Technical Debt Management: A systematic literature review and research agenda for digital government

Mille Edith Nielsen[1, 2], Christian Østergaard Madsen[1, 2], Mircea Filip Lungu[1]

[1] The IT University of Copenhagen, Denmark
[2] Research Centre for Government IT
{miln, chrm, mlun}@itu.dk

Abstract. Technical debt is created when software engineers knowingly or unknowingly introduce shortcuts or unsuitable choices in the development or maintenance of the software system, that will have a negative impact on the future evolution of the system until corrected. Therefore, it is crucial to manage established debt particular in the public sector. The aim of this study is to introduce Technical debt to the field of Digital Government. We create an overview of the state of the art of the knowledge on technical debt management, the methods applied to gain this knowledge, and propose a research agenda to Digital Government scholars. We conduct a systematic literature review, which focuses on the concept of technical debt management. Forty-nine papers published within 2017-2020 are selected and analyzed. We identify several gaps in the existing literature: 1) an absence of theory explaining the relation of events, 2) a shortage of studies conducted in the public sector, 3) and an absence of specific techniques such as observation to study actual technical debt management behavior.

Keywords: Technical debt management, managing legacy systems, systematic literature review, technical debt

1 Introduction

Public and private organizations use information technology (IT) to improve their efficiency and service offerings [1]. During the development of IT-projects and the subsequent operation/maintenance, temporary suboptimal solutions are sometimes introduced to profit from the solutions faster. To capture this reality, Cunningham [2] coined the term technical debt (TD) to explain the process and pitfalls of programming to the management in the banking sector in 1992. Rios et al [3, p. 117] describe TD as a conceptualization of *"problems faced during software evolution considering the tasks that are not carried out adequately during software development."* They conducted a tertiary literature review and found a variation in the application of the term. TD is often associated with any impediment related to the software product and the development process [3]. Griffith et al describe technical debt management (TDM) as comprising *"the actions of identification, assessment, and remediation of technical debt throughout a software system."* [4, p. 1016].

TDM is important because it enables an organization a more optimal use of its resources [5]. For instance: An organization's IT-system breaks down, which leads to reduced production. For management, the possible solutions are to 1) accept the reduced production, 2) conduct a root cause analysis and fix the actual problem, 3) create a work-around that increases momentarily production speed, but does not solve the actual problem and will make the maintenance of the solution harder in the future. Solution 3 may be a viable course of action under certain circumstances, but it will result in the creation of technical debt. Moreover, since the consequences are no longer visible to management, this debt may be forgotten. If proper TDM methods were applied, management could identify the debt and repay it in a timely fashion.

We will claim, TD is important for the field of Digital Government, because TD can hinder the public sector in fully reaping the benefits of digitalization. Scholl encourages the field of Digital Government to engage with other disciplines *"which overlap with Digital Government as a practice area, but which might lack the forward-looking capabilities that Digital Government Research at least can provide in part."* [7, p. 11]. Moreover, Digital Government scholars can contribute to TD research, because they have both domain specific knowledge of the public sector's use of IT and methodological experience in studying IT and operation in this context [8].

In 2017, The Danish Ministry of Finance published an analysis of Danish public IT-systems. The report concluded that 157 of 428 society or business-critical IT systems' technical components (applications and IT-infrastructure) were not fully maintained [9]. Outdated software or hardware components can increase the risk of breakdowns, security breaches, and trouble the maintenance and the future development of the IT-systems maintained [9]. The Swedish National Audit Office conducted a similar analysis and found 70% of their IT-systems were outdated [10].

TD and the concept of legacy systems *"discuss a state of software that is sub-optimal, time constrained, and explain how this state can decrease an organization's development efficiency"* [8, p. 80]. A considerable source of TD originates from software legacy [12], e.g. during continuous development of a system in an outdated environment. In this review, we focus solely on TDM.

TD studies are primarily published in Software Engineering, especially after Cunningham's [2] introduction of the metaphor [3, 5, 13, 14]. However, Information Systems researchers have also published studies on TD [15-16].

We found nine literature reviews and one tertiary review on TD and TDM (Appendix A). While these reviews offer important contributions, they only cover the literature up until 2017 and do not focus on examining methodology, use of theory or unit of analysis. In this study we aim to address these gaps.

We identify 49 TDM papers published within 2017-2020. We find the focus of the TD research fragmented: TD decreases morale, TD is difficult to measure, and numerous tools. The MTD workshops and a tertiary study encourage more research on strategies and management [P13, 3]. The papers primarily present data from open source projects and the private sector. This leaves a gap for research in the public sector. Finally, we offer a research agenda for Digital Government scholars on TDM.

1.1 Aim of the study and research questions

This study reviews the latest published TDM papers (2017-2020). To the best of our knowledge, these papers have not been studied by other secondary studies. The TD field is rapidly evolving with nine secondary studies being published in the past six years. Our study indicates that the publishing rate has not decreased. In a four-year period, TD papers were published in 30 different outlets. This study introduces TD research and a research agenda to the Digital Government scholars. Therefore, we follow established guidelines for conducting systematic literature reviews within the Digital Government field [17, 18]. Our research questions are:

- RQ1. How is TDM studied and in which fields? RQ1a: Which authors and fields have contributed to technical studies debt management since 2017? RQ1b: What are the methods, context, level of analysis and data level used? RQ1c: Which theories and theoretical concepts are applied?
- RQ2. What does the TDM literature focus on? RQ2a Which topics are studied? RQ2b What do the findings show?
- RQ3. What research agenda should the Digital Government scholars investigate in the context of TDM? RQ3a What suggestions does the literature have for future studies? RQ3b What are the identified knowledge gaps in the literature?

The following sections are organized as follows: 2) search process, 3) brief mapping of previous TD literature reviews, 4) analysis, 5) results and suggestions for future study, 6) discussion, limitations, and finally, 7) conclusion. The previous literature reviews, and the pool of papers are listed in Appendix A and B.

2 Search process

The initial search for papers occurred from January to April 2019, with an updated search conducted in March 2020. We applied Webster and Watson's [18] method for conducting systematic literature reviews. Webster and Watson present a three-step process to search for papers [18].
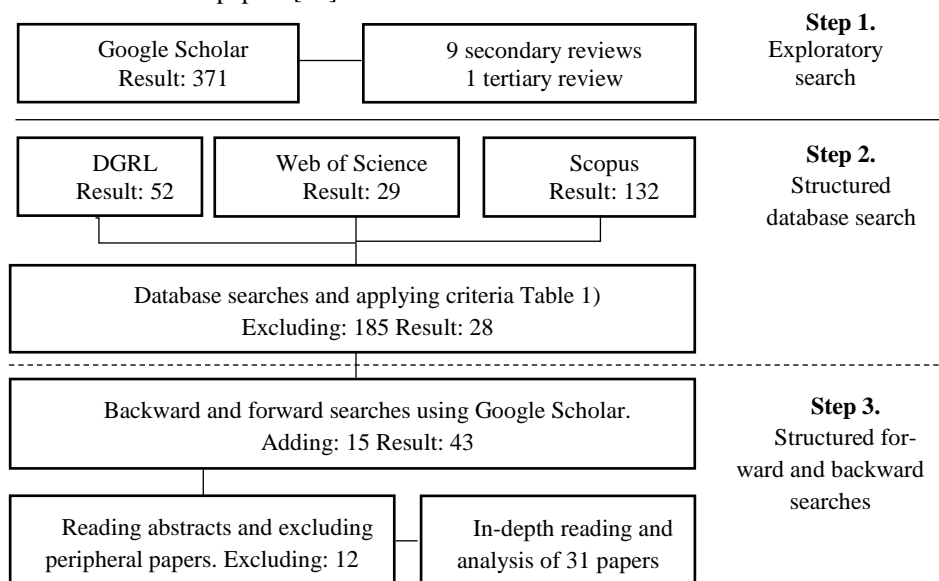

Step 1.
Exploratory search

Google Scholar
Result: 371

9 secondary reviews
1 tertiary review

Step 2.
Structured database search

DGRL
Result: 52

Web of Science
Result: 29

Scopus
Result: 132

Database searches and applying criteria Table 1)
Excluding: 185 Result: 28

Step 3.
Structured forward and backward searches

Backward and forward searches using Google Scholar.
Adding: 15 Result: 43

Reading abstracts and excluding peripheral papers. Excluding: 12

In-depth reading and analysis of 31 papers

Figure 1 illustrates the initial process.

Webster and Watson recommend, that scholars begin a review by searching for papers in known key outlets. The second step is a database search, as it enables the researcher to discover other fields. The third step is a backward-forward search, where papers citing or cited by the pool of papers are identified. The following sections explain each search step in detail.

1. Exploratory search: Using the software tool 'Publish or Perish' (Google Scholar)
2. Database search: Web of Science, DGRL and Scopus
3. Backwards and forward search (Google Scholar citations)

2.1 Explorative search (Step 1)

Webster and Watson [18] recommend starting a review by searching key outlets. They assume that the researcher is familiar with the literature and key outlets. However, we were not familiar with these at the time. Therefore, we chose to deviate from their method in the first step. In January 2019, we conducted an explorative search through Google Scholar to become familiar with the topic and academic literature [19]. We used the software "Publish or Perish" and Google Scholar as an underlying search engine [20]. This search created a foundation for the second step: the structured literature database search. The search informed: the selected search term, the selection criteria, and the temporal limitation. In this search process, we discovered nine secondary literature reviews and a tertiary review (Section 3 & Appendix A). The secondary and tertiary reviews informed our inclusion and exclusions criteria (table 1).

Table 1. Selection criteria (+ included / - excluded)

+/-	Assessment criterion	Argument
+	Topic should be TDM	Only papers focusing on TDM were included
+	Published within the period 2017-2020	The papers posted up until 2017 have been covered by other literature studies [21, 22]
+	Published conference or journal papers only	To ensure high quality, only papers that had undergone a peer-reviewed process were included
+	Only papers written in English	Due to the authors' language skill, we included papers in English only.
-	Books, chapters, blog posts, citations, thesis, presentations, reports, editorials & summaries	The scope of this literature review is to analyze the peer-reviewed papers. To avoid research duplicates books and books chapters were excluded [17].
-	Inaccessible papers	Papers, which appeared in the search but could not be accessed were excluded
-	Duplicates	Only the newest version of the paper was included (typically journal paper). Journal papers are generally more in-depth than conference papers [17].

2.2   Structured database search (step 2)

In February 2019, we searched in the three databases: Digital Government Reference Library (DGRL), Scopus and Web of Science. DGRL version 15.5 contains references to approximately 12.500 peer-reviewed papers within the digital government field.

Webster and Watson [18] suggest using keywords when searching in databases, However, not all papers are presented with keywords in these databases. Therefore, we searched the databases for papers including the expression 'Technical Debt Management' anywhere in the text. We identified 213 primary research papers on TD.

Due to a large number of identified papers, and the existence of secondary and tertiary studies, we decided to introduce two additional selection criteria in our study (table 1). This was done to position our review in relation to the previous secondary and tertiary studies. The secondary studies cover all papers on TD published until early 2017 (see Figure 2). Thus, we focused the review on 1) papers studying TDM specifically, this brought the number of papers to 130, 2) papers published from 2017 and onwards, which reduced the number to 28 papers.

2.3   Forward/backward search (Step 3)

In April 2019, we conducted backward and forward searches [18]. First, we reviewed the references in the previously identified papers (backward search) adding two papers. Second, we used Google Scholar to conduct a forward search for papers citing the identified papers, adding 13 papers. This brought the pool of papers to 31.

In March 2020 we updated the search and repeated step 2 and 3, we identified 18 new papers, bringing the total pool of papers in this review to 49 (Appendix B).

3   Previous secondary studies on TD

In the explorative search, we found nine secondary and one tertiary study. The nine studies cover literature published from 1992-2017 (Figure 2). Benldris et al. [22] and Becker et al. [21] cover the year 2016 completely with a broad view of TD. We position
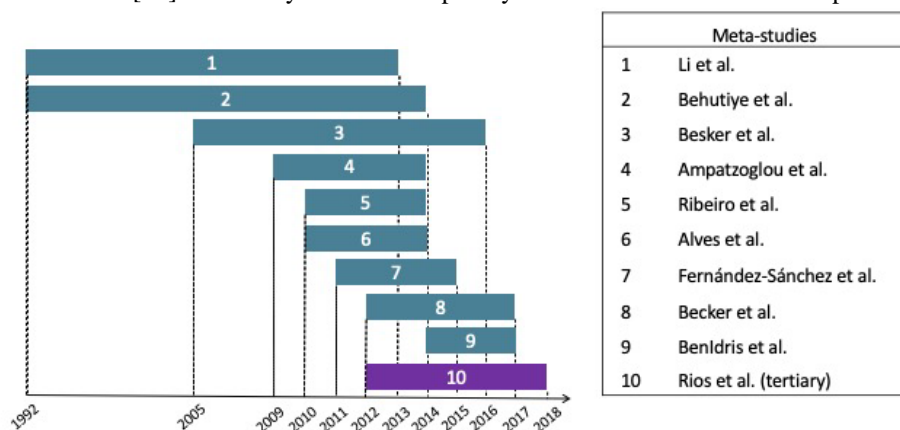

Figure 2 illustrates previous meta-studies research on TD

our literature review to these existing secondary studies in the discussion section. Rios et al. published a tertiary study in 2018, which explores the state of TD research [3]. They evaluate 13 secondary studies from 2012 to March 2018. They identify three research directions and concepts studied in the secondary studies: TD identification, TD concepts, and TDM. They develop a taxonomy of 15 TD types and generate a TD landscape mapping out the TDM activities, strategies and tools. The period from 2017 does not appear to be covered and none of the literature reviews examine the context of the data.

4 Method for analysis

The total pool of papers in our review includes 49 papers. Next, we explain the overall coding process, then we go more into detail in the following section.

The coding process was conducted iteratively. We developed a coding sheet (a template) containing: origin of data, theory and method, and unit of analysis [23]. We applied the template on 10 papers and discussed and adjusted the categories. The first author conducted the coding, while coding issues were discussed among all authors.

4.1 Detailed description of coding elements

This section describes additional categories besides author and journal. We explain the reason for our coding, and the coding process in detail.

The origin of the data. We coded for country and sector (private or public) to uncover where the data originates from. We expanded the categorization to include open source projects. Open source projects allow for anyone, anywhere to contribute to the code without stating who they are or where they are from. In the papers, the authors describe where they extracted the data, e.g. by state the type of organization, from which we could interpret the sector.

Theory and technique. We made an open coding for the technique; thus, we included a description of how the data was extracted. We coded for theory, focusing on explanations of the relation between concepts, observed phenomenon and why these relationships exist [24–26].

Findings and future studies. We coded the papers' main findings, typically from the results, discussion, and conclusion sections to identify the latest findings within TDM studies. Additionally, we coded for the authors' suggestions for future studies.

Unit of analysis and concepts. Webster and Watson [18] suggest creating a concept matrix and adding another dimension: the level of analysis. This dimension analyzes the abstraction level of the paper. This allows for more accurate identification of the existing literature. They suggest three levels of analysis: individual, group and organization. We discovered several papers that analyzed at the IT-system level and added this to the existing three levels of analysis. We coded for the overall concept within the paper, besides TDM. We began with a careful read-through of the first 10 papers. Here, a pattern emerged, and we identified four concepts. They were confirmed after going through the remainder of the pool of papers.

5 Results

Next, we present the findings of our study according to our research questions.

5.1 RQ1 How is TDM studied and in which fields?

RQ1a: Which authors and disciplines have contributed to TDM research since 2017? Papers concerning TDM have been published in 30 different journals and conferences since 2017. The eight outlets containing more than one paper are listed in table 2, the remaining 23 papers are published in 26 different outlets.

The International Conference on TD contains the most papers regarding TDM followed by the Journal of Systems and software. The mapping shows that TDM appears in a variety of outlets within the field of Software Engineering. The 65% of the papers do not specify the countries where the study was conducted. The papers which do specify this, primarily come from the Nordic countries - particularly Sweden [P2, P4, P11, P18, P21, P45]. The papers not specifying the country mainly use open source projects.

Table 2. The most frequent publishing outlets for TDM research

Outlet	No. of papers
International Conference on Technical Debt	7
Journal of Systems and Software	4
ACM SIGSOFT Software Engineering Notes	2
ACM SIGSOFT Software Engineering	2
IEEE Transactions on Software Engineering	2
International Conference on Software Engineering	2
Euromicro Conference on Software Engineering and Advanced Applications	2
International Conference on Software Maintenance and Evolution	2

124 different authors have contributed to the literature of TDM, 15 of them contributed to two papers. Figure 3 presents the top contributors, note that many papers have several authors.
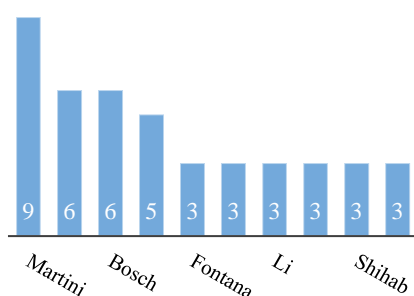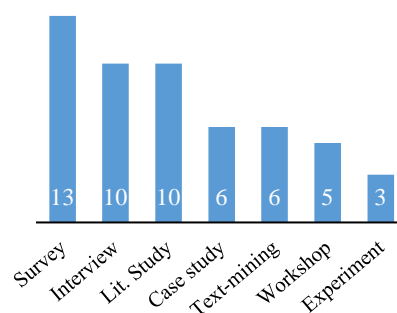
Fig. 3. The most active authors

Fig. 4. The most frequently used techniques

RQ1b: What are the methods, context, level of analysis and data level used? Only half of the papers specify the organizational context, which the data originates from. One paper use data from the public sector, 12 papers present open source data, and 12 papers present data from the private sector. The papers use different techniques during their research. The most frequently used techniques are survey, interviews and literature studies (figure 4). Note that some studies use more than one technique.

   RQ1c: Which theories and theoretical concepts are applied? The authors introduced different concepts and frameworks to explain TD and the underlying interrelations. This was done from different perspectives; assessment, working environment or awareness. The papers seldomly use theory and the word 'theory' rarely occurs in the text of the papers. Two papers are presented as foundation for future theory [P39, P45]. The papers offer indicators and methods on how to reduce TD.

5.2 RQ2: What does the TDM literature focus on?

RQ2a: Which topics are studied? Inspired by Webster and Watson [18], we have identified four main concepts and mapped the papers' level of analysis into a concept matrix table 3). This gives an overview of how the concepts have been studied and which levels or concepts have not received attention in recent studies.

   The concept *TD Assessment* entails research which assesses the effect of TD or as it is called, *interest rate*. This concept is the most analyzed and it has been analyzed at four different levels. Ten of the papers use the concept *Self-admitted technical debt*, which covers TD that is consciously admitted and is visible in code comments. This has been analyzed at the IT-System and Project level. The third concept of *TD awareness* covers the research creating awareness of TD. This can be achieved by visualizing the assessed TD; it is analyzed at the IT-system and the project level. The last concept *Working environment* focuses on morale and organizational culture. This is analyzed at both an individual and an organizational level.

Table 3. Concept matrix illustrating the studied topics

Level of analysis	Individual employee	IT-System	Project	Organization
Self-admitted TD		P28, P19, P31, P17, P30, P27	P12, P23, P8, P41	
TD assessment	P4, P10, P33, P44	P20, P26, P5, P14, P31, P27, P37	P16, P22, P6, P14, P34, P38	P5, P15, P9, P21, P35, P39
TD awareness	P45, P46, P49	P24, P26, P19, P28, P17, P42, P43, P48, P37	P16, P22, P6, P47	
Working environment	P1, P2, P3, P11			P1, P18, P46

A third of the papers analyze TD at the IT-System level, the most frequent analyzed level. The primary focus is TD assessment, which is explored in 21 papers. The organi-

zational level is the least used level of analysis; one explanation is that organizational data is more challenging to access than surveys and on open source projects.

RQ2b: What do the findings show? Almost a third of the papers propose a tool, method, model or technique to aid TDM [P8, P16-20, P29-31, P34, P35, P41, P42, P47, P48]. Their findings focus on the results of evaluating their presented tools. The tools generally aid in TDM e.g. through TD identification or visualization. Increased TD visibility can benefit communication between stakeholders [P37].

The papers vary in approach and focus; thus, the findings of the papers are fragmented. We list seven general findings from the pool of papers: **1)** TD harms software development work [P2-4, P11]. **2)** All roles related to the system are affected [P4], and community-related factors contribute to TD's intensity [P46], however, **3)** the developer morale can be increased by proper management [P11]. **4)** Organizational factors can influence TD [P1], the number of collaborators and the size of the project correlate significant with the amount of TD [P14], and the breadth of the developers' experience lower the amount of TD [P1, P33]. **5)** TD is time consuming: practitioners estimate 36% of development time is wasted due to TD [P4], and TD increases the need to perform additional time-consuming activities [P3]. However, TD is not visible in the backlog [P45] and lack of development processes increase TD [P1] **6)** Architectural debt should be managed early in the process because the early introduction of architectural debt shows it persists during the whole software lifecycle [P2]. **7)** The estimation difficulties are proposed to be solved by a workflow, which *"provide more actual information including TD concepts to the stakeholders"* [P7, p. 600-601].

5.3 RQ3: What research agenda should the Digital Government scholars investigate in the context of TDM?

RQ3a What suggestions does the literature have for future studies? The papers proposing a model, tool, method, approach or technique for TD aim to continue expanding and validating their model against new datasets [P1, P5, P8, P12, P14-15, P18-19, P23, P27, P29-31, P34-35, P41-42, P47-48]. Suggestions for future research for other researchers are scarce. However, both MTD workshops encourage a slight change in direction and provide several suggestions for future studies [P10, P13].

Quantifying the value of TD. The report of MTD workshop of 2016 [P13] suggests the research agenda is on defining, understanding and operationalizing the value of TD. They urge researchers to understand the value that falls outside the core definition of TD, which is essential to how TD plays out in practice. Digital Government scholars can contribute to this with their experience. The 2017 report of MTD workshop [P10] recommends elevating the quantification of TD from low-level code to architectural opportunities. They identify a need to educate stakeholders to raise awareness level.

A better understanding of the metrics. Three papers contain the following suggestions for future work: 1) to research more important metrics in the future [P16], 2) to understand the factors leading to TD [P14], 3) to study more change features that can introduce TD [P31]. Other research should be undertaken to investigate if other types of TD (besides Architectural Debt) have a significant correlation to the estimated wasted time. Thus, it creates a better understanding of the negative impact different TD

has on wasted software development time [P2-3]. Lastly, the organization's maneuverability can be increased by determining the types of debt incurred [P18].

Strategies. Specific suggestions are to investigate concrete architecture problems: how they contribute to file bug-proneness, and possible ways of refactoring [P6]. Vadja et al. encourage future studies to develop methods to assist the stakeholders to estimate their TD [P22] or increase the breadth of the experience [P33]. Additionally, Dong et al suggest exploration of TD recovery strategies due to a lack of discussed actual cases – particularly in the cross-disciplinary environment [P9]. Further research should focus on providing an in-depth understanding of the relationship between TD and developers' morale [P11]. Two papers aim to build a TD theory [P39, P45].

RQ3b What are the identified knowledge gaps in the literature? We identify five gaps which are presented in Table 4. 1) The papers suffer from a lack of theory and therefore cannot explain the relationship between events and concepts [26]. The debt metaphor contains some explanatory power; however, a metaphor cannot substitute for a theory [27]. Two papers present research as possible foundation for theory [P39, P45]. 2) The level of analysis is primarily focused on a project or system level, which leaves a gap in researching the organizational and individual levels. 3) Half of the papers have not specified the organizational context of the data, and the other half investigate open source projects or private companies. Only one paper use data from the public sector, this leaves a gap in research in this sector [P41]. 4) TD is explored through 16 different approaches and techniques - both quantitative and qualitative. Eight papers use more than one technique and combine both quantitative and qualitative methods. However, observation is not used actively as a data gathering technique.

Table 4. Identified gaps, their importance, and suggestions to how they can be addressed

Gap	Why is this a problem	Suggestion to how the gaps can be addressed
1. Lack of research from the public sector	The public sector has other restrictions than open source and the private sector. The TD challenges and management might differ.	Study TD in the public sector.
2. Lack of Theory	Understand the connection of events and why they occur will help in determining how to prevent them from happening [29].	Develop a new theory or import theory from other fields.
3. Lack of venue diversity	TD is relevant to other fields. Different disciplines may offer different perspectives and approaches to TD.	Introduce TD to other fields than Software Engineering
4. Limited abstraction level of analysis	Examining TD from different levels generates a fuller picture. TD's impact goes beyond the IT-system.	Examine TD from an organizational or individual level.
5. Lack of technique diversity	Observations may inform us about people's actual behaviour, rather than how people claim they act [3].	Use observation to address people's behaviour.

This may be a problem, because what people say during interviews, may differ from what they do [28]. 5) All the papers are published within the field of software engineering leaving a gap for venue diversity. Table 4 summarizes the identified gaps in the papers and suggests why and how these gaps could be addressed.

When we compare the TD literature's suggested research RQ3a and the gaps we identify, we find that they can easily be combined. Our gaps contain a high abstraction level, whereas the suggested future research provides very specific suggestions.

6   Discussion

First, we compare the findings of this review to comparable secondary studies. Second, we identify and discuss five findings further, 1) lack of research in the public sector, 2) empirical confirmation of negative effects of TD, 3) a tendency of reinventing the wheel, 4) gaps in diversity of approaches, and 5) the absence of theory.

Comparison to related work. Becker et al. [21] focus on decision-making and criticize the method and objective of the research. They find that the actual decision making was not studied. Ampatzoglou et al. [13] explore how financial aspects are defined and applied when studying TD. They encourage a balance between economic theories and software engineering. Li et al [14] find that code-related TD and its management gain the most attention and encourage future studies to explore the whole TDM process. This corresponds to our findings in the concept matrix, as research on the organizational level or the working environment is limited. The tertiary review by Rios et al. [3] underline several gaps, which are in line with the MTD reports [P13, 3]: more research on strategies and management.

This literature review includes management of all types of TD, in line with several of the secondary studies [5, 13, 14, 21, 30]. However, none of the secondary studies analyze theory nor the organization type the data was extracted from. Furthermore, this review offers insights into the recent literature and present the concept and a research agenda to Digital Government scholars.

Lack of research in the public sector. Half of the papers do not state the organizational context where data was collected. Only one paper appears to have conducted its data collection within the public sector [P41]. This is important, because the public sector is different from open source projects and the private sector. While some challenges may be similar, public organizations are subjected to specific requirements [32]. We suggest future Digital Government studies focus on studying TDM, both in terms of strategies, management and indicators.

A tendency to reinvent the wheel. About a third of the papers propose a tool to support technical debt management [P8, P16-20, P29-31, P34, P35, P41, P42, P47, P48]. They evaluate their presented tool and plan to apply it within a different context. They rarely apply other researchers' tools or methods, instead, they develop their own. This is problematic as it does not advance the field through joint effort; instead of providing and testing a few select tools for practitioners, the number of tools becomes overwhelming.

Empirical confirmations of the negative effects of TD. Technical debt harms software development work, both in terms of morale and wasted time [P2, P3, P4, P11].

12

However, community factors can also intensify TD [P46]. The size of the project correlates with the amount of TD [P14]. TD can be minimized by following some simple guidelines. However, if TD is addressed early it is less time consuming and the morale can be increased [P2, P3, P4, P11]. Thus, we need future studies to explore indicators of TD.

Gaps in the diversity of approaches. TDM is primarily studied from a quantitative approach, 30% of the papers used a qualitative approach. Six of the 31 papers use more than one method. This leaves a gap which can be addressed with the mixed method approach. Mixed methods may provide more comprehensive evidence [24]. None of the papers apply observation, this leaves people's actual behavior concerning TDM unstudied.

Absence of theory. Concerning results, we see an absence of the use of theory, a few papers import theory and approaches from other fields, e.g. portfolio finance and games. We need theories to explain the relationship between the concepts and why these exist [24–26]. The absence of native theory is natural for a young field, where researchers are prone to import theory from a different field instead [33, 34].

7 Conclusion, limitations and future studies

We have introduced TDM to the field of Digital Government and proposed a research agenda for Digital Government scholars. We conducted a systematic literature review to explore: 1) how and where technical debt management is studied, 2) what we know about technical debt and its management and lastly 3) a research agenda for scholars within Digital Government. Our findings are based on papers published from 2017-2020. We have discovered that 1) researchers focus on a specific type of technical debt and how it can be reduced. 2) TDM is still strongly rooted within the field of Software Engineering and 3) is primarily examined in open software projects or the private sector. 4) TDM is studied using primarily quantitative methods, finally, 5) there is a lack of theory to guide the studies and explain findings.

Limitations. We decided to only use the term "Technical debt management" in our literature search which can decrease the external validity, however the backward/forward searches strengthen the external validity. The papers were coded by the first author only which decrease the internal validity, in order to strengthen the internal validity, issues were discussed among all authors.

Research agenda. We suggest Digital Government scholars research technical debt management, TD strategies and TD indicators, so the field is advanced further.

8 Acknowledgement

Appendix A: Previous TD literature reviews

Authors	Title	Year	No. of papers	Period covered
Alves, Mendes, De Mendonça, Spinola, Shull, Seaman	Identification and management of technical debt: A systematic mapping study	2016	100	2010-2014
Ampatzoglou, Ampatzoglou, Chatzigeorgiou, Avgeriou	The financial aspect of managing technical debt: A systematic literature review	2015	69	2009-2014
Behutiye, Rodríguez, Oivo, Tosun	Analyzing the concept of technical debt in the context of agile software development: A systematic literature review	2017	38	1992-2014
Ribeiro, de F. Farias, Mendonça, Spínola	Decision Criteria for the Payment of Technical Debt in Software Projects: A Systematic Mapping Study.	2016	38	2010-2014
Becker, Chitchyan, Betz, McCord	Trade-off decisions across time in Technical Debt Management: a systematic literature review	2018	240	2012-2016
Li, Avgeriou, Liang	A systematic mapping study on technical debt and its management	2015	94	1992-2013
Fernández-Sánchez, Garbajosa, Yagüe	A framework to aid in decision making for Technical Debt Management	2015	51	2011-2015
Besker, Martini, Bosch	Managing architectural technical debt: A unified model and systematic literature review	2018	42	2005-2016
BenIdris, Ammar, Dzielski	Investigate, identify and estimate the technical debt: A systematic mapping study	2018	43	2014-2017

Appendix B: The pool of papers analyzed in the literature review

ID	Author	Title	Year
P1	Besker & Martini	Embracing Technical Debt, from a Startup Company Perspective	2018
P2	Besker, Martini & Bosch	Impact of Architectural Technical Debt on Daily Software Development Work — A Survey of Software Practitioners	2017
P3	Besker, Martini & Bosch	Technical debt cripples software developer productivity: a longitudinal study on developers' daily software development work	2018
P4	Besker, Martini &Bosch	The Pricey Bill of Technical Debt: When and by Whom will it be Paid?	2017
P5	Bowlds, Fossaceca & Iammartino	Software obsolescence risk assessment approach using multicriteria decision-making	2018
P6	Cai, Xiao, Kazman, Mo, Feng,	Design Rule Spaces: A New Model for Representing and Analyzing Software Architecture	2018
P7	Cha, Dong & Vogel-Heuser	Preventing Technical Debt For Automated Production System Maintenance Using Systematic Change Effort Estimation With Considering Contingent Cost	2018
P8	Chicote	Startups and Technical Debt: managing technical debt with visual thinking	2017
P9	Dong, Ocker & Vogel-Heuser	Technical Debt as indicator for weaknesses in engineering of automated production systems	2019
P10	Fontana, Chatzigeorgiou, Trumler, Izurieta, Avgeriou & Nord	Technical Debt in Agile Development: Report on the Ninth Workshop on Managing Technical Debt (MTD 2017)	2017
P11	Ghanbari, Besker, Martini, & Bosch	Looking for peace of mind?: manage your (technical) debt: an exploratory field study	2017
P12	Huang, Shihab, Xia, Lo & Li	Identifying self-admitted technical debt in open source projects using text mining	2018
P13	Izurieta, Ozkaya, Seaman & Snipes	Technical Debt: A Research Roadmap Report on the Eighth Workshop on Managing Technical Debt (MTD 2016)	2017

P14	Jesus & Melo	Technical Debt and the Software Project Characteristics. A Repository-Based Exploratory Analysis	2017
P15	Klotins, Unterkalmsteiner, Chatzipetrou, Gorschek, Prikladnicki & Tripathi	Exploration of technical debt in start-ups	2018
P16	Kosti, Ampatzoglou, Chatzigeorgiou, Pallas, Stamelos & Angelis	Technical Debt Principal Assessment Through Structural Metrics	2017
P17	Liu, Huang, Xia, Shibab, Lo & Li	SATD detector: a text-mining-based self-admitted technical debt detection tool	2018
P18	Magnusson, Juiz, Gómez & Bermejo	Governing technology debt: beyond technical debt	2018
P19	Maldonado, Shibab & Tsantalis	Using Natural Language Processing to Automatically Detect Self-Admitted Technical Debt	2017
P20	Martini	Anacondebt: a tool to assess and track technical debt	2018
P21	Martini, Fontana, Biaggi & Roveda	Identifying and Prioritizing Architectural Debt Through Architectural Smells: A Case Study in a Large Software Company	2018
P22	Martini, Vajda, Vasa, Jones, Abdelrazek & Grundy	Technical debt interest assessment: from issues to project	2017
P23	Mensah, Keung, Svajlenko, Bennin & Mi	On the value of a prioritization scheme for resolving Self-admitted technical debt	2018
P24	Mera-Gómez, Ramírez, Bahsoon & Buyya	A Debt-Aware Learning Approach for Resource Adaptations in Cloud Elasticity Management	2017
P25	Mera-Gómez, Ramírez, Bahsoon & Buyya	A Multi-Agent Elasticity Management Based on Multi-Tenant Debt Exchanges	2018
P26	Rojas, Izurieta & Griffith	Toward Technical Debt Aware Software Modeling	2017
P27	Sierra	Is Self-Admitted Technical Debt a Good Indicator of Architectural Divergences?	2019
P28	Sierra, Shihab, Yasutake & Kamei	A survey of self-admitted technical debt	2019
P29	Skourletopoulos, Mavromoustakis, Mastorakis, Sahalos, Batalla & Dobre	A game theoretic formulation of the technical debt management problem in cloud systems	2017
P30	Wattanakriengkrai, Maipradit, Hata, Choetkiertikul, Sunetnanta & Matsumoto	Identifying Design and Requirement Self-Admitted Technical Debt Using N-gram IDF	2018
P31	Yan, Xia, Shihab, Lo, Yin & Yang	Automating Change-level Self-admitted Technical Debt Determination	2018
P32	Hacks, Hofert, Salentin, Yeong, Lichter.	Towards the definition of enterprise architecture debts	2019
P33	Fagerholm, Becker, Chatzigeorgiou, Betz, Duboc, Penzenstadler, Mohanani Venters	Temporal Discounting in Software Engineering: A Replication Study	2019
P34	Reboucas De Almeida, Treude, Kulesza	Tracy: A Business-Driven Technical Debt Prioritization Framework	2019
P35	Reboucas De Almeida	Business-Driven Technical Debt Prioritization	2019
P36	Njima, Demeyer	Value-based technical debt management: An exploratory case study in start-ups and scale-ups	2019
P37	Rindell, Bernsmed, Jaatun	Managing security in software or: How I learned to stop worrying and manage the security technical debt	2019
P38	Rindell, Holvitie	Security risk assessment and management as technical debt	2019
P39	Becker, Fagerholm, Mohanani, Chatzigeorgiou	Temporal discounting in technical debt: How do software practitioners discount the future?	2019

P40	Brenner R.	Balancing resources and load: Eleven nontechnical phenomena that contribute to formation or persistence of technical debt	2019
P41	Perez, Correal, Astudillo	A proposed model-driven approach to manage architectural technical debt life cycle	2019
P42	Pavlič, Hliš	The technical debt management tools comparison	2019
P43	Aragão, Andrade, Santos, Castro, Lelli, Darin	TestDCat: Catalog of Test Debt Subtypes and Management Activities	2019
P44	Amanatidis, Mittas, Chatzigeorgiou, Ampatzoglou, Angelis	The Developer's Dilemma: Factors affecting the Decision to Repay Code Debt	2018
P45	Besker, Martini, Bosch	Technical Debt Triage in Backlog Management	2019
P46	Palomba, Tamburri, Fontana, Oliveto, Zaidman, Serebrenik,	Beyond Technical Aspects: How Do Community Smells Influence the Intensity of Code Smells?	2018
P47	Li, Liang, Avgeriou	Architectural Technical Debt Identification Based on Architecture Decisions and Change Scenarios	2018
P48	Dai, Kruchten	Detecting Technical Debt through Issue Trackers	2017
P49	Rios, Spínola, Mendonça Seaman	Supporting Analysis of Technical Debt Causes and Effects with Cross-Company Probabilistic Cause-Effect Diagrams	2019

References

1. Cordella, A., Iannacci, F.: Information systems in the public sector: The e-Government enactment framework. J. Strateg. Inf. Syst. 19, 52–66 (2010).
2. Cunningham, W.: The WyCash portfolio management system. ACM SIGPLAN OOPS Messenger. 4, 29–30 (1992).
3. Rios, N., Mendonça Neto, M.G. de, Spínola, R.O.: A tertiary study on technical debt: Types, management strategies, research trends, and base information for practitioners. Inf. Softw. Technol. 102, 117–145 (2018).
4. Griffith, I., Taffahi, H., Izurieta, C., Claudio, D.: A simulation study of practical methods for technical debt management in agile software development. In: Proceedings - Winter Simulation Conference (2015).
5. Alves, N.S.R., Mendes, T.S., De Mendonça, M.G., Spínola, R.O., Shull, F., Seaman, C.: Identification and management of technical debt: A systematic mapping study. Inf. Softw. Technol. 70, 100–121 (2016).
6. Niehaves, B.: Innovation Processes in the Public Sector – New Vistas for an Interdisciplinary Perspective on E-Government Research? Electronic Government. EGOV 2007. (2007).
7. Scholl, H.J.: Digital Government : Looking Back and Ahead on a Fascinating Domain of Research and Practice. Digit. Gov. Res. Pract. 1, 1–12 (2020).
8. Dečman, M., Jukić, T.: Editorial for EJEG Volume 15 Issue 1. 15, 1 (2017).
9. Danish Ministry of Finance: Regeringens kasseeftersyn på it-området. Copenhagen, Denmark (2017).
10. The Swedish National Audit: Föråldrade it-system – Hinder för en effektiv digitalisering, Stockholm (2019).
11. Holvitie, J., Licorish, S.A., Martini, A., Leppänen, V.: Co-existence of the "technical debt" and "software legacy" concepts. CEUR Workshop Proc. 1771, 80–83 (2016).
12. Holvitie, J., Licorish, S.A., Spínola, R.O., Hyrynsalmi, S., MacDonell, S.G., Mendes, T.S., Buchan, J., Leppänen, V.: Technical debt and agile software development practices and processes: An industry practitioner survey. Inf. Softw. Technol. (2018).
13. Ampatzoglou, A., Ampatzoglou, A., Chatzigeorgiou, A., Avgeriou, P.: The financial aspect of managing technical debt : A systematic literature review. Inf. Softw. Technol. 64, 52–73 (2015).

14. Li, Z., Avgeriou, P., Liang, P.: A systematic mapping study on technical debt and its management. J. Syst. Softw. 101, 193–220 (2015).
15. Jeffery, M., Leliveld, I.: Best practices in IT portfolio management. MIT Sloan Manag. Rev. (2004).
16. Magnusson, J., Juiz, C., Gómez, B., Bermejo, B.: Governing technology debt. Proc. 2018 Int. Conf. Tech. Debt - TechDebt '18. 76–84 (2018).
17. Schlichter, B.R., Kraemmergaard, P.: A comprehensive literature review of the ERP research field over a decade. vol. 23 No. 4, pp. 486-520 (2010).
18. Webster, J., Watson, R.T.: Analyzing the past fo prepare for the future: writing a literature review. MIS Q. Vol. 26, No. 2, pp. xiii-xxiii (2002).
19. Boell, S.K., Cecez-Kecmanovic, D.: A hermeneutic approach for conducting literature reviews and literature searches. Commun. Assoc. Inf. Syst. Vol 34. (2014).
20. Harzing, A.W.: Publish or Perish. https://harzing.com/resources/publish-or-perish (2007).
21. Becker, C., Chitchyan, R., Betz, S., McCord, C.: Trade-off decisions across time in technical debt management. In: Proceedings of the 2018 International Conference on Technical Debt - TechDebt '18, pp. 85-94. (2018).
22. BenIdris, M., Ammar, H., Dzielski, D.: Investigate, Identify and Estimate the Technical Debt: A Systematic Mapping Study. Int. J. Softw. Eng. Appl. 9, pp. 01–14 (2018).
23. King, N.: Doing Template Analysis. In: Symon, G. and Cassell, C. (eds.) Qualitative Organizational Research: Core Methods and Current Challenges. SAGE (2012).
24. Blaikie, N., Priest, J.: Designing Social Research: The logic of anticipation. Cambridge (2019).
25. Gregor, S.: The nature of theory in Information Systems. MIS Q. Manag. Inf. Syst. vol. 30, pp. 611–642 (2006).
26. Whetten, D.: What Constitutes a Theoretical Contribution. Acad. Manag. Rev. vol. 14, pp. 490–495 (1989).
27. Derman, E.: Metaphors, Models & Theories. Q. J. Financ. vol. 1, pp. 109–126 (2011).
28. Blomberg, J., Burrell, M.: An Ethnographic Approach to Design. In: Jacko, J.A. (ed.) Human–Computer Interaction Handbook: Fundamentals, Evolving Technologies, and Emerging Applications. Boca Raton, FL (2012).
29. Sutton, R.I., Staw, B.M.: What theory is not. Adm. Sci. Q. vol. 40, pp. 371–384 (1995).
30. Fernandez-Sanchez, C., Garbajosa, J., Yague, A.: A framework to aid in decision making for technical debt management. 2015 IEEE 7th Int. Work. Manag. Tech. Debt, MTD 2015 - Proc. pp. 69–76 (2015).
31. Polit, D.F., Beck, C.T.: Generalization in quantitative and qualitative research: Myths and strategies. Int. J. Nurs. Stud. vol. 47, pp. 1451–1458 (2010).
32. Danish Competition and Consumer Authority: Hvor lang løbetid må en rammeaftale have? https://www.kfst.dk/faq/udbud/nyt-udbud/fase-1/hvor-lang-loebetid-maa-en-rammeaftale-have/ last accessed 2020/05/19
33. Bannister, F., Connolly, R.: The great theory hunt: Does e-government really have a problem? Gov. Inf. Q. vol. 32, pp. 1–11 (2015).
34. Mathiassen, L.: Designing Engaged Scholarship: From Real-World Problems to Research Publications. Engag. Manag. Rev. vol. 1, iss. 1, Article 2. (2017).