applied system innovation

MDPI

Article

Problematizing the Adoption of Formal Methods in the 4IR–5IR Transition

John Andrew van der Poll

Digital Transformation and Innovation, Graduate School of Business Leadership (SBL), Midrand, University of South Africa (Unisa), P.O. Box 392, Pretoria 0003, South Africa; vdpolja@unisa.ac.za;
Tel.: +27-11-652-0316/+27-84-580-4008

Abstract: The adoption of formal methods (FMs) as a software development methodology remains low. Advocates of FMs point to the advantages to be gained by producing highly dependable systems, while critics refer to the steep learning curve required to master the underlying mathematics and logic. The situation was similar for artificial intelligence (AI), but the advent of 4IR–5IR technologies has recently made AI a feasible technology for computing. We believe that the same could hold for FMs. In this article, we considered both the advantages and disadvantages of the use of FMs and unpacked them by problematizing the aspects that need to be considered in the 4IR–5IR worlds to facilitate the use of FMs as a viable software development methodology. We made the case that the 5IR embedding of harmonious collaboration between humans and machines could assist with difficult FM interfaces, similar to how human–computer interaction (HCI) has influenced technical and inflexible systems in the past. Since we view FMs as a technology, we further considered the role to be played by technology adoption, exemplified by the various technology adoption models, e.g., the TOE framework. This article culminates in the formulation of a problematization framework for the adoption of FMs in 4IR–5IR.

Keywords: Fifth Industrial Revolution (5IR); formal methods; formal specification; Fourth Industrial Revolution (4IR); problematization; Quality 4.0; sustainable development goals; technology adoption; Z

check for updates

Citation: van der Poll, J.A. Problematizing the Adoption of Formal Methods in the 4IR–5IR Transition. *Appl. Syst. Innov.* **2022**, *5*, 127. https://doi.org/10.3390/asi5060127

Academic Editor: Felix J. Garcia Clemente

Received: 29 October 2022
Accepted: 28 November 2022
Published: 17 December 2022

1. Introduction

The past couple of decades have witnessed significant advancements within the software engineering community. The industry has observed advances from rudimentary machine code programs to assembly language systems, with somewhat of a plateau being reached for a period with the development of third-generation languages (3GLs); however, this was eventually followed by the dawn of fourth-generation languages (4GLs), or domain-specific languages [1], in which programming became more of a specification exercise. In sync with the progress in programming languages, we saw conceptual development paradigms from traditional procedural languages (3GLs) to object-oriented techniques, in which the emphasis moved from programming code as the primary concern to data as the most important resource in a company's software system, leading to modern data science.

With the realization that data are a vital company resource, loose-standing file systems became databases and the modern data warehouses in which historical data are stored to allow for business intelligence decisions [2]. It should be noted that programming code regained prominence with developments such as agent-oriented programming, in which software agents (programs) perform routine tasks such as archiving and backups [3]. As systems became more complex, these were organized into enterprise resource planning (ERP) systems [4], and intelligent environments emerged as products of the Fourth and Fifth Industrial Revolutions, the latter following close on the heels of the former.

Despite all the aforementioned advances, software challenges remained. Software development projects continued to run over budget, were finished late, or were never completed. A prominent source for these challenges are the regular CHAOS reports [5]. Later, CHAOS reports indeed evidenced an improvement in software development for smaller projects [6] through the use of Agile, but Agile itself is often criticized due to its lack of a rigorous set of upfront user requirements and its possible inability to reach an acceptable project maturity level [7]. Nevertheless, larger software projects continue to face challenges [8].

While it was hoped that the above software development advances would bring about the elusive silver bullet, it did not fully materialize. One such development, specifically related to the development of safety-critical or mission-critical software, wherein human lives may be at stake, was the use of formal methods (FMs) based on techniques of discrete mathematics and logic [9]. Formal methods aim to produce provably correct software or at least highly dependable software, i.e., software of a high quality. Indeed, formal methods projects have had their fair share of success [10] over the years, but the steep learning curves required to get to grips with the underlying mathematical formalisms discouraged many companies from adopting FMs as a serious software development methodology [11]. The value of applying an FMs approach to software development for industrial critical systems, specifically railway applications, avionics, and the automotive industry, together with the value of using the correct software tools for FMs, was discussed by the authors in [12]. With respect to the commercial world, however, the use of FMs remains limited [13].

A comprehensive survey of 130 FMs experts worldwide, in which responses were obtained for 30 questions using a Likert scale instrument [14], confirmed the industry adoption challenges of FMs. The survey produced interesting results, including that: FMs may be viewed as a disruptive technology; they are often sold to industries while still being immature; in the opinion of the respondents, FMs are applicable to any company, irrespective of its size; managers ought to become more aware of the value of FMs; training in FMs is much needed; 4IR-based AI is in need of FMs to assist with making intelligent decisions in the new industrial environment; and policies around the use of FMs should be put in place. In regard to policies, the work by Mario Gleirscher et al. [15] called for an FMs manifesto embodying ten principles related to scope, methodology, integration, explainability, automation, scalability, transfer, usefulness, ease of use (usability—user experience (UX)), and evaluation. Work on developing a manifesto and calling for policies to create awareness in communities of the importance of software research and innovation is also found in [16]. Research conducted in the Netherlands involving a manifesto to create awareness regarding energy-efficient software, the long-term maintainability of software, and software sustainability is presented in [17]. Adopting an FMs approach to software development may facilitate the long-term maintainability of software, thereby promoting sustainability.

The Fourth Industrial Revolution (4IR) led to the enhancement of traditional structures, as well as the revival of technologies that could be considered ahead of their time. For example, artificial intelligence (AI) climbed the Gartner Hype Cycle [18] in the 1970s and 1980s, but then industry lost interest, since the technology of the time was inadequate to realize the benefits of AI. The advent of the 4IR, however, sparked renewed interest in AI as a feasible technology. Concurrently, traditional structures were enhanced to reap the benefits and opportunities of the 4IR. Examples of these are the Airport 4.0 [19] and Quality 4.0 [20] frameworks. Quality 4.0 forms an integral part of this article and is discussed and analyzed in Section 4. The stance taken in this article is that, in the same way that AI was reinvigorated by the advent of the 4IR, the 4IR may well facilitate the use of FMs in the future. In this regard, preliminary ideas on the factors involved in assisting FMs through 4IR technologies are presented in [21].

However, the 4IR mostly concerns the technical aspects of ICTs, notably, AI, cloud computing, the IoT, and many others [22,23]. One may argue that technical skill sets devoid of soft skills do not maximize chances for the successful adoption of a technology. In this regard, we recall the rise of the human–computer interaction (HCI) in response to inflexible computer systems of their time [24]. Following this argument, the 5IR, which concerns harmonious collaboration [25], could be needed to further facilitate the adoption of FMs as a serious software development methodology. Consequently, this article enhances preliminary ideas put forward by the researcher for facilitating the technical use of FMs through the 4IR [21]. Amongst others, the present work enhances this in line with 5IR considerations.

The layout of the article is as follows: Following the introduction, the research questions we aimed to answer are stated in Section 1.1. Our research methodology underlying this work is presented in Section 2, followed by a literature review in Section 3. Aspects addressed include FMs and 4IR and 5IR considerations; technology adoption; and aspects on problematization. The first contribution of this work appears in Section 4 in which we introduce and indicate ambiguities in Quality 4.0, thereby stating a case for the use of FMs, and subsequently enhancing it with 5IR components. A second contribution is utilizing 4IR technologies enhanced with 5IR considerations to develop a problematization framework for the adoption of FMs in these new industries.

1.1. Research Questions (RQs)

Our introduction gives rise to the following RQs:

- To what extent do 4IR structures, for example, Quality 4.0, lend themselves to mathematical formalism? (RQ1)
- How can traditional 4IR structures be enhanced by 5IR aspects? (RQ2)

Our objective was to:

- Develop a problematization framework aimed at facilitating the adoption of FMs through 4IR–5IR technologies.

2. Materials and Methods

The research in this article follows the methodology suggested in Saunders et al.'s Research Onion [26] in Figure 1.

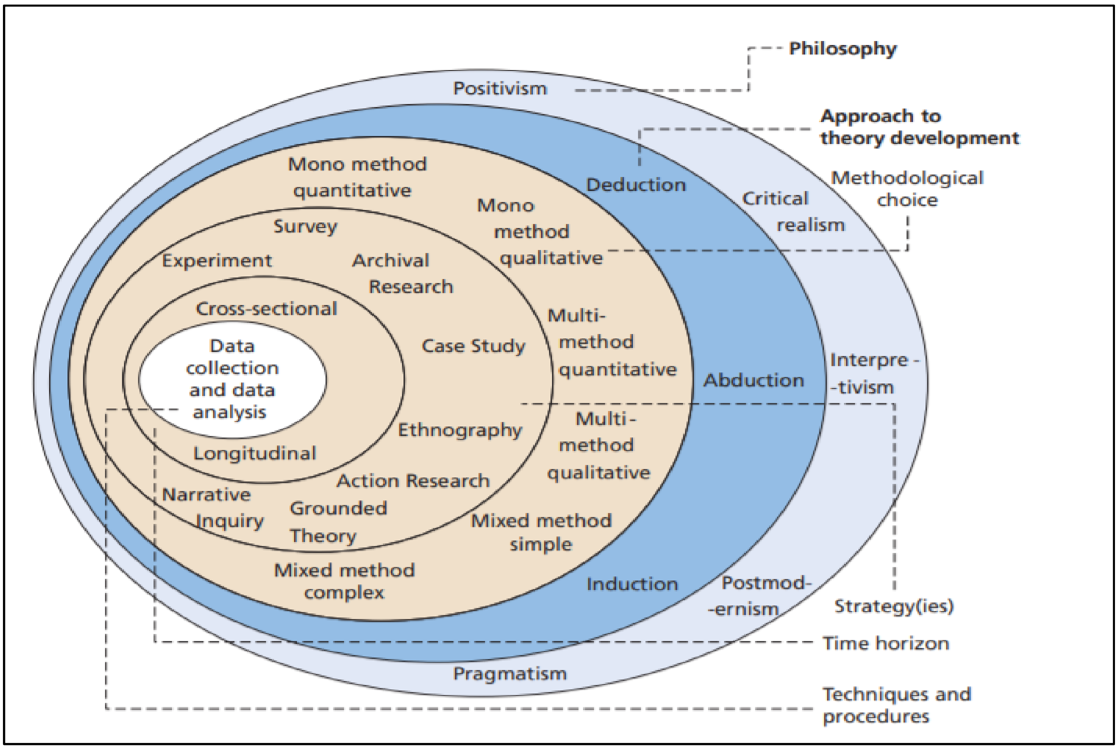

Figure 1. Research Onion. Reprinted with permission from Mark Saunders, Philip Lewis, and Adrian Thornhill ©2018 [26].

Following the onion depicted in Figure 1 from the outer layer, this work followed a mix of positivism and interpretivism as the philosophical stances. It was positivist since we were working with discrete mathematics and logic, which are precise sciences. The work was further interpretive since we also interpreted 4IR and 5IR diagrams. Moving to the next layer—approach to theory development—the work was, again, a mix; it was firstly inductive since a problematization framework was developed, followed by a qualitative validation in Section 5.

At the next layer in from the outside, our methodological choice would be a lightweight mix in that FMs exhibit a pseudo-quantitative character, while the analyses of text and diagrams are often used to describe the 4IR–5IR aspects and add a qualitative choice to the research. The strategy at this point was a survey, but of literature and not yet a survey among humans. Our time horizon was cross sectional since the research was performed at a specific point in time. The techniques and procedures were conceptual at this stage; hence, data collection was via the literature.

3. Literature Review

Following on from the introduction to this article, in this section we addressed the role of FMs as a serious option for correct software development; aspects around the 4IR and moving into the 5IR; technology adoption through viewing all of the FMs, 4IR, and 5IR as sets of technologies; and considerations of problematization, leading to our problematization framework for the adoption of FMs in the 4IR–5IR arena. Part of the Quality 4.0 framework was enhanced for 5IR requirements. We indicated ambiguities in 4IR–5IR frameworks, thereby making a case for the use of FMs.

3.1. Formal Methods in Software Engineering

Conceptually, formality in the science of computing refers to techniques in which mathematics is used for theory and software development. In a more traditional sense, these refer to the study of automata—a system which moves from state to state, consumes a certain input, and produces a certain output [27]—among which the Turing machine is used as the model on which modern systems are based. The FMs we referred to in this article may be viewed as the use of discrete mathematics and logic to model the behavior of software systems. In such a paradigm, system development starts with the construction of a formal specification [28]. Such a specification is then reasoned about to establish its properties by stating and discharging proof obligations (POs). These proof obligations aim to show that the specification has certain desirable properties, for example, that it meets the intention of its natural or semi-formal specification, and that undesirable properties are absent [29], among others.

Formal specification paradigms are often grouped into those that are procedural-like (with traditional pre- and post-conditions), process algebras (e.g., CSP), algebraic (declarative) specifications, and model-based specifications [28]. Each of these techniques has its own unique purposes, advantages, and disadvantages. The author of this article leans towards the model-based specification technique in which a system moves from state to state, owing to their operational nature. Arguably, this fits well with us humans as we move from one state to the next in life. We argue later in this article that the harmonious collaboration of the 5IR likewise fits the model-based specification style. Z [28] is one of the successful model-based specification languages discussed later in this article.

3.1.1. Advantages of Formal Methods

FMs, through the use of a formal specification, assist a specifier in writing a precise specification of a system, thereby eliminating possible ambiguities. Often these also involve investigating the consequences of a specification. Natural- and semi-formal specifications are known to be ambiguous [30], hence our interest in formal specifications.

The following example illustrates the value of formalizing natural language statements, i.e., the utility of FMs.

Example 1. *Consider the following statements in the security policy of a company [30]:*

(1) *All usernames and password combinations must be unique.*
(2) *Every user must have a password.*

From (1) we infer that two or more different users, e.g., U1 and U2, may have the same password, say P1, at any given point in time. Hence, (U1, P1) and (U2, P1) is a valid state of the system.

From (2) we may infer two possible formulations:

- (all x exists y (y is the password of x)) (a), or
- (exists y all x (y is the password of x)) (b).

Now, (b) $\rightarrow$ (a), but not vice versa. Logically speaking, statement (b) is too strong and is not what was intended—(b) states that there is one specific password for all users. This will not work since all users will know each other's password. Formulation (a) is what is intended by the policy statements, and it is only through formalizing the statements that the possible ambiguities emerge.

Further advantages to using FMs are illustrated in Section 4.1.

3.1.2. Formal Specification Techniques

The Z specification language emerged as a popular model-based specification notation. It is based on a strongly typed fragment of Zermelo–Fraenkel set theory [31] and first-order logic. The basic construct in Z is the schema which is a box-shaped structure containing declarations of variables, called components in Z, followed by predicates that further constrain the variables [32]. Therefore, it is reminiscent of a programming language program with the exception that statements in a program are mostly procedural while the logic statements in Z are declarative in nature. That said, we note that the Z syntax was augmented over time to include an *if* ... *then* ... *else* ... construct, reminiscent of procedural constructs in, e.g., third-generation programming languages. Examples of Z schemas are given below. As indicated, one of the advantages of formally specifying a system is that the specifier can reason about the properties of the resultant system. It can be shown that the specification inhibits certain desirable properties, and that undesirable consequences cannot be derived. Example 1 illustrated the value of formality even though actual formal reasoning was not fully conducted.

Formal methods by virtue of a formal specification are not without pitfalls, however, as is illustrated in the following section.

3.1.3. Challenges of Formal Methods

Consider the scenario in which customers order products from suppliers. Upon placing an order, a customer may pay a deposit, or the full price, or may defer payment to a later stage. Abstracting away from the details of defining the state space of the customer–supplier system or aspects of system initialization, the following schema specifies the scenario in which a customer places an order but does not make any payment.

```
_Place_Order____________________________
| Δ Order_System
| o?:Order
|_______________________________________
| o? ∉ orders
| orders' = orders ∪ {c?}
| payment' = payment
|_______________________________________
```

The system receives as input an order ($o?$) which is presently not in the system. The set of orders is updated, and no payment is in effect. In Z, an after state is indicated by a prime ($'$).

At a later stage, the customer is required to make a payment, as captured in schema *Pay_Order*.

Pay_Order

$\Delta\ Order\_System$

$o?: Order$

$amount?: \mathbb{R}$

$o? \in orders$

$payment' = payment \cup \{\ o? \mapsto amount?\}$

$orders' = orders$

The order ($o?$) has been placed before ($o? \in orders$), following schema *Place_Order*. Payment is specified accordingly, and the set of existing orders remains invariant.

Following Z's schema calculus [32], the two schemas may be combined to specify a robust operation:

$$Place\_and\_Pay\_Order \triangleq Place\_Order \wedge Pay\_Order$$

Schema *Place_and_Pay_Order* is given by:

Place_and_Pay_Order

$\Delta\ Order\_System$

$o?: Order$

$amount?: \mathbb{R}$

$o? \notin orders \wedge o? \in orders$

$orders' = (orders \cup \{\ c?\}) \wedge orders' = orders$

$payment' = payment \wedge (payment' = payment \cup \{\ o? \mapsto amount?\})$

The schema *Place_and_Pay_Order* is clearly inconsistent, illustrating that care has to be exercised in applying formal methods. Naturally, reasoning about the specification, by discharging proof obligations arising from the specification, would reveal such inconsistencies. Such an exercise could be manual (by hand), interactive, or automated [33].

We further note that the above system may, in real life, suffer from various database normalization anomalies [34], e.g., an order would not exist in isolation; instead, it would be linked to a customer. Nevertheless, our example is not intended to be an exercise in normalization.

3.2. The Fourth Industrial Revolution (4IR)

In the opinion of the researcher, precise definitions of what the 4IR is and entails remain amiss. That said, the 4IR has been described as following on from the third industrial revolution, which concerned digitalization—inventions in electronics followed by the development of computers, the internet, and further afield, nuclear energy [35]. The 4IR technologies, in turn, blur the lines between biological, physical, and cybernetic realms [36]. Other embedded 4IR technologies include artificial intelligence (AI), 3D printing, the (industrial) internet of things (IIoT), augmented reality, and many others [22,23]. Numerous aspects around 4IR cyber-physical systems (CPSs) came to the fore and high-level frameworks for these have been developed [37,38].

Arguably most important is that the 4IR projects result in a world in which humans and robots work as equals in the workplace. With humans and machines working together, it becomes vital for the algorithms driving these machines to be provably correct and devoid of flaws, especially if the machines are to perform safety-critical tasks in the presence of human colleagues. Clearly, this calls for the use of an FMs approach amongst other robotic software development. Considerations of self-awareness and self-maintenance, as captured by some CPSs, are also gaining prominence.

Humans and robots working together may also lead to challenges of co-existence and the ethics of machines, which is central to the questions arising as a result of the Fifth Industrial Revolution (5IR) which is discussed in the following subsection.

3.3. The Fifth Industrial Revolution (5IR)

Since the 5IR and the related Industry 5.0 are such recent concepts, there is little consensus on how these should be defined [39]. Similar to how the 4IR followed on from the third industrial revolution, the Fifth Industrial Revolution is following on from the 4IR. There may be some difference in opinion regarding who first introduced the idea of the 5IR and the related Industry 5.0. For example, the authors in [25] noted that press articles and websites, e.g., [40,41], already mentioned aspects relating to the 5IR, while the authors in [42] indicted that the related "Industry 5.0" term was coined by Michael Rada [43].

As mentioned earlier in this article, developments often follow a cyclical approach—technical developments followed by a cycle in which the hard-core developments are humanized to make these accessible to humans, e.g., the field of HCI which eventually followed decades of technical developments in computing [24]. In a similar vein, the 5IR may be viewed as the humanizing of the technical 4IR developments of AI, big data, the IIoT, and other 4IR technologies. One may reason that the 5IR is following very closely on the heels of a still underdeveloped 4IR in an attempt to make human sense of the complicated technologies that sprung up over the past decade or two. The time spans between new technological advances appearing are shortening, as is indicated in Figure 2 [25]:

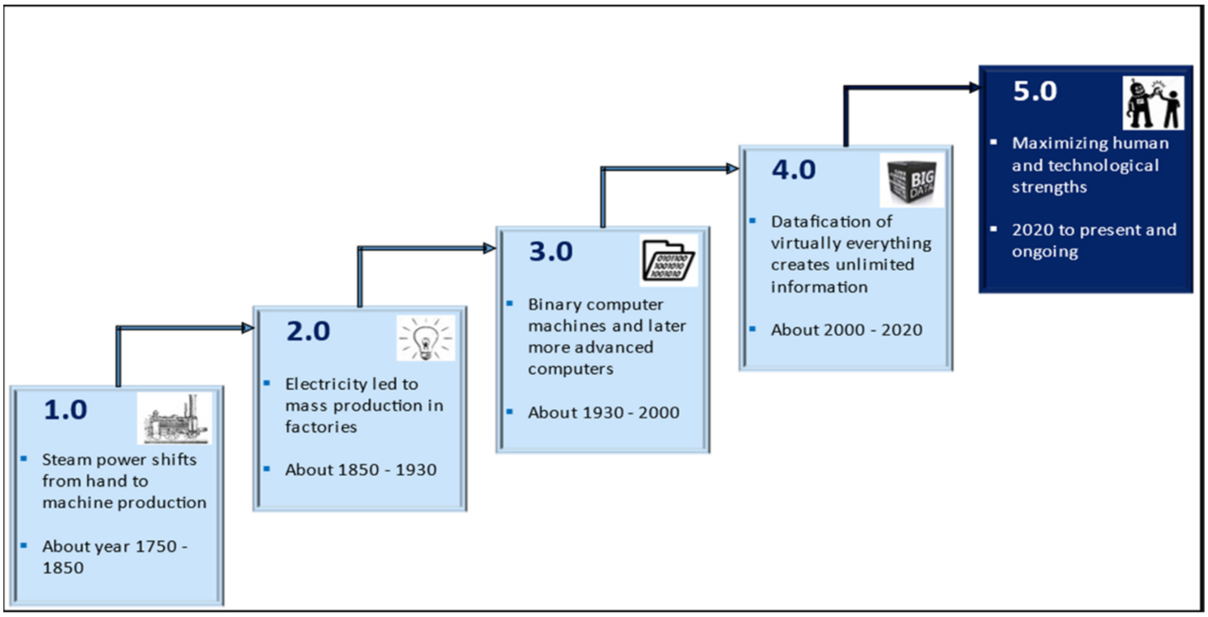

Figure 2. Progression of Industrial Revolutions (Reprinted with permission [25]. @2022, Noble et al.)

With reference to Figure 2, the first revolution lasted 100 years before transitioning to the 2IR, which lasted 80 years until the 3IR lasting 70 years, followed by the 4IR lasting

20 years, followed by the 5IR, which, by all indications, coincided with the advent of the recent COVID-19 pandemic.

The 5IR may be viewed from numerous angles but, in the view of the researcher, the 5IR very much concerns sustainability in line with the United Nations' Sustainable Development Goals (SDGs) [44], which center on equality, quality of human life, climate aspects, clean environments, clean energy, and so forth. In line with these, Mourtzis [45] indicates that two views of the 5IR are emerging.

The first view could be referred to as harmonious collaboration between humans and machines. In this world, humans and machines collaborate with each other as equals [45,46] and there is little or no competition between humans and machines, i.e., the collaboration is harmonious [25]. Humans would perform duties that require creativity and are more innovative in nature, while machines could do the rest, for example, the more mundane tasks—performing routine backups of data, or other repetitive jobs. Of course, that is not to say machines cannot be creative or innovative. In fact, one of the grander goals of AI as a 4IR technology is for machines to attain human intelligence [47]. Collaborative robots would typically observe humans to learn how to perform similar tasks. In this world, the human would (presumably) still be the entity to gain the most from the collaboration, i.e., harmonious collaboration would benefit humans more. Chief among harmonious collaboration would be the ethics of robots which would be built into the underlying algorithms. Still, there would be risks involved for a human working side by side with a machine. It would also involve considering (organizational) cultural issues. These were all built into our problematization framework in Section 4.3.

A second view of the 5IR could be that of developing a bioeconomy instead of a circular economy which often suffers from waste generation [45]. In a bioeconomy, sustainability of our environment takes center stage. Emphasis would be on waste reduction or the total prevention of waste through improved, innovative processes (it should be noted that these are also goals of the 4IR). The 5IR, therefore, promotes synergy with our environment in line with the global SDGs referred to earlier.

Both views of the 5IR aim for greater societal well-being and expanding the pool of stakeholders who stand to benefit from the said revolution [38]. Instead of concentrating on company profits for financial (technical) gain only, the emphasis should be on profits for society in general, thereby again enlarging the pool of stakeholders who may benefit.

As indicated, the 5IR also involves risk management and reduction; hence, the authors in [47] argue that, owing to the small number of humans who comprehend 4IR technologies, innovations in these areas are largely asymmetrical—developments take place among individuals as and when these become available. In contrast, the 5IR could promote symmetrical innovation in systems in which more (hopefully most) humans understand and could, therefore, benefit from a given industrial revolution. Systems that fail to complete a task should be "fail safe" or embed a "safe exit", among other things. Systems should be more forgiving towards non-experts, i.e., the 5IR should embed symmetrical innovation [47]. All of these point to a reduction in risks in the 5IR.

Therefore, the 5IR triangulates three aspects, namely, human centricness, sustainability, and resilience (fail safeness). These aspects and the links with the 4IR are presented in Figure 3 [38].

Figure 3 shows the ongoing evolution of the 5IR flowing from the 4IR, involving mostly technical aspects, and incorporating human collaboration and environmental care aimed at sustainability in line with the 17 SDGs [44].

3.4. Technology Adoption

Both FMs and the 5IR discussed above may be viewed as sets of technologies and may, therefore, be subject to various technology adoption considerations. The technologies involved in FMs include, among others, discrete mathematics and logic, together with a host of underlying software engineering considerations. For the 5IR, all of the 4IR technologies (AI, IIoT, etc.) are suitably augmented by the human–machine collaboration aspects discussed in Section 3.3. Consequently, in formulating a problematization framework for the adoption of FMs in the 5IR, aspects of technology adoption deserve due consideration from both FMs and 5IR perspectives.

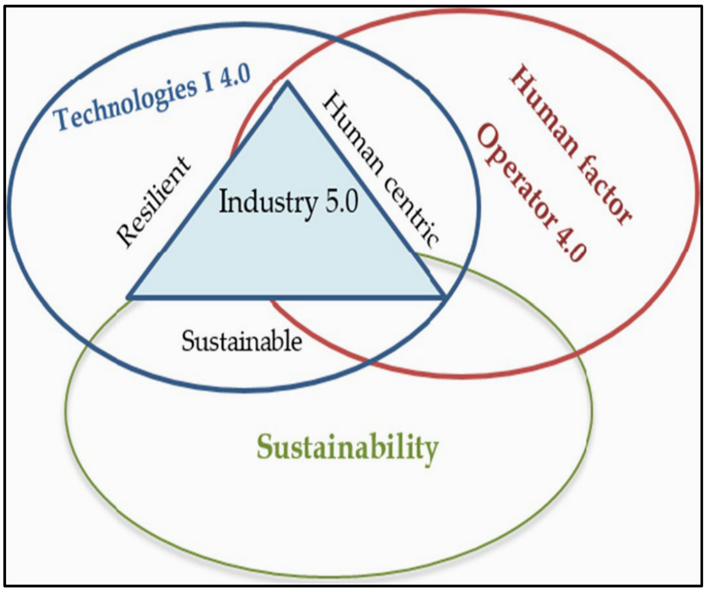

Figure 3. Triangulated 5IR interfaces (Reprinted with permission from [38] @2022, Grabowska et al.)

Various technology adoption frameworks and models have been defined over time, notably the Technology Acceptance Model (TAM) [48]; an extension of TAM, namely, the Unified Theory of Acceptance and Use of Technology (UTAUT) [49]; and many others. However, the Technology–Organization–Environment (TOE) framework originally defined by Tornatzky et al. [50] and depicted in Figure 4 appears to be most suitable for the task at hand. The TOE embeds aspects of technology, linking with technology aspects of FMs and the 4IR; organization, since FMs would inevitably be adopted in a company; and finally, environment, which fits the environmental and SDGs aspects of the 5IR. The suggestion for TOE is embedded in the problematization framework presented in Section 4.3.

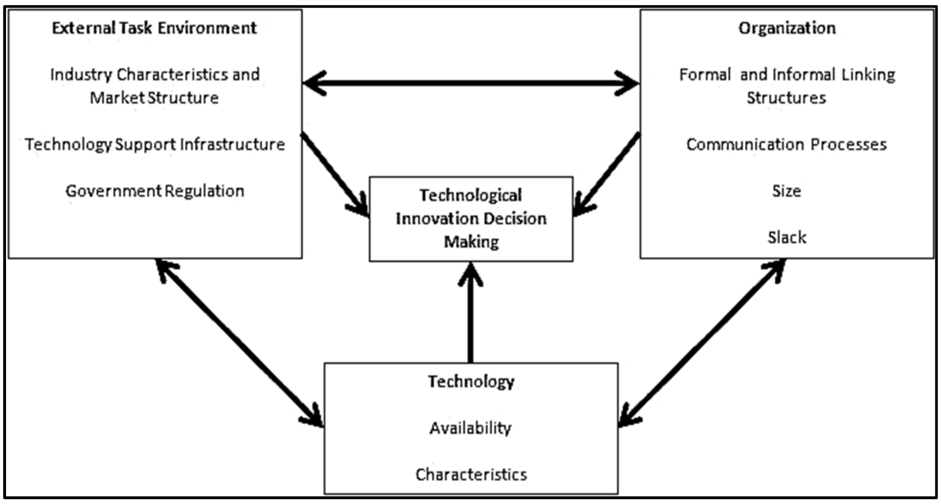

Figure 4. TOE framework (https://edutechwiki.unige.ch/en/Technology-organization-environment_framework/ accessed on 23 November 2022).

Finally, in this literature review, we considered aspects of problematization.

3.5. Problematization

The notion of problematization stems from the idea of questioning the status quo, i.e., the standard ways of viewing the world or the way things are done. Research into problematization stems from the work of Foucault and Kuhn, as discussed in [51,52]. In this context, problematization may be viewed as a disruptive technology similar to a disruptive innovation often observed throughout history [53]. Naturally, the 4IR may be viewed as a set of disruptive technologies that questions traditional ways of doing things, e.g., a machine is not viewed as just a tool anymore; it now becomes an equal colleague, as discussed in Section 3.3. In the same way, FMs' usage could be a disruptive technology, i.e., a departure from the traditional semi-formal software engineering design techniques [14,21].

Morgan [52] views problematization as being three-tiered from higher levels to lower levels. At the top level we have paradigms of thoughts, the second level comprises metaphors, while the lowest level comprises the puzzle-solving activities. The three-tiered structure is indicated in Figure 5.

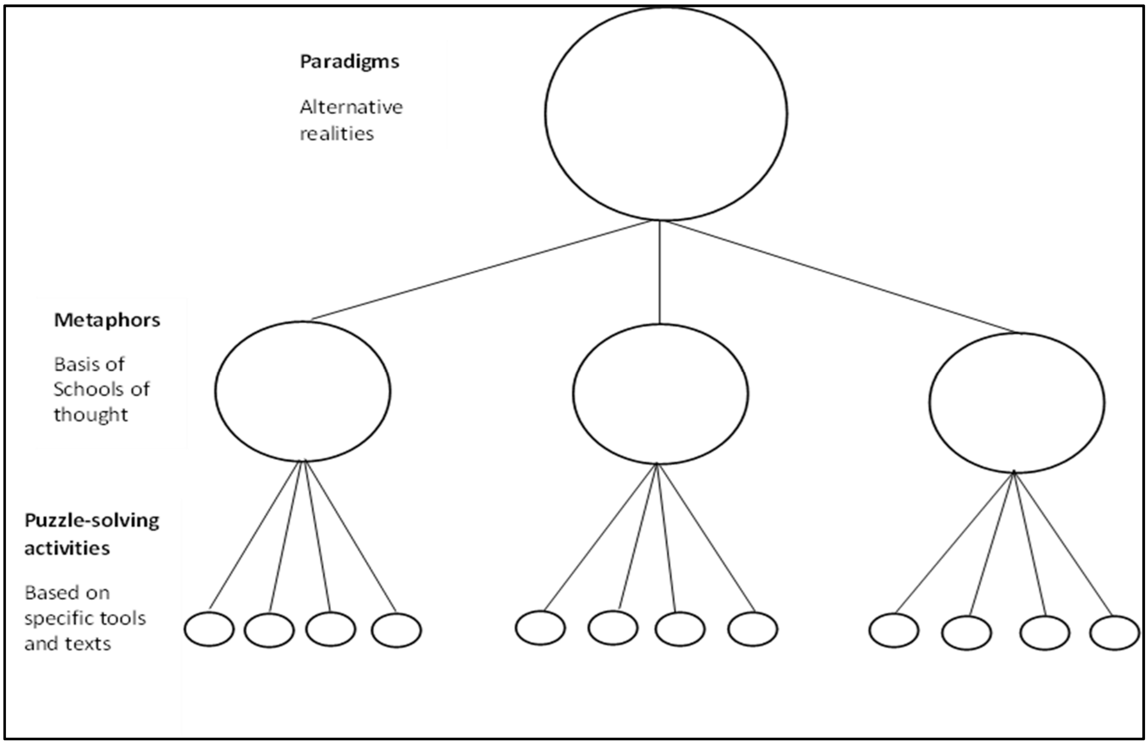

Figure 5. Morgan's (1980) three-tiered problematization [52] (Reprinted with permission from Sage Publishers.)

Morgan's [52] problematization structure fits well with our view of using the 4IR–5IR technologies to advance the use of FMs. As indicated, the top-level paradigms are viewed as alternative realities, much as FMs could be viewed as not only an alternative way to arrive at highly dependable software, but rather complementary to traditional approaches. The metaphors as bases of schools of thought could map onto being pro FMs or anti FMs. It is often remarked in certain circles that some "swear" by FMs, while others "swear" at them. Nevertheless, as indicated in Section 3.1, FMs have advantages in that they elicit design flaws or unclear thinking yet may also suffer from inconsistencies. However, in the latter case, discharging proof obligations arising from a formal specification would elucidate any inconsistencies [29,33]. At the lowest level of the Morgan structure in Figure 5, the puzzle-solving activities map onto a specifier's work in constructing a formal specification or discharging a proof obligation, either by hand, or using a software tool [33].

A useful view of problematization is described by Locke and Golden-Biddle in [54], in which they analyzed a large number of publications that appeared over the course of 20 years (1976–1996). Having analyzed the work, they concluded that publications often turn upon themselves, having made a case for the contribution of the said publication. This can be seen in our own work in Section 3.1.2, where we make the case for FMs, yet also point out challenges with their use. A further round then follows in which we call for the use of POs as a remedy.

Reference [54] classified the extant literature as being problematized into three kinds, namely, incompleteness, inadequacy, and incommensurability. The authors define previous research as incomplete if the extant literature is considered as not going far enough and the current research will enhance it. Research is considered inadequate if it does not sufficiently consider alternate viewpoints and the research making such claims would then consider additional perspectives. Incommensurability goes further than both incompleteness and inadequacy in the sense that extant research is viewed as simply wrong and needs to be replaced by alternative work. We incorporated all three kinds of problematizations in our framework and revisited these in the discussion of the framework in Section 5.

Reference [51] further considers problematization in terms of the nursing profession and presents the emergence of the Danish nursing education: A sick person was admitted to hospital and assigned to the care of an untrained nurse. Having gotten worse, the patient was eventually assigned to a trained nurse, resulting in improved health. The case was, therefore, made for formal nursing programs to be instituted. We believe FMs training in software development [13,14] can benefit in the same vein; hence, training is part of our framework. As indicated in the framework, FMs training involves numerous aspects.

Next, we considered one of the prominent 4IR frameworks, namely, Quality 4.0, and demonstrated how it inhibits ambiguity by attempting to formalize part of it. We also enhanced the said 4IR framework with the 5IR considerations observed in Section 3.3.

4. Results

A prominent open access 4IR framework [20], which is an extension of a third-generation framework, is Quality 4.0, depicted in Figure 6.

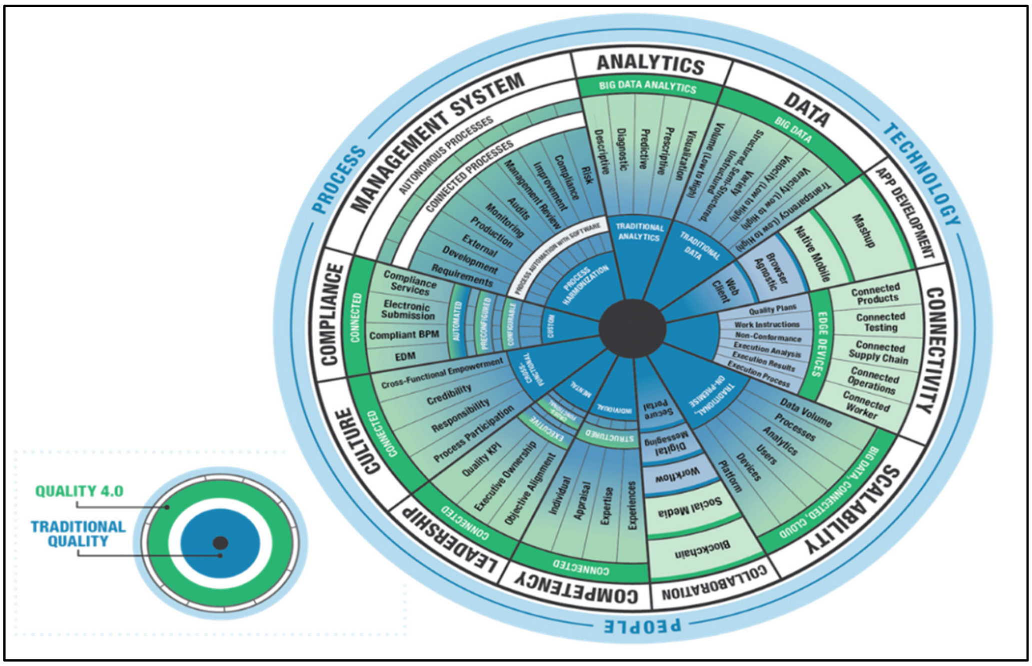

Figure 6. The Quality 4.0 framework (https://www.juran.com/blog/quality-4-0-the-future-of-quality/ accessed on 23 November 2022).

The framework embeds three dimensions, namely, PEOPLE, PROCESS, and TECHNOLOGY in its outer layer. Moving inwards, 11 pillars are defined—*Management System(s)*, *Analytics*, *Data*, *App Development*, *Connectivity*, *Scalability*, *Collaboration*, *Competency*, *Leadership*, *Culture*, and *Compliance*. The 11 pillars are further defined as indicated. Quality 4.0 enhances standard quality (as in the 3IR), indicated by the darker blue core.

4.1. Formalizing Quality 4.0

Quality 4.0, being a comprehensive semi-formal diagram, suffers from well-known challenges of semi-formal notations [55] illustrated next through an analysis of (say) the competency pillar.

Competency has, as its second layer, Connected, which is also the second layer in two or three other pillars, so it is unclear whether these embody the same connectedness or different connectedness, leading to our first ambiguity. Hence:

- *Competency.Connected* = set of *X* (for use below) *Ambiguity #1* (an assumption)

Next, there appears to be a one-to-one mapping between Competency and its connectedness, so we define (leading to our second ambiguity):

- *Competency* ≙ *Competency.Connected* ≙ *{X.Experience, X.Expertise, X.Appraisal, X.Indivudual}* *Ambiguity #2* (an assumption)

Following the Structured subcomponent deeper into the competency sector there are three (3) further subdivisions, but these have no denotations (names), leading to the next ambiguity:

- Undefined subdivisions below the Structured subcomponent *Ambiguity #3*

Since these are not indicated, the Quality 4.0 framework appears to be an abstract and organic (growing) structure. Plausibly, a different Quality 4.0 diagram can be instantiated for each application, leading to:

Ambiguity #4 (an assumption)

The above analysis of the Quality 4.0 framework illustrates some of the challenges associated with 4IR semi-formal notations, similar to what has been experienced in all the industrial revolutions to date. This observation, even in the 4IR, is a further call for the use of FMs involving discrete mathematics and logic [9]. It should be noted that a similar analysis for the Leadership pillar was conducted by the researcher in [21], also leading to a number of ambiguities.

Having subjected Quality 4.0 to a formal analysis, we note that it was probably not designed for such analyses. Rather, it is intended to be a high-level guide to be instantiated for each application.

The above formalization provided an answer to our first research question, RQ1.

4.2. Towards Quality 5.0

As indicated in Section 3.3, the 5IR is mostly about harmonious collaboration between humans and machines and establishing a sustainable environment through a bioeconomy instead of a circular economy. Consequently, the researcher opines that a first step towards a Quality 5.0 framework could be to suitably enhance the Collaboration pillar, i.e., the pillar adjacent to Competency, analyzed above.

A 5IR suggestion of Collaboration is given in Figure 7.

Harmonious collaboration in the form of harmony is added as a top layer in the Collaboration pillar. Together with harmony between humans and machines are considerations of ethics, trust, and risk, as identified in Section 3.3.

The enhancement of the Collaboration pillar with 5IR aspects provides an answer to our second research question, RQ2.

It should be clear that similar ambiguities, as elucidated for the Competency pillar, are present in the 5IR Collaboration pillar as well, again illustrating the value of FMs for software development. Consequently, while the 4IR–5IR technologies could facilitate the adoption of FMs in the software industry, FMs, in turn, are useful for analyzing 4IR–5IR structures.

Next, we turned our attention to the formulation of the problematization framework.

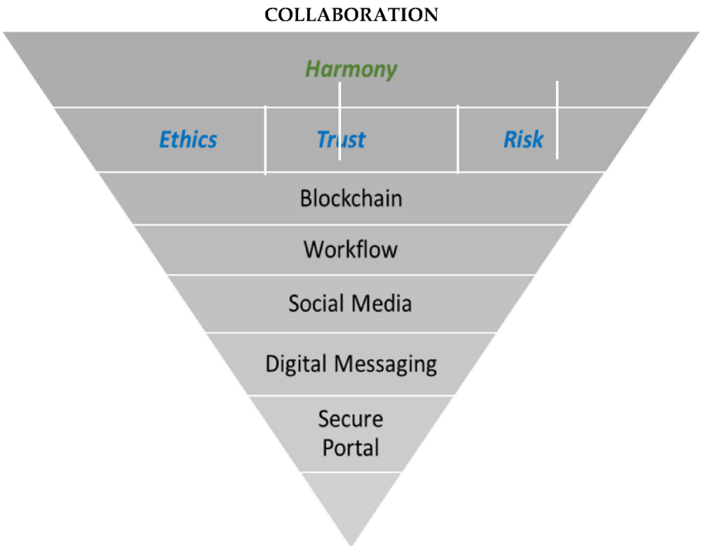

Figure 7. 5IR Enhancement of Collaboration in Quality 4.0 (synthesized by researcher).

4.3. Problematization Framework for the Adoption of FMs in the 4IR–5IR

On the strength of our developments up to this point, we formulated the problematization framework in Figure 8.

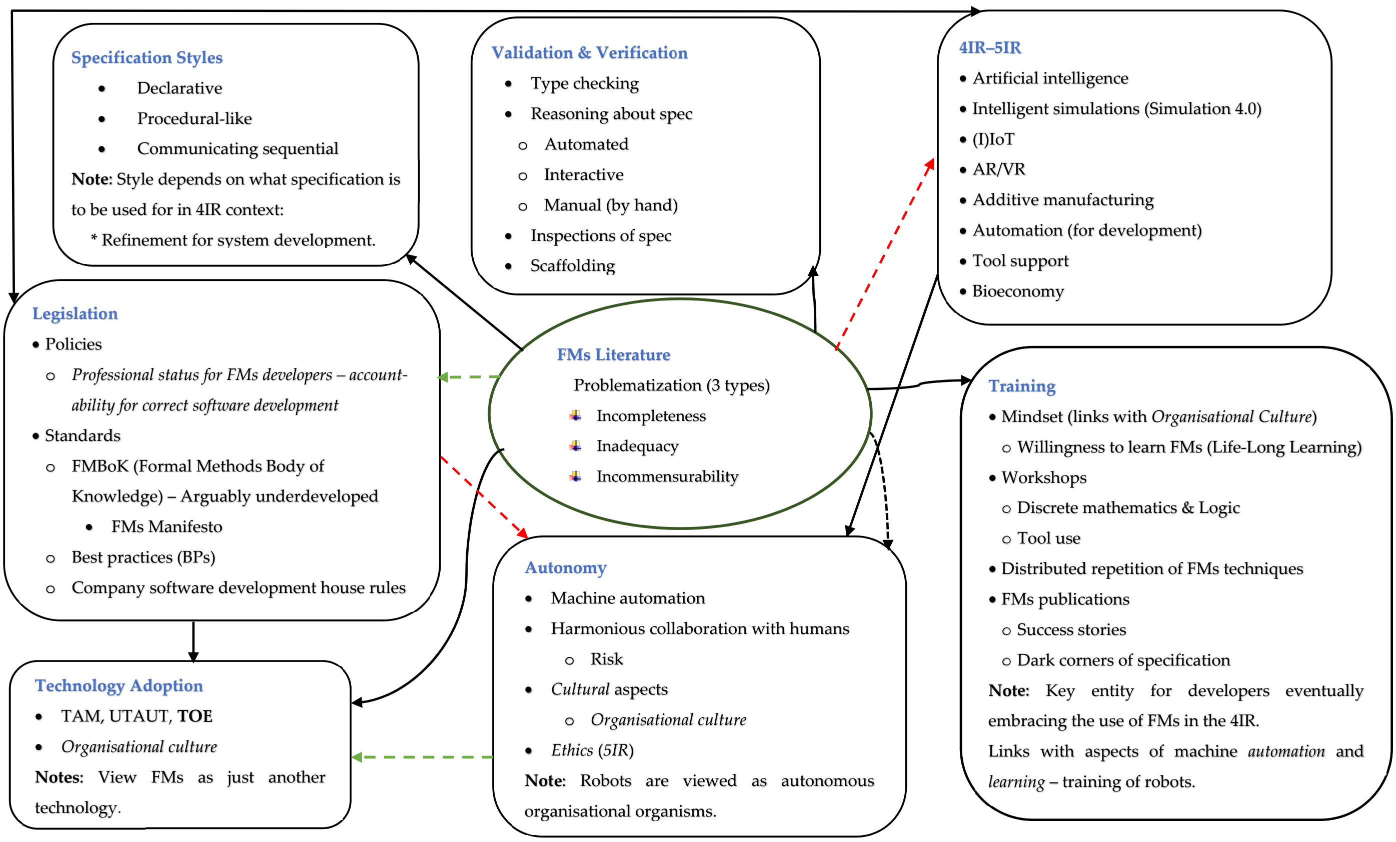

Figure 8. Problematization Framework for the Adoption of FMs in 4IR–5IR.

5. Discussion

Our problematization framework indicates the aspects that ought to be addressed to facilitate FMs becoming a serious contender for industrial software development. Central to these are problematization aspects, namely, incompleteness, inadequacy, and incommensurability. The author believes that FMs as a software methodology may not necessarily be *incomplete*. The necessary techniques and tools, as indicated in the training entity, are there, but ways to effectively use these should be further developed. In this regard, the 4IR–5IR entity addresses the technical aspects needed for these. Specific 5IR aspects addressed in this article, notably, risk, ethics, and harmonious collaboration, are indicated in the autonomy entity, thereby recognizing that robots may become first-rate citizens in the future and could work effectively alongside humans.

Even so, with problematization, the effective use of FMs may well be inadequate at this stage, hence the need for the framework and this article. As indicated, the researcher believes that the technologies of the 4IR and the human aspects in the 5IR will go a long way to improving the *adequacy* of the industrial use of FMs. The use of FMs is certainly not *incommensurable*, either in the literature or in the success stories indicated in the literature, e.g., [12,13]. We view FMs as a technology; hence, aspects of technology adoption come into play. Given the prominence of environmental and organizational aspects in the 5IR, we lean towards the use of the TOE for the adoption of FMs.

Standard aspects of validation and verification, and specification styles, are still a part of the mix in gaining the acceptance of FMs. The training captured in the framework is always an important consideration in acquiring a new skill, and it is no different for the usage of FMs.

The emergence of robots in the workplace and other aspects of the 5IR are bound to involve legislation in the future if they do not already. Similar to professions with professional status, software engineers working with safety-critical software in the presence of robots should use best practices in light of the evolving Formal Methods Body of Knowledge (FMBoK) supported by a manifesto. The said software engineers and their robot colleagues will be accountable for correct processes in the 4IR–5IR.

The black arrows in the framework present aspects that may be inferred from the literature, while the green dashed arrows indicate important, yet (arguably) non-critical, aspects that may not be explicit in the literature. The red dashed arrows indicate associations that the author of this article deems to be critical. These involve associations between 4IR–5IR, the FMs literature, legislation, and the autonomy of robots as first-rate citizens in the 5IR.

The problematization framework in Figure 8, in conjunction with the above discussion, meets our objective in Section 1.1.

6. Conclusions

In this article, aspects regarding the adoption of FMs as a software methodology were considered. The value proposition for the use of FMs was elucidated by showing that FMs make specifications more precise. On the negative side, FMs may inhibit inconsistencies, yet these could be addressed through the discharging of proof obligations arising from a specification. Type checking as a syntactic notion and inspections, as we have carried out, could further assist. Given all of these, we argued that intelligent 4IR technologies and human–machine aspects embedded in the 5IR may greatly facilitate the adoption of FMs as a viable option for software development. We further elicited ambiguities in 4IR frameworks, notably Quality 4.0, and augmented one of the pillars, namely Collaboration, with 5IR aspects. Given the sheer size of the 4IR–5IR domains, we proposed a problematization route to be followed, which considers various problematization aspects. The article culminated in the formulation of a problematization framework in which aspects for FMs adoption in the 4IR–5IR are presented.

Future work in this area may be pursued along a number of avenues. Our framework should be enhanced to recognize the important role to be played by the management of a company as well as governors with respect to government regulations. Leadership, as alluded to by the Quality 4.0 framework, should become part of the framework. The problematization framework indicates aspects, in fact challenges, that need to be further developed to achieve our goal. Therefore, it is by no means a solution framework. Such a solution framework should be developed, starting with an enhanced problematization framework. Once fully fleshed out, surveys among practitioners should be undertaken to determine its utility.

Funding: This research was funded by the University of South Africa through the Article Processing Charge as well as the research professor fund of the author.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Fowler, M. *Domain-Specific Languages*; Pearson Education: Upper Saddle River, NJ, USA, 2010.
2. Mbala, I.N.; van der Poll, J.A. Evaluation of Data Warehouse Systems by Models Comparison. In Proceedings of the 18th JOHANNESBURG Int'l Conference on Science, Engineering, Technology & Waste Management (SETWM-20), Johannesburg, South Africa, 16–17 November 2020; pp. 222–228, ISBN 978-93-86878-46-5. Available online: https://eares.org/siteadmin/upload/6879EAP1120285.pdf (accessed on 15 December 2022).
3. Rodriguez, S.; Gaud, N.; Galland, S. SARL: A general-purpose agent-oriented programming language. In Proceedings of the 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), Warsaw, Poland, 11–14 August 2014; IEEE: New York, NY, USA, 2014; Volume 3, pp. 103–110. [CrossRef]
4. Senaya, S.K.; van der Poll, J.A.; Schoeman, M. *Towards a Framework to Address Enterprise Resource Planning (ERP) Challenges*; Lecture Notes in Networks and Systems (LNNS); Springer: New York, NY, USA, 2022; Volume 236, pp. 57–71. ISBN 978-981-16-2380-6. [CrossRef]
5. Clancy, T. *The Chaos Report*; The Standish Group: Boston, MA, USA, 1995.
6. Eveleens, J.; Verhoef, C. The rise and fall of the chaos report figures. *IEEE Softw.* **2009**, *27*, 30–36. [CrossRef]
7. Corrigan, M.J.; van der Poll, J.A.; Mtsweni, E.S. The Project Management Information System as Enabler for ICT4D Achievement at Capability Maturity Level 2 and Above. In *Communications in Computer and Information Science (CCIS)*; Springer: New York, NY, USA, 2019; Volume 933, pp. 295–312.
8. Flyvbjerg, B.; Budzier, A.; Lee, J.S.; Keil, M.; Lunn, D.; Bester, D.W. The Empirical Reality of IT Project Cost Overruns: Discovering A Power-Law Distribution. *J. Manag. Inf. Syst.* **2022**, *39*, 607–639. [CrossRef]
9. Bjørner, D.; Havelund, K. Years of Formal Methods—Some Obstacles and Some Possibilities? In *Chapter 40 in FM 2014: Formal Methods*; Lecture Notes in Computer Science (LNCS); Springer International Publishing: Cham, Switzerland, 2014; Volume 8442, pp. 42–61.
10. Freitas, L.; Woodcock, J.; Zhang, Y. Verifying the CICS File Control API with Z/Eves: An experiment in the verified software repository. *Sci. Comput. Program.* **2009**, *74*, 197–218. [CrossRef]
11. Parnas, D.L. Really rethinking 'formal methods'. *Computer* **2010**, *43*, 28–34. [CrossRef]
12. ter Beek, M.H.; Larsen, K.G.; Ničković, D.; Willemse, T.A. Formal methods and tools for industrial critical systems. *Int. J. Softw. Tools Technol. Transf.* **2022**, *24*, 325–330. [CrossRef]
13. Nemathaga, A.; van der Poll, J.A. Adoption of Formal Methods in the Commercial World. In Proceedings of the Eight International Conference on Advances in Computing, Communication and Information Technology (CCIT 2019), Birmingham, UK, 23–24 April 2019; pp. 75–84. [CrossRef]
14. Garavel, H.; Maurice, H.; ter Beek, M.H.; van de Pol, J. The 2020 Expert Survey on Formal Methods. In *Formal Methods for Industrial Critical Systems*; Lecture Notes in Computer Science (LNCS), FMICS 2020; Ter Beek, M.H., Ničković, D., Eds.; Springer: Cham, Switzerland, 2020; Volume 12327, pp. 3–69. [CrossRef]
15. Gleirscher, M.; van de Pol, J.; Woodcock, J. *A Manifesto for Applicable Formal Methods*; Cornell University: Ithaca, NY, USA, 2021. [CrossRef]
16. Knols, S. Powering the Invisible Engine. *I/O Magazine*. March 2020, pp. 10–11. Available online: https://www.versen.nl/assets/manifesto/IO-magazine-Manifesto-VERSEN-2020.pdf (accessed on 21 November 2022).
17. VERSEN. Manifesto on Software Research and Education in The Netherlands. CWI, University of Twente. nd. Available online: https://www.versen.nl/assets/manifesto/manifesto_software_onderzoekers_06-2.pdf (accessed on 21 November 2022).
18. Dedehayir, O.; Steinert, M. The hype cycle model: A review and future directions. *Technol. Forecast. Soc. Chang.* **2016**, *108*, 28–41. [CrossRef]

19. Halpern, N.; Budd, T.; Suau-Sanchez, P.; Brathen, S.; Mwesiumo, D. Towards Airport 4.0: Airport digital maturity and transformation. In Proceedings of the 23rd Air Transport Research Society World Conference, Amsterdam, The Netherlands, 2–5 July 2019.
20. Juran, J.M. Quality 4.0: The Future of Quality? Web Blog. Available online: https://www.juran.com/blog/quality-4-0-the-future-of-quality/ (accessed on 20 October 2022).
21. van der Poll, J.A. Towards a Problematization Framework of 4IR Formalisms: The Case of QUALITY 4.0. In *Proceedings of the International Conference on Intelligent Vision and Computing (ICIVC 2021), Proceedings in Adaptation, Learning and Optimization*; Sharma, H., Vyas, V.K., Pandey, R.K., Prasad, M., Eds.; Springer: Cham, Switzerland, 2022; Volume 15, pp. 212–226. Available online: https://link.springer.com/chapter/10.1007/978-3-030-97196-0_18 (accessed on 12 July 2022).
22. Bayode, A.; van der Poll, J.A.; Ramphal, R.R. 4th Industrial Revolution: Challenges and Opportunities in the South African Context. In Proceedings of the 17th International Conference on Science, Engineering, Technology & Waste Management (SETWM-19), Johannesburg, South Africa, 18–19 November 2019; pp. 174–180.
23. Yang, F.; Gu, S. Industry 4.0, a revolution that requires technology and national strategies. *Complex Intell. Syst.* **2021**, *7*, 1311–1325. [CrossRef]
24. Rodden, T.; Cheverst, K.; Davies, K.; Dix, A. Exploiting context in HCI design for mobile systems. In *Workshop on Human Computer Interaction with Mobile Devices*; University of Glasgow: Glasgow, Scotland, 21–22 May 1998.
25. Noble, S.M.; Mende, M.; Grewal, D.; Parasuraman, A. The Fifth Industrial Revolution: How harmonious human–machine collaboration is triggering a retail and service [r] evolution. *J. Retail.* **2022**, *98*, 199–208. [CrossRef]
26. Saunders, M.; Thornhill, A.; Lewis, P. *Research Methods for Business Students*, 8th ed.; Pearson: London, UK, 2018.
27. Woods, D.; Neary, T. The complexity of small universal Turing machines: A survey. *Theor. Comput. Sci.* **2009**, *410*, 443–450. [CrossRef]
28. Ibrahim, R.; Amin, A.A.B.; Saringat, M.Z. The Z Specification for Exam Scheduling System (ESS) thru Genetic Algorithm. In Proceedings of the 2020 International Conference on Computing and Information Technology (ICCIT-1441), Tabuk, Saudi Arabia, 9–10 September 2020; pp. 1–5. [CrossRef]
29. Larson, B.R.; Chalin, P.; Hatcliff, J. BLESS: Formal specification and verification of behaviors for embedded systems with software. In *NASA Formal Methods Symposium*; Springer: Berlin/Heidelberg, Germany, May 2013; pp. 276–290.
30. Buthelezi, M.P.; van der Poll, J.A.; Ochola, E.O. Ambiguity as a Barrier to Information Security Policy Compliance: A Content Analysis. In Proceedings of the 2016 International Conference on Computational Science and Computational Intelligence, Las Vegas, NV, USA, 15–17 December 2016; pp. 1360–1367.
31. Enderton, H.B. *Elements of Set Theory*; Academic Press: New York, NY, USA, 1977.
32. O'Regan, G. Z formal specification language. In *Concise Guide to Formal Methods*; Springer: Cham, Switzerland, 2017; pp. 155–171.
33. Ackermann, J.G.; van der Poll, J.A. Reasoning Heuristics for the Theorem-Proving Platform Rodin/Event-B. In Proceedings of the 2020 International Conference on Computational Science and Computational Intelligence (CSCI'20), Las Vegas, NV, USA, 16–18 December 2020; pp. 1800–1806. [CrossRef]
34. Kennedy, D. Database Design and the Reality of Normalisation. In Proceedings of the 13th Annual NACCQ Conference, July 2000; Available online: https://www.in-site.co.nz/misc_links/papers/kennedy167.pdf (accessed on 5 May 2012).
35. Troxler, P. Making the 3rd industrial revolution. In *Fab Labs: Of Machines, Makers and Inventors*; Transcript Publishers: Bielefeld, Germany, 2013.
36. KPMG. *The Factory of the Future—Industry 4.0: The Challenges of Tomorrow*; KPMG: London, UK, 2016.
37. Mokwana, D.R.; van der Poll, J.A. Towards a Framework for Fourth Industrial Revolution (4IR) Cyber Physical Systems (CPSs). In Proceedings of the Association for Global Business Advancement (AGBA 2022): An International conference, Istanbul, Turkey, 2–4 July 2022.
38. Grabowska, S.; Saniuk, S.; Gajdzik, B. Industry 5.0: Improving humanization and sustainability of Industry 4.0. *Scientometrics* **2022**, *127*, 3117–3144. [CrossRef]
39. Madsen, D.Ø.; Berg, T. An Exploratory Bibliometric Analysis of the Birth and Emergence of Industry 5.0. *Appl. Syst. Innov.* **2021**, *4*, 87. [CrossRef]
40. Aryu Networks. What Will the 5th Industrial Revolution Look Like? *AryuNetworks.com*. 21 April 2020. Available online: https://aryunetworks.com/what-will-the-5th-industrial-revolution-look-like/ (accessed on 28 October 2022).
41. Gauri, P.; Van Eerden, J. What the Fifth Industrial Revolution Is and Why It Matters. *Europeansting.com*. 16 May 2019. Available online: https://europeansting.com/2019/05/16/what-the-fifth-industrial-revolution-is-and-why-it-matters/ (accessed on 28 October 2022).
42. Akundi, A.; Euresti, D.; Luna, S.; Ankobiah, W.; Lopes, A.; Edinbarough, I. State of Industry 5.0—Analysis and Identification of Current Research Trends. *Appl. Syst. Innov.* **2022**, *5*, 27. [CrossRef]
43. Rada, M. Industry 5.0—From Virtual to Physical. *LinkedIn*. 7 March 2018. Available online: https://www.linkedin.com/pulse/industry-50-from-virtual-physical-michael-rada (accessed on 3 February 2022).
44. Lu, Y.; Nakicenovic, N.; Visbeck, M.; Stevance, A.-S. Policy: Five priorities for the UN Sustainable Development Goals. *Nature* **2015**, *520*, 432–433. [CrossRef] [PubMed]
45. Mourtzis, D. Towards the 5th industrial revolution: A literature review and a framework for process optimization based on big data analytics and semantics. *J. Mach. Eng.* **2021**, *21*, 5–39. [CrossRef]

46. Van der Poll, J.A. Can I Trust My Android? Graduate School of Business Leadership, Unisa. Editorial: Jive Media Africa. 2020. Available online: https://www.unisa.ac.za/sites/sbl/default/News-&-events/Articles/Can-I-trust-my-Android (accessed on 28 October 2022).
47. Fanoro, M.; Božanić, M.; Sinha, S. A Review of 4IR/5IR Enabling Technologies and Their Linkage to Manufacturing Supply Chain. *Technologies* **2021**, *9*, 77. [CrossRef]
48. Hassler, E.; MacDonald, P.; Cazier, J.; Wilkes, J. The Sting of Adoption: The Technology Acceptance Model (TAM) with Actual Usage in a Hazardous Environment. *J. Inf. Syst. Appl. Res.* **2022**, *14*, 12–20.
49. Chang, A. UTAUT and UTAUT 2: A Review and Agenda for Future Research. *J. Win.* **2012**, *13*, 106–114. [CrossRef]
50. Malope, E.T.; van der Poll, J.A.; Ncube, O. Digitalisation Practices in South-African State-Owned Enterprises: A Framework for Rapid Adoption of Digital Solutions. In Proceedings of the 54th Hawaii International Conference on System Sciences (HICSS-54), Kauai Island, HI, USA, 5–8 January 2021; pp. 4590–4599. Available online: https://hdl.handle.net/10125/71174 (accessed on 28 October 2022).
51. Frederiksen, K.; Lomborg, K.; Beedholm, K. Foucault's notion of problematization: A methodological discussion of the application of Foucault's later work to nursing research. *Nurs. Inq.* **2015**, *22*, 202–209. [CrossRef]
52. Morgan, G. Paradigms, metaphors, and puzzle solving in organization theory. *Adm. Sci. Q.* **1980**, *25*, 605–622. [CrossRef]
53. Dzimba, E.; van der Poll, J.A. Disruptive Innovation at the Base-of-the-Pyramid: Negotiating the Missing Links. *J. Open Innov. Technol. Mark. Complex.* **2022**, *8*, 171. [CrossRef]
54. Locke, K.; Golden-Biddle, K. Constructing opportunities for contribution: Structuring intertextual coherence and "problematizing" in organizational studies. *Acad. Manag. J.* **1997**, *40*, 1023–1062. [CrossRef]
55. Egyed, A.; Wile, D. Support for managing design-time decisions. Software Engineering. *IEEE Trans. Softw. Eng.* **2006**, *32*, 299–314. [CrossRef]