Closing the GaaP: Lessons Learned from a Web-based Analysis Tool for Practitioners

Peter Kuhn[1], Dian Balta[1] and Florian Matthes[2]

[1]Fortiss GmbH, Guerickestraße 25, 80805 Munich, Germany
[2]Technical University of Munich, Arcisstraße 21, 80333 Munich, Germany

Abstract
Government as a Platform (GaaP) is an approach to government that enables user-friendly and efficient public services through the implementation of platform structures and principles such as modularization and co-creation. While some countries have been or are currently implementing the approach successfully, other countries experience barriers in the application of GaaP. Recent literature is attempting to address these barriers, for example by prac-tice-oriented conceptualizations of GaaP. However, those conceptualizations still need to be applied with expert knowledge. We develop a tool for practitioners that makes an exemplary GaaP concept applicable without expert knowledge. We present lessons learned from the development process and discuss implications for theory and practice.

Keywords
Government as a Platform, Platformization, Web App

1. Introduction

Government as a Platform (GaaP) is a promising approach to the digital transformation of the public sector. Following the GaaP approach, the public sector is transformed to an open platform on which people inside and outside government can innovate and contribute in order to co-create better public services [1]. GaaP can contribute to increased user-friendliness of public services [2] and higher efficiency of the public sector [3]. On a technical level, these benefits are enabled by modular and open platform infrastructures [4]. While some countries have been or are currently implementing the approach successfully (e.g. Italy [2] and the UK [4]), other countries experience barriers in the application of GaaP [5]. One perceived barrier is the lack of a uniform definition of GaaP as a concept. Recent literature is trying to close that gap by proposing models (e.g. [6]) and frameworks (e.g. [7]).

However, those conceptualizations are theoretical and need to be applied with expert knowledge. Arguably, there is a gap between the theoretical knowledge and the practicable applicability of GaaP concepts. For example, O'Reilly proposes the adoption of "service-oriented architecture for all your applications" [1], but the "how" remains open. This is also a challenge for GaaP theory, since the evaluation of its concepts depends of their applicability in practice.

In an attempt to close that gap, the goal of this ongoing research is to develop a tool that

EGOV-CeDEM-ePart 2022, September 06–08, 2022, Linköping University, Sweden (Hybrid)
pkuhn@fortiss.org (P. Kuhn); pkuhn@fortiss.org (D. Balta); pkuhn@fortiss.org (F. Matthes)
0000-0001-6774-2904 (P. Kuhn); - (D. Balta); 0000-0002-6667-5452 (F. Matthes)

allows the application of an exemplary GaaP concept by practitioners. This allows the collection of challenges of making GaaP concepts applicable and provides insights on how to solve them. To this end, we first develop a tool design and then develop a web app that follows that design. We apply an iterative approach with weekly sprints and conduct three expert interviews in order to ensure the suitability of the tool for practitioners. Finally, we infer three lessons learned from the tool development and discuss their implications. We contribute to practice by providing a tool that can be used to support the application of GaaP in practice. We contribute to theory by providing lessons learned that can help improve the conceptualization of GaaP concepts towards better applicability.

2. Theoretical Background

Platforms can be defined as systems that consist of a stable platform core and a variable periphery [8, 9], often in form of an ecosystem [10]. The link between those two parts has been conceptualized as boundary resources [11, 12]. For digital platform ecosystem three roles can be distinguished [13]: The platform owner owns the platform and makes fundamental decisions about the platform design and boundary resources. Complementors are external actors who create complementary functionality on the ecosystem and thereby create value. Users are other external actors who use the products and services created by complementors [13, 14]. The co-creation that happens between those actors is a constituting feature of platforms [15, 16]. On an infrastructure level, the transformation from silo-based to platform-oriented infrastructures is currently a topic in theory and practice [17, 18, 19]. The concept of GaaP was coined by Tim O'Reilly, describing it as viewing the Government as an open platform on which people inside and outside the government can innovate and contribute so that better public services can be co-created (O'Reilly, 2011). Over the years, several different perspectives on and conceptualizations of GaaP have been developed, e.g. GaaP as an approach to digital infrastructure [20]. Scholars have highlighted the benefits of GaaP, e.g. reduction of costs [6] and better outcome at the same time [21]. While the definition and conceptualization of GaaP is still subject of research [6, 22], several reoccurring underlying principles of the approach can be stated. E.g. GaaP builds upon openness [1, 22] and harnesses the innovative power from the outside [23, 1] by fostering participation [21, 1]. Crucially, the role of the state changes from a service provider to the owner of the platform [23, 21, 1].

The analysis of public sector initiatives regarding their platform character has been considered by many scholar. Brown et al. propose a "platform assessment framework" with the dimensions "organizational form", "market dynamic" and "architectural" structure" [4]. A similar distinction is made by Bender and Heine, who consider the "Platform Ecosystem", the "Platform Architecture", and the "Platform Governance" [6]. The framework by [7], considers "elements and roles" of the platform, its "theory-derived principles" and the platform "management and governance". The success of GaaP implementations in practice can be quantified using approaches from appropriate maturity models using different levels or stages. In general, e-government literature has made use of such stage models in various areas. For example, the four-stage maturity model in [24] is well-known for measuring success of eGovernment transformations, featuring the cultivation of intranet (i.e., horizontal and vertical integration), extension of services, maturity of openness, and revolution in broader service accessibility as the core levels. More recently, Ham et al. [25] has offered an extended and comprehensive extension of the stages in [24], providing legal, operational and technical criteria for measuring the openness of government innovation. The measurements in this research are inspired by examples and the DSR method for creating custom maturity models in IS [26].

3. Methodology

While the overall research endeavor is following the design science research paradigm [27], the methodology of this ongoing research paper can be described as a qualitative analysis approach to explorative research [28]. The goal was to develop a tool that is easy-to-use, derived from sound theoretical concepts, and provides measurable results for the application of GaaP and infer lessons learned from the process. We develop the findings through an argumentative-deductive analysis [28]. The research was organized around a practical course at Technical University of Munich that started in October 2021 and was finished in February 2022. Our research process can be described in three phases: 1. development of the tool concept, which happened before the course, 2. development of the tool - a web app - including 3 expert interviews, which happened during the course, and 3. the development of the lessons learned which happened after the course. The tool design happened based on our previous research and publications on GaaP, in particular [5, 7]. We drew from this knowledge when preparing the course project, which was designed to make sure that the task for the students is sound with requirements from GaaP theory and practice and at the same time realistic for the time constraints of the course. Using our knowledge of GaaP literature and the experience of the course from previous years we develop the tool design multiple meetings among the authors. The tool design is based on three requirements, infers challenges and defines a solution approach. The development of the web app was the main content of the course. The student team consisted of four graduate level students, who were supervised by the authors. With few exceptions, such as the Christmas break, the development team performed weekly development sprints which were discussed with the authors in weekly meetings. We used agile development tools (namely gitlab and trello) to track the progress and prioritise todos. The progress was presented to other students of the course in December for feedback on the usability of the tool. The three expert interviews were conducted in December and January 2021 and served the purpose of collecting feedback from practice regarding the tools suitability to support GaaP application in practice. The interviews lasted between 37 and 54 minutes and were recorded in order to be systematically analysed. The experts work at FITKO and are responsible for the enterprise architecture management of the federal IT architecture. In their role as enterprise architects, the experts apply the GaaP approach to an infrastructure of a federal country with decentral IT systems. Given that the FITKO has a coordinating role, the experts are reliant on the compliance and engagement of their stakeholders. In this context, they have interest in applying the tool because of its potential to explain their work and ideas to those stake-holders more easily and to justify decisions. The insights of the interviews were used in the following weekly meetings in order to adjust and prioritise the web app development. The lessons learned we inferred based on the accumulated insights from the weekly meetings. We focussed on the challenges of making the exemplary

GaaP concept applicable in practice. As the team of students didn't know the concept, the framework and its dimensions had to be explained by the authors. The pro-cess of translating the framework happened in numerous discussions between the students and the authors.

4. Designing a tool for the application of GaaP in practice

The goal of this research is to investigate the application of GaaP concepts in practice by means of a tool for practitioners. In order to do so we start from a simple set of requirements for such a tool. The tool shall be theory-based in order to ensure that insight from theory are considered in practice. The results of the tool shall be measurable, in order to provide concrete indications for practitioners on what to do and, finally, the tool shall be easy to use, in order to avoid any hurdles for practitioners to use the tool in their application of GaaP. This set of requirements raises several challenges, see figure 1.

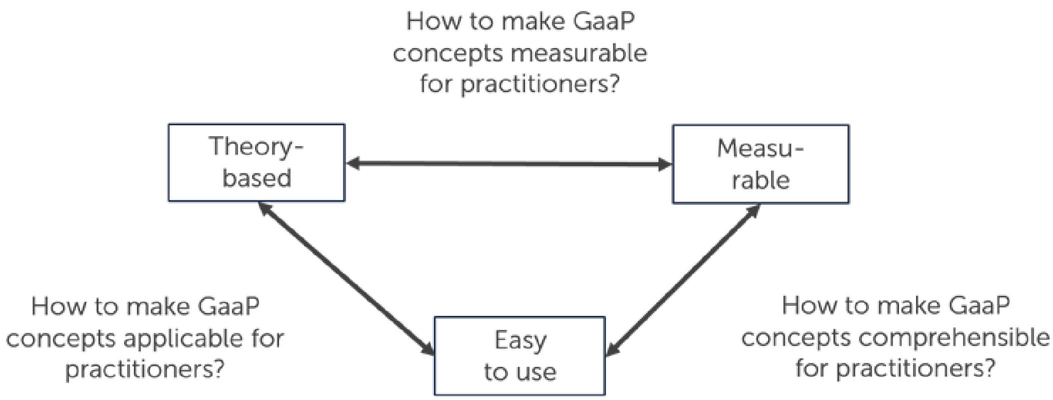

Figure 1: Challenges for the application of GaaP concepts in practice

How to make GaaP concepts applicable for practitioners? Literature on GaaP is rooted in practical observations and provides many hints for practitioners. A good example is the original publication by O'Reilly which is drawing from existing platforms in the real world, such as facebook, and proposes lessons. However, these lessons remain open for interpretation. For example, O'Reilly proposes openness for innovation and provides examples for that. Yet, the author provides no formula for practical decisions such as the balance between openness and access control of APIs. For a tool to be theory-based and easy to use at the same time, it has to bridge this gap. That is, it needs to find a solution on how to make GaaP concepts easily applicable for practitioners. In our prototype we try to achieve this by developing a web app, given the extensive existing knowledge on their UX and their flexibility to be adapted for very different purposes. **How to make GaaP concepts measurable for practitioners**? Literature on GaaP mainly provides principles and concepts. However, practitioners need feedback and quantifiable measures in order to steer and optimize their work towards GaaP. For a tool to be theory-based and measurable at the same time, it has to bridge this gap. That is, it needs to find a solution on how to make GaaP concepts measurable. In our prototype we try to achieve this by using maturity levels. Maturity levels have been extensively used in measuring stages in the e-government domain, e.g. [24]. **How to make GaaP concepts comprehensible for practitioners**? Literature on GaaP shows that various aspects have to be considered when applying the approach. However, the more aspects are considered the more difficult it is to explain all aspects and dependencies to practitioners. Literature on GaaP shows that communicating GaaP to stakeholders is important and difficult at the same time [5]. For a tool to be easy to use and measurable at the same time, it has to bridge this gap. I.e. it needs to find a way to make GaaP concepts comprehensible. In our prototype we chose to address this challenge by making the tool self-explanatory as much as possible. By doing so we avoid that expert knowledge is needed for its usage.

5. A maturity-level-based, self-explanatory web app

We built a web app [1] that serves the purpose of making an exemplary GaaP concept applicable in practice. We chose the framework [7] because practitioners are explicitly stated as its target group. The framework consists of three dimensions: platform principles, platform elements and roles, and platform management. In order to reflect the three dimensions of the framework, the web app consists of a one page application with three sections, each considering on of the dimensions (Figure 2). On the left, section A, a questionnaire on platform principles, in the middle, section B, a modelling space for elements and roles and on the right, section C, a result section with management tips are located. The design follows the work flow – from left to right – that a user would take: first, answer the questions on the left, then model the platform in the middle and finally read the results on the right.

The questionnaire section (A) consists of 15 questions covering the platform principles of the framework. The principles are covered with one to three questions per maturity level, where a positive answer (thumb up) indicates that statement of the question is fulfilled in the infrastructure at hand. A negative answer (thumb down) indicates the opposite. The underlying principle is not communicated to the user. The modelling section (B) consists of containers for the core, boundary and ecosystem components, as well as their elements and roles. The functionality is drag-and-drop. A tool bar provides additional functionalities. As a special feature, the background of the modelling area can be filled with a figure of the infrastructure, to support the easy tracing of the elements. The components can be connected with arrows and the direction of an arrow is adjustable. Each element has a dropdown menu to adjust its type, to copy and delete it. The results section (C) consists of a general score which runs from 0 to 100 percent (details on its calculation see next paragraph), a list of clues, indicating (im)perfect features of the modelled infrastructure and a breakdown of the questionnaire answers by principle.

The general score is calculated based on the questionnaire answers (50%) and an automated assessment of the infrastructure modelling (50%). The assessment of the questionnaire answers is realized by a maturity model in which two to three questions per principle and level determine the maturity and, thus, the score. The assessment of the infrastructure modelling is based on 26 underlying statements that – if true – increase the score. An example is the existence of a single container for the core, boundary and ecosystem. Another is that each container has at least on

[1]https://peng.fortiss-demo.org/platformize-webapp/

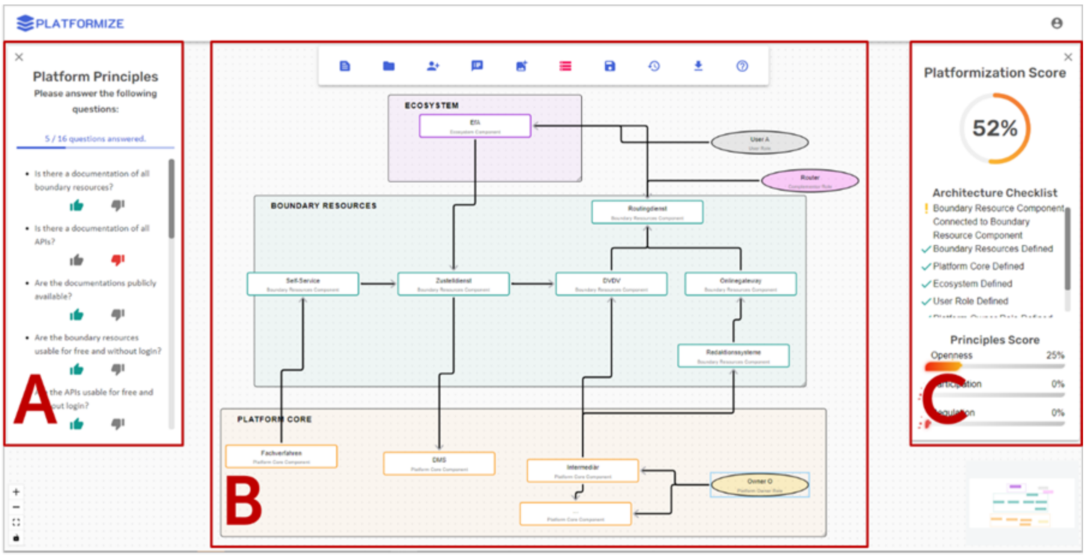

Figure 2: Screenshot of the web-based tool

component. If the statement is not fulfilled, a corresponding clue is created and displayed in the results section. The use of the tool is supported by a tutorial and is supposed to mirror the current platform. I.e. in a first step in order to analyze, the questions and the modelling happens as it currently is. This ensures a fair assessment. In a second step the results can be used to test improvements. E.g. if the clues suggest to add a boundary resource, this change can be entered into the tool and the effect on the score and the platform assessment can be tested. The tool has a report functionality which allows to print out the analysis with all questions and their answers, the modelling and the analysis results. In an exemplary use of the tool, a practitioner starts by answering the questions on platform principles such as openness on the right by clicking the buttons to signal affirmation or negation. Then the person models the infrastructure in question compromising its components and their interfaces. Lastly the practitioner uses the results section on the right in order to understand the current platformization level of the infrastructure and get tips on how to further improve the GaaP approach.

6. Discussion and lessons learned

In the course of this ongoing research we collected various insight on making GaaP applicable to practitioners. In the following we discuss the solution approach that we used and extend it by defining broader lessons learned from this research. A maturity-level-based knowledge semantic is suitable for measuring the GaaP-Progress in a score based on binary criteria (yes/no questions). However, the definition of the criteria proved to be challenging. For example, the existence of core components is arguably a valid criterion. The platform should have at least one core component. But the number and composition of core components is difficult to capture in yes/no criteria. While maturity levels provide a suitable frame for measurability they need to be filled with content. Making the app self-explanatory was realized through various means and on different levels. First, a general flow pattern from left to right is supposed to help understand the parts and the order of the analysis. Second, typical icons and visualization were used. Finally, features of the web app such as a tour guide offered help explaining the functionalities as well as the underlying concepts. While self-explanatoriness was pursued, the number of concepts and ideas that have to be explained when applying GaaP is big. Closing the knowledge gap for non-platform-experts can thus be challenging. Lastly, using a web app has many advantages such as easy sharing via an URL. Also, in contrast to apps, many devices support it. However, a web app also comes with a set of expectations for its functionality. For example, the drag-and-drop area had to be extended by many features such as key board short cuts which were not included in the used framework. While web apps are flexible and mobile, they also require a high level of usability.

Making GaaP concepts applicable for practitioners is challenging since the gap between theory and practice is wide. While the tool represent a handy approach to making theoretical concepts tangible in practice, its application still requires expert knowledge. A particular role for the experts is to interpret the results from the tool and to derive context-sensitive courses of action. This links also to the remaining leeway of maturity levels that we discussed. Making GaaP concepts measurable is challenging since there is a lack of descriptions of successful GaaP applications. Measurability needs criteria and without examples from practice these criteria are difficult to define. While there are some descriptions of GaaP in practice, e.g. from the UK [4] and Italy [2], other countries that apply the approach are not publicly documenting their approach. This leads to little knowledge to draw from and hinders making GaaP concepts measurable. Making GaaP concepts comprehensible suffers from the lack of narratives that support the explanation of GaaP aspects and features. While some narratives already exist, e.g. "Doing more with less" [3], other aspects such as the role of openness of GaaP are harder to explain and, thus, make it less comprehensible.

7. Conclusion

While some countries have been or are currently implementing GaaP successfully, (e.g. Italy [2] and the UK [4]), other countries struggle with the lack a clear definition of the approach. Recently, literature has provided conceptualizations of GaaP, which, however, are only applicable with expert knowledge. We developed a web-based tool that allows the application of an exemplary GaaP concept by practitioners. The web app is based on a tool design that we developed based on our previous GaaP research, in particular [5, 7]. In the process of the tool development we identify challenges and provide a solution approach in the web app. The insight that were gathered in the course of the research are presented as lessons learned, which show potential avenues for future research and provide implications for practice.

Our research is ongoing and has many limitations for its general applicability. Among others, we developed the tool for only one framework. Some or all of the lessons learned could be different for other conceptualizations of GaaP. Also, the tool has not been extensively evaluated in practice yet. So far, its added value can therefore not be assessed. The interviews gave a first positive feedback but further evaluation is necessary. Finally, the web app is a prototype. The usage of the tool in a daily setting would require additional functionalities such as saving and managing different analyses. As the web app did not have these functionalities, yet we could not considered these aspects in this paper. Finally, the focus of this research was technical. However, the implementation of GaaP has several other important dimensions, such as social and cultural aspects, which also should be considered. Despite these limitations, we draw the following conclusions. First, making GaaP conceptualizations applicable for practice is possible but challenging. Different from other concepts in the e-government domain, GaaP needs further research that supports the translation of theoretical into practical aspects. Second, many theoretical concepts such as openness would profit from more and detailed case descriptions in order to understand its subparts and mechanics in practice. While openness is already central in Tim O'Reilly's contribution and an important topic in the e-government domain in general, there are only few description on its practical aspects. Third, established analysis concepts such as maturity assessment schemas can contribute to making GaaP applicable in practice. This is to say, that GaaP literature can and should profit from drawing from existing research in related fields such as on maturity levels. General IS platform literature is another good example. Taken together, further research should provide more practice examples, develop more narratives and build concepts with tools in mind. This would involve developing an enhanced version of the tool in the course of an action design research [29]. An enhancement would involve linking the maturity level analysis to established maturity models (e.g. [24]).

While it is still ongoing, we believe that our research is potentially valuable to both theory and practice. In particular, we hope that the presented insights and future work support the application of GaaP in practice and increases user-friendliness and efficiency in the public sector.

Acknowledgments

This research was made possible in part by funding from the german NEGZ e.V. We want to thank Karim Arabi, Maximilian Eder, Ludwig Weide, and Marc Sichart von Sichartshofen.

References

[1] T. O'Reilly, Government as a platform. innovations: Technology, governance, Globalization 6 (2011) 13–40. URL: https://doi.org/10.1162/INOV_a_00056. doi:10.1162/INOV_a_00056.

[2] A. Cordella, A. Paletti, Government as a platform, orchestration, and public value creation: The italian case, Government Information Quarterly 36 (2019) 101409. URL: https://doi.org/10.1016/j.giq.2019.101409. doi:10.1016/j.giq.2019.101409.

[3] M. Janssen, E. Estevez, Lean government and platform-based governance—doing more with less, Government Information Quarterly 30 (2013) 1– 8. URL: https://doi.org/10.1016/j.giq.2012.11.003. doi:10.1016/j.giq.2012.11.003.

[4] A. Brown, J. Fishenden, M. Thompson, W. Venters, Appraising the impact and role of platform models and government as a platform (gaap) in uk government public service reform: Towards a platform assessment framework (paf, Government Information Quarterly 34 (2017) 167–182. URL: https://doi.org/10.1016/j.giq.2017.03.003. doi:10.1016/j.giq.2017.03.003.

[5] P. Kuhn, M. Buchinger, D. Balta, F. Matthes, Barriers of applying government as a platform in practice: Evidence from germany, in: Proceedings of the 55th Hawaii International Conference on System Sciences. Hawaii International Conference on System Sciences, 2022. URL: https://doi.org/10.24251/HICSS.2022.328. doi:10.24251/HICSS.2022.328.

[6] B. Bender, M. Heine, Government as a platform? constitutive elements of public service platforms, in: A. Kö, E. Francesconi, G. Kotsis, A. Tjoa, I. Khalil (Eds.), Electronic Government and the Information Systems Perspective, Springer International Publishing, 2021, p. 3–20. URL: https://doi.org/10.1007/978-3-030-86611-2_1. doi:10.1007/978-3-030-86611-2_1.

[7] P. Kuhn, S. Dallner, M. Buchinger, D. Balta, Towards "government as a platform": An analysis framework for public sector infrastructure, Wirtschaftsinformatik (2022). URL: https://aisel.aisnet.org/wi2022/e_government/e_government/4, proceedings. WI 2022.

[8] C. Baldwin, C. Woodard, The architecture of platforms: A unified view, SSRN Electronic Journal (2008). URL: https://doi.org/10.2139/ssrn.1265155. doi:10.2139/ssrn.1265155.

[9] A. Tiwana, Platform Ecosystems: Aligning Architecture, Governance, and Strategy, Newnes, 2013.

[10] M. Jacobides, C. Cennamo, A. Gawer, Towards a theory of ecosystems, Strategic Management Journal 39 (2018) 2255–2276. URL: https://doi.org/10.1002/smj.2904. doi:10.1002/smj.2904.

[11] A. Ghazawneh, O. Henfridsson, Governing third-party development through platform boundary resources, 2010. URL: http://urn.kb.se/resolve?urn=urn:nbn:se:hj:diva-14028.

[12] A. Ghazawneh, O. Henfridsson, Balancing platform control and external contribution in third-party development: The boundary resources model, Information Systems Journal 23 (2013) 173–192. URL: https://doi.org/10.1111/j.1365-2575.2012.00406.x. doi:10.1111/j.1365-2575.2012.00406.x.

[13] A. Hein, M. Schreieck, T. Riasanow, D. Setzke, M. Wiesche, M. Böhm, H. Krcmar, Digital platform ecosystems, Electronic Markets 30 (2020) 87–98. URL: https://doi.org/10.1007/s12525-019-00377-4. doi:10.1007/s12525-019-00377-4.

[14] M. van Alstyne, G. Parker, S. Choudary, Pipelines, platforms, and the new rules of strategy, Harvard Business Review 94 (2016) 54–62.

[15] M. Reuver, C. Sørensen, R. Basole, The digital platform: A research agenda, Journal of Information Technology 33 (2018) 124–135. URL: https://doi.org/10.1057/s41265-016-0033-3. doi:10.1057/s41265-016-0033-3.

[16] M. Schreieck, M. Wiesche, H. Krcmar, Capabilities for value co-creation and value capture in emergent platform ecosystems: A longitudinal case study of sap's cloud platform, Journal of Information Technology 026839622110237 (2021). URL: https://doi.org/10.1177/02683962211023780. doi:10.1177/02683962211023780.

[17] B. Bygstad, O. Hanseth, Transforming Digital Infrastructures through platformization, ECIS, 2018.

[18] B. Bygstad, E. Øvrelid, Architectural alignment of process innovation and digital infrastructure in a high-tech hospital, European Journal of Information Systems 29 (2020) 220–237. URL: https://doi.org/10.1080/0960085X.2020.1728201. doi:10.1080/0960085X.2020.1728201.

[19] R. Törmer, S. Henningsson, in: Platformization and Internationalization in the LEGO Group. HICSS Proceedings. HICSS, 2020. URL: https://doi.org/10.24251/HICSS.2020.709.

doi:10.24251/HICSS.2020.709.
[20] R. Pope, Playbook: Government as a platform, 2019.
[21] J. Millard, Open governance systems: Doing more with more, Government Information Quarterly 35 (2018). URL: https://doi.org/10.1016/j.giq.2015.08.003. doi:10.1016/j.giq.2015.08.003.
[22] H. Seo, S. Myeong, The priority of factors of building government as a platform with analytic hierarchy process analysis, Sustainability 12 (2020) 5615. URL: https://doi.org/10.3390/su12145615. doi:10.3390/su12145615.
[23] J. Fishenden, M. Thompson, Digital government, open architecture, and innovation: Why public sector it will never be the same again, Journal of Public Administration Research and Theory 23 (2013) 977–1004.
[24] K. Andersen, H. Henriksen, E-government maturity models: Extension of the layne and lee model, Government Information Quarterly 23 (2006) 236–248. URL: https://doi.org/10.1016/j.giq.2005.11.008. doi:10.1016/j.giq.2005.11.008.
[25] J. Ham, J.-N. Lee, D. Kim, B. Choi, Open innovation maturity model for the government: An open system perspective, in: ICIS 2015 Proceedings, 2015. URL: https://aisel.aisnet.org/icis2015/proceedings/eBizeGov/15.
[26] J. Becker, R. Knackstedt, J. Pöppelbuß, Developing maturity models for it management, Business & Information Systems Engineering 1 (2009) 213–222. URL: https://doi.org/10.1007/s12599-009-0044-5. doi:10.1007/s12599-009-0044-5.
[27] A. Hevner, A. R, S. March, S. T, P. Park, R. J., Sudha, Design science in information systems research, Management Information Systems Quarterly 28 (2004) 75.
[28] T. Wilde, T. Hess, Forschungsmethoden der wirtschaftsinformatik, WIRTSCHAFTSINFORMATIK 49 (2007) 280–287. URL: https://doi.org/10.1007/s11576-007-0064-z. doi:10.1007/s11576-007-0064-z.
[29] M. Sein, O. Henfridsson, S. Purao, M. Rossi, R. Lindgren, Action design research, MIS Quarterly 35 (2011) 37–56. URL: https://doi.org/10.2307/23043488. doi:10.2307/23043488.