CloudWave: where Adaptive Cloud Management Meets DevOps

Dario Bruneo*, Thomas Fritz§, Sharon Keidar-Barner‖, Philipp Leitner§, Francesco Longo*, Clarissa Marquezan¶, Andreas Metzger¶, Klaus Pohl¶, Antonio Puliafito*, Danny Raz‡, Andreas Roth**, Eliot Salant‖, Itai Segall‖, Massimo Villari*, Yaron Wolfsthal‖, Chris Woods†

*Università degli Studi di Messina, Italia - {dbruneo, flongo, apuliafito, mvillari}@unime.it
†Intel, Ireland - chris.woods@intel.com
‡Technion University, Israel - danny@cs.technion.ac.il
§University of Zurich, Switzerland - {fritz,leitner}@ifi.uzh.ch
¶University of Duisburg-Essen, Germany - {clarissa.marquezan,andreas.metzger,klaus.pohl}@paluno.uni-due.de
‖ IBM, Haifa Research Lab, Israel - {sharon,salant,itais,wolfstal}@il.ibm.com
** SAP, Germany - andreas.roth@sap.com

***Abstract*—The transition to cloud computing offers a large number of benefits, such as lower capital costs and a highly agile environment. Yet, the development of software engineering practices has not kept pace with this change. Moreover, the design and runtime behavior of cloud based services and the underlying cloud infrastructure are largely decoupled from one another.This paper describes the innovative concepts being developed by CloudWave to utilize the principles of DevOps to create an *execution analytics* cloud infrastructure where, through the use of programmable monitoring and online data abstraction, much more relevant information for the optimization of the ecosystem is obtained. Required optimizations are subsequently negotiated between the applications and the cloud infrastructure to obtain *coordinated adaption* of the ecosystem. Additionally, the project is developing the technology for a *Feedback Driven Development Standard Development Kit* which will utilize the data gathered through execution analytics to supply developers with a powerful mechanism to shorten application development cycles.**

***Index Terms*—Cloud Computing, DevOps, Coordinated Adaptation, Feedback Driven Development**

I. Introduction

Currently within ICT, cloud computing plays a key role and is recognized as one of the most significant technologies for boosting productivity, economic growth and job [1] development. Cloud computing offers a broad range of service delivery models: IaaS (infrastructure as a service), PaaS (platforms, operating systems, execution environments as a service), SaaS (software applications delivered as a service), and BPaaS (business processes offered as a service), and provides tremendous market opportunities, up to an overall cumulative impact on (EU) GDP of EUR 957 billion, and 3.8 million jobs, by 2020 [1].

As cloud computing takes hold, the challenges of fully realizing its potential become evident. First, cloud computing has long been seen as mere cost saver. However, today, IT leaders increasingly consider the improvement of business agility as well as faster innovation the major strategic reasons for adopting this paradigm. Yet, the engineering methods and tools used to develop cloud services have not yet made the leap towards these expectations. Thus, there is a need to increase the clock speed of innovation by improving agility in designing software and operating cloud based service as well as for improving cloud infrastructure adaptivity and the interaction between the higher level software & service goals and the lower level infrastructure (see NESSI [2]).

Moreover, cloud applications are often designed with incomplete knowledge about their actual usage profile, delivery model, and the reliability of the cloud infrastructure. This may lead to unforeseen runtime situations, resource utilization inefficiencies, performance degradation, stability problems, and even failures that can cause outages, as recently observed with leading service providers. These can result in loss of competitive advantage for service providers due to current and potential client's concerns about Quality of Service (QoS) and usage risks [3].

Various solutions have been proposed to alleviate the cloud QoS problem. For the most part, cloud providers have focused on advancements in cloud infrastructure technologies. However, in spite of significant technological progress in the area of cloud infrastructure, current cloud computing platforms still fall short of delivering truly compelling end-to-end QoS. We believe that this gap is a fundamental consequence of both application-level and infrastructure-level deficiencies: applications are not designed to explore the distribution, characteristics, and dynamic behavior of cloud infrastructure while existing cloud technology stacks (such as OpenStack [4] and CloudStack [5]) do not provide concrete mechanisms to facilitate such cloud application development.

Thus, new, holistic, cloud-specific engineering methods are required. Such a novel paradigm should facilitate design and

The research leading to these results has received funding from the European Community's Seventh FrameworkProgramme (FP7/2007-2013) under grant agreement n. 610802.

delivery of reliable cloud services and should support their continuous adaptation to changing environment conditions and market requirements, speeding up innovation cycles. Here, we describe an attempt to address this challenge by delivering novel technologies and methods for improving both the development of SaaS solutions and the management of their operation and execution.This new development and management paradigm called CloudWave is built upon new technologies and open standards, while leveraging assets and outcomes of relevant FP7 EU projects such as RESERVOIR [6], FI-WARE [7], Optimis [8] and others.

In this paper, we provide an in depth description of the proposed architecture and framework for a novel cloud software stack (Section II) and we describe a motivating scenario showing how these new concepts enable highly effective development and deployment of complex cloud applications (see Section III).

Overall, we demonstrate how the proposed solution can be used to develop and deliver services in an agile, effective, efficient and reliable manner across the future computing, while utilising open source approaches also in comparison with related scientific works and projects (Section IV). Section V concludes this paper by providing insight on future work.

II. Concept and Innovation

Cloudwave takes inspiration from the DevOps approach [9] of promoting communication, collaboration, and integration between developers and operational teams and shortening the loop between software creation, deployment, operation, and feedback.

Unlike shrink-wrapped or on-premises software, cloud software is often written, hosted, and deployed by the same team. This allows the entire life cycle of a cloud application to be controlled by its creator. CloudWave focuses on an iterative cycle of application evolution and improvement through (1) continuous analysis of application-centric operational data and (2) applying adjustments during operations. This ongoing process can be exploited in order to deliver continuously improving applications, which rapidly evolve and dynamically adapt to their execution environment, connected devices, and user requirements. Enabling this revolutionary development and management approach across all cloud delivery models and management mechanisms is the challenge addressed by CloudWave. To the best of our knowledge, CloudWave is the first technology-based, holistic solution leveraging DevOps' principles.

A. Innovation Pillars

CloudWave presents a new paradigm for developing cloud applications which support the creation of high-QoS service for the Future Internet. This paradigm is supported by three key *innovation pillars*: (1) Execution Analytics, (2) Coordinated Adaptation, (3) Feedback-Driven Development. These are depicted in Figure 1 and described in detail below.

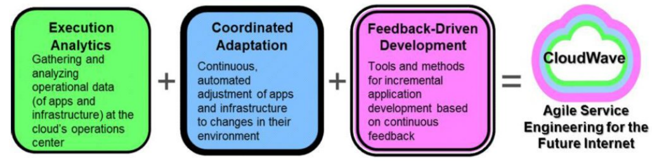

Fig. 1. The main three Pillars of CloudWave

1) Execution Analytics: A fundamental idea introduced by CloudWave is an integrated analytics framework where specialized mechanisms and algorithms are used to dynamically introspect and analyse the behaviour of cloud applications. The framework seamlessly integrates real-time usage information from (1) data centre infrastructure, including computing, storage and network resources, along with device and sensor data from the Internet of Things; and (2) runtime data from applications, including operation and context information, user interaction patterns and data from third-party services available via the Internet of Services (IoS). Overall, this framework generates a consolidated view of all operational data related to the application and its execution environment, providing a basis for decisions on various optimizations and transformations that are to be applied to the cloud infrastructure and the applications. See the left part of Figure 2.

a) A new execution analytics – a unified open approach: The majority of both academic and commercial solutions provide an incomplete view of the application. They either form part of the application and can closely monitor its behaviour [10], or they form part of the infrastructure and focus on resource utilisation (cf. [11]). Recently proprietary solutions have been released which try to combine these two focus areas [12].

The CloudWave proposal goes further offering two clear advancements. Firstly CloudWave offers a unified open approach: the infrastructure, the application and even the IoT execution context are all monitored and holistically analysed. This allows observations to be correlated across each of these areas resulting in the ability to make balanced decisions on optimisation and application function. Secondly, in addition to providing analytics which is consumed by automated decision making systems, CloudWave provides additional analytics which will be consumed by both developers and operators of the software. This human inspection of the analytics is key to CloudWave's Feedback Driven Development approach, and provides humans with key data on the computational and commercial performance of their software.

b) Main capabilities - programmability and abstraction: This unified analytics approach presents a significant challenge due to the huge volume of data produced by the cloud infrastructure, cloud applications and IoT execution context. CloudWave introduces two capabilities to address this: (i) Programmable Monitoring and (ii) Online Data Abstraction. Both are supported through standardized interfaces.

(i) Programmable Monitoring, enables the deployment of flexible monitoring strategies to focus on relevant data in order to support optimization decisions. With this, we extend the state of art, as earlier efforts for adaptive monitoring [13] were

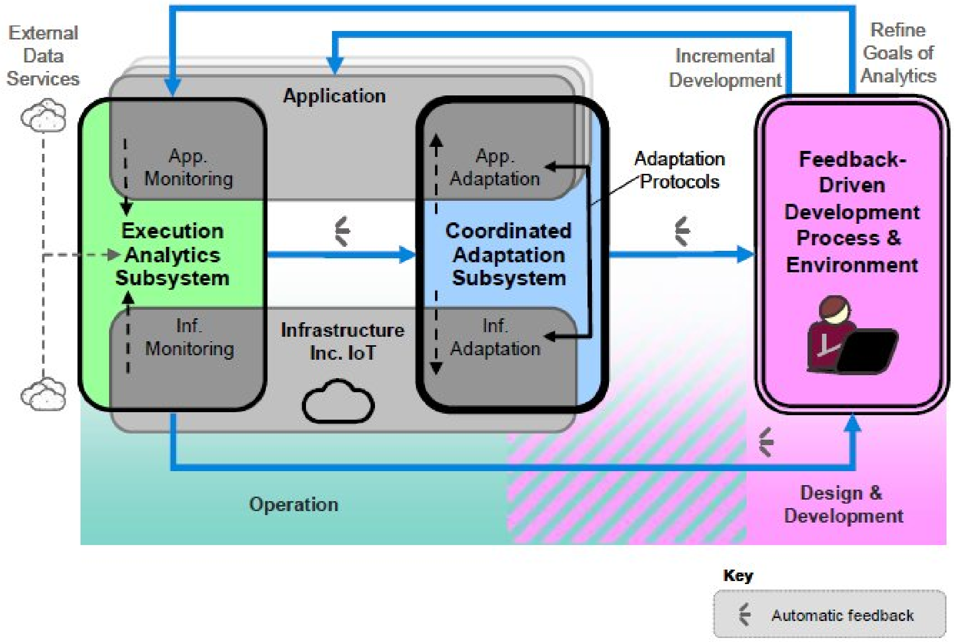

Fig. 2. The building blocks of the CloudWave paradigm, in which two main feedback loops are depicted.

mainly focused on cost reduction and did not have an explicit support for cloud environments. In part, CloudWave will reuse monitoring technologies developed in the RESERVOIR and Optimis FP7 projects.

(ii) Online Data Abstraction, enables the reduction of data through data filtering, compression, abstraction and exploitation of data dependencies according to predefined or user-specified policies, while maintaining the informational value of the data.

To support both the automated adaption and feedback driven development approaches the analytics framework will be architected to enable real-time collection and analysis of application-centric operational data from all sources, and to provide the developers with rapid indications about the potential impact of application transformations.

2) Coordinated Adaptation: The second innovation pillar of CloudWave is a new software technology where cloud applications and the underlying ICT infrastructure automatically adapt, in a coordinated manner, to dynamic changes in their environment so as to optimise service quality and cost. This capability, called Coordinated Adaptation, exploits the Execution Analytics Framework which, based on all relevant operational data, generates the feedback that dynamically drives adaptation actions across all layers of the service delivery environment (including the cloud infrastructure, end-user devices and the software components of the service). See the central part of Figure 2.

a) Real-time effective coordination across all layers: Currently, techniques and frameworks exist for software and system adaptation (e.g., self-* systems [14]), autonomic infrastructures (e.g., [15]) and service adaptation (e.g., SLA re-negotiation [16]). However, these solutions were mainly developed independently of each other, prior to the emergence of the cloud as mainstream computing paradigm. Thus, the current state-of-the-art neither leverages the capabilities provided by the cloud, nor does it address the coordination of adaptation actions across cloud applications and infrastructure, giving rise to potential conflicts and diminishing service quality [17]. CloudWave will radically improve the state of art by introducing a cloud-ready solution where adaptation is done through real-time effective coordination across all layers of the service delivery environment. Building upon the service delivery model of the cloud, CloudWave allows the holistic analysis data from both application and supporting infrastructure in real-time, enabling a new form of coordinated and balanced optimisation to occur.

b) Techniques enabling adaptation: CloudWave will introduce new techniques for cloud-focused adaptation including, the smart offloading of computations between cloud infrastructure and end-user devices, runtime binary-level optimization of application code and coordinated infrastructure/service adaptation of e.g., resource allocation or system/service configuration.

A set of Coordinated Adaptation Protocols will allow both applications and the cloud infrastructure to request adaption, this “negotiation” process allows the application, and the infrastructure to request actions (e.g., reallocation of resources), and exchange state information to support adaptation decisions. This negotiation process will be supported by standard interfaces and implemented by a set of software layers called Adaptation Engines.

c) Reliability during adaptation: The strong focus on adaptation reliability is essential to ensure the quality and cost of adaptation actions (e.g., in preventing adaptation actions that have unsafe side effects, cf. [18]). To support this an emphasis will be placed on ensuring the reliability of adaptation decisions by studying the effect of adaptations after their implementation.

3) Feedback-Driven Development: In parallel to coordinated adaptation, CloudWave will advocate and enable a radically new approach to building cloud applications, called feedback-driven development, taking ease-of-development to a new level. In this approach, developers exploit real-time data from actual usage in the field to support software construction within agile feedback loops. Feedback-driven development, like Coordinated Adaptation, will exploit all relevant operational data provided by the Analytics Framework. However, while Coordinated Adaptation is automated, feedback-driven development involves manual development of the software by the programming team. Both capabilities build upon the fact that the cloud enables radical exploitation of operational data gathered at the operations centre, where applications are developed, hosted and deployed.

CloudWave will address the fundamental technical challenges related to feedback-driven development and will introduce new technologies, methodologies and tools to overcome these challenges. Among others, some of the key challenges in this context are (1) the deluge of operational data, (2) the requirement for agile development cycles and (3) reliability during software evolution. See the right part of Figure 2.

a) Handling data deluge: Operational data gathered from business processes, applications, and the underlying cloud infrastructure may be overwhelmingly large, making it impractical for developers to absorb and intelligently exploit it. CloudWave will develop means to filter and expose this data to developers in a digestible way and to control the data generation. Those means will provide just the right amount of data that is most relevant during a given development activity and will use appropriate interaction modalities like the FDD Standard Development Kit (FDD SDK) and a powerful feedback reporting service to do so.

b) Highly agile development cycles: In contrast to pre-cloud agile development methods [19], feedback-driven development is based on real-time feedback. In order to take advantage of such timely data, the feedback-driven development paradigm should support development cycles that are radically shorter than those supported by current methods. CloudWave will provide methods and tools to facilitate such short development cycles, aiming to reduce their duration to several days by minimizing development overheads.

c) Reliable software evolution: As with software adaptation, the reliability of feedback-driven software transformations is of utmost importance, especially due to the high cost of manual programming effort required to evolve the software. Accordingly, CloudWave will place special emphasis on ensuring the reliability of these software transformations. Among other means, CloudWave will provide specialised tools and methods for testing software transformations and their impact on application quality and performance.

Overall, the feedback-driven development approach with its supporting services and tools and will enable a new style of software development, which is incremental, usage-driven, and is optimised for the cloud era. This new software development style will allow applications to safely grow from a core set of features, whilst developers can continuously assess whether extensions and optimizations will deliver return on investment. We are confident that with the growing prevalence of cloud computing, this method has the potential to become the industry-standard method of application development for the Future Internet.

III. Motivating Scenario

In this section, we present a motivational scenario able to demonstrate how the innovations of CloudWave will enable development and deployment of adaptive Future Internet services.

Let us consider a virtual Telco service operator developing value-add cloud/IoT hybrid services. The operator has developed a cloud service for organisational collaboration using communication technologies that span from corporate computers to employees' private smart phones. The cloud service relies on the customer's private cloud for data storage and encryption. The first pilot of this service was tested by a small organisation and then the Telco operator managed to convince a big healthcare organization to pilot the service across its largest division in order to have a unique opportunity to penetrate the lucrative healthcare market. The main concern is that it would take an unacceptable long time to characterize usage patterns before parametrizing and optimising the service accordingly.

Moreover not all requirements are clear at this stage. In order to delay modifications to the service implementation until they are proven as essential to meet the QoS demands, the Telco operator decides to use the CloudWave agile service engineering paradigm and technology. This allows the Telco to dynamically customise its service according to the Agile Feedback Driven Development methodology.

The following events occur:

1) The developers program the CloudWave Execution Analytics Subsystem to capture usage profiles and enable targeted infrastructure monitoring.
2) The Telco operator then launches the service, which runs across the healthcare organization employees' smart phones, leveraging the private cloud.
3) Within a short period of time, the CloudWave Execution Analytics Subsystem returns feedback to the developer, showing performance statistics of the application. Among the findings, the developers observe that the response time of the service is slower during specific periods of the day.
4) The developers consult with the advisory board of the healthcare organization and, upon short introspection, the managers explain that the highest load on their ICT infrastructure occurs twice daily when the next shift takes over, reflecting the large amounts of data that need to be shared between employees. At this point, a new requirement for bounded service response time is introduced that was not part of the original set of requirements.
5) To meet this new challenging requirement, the Telco operator's developers add a new feature to their application: a concurrent video transfer function is implemented

to enhance scalability. The Coordinated Adaptation Subsystem automatically allocates a sufficient amount of resources for this function through negotiation between the application and the cloud infrastructure, when the Execution Analytics Subsystem indicates increased usage.

6) Service deployment continues without interruption. Data provided by the feedback indicates that the average response time has improved, while in a few occasions, the cloud infrastructure could not allocate all the resources requested by the service. The consolidated feedback mechanism reveals that at those times, numerous service instances were busy with encryption operations.
7) The developers add new application management code that off-loads encryption operations from the cloud to smart phones when the CloudWave Execution Analytics Subsystem indicates that the cloud cannot allocate additional resources.
8) At this point, all requirements on the service have been met within the aggressive dead-line, and the healthcare organization management is satisfied with the demonstrated QoS granting a large 10-year ICT services to the Telco operator.

This scenario illustrations how Cloudwave enables the construction of cost-effective high quality services. Although the client introduced a new requirement after the development has started, the developers could address it efficiently by programming the monitoring goals using the Execution Analytics Subsystem and by using three agile development cycles of Feedback-Driven Development.

IV. Related work

In [20] and [21] authors provide an exhaustive description of the DevOps approach. They describe how the current process of creating software is split between the developers, who write, build, and test the application, and an operational team which deploys and manages the application. This separation in teams and the resulting poor communication between them often results in many issues. To overcome this a method of automating the process of deployment is required. The Model-driven based approach is proposed in [22] and [23]. In the latter, the *MODACLOUD* solution is described. The Model-Driven Development proposed is aimed at designing software systems in a cloud-agnostic way. The applications investigated in [23] run on multiple clouds. Similarly to the solution proposed in this work, the authors of [23] have identified two main areas of modeling times: *Design Time* and *Real Time*. However these papers do not include anything similar to the Feedback-Driven Development presented by CloudWave.

A sub-technique of Model-driven approach is the *Application Lifecycle Model* as described in [24]. Here, the *application-centric* five cloud keys are identified, that is: *1) Application Design*, *2) Development / Migration*, *3) Deployment*, *4) Maintenance* and *5) Withdraw*. For PaaS frameworks, the Deployment phase presents a complex component deployment topology, since it is described with seven sub-steps. Another example is the Application Lifecycle Management Platforms for PaaS frameworks as reported in [25] and [26]. The paper describes a complex DevOps Tool/Model Chains XML based. Their system is named *Service Lambda* for Cloud Computing. The paper's proposed architecture shows a high degree of complexity.

Further investigations on possible facilities to support DevOps are conducted in [27] and [25]. The idea of providing further information to Developers and Operators by exploiting system logs is explored in [27], here the author remarks how it is possible to identify Error-Prone software components using log maintenance history. Analysis of applications based on Hadoop Clouds is reported in [25]. This work goes further and shows how to cross-correlate logs. Finally, [28] presents an early approach for helping developers to capture information about the applications they create, at the time of creation, within an IDE. Ultimately this aids code maintenance and knowledge sharing for any application, including those deployed in the cloud. CloudWave proposes an approach in which the Developer is strongly involved in the *design* of an application and *Analysis* of its run time behaviour. Papers that include a similar approach include [29], which attempts to modify the Eclipse IDE to better support Programming in the Cloud. In this work the authors present what are the main challenges to be addressed when modifying Eclipse to become a valid on-line IDE.

V. Conclusions and Future Work

CloudWave arose from the perception that the transition to Cloud-based services demands both faster development cycles and high degree of exploitation of the Cloud infrastructure in order to obtain high QoS levels. While many efforts exist to obtain higher performance levels through advancement of the infrastructure technology, CloudWave is unique in taking a holistic approach by examining the interaction of running applications and the infrastructure as a complete eco-system. The infrastructure is capable of monitoring the runtime environment, and through the use of execution analytics, both the infrastructure and applications are able to collaborate to provide dynamic reconfiguration. In addition to the automatic reconfiguration that the system can provide, developers are given access to a powerful SDK which implements the concept of DevOps to aid in shortening iterative development cycles. The development of CloudWave will be closely accompanied by experimentation and verification of results through the use of real world industrial use cases as benchmarks. A reference architecture will be created, where all functional specification, interface descriptions and portions of the code will be released as OpenSource.

References

[1] Communication from the Commission to the European Parliament, "Unleashing the Potential of Cloud Computing in Europe, 2012: http://ec.europa.eu/information_society/activities/cloudcomputing/docs/com/com_cloud.pdf."

[2] NESSI Cloud White Paper: http://www.nessieurope.com/Files/Private/120718_NESSI_Cloud_WhitePaper_July.pdf.

[3] W. Chang, H. Abu-Amara, and J. Sanford, "Challenges of enterprise cloud services," in *Transforming Enterprise Cloud Services*, pp. 133–187, Springer Netherlands, 2010.

[4] OpenStack Open Source Cloud Computing Software: www.openstack.org.

[5] Apache CloudStack: Open Source Cloud Computing, http://cloudstack.apache.org.

[6] B. Rochwerger *et al.*, "Reservoir - when one cloud is not enough," *Computer*, vol. 44, pp. 44–51, 2011.

[7] FI-WARE cost-effective creation and delivery of Future Internet applications
http://www.fi-ware.eu/ December 2013.

[8] Optimis - Optimized Infrastructure Services
http://www.optimis-project.eu/ December 2013.

[9] P. Debois, "Devops: A software revolution in the making?," *Journal of Information Technology Management*, 2011.

[10] M. Salehie and L. Tahvildari, "Self-adaptive software: Landscape and research challenges," *ACM Trans. Auton. Adapt. Syst.*, vol. 4, pp. 14:1–14:42, May 2009.

[11] J. Moore and J. Chase, "Data center workload monitoring, analysis, and emulation," in *in Eighth Workshop on Computer Architecture Evaluation using Commercial Workloads*, 2005.

[12] E. Caron, L. Rodero-Merino, F. Desprez, and A. Muresan, "Auto-Scaling, Load Balancing and Monitoring in Commercial and Open-Source Clouds," rapport de recherche, INRIA, Feb. 2012.

[13] D. Breitgand, R. Cohen, A. Nahir, and D. Raz, "On cost-aware monitoring for self-adaptive load sharing," *Selected Areas in Communications, IEEE Journal on*, vol. 28, no. 1, pp. 70–83, 2010.

[14] M. Hinchey and R. Sterritt, "Self-managing software," *Computer*, vol. 39, no. 2, pp. 107–109, 2006.

[15] F. Andr *et al.*, "Architectures & infrastructure," in *Service Research Challenges and Solutions for the Future Internet* (M. Papazoglou, K. Pohl, M. Parkin, and A. Metzger, eds.), vol. 6500 of *Lecture Notes in Computer Science*, pp. 85–116, Springer Berlin Heidelberg, 2010.

[16] S. Benbernou *et al.*, "Modeling and negotiating service quality," in *Service Research Challenges and Solutions for the Future Internet* (M. Papazoglou, K. Pohl, M. Parkin, and A. Metzger, eds.), vol. 6500 of *Lecture Notes in Computer Science*, pp. 157–208, Springer Berlin Heidelberg, 2010.

[17] S. Guinea, G. Kecskemeti, A. Marconi, and B. Wetzstein, "Multi-layered monitoring and adaptation," in *Proceedings of the 9th International Conference on Service-Oriented Computing*, ICSOC'11, (Berlin, Heidelberg), pp. 359–373, Springer-Verlag, 2011.

[18] R. Lemos *et al.*, "Software engineering for self-adaptive systems: A second research roadmap," in *Software Engineering for Self-Adaptive Systems II* (R. Lemos, H. Giese, H. Mller, and M. Shaw, eds.), vol. 7475 of *Lecture Notes in Computer Science*, pp. 1–32, Springer Berlin Heidelberg, 2013.

[19] R. Martin, *Agile Software Development: Principles, Patterns, and Practices*. Alan Apt Series, Prentice Hall/Pearson Education, 2003.

[20] W. Johannes, "Concepts for integrating devops methodologies with model-driven cloud management based on tosca," Master's thesis, 2013.

[21] J. Wettinger, M. Behrendt, T. Binz, U. Breitenbcher, G. Breiter, F. Leymann, S. Moser, I. Schwertle, and T. Spatzier in *CLOSER* (F. Desprez, D. Ferguson, E. Hadar, F. Leymann, M. Jarke, and M. Helfert, eds.), pp. 437–446, SciTePress.

[22] M. Miglierina, G. Gibilisco, D. Ardagna, and E. Di Nitto, "Model based control for multi-cloud applications," in *Modeling in Software Engineering (MiSE), 2013 5th International Workshop on*, pp. 37–43, 2013.

[23] D. Ardagna, E. Di Nitto, P. Mohagheghi, S. Mosser, C. Ballagny, F. D'Andria, G. Casale, P. Matthews, C.-S. Nechifor, D. Petcu, A. Gericke, and C. Sheridan, "Modaclouds: A model-driven approach for the design and execution of applications on multiple clouds," in *Modeling in Software Engineering (MISE), 2012 ICSE Workshop on*, pp. 50–56, 2012.

[24] K. Tang, J. M. Zhang, and C. H. Feng, "Application centric lifecycle framework in cloud," in *e-Business Engineering (ICEBE), 2011 IEEE 8th International Conference on*, pp. 329–334, 2011.

[25] W. Shang, Z. M. Jiang, H. Hemmati, B. Adams, A. Hassan, and P. Martin, "Assisting developers of big data analytics applications when deploying on hadoop clouds," in *Software Engineering (ICSE), 2013 35th International Conference on*, pp. 402–411, 2013.

[26] S. Hosono and Y. Shimomura, "Application lifecycle kit for mass customization on paas platforms," in *Services (SERVICES), 2012 IEEE Eighth World Congress on*, pp. 397–398, 2012.

[27] W. Shang, "Bridging the divide between software developers and operators using logs," in *Software Engineering (ICSE), 2012 34th International Conference on*, pp. 1583–1586, 2012.

[28] A. Guzzi, M. Pinzger, and A. van Deursen, "Combining micro-blogging and ide interactions to support developers in their quests," in *Software Maintenance (ICSM), 2010 IEEE International Conference on*, pp. 1–5, 2010.

[29] L. Wu, G. Liang, S. Kui, and Q. Wang, "Ceclipse: An online ide for programing in the cloud," *2013 IEEE Ninth World Congress on Services*, pp. 45–52, 2011.