Chapter 26
Conclusion and reflections

Abstract In this Chapter, we look back on the results presented in this book. As mentioned at the start of this book, the field of ACET is rather rich and diverse. As such, this book could only provide a humble beginning towards the creation of a more complete understanding of ACET and the development of an integrated set of instruments supporting ACET in practice. In this Chapter, we therefore critically reflect on our experiences with the development of "large scale" design artefact, such as an integrated method for ACET, as the research programme set out to do.

We will conclude with a list of suggestions for possible follow up research in the domain of ACET. This list combines both the reflection on our experiences in the development of large scale design theory, as well as opportunities for further research on the level of the specific components as presented in this book.

This Chapter was authored by:

Hend*erik* A. Proper, Stephan Aier and Robert Winter

26.1 Introduction

In this Chapter, we will start (in Section 26.2) by briefly looking back on the results presented in this book. As mentioned at the start of this book, in Chapter 1, the field of ACET is rather rich and diverse. This is reflected in the results presented in this book, which originate from a broad research programme on ACET, involving four applied research projects. As such, this book could only provide a humble beginning towards the creation of a more complete understanding of ACET and the development of an integrated set of instruments supporting ACET in practice.

As most of the research conducted in the context of the ACET research programme, involved a design science approach (March and Smith, 1995; Hevner et al., 2004; van Aken, 2004; Peffers et al., 2007; Venable et al., 2012; Peffers et al., 2012;

Gregor and Hevner, 2013; Wieringa, 2014; Venable et al., 2016), or at least covered some stages of design science, it is certainly relevant to reflect on the experiences gathered in the context of the ACET project. Even though this book only reports on an initial understanding of ACET and an initial set of components (of an integrated set) of instruments supporting ACET in practice, the ambitions at the start of the research programme were higher. It was, indeed, the ambition to develop an integrated design theory for ACET. This provides a good reason to, in Section 26.3, critically reflect on our experiences with the development of "large scale" design artefact, such as an integrated method for ACET, as the research programme set out to do.

Finally, we conclude with a list of suggestions for possible follow up research in the domain of ACET. This list combines both the reflection on our experiences in the development of large scale design theory, as well as opportunities for further research on the level of the specific components as presented in this book.

26.2 Summary of results

In Part I, an analysis was provided of the current state of the ACET practice. This was used as an inspiration for a more detailed exploration of the challenges facing ACET from a more theoretical perspective. This, in particular, resulted in explorations of:

- the types of change and transformations that may occur (Chapter 6),
- enterprise transformation from the perspective of social systems (Chapter 7),
- the role of sub-cultures in the coordination of transformations (Chapter 8),
- the role of a *use perspective* for architectural coordination (Chapter 9),
- the role of stakeholders and a strategy to better engage them (Chapter 10),
- the information information requirements for doing ACET (Chapter 11),
- the sustainable organisational establishment of doing ACET (Chapter 12),
- the landscape of modelling languages for ACET (Chapter 13),
- the role of architecture principles (Chapter 14), and
- the motivation and rationalisation of architectural decisions (Chapter 15).

In Part III, we discussed a collection of components for a possible design theory for ACET, which were "harvested" from the work of the individual researchers in the programme. Collectively, these components aimed to address the challenges as they had been identified in Part I from an empirical perspective, and Part II from a more theoretical and / or literature perspective. The harvested components included:

- definitions of the key concepts underlying ACET (Chapter 16),
- a reference framework of services needed for doing ACET (Chapter 17),
- a strategy to engage stakeholders in decision-making during ACET (Chapter 18),
- guidelines to use models as boundary objects (Chapter 19),
- a reference model for the information requirements for ACET (Chapter 20),

- an approach for component-based modelling language engineering (Chapter 21),
- guidelines for the semi-formal definition of architecture principles (Chapter 22),
- a framework to explicitly capture architecture design decisions (Chapter 23),
- a formal reasoning system for architecture design decisions (Chapter 24), and
- strategies for situation specific adaption of ACET (Chapter 25).

26.3 Reflections on the development of large scale methods

As stated before, the ambitions at the start of the ACET programme were to develop an integrated design theory for ACET. The ideal would have been to establish a validated topology of ACET components, provide support for integrating these components and concrete, situation specific, configuration rules. As we will discuss below, this soon turned out to be too ambitious.

26.3.1 Change of programme strategy

Each of the different projects involved in the ACET programme conducted explorations of different aspects of the ACET problem space (see Part II). Soon, the heterogeneity and multifacetness of these aspects showed that the development an integrated design theory for ACET would be too ambitious. A choice had to be made between the creation of a "superficial" overall method for ACET, or a, for the moment, set of disconnected and partial, yet well founded, components towards a more comprehensive method for ACET. This resulted in a change of strategy, where the research efforts were compartmentalised, in the sense that each of the involved researchers focussed on a specific (set of related) aspects, with the aim to develop an initial explanatory theory covering the aspect (see Part II), and possibly a method component / fragment meeting the needs of covering that aspect (see Part III).

The work, as conducted by the individual researchers, can also be said to correspond to a set of focussed experiments towards the establishment of clearer requirements on an integrated ACET design theory. This can be seen as a strategy to deal with, what van Aken and Nagel (2004), call the "fuzzy front end" of design-oriented research. In the case of ACET, the potential "fuzziness" is exacerbated by the fact that ACET would require the development of a large scale method, to cover all work needed in architectural coordination of enterprise transformations. In the remainder of this Chapter, we aim to reflect on this in more depth.

26.3.2 A method as a design theory

We see a method as capturing guiding / prescriptive knowledge on how to do / organise certain tasks. As such, it corresponds to a *reference model* that guides / directs the planning and / or execution of tasks in specific situations. This could, e.g., be a reference model defining a flow of work, or a set of rules / principles that should be upheld, or defining a (modelling) language to be used.

When a method is further enriched with guidelines to tune the method towards situations at hand, one may refer to the method as a *situational* method. This corresponds to the notion of *artefact mutability* as defined by Gregor and Jones (2007), where in this case, the method is the artefact. Note: the guidelines for situational adaption can be seen as a "situational adaption method", and as such, would be defined by its own reference model.

A method, as a reference model for action, should provide a clear identification of what its *claimed working* is, in terms of what would be achieved when indeed following the method, in a context that meets given pre-conditions. For example: (timely) availability of inputs needed inputs, situational factors such as the abilities of actors involved, etc. In terms of Gregor and Jones (2007), this *claim* leads to a number of *testable propositions*, which, can also be said to correspond to the *theory* as brought forward by the method. In other words, the *design theory* of the method. As Wieringa (2014) puts it "*design science studies the interactions between an artefact and its context. We call theories about (artefact × context) design theories.*"

In the design / creation of a method, different styles of reasoning can be used. It may be derived *deductively* from other (design) theories. This deduction corresponds to the *justificatory knowledge* as discussed by Gregor and Jones (2007). The resulting method, can / should then be validated in real world situations. This can potentially also lead to refinements on the pre-conditions under which the method will produce its claimed effects.

A method can also be derived *inductively* from a broad corpus of real world cases, by observing patterns, and hypothesising over these. In this case, the *justificatory knowledge* would pertain to the empirical evidence found in the corpus of cases, and the observed patterns.

When there is only a limited number of cases available, one may resort to *abductive* reasoning, looking for patterns that at least match the available cases. Further evidence can then be sought by further validation on new cases and / or experiments.

For the development of a design artefact in general, and a method in particular, several other theories may be useful, providing more means to develop the *justificatory knowledge* underpinning the method's design theory. One may, for example, use and / or develop theories that capture propositions regarding the domain *in* and / or *on* which one aims to use the artefact / method.

26.3.3 Complexity and uncertainty for the use of methods

The *Information Services Procurement Library* (ISPL) (Franckson and Verhoef, 1999; Proper, 2001), identifies different situational factors that should be taken into account when defining project plans, and risk mitigation strategies in particular. These situational factors are classified into four classes along two axes: *complexity* versus *uncertainty* and *target domain* versus *project domain*, where *target domain* refers to the domain in which the project / service is to make a change, while *project domain* refers to the project / service itself.

The effort of developing a method (such as ACET) also has to deal with a number of complexities and uncertainties, covering both the target domain of the method, i.e. the domain in which it should have its (claimed) effect / result, and the method itself. This includes:

Target domain uncertainty – dealing with uncertainties about properties that hold *in* the method's target domain, the specific aspects / parameters involved, the understanding of the actual class of problems that the method aims to contribute to, the stakeholders involved, their specific attitudes, etc.

Target domain complexity – involving the complexity of the (properties of the) method's target domain, complexity / heterogeneity of the specific aspects / parameters involved, complexity and variety of the class of problems the method aims to contribute to, heterogeneity of the (stakes of the) stakeholders involved, etc.

Method domain uncertainty – dealing with uncertainties about the precise working of specific concepts that can be used in the construction of the method, uncertainty about the validity of pre-conditions, etc.

Method domain complexity – pertaining to the complexity of the method itself in terms of the complexity of its reference model, complexity of the guidelines towards situational adaptation and implementation, complexity of the definition of its pre-conditions, etc.

It is important to realise that when there is a large variety in the class of problems in which the method aims to "do its work", then the *law of requisite variety* (Ashby, 1956) implies that the complexity of the method itself should be high enough to meet this variety. Either in terms of a high complexity / variety of the reference model that defines the method, or in terms of the guidelines defining its situational adaption. Note: this is separate from the *inherent* complexity of the "work" that the method aims to get done, which will also need to be reflected in the method itself.

26.3.4 Research methodological guidance

In the development of a method, different strategies can be used to deal with complexities and uncertainties. In the case of the ACET programme, the observed (Part I) uncertainty of the target domain, exacerbated by the complexity of the target domain, led to the conclusion that a series of *pre-studies* was needed into the different factors involved (Part II), as well as experiments with possible compoments of a method (Part III). This resulted in a reduction of the uncertainties, and an increased understanding of the complexities involved.

The strategy followed by the ACET programme can be seen as a strategy to deal with, what van Aken and Nagel (2004), call the "fuzzy front end" of design-oriented research. In the case of ACET, the potential "fuzzyness" of this front-end is exacerbated by the fact that it would would require the development of a large-scale method to cover all work needed in architectural coordination of enterprise transformations, i.e. resulting in a high complexity on top of all the uncertainty.

Just as Franckson and Verhoef (1999); Proper (2001) define strategies and heuristics on how to deal with complexity and / or uncertainty with regard to information systems related projects and services, one would need similar guidance for the development of methods. Methods being design theories, one would expect that the field of design science would provide such guidance. Regretfully, however, we did not find much guidance on literature.

At the same time, one can certainly observe the existence of large-scale methods, such as TOGAF (The Open Group, 2011), ISPL (Franckson and Verhoef, 1999), ITIL (Axelos, 2015), etc. These are, however, typically "best-practice" based methods lacking rigorous validation, justificatory knowledge, and / or testable propositions.

Another way to deal with the uncertainties and complexities facing the development of a method, is to enable early validations of method components, so as to obtain early feedback. In general, designed artefacts, such as methods, should be evaluated with regard to their ability to solve the addressed design problem (March and Smith, 1995). Traditionally, the predominant approach in design science is that of evaluating artefacts once they have been designed ready for use (see e.g. Hevner et al., 2004; Peffers et al., 2007). However, to enable feedback loops as early as possible, Venable et al. (2012) proposed the notion of *pre* and *post* artefact evaluation, where pre artefact evaluations involve evaluations of artefacts *before* they are built and post artefact evaluations are evaluations of artefacts *after* they have been built. That first differentiation of evaluation perspectives specifically targets the fact that feedback loops should be applied as early as possible, and not only after the design has been completed.

In a further differentiation, Venable et al. (2016) later distinguish naturalistic and artificial evaluations as well as formative and summative evaluations. This allows for the design of multi-step evaluation strategies, that provide many feedback opportunities, reflecting the changing character of the artefact as it matures in the design process. The distinction between evaluation characteristics reflects however more the "how" of evaluation than the "what". As the artefact matures during the design process, different aspects of design knowledge dominate that should be matched by corresponding evaluation episodes.

26.3.4.1 Multiple levels of detail in method design

An interesting contribution on the challenge of large-scale design artefact development, is made by Daeuble et al. (2015). They provide a strategy for the (incremental) development of large-scale design artefacts, that suggest to split a design artefact into smaller components based on a *segmentation framework*.

While being a potential relevant way to deal with the *complexities* that faced the ACET programme, it would not have solved the *uncertainties* which the programme had to deal with. We would still have found ourselves at the fuzzy front-end of design science. Nevertheless, towards follow up research towards a more integrated ACET method, it could make sense to use a *segmentation framework* to structure further research efforts.

Based on our joint past experiences in the development of methods, we would suggest, for now, to consider the following levels of detail in such segmentation framework for methods:

Abstract structure – An overall perspective on roles, responsibilities, and tasks involved, types of deliverables to be produced, temporal dependencies between tasks, etc.
This would lead to one method component defining the overall "rhythm", with possible guidelines for situational adaptation, possibly in terms of general principles.

Concrete structure – A further operationalisation of the operational perspective in terms of concrete deliverables, and specific requirements on modelling languages to be used in producing these deliverables, approaches and techniques to be used, etc.
This is likely to lead to a set of alternative method components, whose relevance depends on the situational contexts, where each resulting component may have additional rules to tune things to a situation at hand.

Tools & techniques – A collection of (small-scale) methods, approaches and techniques to support / direct the creation of the deliverables as identified in the (large scale) method. For example, an approach for stakeholder management, modelling languages used to represent deliverables, collaborative decision-making, etc.

Reference material – Reference models, partial models / designs, design patterns, etc. that can be used as starting points, or guidelines, towards the creation of actual deliverables.

In terms of this suggested segmentation framework, the results as discussed in this book, can be positioned as show in Table 26.1. Note that we certainly do not claim that the elements listed in Table 26.1 are an integrated method for ACET.

Abstract structure	Exploration	- Types of change and transformations that may occur (Chapter 6) - enterprise transformation from the perspective of enterprises being social systems (Chapter 7) - The role of sub-cultures in the coordination of enterprise transformations (Chapter 8) - The role of a *use perspective* for ACET, in particular the *use* of the created architectural artefacts (Chapter 9) - The engagement of stakeholders during ACET, (Chapter 10) - How a sustainable discipline of doing ACET can be established in an organisation (Chapter 12)
	Components	- Definitions of the key concepts underlying ACET (Chapter 16) - A reference framework of services needed for doing ACET (Chapter 17 - An overall strategy to engage stakeholders in decision-making during ACET (Chapter 18) - Guidelines to use models as communication devices (Chapter 19)
Concrete structure	Exploration	- The information information requirements for doing ACET (Chapter 11) - The landscape of modelling languages for ACET (Chapter 13)
	Components	- Strategies for situation specific adaption of ACET (Chapter 25)
Tools & techniques	Exploration	- The role of architecture principles (Chapter 14) - The motivation and rationalisation of architectural design decisions (Chapter 15)
	Components	- A reference model[1] for the information requirements for ACET (Chapter 20) - A method for value-based componential language engineering to express the models needed during ACET (Chapter 21) - Guidelines for the semi-formal definition of architecture principles (Chapter 22) - A framework to explicitly capture architecture design decisions (Chapter 23) - A logic-based framework to formally reason on architectural design decisions (Chapter 24)

Table 26.1 Mapping of results presented in this book

26.4 Suggestions for future research

We conclude this Chapter with a list of suggestions for possible follow up research in the field of ACET, and the further development of design science research in general. This should only be considered as a source of inspiration. It is in no way intended as a formal research agenda. We certainly plan to follow up on some of these suggestions ourselves.

26.4.1 Socio-cultural context of ACET

The challenges regarding the social and cultural aspects as identified in Chapter 6, 7 and 8, have been met only partially by the components discussed in Chapter 17, 18 and 19. Many more challenges remain, for example, with regards to:

- A deeper understanding is needed of the nature of the social and cultural "forces" that may influence (initiate, strengthen, hamper or derail) change in organisations, and possible "indicators" and "levers" to observe and mitigate these forces.
- Different strategies, contingent on the specific social and (sub-)cultural contexts, need to be developed, to indeed achieve needed architectural coordination. This includes the elaboration collaborative decision-making strategies and stakeholder management when dealing with social complexity.
- When using an explicit method for "doing" ACET, this method should, of course, be *institutionalised*. As such, this is also an organisational change, being exposed to the similar social and / or cultural forces as the actual architectural coordination of an enterprise transformation has to deal with. How these forces influence the use and uptake of a method for ACET (and its elements / components) needs further investigation.

26.4.2 Enterprises are in motion

In line with Rouse (2005), this book considered enterprise transformation as being top-down initiated, pre-meditated, and fundamental change. There are, however, many more changes happening in organisations. These changes are also likely to (gradually) lead to an architectural impact, and should, therefore, also be included in architectural coordination. Examples of such types of changes include:

1. *Organisational drift*, dealing with gradual misalignment between an enterprise's original intent (strategy, business model, operating model, etc.), and the actual operational activities (Mandis, 2013).
2. *Self-organisation* as can be found in the context of self-steering teams (Prakken, 2000; Achterbergh and Vriens, 2009)

3. *Bricolage & emergence* may, as argued by Ciborra (1992), provide enterprises with strategic advantages in terms of the bottom-up evolution of socio-technical systems that will lead to outcomes that are deeply rooted in an enterprise's organisational culture, and hence much more difficult to imitate by others.

On might, therefore, even go as far as to say that enterprises are *in motion* (Proper, 2014), where the word *motion* refers to "*an act, process, or instance of changing place*" (Meriam–Webster, 2003). Sometimes this indeed involves a top-down and pre-meditated enterprise transformation, but there is more change happening in an enterprise than transformations.
This leads, a.o., to the following challenges for architectural coordination:

- How does change in organisations occur? Especially when we do not only consider top-down initiated, and premeditated, enterprise transformations. What are the needs for, and potential role of, architectural coordination in organisational change.
  As Magalhães and Proper (2016) argue, this also requires a stronger and deeper interaction between organisational sciences, management sciences, and enterprise engineering / architecting.
- Strategies are needed to "detect" bottom-up / emergent changes as they occur in organisation. Techniques such as process mining van der Aalst (2011), software cartography Sousa et al. (2011, 2009) and enterprise cartography Sousa et al. (2011) are first examples of such techniques.

26.4.3 Enterprise architecture modelling

Chapter 13 explored some of the challenges in regards to models and modelling in the context of ACET. Models can capture crucial information regarding an enterprise's current architecture, its possible future architecture(s), as well as contextual information. In Chapter 11, the landscape of possibly relevant information was explored, while Chapter 9 explored the need for a use perspective on the deliverables (including models) produced in an ACET context. Taking this as input, Chapter 19 provided guidelines for the use of models as boundary objects, bridging different groups of stakeholders. Chapter 20 provided a reference framework for information requirements for ACET, this providing a first insight into the broad landscape of information that can be captured by models. Finally, Chapter 21 provided a general approach to reason about the added value of models in relation to its planned use.
Remaining challenges include:

- Translation of the information requirements, as identified in Chapter 20 to an integrated landscape of modelling languages needed to capture the needed information.
  Standards such as ArchiMate (Iacob et al., 2012) do indeed provide a large coverage of this landscape. However, the continuous extension of the standard, and

the plethora of suggested extensions and complementary modelling languages, indicates that more work is necessary (Bjeković et al., 2012).
In addition, the advent of domain specific modelling languages, also indicates that a one-size-fits all general purpose language may not be the right / complete answer (Bjeković et al., 2012, 2014).
- Further elaboration of the guidelines to use models as boundary objects, in particular in relation towards different stakeholders and collaborative decision-making. This should also involve a clearer positioning of the role concepts of views and viewpoint from the IEEE (2011) standard.

26.4.4 Enterprise architecture principles

Chapter 14 explored the role of enterprise architecture principles in ACET, while Chapter 22 provided a strategy to better capture architecture principles in a semi-formal language, as well as providing potential evaluation strategies to assess the compliance to architecture principles.
Remaining challenges include:

- Further elaboration of the linkage between the concept of enterprise architecture principles and regulations, in particular where it concerns the formulation and capturing of motivations.
- Strategies to enable, in a not intrusive way, the formalisation (in semi-formal languages) of architecture principles, as part of their formulation process.
- Improvement of hybrid (human and machine) based assessment of the compliance of designs and / or actual implementations to architecture principles.
- Strategies to inspire / guide (architectural) design processes based on enterprise architecture principles. In other words, not just assessing if a “finished” design complies to a set of principles, but pro-actively aid / influence architects and designers during the actual design processes.

26.4.5 Architectural decision-making

In Chapter 15 we explored the need for, and context of, making architectural decisions more explicit. This referred both to the process of decision-making, as well as the capturing of the actual decisions in terms of underlying motivations and trade-offs. Chapter 23, provided initial strategies to indeed make architectural design decisions more explicit, while Chapter 24 illustrated how formal logics can be used to reason over design decisions.
Some of the remaining challenges include:

- Integration into the collaborative decision-making processes as discussed in Chapter 18. It is in such processes, where important (high level) decisions are made, that provide input / context for more detailed design decisions later on.
- Integration with the formulation of architecture principles. The work reported in Chapter 23 focuses on decisions regarding architectural designs in terms of e.g. ArchiMate (Iacob et al., 2012). However, the selection / formulation of enterprise architecture principles, also involve a (design) decision with a rationale / motivation.

  It would thus be relevant to extend the work reported in Chapter 23 to make design decisions underlying architecture principles more explicit.
- Integration of enterprise architecture principles into architectural decision-making processes. This basically mirrors the point made above on the use of enterprise architecture principles to guide design processes.
- More research is needed on how to operationalise / leverage the design rationales, in other words, further increasing the return of capturing effort. This work would extend the work reported in Chapter 24.
- More experimentation is needed on how to capture rationalisation of design decisions in a more natural way (reducing the capturing effort). In other words, as a natural result from the decision-making process, where the capturing effort directly benefits the progress and quality of the decision-making process rather than hampering it.

26.4.6 Integrated method for ACET

As we mentioned before, the ambitions at the start of the ACET programme were to develop an integrated method for ACET. Such a design theory would have needed to involve a topology of ACET components, provide support for integrating these components and concrete situation specific configuration rules.

As discussed in Section 26.3, this book brings together the results of what can be said to be "experiments" in the "fuzzy front end" of design science. The framework as discussed in Subsection 26.3.4.1 provides a suggested structure in the further development large-scale methods, such as an integrated method for ACET.

The constructs identified in Chapter 16 provide a conceptual core of an ACET, while the framework as discussed in Chapter 17 provides a landscape map of the involved competences. An integrated method for ACET should provide guidance for the tasks of these competences, as far as they are related to architectural coordination.

Further research could therefore involve the "populating", in terms of relevant method fragments, of the generic framework discussed in Subsection 26.3.4.1 for ACET, while using the framework presented in Chapter 17 as a domain specific, i.e. ACET specific, landscape map.

26.5 Conclusion

This Chapter provided a summary of the results presented in this book, as well as critical reflection regarding the development of large-scale design theories, as we intended within the ACET programme. We finished with the identification of a series of topics for further research.