

Introduction

Solutions Delivery (SD) is a critical capability of the IT Capability Maturity Framework (IT-CMF). IT-CMF is a continually evolving framework and has been updated to ensure its ongoing relevance in supporting the delivery of business value from IT in the digital era. SD covers all aspects of delivering an IT solution to meet the organization's business requirements, including design, development, integration, validation, and deployment. It is a core differentiator in digital transformation, with the potential to transform both internal processes and the organization's external market presence. The SD content is based on rigorous academic research and has been further developed in collaboration with experts from leading organizations to ensure it reflects both state-of-the-art organizational practice and academic research. This paper outlines the key insights from this research and how it has informed the ongoing development of SD.

Delivering Solutions in the Digital Age

An organization's capability to deliver IT solutions is increasingly becoming a key competitive differentiator in the digital era. Techniques such as Agile [1], Lean [2], and DevOps [3] are transforming the organization's internal processes. Organizations increasingly depend on software to differentiate their products and services [4][5].

The primary challenge for organizations has shifted from solving technical problems to delivering value to customers [5]. One such challenge is how to reduce the development and deployment life cycle from years to months or even weeks [4] in order to increase the time to market. Increased frequency of delivery and a move towards 'continuous' deployment and integration can all offer benefits in terms of quality and consistency [6], by allowing for the early feedback from real users [7] and reducing risk [8]. This is facilitated through automation or built-in quality, as well as improvements in coding standards, reference architectures, and integrated development environments (IDEs) [6]. These changes require both cultural and structural changes, including effective product ownership and prioritization [9], shifting to smaller, cross-functional development teams [10], and a focus on 'features' instead of components [5].

This move towards rapid development changes how requirements are engineered [11]. There is a move away from detailed requirements specification towards techniques such as user stories, which rely less on committing to a detailed description prior to development, and more on providing reference points throughout the iterative and collaborative delivery process [11]. This process is considered complete not once the requirements are met, but once customers have actually adopted the product [6]. Additionally, the process of understanding needs should start earlier and happen

continuously [4], and should be direct, through customer feedback, analytics, and observation of use, rather than indirect, through upfront business analyses [5].

Throughout the design, development, and integration phases, the focus should be on satisfying evolving needs, as well as ensuring that decisions are guided by principles such as maintainability, testability, compliance, and reuse [12]. For this reason, there is a move towards 'plasticity' in design, maintaining modular architectures and flexible designs [13], and maintaining transparency throughout the process. The software architecture and system design should drive efficiency, for example, by maximizing code reuse and ensuring a modular software architecture with clear interfaces [9]. The decision of whether to build or buy solutions is also integral to balancing cost, efficiency, and quality in design [14]. Organizations should ensure they review and update their methodologies and processes, keeping an eye on the latest developments in these areas, as well as commercial off-the-shelf (COTS) products, to ensure they 'continue to increase productivity, time to market, and quality' [9].

There is a move away from sequential steps, and the delivery process is faster with more steps happening concurrently. A process of continuous integration 'is typically automatically triggered and comprises inter-connected steps such as compiling code, running unit and acceptance tests, validating code coverage, checking compliance with coding standards, and building deployment packages' [6]. This involves a 'shift left' strategy when it comes to testing, daily build of the product for testing [15], and moving testing activities earlier in the cycle to allow for the early identification of problems [6]. Best practices in testing prior to deployment include peer review, internal use, and extensive automated testing [8] [16].

Companies are also moving to a 'continuous deployment' approach, with the ability to deploy software at will [7]. This supports a culture of experimentation, allowing for feedback from real users [7], and treats failures as opportunities for learning, rather than 'occasions for assigning blame' [8].

Conclusions

SD is a core differentiator in digital transformation, with the potential to transform both internal processes and the organization's external market presence. SD has been updated to support an increase in efficiency of the delivery process, while also focusing on maximizing quality. Although the new iteration of SD does not mandate the use of Agile or continuous deployment practices, it does support those practices, as well as offering a path to improvement for companies dealing with legacy issues and systems. SD supports both the delivery of a product and the continual release of features. Additionally, SD measures and provides support on improving digital best practices, agile and lean processes, continuous delivery, collaboration, and cross-functionality, all with a razor sharp focus on the customer and user.

Research Methods

In addition to undertaking rigorous academic research, this iteration of SD was developed through a collaborative workgroup model with expert practitioners from leading organizations, who are part of the wider Innovation Value Institute (IVI) community. The workgroup was established as a community of practice to understand, share, and iteratively advance practice in SD and relevant digital developments, such as DevOps.

References

- [1] T. Dybå and T. Dingsøy, 'Empirical studies of agile software development: A systematic review', *Inf. Softw. Technol.*, vol. 50, no. 9–10, pp. 833–859, Aug, 2008.
- [2] T. Ohno, *Toyota Production System: Beyond Large-Scale Production*, 1988.
- [3] G. Kim, J. Humble, P. Debois, and J. Willis, *The DEVOPS handbook - how to create world-class agility, reliability, and security in technology organizations*, 1st ed. 25 NW 23rd Pl, Suite 6314, Portland, OR 97210: IT Revolution Press, LLC, 2016.
- [4] P. Andén, O. Burkacky, J. Kupferschmid, and A. Rocha, 'Software or nowhere: The next big challenge for hardware OEMs', in *Software development handbook: Transforming for the digital age*, McKinsey&Company, 2016.
- [5] F. Fagerholm, A. S. Guinea, H. Mäenpää, and J. Münch, 'Building blocks for continuous experimentation', in *Proceedings of the 1st international workshop on rapid continuous software engineering*, pp. 26–35, 2014.
- [6] B. Fitzgerald and K. J. Stol, 'Continuous software engineering: A roadmap and agenda', *J. Syst. Softw.*, vol. 123, pp. 176–189, Jan, 2017.
- [7] M. Leppänen *et al.*, 'The highways and country roads to continuous deployment', *IEEE Softw.*, vol. 32, no. 2, pp. 64–72, 2015.
- [8] D. G. Feitelson, E. Frachtenberg, and K. L. Beck, 'Development and deployment at facebook', *IEEE Internet Comput.*, vol. 17, no. 4, pp. 8–17, 2013.
- [9] P. Andén, O. Burkacky, T. Strålin, and A. Rocha, 'Software status: Diagnosing development performance', in *Software development handbook: Transforming for the digital age*, McKinsey&Company, 2016.
- [10] A. Steiber and S. Alänge, 'A corporate system for continuous innovation: the case of Google Inc.', *Eur. J. Innov. Manag.*, vol. 16, no. 2, pp. 243–264, Apr, 2013.
- [11] N. Niu, S. Brinkkemper, X. Franch, J. Partanen, and J. Savolainen, 'Requirements engineering and continuous deployment', *IEEE Softw.*, vol. 35, no. 2, pp. 86–90, 2018.
- [12] A. Patwardhan, R. Patwardhan, and S. Vartak, 'Self-contained cross-cutting pipeline software architecture', *ArXiv Prepr. ArXiv160607991*, 2016.

- [13] M. Jones and J. Rose, 'Digital all the way down; innovation in the software industry', in *FEWEB digital innovation workshop: organizing for digital innovation*, 2016.
- [14] C. Cohn, 'Build vs. buy: How to know when you should build custom software over canned solutions', *Forbes*, 15 Sep, 2014.
- [15] P. Andén, S. Comella-Dorda, A. Rocha, and T. Strålin, 'Teaching elephants to dance (part 2): Empowering giants with agile development', in *Software development handbook: Transforming for the digital age*, McKinsey&Company, 2016.
- [16] P. Andén and T. Strålin, 'Quality code: Driving successful software development', in *Software development handbook: Transforming for the digital age*, McKinsey&Company, 2016.

Contributing Authors

Louise Veling, *Research Fellow*, Innovation Value Institute.

Catherine Crowley, *Research Fellow*, Innovation Value Institute.

Linda Beckett, *Senior Development Manager*, AIB.

Graeme Clarke, *Senior Development Manager*, SQS.

Eamon Kelleher, *IT Manager*, Office of the Revenue Commissioners, Ireland.

John MacHale, *Consulting Director*, NashTech.

Laura Mc Quillan, *Agile Change Consultant*, Hawthorn Tree Consulting.

Shaun Percival, *Technical Architect*, BearingPoint.

About IVI

The Innovation Value Institute (IVI) is a multi-disciplinary research and education establishment co-founded by Maynooth University and Intel Corporation. IVI researches and develops management frameworks to assist business and IT executives deliver digitally enabled business innovation. IVI is supported by a global consortium of likeminded peers drawn from a community of public and private sector organizations, academia, analysts, professional associations, independent software vendors, and professional services organizations. Together, this consortium promotes an open ecosystem of research, education, advisory support, international networking, and communities-of-practice. IVI is supported through Enterprise Ireland's and IDA's Technology Centre programme.

Contact IVI

For more information on this capability, IT-CMF and other IT management topics, or on becoming a member of IVI's international research consortium, please visit www.ivi.ie or contact us at: ivi@nuim.ie or +353 (0)1 708 6931.



Innovation Value Institute, IVI, IT Capability Maturity Framework, and IT-CMF are trademarks of the Innovation Value Institute. Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this document, and the Institute was aware of a trademark claim, the designations have been printed with initial capital letters or all in capital letters.