HAL
open science

Why Not Let IT Fail? The IT Project Success Paradox

Paul J. Ambrose, David Munro

To cite this version:

Paul J. Ambrose, David Munro. Why Not Let IT Fail? The IT Project Success Paradox. InternationalWorking Conference on Transfer and Diffusion of IT (TDIT), Jun 2013, Bangalore, India. pp.579-582, 10.1007/978-3-642-38862-0_37 . hal-01467803

HAL Id: hal-01467803

https://inria.hal.science/hal-01467803v1

Submitted on 14 Feb 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License

Why Not Let IT Fail? The IT Project Success Paradox

Paul J. Ambrose, David Munro

University of Wisconsin-Whitewater, Whitewater, WI 53190, USA

ambrosep@uww.edu, munrod@uww.edu

Abstract. Is a focus on information systems or information technology success a myopic view of evaluating IT success and failure? Are success and failure the opposite ends of a continuum for evaluating IT projects? Conventional measures of success such as meeting cost, time, budgets, and user needs do not address positives that can emerge from failures. We contend that a focus on success and failing to factor the possibility of failure actually hamper IT projects. An organizational mandate that does not allow for failure does not promote risk taking and innovation. It can also foster a project climate fraught with undesirable or unethical behavior and stress among developers, while failing to capture positive lessons that could emerge from IT project failure.

Keywords: IS success evaluation, IT failure, innovation

1 Introduction

Social and business norms expect success at everything we do - at work, at play, and at home. As a society we remember winners and not losers. We do remember failures, though not in the same positive sense as we glorify successes. It is no different when it comes to evaluating, or setting expectations for IT (Information Technology) projects. Success is mandated explicitly or implicitly by project stakeholders. Perhaps that expectation is justifiable. After all business stakeholders invest time, effort and monetary resources in IT projects and as good stewards of business capital, have a rightful expectation of project success defined by the project meeting its set objectives and goals. But then the focus just on immediate project success could be too narrow a world view, and one can even argue that this could be detrimental especially for the overall or long term success of an organization. After all doesn't the old adage "Failure is the stepping stone for success" seem to encourage not willful actions that precipitate failure but the latitude to make mistakes so that we may learn from such errors and consequently become better? Or become more successful in the long run?

The passion to evaluate IT success in IS (Information Systems) academic research, however elusive that evaluation may be, could be traced to early influences on the discipline that originated in the scientific computing foundations and thoughts. Shan-

adfa, p. 1, 2011.
© Springer-Verlag Berlin Heidelberg 2011

non and Weaver's seminal 1949 work "A Mathematical Theory of Communication" set the stage and standards for information processing with the focus being on the fidelity of information as it moves from the source to destination, and consequently improving technical aspects of communication. At the core of DeLone and McLean's 1992 and 2003 synthesizing work of past attempts to evaluate IS success we find Shannon and Weaver's thesis on technical communication. DeLone and McLean's IS success constructs and their derivatives continue to play a significant role in the measurement of IT success in IS literature. However it is ironic that a model with theoretical roots in science and engineering is attempting to address a problem that is social sciences in nature. The danger of this approach is that we could be missing behavioral factors that need to be addressed ground up in a theoretical model rather than these factors being grafted on top of a scientific model.

One may wonder why such a deterministic technical approach to evaluation IS/IT project success is being criticized, as after all don't we need a systematic approach to evaluate success? The counter to that argument however is twofold. First the time, size, and cost and effort are difficult to determine at the project initiation stage as IT projects follow the "hurricane" model (Dennis et al, 2009, pg 82) where these parameters can be better estimated only when the project is well underway, just as the actual landing sites of hurricanes are known only after landfall. Second, IT has been a key enabler of business transformation, and IT led transformation projects are inherently high risk, high reward projects where the possibility of failure is implicit and integral to the project. Ignoring failure can lead to not learning from such experiences and set the stage for the same mistakes to be committed again. To dig a bit deeper into this subject, let us consider three specific areas where a focus purely on success and failing to factor the possibility of failure in a positive manner is not desirable for IT projects.

2 Central Arguments

2.1 Success obsession hinders innovation:

A holistic approach to developing projects and services that meet a myriad of stakeholder needs could be addressed from the perspective of design thinking. Tim Brown elaborates the core principles behind design thinking to include feasibility (it is functionally possible to deliver the needs using technology), desirability (does the project make sense to users) and viability (can it be a sustainable business model). Further, design thinking espouse an organizational culture that accepts "forgiveness" later than asking for "permission" ahead. This is essential to foster innovation by encouraging risk taking which can also lead to mistakes being made. However if the business mandate is for IT projects to meet time, scope and cost, per traditional project management principles, IT professional are likely to adopt a risk averse strategy that curbs innovation. Current IS/IT success models seem to fail to take this into account.

In addition, developers have a disincentive for being innovative in development approaches, methods or tools when success is determined for the traditional perspective currently being followed. In some cases this reluctance to be innovate is because the benefits are not yet well known and may in fact may not exist. The "let someone else try it first" approach is safer. Furthermore changes due to an innovation that is introduced in a process will introduce new risk factors to the project, thus making it harder to evaluate the risk and trade-offs of doing the innovation, which leads to project managers viewing these innovations as unnecessary risks. Perhaps the biggest reason to discourage them is the very fact that implementing the innovation will take time away from the 'actual project'. For example the learning curve alone will likely take more time on this project then the innovation will provide on this project. It is not surprising that often innovation has to forced top down by upper level management, when most often it would be more effective and cheaper for the organization to have it arise organically from the bottom up.

2.2 Success obsession can foster unhealthy stress:

Let's start by examining how the stress people feel gets played out. It has to be noted that the stress individual feel about avoiding failure is not uniform and will vary greatly by many factors including their work experience, personality, and factor outside of work to just name a few. However we would be naive to assume that at least some if not most of the individuals on a development effort feel a lot of pressure for a project to succeed, or at least not be the cause of it failing. While the concept of a professional athlete "choking under pressure" has been studied for decades, Sanders and Walia (2012) have shown extended the economic theory that this is not limited to athletes. To quote from their conclusion "Pressure may reduce not only reduce productive output but, more fundamentally, may erode incentives to put forth productive input" While the stress may cause some people to underperform others may leave the organization when their stress levels are too high. As a person's stress from work spills over to their personal lives they are often encouraged by concerned family and friends to consider looking for a less stressful job environment. When this does happen and an individual leaves a project the remaining people on the project are left with more stress from either having to pick up the work of the individual who leaves or in communicating and bringing up to speed the person who replace them. A third way people can deal with stress is to just give up trying to meet what they perceive as unrealistic expectations that are thrust upon them. That is they basically deal with the stress of failure by assuming it is inevitable and hence do what they can but not really wholeheartedly throw themselves into the project.

2.3 Success obsession can motivate undesired behavior:

Austin (2001) highlights the effect of time pressure on developers where developers were known to take short cuts while dealing with unanticipated project complications, even if these shortcuts are not in the best interest of the project. Austin showcases an example where developers deliberately planted a "bug" in the software that would buy them time to do more work in the project when the project "breaks" during production deployment. So holding down people to hard success measures could encourage them to circumvent boundaries that could be considered as being unethical. This is not an isolated report. Zelazny (2011) reported based on a survey that "software development team members do not consider the internal quality attributes of modifiability, portability, and reusability when considering an information systems development project successful." Considering that the development team members are the people who most benefit from these attributes in their day-to-day work shows the degree in which the area takes back seat in pursuit of success.

3 Conclusion

So where does this all lead to? The bottom line question would be, how do we positively address IT failures as part of IT success evaluation? The motivation is to encourage IT professional to work innovatively and take necessary risks and be proactive contributors on projects. A technical and deterministic view of IS success may not be the right solution to meet that end. Should we then develop a model for IS failure than success?

References

1. Austin, R.D., The effects of time pressure on quality in software development: An agency model. Information Systems Research, 2001. **12**(2): p. 195-207.
2. Brown, T. and B. Katz, Change by design : how design thinking transforms organizations and inspires innovation. 1st ed. 2009, New York: Harper Business. viii, 264 p.
3. DeLone, W.H. and E.R. McLean, Information Systems Success: The Quest for the Dependent Variable. Information Systems Research, 1992. **3**(1): p. 60-95.
4. DeLone, W.H. and E.R. McLean, The DeLone and McLean Model of Information Systems Success: A Ten-Year Update. Journal of Management Information Systems, 2003. **19**(4): p. 9-30.
5. Dennis, A., B.H. Wixom, and D.P. Tegarden, Systems analysis design, UML version 2.0 : an object-oriented approach. 3rd ed. 2009, Hoboken, NJ: J. Wiley. xviii, 581 p.
6. Sanders, S., Walis B., "Shirking and 'choking' under incentive-based pressure: A behavioral economic theory of performance production", Economics Letters, Volume 116, Issue 3, September 2012, Pages 363–366
7. Shannon, C.E. and W. Weaver, *The mathematical theory of communication*. 1949, Urbana,: University of Illinois Press. 125.
8. Zelazny, Lucian M., "Toward a Theory of Information System Development Success: Perceptions of Software Development Team Members", June 8, 2011 Dissertation, Virginia Polytechnic Institute.