Terminology for Evolving Design Artifacts

Hannes Schlieter, Jeannette Stark, Martin Burwitz, and Richard Braun

Technische Universität Dresden, Chair of Wirtschaftsinformatik, esp. Systems Development, Dresden, Germany
{hannes.schlieter,jeannette.stark,martin.burwitz,richard.braun}@tu-dresden.de

Abstract. Many design researchers evolve artifacts in succeeding projects. Yet, these researchers lack a terminology to describe how their artifacts evolve. We provide such a terminology by paralleling concepts from evolution with design artifacts using examples from conceptual modeling. We found seven concepts from evolution that we think are useful to describe evolving design artifacts. Evaluating whether these concepts have been addressed, we identified six conceptual modeling design studies, whose authors have addressed some of the concepts with their own words. Using two of these studies, we explain how terminology from evolution can be used to describe evolving design artifacts. We hope that our results are useful to be integrated in design science procedure models to help researchers increasing rigor and relevance of their research, e.g. by allowing to clarify how the artifact at hand has evolved or to describe the evolutionary distance to preceding artifacts.

Keywords: Design artifacts, Evolution, Coevolution, Design Science, Conceptual Modeling.

1 Introduction

Researchers often evolve their modeling artifacts in succeeding design projects. Thereby, artifacts sometimes evolve differently. For example, the Unified Modeling Language (UML) specifications were revised to fix shortcomings and support concept integration (cf. [1]) leading to a successive UML versions. In this type of evolution, successive UML versions evolve within the context of general-purpose modeling and consider e.g. changed user needs. An example for another type of evolution comprises parts of the UML being transferred to the field of systems engineering, leading to SysML [2]. In this case evolution happens within a specific domain so that design researchers need to address the specific requirements of the new domain. Yet another type of evolution may be distinguished for versions of UML (e.g. UML 2.0 and 2.5) and corresponding revisions of standards-compliant UML modeling tools (e.g. [3]) that were evolved based on revisions of the UML. In this type of evolution, evolving the modeling tool depends on how the corresponding modeling language evolves and is thus, dependent on the evolution of another artifact.

14th International Conference on Wirtschaftsinformatik,
February 24-27, 2019, Siegen, Germany

Prior design research has discussed evolution in the context of mutability [4–6]. Mutability is a core component of design research and researchers are asked to address mutability when publishing their results [7]. For mutability, two paradigmatic perspectives are distinguished: in-design and in-use [4, 5]. In-design mutability refers to the degree of artifact change encompassed by the design theory [7] and allows for designing mutable artifacts that are adaptable to varying organizational contexts [4] and situations [5]. In-use mutability refers to artifacts that evolve over time with users adapting the artifact to new situations, e.g. through configuration [5] or tailoring [8]. Gregor and Jones state that mutability goes even further by "allowing for a certain amount of adaptation or evolution", [7], p. 326. However, how artifacts evolve and adapt is so far not specified in in-use and in-design mutability, so that design researchers lack a specification to describe artifact evolution and hence, cannot address mutability of their design artifacts for this particular aspect.

Specifying how artifacts evolve may help design researchers to increase rigor and relevance for their design projects. Rigor in design science depends on how researchers ground their project with existing artifacts [9]. Being able to specify how artifacts evolve instead of just claiming that it evolves may help identifying important requirements for the evolution at hand. Relevance in design science depends on how researchers address opportunities and problems with their artifacts in the application domain [9, 10]. Specifying how their artifact evolves from preceding artifacts may help to address relevance more precisely. For example, differentiating between different types of evolution allows for discussing the evolutionary distance more precisely e.g. by explaining what features have been added or removed, whether new domain has been entered or an artifact has now coevolved with another artifact. Beyond helping researchers to enhance rigor and relevance, specifying how artifacts evolve contribute to discuss evolutionary lineages on different levels such as single artifacts or artifacts of a domain such as conceptual modeling design artifacts. Investigating evolutionary lineages for single artifacts can also help to discuss and compare maturity of different artifacts and to reveal future potential for evolution. Investigating evolutionary lineages of different domains contribute also to the discussion of how mature domains, such as conceptual modeling, are in contrast to other IS domains. Furthermore, discussing evolution in the context of mutability may help to further advance mutability research.

In this paper we aim to investigate concepts to specify evolving design artifacts. Although we believe that specifying design artifacts is important for all IS domains that develop design artifacts we focus on conceptual modeling design research in this paper. Conceptual modeling is a major topic of the Wirtschaftsinformatik domain, which is primarily design-oriented [11]. Furthermore, this domain was chosen as the authors are familiar with it so that concepts can be transferred to this domain more easily and better examples can be given than for other domains. Finally, to allow for comparing evolutionary lineages between domains, other domains will follow in future research. This paper represents an interim struggle in theorizing [12] that needs to be built on to provide for more rigor and relevance in design science projects, to develop and compare evolutionary lineages and to enhance mutability research.

Hence, these research goals still need to be addressed in succeeding research. Yet, this paper also contributes a specific outcome and concepts discussed in this paper can already be used as a terminology to describe evolution more specific than it was possible before so that evolution as one aspect of mutability can already be specified more precisely as demanded by [7]. For investigating concepts to describe evolution and deriving the terminology of evolving design artifacts, we use argumentative reasoning between evolution of design artifacts and evolution in biology. For a first proof of relevance, we employ a content analysis according to Krippendorff [13]. Using this method, we investigate whether concepts identified to describe evolution have already been applied in recent conceptual modeling design science research.

2 Theoretical Background

2.1 Mechanisms that Drive Evolution

Evolution is described as a change in the heritable characteristics of populations over successive generations [14] and is pushed forward through diverse mechanisms, including mutation, migration, natural selection and gene drift [15]. **Mutations** alter the genome of organisms and result from random errors during DNA replication [16]. For example, a mutation can cause a change in a beetle's genotype from genes for grey to white coloration (Fig. 1a). Those beetles that gain genes for white coloration can have offspring with a gene for white coloration, too. Accordingly, genes for white coloration can be more frequent in future populations. Mutations may result in a gain of a new feature or a loss of an ancestral feature. For example, the bacterium Sphingobium has evolved a metabolic pathway as a new feature to degrade toxins [17] whereas birds have lost their wing function when there was no need to fly. In analogy to evolution in biology, we refer to mutation of design artifacts as a gain of a new function or withdrawal of an ancestral function. Yet, there are also differences within these two contexts. While in biology mutations occur randomly and might have a positive, neutral or negative impact on survival and reproduction capabilities [16], mutations in design science can typically be planned and can hence, rather be classified as non-random. As design researchers plan mutations also the impact is more likely to be positive than in biology. Nonetheless, using the analogy in conceptual modeling design science may help to specify a gain or a loss of an ancestral feature. Yet, the implication of impacting randomly versus non-randomly needs further investigation for design artifacts and should therefore not be made. As an example, for conceptual modeling design artifacts, we refer to the Business Process Modeling and Notation (BPMN) that has experienced major and minor revisions that have continuously produced a variety of BPMN features. For example, the major revision that lead to version 2.0, not least introduced the choreography diagram as a mean to model interactions of multiple communication partners. An additional feature of this mutation comprises the extension mechanism that enables modelers to systematically extend the BPMN core by custom elements.

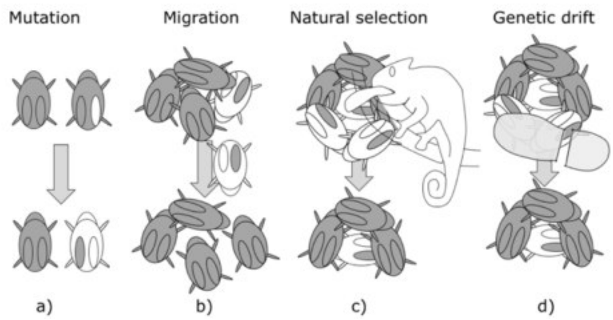

Figure 1. Evolutionary mechanisms: mutation and migration to induce variation as well as natural selection (non-random) and genetic drift (random) for transferring variation

While mutations lead to genetic variation within a population, **migration** allows transferring this variation to further populations of a specie (Fig. 1b). Imagine, beetles with genes for white coloration migrate into another population and thus, allow for variety in a second population. Please note that a specie usually includes several populations. A population is defined as a group of organisms, in which reproduction takes place [18]. In contrast, organisms of a specie only potentially interbreed as they live in distinct areas and thus, may form separate populations whose organisms do usually not interbreed. Migration allows referring to transferring design artifacts to another context and thus, give way for an own evolutionary lineage. For example, BPMN has initially addressed the management of business processes, but has already been used in a variety of other domains, that even sometimes lead to new BPMN species by using the extension mechanism (e.g. [19–22]).

Mutation and migration allow for variation within populations and create the basis for two further mechanisms: natural selection and gene drift. **Natural selection** acts on variation by non-random sorting [23]. Imagine, white beetles live in a habitat with green plants so that predators can easily spot them. In this case, they are not likely to survive and reproduce, which lead genes for white coloration being reduced non-randomly for future populations (Fig. 1c). Also, **gene drift** acts on variation. Yet, while natural selection acts non-randomly, gene drift acts randomly [24]. For example, a beetle population with white and grey coloration is threatened by a storm, in which most of the white beetle die resulting in a random drift towards grey beetles (Fig. 1d). Genetic drift may cause two populations starting with the same genotype drifting apart into divergent populations or even species [25].

2.2 Outcome of Evolution

Evolutionary mechanisms lead to diverse evolutionary outcomes such as speciation [26], adaptation [27, 28], coevolution [29], and extinction [30]. **Speciation** refers to populations of one specie diverging into two or more species and is made responsible for diversity of life [31]. After having diverged, organisms of the two species might still produce common offspring that are in most cases infertile such as mules resulting from horses reproducing with donkeys [32]. A specie is used as a concept to differentiate organisms that cannot reproduce even if manipulated (e.g. moved to another area) while population is used to differentiate organisms of a specie that actually do not mate, e.g. as they live in different areas or times. In design science, **species** may be used in analogy to **artifact types** as this concept helps indicating that design artifacts can potentially be used to produce offspring that address similar requirements or allow to solve similar tasks. In this sense, succeeding artifacts do not represent distinct artifact types but rather succeeding artifact instances of the same type. Furthermore, **populations** may be used in analogy to **artifact variants** to indicate that instances of artifact types can be migrated into another context. For example, the core BPMN 2.0 can be considered a BPMN artifact type, which meets requirements for modeling business processes. Using the core BPMN 2.0 for software process modeling [20] led to a new artifact variant that was able to meet the same requirements in a new context. Based on this artifact variant, the revision and extension of the core BPMN 2.0 regarding new context-related requirements evolved a new BPMNt artifact type. While **speciation** leads to a new specie, **extinction** means that an entire specie disappears. In biology, speciation and distinction happen regularly [30]. When using speciation and extinction to describe design artifacts, we might rather refer to **entry** and **exit** in analogy to the product life cycle [33].

Adaptation refers to organisms evolving adaptive traits by mutation and natural selection and characterizes a population's movement towards a fit to the environment [34]. Adaptations mostly occur through mutations, which impact positively, negatively or neutrally on the organism's survival and reproduction capabilities. Based on how mutations impact, they may prevail or be sorted out. Adapting artifact instances by introducing a new function usually occurs in succeeding **artifact versions**. For example, the introduction of the choreography diagram as a new feature in version BPMN 2.0 can be considered an adaptation to the rising demand for modeling and managing communication between business partners (cf. [35]).

Coevolution occurs across pairs of species that interact with each other [36]. These interactions can lead to adaptations of one specie to cause adaptations in another specie and vice versa. Accordingly, a cycle of adaptations emerges that is referred to as coevolution [29]. Based on how species interact, coevolution is distinguished into conflict and cooperation. Conflict usually emerges between predators and prey. Predators can coevolve to catch their prey more effectively. As a result, selective pressure increases for their prey and leads to coevolving the prey, e.g. to escape more effectively. Adaptive traits developed in conflict comprise speed of movement, ability to detect and habitat choice [37]. Cooperation e.g. occurs between insects and flowers that reward insects with nectar but rely on them for reproduction. Insects and flowers have evolved interactions such as communication via scent or attraction via color [38]. Coevolution has already been applied to organizational adaptation [39], strategic management [40], and strategic alliances [41]. It has also been used for IS by discussing mutual adaptations of business and IS strategies and for coevolutionary choices of platform-based ecosystem owners [42] and the ecosystem's environment [43]. Using coevolution as analogy for evolving design artifacts, we consider the BPMN an artifact type in coevolution with process execution languages (e.g. XPDL or BPEL) that represent business processes in context of machine-based workflow execution. These artifact types and the BPMN artifact type complement one another in terms of language concepts, semantics and interchange formats [63].

Coevolution in biology may lead to **endosymbiosis** that references to cases where one partner is integrated into the other partner's body [44]. This concept has been shaped by Mereschkowsky [45] and has been used as example that evolution may lead to more complexity. Endosymbiosis can be used as analogy to describe evolving design artifacts that incorporate versions of other artifact types. Referring the example of BPMN, the *Lane* concept may be classified as endosymbiotic integration of swim lanes from other modeling approaches (cf. [46, 47]). Please note a distinction between endosymbiosis and coevolution: In coevolution an artifact type adapts with another artifact type. In case of endosymbiosis only certain versions of an artifact type may be incorporated while other versions of that artifact type may remain independent.

3 Concepts and Relationships for Evolving Design Artifacts

We summarize concepts and relationships that connect these concepts in Table 1. Furthermore, we show how these concepts and relationships interrelate in Fig. 2. Examples are taken from conceptual modeling languages. This provides a larger scope of consideration in comparison to the specific BPMN examples in Section 2. As depicted in Fig. 2, we parallel species with artifact types, in analogy to an instance-type relationship, to reflect the idea behind its instances. The artifact instances, which in biology reflect organisms, can further be paralleled with artifact versions indicating that versions evolve, e.g. during iterations [48], [9]. Furthermore, populations are paralleled with artifact variants as this concept allows for summarizing versions that are designed for different application domains and that can develop their own evolutionary lineage. Among relationships, adaptation refers to gaining new or loosing ancestral functions that lead to a better fit to the application domain. Accordingly, adaptations reflect iteration cycles in design science.

Migration relates artifact versions to variants indicating that a particular version is used for another application domain. A further relationship, that we found useful to describe evolving design artifacts is coevolution. Coevolution can happen between versions of one artifact type and those versions of another artifact type and is therefore used as relationship between artifact types. In contrast, endosymbiosis may be used with the term incorporation for design artifacts, meaning that artifact versions of one artifact type may be incorporated for artifact versions of another artifact type.

Table 1. Notions to describe evolving design artifacts and examples from conceptual modeling

Concepts from biology	***Use for IS design artifacts***	***Example***
Specie: Potentially interbreeding organisms	Artifact type: Summarizing artifacts with similar goals and user requirements	Enterprise Modeling Languages
Population: Organisms of a specie that actually mate, e.g. as they live in the same habitat	Artifact variant: Summarizing artifact versions that are used for the same application domains	Enterprise Architecture Languages, Business Process Modeling Languages, Software Modeling Languages
Organism: Instance of a specie	Artifact version: Instance of an artifact type	BPMN 1.0/1.1/1.2/2.0 [49], UML 1.0-1.5/2.0/2.5 [1]
Relationship from biology		
Adaptation: Developing or losing traits as well as non-random sorting towards a fit to the habitat	Adaptation: developing or losing functions and sorting these functions, e.g. towards user acceptance	Revisions of UML specifications in order to fix shortcomings and support concept integration, for instance (cf. [1])
Migration: Transferring variation (e.g. mutations) in between populations	Migration: e.g. using an artifact version within another context	Parts of UML are transferred to the field of systems engineering, leading to SysML [2]. Some parts are completely reused (e.g., Use Case Diagrams), while other parts are adapted domain-specifically (e.g. Block Definition Diagrams [50])
Coevolution: Evolution of a species causes adaptations in a second specie and vice versa. Is distinguished in conflict and cooperation.	Coevolving in cooperation: Artifact types that evolve together (modeling grammar and its tool)	Versions of the UML (e.g. UML 2.0 and 2.5) and corresponding revisions of standards-compliant UML modeling tools (e.g. [3])
Endosymbiosis: Incorporation of an organism within another species' organisms	Incorporation: Artifact versions of an artifact type being incorporated in versions of another artifact type	Use Case Diagrams were integrated in UML (cf. [51]). Parts of the UML were refined in SysML (e.g. Ports [2])

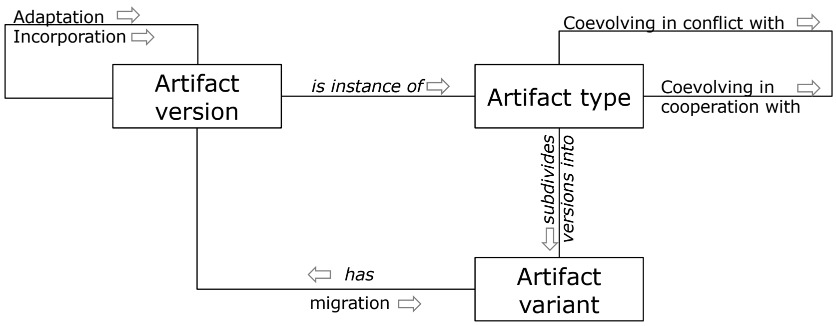

Figure 2. Relating concepts and relationships derived from evolutionary biology (cursive relationships are only integrated for a better readability)

4 Evaluation

Although this works focusses on discussing concepts and relationships that might be used for further theorizing in future work, we include a pilot content analysis following Krippendorff [13] to investigate whether these concepts and relationships are relevant to describe evolving design artifacts. We believe that "when theories are particularly interesting or important, there should be greater leeway in terms of empirical support" [52], p. 383, allowing for just a small demonstration to show that concepts and relationships of this study may be of relevance. Of course, subsequent research is required to assess whether concepts and relationship will hold up [52]. In this pilot study we focus on recent conceptual modeling design science research and limit the analysis to articles published in the Business & Information Systems Engineering (BISE)-journal during the last five years. This journal represents the flagship of the Wirtschaftsinformatik domain, which is primarily design-oriented [11].

We found six conceptual modeling design studies that we have summarized in Table 2. In these studies, we could detect all relationships identified in Section 2 and 3, including adaptation, migration, incorporation and coevolution. Hence, we could provide a first proof that identified concepts and relationships are relevant to describe evolving conceptual modeling design artifacts. Furthermore, we identified the words the authors used to refer to these concepts. We found that adaptation is primarily referred to with "extend" while migration is rather referred to "extent" plus a specification such as "extends by combining" [53], p. 65 or "extends with the required aspects" [54], p. 318. In contrast, incorporation was referred to in different ways using words such as "building on" [55], p. 252, and "the techniques were sequentially arranged to form" [55], p. 258. Likewise, coevolution was referred to in different ways such as with "corresponding software prototype" [55], p. 251. Although we found words how authors describe the different types of evolution that we have presented in Table 1, we refrain from suggesting words to address these types of evolution. For discussing how authors may refer to these concepts we expect that more empirical research is required.

Providing an example how to use the terminology, we use the study from Johannsen and Fill [55] in Fig. 3a and Gailly et al. [59] in Fig. 3b. Johannsen and Fill have developed an artifact type that represents a roadmap for Business Process Improvement (BPI) with conceptual modeling support. Whereas in their first version they describe the method [61], they add conceptual modeling support in succeeding studies [55], [62]. The authors describe these artifact versions as first contribution (BPI-roadmap) and second contribution (modeling support) [55], p.252 and develop these contributions in separate papers. Johannsen and Fill further incorporate versions of artifact types such as SIPOC and Ishikawa, which they further describe as foundations of their artifact type [55], p.253f. Furthermore, we found their artifact type in coevolution with the ADOxx platform [63]. This platform has been developed by members of the current research group so that implications of the evolution of one artifact may be implemented for the other artifact.

Table 2. Studies that evolve conceptual modeling design artifacts

Ref.	*Artifact type*	*Types of evolution*	*Coding examples*
[56]	Business Activities for modelling Processes and process-rel. RBAC models	**Adaptation** of Business Activities UML 2.0 extension [57] for process-related delegation models as a new artifact version	"[…] implemented our approach as a delegation extension […]", p. 235
[53]	Business model repr. lang. for business model analysis	**Migration** of VDML [58] by adaptation for specialized VDML business model constructs and views	"This paper extends the previous work by combining […]", p. 65
[59]	Recommendation-based conceptual modeling and ontology evolution framework (CMOE+)	**Incorporation:** The variant incorporates e.g. versions of the of the Unified Foundational Ontology [60] **Migration**: Conceptualization of a generic framework (CMOE+) and extending this framework for a BPMN instantiation as a new variant **Coevolution**: The variant exists in coevolution with the eclipse plugin extension for BPMN2 Modeler	"[UFO] was selected as a core ontology", p. 242 "CMOE+ […] may be instantiated and further specialized to support different concrete modeling languages.", p. 236
[55]	Business process improvement (BPI) roadmap with conceptual modeling support	**Incorporation**: the BPI roadmap (first version) incorporates versions of artifact types such as SIPOC and KPI **Adaptation**: Conceptualizing a BPI roadmap [61] and adapting it for further conceptual modeling support [62], [55] as a new artifact version. **Coevolution** with ADOxx [63]	"The roadmap […] is […] building on BPI techniques", p. 252; "the techniques were sequentially arranged to form a roadmap", p. 258 "Based on this BPI roadmap, a domain-specific conceptual modeling method (DSMM) has been developed.", p. 251 "a corresponding software prototype has been implemented using a meta modeling platform ", p. 251
[54]	Extended ITML for IT platforms and multilevel model of IT platforms	**Migration** of Memo ITML [64] for IT platform constructs as a new variant of MML [65] **Coevolution**: Memo ITML for IT platforms exists in coevolution with the Flexible Meta modeling language implemented in Xmodeler [66]	"we undertake an attempt to extend ITML […] with the required aspects", p. 318 "we apply FFML to model different aspects of IT platforms.", p. 234
[67]	Business-oriented Service Description Language	**Migration**: Developing a UML-profile for the graphical Notation of BoSDL as a variant of UML	"[…] develop such a format based on the UML, which we adapted by creating a profile", p. 12

Gailly et al. [59] developed a recommendation-based conceptual modeling and ontology evolution framework (CMOE+) that they present as a generic framework in its first version. In its second version a specific instantiation for CMOE+BPMN was developed. Because of the different versions being either generic (CMOE+) or specific (CMOE+BPMN) we present these versions as variants. The authors describe that this artifact type "might further be instantiated and further specialized to support different concrete modeling languages" [59], p.236 indicating that there may be further variants. Gailly et al. incorporate versions of artifact types such as the Unified Foundational Ontology (UFO) [60] into their artifact. Furthermore, they describe that they implemented an eclipse plugin in their BPMN2 modeling tool, which may indicate that CMOE+BPMN is in coevolution with BPMN2 modeler. Please note that these two examples represent our interpretations of the authors' publications. Although two authors of this paper coded independently and discussed and merged their results, our view may include misinterpretations and is limited to aspects we found relevant. We suggest the terminology for evolving design artifacts to be more powerful in case the authors use it themselves to describe their artifact evolution rather than other researchers trying to interpret their results.

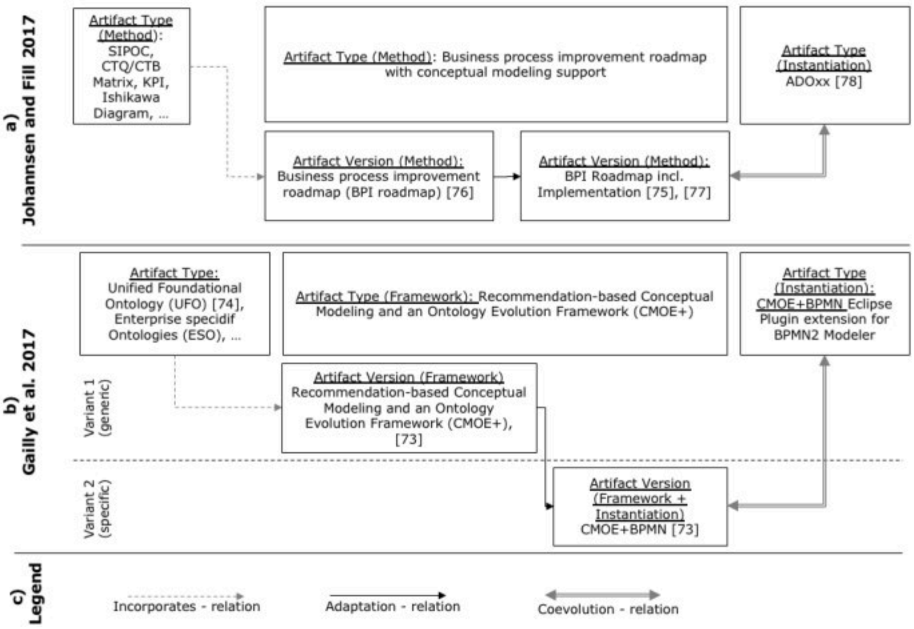

Figure 3. Describing an artifact evolution using a) the example from [55], b) the example from [59] and c) providing a legend for the different relations that occur in evolving design artifacts

5 Discussion

The terminology to describe evolving design artifacts, summarized in Table 1 and Fig. 2, can be used to help researchers to increase rigor by grounding their projects with existing artifacts. Accordingly, design researchers may clarify how present and prior artifacts relate to one another or can describe artifacts that they coevolve with. Thereby, design researchers address mutability of their design artifacts such as Gregor and Jones asked for [7]. Furthermore, we suggest that the terminology for evolving design artifacts helps to decide where to focus on when evaluating design artifacts as suggested by Gregor and Hevner [68]. For example, when adapting an artifact version that has been evaluated profoundly and that has positive impact, evaluations of the new version may focus on those features that have been added or abandoned instead of again evaluating the whole version. In contrast, when evaluating an artifact variant, evaluation can focus on distinct characteristics of the application domain the variant is used in.

To improve relevance, design researchers can use the terminology for evolving design artifacts to clarify their contribution. In analogy to the design science contribution types presented in [68], design researchers can distinguish different contribution types for their artifacts. As one contribution type, their artifact instance may base on well-developed and evaluated artifact instances and thus, include mature knowledge. In this case, design researchers could discuss the distance between their current and preceding artifact instances, e.g. by devising the number of new functions or abandoned functions, as well as the effort required to implement or abandon these functions. As another contribution type, researchers might describe their design artifacts to only have a limited evolutionary lineage. In this case, design researchers justify their contribution by discussing the novelty and originality of their ideas as well as the problems their artifacts help to solve. Furthermore, as mutations in design projects are subject to human activity, relevance for future design projects might be increased by facilitating mutations. For example, allowing for more creativity in design projects or for appropriate time slots for evolving the design artifact might impact on the number and quality of mutations. For instance, the act of creativity has been discussed as bisociation, that can be achieved by relating issues that at first seem unrelated to produce new insights [69]. Bisociation has been discussed in psychology but to our knowledge not been reflected for design science.

So far, the terminology for evolving design artifacts does not fully relate to recent mutability approaches. In-design mutability refers to the degree of change encompassed within one artifact version [7] and does not reflect relationships between artifacts. In-use mutability describes artifacts to evolve over time by allowing users to adapt the artifact. However, also designers evolve their artifacts (e.g. [53–55, 59]), which is not reflected in in-use mutability. We suggest that the terminology for evolving design artifacts may enhance in-use or may be treated as a unique aspect.

6 Limitations

As this work represents an interim struggle in theorizing we include a discussion of potential boundaries that need further investigation. We have limited our analysis to conceptual modeling design research but presume that these concepts can be applied to other IS domains that deal with design science. We suspect that coevolution is possibly not a relevant type to describe artifact evolution in any IS domain. In conceptual modeling, coevolution was reconstructed for artifact types and their corresponding modeling tool. Accordingly, for this domain we found coevolution to be tool-dependent and might hence not apply to any domain. Yet, more research is needed to specify whether coevolution is indeed dependent on tools.

Furthermore, one might question whether concepts of the terminology for evolving design artifacts may apply to any conceptual modeling project. In the evaluation we have only focused on types of artifact evolutions that were published in the BISE-journal from 2014-2018. We found that artifact evolutions had a low to mid-range complexity including one to three different evolutionary relationships. For example, Fill and Johannsen described an artifact evolution that includes an incorporation, adaptation as well as coevolution [55]. In contrast, other artifact evolutions only included one type of artifact evolution and might in this sense be described as less complex in terms of evolutionary steps. Yet, we suspect that concepts of the terminology for evolving design artifacts may be also used for more complex evolutions such as the evolution of the UML or BPMN. We have already pinpointed to how these different types of evolution may correspond to the development of UML in the introduction but agree that more research is needed to proof whether concepts can also be applied to large artifact design projects.

7 Conclusion and Future Research

As design artifacts evolve over time, design researchers need a terminology to describe this evolution. We have developed such a terminology based on evolution in biology and evaluated whether its concepts and relationships are already used, although in other words. We found that authors describe evolution in their own words but lack a consistent terminology. For further evaluation we suggest to use a larger sample size and to check the coders' interpretations against the designers' experience by conducting interviews. Thereby, usefulness of the terminology and the need for further notions may be assessed. So far, we only provide examples of evolving artifacts [55, 59]. A systematic approach to describe evolving design artifacts might help to compare evolutions of different design artifacts and thus, to render evolution of design artifacts more transparent. The terminology derived in this paper bridges the gap between the need to address mutability [7] and the guidance required to do so [70]. So far, this guidance is available as a terminology and not as a mid-range theory [71] between evolution in biology and evolution in design artifacts. To further approach theory, e.g. boundaries need to be investigated, concepts need to be refined for other domains, propositions about e.g. typical sequences in which types of evolutions occur should be developed and translated into hypotheses.

References

1. Platt, R., Thompson, N.: The Evolution of UML. In: Encyclopedia of Information Science and Technology, Third Edition. pp. 1931–1936. IGI Global (2015).
2. Object Management Group: Systems Modeling Language (SysML) - Verion 1.5, (2017).
3. Modelio, https://www.modelio.org/. Download: 2018-08-01
4. Pöppelbuss, J., Goeken, M.: Understanding the Elusive Black Box of Artifact Mutability. In: Wirtschaftsinformatik. pp. 1557–1571 (2015).
5. Sjöström, J., Agerfalk, P.J., Lochan, A.R.: Mutability Matters: Baselining the Consequences of Design. Procedings MCIS. (2011).
6. Gregor, S., Iivari, J.: Designing for mutability in information systems artifacts. Inf. Syst. Found. Theory Represent. Real. Eds Hart Gregor Aust. Natl. Univ. Press Canberra (2007).
7. Gregor, S., Jones, D.: The anatomy of a design theory. J. Assoc. Inf. Syst. 8, 312–335 (2007).
8. Trigg, R.H., Bødker, S.: From implementation to design: tailoring and the emergence of systematization in CSCW. In: Conf. on Comp. supp. coop. work. pp. 45–54. ACM (1994).
9. Hevner, A.R.: A three cycle view of design science research. Scand. J. Inf. Syst. 19, 4 (2007).
10. Peffers, K., Tuunanen, T., Rothenberger, M.A., Chatterjee, S.: A Design Science Research Methodology for Information Systems Research. J. Manag. Inf. Syst. 24, 45–77 (2007).
11. Winter, R.: Design science research in Europe. Eur. J. Inf. Syst. 17, 470–475 (2008).
12. Weick, K.E.: What theory is not, theorizing is. Adm. Sci. Q. 40, 385–390 (1995).
13. Krippendorff, K.: Content analysis: An introduction to its methodology. Sage (2012).
14. Hall, B., Strickberger, M.W.: Strickberger's evolution. Jones & Bartlett Learning (2008).
15. Hartl, D.L., Clark, A.G., Clark, A.G.: Principles of population genetics. Sinauer associates Sunderland (1997).
16. Mayr, E.: What evolution is. Science Masters Series (2001).
17. Copley, S.D.: Evolution of a metabolic pathway for degradation of a toxic xenobiotic: the patchwork approach. Trends Biochem. Sci. 25, 261–265 (2000).
18. Dobzhansky, T., Dobzhansky, T.G.: Genetics of the evolutionary process. Columbia University Press (1970).
19. Kopp, O., Binz, T., Breitenbücher, U., Leymann, F.: BPMN4TOSCA: A Domain-Specific Language to Model Management Plans for Composite Applications. In: Mendling, J. and Weidlich, M. (eds.) Business Process Model and Notation. pp. 38–52. Springer (2012).
20. Pillat, R.M., Oliveira, T.C., Alencar, P.S.C., Cowan, D.D.: BPMNt: A BPMN extension for specifying software process tailoring. Inf. Softw. Technol. 57, 95–115 (2015).
21. Graja, I., Kallel, S., Guermouche, N., Kacem, A.H.: BPMN4CPS: A BPMN Extension for Modeling Cyber-Physical Systems. In: 2016 IEEE 25th Int. Conf. on Enabling Techn.: Infrastr. for Coll. Ent. pp. 152–157 (2016).
22. Yousfi, A., Bauer, C., Saidi, R., Dey, A.K.: uBPMN: A BPMN extension for modeling ubiquitous business processes. Inf. Softw. Technol. 74, 55–68 (2016).
23. Gregory, T.R.: Understanding natural selection: essential concepts and common misconceptions. Evol. Educ. Outreach. 2, 156 (2009).
24. Masel, J.: Genetic drift. Curr. Biol. 21, R837–R838 (2011).
25. Lande, R.: Fisherian and Wrightian theories of speciation. Genome. 31, 221–227 (1989).

26. Gavrilets, S.: Perspective: models of speciation: what have we learned in 40 years? Evolution. 57, 2197–2215 (2003).
27. Dobzhansky, T.: On some fundamental concepts of Darwinian biology. In: Evolutionary biology. pp. 1–34. Springer (1968).
28. Nakajima, A., Sugimoto, Y., Yoneyama, H., Nakae, T.: High-Level Fluoroquinolone Resistance in Pseudomonas aeruginosa Due to Interplay of the MexAB-OprM Efflux Pump and the DNA Gyrase Mutation. Microbiol. Immunol. 46, 391–395 (2002).
29. Wade, M.J.: The co-evolutionary genetics of ecological communities. Nat. Rev. Genet. 8, 185 (2007).
30. Benton, M.J.: Diversification and extinction in the history of life. Science. 268, 52–58 (1995).
31. Wilson, R.A.: Species: New interdisciplinary essays. (1999).
32. Short, R.V.: The contribution of the mule to scientific thought. J. Reprod. Fertil. Suppl. 359–364 (1975).
33. Klepper, S.: Entry, exit, growth, and innovation over the product life cycle. Am. Econ. Rev. 562–583 (1996).
34. Orr, H.A.: The genetic theory of adaptation: a brief history. Nat. Rev. Genet. 6, 119 (2005).
35. Decker, G., Kopp, O., Leymann, F., Pfitzner, K., Weske, M.: Modeling service choreographies using BPMN and BPEL4Chor. In: Int. conf. on Adv. Inf. Syst. Eng. pp. 79–93. (2008).
36. Futuyma, D.J., Slatkin, M.: Coevolution. Sinauer Associates Incorporated (1983).
37. Sih, A.: Predator and prey lifestyles: an evolutionary and ecological overview. Predation Direct Indirect Impacts Aquat. Communities. 203–224 (1987).
38. Lunau, K.: Adaptive radiation and coevolution—pollination biology case studies. Org. Divers. Evol. 4, 207–224 (2004).
39. McKelvey, B.: Perspective-quasi-natural organization science. Organ. Sci. 8, 351–380 (1997).
40. Barnett, W.P., Hansen, M.T.: The red queen in organizational evolution. Strateg. Manag. J. 17, 139–157 (1996).
41. Koza, M.P., Lewin, A.Y.: The co-evolution of strategic alliances. Organ. Sci. 9, 255–264 (1998).
42. Peppard, J., Breu, K.: Beyond alignment: a coevolutionary view of the information systems strategy process. ICIS 2003 Proc. 61 (2003).
43. Tiwana, A., Konsynski, B., Bush, A.A.: Research commentary-Platform evolution: Coevolution of platform architecture, governance, and environmental dynamics. Inf. Syst. Res. 21, 675–687 (2010).
44. Kutschera, U., Niklas, K.J.: Endosymbiosis, cell evolution, and speciation. Theory Biosci. 124, 1–24 (2005).
45. Mereschkowsky, C.: Über natur und ursprung der chromatophoren im pflanzenreiche. Biol. Cent. 25, 293–604 (1905).
46. Rummler, G.A., Brache, A.P.: Improving performance: How to manage the white space on the organization chart. John Wiley & Sons (2012).
47. Binner, H.F.: Anforderungsgerechte Datenermittlung für Fertigungssteuerungssysteme. Beuth Verlag, Berlin (1987).
48. Verschuren, P., Hartog, R.: Evaluation in design-oriented research. Qual. Quant. 39, 733–762 (2005).
49. Chinosi, M., Trombetta, A.: BPMN: An introduction to the standard. Comput. Stand. Interfaces. 34, 124–134 (2012).

50. Weilkiens, T.: Systems engineering with SysML/UML: modeling, analysis, design. Elsevier (2011).
51. Jacobson, I.: Object-oriented software engineering: a use case driven approach. Pearson Education India (1993).
52. Sutton, R.I., Staw, B.M.: What theory is not. Adm. Sci. Q. 371–384 (1995).
53. Roelens, B., Poels, G.: The development and experimental evaluation of a focused business model representation. Bus. Inf. Syst. Eng. 57, 61–71 (2015).
54. Kaczmarek-Hess, M., de Kinderen, S.: A multilevel model of IT platforms for the needs of enterprise IT landscape analyses. Bus. Inf. Syst. Eng. 59, 315–329 (2017).
55. Johannsen, F., Fill, H.-G.: Meta Modeling for Business Process Improvement. Bus. Inf. Syst. Eng. 59, 251–275 (2017).
56. Schefer-Wenzl, S., Strembeck, M.: Modeling Support for Role-Based Delegation in Process-Aware Information Systems. Bus. Inf. Syst. Eng. 6, 215–237 (2014).
57. Strembeck, M., Mendling, J.: Modeling process-related RBAC models with extended UML activity models. Inf. Softw. Technol. 53, 456–483 (2011).
58. OMG: Value Delivery Modeling Language (VDML), (2014).
59. Gailly, F., Alkhaldi, N., Casteleyn, S., Verbeke, W.: Recommendation-Based Conceptual Modeling and Ontology Evolution Framework (CMOE+). Bus. Inf. Syst. Eng. 59, 235–250 (2017).
60. Guizzardi, G., Wagner, G., Almeida, J.P.A., Guizzardi, R.S.: Towards ontological foundations for conceptual modeling: the unified foundational ontology (UFO) story. Appl. Ontol. 10, 259–271 (2015).
61. Johannsen, F., Fill, H.-G.: Codification of knowledge in business process improvement projects. (2014).
62. Johannsen, F., Fill, H.-G.: Supporting business process improvement through a modeling tool. In: Domain-Specific Conceptual Modeling. pp. 217–237. Springer (2016).
63. Fill, H.-G., Karagiannis, D.: On the conceptualisation of modelling methods using the ADOxx meta modelling platform. Enterp. Model. Inf. Syst. Archit. 8, 4–25 (2013).
64. Heise, D.: Unternehmensmodell-basiertes IT-Kostenmanagement als Bestandteil eines integrativen IT-Controllings. Logos Verlag Berlin GmbH (2013).
65. Frank, U.: The MEMO meta modelling language (MML) and language architecture. ICB-research report (2011).
66. Frank, U.: Multilevel modeling. Bus. Inf. Syst. Eng. 6, 319–337 (2014).
67. Schlauderer, S., Overhage, S.: BoSDL: An Approach to Describe the Business Logic of Software Services in Domain-Specific Terms. Bus. Inf. Syst. Eng. 1–21.
68. Gregor, S., Hevner, A.R.: Positioning and presenting design science research for maximum impact. MIS Q. 37, (2013).
69. Koestler, A.: The act of creation. (1964).
70. Gehlert, A., Schermann, M., Pohl, K., Krcmar, H.: Towards a Research Method for Theory-driven Design Research. In: Wirtschaftsinformatik (1). pp. 441–450 (2009).
71. Kuechler, W., Vaishnavi, V.: A framework for theory development in design science research: multiple perspectives. J. Assoc. Inf. Syst. 13, 395 (2012).