GUIDING THE DEVELOPMENT OF ENTERPRISE DESIGN APPROACHES

M. de Vries[1]*

ARTICLE INFO

Article details
Presented at the 27th annual conference of the Southern African Institute for Industrial Engineering (SAIIE), held from 27-29 October 2016 at Stonehenge in Africa, North West, South Africa

Available online 11 Nov 2016

Contact details
* Corresponding author
marne.devries@up.ac.za

Author affiliations
1 Department of Industrial and Systems Engineering
University of Pretoria, South Africa

DOI
http://dx.doi.org/10.7166/27-3-1621

ABSTRACT

Enterprise engineering (EE) is an emerging discipline that aims to address several phenomena in the evolution of an enterprise. Previous research highlighted a particular phenomenon that requires more research — i.e., the *inability* of the enterprise as a complex socio-technical system to adapt swiftly to rapidly-changing environments. Although there are numerous theoretical design approaches (with their own methodologies, frameworks, and modelling languages), there is little empirical evidence about their effectiveness. Furthermore, research indicates that multiple enterprise design approaches are used concurrently in industry, with each approach focusing on a sub-set of stakeholder concerns. The proliferating design approaches do not necessarily explicate their conditional use in terms of contextual prerequisites and demarcated design scope; and this also impairs their evaluation. This article provides guidance for the construction of design approaches that will enable researchers to contribute to the systematic growth of the EE knowledge base.

OPSOMMING

Ondernemingsingenieurswese (OI) is ’n opkomende dissipline wat fokus op fenomene wat verband hou met die evolusie van die onderneming. Vorige navorsing het getoon dat een spesifieke fenomeen addisionele navorsing benodig, naamlik die *onvermoeë* van die onderneming as ’n komplekse sosio-tegniese stelsel, om aan te pas by ’n dinamies-veranderende omgewing. Alhoewel veelvuldige teoretiese ontwerpbenaderings bestaan (insluitende metodologieë, raamwerke en modelleringstale), is daar ’n gebrek aan empiriese bewyse rakende hul effektiwiteit. Verder toon navorsing dat maatskappye gewoonlik gebruikmaak van ’n kombinasie van ondernemingsontwerpsbenaderings wat elkeen op ’n ander stel belange fokus. Nuwe ontwerpsbenaderings dui egter nie altyd hul beperkte en voorwaardelike bruikbaarheid aan in terme van konteks en omvang nie, en hul evaluasie word derhalwe bemoeilik. Hierdie artikel voorsien riglyne aan navorsers vir die konstruksionele ontwerp van nuwe ontwerpsbenaderings om sodoende die sistematiese ontwikkeling van die OI kennisbasis aan te moedig.

1 INTRODUCTION

Enterprise engineering (EE) is an emerging discipline that has emerged from the need for a comprehensive view of the enterprise [1, 2, 3, 4]. Recent research has focused on defining the domain of the EE discipline — i.e., identifying the phenomena and core problems of interest that require research within the discipline [5]. Although participants in that survey identified multiple phenomena/problems of interest, one phenomenon was highlighted: “The enterprise as a complex socio-technical system and its *inability* to adapt swiftly to rapidly-changing environments” [5, p 58]. The research also highlighted the need for appropriate *methodologies* and *architecture* *description* (using typologies and modelling languages) [5]. Yet numerous theoretical methodologies, frameworks, and modelling languages already exist [6]. Another survey, performed by Blowers [7], also indicated that two-thirds of the sampled enterprises applied a *hybrid approach*, using multiple modelling languages and frameworks. The diversity of modelling languages is not surprising if we recognise that enterprise stakeholders have different concerns and ideologies, and that they use descriptions/models that elucidate their own particular concerns [8].

Prior to constructing a new approach, researchers usually present evidence of the existence of a phenomenon and the inability of existing approaches to address the phenomenon adequately. Rather than stifling the emergence of new enterprise design approaches, this article encourages the enterprise designer to define his/her own ideologies in respect of enterprise design, to explicate the stakeholder concerns that will be addressed by their approach and the conditional use of their approach in terms of enterprise context and demarcated design scope. Explication will not only improve the usefulness of a new design approach: it will also enable its evaluation against its initial goals and comparison with existing approaches, thus demonstrating its contribution to the EE knowledge base. The article thus addresses the following research question:

Which principles should guide approach developers when designing and explicating new approaches, so that they are useful to the user of the approach, and effectively contribute to the existing EE knowledge base when explicating their design approach?

Key concepts embedded in the research question — namely, *principles* and *approach* — are clarified before presenting existing theory to extract useful *principles*. When referring to *principles*, Aier *et al.* [9] developed a useful meta-model for defining an enterprise architecture principle: (1) the principle statement, (2) its rationale, (3) principle measures, (4) implications, and (5) key actions. Similar to Beer *et al.* [10], we believe that the meta-model is also useful within associated contexts — e.g., to define principles to develop new/adapted *enterprise design approaches*. Clarifying the concept of *approach,* the classification schema of De Vries *et al.* [5], called the enterprise evolution contextualisation model (EECM), indicates that theoretical enterprise design *approaches* could be described in terms of four components:

- Concept of the enterprise and paradigm of value-creation.
- Dimensions, defining the design scope of the approach in terms of organising scope, design domains, and concerns and constraints.
- Mechanisms and practices to enable enterprise design.
- Approach classifiers that depict the common patterns of existing approaches.

Although the EECM is a representation of existing enterprise design approaches, it may lack prescriptive abilities. The next section indicates how we apply design science research to develop a set of guiding *principles* for approach developers.

2 RESEARCH METHOD

Design research acknowledges the development of *principles* as a valid form of knowledge contribution [11]. Referring to the design research cycle steps of Peffers *et al.* [12], this article focuses on the first three steps as follows:

1. *Identify problem and motivate:* Section 1 states the problem — i.e., the proliferation of enterprise design approaches that do not necessarily explicate conditional use in terms of contextual prerequisites and demarcated design scope. This impairs their evaluation and the systematic growth of the EE knowledge base.
2. *Define objectives of solution:* Section 1 also identifies the need to guide the development of new/adapted enterprise design approaches by developing *principles* that would be useful to the approach developer.
3. *Design and develop*: Section 3 presents existing theory that would enable the development of guiding *principles*. We argue that EECM embodies typical constructional components of an enterprise design approach, but also draws on prescriptive knowledge of information system *design theory* to develop a prescriptive set of eleven *principles*. The principles have been refined via a focus group discussion, consulting with five participants (approach designers) on the clarity, validity and usefulness of the principles. The refined principles are presented in Section 4.

Although this article only focuses on the first three steps of the design research cycle, we elaborate on future work in Section 5, which include further refinement, demonstration, and evaluation of the *principles*.

3 BACKGROUND

We base our guiding *principles* on several existing knowledge areas. Section 3.1 defines the enterprise as a system and discusses enterprise engineering (EE) as a discipline. Section 3.2 provides an introduction to EECM as a way of guiding the development of new approaches, highlighting the *foundation* of enterprise design approaches — i.e., defining the *intentions* of an approach, its demarcating *design scope*, and the associated *mechanisms and practices*. Section 3.3 presents components of information system *design theory*.

3.1 The enterprise system and enterprise engineering

Although several analogies are used to define the enterprise [13], several academics regard the enterprise as a *system* [4, 8, 14, 15]. Yet systems are not objective constructs that everyone can observe or touch; rather, they are mental constructs that consist of interrelated parts that we experience as a whole [8]. Naming a system implies that we distinguish the system from its environment, specifying a border or boundary.

The observer of the enterprise usually describes the enterprise system from two different positions [8]. First, the observer could observe the system from *outside the boundary*, and describe it as a black box (using *black box models*) constrained by its environment. The observer typically describes phenomena beyond enterprises to understand inputs, transformations, and outputs of social and economic processes, using an appropriate modelling practice such as systems dynamics [8]. Second, the observer could observe the system from *inside the boundary*, describing it in terms of its internal structures (using *white box models*) and its reaction to environmental factors. Whereas *black box* models are useful to management when controlling a system, *white box* models are useful to the enterprise designer to (re-)design or maintain the system [14].

Enterprise engineering (EE), defined as "the body of knowledge, principles, and practices to *design* an enterprise" [4], highlights the *white box* models — i.e., the *inside-the-boundary* perspective. Espejo and Reyes [8] label these the *cohesion function* of the enterprise. The *cohesion function* needs to *steer* "the implementation function in the direction of the collective's purposes" [8]. Focusing on the *adaptability* of the enterprise, Beer [16] claims that the *cohesion function* (taking an *inside-the-boundary* perspective) is not sufficient to ensure adaptability. Two additional functions are required; an *intelligence function* and a *policy function*. The *intelligence function* needs to take an *outside-the-boundary* perspective on future opportunities and threats, and on innovative ways to deal with those opportunities and threats. The *policy function* refers to policy-makers who need to respond to both sources of complexity: *inside-the-boundary* matters of internal efficiency, and the vital signs from *outside-the-boundary* [16].

The EE community has also emphasised the need for research on enterprise *adaptability,* since a prominent phenomenon is still the inability of the enterprise to adapt swiftly to rapidly-changing environments [5]. In terms of the three key functions identified by Beer [16] to ensure enterprise adaptability, numerous approaches are available in the literature. Some focus more on the *intelligence* and *policy functions*, such as systems dynamics approaches. Systems dynamics models provide a way to understand the *outside-the-boundary* complexities, demonstrating to *policy-makers* the effect of different scenarios and decisions [17]. According to Lapalme [18], many approaches within the EE community still focus on the *coherence function*. And there are some approaches within the EE community that also include the *intelligence function* and the *policy function* [19].

3.2 The essence of design approaches for enterprises

As stated in Section 1, previous research has validated the *enterprise evolution contextualisation model* (EECM) as a representation of existing theoretical enterprise design approaches [6]. Although EECM was not presented as a prescriptive model for developing new approaches, we argue that the first two components would be useful to an approach designer who seeks to formulate the *intentions and scope* of a new approach; whereas the third component provides the way to address the approach *intentions and scope*.

Component 1: Concept of the enterprise and paradigm of value-creation

The *concept of the enterprise* relates to the different concepts and/or analogies used by authors to define an enterprise — e.g., referring to enterprises as machines, organisms, multi-minded systems, cultures, political systems, and organised complexities [13, 15, 20]. One of the prominent ways of defining the enterprise is to view the enterprise as a system, as discussed in section 2.1.

The *belief/paradigm of creating value* refers to the philosophical stance of the approach author(s) and their belief system about what should create value for an enterprise. For example, some approach authors value an aggregate view/blueprint of an enterprise for directing the enterprise in terms of required high-level processes and IT capabilities [21, 22, 23]. Other approach authors value the continuous process of transformation to translate business vision and strategy into effective enterprise change [24, 25, 26, 27]. Authors emphasise different perceived challenges, insights, objectives, and concerns [28], which then influence their *value-creation paradigm*.

Together, the *concept of the enterprise and the paradigm of value-creation* direct the entire approach, also providing the grounds for the type of activities included in the supporting *mechanisms and practices* [6]. For example, Dietz *et al*. [20, p 93-95] indicate that their *concept of the enterprise* is an "organised complexity", a "complex adaptive system" analogous to an "improvisational theatre", whereas their *paradigm of value-creation* is "enterprise (re)-development in an all-encompassing way".

Component 2: Dimensions (enterprise scope, design domains, concerns, and constraints)

The *enterprise scope* provides a dimension to demarcate the scope of design for a specific approach, by referring to the *internal structures* of the enterprise — e.g., business units, lines of business, departments, programmes and projects — and the *external legal entities*, such as government, partners and suppliers. Espejo and Reyses [8] provide a coarser distinction of *enterprise scope*: *inside-the-boundaries* scope versus *outside-the-boundaries* scope.

Design domains provide another way to demarcate design scope. Although the literature reveals many different conceptualisations for *design domains*, demarcation of boundaries is contextual and depends on the intentions of the observer/analyst [4, 8]. Thus the demarcation of design domains also depends on EECM's first component — i.e., the approach author's intentions or *paradigm of value-creation*. For example, Hoogervorst [3, p 134] maintains that the demarcation/delineation of domains reveals "functional or constructional *system facets* for which design activities are required". Dietz [14] delineates design domains as *sub-systems* for which design activities are required, whereas Zachman [29] provides a different way of demarcating design domains using *six interrogatives* (what, how, where, who, when, why). Others demarcate design domains *without* explicating how to demarcate them; e.g., The Open Group [30] defines three explicit design domains: business, information system (which includes application and data), and technology. Depending on their intentions, approach authors often add mechanisms such as languages, methods, and practices to develop descriptive representations of the design domains [6]. Although there are several notation standards that describe different design domains, Lankhorst [31] reasons that the existing languages fail to describe, analyse, and visualise all the essential elements of an enterprise. Yet the appropriateness of languages and representations once again depends on the intentions of the approach author. For example, Dietz [14, p 8] prescribes a conceptual model of the enterprise that is "coherent, comprehensive, and concise", only showing the essence of the operation of an enterprise; whereas Frank [32] believes that users should have the ability to define their own conceptual models rather than using prescribed languages and notations.

Concerns and constraints acknowledge a third way of demarcating enterprise design scope. Approach authors highlight particular *concerns* or *areas of concern* (depending on their *paradigm of value-creation)* that should be addressed during enterprise design. For example, Parmenter [33] highlights six areas of concern: financial asset utilisation, operational performance, customer satisfaction, employee satisfaction, community engagement, and learning and growing. Alternatively they prescribe mechanisms for gathering, aligning, and measuring stakeholder *concerns/requirements*, which should be used when designing and evaluating a new or future version of the enterprise or one of its sub-systems [6]. Mechanisms such as big data technologies and customer experience management (CEM) accelerate the process of identifying emerging customer concerns [34]. EECM highlights that some approaches, such as that of Hoogervorst [3], take a holistic stance to identify and align multiple concerns during enterprise design [6]. *Constraints* impose further restrictions on pre-defined *concerns*; e.g., the overall *cost* of a (re-)designed sub-system may not exceed a pre-defined budget.

Component 3: Mechanisms and practices

According to de Vries *et al.* [6], various different kinds of mechanisms and practices are prescribed by enterprise design approaches. Two prominent mechanisms, *governing principles* and *methodologies*, enact the *value-creation paradigm* and the *design scope* of the approach. *Governing principles* or decision criteria are used to guide enterprise design, whereas *methodologies* provide generic steps for designing the enterprise [6]. We provide examples of *methodologies* for approaches that focus on two levels of *enterprise scope*:

- *Outside-the-boundary* approaches, such as the strategic process of Johnson *et al.* [35], prescribe *methodologies*: to sense emerging *concerns* (threats and opportunities) within the environmental context of the enterprise; to develop alternative scenarios (represented, for example, as systems dynamics scenarios as defined by Radzicki *et al.* [17] or as business model scenarios as defined by Ostewalder *et al.* [36]); and to evaluate the effect of the scenarios on other enterprise *concerns*.
- *Inside-the-boundary* approaches, such as The Open Group's [30] architecture development method (ADM), prescribe *methodologies*: to identify and address enterprise *concerns* systematically; to develop alternative constructions (e.g., alternative descriptive representations for a particular design domain); and to evaluate the effect of alternative constructions on pre-defined *concerns* [6].

Van Aken [37] emphasised the need to guide *prescriptive theory* ('how to' knowledge) within the management discipline. Gregor and Jones [38] identified a similar need within the information system (IS) discipline, but also provided guidance in the form of eight components for *design theory* in IS. Since there is little *prescriptive theory* to guide the development of enterprise design approaches, we investigated the applicability of Gregor and Jones' [38] eight components of design theory as a way to formulate *principles* that would guide approach development.

3.3 Components of design theory

Gregor and Jones [38] argue that any design theory should include at least six of eight components (marked with * in Table 1).

The next section uses the eight components of Gregor and Jones [38] and the knowledge about existing theoretical design approaches, as represented by EECM, to suggest a set of eleven *principles*.

Table 1: Eight components of IS theory, based on Gregor and Jones [38]

No	Component for IS theory	Description
1*	Purpose and scope	'What the system is for'; the goals, scope and boundaries of the theory. Offerman *et al.* [39] also add *role coverage* as a way to define scope.
2*	Constructs	Representations of the entities of interest in the theory using words, mathematical symbols, or parts of a diagram.
3*	Principle of form and function	The abstract 'blueprint' that describes the artefact — i.e., the structure, organisation, and functioning of the design product of design method.
4*	Artefact mutability	The changes in state of the artefact anticipated in the theory — i.e., what degree of artefact change is encompassed by the theory.
5*	Testable propositions	Truth statements about the design theory that can either be proved or falsified.
6*	Justificatory knowledge	The underlying knowledge or theory from the natural or social or design sciences that gives a basis and explanation for the design.
7	Principles of implementation	A description of processes to implement the theory (either product or method) in specific contexts. Either *principles* or a generic design *method* can guide the implementation.
8	Expository instantiation	A physical implementation of the artefact that can assist to represent the theory both as an illustrative device and for the purposes of testing.

4 PRINCIPLES TO DEVELOP APPROACHES AS ARTEFACTS

This section presents the set of ten principles according to the principles meta-model of Aier *et al.* [9]. It includes a statement, rationale, implications (for the approach developer), and measures. We excluded the fifth component (key actions) of the meta-model, since these will vary for each application of the principle.

Principle A: Explicit concept of the enterprise

Statement: A design approach should indicate how an enterprise is perceived or *conceptualised.*

Rationale: Different analogies are used to conceptualise the enterprise, such as machines, biological systems, and psychic prisons [6], which may also differ from one industry to the next. The conceptualisation of the enterprise also has an influence on how the approach author demarcates *design domains* and provides *descriptive representations* of the enterprise.

Implications:

- Provide a description of the enterprise, using analogies.
- Provide a motivation, also referring to *supporting theory*, for using the particular enterprise conceptualisation.

Measures: No additional measures.

Principle B: Explicit phenomenon

Statement: A design approach should provide evidence for a *phenomenon* or *class-of-problems*, i.e. similar kinds of problems.

Rationale: Phenomena "that are not fully understood cannot be properly addressed and improved" [40, p 49]. Although numerous approaches may already exist to address a well-understood phenomenon, there is still a lack of knowledge about approach performance [5].

Implications:

- Produce *sufficient* evidence that an existing *phenomenon* or *class-of-problems* exists, but that it is inadequately addressed by existing theory or theory application.

Measures: No additional measures.

Principle C: Explicit paradigm of value creation

Statement: A design approach should state a *paradigm of value-creation* as a testable proposition for addressing an existing *phenomenon* or *class-of-problems.*

Rationale: Creating testable propositions for existing and new approaches provides a starting point to extend the existing EE knowledge base.

Implications:

- State the intended *paradigm of value-creation* in terms of a testable proposition, which may be in the form 'If approach X is instantiated, then it will achieve the intended value, or it will be better in some way than other approaches'.

Measures: No additional measures.

Principle D: Explicit means (ways) of demarcating and representing design scope

Statement: A design approach should clearly define and motivate the way to demarcate design scope (*enterprise scope, design domains,* and *concerns/requirements*) relevant to the approach.

Rationale: Demarcation of design scope (*enterprise scope, design domains, concerns/requirements*) is contextual and depends on the intentions (*paradigm of value-creation*) of the observer/ analyst [4, 8].

Section 3.2 provides definitions of *enterprise scope*, *design domains* and *concerns*, which we repeat for clarity:

The *enterprise scope* provides a dimension to demarcate the scope of design for a specific approach, by referring to the *internal structures* of the enterprise — e.g., business units, lines of business, departments, programmes and projects — and the *external legal entities*, such as government, partners and suppliers.

The demarcation of *design domains* depends on the approach author's intentions or *paradigm of value-creation*. For example, Hoogervorst [3, p 134] maintains that the demarcation/delineation of domains reveals "functional or constructional *system facets* for which design activities are required".

Concerns acknowledge a third way of demarcating enterprise design scope. Approach authors highlight particular *concerns* or *areas of concern/requirements* that should be addressed during enterprise design. For example, Parmenter [33] highlights six areas of concern: financial asset utilisation, operational performance, customer satisfaction, employee satisfaction, community engagement, and learning and growing.

Implications:

- Define the way to demarcate design domains (e.g., using systems theory or demarcation heuristics).
- Define the way to demarcate concerns per design domain (e.g., using generic areas of concerns from theory or using heuristics to identify concerns).

Measures: No additional measures.

Principle E: Well-demarcated and well-defended design scope

Statement: A design approach should define and defend the intended design scope to achieve the intended *value-creation*.

Rationale: The new approach should demarcate an appropriate scope to achieve the intended value-creation, and relate to existing theory that focuses on a similar scope.

Implications:

- Identify the overall *scope* of the approach — i.e., focusing on *inside-the-boundary* complexities versus *outside-the-boundary* complexities.
- Identify the scope of the approach in terms of *design domains* that will be addressed by the approach.
- Identify the scope of the approach in terms of areas of *concern* for different *stakeholders* that will be addressed.
- Use additional ways of scoping to define the context for which the approach is intended — e.g., a specific industry.
- Indicate the actual *implementation* scope used to demonstrate the approach.
- Identify the role players, especially the main *users* of the approach.

Measures:

- Is the scope described in such a way as to relate to existing theory that covers a similar design scope?

- Is the scope described in such a way as to know whether an application context falls within the scope of the approach?

Principle F: Representations of design scope

Statement: A design approach should clearly define and motivate notation standards that are used to adequately describe/represent the design scope.

Rationale: Multiple languages and notation standards already exist to represent different perspectives or design domains of the enterprise. Yet existing notation standards are usually based on a particular notion about the nature of the enterprise and its sub-systems. New notions and perspectives of the enterprise may require a deviation from existing notation standards. As an example, the Business Process Modelling Notation (BPMN) standard may be used to depict the business organisation domain. Yet, Dietz [14] criticizes BPMN for being too implementation-specific and motivates the need for representing the *essence* of enterprise operation by suggesting a new notation standard, i.e. the Design and Engineering Methodology for Organisations (DEMO).

Implications:

- Define notation standards that are used to describe *design domains*.
- Motivate any deviation from existing standards.

Measures: No additional measures.

Principle G: Approach form and function

Statement: A design approach should clearly define the constructs and features of the approach.

Rationale: Every approach has a distinct set of constructs that describe the detail of the approach. The four components of EECM provide a meta-model of typical constructs of an approach. The third component (*mechanisms and practices*) also highlights ten different kinds of constructs that may form part of an approach.

Implications:

- Define the overall structure and organisation of the approach.
- Define the *mechanisms and practices* explicitly, stating their form (conceptual parts) and function, ensuring that their interpretation is clear.
- Define how the *mechanisms and practices* are related to one another.
- Define the roles involved when using the *mechanisms and practices*.

Measures:

- Are the structures described comprehensively enough to be useful and transferable?
- Did you validate the interpretation of the *mechanisms and practices* (their form and function) with appropriate research participants?
- For each *mechanism* or set of *practices*, is it clear which roles are involved?

Principle H: Justificatory knowledge

Statement: A design approach must provide explanatory knowledge that links the *paradigm of value-creation* with its constructional components.

Rationale: The justificatory knowledge provides an explanation of why an artefact is constructed as it is, and why it works. Pointers to *some* kernel theories would provide researchers and practitioners with information that would be useful when comparing or combining approaches. It may also be possible that new theories are formulated, which cannot be traced to kernel theories.

Implications:

Define kernel theories on which the approach and its components are based, and on how they are related to different components of the approach.

Measures: No additional measures.

Principle I: Approach mutability

Statement: A design approach should clearly state possibilities for tailoring the approach, within the pre-defined design scope.

Rationale: Since the design approach may not have been demonstrated for multiple instances within the pre-defined design scope, the designer needs to identify possibilities for tailoring the approach. Design approaches need to address the dynamic nature of the enterprise and its environment, which is also a key concern within the EE discipline [5].

Implications:

Define foreseeable changes — i.e., approach constructs that will change, and the kinds of change that would be required.

Measures:

- Are conditions for changes described?

- Is it clear which parts could possibly change and to what they could change?
- Are possibilities for tailoring the approach defined to enable extension of the approach in future?

Principle J: Principles of implementation (conditional)

Statement: A design approach may incorporate guidance for implementing the approach.
Rationale: This principle is *conditional*, since the designer needs to consider the pre-defined design scope and decide whether additional advice would add value, e.g. additional advice may be required if the approach has been designed for the health industry. Even for other industries, the designer may need to provide additional advice to indicate how the approach should be used in specific contexts.

Implications:
- Define tailoring advice.
- Define advice regarding introduction into real-life settings.

Measures:
Does the advice for implementing the approach cover different settings within the scope, or is it at least clear about the scope to which it applies?

Principle K: Expository instantiation (optional)

Statement: A design approach may incorporate an instantiation.

Rationale: This principle is *optional*. A realistic implementation of an approach contributes to the identification of potential problems in its design, also demonstrating its worth. Even though the designer should have implemented the approach to evaluate its utility, the implementation results do not necessarily form part of the *construction* of the design approach.

Implications:
Report on a real-life implementation of the approach.

Measures:
- Does the implementation cover a case within the scope, and is it covering the main *mechanisms and practices* of the approach?
- Is the implementation specific enough to be illustrative?

5 CONCLUSION AND FUTURE RESEARCH

The proliferation of enterprise design approaches currently impairs the systematic growth of the EE knowledge base, since the many approaches do not necessarily explicate their conditional use, contextual prerequisites, and demarcated design scope. Drawing from existing theory on theoretical enterprise design approaches [6] and eight components for *design theory* in IS [38], we started a design research cycle to develop eleven *principles* to guide the approach developer. Future work will continue by demonstrating the principles used to develop a new enterprise design approach, and finally evaluating the usefulness of the principles.

The *principles* will not only be useful to approach developers in guiding them during the development and construction of the new/adapted approach, but should also be useful to the knowledge base guardian (research mentor/external examiner) when evaluating the contribution for relevance and rigour. Currently the principles only provide guidelines for the *construction* of a new/adapted enterprise design approach. De Vries and Berger [41] provide additional guidance on appropriate research methods for enterprise approach design, highlighting *action design research*, whereas Venable *et al*. [42] provide guidance for *evaluating* design science research.

6 ACKNOWLEDGEMENT

The researcher is grateful for the valuable inputs of the focus group participants who validated the clarity, validity and usefulness of the principles. Their suggestions were incorporated and are already reflected in Section 4.

REFERENCES

[1] **Kappelman, L.A.** 2010. *The sim guide to enterprise architecture*, Boca Raton: CRC Press.

[2] **Van Tonder, C.L. & Roodt, G.** 2008. *Organisation development: Theory and practice*, Pretoria: Van Schaik Publishers.

[3] **Hoogervorst, J.A.P.** 2009. *Enterprise governance and enterprise engineering*, Diemen: Springer.

[4] **Giachetti, R.E.** 2010. *Design of enterprise systems*, Boca Raton: CRC Press.

[5] **De Vries, M., Gerber, A. & Van der Merwe, A.** 2015. The enterprise engineering domain, in *Advances in enterprise engineering ix*, D. Aveiro, R. Pergl, and M. Valenta, editors. Switzerland: Springer, pp 47-63.

[6] **De Vries, M., Van der Merwe, A. & Gerber, A.** 2015. Extending the enterprise evolution contextualisation model, *Enterprise Information Systems*, doi: 10.1080/17517575.2015.1090629, http://www.tandfonline.com/doi/pdf/10.1080/17517575.2015.1090629.

[7] **Blowers, M.** 2012. *Hybrid enterprise architecture frameworks are in the majority*, [accessed 12 April 2012], available from: http://ovum.com/2012/03/22/hybrid-enterprise-architecture-frameworks-are-in-the-majority/.

[8] **Espejo, R. & Reyses, A.** 2011. *Organisational systems*, Berlin Heidelberg: Springer-Verlag.

[9] **Aier, S., Fischer, R. & Winter, R.** 2011. Construction and evaluation of a meta-model for enterprise architecture design principles, in *Wirtschaftinformatik proceedings 2011, paper 51*.

[10] **Beese, J., Aier, S. & Winter, R.** 2015. On the role of complexity for guiding enterprise transfomations, in *Advances in enterprise engineering ix*, D. Aveiro, R. Pergl, and M. Valenta, editors. London: Springer.

[11] **Hevner, A.R., March, S.T., Park, J. & Ram, S.** 2004. Design science in information systems research*, MIS Quarterly*, 28(1), pp 75-105, available from: http://wise.vub.ac.be/thesis_info/design_science.pdf.

[12] **Peffers, K., Tuunanen, T., Rothenberger, M. & Chatterjee, S.** 2008. A design science research methodology for information systems research*, Journal of MIS*, 24(3), pp 45-77.

[13] **Morgan, G.** 2006. *Images of organisations*, Thousand Oaks: Sage Publications.

[14] **Dietz, J.L.G.** 2006. *Enterprise ontology*, Berlin: Springer.

[15] **Gharajedaghi, J.** 2011. *Systems thinking: Managing chaos and complexity*, 3rd edition, Burlington, USA: Elsevier.

[16] **Beer, S.** 1979. *The heart of enterprise*, Chichester: Wiley.

[17] **Radzicki, M.J. & Taylor, R.A.** 2008. Origin of systems dynamics: Jay w. Forrester and the history of systems dynamics, in *U.S. Department of energy's introduction to systems dynamics*.

[18] **Lapalme, J.** 2012. *3 schools of enterprise architecture*, Enterprise architecture research forum meeting 30 January 2012, Editor.

[19] **Lapalme, J.S. & Guerre, D.W.** 2014. Enterprise-in-environment adaptation: Enterprise architecture and complexity management, in *A systematic perspective to managing complexity with enterprise architecture*, P. Saha, Editor. Singapore: IGI Global.

[20] **Dietz, J.L.G., et al.** 2013. The discipline of enterprise engineering*, International Journal of Organisation Design and Engineering*, 3(1), pp 86-114.

[21] **Ross, J.W., Weill, P. & Robertson, D.C.** 2006. *Enterprise architecture as strategy: Creating a foundation for business execution*, Boston: Harvard Business School Press.

[22] **Boar, B.H.** 1999. *Constructing blueprints for enterprise it architecture*, New York: J. Wiley.

[23] **Winter, R. & Fischer, R.** 2007. Essential layers, artefacts, and dependencies of enterprise architecture, *Journal of Enterprise Architecture*, 3(2), pp 7-18.

[24] **Lapkin, A.** 2008. *Gartner clarifies the definition of the term 'enterprise architecture'*, [accessed 9 September 2015, available from: https://www.gartner.com/doc/740712/gartner-clarifies-definition-term-enterprise.

[25] **Bernard, S.A.** 2005. *An introduction to enterprise architecture ea3*, 2nd edition, Bloomington: Authorhouse.

[26] **Schekkerman, J.** 2004. *How to survive in the jungle of enterprise architecture frameworks*, 2nd edition, Victoria: Trafford Publishing.

[27] **GAO.** 2006. *Enterprise architecture: Leadership remains key to establishing and leveraging architectures for organisational transformation*, [accessed 1 February 2008], available from: http://www.gao.gov/new.items/d06831.pdf.

[28] **Lapalme, J.** 2012. Three schools of thought on enterprise architecture, *IT Professional*, 14(6), pp 37-43.

[29] **De Vries, M.** 2012. *A process reuse identification framework using an alignment model. Phd thesis*, Pretoria: University of Pretoria.

[30] **The Open Group.** 2009. *Togaf version 9.0, enterprise edition*, [accessed 1 June 2009], available from: https://www.opengroup.org.

[31] **Lankhorst, M.e.a.** 2009. *Enterprise architecture at work: Modelling, communication and analysis*, 2nd edition, Berlin, Heidelberg: Springer-Verlag.

[32] **Frank, U.** 2014. Multilevel modeling, *Bus Inf Syst Eng*, 6(6), pp 319-337, available from: https://www.wi-inf.uni-duisburg-essen.de/FGFrank/documents/Zeitschriftenartikel/10.1007_s12599-014-0350-4.pdf.

[33] **Parmenter, D.** 2010. *Key performance indicators: Developing, implementing and using winning kpi's*, 2nd edition, Hoboken: John Wiley & Sons.

[34] **Dumbill, E.** 2012. *Planning for big data*, Cambridge: O'Reilly.

[35] **Johnson, G., Scholes, K. & Whittington, R.** 2006. *Exploring corporate strategy*, London: Prentice Hall.

[36] **Ostewalder, A., Pigneur, Y. & Clark, T.** 2010. *Business model generation*, 1st edition, Hoboken, NY: Wiley.

[37] **Van Aken, J.** 2004. Management research based on the paradigm of the design sciences: The quest for field-tested and grounded technological rules, *Journal of Management Studies*, 41(2), pp 219-246.

[38] **Gregor, S. & Jones, D.** 2007. The anatomy of a design theory, *J Assoc Inf Syst*, 8(5), pp 312-335.

[39] **Offermann, P., Blom, S. & Bub, U.** 2010. Proposal for components of method design theories: Increasing utility of method design artefacts, *Bus Inf Syst Eng*, 2010(5), pp 295-304.

[40] **Hoogervorst, J.** 2016. *On the realization of strategic success*, Antwerp: Antwerp Management School.

[41] **De Vries, M. & Berger, S.** 2016. An action design research approach within enterprise engineering, *Systematic Practice and Action Research*, available from:
http://link.springer.com/article/10.1007/s11213-016-9390-7.

[42] **Venable, J., Pries-Heje, J. & Baskerville, R.** 2016. Feds: A framework for evaluation in design science research, *European Journal of Information Systems*, 25(1), pp 77-89, available from:
http://www.palgrave-journals.com/ejis/journal/v25/n1/full/ejis201436a.html.