Managing Architectural Change and Governance in Nascent Platform Ecosystems

Richard Tee
LUISS
Department of Management
rtee@luiss.it

Jason Woodard
Olin College
Engineering and Entrepreneurship
jason.woodard@olin.edu

Ben Eaton
University of Surrey
Business School
b.d.eaton@surrey.ac.uk

Abstract
Our paper focuses on nascent platform ecosystems, analyzing how firms take different approaches to navigating issues of change and control and how this subsequently affects overall generativity of the firms’ respective projects. In particular, we emphasize that focal firms not only take different approaches to stimulating the development of the ecosystem’s components and complements, but that this process can vary in sequence and degree. Contrasting with existing work, we show that in the case of nascent platform ecosystems, flexibility of the core architecture may also be important to promote generativity. By analyzing contrasting processes in which firms navigate the paradoxes of change and control, our paper helps explain why such dynamics play out differently during the nascent stages of a platform ecosystem.
Managing Architectural Change and Governance in Nascent Platform Ecosystems

1. Introduction

Platforms such as Facebook, Android, and Square are both powerful and dangerous: they enable the generative dynamics of digital innovation (Zittrain 2008; Yoo et al. 2010) while presenting a myriad of strategic challenges for their owners and the members of their surrounding business ecosystems (Thomas et al. 2014; Gawer 2014). Tilson et al. (2010) conceptualize these challenges as paradoxes; focusing on the issues of change and control. The paradox of change refers to the way in which stability and flexibility are managed. Platforms need to be sufficiently stable to allow ecosystem partners to contribute to them, yet flexible enough that they can evolve and adapt to the needs of a growing installed base. The paradox of control refers to the degree to which the governance of a platform is centralized around a focal firm versus distributed among a wider ecosystem. Such control can have important competitive implications, as platforms often give rise to technological or strategic control points (Baldwin 2015). At the same time, control cannot be exerted too strongly, as other actors may respond by rejecting or subverting the platform (Tilson et al. 2010).

Of course, the challenge of any paradox is that there is no straightforward way to resolve it. Therefore, we focus instead on how organizations might approach these paradoxes differently, and explain their subsequent impact on promoting or inhibiting the generativity of their ecosystems. In particular, we tackle these paradoxes by taking a contingent approach, focusing on when and where a platform owner might fruitfully choose to tackle them. Regarding the paradox of change, when is stability or flexibility more important (earlier or later in the evolution
of a platform ecosystem)? In terms of the paradox of control, where in the architecture (at the core or the periphery) is centralized or distributed governance more significant?

These concerns are of particular interest in the early, nascent stages of platform ecosystems. Previous research (Baldwin and Woodard 2009; Eaton et al. 2015) has shown that established platform architectures benefit both from stability in the platform core and flexibility in the periphery. Therefore, firms typically need to find ways to manage a platform’s core components, while also stimulating others to develop peripheral complements for the core architecture (West 2002; Boudreau 2010). However, our understanding of how these arrangements are configured in the context of nascent platform ecosystems is not well developed. We therefore address the following research question: “How do firms manage architectural change and governance in order to promote generativity in nascent platform ecosystems?”

We adopt a comparative case study approach (Yin 2009) and analyze two related software platforms, OpenStack and CloudStack, which emerged through the actions of two different focal firms as they sought to establish strategic positions in the nascent ecosystem for “cloud-based” computing services. We use qualitative data from interviews and archival sources to explore how the focal firms managed innovation within each platform’s core architecture. In doing so we explore the relationships between architectural flexibility and stability, and centralized and distributed forms of control.

Our findings suggest ways in which the two platforms’ approaches to the paradoxes of change and control subsequently affect the generativity of their ecosystems. The OpenStack project was
characterized by a dynamic approach to managing change and control. Rackspace, its primary initial sponsor, was flexible in stimulating component contributions driven by prioritizing external contributions. Rackspace also played an active role in terms of governance, even when control ultimately became more distributed. Overall, this flexibility ensured that OpenStack’s evolution proceeded in a robust and energetic way, despite some contention among members of its ecosystem. In contrast, Citrix, the initial primary sponsor of CloudStack, focused more on the stability of its core architecture, perceiving a lower need for component contributions from other firms. While Citrix also decentralized control by donating CloudStack to the Apache Foundation, the development of the platform stagnated, suggesting that Citrix was insufficiently adaptable in developing an ecosystem that encourages the creation of complements or components to its core architecture.

Our paper responds to the call by Tilson et al. (2010) for research that better understands ways in which infrastructural change shapes IT governance and the development of digital infrastructures. We show how firms take different approaches to navigating the paradoxes of change and control when establishing nascent platform ecosystems, and how this subsequently affects overall generativity of the firms’ respective projects. In particular, we emphasize that focal firms not only take different approaches to stimulating the development of components and complements, but that this process can vary in sequence and degree.

We contribute to current work on digital innovation (Yoo et al. 2010; Tanriverdi et al. 2010) by highlighting how generativity needs to be approached differently during a platform ecosystem’s nascent stages. In particular, our analysis suggests that the way firms manage the paradoxes of
change and control cannot be resolved by simply finding a middle ground between flexibility and stability, or between centralized and distributed control. Rather, what is key is the ability to adapt to changes in the ecosystem (e.g., concerns of other stakeholders as the ecosystem evolves).

The dynamics observed in our setting appear to be inconsistent with existing theory. In particular, our findings depart from existing work that suggests the importance of having a stable core platform, followed by variety in terms of complements (Baldwin and Woodard, 2009). Contrasting with existing work, we show that in the case of nascent platform ecosystems, flexibility of the core architecture may also be important to promote generativity. By analyzing contrasting processes in which firms navigate the paradoxes of change and control, our paper helps explain why such dynamics play out differently in nascent platform ecosystems.

2. Conceptual Development

This section draws on existing literature, which helps us develop a conceptual framework that depicts different trajectories firms might follow as they manage the paradox of change and paradox of control. We first highlight platform architectures and the broader ecosystem in which they are situated. Next, we draw on work on digital infrastructures to show different challenges and tensions that may arise in digital platforms. Drawing on this literature, particularly the paradoxes of change and control in digital infrastructure, we then introduce a conceptual framework that shows different options organizations have in navigating these paradoxes.
2.1 Platform Ecosystems

The focus of our study are platforms, “foundations upon which other firms can build complementary products, services or technologies” (Gawer, 2009. p. 54). Platforms are based on modular architectures (Ulrich 1995) that are partitioned into a core and a periphery (Baldwin and Woodard 2009). In this way the core architecture represents a set of functional modules with low variety and high reusability. These functional modules are selectively called upon, used and reused through interfaces by platform complements, which are arrangements of modules in the architecture of the platform periphery. These peripheral modules can express high variety, can be frequently changed, and are less often reused or called upon by other modules.

Digital platforms constitute digital artifacts (Kallinikos et al. 2013; Yoo et al. 2010), and may exhibit a high degree of generativity (Zittrain 2008, Henfridsson and Bygstad 2013). However, the digitality of such platforms also facilitates focal firms' abilities to exercise control over a platform ecosystem (Eaton et al. 2015). As focal firms sit at the nexus of the platform ecosystem, they benefit from access to privileged information (Boudreau 2005). This, combined with the possibility they may have property rights over code in the platform core, bestows them with power to control activities within the ecosystem in the manner of a “public-interest regulator” (Farrell and Katz 2000). One means of controlling ecosystem activities is through the specification of design rules that govern the use of interfaces (Baldwin and Clark 2000). This ability to control the innovative activities of other participants can lead to tensions in the ecosystem between focal firms and other members (Eaton et al. 2015).
We are particularly interested in digital platforms (Tiwana et al. 2010), where the degree to which these are open (West 2007; Boudreau 2010) enables innovation at different levels. At one level, it may enable other firms to act as complementors (Adner & Kapoor 2010) and build sets of peripheral modules around the platform core. However, the openness of these platform interfaces may go further and extend into the core architecture itself, so that at another level firms, acting as component suppliers (Adner & Kapoor 2010), can change existing or even build new modules in the core architecture. As we highlight next, this topic is not well understood, and the possibility of enabling frequent innovation of core platform modules may lead to tensions within the ecosystem in which the platform is situated, where some members require functional stability and others require flexibility (Tilson et al. 2010).

2.2 Platform Tensions and Digital Infrastructures

We draw on the literature on digital infrastructures (Tilson et al. 2010) to understand how a focal firm manages emerging tensions that may arise in platform ecosystems. The principal task in managing innovation in such infrastructures becomes one of creating and capturing value by growing an installed base (Tilson et al. 2010). In open platform architectures (West 2003), this requires growing the number of users and third parties, who contribute modules to the platform. Once an initial installed base has been developed, the digital ecosystem (Iansiti and Levien 2004) can benefit from network effects (Katz et al. 1994; Eisenmann et al. 2006; Farrell et al. 2007), which can drive further growth. The continued growth in the installed base requires focal firms to manage the ongoing evolution of their digital infrastructures. The design challenge of establishing a digital infrastructure and then evolving it in order to promote further growth is described by Hanseth and Lyytinen (2010) as an issue of dynamic complexity. Their solution to
this complexity is to first generate early growth through bootstrapping and attracting an initial installed base through simplicity and usefulness in design. This is then followed by promoting further growth through encouraging adaptability in design, for which they recommend modular and generative architectures (Hanseth and Lyytinen 2010).

However, digital infrastructures provide further challenges as they are scaled. Such challenges often manifest themselves as tensions and emerge in two ways. First, a growing digital infrastructure may become increasingly constrained by prior investments and design decisions. Second, competition may develop within the associated ecosystem over decision rights concerning the evolution of the infrastructure. These tensions are identified by Ciborra et al. (2000) as issues of change and control, and are conceptualized further by Tilson et al. (2010) as a set of two paradoxes. First, they identify a paradox of change, which concerns the need for a digital infrastructure to be both stable and flexible. Stability within the infrastructure is necessary to attract the contributions of ecosystem partners. Flexibility is simultaneously required in order to allow the digital infrastructure to grow and scale. Second, they identify a paradox of control, which refers to the distribution of decision rights concerning the evolution of an infrastructure. These rights can be distributed between a centralized actor, such as a focal firm, or across a distributed set of actors, such as the different members of a wider ecosystem that contribute to the innovation of modules.

As Tilson et al. (2010, p. 254) identify, the establishment of control points allow for different decisions to both centralized and decentralized in a growing ecosystem is of high importance. This is because it affects the attractiveness of participation in a digital infrastructure to different
sets of actors. The approach taken by Tilson et al. (2010) to conceptualize the contradictory elements that make up the paradox of change and the paradox of control is to treat their relationship dualistically (Gupta et al. 2006). From such a perspective, paradoxical elements can be treated as mutually enabling and constitutive (Farjoun 2010), such that one element is an enabling mechanism for the other to exist as an outcome, and vice versa. In this way, when treating the elements of the paradox of change dualistically, we see change working as a mechanism that can lead to stability as an outcome. In the following section we develop a simple conceptual framework that depicts how organizations might navigate these paradoxes.

2.3 Navigating the Paradoxes of Change and Control

Understanding the issues of change and control as a set of paradoxical dualities can provide insights into managing the growth and evolution of digital platforms. The partitioning of platform architecture into a set of core and peripheral components provides different areas where our understanding of these paradoxes can be applied. Furthermore this arrangement allows us to examine how focal firms, over time, manage the evolution of their platforms to respond to changes and adjust their emphasis between the opposite poles of both paradoxes. This is all the more relevant as our understanding of the dynamics of this process is limited.

In order to address this, we take our current understanding and apply a contingent approach to explain how focal firms manage the evolution of their platforms in response to change. We do this by focusing on the where and when of these paradoxes. We identify where along the continuum of the opposite poles of the two paradoxes they position their efforts and when in the evolution of their nascent platforms do they do this. We depict our approach using these two
paradoxes in a 2x2 matrix (Figure 1) which allows us to illustrate where a focal firm places its emphasis in relation to the paradoxes. The vertical axis focuses on change and represents a continuum between a focal firm’s emphasis on stability versus flexibility in the core architecture at a point in time. The horizontal axis focuses on control and represents a continuum between a focal firm’s emphasis on centralized versus distributed governance. By plotting where a platform is positioned with respect to each paradox at different points in time we can get a better understanding of the dynamics of how focal firms manage the evolution of their platform core.

<<Insert Figure 1 about here>>

3. Research Design

We adopt a case study method (Yin 2009) grounded in qualitative data (Eisenhardt 1989). We choose this grounded approach to building theory for two reasons. First, there is a lack of theory describing the phenomenon we are interested in and our research provides an opportunity for theory to be built inductively. Second, the phenomenon that we study is inherently processual, which, given its sequential and changeable nature, makes it suited to a case-based approach to theory building (Eisenhardt 1989).

We conducted a comparative case study focusing on two open-source initiatives, OpenStack and CloudStack, which develop software for managing “cloud-based” computing services. We focus on an emerging segment of the cloud computing market known as “infrastructure as a service” (IaaS), referring to “the capability provided to the consumer (…) to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to
deploy and run arbitrary software, which can include operating systems and applications.”¹ We focus in particular on the software for orchestrating cloud-based computing resources such as processing power, network bandwidth, and storage capacity. Taken together, these software layers are known as cloud management platforms (CMPs), “integrated products that provide for the management of public, private and hybrid cloud environments.”²

Within the market for IaaS, Amazon Web Services (AWS), particularly Elastic Compute Cloud (EC2) and Simple Storage Service (S3), constitute a significant part of the market.³ Another player, VMware, was an early pioneer in virtualization software, and its software is used primarily by corporate customers to create “private clouds” offering scalable on-demand computing resources within a traditional enterprise IT environment. OpenStack and CloudStack subsequently launched, and were led by Rackspace and Citrix respectively before their governance was transferred to independent software foundations. As CMPs, these two initiatives arose in part as a competitive response to both Amazon and VMware, and partly because of a growing demand for “hybrid cloud” services that allow firms to deploy resources across a combination of public and private cloud infrastructure.

We chose OpenStack and CloudStack as the focus of our case comparison for several reasons. First, as open-source software projects and digital platforms, they are exemplars of the phenomenon we wish to study. They consist of a broad ecosystem of heterogeneous actors that

---

contribute innovations to modules both within the core and the periphery of the overall platform architectures. Furthermore, focal firms exist in both ecosystems, who act to some degree as public regulators in their management of platform governance. Second, the evolution of their respective platform ecosystems occurs through discrete changes to architecture and governance regimes that can be traced over time in the form of design moves (Woodard et al. 2013). These changes are more visible than typical IT or product development projects, as they occur largely in the context of open-source software projects whose development process is documented and publicly accessible. Finally, as digital infrastructures they support a rapidly growing part of the economy. However, until recently, digital infrastructures have been neglected in research on both information systems and innovation management (Hanseth and Lyytinen 2010; Henfridsson and Bygstad 2013; Monteiro et al. 2014).

Our case evidence draws on a combination of archival and interview data. We collected several types of archival data, including official company material (from press releases and company websites) and material from third-party sources (analyst reports, online webinars, and industry-targeted websites covering cloud computing). We used this data to create a timeline of events related to the firms leading our two focal projects. This procedure followed an iterative approach consisting of several steps. First, one of the authors collected a set of articles related to infrastructure-as-a-service (IaaS) cloud computing initiatives. The key criteria for inclusion were (1) whether the article focused on the IaaS segment, and (2) whether it covered, directly or indirectly, our focal projects. Next, this set of articles was reviewed by a research assistant, who populated a database containing several items of interest: a summary of the event (or in some

---

4 The industry-focused websites included CloudAve, Forbes CIO Network, Forrester Research, Gartner Blog Network, GigaOm Cloud, ReadWriteCloud, the Register, Talkin’ Cloud, and Wired. By using a variety of sources, we tried to avoid overlooking any major event relevant to our setting.
cases events) described in the article, the firms involved, its strategic rationale, and its technical implications. From the full set of events in our data set, we selected the subset that fit Woodard et al. (2013)’s definition of a design move (“discrete strategic actions that enlarge, reduce, or modify a firm’s stock of designs”). This set of moves provided the main input for our empirical analysis.

In addition to our archival data, we conducted interviews with informants who were actively involved in the design or strategy of our two focal initiatives. These informants spanned a variety of roles, both technical (e.g., senior software development engineers) and managerial (e.g., senior directors, VP of product development). Interviews were done face-to-face when possible, and otherwise through video conferencing or telephone. When possible, we recorded the interviews, which were then fully transcribed. The interviews allowed us to get a better understanding of the motivations underlying the decisions we observed through public sources. Individual quotes (anonymized as agreed with interviewees) were also included to illustrate firms’ decision making.

We iterated on the selection of design moves as our understanding of the setting progressed and our conceptual framework came into focus. Initially, our research broadly examined ecosystem strategies in open-source cloud computing initiatives. As the project continued, we narrowed our focus to examine the question of how firms manage the paradox of control and paradox of change in digital infrastructures. The design moves, summarized in Table 1A and 1B serve as an overall summary of our two main cases. We also highlight individual industry articles in the Appendix, which are used to refer to specific details relevant for each case.
4. Findings

In this section we present our case findings. We first summarize the overall trajectory of both cases and then describe each case in more detail separately (in the subsequent sections we use the notation \{Bx\} to refer to industry sources summarized in the Appendix).

4.1 Summary of the Two Cases

OpenStack was launched in July 2010 by Rackspace and NASA to develop a suite of open-source software for managing cloud-based computing resources. NASA had been working on a project called Nova, which mimicked Amazon’s EC2. In turn, Rackspace’s Swift project was similar to S3. Eventually, members from each organization connected and pursued this opportunity collaboratively.\(^5\) Rackspace, a hosting services firm, decided to enter the segment of open-source CMPs for multiple reasons. First, developing OpenStack provided Rackspace the opportunity to draw away customers and compete more effectively with Amazon’s cloud services, whose fully integrated offerings posed a significant threat to Rackspace’s hosting services. Second, developing OpenStack as an open-source project made it more difficult for any other firm to secure a dominant position in the CMP layer, considered a strategically important position to hold in order to service the growing demand for “hybrid cloud” services. CloudStack was created in 2008 by a startup called Cloud.com. Citrix became involved with CloudStack

---

when it purchased Cloud.com in July 2011. Citrix, a virtualization software vendor, also decided to enter the segment of open-source CMPs for competitive reasons.⁶

Throughout OpenStack’s core architecture has been more open modular and flexible than CloudStack’s, and this has facilitated its evolution through contributions from different ecosystem component suppliers. CloudStack’s more integrated architecture, adopted from launch, provided an initial stability, which combined with interfaces adopted from Amazon encouraged an initial base of users who felt secure to build platform complements. However, over time, OpenStack was able to attract more innovation in its core, and thereby a greater installed base of users attracted by the superior range of functionality upon which richer complements can be developed. While CloudStack, through Citrix’s donation of its source code to Apache, adopted a distributed governance model six months earlier than OpenStack, it has been less energetic in engaging a developer community to build additional components in its core architecture.

Though CloudStack was the early leader due to its technical maturity and coherence, OpenStack eventually overtook it in terms of developer contributions and customer adoption. While its long-term success remains an open question, OpenStack has emerged as the front runner in open-source CMPs, considering the support it has gained of enterprise IT vendors including IBM, HP Intel, Dell, and VMware. By contrast, CloudStack has been less successful in terms of gaining market adoption. Perhaps as an acknowledgment of this, Citrix formally rejoined the OpenStack community in April 2015 and pledged to “integrate Citrix cloud infrastructure and networking

---

⁶An example from mobile telephony is the Android operating system, led by Google and made available for free. This eliminated the market for fee-based operating systems (e.g., from Symbian or Microsoft).
solutions with leading OpenStack distributions.”{B1}. Next, we discuss both OpenStack and CloudStack in more detail. For each platform we focus on its launch, subsequent growth and adoption, community engagement, and end with an overall appraisal.

4.2 Rackspace and OpenStack

Launch

OpenStack was announced July 2010,{B2} based on a collaboration between Rackspace and NASA. Its initial functionality was limited; Rackspace brought object storage (codenamed Swift), NASA brought compute (codenamed Nova). However, from the start there was the clear intention to bring in industry leaders and other players in order for them to contribute to the core functionality of the platform – and thereby, from an immature start, rapidly grow its core functionality and attract customers to build a market presence. The philosophy was that by working together OpenStack members could build a platform that would compete against the incumbents. In this way it was dubbed by some commentators as the “Android of the Cloud”{B3}. To that end more than 25 companies had joined as founding members at its launch including Citrix, Cloud.com, Dell and Intel. By looking at the development of OpenStack we can see how the platform has grown in terms of membership and customers, how its core functionality has developed, how its governance regime has evolved and finally how it has responded to concerns in its early years.
Growth and Adoption

The launch of OpenStack generated considerable interest from within the industry so that by December 2010, Rackspace was able to claim that the membership of OpenStack has grown from over 20 to over 40 organizations, now including Cisco, in six months. The growth in OpenStack’s core membership continued with large organizations such as HP signing up in July 2011, followed by AT&T in January 2012, IBM and Red Hat in April 2012, and VMWare in September 2012. Following the launch of the OpenStack Foundation, at which it claimed to have 5,600 individual and corporate members, the project continued to attract significant industry members including Nimbula in October 2012 and EMC in December 2012.

Community Engagement

There was considerable interest in OpenStack from the start. For example in October 2010 Microsoft announced the intention that its hypervisor (Hyper-V) would be compatible with OpenStack in order to spur adoption. Furthermore, the signing up of large industry players such as HP and IBM as OpenStack members, not only signaled their intent to contribute to core functionality, but also to distribute OpenStack deployments to their customer bases. The increasing adoption of OpenStack deployments by large organizations was demonstrated by Sony setting aside its Amazon Cloud deployments and adopting OpenStack instead. As vendors adopted OpenStack as their go to market cloud offering, they increasingly developed sophisticated certification programs to partners in order to develop their channels, increase adoption and to compete against other OpenStack vendors. Throughout the evolution of
OpenStack both Rackspace and then the OpenStack Foundation cultivated the engagement of the wide ecosystem of developers, users and administrators through a series of conferences {B13}.

While OpenStack was launched with the philosophy that its core functionality would be developed using the contribution of many organizations, including the founder members, the governance of this consortium, in terms of what code was actually released, was largely centralized around Rackspace in the early days. What followed was a slow and gradual process towards governance becoming more decentralized among OpenStack members. However the road to this transition was lined with numerous setbacks and controversies. In December 2010, shortly after launch, Rackspace acquired CloudStack founding member Cloudkick, attracted by the organization’s powerful server management tools {B14}. This sent a message that Rackspace was cannibalizing its co-members. This was followed up two months later by Rackspace’s acquisition of Anso Labs, another founding member of OpenStack {B15}. This move sparked debate and concern from within OpenStack’s membership and by commentators. The key concern was that these acquisitions were leading to Rackspace’s dominance of OpenStack governance, especially as the acquisition of Anso Labs gave Rackspace a majority of seats on the project’s governance board {B16}. These concerns were heightened when Rick Clark, a Rackspace employee and chief architect of OpenStack, left Rackspace and the OpenStack project citing his concerns over the governance model as part of the reasons for his departure {B17}.

Under mounting pressure from its co-members, the wider industry and commentators, Rackspace publically stated in October 2011 that it intended to give up control of the OpenStack project to an independent foundation with the announcement “that OpenStack is on the course for
autonomy, with intellectual property and assets set to transfer officially to the OpenStack Foundation in 2012”{B18}.

The community’s response to Rackspace’s announcement was somewhat muted, and in the months that followed there were a number of voices expressing continued concern regarding the perception that Rackspace was too dominant in its control of OpenStack and that this would harm the contributions from the wider community {B19}. Clark continued to speak out {B20}; other commentators took a different angle, claiming that it is easier for an open-source decentralized community to build on a stable mature base of code than “than trying to build complex software from scratch,”{B21} and that immature platforms such as OpenStack benefit from centralized control. This perspective was complemented by those of others, which would indicate that the emerging OpenStack Foundation governance structure, based around a tiered system of membership and power, was effective:

“The project is transitioning from an effort largely driven by Rackspace to a broader OpenStack Foundation with control to be spread among several vendors including Platinum members AT&T, Canonical, HP, IBM, Red Hat and SUSE joining Rackspace. Gold members include Cisco, ClearPath Networks, Cloudscaling, Dell, Dreamhost, Morphlabs, NetApp and Yahoo. There is precedent for this model succeeding. The Eclipse Java IDE really took off once IBM ceded control to a multi-vendor foundation.”{B22}

However in August 2012, shortly before the official launch of the OpenStack Foundation, disadvantages of this tiered governance structure were also becoming apparent. It was reported that corporate partners were intimidating individual members who were attempting to join the
board \{B24\}. This was widely interpreted as reflecting badly on OpenStack’s claims to operate as an open platform.

Finally on the 19\textsuperscript{th} of September 2012, nearly 12 months after Rackspace’s initial announcement, the independent OpenStack Foundation was officially launched. At this stage the foundation was supported by $10 million in funding, had 5,600 individual and corporate members, and was led by a board of directors, a technical committee and a user committee \{B25\}. The corporate members were divided into a number of tiers (platinum members, gold members, infrastructure donors, corporate sponsors, and supporting organizations) depending on the level of support that they gave the project.

“\textit{The tiered system in the board, about which some criticism came up in the past, still exists. But the response from the community is muted than before. OpenStack has set up the foundation in such a way that Board is mainly for administrative and marketing purposes while the technical committee has full control over the technical direction. This is definitely not an Apache Foundation kind of a solution but something that has a potential to work while keeping independence from the 1\% and protecting the interests of the 99\% (in most cases). Since CloudStack is sitting in the Apache Foundation waiting for defections to happen, the competition might keep OpenStack board honest for the time being.}\”\{B23\}

Therefore, Rackspace's approach of managing the community ultimately seems to reflect its announcement when it acquired Anso Labs and addressed concerns explicitly when announcing the acquisition:
“We also recognize this takes two of the largest contributors to the project today and puts them under one umbrella. For some people that will create concerns that Rackspace exerts too much control over the project. Our intent is quite the opposite. We launched this project because we felt that as part of a broader community, we could build something better.”

Appraisal

More than six years on from its launch in 2010, OpenStack has grown to over 30,000 individual and corporate members in over 170 countries around the world. As its membership has grown, increasing numbers of developers have contributed functionality to core architecture. In this way its architecture has grown from an immature platform to a sophisticated one, with 32 core components, over 14 semi-annual distributions. The OpenStack Foundation was launched roughly two years after the emergence of OpenStack. The two years leading up to the launch of the foundation saw frustration in the OpenStack ecosystem concerning the centralization of governance around Rackspace, and this has large dissipated since the transfer of governance and ownership to the OpenStack Foundation.

At the announcement of OpenStack’s launch, Lew Moorman, President, Cloud and CSO at Rackspace stated, “We are founding the OpenStack initiative to help drive industry standards, prevent vendor lock-in and generally increase the velocity of innovation in cloud technologies” {B26}. As part of this aim, OpenStack never adopted the APIs of its major competitors such as Amazon, in order to attract further adoption. In the early years of OpenStack, as it was

---

establishing itself, this was a concern among commentators and the ecosystem \{B27\}. Even up until July 2013, nearly a year after the launch of the OpenStack foundation, some members of OpenStack such as Randy Bias, CEO of Cloudscaling, were suggesting the adoption of Amazon’s APIs \{B28\}. A director at Rackspace explained the decision to not adopt AWS APIs by equating their potential to act as a control point to a “zombie”:

“Amazon seeks to establish its proprietary AWS API as the de facto standard for cloud computing […] Although there is no cure for AWS zombies, there is a vaccine that can prevent infection: OpenStack. Its native APIs are *not* AWS-API-compatible, so they will never expose you to the AWS zombie plague.”

Overall, OpenStack’s development can be characterized as focusing on active engagement with other ecosystem members (through, for example, developer conferences) and dynamic approach to governance. For instance, through the OpenStack foundation it responded and adjusted to concerns from community regarding ownership and control of the foundation. Its origin as a collaborative project between Rackspace and NASA is reflected in its product architecture, which is relatively modular. One interviewee contrasted OpenStack and CloudStack’s project structure as follows:

“OpenStack is really a confederation of 16 different projects, so there’s Nova, there’s Swift, there’s Dell’s Crowbar, all these things that comprise OpenStack. CloudStack really is just a product. There aren’t subprojects. It's all one, great big – it’s one product that everything works together. The upside for us is that it works. The downside is that when you have that modularity people can do really interesting little things.”
4.3 Citrix and CloudStack

Launch

CloudStack 2.0 was first previewed to the public in May 2010. It was developed by startup Cloud.com, which was until this point known as VMOps. The CloudStack codebase and functionality was already relatively capable and mature at launch {B29}. What is common for both Rackspace and Citrix is the perceived threat of value increasingly shifting to their respective key rivals. One interviewee reflected as follows on this potential commoditization of virtualization software (referred to in the industry as “hypervisors”):

“Amazon came and showed everyone how you can make money off the cloud [...] And we’re getting to where there’s actually quite a few orchestration layers out there for people to choose from and they’re all open source. They all work with all the hypervisors. They’re all [...] becoming a commodity.”

Growth and Adoption

In July 2012 Citrix acquired Cloud.com, reportedly for $200M {B30}, and took over ownership of CloudStack. Citrix then donated CloudStack source code to the Apache Software Foundation (ASF), which accepted it as an Apache Incubator project {B31}. According to reports the rationale behind this decision was that Citrix hoped to “abandon the one-man show” and generate contributions from others to its core functionality, with the assumption being that developers were nervous of contributing to a single-vendor driven project {B32}.
At the same time as donating the platform to the ASF, Citrix also announced that CloudStack would be embracing Amazon Web Services APIs to ensure compatibility with Amazon’s cloud offerings \{B33\}. One informant commented on the issue of AWS API compatibility, and its strategic importance, as follows:

“With CloudStack now being open, I think there’s a number of major differentiators. One is that it’s aligned in terms of API compatibility with AWS, which is the world’s biggest cloud, and OpenStack isn’t—that’s a key differentiator. (...)Building a real ecosystem around the APIs, so that it’s very difficult for anyone else to compete, I think that’s a sustainable advantage.”

This was interpreted by commentators as a move to encourage further adoption of the platform, although they were somewhat critical \{B34\}. It transpired that in October 2012 that Cisco was partnering with Citrix in order to build “integrated cloud orchestration” based on Citrix CloudPlatform, which provides evidence for organizations other than Citrix starting to contribute to the development of the core following the donation of CloudStack to ASF \{B35\}. Apache CloudStack graduated from the ASF Incubator in March 2013 \{B36\}.

Unlike OpenStack, CloudStack did not benefit from widespread marketing. This was largely on account of its being donated to ASF \{B37\}. Consequently, major organizations choosing to distribute or adopt deployments were generally not publicized. However, it was deployed or distributed by a number of major organizations including Alcatel Lucent, British Telecommunications, CenturyLink, Datapipe, Edmunds.com, Fujitsu, IBM (SoftLayer), Nokia, NTT, Orange, TATA Communications, TomTom, Verizon, WebMD, and Zynga \{B38\}. 
Community Engagement

There are some surface similarities to CloudStack’s approach to community engagement, but also several key differences. In its initial state CloudStack was owned by Cloud.com. Much, but not all, of the source code was published under the GNU General Public License (GPL) version 3 \cite{B39}, which meant that developers were free to adopt and adapt the code, but that changes to the CloudStack core were governed by Cloud.com alone.

Following the acquisition of Cloud.com by Citrix the control of CloudStack passed into the hands of Citrix \cite{B40}. Apart from the change of ownership, the governance regime is not reported as changing, such that Citrix was in sole command over what functional changes are made to the platform. In August 2012 Citrix released the rest of CloudStack’s code under the GPL \cite{B41}, which meant that all of the code was now publically available to adopt, distribute and adapt. However, the core was not yet open for third parties to help evolve the codebase.

The donation by Citrix of CloudStack to the Apache Foundation in April 2012 moved control of the platform away from a single organization. An interview with Mark Hinkle, director of community at Citrix Systems provided the following reasons for this move:

"Being under the Apache umbrella [...] means that companies will feel more confident committing to being part of the CloudStack project. The company could try to do its own foundation for CloudStack, but [...] it’s really hard to do your own foundation. [...] Apache is a proven governance model. It’s legitimate, attracts a high caliber of developer – people understand the Apache Foundation... Why recreate the wheel?” \cite{B42}
Citrix’s relation to OpenStack is notable as it had originally joined the initiative, but subsequently changed course. It is important to clarify what Citrix was proposing when it announced to “commercialize OpenStack”: to provide a supported distribution of OpenStack integrated with Citrix’s technologies, specifically XenServer. The following quote from one of our interviewees straightforwardly explains why Citrix would reverse this decision:

“Cloud.com [or] CloudStack have a much larger installed base, paid and unpaid, than OpenStack do. It’s been there longer so of course that makes sense. I think that in terms of a pure business decision it makes more sense to sell something you have, than to sell something that you don’t have yet, and with OpenStack we didn’t quite have it yet, and it was continually six months away from being done. And we weren’t going to make money on something that we couldn’t sell, but Citrix is able to make money by buying Cloud.com and then selling CloudStack.”

Our takeaway is that the approach taken by Citrix was indeed a sound business decision, but only if a firm considers this choice in terms of traditional product revenue terms. By switching its support to CloudStack, Citrix could potentially participate in the growth of this ecosystem. However, leveraging this opportunity would have required more investment to implement successfully, and Citrix’s approach appears not to have had the anticipated outcome, in particular compared to Rackspace’s success with OpenStack.

---

Appraisal

Unlike OpenStack, CloudStack received comparatively little publicity. This has been attributed \{B43\} to the fact that CloudStack was donated to the Apache Software Foundation, which lacks corporate sponsorship, unlike the OpenStack Foundation which received a significant amount of broad corporate involvement.

Nevertheless since it was donated to ASF, it failed to attract the interest of one or more larger corporations that OpenStack benefited from. The community\{B44\} that develops and maintains CloudStack consists of two types of members, the active project management committee and committers. Both these types of members were represented almost entirely by individual developers rather than by corporate groups. Citrix appears to have disengaged from CloudStack. This comes across in a number of ways. First, it made little effort to promote and support developer conferences, which recently appear to lack any form of corporate sponsorship, or have been abandoned entirely \{B45\}. Second, it disengaged from the Apache CloudStack Blog; having initially sponsored weekly updates in the 8-month period from January to the start of September 2012, it then gave up these efforts.\textsuperscript{9}

In an interview with Citrix System's director Mark Hinkle at the time of CloudStack moving to ASF, Hinkle offered up another comment:

"Some companies, say Oracle, turn to Apache when they no longer want to be responsible for a project like OpenOffice.org. [...] Citrix is going to be upping its investment considerably in

\textsuperscript{9} The Apache CloudStack web pages (http://cloudstack.apache.org/index.html) contain a number of broken links and do not appear to be particularly well maintained.
moving to Apache. In addition, Citrix will step up as an Apache Platinum Sponsor to help offset the costs of managing CloudStack and integrating it into Apache.\textsuperscript{[B46]}

Ironically, it would seem that Citrix, once it had obtained a mature cloud standard that it could commercialize itself, may have ended up emulating Oracle.

Overall, CloudStack’s development and approach to its wider ecosystem can be characterized as being based more on arms-length engagement (e.g. its developer conference was organized by a third party, rather than main project sponsor Citrix). Likewise, governance was also not actively managed by Citrix, but instead taken up by the Apache Foundation. Its origins as a commercial product developed by a single organization is also reflected in its product architecture, which is relatively integral.

4.4 Case Analysis

This section draws on the findings described earlier, and provides analysis in terms of the two paradoxes we are interested in. Overall, what emerges from our findings is that the paradox of change and paradox of control need to be addressed adaptively, for example by actively engaging with the ecosystem and responding to and addressing their concerns.

Paradox of Control: Centralized vs. Distributed Governance

Rackspace actively managed the paradox of control, by responding to and adapting governance to concerns raised by other ecosystem members. Here it moved from initially a more centralized governance (when OpenStack was owned solely by Rackspace) to an independent OpenStack Foundation. In case of the latter, here too Rackspace continued to engage with ecosystem
members about how the OpenStack Foundation was run. By contrast, while CloudStack also changed from centralized and distributed governance, Citrix provided a more arms-length approach to governance. This can be illustrated when it moved from more centralized governance (when CloudStack was owned by Cloud.com, and then acquired by Citrix) to more distributed (when it was donated to the Apache Foundation).

Paradox of Change: Stable vs. Flexible Architecture

In terms of architecture, OpenStack was initially a more immature product and a less developed architecture (note that there were just two components comprising the first OpenStack release called “Austin”). In response, Rackspace focused on engagement with other ecosystem members to develop the product and add core components. This is in contrast with an alternative option where Rackspace would have focused on making OpenStack as mature as quickly as possible by developing core components themselves. By contrast, CloudStack was a more mature product and was launched as a commercial product from the outset. Citrix focused less on bringing in other ecosystem members, but rather use more mature product to enter the CMP market. Here, an alternative option for Citrix would have been to keep aligning with OpenStack.

Overall, OpenStack was able to overcome challenges (e.g., its technological immaturity) that may otherwise have hampered its development. That OpenStack was able to overtake CloudStack is surprising given that existing work focuses on a developing stable (mature) core architecture, only after which the focal firm focuses on flexibility in periphery by complementors. Figure 2 provides a visual depiction of how our two focal cases navigated these paradoxes.
5. Discussion and Concluding Remarks

5.1 Paradox of Control: Outcome vs. Process

How did the paradox of control play out in our setting? We consider control as both a socio-technical outcome as well as a process. The social element of the control paradox is primarily driven by governance: here, control is about whether and how governance was centralized vs. distributed. Overly centralized governance may be contested by other stakeholders. In our case, Rackspace responded to these concerns by adapting the governance of the OpenStack Foundation, mitigating concerns about the establishment of governance-based control points.

The technical aspect of control is, in our setting, most clearly reflected in the way technological control points were addressed. Here, APIs might act as an “obligatory passage point” (Tilson et al. 2010, p. 8). OpenStack focused on avoiding relying on APIs controlled by other firms (especially AWS APIs), while CloudStack embraced compatibility. Overall, in our setting the social element of control appears to be a primary driver of how control is achieved as an outcome. For example, Rackspace’s process of managing other stakeholders was more effective than technological “shortcuts” (e.g., other firms’ control points such as Citrix’s reliance on AWS APIs).

We consider process to be a key driver in managing the paradox of control (i.e., how to manage the governance process itself). Our findings show that it is not just about moving between
centralized vs. distributed governance, but also how the focal firm manages this process. See for instance Rackspace and its deliberate management of the OpenStack Foundation and engagement with stakeholders and their concerns. This contrasts with Citrix and the Apache Foundation, which simply moved between centralized or distributed governance, without clear engagement with stakeholders. As a result of the dynamics taking place in this process, a seemingly similar trajectory from centralized to distributed governance can have different outcomes depending on how this process is managed. In other words, what is nominally a similar outcome (e.g., distributed governance) can have very different results.

5.2 Generativity and the Paradox of Change

Tilson et al. (2010) highlight how the relationship between generativity and control remains poorly understood. However, it is not just control and generativity that remains poorly understood, we also lack understanding about the role of change and generativity. In particular, OpenStack’s and CloudStack’s evolution contrasts with Baldwin and Woodard (2009), who prescribe the importance of a stable core and flexible periphery.

How did the paradox of change influence generativity? Our setting highlights how the relative technological maturity or immaturity of the core architecture can serve as either a help or hindrance in enabling change. This may in turn stimulate or curb the platform's generativity. CloudStack’s core architecture was more mature and its architecture less modular, in part a result of being developed by a single organization. CloudStack was initially more generative in the periphery, as evidenced by software maturity and greater number of initial deployments. CloudStack’s approach turned out to be effective in terms of adoption and deployment, but
counterproductive in stimulating generativity of the core architecture. In turn, OpenStack’s core architecture was less mature. As a result of the relative technological immaturity, OpenStack was initially less generative in the periphery based on deployments and adoption. Rackspace decided to pursue flexibility by engagement with other stakeholders, requiring a more modular architecture. This approach later turned out to be more generative in both the core architecture (where components were added) and the periphery (in terms of deployment and adoption of OpenStack). Overall, rather than stability, it was flexibility of the core architecture, as well as how this process of change was managed, that was key in stimulating generativity.

5.3 Addressing Paradox by Adapting and Responding to the Ecosystem

The nature of paradox underlines the importance of the duality inherent in both the paradox of change and the paradox of control; each paradox is contradictory but also mutually enabling (Farjoun 2010). Therefore, how firms approach each paradox needs to go beyond finding a middle ground or otherwise simply balance the dualities of each paradox. Instead, we suggest that the key challenge for each paradox is to know how to adapt to and navigate the particular circumstances in which the firm finds itself.

Overall, we characterize the approach that was most effective in addressing each paradox as deliberate and adaptive. Rackspace’s approach is marked by strong engagement, adapting their approach to concerns voiced by stakeholders. This deliberate and adaptive approach allowed them to overcome seeming disadvantages (such as the relative immaturity of its core architecture). In contrast, the approach taken by Citrix can be characterized as more arms-length and less engaged with stakeholders. Even though CloudStack had a technological head start over
OpenStack, both this lead as well as the move towards distributed governance were insufficient in getting other firms to contribute.

Taken together, these reflections provide further insight into the way firms can effectively manage the paradoxes of control and change during an ecosystem’s nascent stage. We use these paradoxes to contrast how firms manage a core architecture in the early stages of a platform ecosystem, and explain how these different trajectories subsequently hamper or promote generativity.
Figures and Tables

**Figure 1**: 2x2 matrix depicting how firms may over time navigate the paradox of change and paradox of control

**Figure 2**: 2x2 matrix depicting how our two platforms navigated the paradox of change and paradox of control

<table>
<thead>
<tr>
<th>Date</th>
<th>Event</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2010-07-19</td>
<td>Rackspace and NASA launch OpenStack project</td>
<td>Rackspace and NASA launch OpenStack project and release first two components under Apache 2.0 license; Rackspace creates a wholly owned subsidiary (OpenStack LLC) to own and manage OpenStack-related intellectual property</td>
</tr>
<tr>
<td>2010-12-16</td>
<td>Rackspace Acquires Cloudkick</td>
<td>Rackspace Acquires Cloudkick to Provide Powerful Server Management Tools for the Cloud Computing Era</td>
</tr>
<tr>
<td>2011-01-18</td>
<td>Rackspace Celebrates Six-Month Anniversary of OpenStack</td>
<td>Rackspace Celebrates Six-Month Anniversary of OpenStack, Backed By Over 40 Cloud Computing Industry Leaders</td>
</tr>
<tr>
<td>2011-02-09</td>
<td>Rackspace Acquires Anso Labs</td>
<td>Rackspace Acquires Anso Labs; Furthers Commitment to OpenStack</td>
</tr>
<tr>
<td>2011-07-28</td>
<td>HP joins OpenStack</td>
<td>HP has joined OpenStack, the open-source cloud software project led by Rackspace.</td>
</tr>
<tr>
<td>2011-10-06</td>
<td>Announcement of independent OS foundation</td>
<td>Rackspace announced Wednesday via a blog post that it will be giving up control over its OpenStack cloud computing project to an independent foundation</td>
</tr>
<tr>
<td>2012-01-09</td>
<td>AT&amp;T Joins OpenStack</td>
<td>AT&amp;T became the first U.S. telecom service provider to join OpenStack, the organization that has developed an open-source cloud software stack</td>
</tr>
<tr>
<td>2012-04-12</td>
<td>IBM &amp; Red Hat join OpenStack</td>
<td>IBM and Red Hat join OpenStack initiative</td>
</tr>
</tbody>
</table>
2012-04-16  | Rackspace Cloud adopts OpenStack | Rackspace announces that its Rackspace Cloud service is now powered by OpenStack

2012-09-19  | OpenStack Foundation launched | The OpenStack Foundation officially launches with initial funding of $10m; Rackspace transfers OpenStack LLC assets to OpenStack Foundation

2012-10-16  | OpenStack Summit | Nimbula Joins OpenStack Community

2012-12-12  | EMC joins OpenStack | EMC has joined the OpenStack open-source cloud effort.

2015-04-21  | Citrix sponsors OpenStack Foundation | Citrix rejoins the OpenStack Foundation as a corporate sponsor, pledges collaboration to help drive cloud interoperability standards

**Table 1A: OpenStack Design Moves**

<table>
<thead>
<tr>
<th>Date</th>
<th>Event</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2010-05-04</td>
<td>Cloud.com launched, releases preview of CloudStack 2.0</td>
<td>VMOps emerges from stealth mode as Cloud.com, releases preview of CloudStack 2.0 source code under the GNU General Public License version 3</td>
</tr>
<tr>
<td>2011-05-25</td>
<td>Project Olympus announced</td>
<td>Citrix announces intent to develop Project Olympus, a commercial implementation of OpenStack optimized for Citrix’s XenServer virtualization products</td>
</tr>
<tr>
<td>2011-07-12</td>
<td>Citrix acquires Cloud.com</td>
<td>Citrix acquires Cloud.com and its associated product portfolio, including CloudStack; confirms intention to continue contributing to OpenStack</td>
</tr>
<tr>
<td>2011-08-29</td>
<td>Cloud.com goes open source</td>
<td>After taking control of the CloudStack cloud management framework through its acquisition of Cloud.com back in July, Citrix Systems is now open sourcing the code behind the tool.</td>
</tr>
<tr>
<td>2012-02-29</td>
<td>CloudStack supports OpenStack Swift</td>
<td>Citrix releases CloudStack 3, including support for OpenStack’s object storage component</td>
</tr>
<tr>
<td>2012-04-03</td>
<td>CloudStack donated to Apache Foundation</td>
<td>Citrix donates CloudStack source code to the Apache Software Foundation (ASF), which accepts it as an Apache Incubator project; Citrix cancels involvement in OpenStack</td>
</tr>
<tr>
<td>2012-05-09</td>
<td>Citrix launches CloudPlatform</td>
<td>Citrix launches CloudPlatform, a commercial distribution of Apache CloudStack</td>
</tr>
<tr>
<td>2012-10-17</td>
<td>Citrix partners with Cisco</td>
<td>Cisco and Citrix partner to build “integrated cloud orchestration” based on Citrix CloudPlatform</td>
</tr>
<tr>
<td>2012-11-06</td>
<td>CloudStack stable release</td>
<td>First Apache-blessed CloudStack code debuts</td>
</tr>
</tbody>
</table>

**Table 1B: CloudStack Design Moves**
References


Boudreau, K., 2005. The boundaries of the platform: Vertical integration and economic incentives in mobile computing.


### Appendix – Blog References

<table>
<thead>
<tr>
<th>Code</th>
<th>Blog Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>{B3}</td>
<td>2010/07/19 Om Malik / GigaOM - OpenStack Wants to Be Android of The Cloud - <a href="https://gigaom.com/2010/07/18/openstack/">https://gigaom.com/2010/07/18/openstack/</a></td>
</tr>
<tr>
<td>{B6}</td>
<td>2012/04/05 Barb Darrow / GigaOM - Red Hat, IBM sign on for OpenStack - <a href="https://gigaom.com/2012/04/05/red-hat-ibm-sign-on-for-openstack/">https://gigaom.com/2012/04/05/red-hat-ibm-sign-on-for-openstack/</a></td>
</tr>
<tr>
<td>{B7}</td>
<td>2012/09/07 Barb Darrow - Finally! VMware joins the OpenStack Foundation, this time for real - <a href="https://gigaom.com/2012/09/07/finally-vmware-joins-the-openstack-foundation-this-time-for-real/">https://gigaom.com/2012/09/07/finally-vmware-joins-the-openstack-foundation-this-time-for-real/</a></td>
</tr>
<tr>
<td>{B8}</td>
<td>2012/09/19 Chris Talbot - OpenStack Foundation Launches with 5,600 Members - <a href="http://talkincloud.com/talkin039-cloud/openstack-foundation-launches-5600-members">http://talkincloud.com/talkin039-cloud/openstack-foundation-launches-5600-members</a></td>
</tr>
<tr>
<td>{B9}</td>
<td>2012/10/16 Chris Talbot - OpenStack: DreamHost Targets Amazon EC2 With DreamCompute - <a href="http://talkincloud.com/talkin039-cloud/openstack-dreamhost-targets-amazon-ec2-dreamcompute">http://talkincloud.com/talkin039-cloud/openstack-dreamhost-targets-amazon-ec2-dreamcompute</a></td>
</tr>
<tr>
<td>{B10}</td>
<td>2012/12/12 Barb Darrow / GigaOM - EMC follows VMware, rest of world into OpenStack - <a href="https://gigaom.com/2012/12/12/emc-follows-vmware-rest-of-world-into-openstack/">https://gigaom.com/2012/12/12/emc-follows-vmware-rest-of-world-into-openstack/</a></td>
</tr>
<tr>
<td>{B13}</td>
<td>2012/04/05 Barb Darrow / GigaOM - Red Hat, IBM sign on for OpenStack - <a href="https://gigaom.com/2012/04/05/red-hat-ibm-sign-on-for-openstack/">https://gigaom.com/2012/04/05/red-hat-ibm-sign-on-for-openstack/</a></td>
</tr>
<tr>
<td>{B15}</td>
<td>2011/03/08 Arik Hesseldahl / NewEnterprise - Rackspace Turns Anso Labs Into a Cloud Services Business Unit - <a href="http://allthingsd.com/20110308/rackspace-turns-anso-labs-into-a-cloud-services-business-unit/">http://allthingsd.com/20110308/rackspace-turns-anso-labs-into-a-cloud-services-business-unit/</a></td>
</tr>
<tr>
<td>{B17}</td>
<td>2011/03/24 Alex Williams / ReadWriteCloud - Over Concerns About Openness, OpenStack Founder Leaves Rackspace - <a href="http://readwrite.com/2011/03/24/questions-about-openstack-is/">http://readwrite.com/2011/03/24/questions-about-openstack-is/</a></td>
</tr>
<tr>
<td>{B18}</td>
<td>2011/10/10 Rackspace - OpenStack takes further steps toward autonomy - <a href="http://www.rackspace.nl/press-releases?page=2">http://www.rackspace.nl/press-releases?page=2</a></td>
</tr>
<tr>
<td>{B19}</td>
<td>2012/03/09 Randy Bias - Open Communities Deserve Equitable Governance - <a href="http://cloudscaling.com/blog/cloud-computing/open-communities-deserve-open-communication/">http://cloudscaling.com/blog/cloud-computing/open-communities-deserve-open-communication/</a></td>
</tr>
<tr>
<td>{B20}</td>
<td>2012/03/22 Rick Clark - Why I Left Rackspace and What About Openstack - <a href="https://dendrobates.rustedhalo.com/2011/03/22/rackspace-openstac/">https://dendrobates.rustedhalo.com/2011/03/22/rackspace-openstac/</a></td>
</tr>
<tr>
<td>{B22}</td>
<td>2012/07/15 Barb Darrow - OpenStack faces the terrible twos - <a href="https://gigaom.com/2012/07/15/openstack-faces-the-terrible-twos/">https://gigaom.com/2012/07/15/openstack-faces-the-terrible-twos/</a></td>
</tr>
<tr>
<td>{B27}</td>
<td>2012/04/18 Barb Darrow - Is it too late for OpenStack? - <a href="https://gigaom.com/2012/04/18/is-it-too-late-for-openstack/">https://gigaom.com/2012/04/18/is-it-too-late-for-openstack/</a></td>
</tr>
<tr>
<td>{B31}</td>
<td>2012/04/03 Joe Brockmeier / ReadWriteWeb - More Cloud Turbulence: CloudStack Heads to Apache - <a href="http://readwrite.com/2012/04/03/more-cloud-turbulence-cloudsta/">http://readwrite.com/2012/04/03/more-cloud-turbulence-cloudsta/</a></td>
</tr>
<tr>
<td>{B34}</td>
<td>2012/04/03 Christian Reilly / CloudAve - Wig Wam Bam. - <a href="https://www.cloudave.com/18626/wig-wam-bam/">https://www.cloudave.com/18626/wig-wam-bam/</a></td>
</tr>
<tr>
<td>Source</td>
<td>Date</td>
</tr>
<tr>
<td>--------------</td>
<td>------------</td>
</tr>
<tr>
<td>{B35}</td>
<td>2012/10/18</td>
</tr>
<tr>
<td>{B36}</td>
<td></td>
</tr>
<tr>
<td>{B37}</td>
<td>2014/6/18</td>
</tr>
<tr>
<td>{B38}</td>
<td></td>
</tr>
<tr>
<td>{B39}</td>
<td></td>
</tr>
<tr>
<td>{B40}</td>
<td></td>
</tr>
<tr>
<td>{B42}</td>
<td>2012/04/03</td>
</tr>
<tr>
<td>{B43}</td>
<td>2014/6/18</td>
</tr>
<tr>
<td>{B44}</td>
<td></td>
</tr>
<tr>
<td>{B45}</td>
<td></td>
</tr>
<tr>
<td>{B46}</td>
<td>2012/04/03</td>
</tr>
</tbody>
</table>