Author's version. Published in the:
Proceedings of the IEEE International Conference on Service Oriented Computing & Applications SOCA 2011, Irvine

DYNO: A Notation to Leverage Dynamic Network Effects in PaaS Ecosystems

Ulrich Scholten, Robin Fischer, Christian Zirpins
Karlsruhe Institute of Technology
Karlsruhe, Germany
{Ulrich.Scholten, Robin.Fischer, Christian.Zirpins}@kit.edu

Simone Scholten
SAP Research
Walldorf, Germany
Simone.Scholten@SAP.com

***Abstract*—Successful Platform-as-a-Service (PaaS) providers excel in governing their service performance by leveraging economic network effects, which implicitly control PaaS ecosystems. Up-to-now, there exists only little work that addresses the challenge of governing service compositions that are created by independent service providers and service consumers on top of PaaS offerings. In this technical report, we propose a notation for dynamic network effects (DYNO) allowing modeling and analyzing network effects in such PaaS-based service networks. The notation is further designed to help platform providers in placing control mechanisms into the PaaS ecosystem to guarantee high quality of service. We exemplify how business analysts and service engineers may use DYNO-models to understand and shape network effects in dynamic PaaS ecosystems based-on a real world case.**

Keywords-network notation, network dynamics, base value, control, self-organization, PaaS

I. Introduction

Traditionally, IS development processes have been centrally orchestrated through a single process owner [1]. The distributed service-oriented computing model in Cloud computing, however, changes this paradigm: Increasingly, the formerly centralized process ownership is now shared between Platform-as-a-Service (PaaS) providers, their surrounding ecosystems of Software-as-a-Service (SaaS) providers, and service consumers [2, 3, 4, 5]. By exposing dedicated interfaces, PaaS providers leverage external development resources and creativity to extend their service portfolio. The challenge, however, is to control autonomous service providers in highly responsive and dynamic service networks, maintaining high quality SaaS offerings that complement the original PaaS offer. Similarly, SaaS providers face the challenge of positioning their service offer in a context, where existing network dynamics and scale effects are exploited. Their undertaking needs to find the right strategic network position, where the own value contribution can ride on prevalent network dynamics.

In previous work, we experienced that many of the ventured PaaS offerings did not flourish due to a lack of quality performance [6] or missing market acceptance of offered SaaS applications [5]. A tool or solution for cloud business analysts and service engineers on how to find the best position in a dynamic context does - to our best knowledge – not exist as of today. At the same time, process models are understood in a very explicit way, focusing on aspects such as data or message flows and business protocols. However, in the dynamic context of Cloud computing, system or network effects are originating in more indirect (i.e., implicit) patterns and relationships. They cannot be directly modeled through service choreographies or process orchestration.

In this paper, we introduce a notation for dynamic network effects (DYNO), empowering business analysts and service engineers to govern quality of service. We thereby equip Cloud service providers ('protagonists') with tools to understand network dynamics and its toeholds to control PaaS ecosystems. The following research questions inspired the development of DYNO:

(a) Where are dynamic processes around the protagonist's value proposition ("base value") located? (b) Where are toeholds to initiate these dynamic processes located? (c) Where are service-providing IT-systems exposed to network effects and, thus, required to scale quickly to maintain high quality of service? (d) How can quality of service be controlled in the PaaS ecosystem?

The groundwork for this notation has been made through various research activities and studies. An understanding on dynamic processes and base value was gathered through explorative analysis of several successful platform providers in [7, 5]. Understanding of technical requirements was developed through market analysis in conjunction with laboratory experiments [8] as well as experiments in the context of the research platform AGORA in cooperation with SAP Research [10]. This technical research is ongoing, currently focusing on monitoring mechanisms for scalable automated control. Insights into the relevance of protagonists' control on quality of service and the respective designs of distributed control settings have been gathered through a longitudinal analysis of a selection of service intermediaries [6].

The paper is structured as follows: After setting the context of related work (section II), we summarize the specific challenges of distributed PaaS ecosystems (section III). Based on the aforesaid, we suggest a meta-model for a dynamic network notation (section IV) and a corresponding graphical representation (section V). A real world PaaS showcase exemplifies the applicability of the notation from the perspective of PaaS providers (section VI). The paper closes with a conclusion and an outlook on further research.

II. Related Work

Our research is to be positioned into the fields of service-oriented computing with special focus on PaaS offerings. A general framework and research context is provided by the eOrganization group at KIT, conducting research on Cloud Service Engineering, "a discipline that applies a systematic approach to create value added services on top of a Cloud Computing Infrastructure" [9].

Research with the related goal of "developing, monitoring and optimizing SOA-enabled business processes in service networks" [11] is conducted in the frame of the S-Cube project. There, service networks are described in a Service Network Notation (SNN), respectively in a Graphical Service Network Modeling Language to help solving optimization problems for process-based performance indicators (KPIs) and service level agreements (SLAs) [12, 13].

Furthermore, the e3value group at VU University Amsterdam significantly contributed to the modeling of value flows in networks. In doing so, the group proposes generic solutions for common control problems, so called 'control patterns' [14]. The underlying transactional design is helpful to describe and depict direct control relationships.

Both related research streams, however, lack the ability to model control complications with respect to network self-organization and stakeholding power of platform providers in dynamic PaaS ecosystems. Moreover, the proposed approaches do not allow investigating on network complications, i.e. on the implications of network dynamics. A third major point of differentiation, presented in this paper, lies in our specific attribution of implicit relations with groups of participants that neither require explicit denomination, nor quantification, nor modeling e.g. a big group of unknown potential service consumers or service providers, addressed by a PaaS providers.

Grounding theory to our research can be found in system theory [15], control theory [16], as well as Sterman's works on dynamic markets [17].

III. Governance in PaaS Ecosystems

Substantiated through their survey, [2] highlight the complexity of Cloud service landscapes with its immanent difficulty to "be centrally controlled by a single party without severe scalability implications." We will now break it down into major control respectively major network complications to shed light into this stated complexity. In doing so, we provide the foundations required to identify the requirements of a notation for dynamic networks effects in PaaS ecosystems.

A. Major Control Complications

Caused by the inter-organizational characteristics of Cloud-based service compositions, the stakeholding power of a protagonist (the participant whose view point is modeled, e.g. platform provider) on other participants is limited. We identified the following complications [3, 5, 7]:

1) Zones of controllability: The protagonist's stakeholding power diminishes with every additional level of indirection as it steps further away from its proprietary domain. The center domain allows highest observability and direct control on all transactions and activities; direct enforcement is possible. The latter implies that quality can be assured, either through directives or through enforcement in this area [for details see 4, 7]. We thus speak of the 'Control Area'. The consecutive zone describes the area where the protagonist can only indirectly influence activities through targeted information and incentives. We speak of the 'Influence Area'. The outer zone embraces those participants who are indifferent or opposed to the protagonists activities; we speak with reference to white noise in control theory of the 'Noise Area'.

2) Self-Organization: Autonomous participants self-organize over time, adapting their value in pursuit of long-term profit optimization. This causes a dilemma as self-organization in the Influence Area: Nourishing external creativity for the price of reduced observability and controllability compared to their dependent counterparts within the boundaries of the Control Area.

B. Major network complications

As a consequence of the PaaS ecosystems' dynamics, three major network complications arise [4, 18, 19]:

1) Dynamic Loops: The PaaS/SaaS scenario offers a multitude of possible loops, also referred to as 'network externalities'. Examples for enforcing loops are economies of scale, where a rising number of users increases PaaS attractiveness to potential users, as well as to service providers - invigorating their willingness to develop and deploy services and, hence, to complement the PaaS overall service offer. In the following, we will call them 'complementors'. Stimulated service provision in reciprocity causes a complementarity effect that increases platform attractiveness to service consumers. On the other side, negative loops may exist that are fed by competitive dynamic activities, digging off demand-sided attractiveness. In this paper, we use the term 'dynamic process' when speaking of an overall network system that may include several dynamic loops.

2) Base Value: Once, a dynamic process is set off, loops may self-fertilize and potentially scale towards a market dominating position ('lock-in situations') over time (e.g. the stable position of Apple's App Store, making it particularly difficult for challengers to successfully introduce competitive solutions). However, the same inertia is faced when trying to initiate system dynamics. Prerequisites to success is a fundamental value proposition ('base value') to attract a critical mass of first movers among SaaS providers and consumer that eventually ignite a dynamic loop.

3) *Scalability:* Once a dynamic loop is ignited, prerequisite for its proliferation are technical environments that are able to quickly adjust to the increasing demand. At

the same time occurring drop-offs demand to quickly release resources. We refer to both capabilities as ‘scalability’.

IV. THE NOTATION

The design goal of DYNO is to enable business analysts and service engineers to layout their PaaS infrastructure in accordance to the context of the inter-organizational characteristics of the PaaS ecosystem. We therefore introduce a notation that is able to circumscribe weaknesses (e.g. lack of base value) and leverages on opportunities (e.g. network loops). Modeling PaaS ecosystems on the level of network effects allows conceiving business processes in a targeted and effective way that integrates abilities to react and to influence on external dynamics in a sustainable way.

We thus formulate the following basic requirements that a notation for dynamic network effects (DYNO) in PaaS ecosystems needs to fulfill:

- R1: All PaaS ecosystems participants (service providers, service consumers) and their respective activities must find a representation in DYNO models.
- R2: Models must incorporate relationships that reflect both transactions and influences between PaaS ecosystem participants;
- R3: Models must be able to incorporate working points to exert control and respective control mechanisms for PaaS providers;
- R4: Each ecosystem participant (service providers or consumers) must be ascribed to a non-ambiguously defined zone of controllability;
- R5: All ecosystem participants may be ordered in groups.

Following the guidelines of Modeling (GoM) we have to assure syntactic correctness [20] for all PaaS ecosystems, modeled with our notation. To do so, it is ‘indispensable to have an explicit (documented) meta-model’ [20]. Therefore we created a meta-model that captures structure (participants, relationships) and behavior (activities), consistent with inter-organizational characteristics of a PaaS ecosystem. Details to this meta-model can be found in [21].

The notation is designed to be used by business analysts when designing PaaS-arrangements. At the same time it should create a point of departure for service engineers towards a context-optimized service design. We hence require a depiction that is equally intuitively understood by users with a business background, who are traditionally familiar with Flow-Chart-oriented formats, and by users with an IT background, rather acquainted with notations like UML or Petri-Nets. We chose to design our symbols akin to the BPMN 2.0 conversation elements [19]. However, the symbols applied in this paper shall be considered preliminary, as user-tests are ongoing within business and IT communities in order to search for the most intuitive depiction for interdisciplinary user groups.

1)Protagonist Control – As defined earlier, the ‘protagonist’s view point and control center is not explicitly modeled. Throughout the design process of the notation, we learnt that the DYNO will be most expressive, once we do not model this element. Instead, we visualize its control activities at the respective control points.

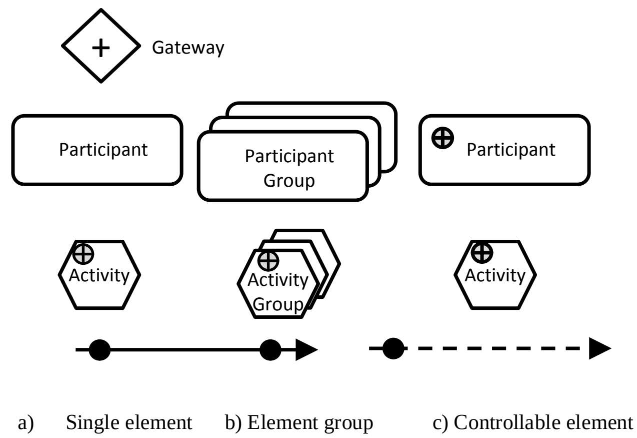

Figure 2. Controlled Elements

2) Gateway – Similarly to BPMN 2.0 Process diagrams, we apply the diamond symbol with included ‘+’. Each Gateway always behaves according to a synchronizing merge, awaiting all incoming edges to complete before triggering the (single) outgoing edge (see fig. 2).

3) Participant – Again we adopt BPMN’s conversation symbols for Participants (see fig. 2). The following two extensions to the diagram symbol are made:

- The existence of the attribute ‘controllability’ is depicted through a screw head, always located in the upper left corner of the Participant symbol.
- Participants groups are depicted through 3 overlaid Participant symbols. In cases of controllability = true, the screw head is placed on the upper Participant symbol.

4) Activity – We choose the octagon, as used to depict communications in BPMN 2.0 Conversation (fig. 2) to visualize activities in DYNO.

- As by definition, an Activity can only take place in the Control Area. The symbol mandatorily carries the screw head symbol in the upper left corner.
- In unison to Participant Groups, Activity Groups are depicted through 3 overlaid Activity symbols.

5) Relationships – Relationships are displayed through directed edges. A transaction is depicted through a

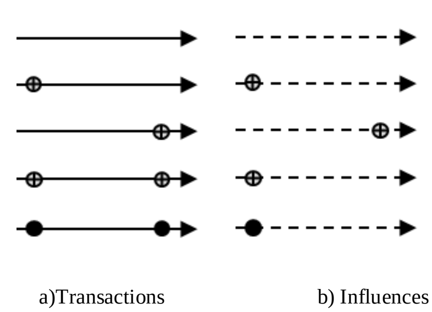

Figure 3. Relationships and Control Mechanisms

continuous arrow, an influence through a dashed arrow. The direction is indicated at the target-side through a filled arrow head. Similarly to the controlled elements, a screw head is placed at the source- or target side on the edge, whenever the attribute controllability is set to the value ‘ true’ (fig. 3).

6) Control Mechanism – Wherever a control point is controlled through protagonist control with one or several control mechanisms, we draw the screw head solid (fig. 3).

7) Zones – The Control Area is illustrated through a continuous grey line, the Influence Area through a dotted line of the same strength. By definition, the Noise Area is all outside space. A division is depicted through a grey box of smaller thickness. An activated scalability attribute is symbolized through an infinity symbol, located in the top left corner of the box. For the Division Group, we follow the same procedure as with groups of Controlled Elements. Descriptive text in all zone-symbols is equally kept in grey-scale.

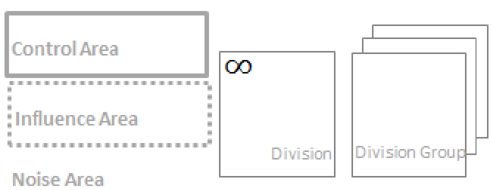

Figure 4. Zones

V. EXAMPLE

Fig. 5 illustrates the notation at a simple example of a PaaS.

VI. CONCLUSION AND OUTLOOK

This paper introduced a notation enabling business analysts and service engineers to design platforms in the dynamic context of inter-organizational cloud-based networks (e.g. PaaS ecosystems). The DYNO notation attributes explicit attention to the control challenges of stakeholding power and self-organized autonomous service ecosystems. Secondly, it takes account of network

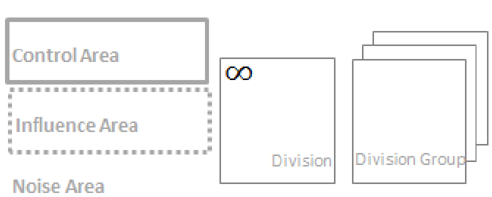

Figure 4. Zones

complications, i. e. network dynamics and base value contribution. The case study of an international e-Learning platform exemplifies how the notation can be used to sustainably optimize a PaaS ecosystem set-up from the platform provider’s perspective, when considering relevant environmental conditions and when applying available control mechanisms.

The next step in our ongoing research is the implementation of an online editor for DYNO. The editor shall allow going beyond a pure depiction of DYNO’s syntax and semantics and allow for loop discovery, suggestions in system optimization and the inclusion of simulative aspects.

REFERENCES

[1] L.J. Kirsch, “Portfolios of control modes and IS project management,” Information Systems Research, vol. 8, No. 3, Sep. 1997, pp. 215 – 239.

[2] A. Lenk, M. Klems, J. Nimis, S. Tai, T. Sandholm, “What’s inside the cloud? An architectural map of the cloud landscape,” Proc. Cloud’09, Vancouver, IEEE Press, May 2009, pp. 23 – 31

[3] R. Fischer, U. Scholten, S. Scholten, S. Tai, S. “Information-based control of service-enabling ecosystems,” Proc. Second International Workshop on Enabling Service Business Ecosystems, Athens, 2009, pp. 1 – 14.

[4] U. Scholten, R. Fischer, D. Bojkov, N. May, “Supply chain control building on emergent self-organizing effects,” Proc. 4th Academic Symposium on Supply Management, Würzburg, Springer, May 2011, pp. 311 - 335.

[5] S. Scholten, U. Scholten, “Platform-based innovation management: Directing external innovational efforts in complex self-organizing platform ecosystems”. Proc. PICMET 2010, Bangkok, IEEE, Jul. 2010, pp. 1 – 12.

[6] Scholten, U., Fischer, R.; Zirpins, C. “Perspectives for Web service intermediaries: How influence on quality makes the difference,” Proc. 10th International Conference on Electronic Commerce and Web Technologies (EC-Web), Linz, pp. 145-156.

[7] S. Scholten, U. Scholten, R. Fischer, “Composite solutions for consumer-driven supply chains,” Proc. Supply Management Symposium’10, Würzburg, Supply Management Research, Springer, 2010, pp. 277 – 301.

[8] R. Fischer, U. Scholten, S. Scholten, “ A reference architecture for feedback-based control of service ecosystems,” Proc. 4th International Conference on Digital Ecosystems and Technologies DEST’10, DUBAI, IEEE, 2010, pp. 1 – 6.

[9] S. Tai, J. Nimis, A. Lenk, M. Klems, “Cloud service engineering,” Proc. 32nd ACM/IEEE International Conference on Software Engineering, vol. 2, ACM, 2010, 475 – 476.

[10] N. May, U. Scholten, R. Fischer, “Towards an automated gap analysis for e-Service portfolios,” Proc. 8th International Conference on Services Computing SCC’11, Washington, 2011, in print.

[11] S-Cube Project, www.s-cube-network.eu, retrieved August 2011.

[12] M. Bitsaki, O. Danylevych, W.J. van den Heuvel, G. Koutras, F. Leymann, M. Mancioppi, C. Nikolaou, M. Papazoglou, “An architecture for managing the lifecycle of business goals for partners in a service network, Towards a service-based Internet,” Springer, 2008, pp. 196 – 207.

[13] O. Danylevych, D. Karastoyanova, F. Leymann, “Service networks modelling: A SOA \& BPM standpoint,” Journal of Universal Computer Science, vol. 26, No. 13, 2010, pp. 1668 – 1693.

[14] V. Kartseva, J. Gordijn, Y.H. Tan, “Value-based design of Networked enterprises using e3control patterns,” 15th IEEE International Requirements Engineering Conference, 2007, pp. 91 – 100.

[15] I. Prigogine, G. Nicolis, “Self-Organization in nonequilibrium systems: From dissipative structures to order through fluctuations,” Wiley, New York, 1977.

[16] O. Föllinger, “Regelungstechnik,“ Hüthig-Verlag, 1990.

[17] J.D. Sterman, “ Business dynamics, systems thinking and modeling for a complex world,” McGraw-Hill, 2000.

[18] C. Shapiro, H.R. Varian, “Information rules, A strategic guide to the network economy,” Harvard Business School Press, 1999.

[19] Object Management Group, “Documents Associated with Business Process Model and Notation (BPMN) v 2.0,” Jan 2011, http://www.omg.org/spec/BPMN/2.0J.

[20] Becker, M. Rosemann, C. von Uthmann, “Guidelines of business process modeling,” Business Process Management, Springer, 2000, pp. 241 – 262.

[21] Business Process Technology research group at the Hasso-Plattner-Institute, http://bpt.hpi.uni-potsdam.de/, retrieved Aug. 2011

[22] U. Scholten, R. Fischer, C. Zirpins, S. Scholten, “DYNO – the Dynamic Network Notation,” Technical Report, Karlsruhe Institute of Technology, 2011.