queue lengths could be predicted even for distributed cloud-based systems. These performance predictions could be used to adapt the Service-Level-Agreements (SLAs) of the connected cloud-systems, while one of the main components of FMC-QE is the service request. These predictions could be further integrated into the algorithms of self-adaptive systems while the hierarchical approach reduces the complexity dramatically. In this publication, the performance predictions are integrated into a spreadsheet program, but as said, it is not limited to this.

In the future, it is planned to further integrate calculations of the FMC-QE Tableau to BPMN as BPMN is a widely used modeling notation. Therefore, patterns for the hierarchical modeling will be defined to transform BPMN Diagrams or further annotate it.

Furthermore, as already described in Section III the Palladio approach seems to address similar problems, therefore, cooperation would be from interest. One possible connection point could be the integration of the FMC-QE calculation (FMC-QE Tableau) into Palladio, another could be an exchange of experience in the area of model transformations, as it is said, that this a current research question in Palladio [15].

Also, a more extensive comparison to the predictive

 ISBN: 978-1-61208-781-8

process monitoring approaches such as mOSAIC [18] or [22], which are often simulations in contrast to the numerical analysis shown here, would further sharpen the results, as already done in [7].

ACKNOWLEDGMENT

The author would like to thank Prof. Dr.-Ing. Werner Zorn for countless useful discussions during the development of FMC-QE. The author would furthermore thank Dr. Helge Fischer for encouraging to this publication.

REFERENCES

[1] M. Chui *et al.*, "Ten it-enabledbusiness trends for the decade ahead," McKinsey Global Institute, Tech. Rep., May 2013.

[2] D. W. Cearley, "Top 10 strategic technology trends for 2019," Gartner, Tech. Rep., 2019.

[3] S. Kluth, "Quantitative Modeling and Analysis with FMC-QE," Ph.D. dissertation, Hasso Plattner Institute for Software Systems Engineering at the University of Potsdam, Potsdam, Germany, July 2011.

[4] W. Zorn, "FMC-QE - A New Approach in Quantitative Modeling," in *Proceedings of the International Conference on Modeling, Simulation and Visualization Methods (MSV 2007) within WorldComp '07*, H. R. Arabnia, Ed. Las Vegas, NV, USA: CSREA Press, June 2007, pp. 280 – 287.

[5] T. Porzucek, S. Kluth, F. Copaciu, and W. Zorn, "Modeling and Evaluation Framework for FMC-QE," in *Proceedings of the 16th IEEE International Conference on the Engineering of Computer-Based Systems (ECBS2008)*. Los Alamitos, CA, USA: IEEE Computer Society, April 2009, pp. 237–243.

[6] M. Seelig *et al.*, "Comparison of Performance Modeling and Simulation - a Case Study," in *Proceedings of the 15th IEEE International Conference on Engineering of Computer-Based Systems (ECBS 2008 - Belfast, UK)*, D. W. Bustard and R. Sterritt, Eds. Los Alamitos, CA, USA: IEEE Computer Society, March 2008, pp. 49–56.

[7] T. Porzucek, M. Fritzsche, S. Kluth, and D. Redlich, "Combination of a Discrete Event Simulation and an Analytical Performance Analysis through Model-Transformations," in *Proceedings of the 17th IEEE International Conference on the Engineering of Computer-Based Systems (ECBS2010)*. Los Alamitos, CA, USA: IEEE Computer Society, March 2010, pp. 183–192.

[8] A. Knöpfel, B. Gröne, and P. Tabeling, *Fundamental Modeling Concepts: Effective Communication of IT Systems*. John Wiley & Sons, März 2006.

[9] P. J. Denning and J. P. Buzen, "The Operational Analysis of Queueing Network Models," *ACM Computing Surveys (CSUR)*, vol. 10, no. 3, pp. 225–261, September 1978.

[10] C. A. Petri, "Kommunikation mit Automaten - Communication with automata," Ph.D. dissertation, Institut für instrumentelle Mathematik, Bonn, Germany, 1962.

[11] S. Wendt, "Operationszustand versus Steuerzustand - eine äusserst zweckmässige Unterscheidung - operational state vs. control state - an extremely useful distinction," University of Kaiserslautern, Kaiserslautern, Germany, Tech. Rep., Februar 1998.

[12] J. D. C. Little, "A Proof of the Queueing Formula $L = \lambda * W$," *Operations Research*, vol. 9, no. 3, pp. 383–387, May - June 1961.

[13] R. H. Reussner, S. Becker, J. Happe, R. Heinrich, A. Koziolek, H. Koziolek, M. Kramer, and K. Krogmann, *Modeling and Simulating Software Architectures – The Palladio Approach*. Cambridge, MA: MIT Press, October 2016.

[14] R. Heinrich, D. Werle, H. Klare, R. Reussner, M. Kramer, S. Becker, J. Happe, H. Koziolek, and K. Krogmann, "The palladio-bench for modeling and simulating software architectures," in *Proceedings of the 40th International Conference on Software Engineering: Companion Proceedings*, ser. ICSE '18. New York, NY, USA: ACM, 2018, pp. 37–40.

[15] S. Seifermann, D. Werle, and M. Ebada, "Mapping data flow models to the palladio component model," in *Proceedings of the 10th Symposium on Software Performance (SSP)*, ser. Softwaretechnik Trends, 2019, pp. 41–43.

[16] S. Kluth, "Discussion on the zorn-hypothesis on inter-server control flow and product form queueing networks," in *Proceedings of the 19th IEEE International Conference and Workshop on the Engineering of Computer Based Systems (ECBS2012)*, S. V. Miroslav Popovic, Bernhard Schätz, Ed. Los Alamitos, CA, USA,: IEEE Computer Society, 2012, pp. 126–134.

[17] T. Steckenborn, "A joint exploration of SAP Cloud Platform Workflow," SAP, Tech. Rep., 2019.

[18] U. V. A. Cuomo, M. Rak, "Performance prediction of cloud applications through benchmarking and simulation," *Int. J. of Computational Science and Engineering*, vol. 0, no. 00, 2004.

[19] W. Zorn, "Hierarchical Modeling based on Service Requests," 2007, Hasso-Plattner-Institute, University of Potsdam.

[20] "Specification Business Process Model and Notation(BPMN) Version 2.0.2," Object Management Group, Tech. Rep., Dec. 2013.

[21] "BPMN Poster," BPM Offensive Berlin, Hasso-Plattner-Institut, Humboldt University Berlin, camunda, inubit, signavio, Tech. Rep., 2011.

[22] B. K. John J. Prevost, KranthiManoj Nagothu and M. Jamshidi, "Prediction of Cloud Data Center Networks Loads Using Stochastic and Neural Models," in *Proc. of the 2011 6th International Conference on System of Systems Engineering*, 2011.