Assessing Service Deployment Readiness using Enterprise Crowdsourcing

Maja Vukovic, Jim Laredo, Yaoping Ruan, Milton Hernandez
IBM T.J. Watson Research Center
Hawthorne, NY, USA
{maja, laredoj, yaoping.ruan, miltonh}@us.ibm.com

Sriram Rajagopal
IBM Global Technology Services
Chennai, India
srirraja@in.ibm.com

***Abstract*— Service delivery centers are shared hosting environments where a variety of types of IT systems and applications for multiple customers are managed. Multiple aspects affect the deployment in shared environments, including applications supported, network topology, security requirements, etc., knowledge of which is distributed across multiple delivery experts. An accurate snapshot of the delivery environment is a key factor for successful deployment of new service offerings.**

Enterprise crowdsourcing is a process where a group of network-connected experts solve problems. We employ enterprise crowdsourcing to engage 250 service delivery experts to facilitate early discovery of key information about 1300 customers' environments, as part of the deployment of a novel identity access management service. We propose a model for deployment complexities, and use the discovered knowledge to expose the service deployment complexities. We discuss how our approach efficiently deals with requirements and infrastructure detection and how it quantifies readiness of service deployment.

***Index Terms*—Service Delivery, Deployment, Crowdsourcing**

I. INTRODUCTION

As the complexity and heterogeneity of IT systems increased, IT service hosting model evolved from isolated environments in 1980's to a global delivery model in 2000's. Nowadays 100000 of system administrators support 1000s of processes, running on thousands of IT systems on behalf of thousands of customers worldwide. The multifaceted, shared and dynamic nature of service delivery centers makes the deployment of new service offerings challenging. For example, network topology, sever access applications and authentication mechanisms are some of the aspects in which heterogeneity is reflected across customers.

A set of acceptance criteria defines a successful rollout of a service offering in hosting environments. To enable efficient project execution, deployment teams require knowledge about a) unsupported capabilities and b) additional development require-ments, arising from diversity of customers' environments. This knowledge about IT infrastructure (e.g. applications, servers and governance processes) is held in collective possession of delivery teams (including current, past and new members). Any delays in discovering the customer's environment, may lead to new requirements and impact the deployment timeline and cost.

To increase efficiency and quality of service deployment we propose integration of enterprise crowdsourcing [1] for environment profiling. This approach enables us to quickly reach out to a set of distributed team members to collectively gather or verify knowledge about a particular it asset or process. It further facilitates incremental updates to the known complexities and/or environment snapshots arising from successive service deployments. In parallel, we define a process to categorize deployment complexities based on platform and operational characteristics of the service delivery platform for a particular set of customer(s). This information enables automated computation of the complexities and allocating deployment and development tasks to the globally distributed teams.

This paper makes following three contributions:

1. We present a framework and method for automated categorization of requirements complexity to assess deployment readiness in large-scale, shared, hosting environments. The framework provides a mechanism for crowd-driven knowledge gathering and means of defining project specific template-based complexity definitions (in service delivery centers).
2. We describe how we implemented a system that embodies complexity templates to support automated categorization with pre-defined classes of deployment complexity in IT delivery centers, corresponding questionnaires and rules for processing them.
3. We present a model to provide a quantitative evidence of deployment readiness.

The rest of the paper is structured as follows. Section 2 provides more details about the global delivery model and service deployment challenges in large-scale, shared hosting environments. Section 3 describes the typical deployment process and our proposed model for assessment of deployment readiness. Section 4 explains the pro-cess of enterprise crowdsourcing and how we extended enterprise crowdsourcing service BizRay [1] for purposes of this work. Section 5 presents two variants of our approach to assess deployment readiness for a novel identity access management service [2], and how enterprise crowdsourcing has accelerated knowledge discovery. Section 6 discusses our experiences. Section 7 puts our work in the context of state of the art in enterprise

978-3-901882-50-0

crowdsourcing and deployment complexity assessments. Finally, Section 8 concludes and presents our future work.

II. Deployment Challenges in Delivery Centers

Service delivery centers have in the past decade adopted a global delivery model [7], based on the factory prototype, to efficiently provide a stack of IT services. This model as a result, the services delivered take form of globally standardized processes, with adaptive dispatching where activities of services operations are assigned to system administrators (SAs) and subject matter experts (SMEs) based on the domain and complexity of work. SMEs are grouped into "pools" based on a common competency that is executed for a set of customers. The knowledge about infrastructure, user practices, and policies is captured into common repositories. This knowledge is used to manage on-going operations, ensure compliance posture, etc.

Deployment of new service offerings in these shared environments is challenging for a number of reasons arising from the complexity of the delivery environment and the operating processes. First of all, each customer has different infrastructure (e.g. IT systems, O/S versions) that leads to the environment heterogeneity. Furthermore, the servers may get decommissioned over the time, O/S upgraded, applications added, etc., making the environment evolve over the time. Similarly, there may be custom governance processes to address specific customer (and/or industry) requirements to conform to the regulations when managing the systems Often times in conflict with current practices and directives for productivity improvements. Secondly, deployments of service offerings are often done in isolated manner, allowing for little or no reusability of knowledge about customers environment, and limiting the share of deployment experiences and lessons learnt between the different deployments. For example, if a service offering was deployed to a set of delivery pools and customers, capturing this experience is critical to understand the suitability and readiness of the same pools and customers for deployment of another service offering. Third, organizational structures may get in the way, as their missions are not always aligned. A corporate directive may be mandated at the global level, but the implementation of the deployment relies on the different geographies, with their specific regulations, and ultimately at the account levels, where budgets are allocated and managed for each offering. One way to coordinate these organizations is to be able to understand the complexity and impact of each deployment and roll up into a global view treating all deployments as a portfolio and helping in the prioritization of the different accounts.

Some offerings consist of multiple phases. It is not unusual to see a requirements phase, followed by a development effort with the corresponding testing, and finally the actual deployment of the offering assets, both generic ones, and those customized for the particular account. On each step it is important to identify the SMEs that have the information or skill or know where to go to retrieve it, or who to ask. Without the data upfront it is very likely that it will result in an incomplete implementation with not all the actual requirements covered and easily jeopardizing the timely deployment.

III. Crowdsourcing for Service Deployment Readiness

This section uses the deployment of ID management service, called IAM [2], as a case study to describe the service deployment process and challenges. It presents our proposed approach to assessing service deployment readiness.

A. Deploying ID Management as a Service

IAM provides privileged access control to service delivery teams and is being deployed to all delivery centers with the strategic goal of increasing security and productivity while reducing cost arising from repetitive ID management tasks. IAM service provides a role-based access control mechanism separate from the server endpoint's operating systems. It consists of an ID vault with definition and assignment of roles, individuals, and server endpoints. A single sign on password broker performs credential retrieval from the ID vault and monitors the logon application and injects the retrieved credentials based on a pre-defined control flow. The ID vault and single sign on policy server are shared by multiple delivery centers. Challenges arise from capturing the existing logon workflow to determine appropriate platform to host the single sign on agent and develop the correct control flow.

B. Assessment of Deployment Readiness

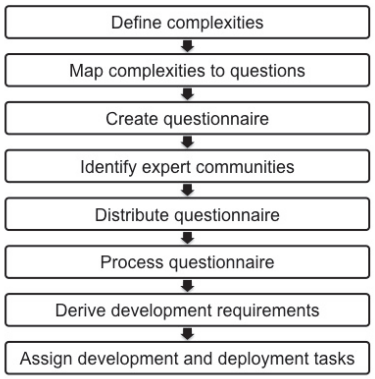

Fig. 1. Assessing Service Deployment Readiness

Figure 1 describes the proposed process of assessing service deployment readiness. We define deployment complexities and map them into a set of questions. We identify the target communities and distribute the questionnaire to the focal person of each team (pool) using a crowdsourcing service [1]. Questionnaires may be pre-populated from the enterprise data sources to simplify the human effort; and can be reused for multiple deployments to incrementally update the data. Incomplete questions can be delegated to experts or will trigger follow-ups. We use the crowdsourcing responses to get the complexities and derive development and deployment tasks.

C. Deployment Complexity and Questionnaire

Figure 2. shows three types of IAM deployment complexities: application, network topology, and business operation. The application category captures properties of applications, such as authentication method. Network topology maps how the endpoints are connected, e.g. VPN. Business operation shows dependency on other systems, such as ticket management.

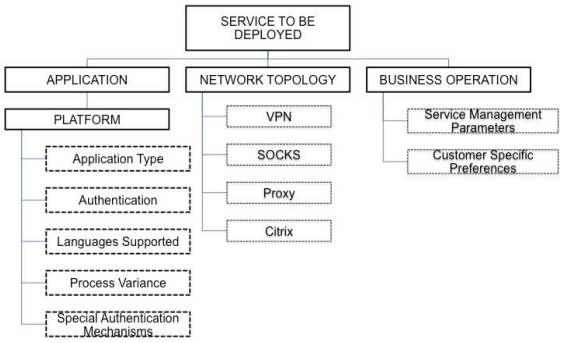

Fig. 2. Complexity hierarchy

Each complexity element is mapped to one or more questions, and assigned a quantitative score representing effort (1= lowest to 10=highest) required for the specific deployment challenge, as shown on Table 1.

TABLE I. DEPLOYMENT COMPLEXITIES

Type	Complexity
Hop server is required for Citrix and RDP	3
Mainframe access via PCOMM	3
Mainframe access via PCOMM (OS400 hub)	3
New language requirement	5
RDP Hopping: SAs log into domain controller or other Windows server before opening new RDP or Putty session	5
AS400 Hop access via PCOMM	6

IV. ENTERPRISE CROWDSOURCING USING BIZRAY

This section summarizes the core capabilities of crowdsourcing service BizRay [1] and extensions made to support the deployment of the IAM security service.

A. Crowdsourcing Process

BizRay captures knowledge requests in the distributed questionnaire artifact consisting of one or more sections, each of which contains one or more questions about the business entity, such as a server or a business application. BizRay manages its lifecycle, similar to a workflow system and facilitates delegation of requests and their subtasking, as shown in Figure 1. More than one expert (user) can complete each questionnaire instance. If the information gathered is incomplete or unidentified, the user can forward the request to another expert, asking for their input. As experts contribute their knowledge, the system keeps track of their identity resulting in the formation of micro communities around the object of that inquiry. BizRay provides a mechanism to send out reminders and escalations to users who did not respond to the initial request, as well as point-based incentives framework. Users are awarded a customizable number of points (or fractions thereof) for each completed task or for forwarding the request to an expert capable of completing it.

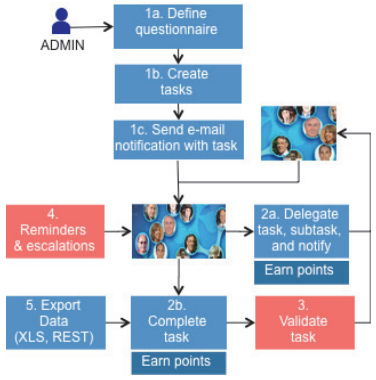

Fig. 3. Enterprise Crowdsourcing using BizRay [1]

B. BizRay extensions for deployment readiness process

Questionnaires in BizRay comprise of sections – each of which comprise a number of questions. All sections of a questionnaire are visible to the user by default. For the purpose of our deployment we required a flexible set of sections, as one or more may be relevant to a given set of pools, such as a section about AS400, UNIX, Windows and Mainframe. Users may answer one or more sections.

To enable optional and interdependent sections, BizRay supports a concept named "hide-able" sections, as shown in Figure 4. "Hide-able" sections are not displayed by default. Each "hide-able" section has an associated button that once clicked renders the section visible to the user. Even if one or more questions in a "hide-able" section are assigned to the parent task owner, the section is hidden until the user chooses to answer the questions and activates the section. However, if one or more questions in a "hide-able" section are assigned in a sub-task, such a section is rendered visible when the sub-task owner opens the task. When one or more questions are answered in a "hide-able" section, the section is rendered visible to the parent task owner as well as the sub-task owners.

The user can hide the section by clicking on the "Remove" button. The hidden section can be activated again by clicking on the button corresponding to this section.

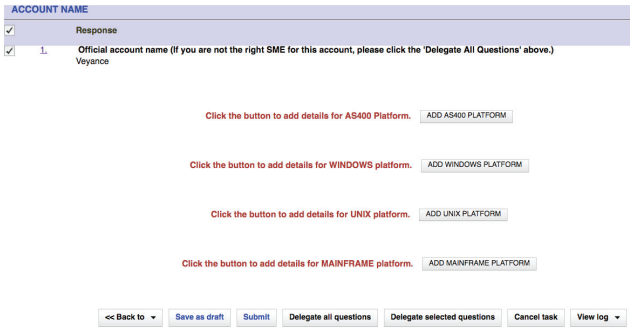

Fig. 4. Before hide-able section is opened

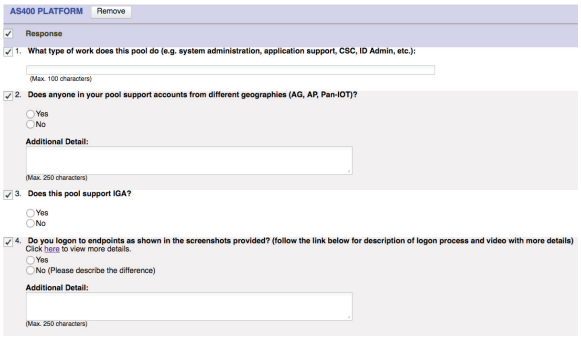

Fig. 5. Activation of "hide-able" AS400 section and embedded help-links

Interdependent sections can be created by setting the property "Hide button if previous section is hidden" to true. In this case, the button to activate the section itself is hidden until the user does not activate the preceding "hide-able" section. Once the preceding section is visible, the button to activate this section is rendered. Questions in "hide-able" sections can be optional or mandatory. If a "hide-able" section is not activated or if an activated "hide-able" section is removed or closed, the questions belonging to that section are not mandatory

BizRay enables embedding help links for each question, highlighted in Figure 5. These links point to help files (e.g. videos). For the purposes of our work, this is was an important feature to guide users and avoid overhead in human support.

V. Deployments

This section describes two enterprise crowdsourcing campaigns designed to uncover the details of customer's environment, facilitating deployment of IAM. Each questionnaire instance represents the information for a single "pool-account" combination. Multiple SMEs from a single pool may contribute to the same questionnaire. This can be achieved by "delegating" some or all questions of the questionnaire to multiple SMEs. Each SME can answer questions for any of the supported platforms. A deployment lead delegates the questionnaire to delivery account SMEs, as in Figure 6., and thereby initiates the assessment process.

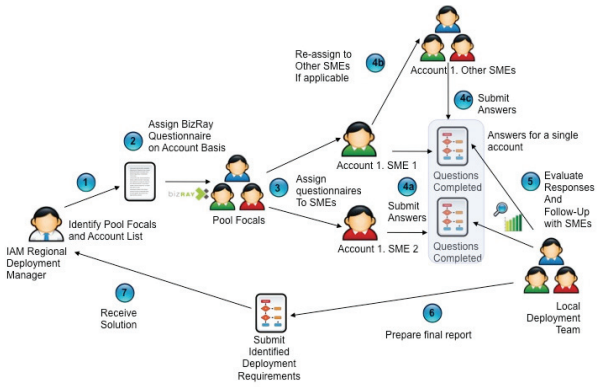

Fig. 6. IAM Requirements Gathering Process using Crowdsourcing

Figure 7 provides an architectural overview of the entire system that supports the deployment readiness process. The entry point to the system is the deployment request and its properties. The deployment request specifies the complexities for the deployable service and its mappings to questions. In addition it contains the target which and accounts in a delivery environment. Questions may be prepopulated from enterprise data repositories. Expertise manager tracks the responses and builds the expertise profile, by associating users to different infrastructure, operations and user practices that they reported. The output of the overall process is assignment of experts to discovered development and deployment tasks.

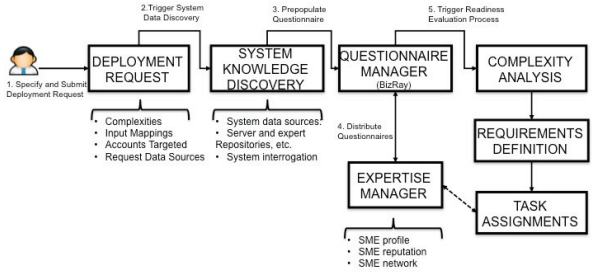

Fig. 7. System Architecture

A. Customer Profile Questionnaire #1

The first questionnaire was designed to capture the complexities of logon methods for 128 pool-accounts. It consisted of 250 questions in 15 sections, asking for details about managing systems, scripts, applications, logon applications, change management tools and Web interfaces used by each pool-account, shown in Table 1. We also requested server inventory, which was provided by 109 accounts. Seven sections were custom (Windows and Unix based) and used to derive complexities shown in Table 2. 89 questionnaires were completed, and 109 responses were saved in total (partial and complete by 110 unique respondents, taking 13 days on average to complete the questionnaire.

TABLE II. BREAKDOWN OF APPLICATIONS USED IN QUESTIONNAIRE 1

Data captured	Number of accounts
Managing systems	31
Scripts used (15 scripts discovered)	11
Logon applications (55 non-unique, 45 unique reported)	36
Application details	13
Web interface details	18
Change management details	95

TABLE III. COMPLEXITIES FOR NETWORK TOPOLOGY IN QUESTIONNAIRE 1

Complexity	Num. of accounts
Windows case #0: No hop/Citrix server but only VPN	27
Windows case #1: Hop server is able to access IAM	21
Windows case #2: Hop server is the same as domain controller	2
UNIX case #0: No Hop/Citrix/SOCKS server but only VPN	9
UNIX case #1: User uses Putty on Citrix server to access IAM	6
UNIX case #2: User uses access endpoints by ssh onto proxy/hop server first	5
UNIX case #3: User uses access endpoints via SOCKS server	14

B. Customer Profile Questionnaire #2

Second questionnaire was a simplified version of first and grouped in 4 sections, one for each of the following platforms: AS/400, Mainframe, Win and Unix, as shown in Figures 4-6. We developed an analytics module to automatically categorize collected data, and derive complexities, such as uncovering pools that are not using standard set of logon applications. 1300 questionnaires were deployed worldwide. Table 4 shows the distribution of platforms supported, based on completed questionnaires.

TABLE IV. NUMBER OF ACCOUNTS AND PLATFORMS SUPPORTED

Platform(s)	Num. of accounts
AS400 only	78
MF only	129
Unix only	257
Windows only	393
Multiple platforms	254
No platform provided	54

Table 5 shows the complexities for three pool-accounts. Responsed about the logon applications used for each platform, and logon methodology (pertaining to topology) were processed to derive complexity categories. For example, account 2 in contrast to other two accounts supported by pool 1 has a requirement of German language that increases the complexity of deployment. Logon applications were used to determing whether pool-account is standard or not (e.g. uses standardized tools). For example, RDP and Putty are standard, while Dameware and VSphere are not. Using this approach we have discovered 62 non-standard pool-accounts.

TABLE V. QUESTIONNAIRE 2 COMPLEXITY REPORT

Pool Code	Pool-Code-1	Pool-Code-1	Pool-Code-1
Account Name	Account 1	Account 2	Account 3
Platform	Windows	Windows	Windows
Logon Apps	RDP	RDP	RDP
Logon Method	RDP from local desktop	RDP from local desktop	RDP from local desktop
Topology	Direct	Direct	Some accessible from terminal servers
Non-English	No	*Yes - German*	No
Windows Domain	*Yes*	*Partial*	*Partial*
Complexity	**6**	**11**	**6**

C. Deployment Results

Figure 8. depicts the user behavior in completing the questionnaires. We observe that the first response for questionnaire 1 occurs 13 days after it's roll-out. In contrast, for questionnaire #2 across all three regions (EMEA, Americas and India) we obtain responses within the first day of deployment. 106 tasks were reassigned (100 of which were parent tasks) for first questionnaire. For questionnaire #2 total of 1300 tasks were assigned to SMEs, 1223 were reassigned out of which 1215 were parent task, and 1040 were completed by 494 unique SMEs. Similar response rates for questionnaire #2 occur across all the regions. On average in all geos within the first day of deployment 52% of the tasks were completed, indicating that the time requirements were reasonable.

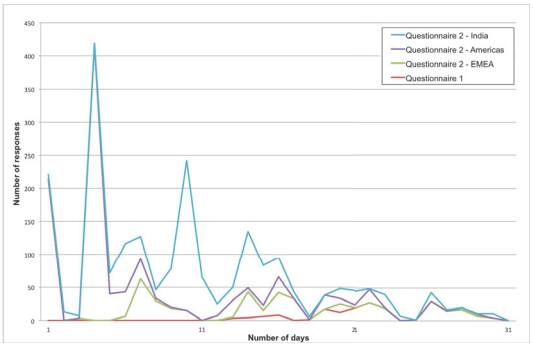

Fig. 8. Distribution of number of daily responses for two deployments

VI. DISCUSSION

Delivery experts have limited time available on their hands, due to the utilization rates that they have to meet, and a growing number of requests that they need to handle. Even though that the questionnaires are mandated by the management, getting SAs time to complete the questionnaires is challenging. During the first deployment it became evident that 250 questions was overwhelming for the users. Even though the 250 questions are designed to target multiple users, each completing a subset of questions, the common perception

was that there are too many questions. So in our second deployment, we divided the questionnaire into different categories based on the system platforms accessed by each SME, further facilitating effective delegation (e.g. questions about AS400 can be routed to the AS400 specialist).

A number of technical and usability challenges were uncovered. In the first deployment, it became clear that the video clips were unusable in certain geographies due to the bandwidth limitations. We have replaced this by set of screenshots, embedded in the help link. We further encountered issues with the different browsers being used, and have introduced an auto-save capability to ensure that the questionnaire data is not lost. We have worked closely with SAs, explaining any misleading questions in the questionnaires – for example many SAs were not familiar with security policies of a given account.

VII. Related Work

Our methodology is based on experiences in deployments of solutions in large-scale hosting environments and challenges in quantifying the deployment complexities and early discovery of new requirements. Diao and Keller [4] have introduced a framework for measuring the complexity of IT system configuration and IT service management processes. This framework defines complexity along several dimensions including execution, coordination, and memory. For example, the execution complexity is characterized by the number of tasks and context switches between tasks, as well as the degree of automation: automatic, tool-assisted, and manual. In contrast, our work tackles the problem of deployment readiness for service solutions in multi-customer environments, their infrastructure discovery and dependencies.

Bose et al. [4] describe a global delivery framework model implemented by a large global service provider, where pools of experts grouped by competency manage customers' environment. They discuss challenges in deploying a system for global processes whose lifecycle spans multiple delivery centers globally and across a set of customers. Our work has been deployed in such a GDF-based environment. The usage of enterprise crowdsourcing has enabled rapid discovery of pools and their operations, as well as tooling used for system management to drive the standardization efforts.

Solution deployments follow the traditional project management model. A project manager works with geography leads that manage delivery teams are rolling-out the solution. In IAM initially deployments were performed without technical assessments and caused late discovery of new requirements. For example, as the teams were about to deploy the access agents to system administrator's desktops they would discover an unsupported application. The deployment would be on-hold until the developer builds a new profile. Also as the process was not globally coordinated, different geo teams brought up same requirements.

On-going research in the system administration discipline is focusing on establishing cost models and complexities of service delivery [5,6].

Rise of Web 2.0 has spurred the design of a variety of applications that employ crowdsourcing. Vukovic and Naik [7] engage in-house expert communities to obtain snapshots of current hosting environment to drive IT management such as asset management, maturity assessment, or migration planning. We use crowdsourcing to derive deployment complexities in a large-scale global solution roll-out.

VIII. Conclusion

To increase efficiency and quality of service deployment we proposed integration of enterprise crowdsourcing with automated environment profiling. Enterprise crowdsourcing enables us to quickly reach out to a set of distributed team members to collectively gather and verify knowledge about an IT asset or process. We defined a method to categorize deployment complexities based on platform and operational characteristics of the service delivery platform for a particular set of customer(s). This information enables automated computation of the deployment complexities. Crowdsourcing facilitates incremental updates to the known environment complexities arising from successive deployments.

The benefits of this approach are threefold: a) it provides quantitative evidence to assess readiness of deployment; b) it facilitates governance decision making for development and c) provides reference for development requirements. Our future work will involve development of a decision-theoretic model to efficiently route crowdsourcing tasks based on the tradeoffs in time and cost of the completion of the entire service deployment process.

References

[1] J. Laredo, M. Vukovic, S. Rajagopal, "Service for Crowd-Driven Gathering of Non-Discoverable Knowledge". Proceedings of International Conference on Service Oriented Computing (ICSOC '11). December 2011.

[2] K. Bhaskaran, M. Hernandez, J. Laredo, L. Luan, Y. Ruan, M. Vukovic, "Privileged Identity Management in Enterprise Service-Hosting Environments", Proceedings of Network Operations and Management Symposium (NOMS 2012)

[3] Yixin Diao, Alexander Keller: Quantifying the Complexity of IT Service Management Processes. DSOM 2006:61-73

[4] A. Bose, A. Heching, S. Sahu, "A Framework for Model-Based Continuous Improvement of Global IT Service Delivery Operations," IEEE International Conference on Services Computing, 2008

[5] A. Couch, N. Wu, and H. Susanto, "Toward a Cost Model for SystemAdministration," in Proc. 19th Large Installation System Administratiom Conference, San Diego: USENIX, Dec. 2005.

[6] D. Patterson, "A Simple Way to Estimate the Cost of Downtime," in Proc. 16th Large Installation System Administration Conference, Philadelphia: USENIX, 2002.

[7] M. Vukovic, V. Naik, Managing Enterprise IT Systems Using Online Communities. IEEE Services Computing Conference (SCC) (2011).