geocost: Cost-latency Optimized Multi-zone Cloud Storage Solution
(Technical Paper)

Agile or not? Factors that Motivate Agile Development Method to Emerge and to be Popular
(STS Paper)

A Thesis Prospectus Submitted to the
Faculty of the School of Engineering and Applied Science
University of Virginia • Charlottesville, Virginia
In Partial Fulfillment of the Requirements of the Degree
Bachelor of Science, School of Engineering

Haotian Liu
Fall, 2019

On my honor as a University Student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments

Signature _____ Date _____
Haotian Liu

Approved _____ Date _____
Haiying Shen, Department of Computer Science

Approved _____ Date _____
Thomas Seabrook, Department of Engineering and Society

Introduction

The past decades have witnessed the booming of the software engineering industry; in 2019, several major tech companies such as Microsoft and Apple have subsequently surpassed $1 trillion market value in recent years (Vlastelica 2019). Today, it is no joke to say that every company either hires or needs a software engineer. A vast majority of traditional companies moved their services online, while new business models based on software as a service (SaaS) such as Facebook (social network), Robinhood (investment) and Uber (ridesharing) are emerging at an unprecedented rate. In these companies, software development teams form the backbone of the company's operation and consist mainly of software engineers and project managers. Since commercial software development usually involves constant communication and interactions between different developers and teams, many teams and companies choose to follow a particular software development method that helps standardize and streamline software development process.

As a part of the software development process, software development methods have seen dramatic evolution over recent decades. Earlier methodologies such as plan-driven and waterfall model were once popular when commercial software burned into a CD were sold like books in dedicated "software stores" in the 1980s and 1990s, while Agile software development method first emerged in 2001's Agile Manifesto (Beck et al. 2001) and quickly gained huge popularity in the software industry, continuing to blur some very well-defined boundaries in plan-driven and waterfall development such as business requirement and product delivery timeframe. Agile development method focuses on delivering the Minimum Viable Product (MVP) by shortening the development cycle and trimming down business requirements, thereby achieving faster product rollouts and updates (Duc et al. 2016).

Contrasting to plan-driven and waterfall development, Agile development focuses on iterate rather than incubate development and has transformed from a conference memo to rather a norm in today's software industry in less than 10 years (Dingsøyr et al. 2012). This swift shift in development methods is worth studying since it may serve as a prism through which some of the most important changes in commercial software industry can be observed. In this thesis prospectus, I will focus on discussing some key factors that resulted in this shift such as the rapid evolution of cyberinfrastructure, the emergence of SaaS (Software as a Service) business model, the STS framework my topic fits into, and outlining a set of future STS research methods.

Technical Topic

My capstone research (UVa Pervasive Communications lab) focuses on building a cost analysis model for large-scale distributed storage-based cyberinfrastructures such as AWS S3, Microsoft Azure and Google Cloud Storage. Since today's websites often delivers large amount of multimedia data (images, sounds and videos) and are often expected to not only be fast (e.g. a latency higher than 100 ms (Henty 2015) will bring an impression of "slow") but also highly available (i.e. site keeps functioning even in regional network / hardware failure) (Vohra 2017). These requirements, together with the emergence of cloud-based storage services such as S3, Azure and Google Cloud Storage led to many distributed, cloud-native architectures in modern web development. A website built on distributed infrastructure have its data stored, often with multiple replications, in multiple data centers located in multiple regions / availability zones to provide high speed (data from the nearest data center is served) and high availability (if one or more data centers is offline, the site can still function).

Storing multiple copies of data and having multiple web servers across multiple availability zones means that more data needs to be stored and therefore proposes a cost challenge for medium-to-low budget websites, especially when major cloud storage service providers (S3, Azure, Google Cloud Storage) all charges differently for data stored in different data centers (Amazon 2019). Data centers located in high-traffic areas, such as the AWS datacenter in US East – Virginia region are significantly more expensive than its counterparts located in other areas. Therefore, finding the balance between cost and performance (i.e. site latency and availability) could be hard without a properly configured cost model of each major cloud storage service.

Our team aims to build such a model that correlate cost of cloud storage and website latency with unsupervised learning on results of benchmarking cloud storage services. Our team set out to understand the difference in performance for each major data center (East / West coast in U.S., Canada, East / South Asia, Europe and South America) by writing benchmarking services that send concurrent storage download / upload requests from multiple clients from remote VMs in GENI, a university-based distributed computing resource network from multiple locations in the U.S. including Los Angeles (UCLA), Utah (University of Utah) and Virginia (University of Virginia). Data aggregated from these benchmarking tests will then be used towards unsupervised machine learning for daily pattern extraction and making intelligent predictions of future latencies and costs.

Since low latency and high availability have become two paradigms for modern web architecture (Matsudaira 2012), this model will help medium-to-low budget websites to plan out their hosting plan with costs in mind and therefore achieve the optimal balance between performance and cost requirements.

STS Topic

In the research of factors that motivate Agile development method to emerge and to be popular, there are several key stakeholders, including software developers, project managers (often referred to as "scrum master" in Agile terminology) and software users (Bass 2014). Working together, these stakeholders create two essential non-physical artifacts: the software (often written in code), and development methodologies (often written down as a part of a developer's guide). The latter artifact is of primary interest in my STS research since it serves as a general guideline upon which the first artifact is continuingly produced and sometimes guarantees its quality.

The STS framework that will be utilized in determining factors that motivate Agile development to emerge and to be popular is the Actor Network Theory (ANT). Modern software development teams and software users, together with artifacts they produce (software and development methodologies), form a nearly perfect real-world simulation of a network, defined in Actor Network Theory as the sum total of individual human or non-human actants. Actor Network Theory accounts for all actants within a network, whether human or not, both with a certain amount of value and agency (Sayes 2013). In the realm of software development, an actant can be interpreted as either a previously defined stakeholder or non-physical artifact. Each of these actants should be treated with equal agency, as each has the ability to directly affect and change other actants within the same network. Actor Network Theory therefore provides a great analysis framework to simulate the network of interactions amongst software developers, scrum masters, software users, the software itself and software development methodologies. It is worth noting though many actants in this "software development network" are frequently actants of other overlapping networks as well. Therefore, while using the Actor Network Theory,

implemented as the software development network in this case to determine factors that motivate Agile development method to emerge and to be popular, it is necessary to account for influences other outsider networks may project onto actants within the software development network. In addition, Actor Network Theory often faces the critique that it sometimes puts too much emphasis on non-human actants and overlooks human morality (Sheldon 2010). However, this critique is of little concern in the software development network: it is well-known that code quality plays the leading role in the software industry. Since software development method is a direct contributor to a developer's code quality, it is also undoubtedly of equal or even more importance to other human actants in software development network.

This research is important not only because some of the most important changes in software industry can be traced through finding key factors that led to evolution of software development methods, but also may help in identifying some deficiencies of the Agile method. While Agile is commonly seen as an industrial norm in today's software industry, it is in no way a perfect software development method: in recent years, employee-related accidents, while still relatively sparse, in Agile-following companies shows a tendency of increasing in numbers and severity (Price 2019). Key deficiencies of Agile development method include the lack of process that allow developers to easily get sidetracked and lack of long-term (Lucidchart 2019). Studying the factors that motivate the Agile development method to emerge and to be popular may help pinpoint some key "fixes" to Agile development, thereby making it an overall better software development methodology.

Research Question and Methods

Due to the specificity of the chosen topic, the research question of this paper can be easily condensed to just one simple question: what are the factors that motivate Agile development to emerge and to be popular? In spite of its seemingly simplistic nature, understanding this question requires first-hand information from real software engineers and an extended background knowledge of the history and current climate in software development industry. Hence, one research method that will be utilized is interviewing real software engineers and scrum masters from Agile-following tech companies. The interviewees will first be asked a series of general questions, such as "how do you like the Agile method your company is following?" and then a set of questions specific to their background and technical expertise, such as "how do you think following Agile method in your daily work has shaped you as a professional software developer in your company?". The answers to these questions will be used as first-hand supporting information to consolidate my arguments while accompanied by another set of literature review on previous studies and news on the Agile development methods, since amount of publications on Agile development method is fairly extensive (Dingsøyr et al. 2010). Therefore, this paper will employ two research methods: in-person interview and literature review.

Conclusion

The final deliverable of my technical topic (building cost analysis model on large-scale distributed storage-based cyberinfrastructures) will be a set of cost models and algorithms in form of code and documentations hosted in an online repository.

The final deliverable of my STS research will be an STS research paper on factors that motivate Agile to emerge and to be popular plus a series of in-person interview records and

literature review notes. These deliverables will help foster better understanding of underlying reasons of Agile development method in the broader context of a custom-defined software development network while pinpointing some key “fixes” to some deficiencies of Agile method, such as increased employee mental pressure due to greater time and commitment (Fridman 2016) so that it can be further optimized and benefit every actant in the software development network.

References

Vlastelica, Ryan. (2019) Apple's $1.02 Trillion Valuation – a 17% Gain From August's Low closes in on Microsoft's Lead. Retrieved October 27, 2019, from https://fortune.com/2019/09/12/apple-regains-one-trillion-valuation/

Beck, Kent, et al. (2001) Manifesto for Agile Software Development. Retrieved October 28, 2019, from http://agilemanifesto.org/

Dingsøyr, Torgeir, et al. (2012) A decade of agile methodologies: Towards explaining agile software development. Retrieved October 29, 2019, from https://reader.elsevier.com/reader/sd/pii/S0164121212000532?token=63097B9B21147C8122D7BF0C3150BBF348BE88254B4BDF9251A7969F65E4FCDE10FDBDEE293AAFD983B6F38593728B3A

Dingsøyr, Torgeir, et al. (2010) Agile Software Development: An Introduction and Overview. Retrieved October 28, 2019, from https://www.researchgate.net/publication/234274661_Agile_Software_Development_An_Introduction_and_Overview

Duc et al. (2016) Minimum Viable Product or Multiple Facet Product? The Role of MVP in Software Startups. Retrieved October 29, 2019, from https://link.springer.com/chapter/10.1007/978-3-319-33515-5_10

Henty, Steve. (2015) UI Response Times. Retrieved November 4, 2019, from https://medium.com/@slhenty/ui-response-times-acec744f3157

Vohra, Deepak. (2017) Developing a Highly Available Website. Retrieved October 18, 2019, from

https://link.springer.com/chapter/10.1007/978-1-4842-2598-1_16

Amazon.com, Inc. (2019) Amazon S3 Pricing. Retrieved October 27, 2019, from https://aws.amazon.com/s3/pricing/

Matsudaira, Kate. (2012) Building Scalable Web Architecture and Distributed Systems. Retreived October 22, 2019, from https://www.drdobbs.com/web-development/building-scalable-web-architecture-and-d/240142422

Bass, Julian. (2014) Scrum Master Activities: Process Tailoring in Large Enterprise Projects. Retrieved October 29, 2019, from https://ieeexplore.ieee.org/abstract/document/6915249

Sayes, Edwin. (2014) Actor–Network Theory and methodology: Just what does it mean to say that nonhumans have agency? Retrieved October 28, 2019, from https://journals.sagepub.com/doi/full/10.1177/0306312713511867

Sheldon, Ben. (2010) Criticism of Actor-Network Theory. Retrieved October 29, 2019, from https://island94.org/2010/01/Criticism-of-Actor-Network-Theory.html

Price, Rob et al. (2019) Employee dies at Facebook's Silicon Valley headquarters in apparent suicide. Retrieved October 14, 2019, from https://www.businessinsider.com/one-person-dead-by-suicide-at-facebook-office-building-2019-9

Fridman, Adam. (2016) The Massive Downside of Agile Software Development. Retrieved October 11, 2019, from https://www.inc.com/adam-fridman/the-massive-downside-of-agile-software-development.html

Lucichart Content Team. (2019) 3 Disadvantages of Agile Methodology. Retrieved October 12, 2019, from

https://www.lucidchart.com/blog/3-disadvantages-of-agile-methodology