International Journal of Cooperative Information Systems
Vol. 20, No. 1 (2011) 83–136

DOI: 10.1142/S021884301100216X

World Scientific
www.worldscientific.com

DEFINITION AND VALIDATION OF REQUIREMENTS FOR COLLABORATIVE DECISION-MAKING IN ENTERPRISE ARCHITECTURE CREATION

AGNES NAKAKAWA[*,‡], PATRICK VAN BOMMEL[*,§]
and H. A. ERIK PROPER[*,†]

[*]*ICIS, Radboud University Nijmegen, P.O. Box 9010,*
6500 GL Nijmegen, The Netherlands
[‡]*A.Nakakawa@science.ru.nl*
[§]*pvb@cs.ru.nl*

[†]*Public Research Centre Henri Tudor, Luxembourg*
e.proper@acm.org

Gartner advises that for enterprise architecture development to be successful, it is vital that enterprise architects ensure effective communication and also form virtual teams that create and agree on enterprise architecture content. One of the ways to achieve this is to enforce Collaborative Decision Making (CDM) during enterprise architecture creation. Guided by Design Science, we are developing a method referred to as Collaborative Evaluation of (Enterprise) Architecture Design Alternatives (CEADA) to enable CDM during enterprise architecture creation. The method attempts to resolve challenges in enterprise architecting that are caused by ineffective collaboration between enterprise architects and organizational stakeholders. Requirements for CEADA have been defined based on the causality analysis theory, the generic decision-making process, enterprise architecture frameworks (and literature), and the CDM theory. In addition, Collaboration Engineering has been used to design a collaboration process to address these requirements. Models describing the requirements and the design of the collaboration process, have been evaluated using the analytical, experimental, and observational methods. This paper discusses the implications of findings from these evaluations and presents the validated requirements for realizing CDM in enterprise architecture creation. Thus, this research generally attempts to strengthen enterprise architecting guidelines with collaborative activities, so as to enable effective execution of collaboration-dependent tasks.

Keywords: Enterprise architecture; collaborative decision making.

1. Introduction

In a business environment, "*dynamics is the only constant while adaptiveness is the natural variable*".[1] Thus, for an organization to survive, it has to deal with challenges or changes in its business environment, and strategic management (i.e. strategy formulation, implementation, and evaluation) is one of the traditional approaches it may use.[3] However, strategic management often fails due to inadequate strategy implementation, rather than unforeseen circumstances.[50] For successful strategy implementation or execution, the definition of the strategy should be proper (i.e. specific, unambiguous, achievable, relevant, and actionable) and based on a thorough analysis of the impact of the intended change.[3] An effective approach to formulating such an explicit strategy definition and then successfully executing it, is enterprise architecture.[3,7] Precisely, enterprise architecture is the appropriate way for an organization to: deal with inflexibility in its business operations[6]; manage organizational changes[3]; master organizational complexity[7]; and effectively align all its aspects.[3,4,51,52]

Although there are several definitions of (enterprise) architecture, in this research our definition is derived from Ref. 7, where architecture is theoretically defined as "*the normative restriction of design freedom*", and practically defined as "*a consistent and coherent set of design principles*". Therefore, herein we perceive enterprise architecture as the normative means to direct enterprise transformations. These normative means are represented in the form of principles, requirements, and models of various views; and their role is to be a normative instrument during an intended organization transformation. Although enterprise architecture offers numerous benefits to organizations, reaping them essentially depends on the success of the enterprise architecture development process. The success of enterprise architecture initiatives is not hindered by technical reasons,[8] but instead by other issues that, in our view, mainly sprout due to ineffective collaboration between organizational stakeholders and enterprise architects during architecture development. This argument certainly excludes political issues which were reported in Refs. 8 and 53, as one of the key hindrances of enterprise architecture development as well as collaboration among organizational stakeholders. Section 2 discusses these issues and explains how addressing them has not been given enough attention.

On addressing majority of the problematic issues that arise during enterprise architecture development, Gartner,[54] e.g. advises that it is vital for enterprise architects to ensure effective communication with organizational stakeholders, and to form active teams that create and agree on enterprise architecture content. One of the ways to achieve this is to enforce Collaborative Decision Making (CDM) during enterprise architecture creation. However, literature (see discussion in Sec. 2) still lacks a standard (but flexible) operational outlook on successfully realizing CDM in enterprise architecture creation, without overdependence on professional facilitators. Therefore, with a focus on filling this gap, this research adapts the Design Science research methodology to develop a method, referred to as Collaborative Evaluation of (Enterprise) Architecture Design Alternatives (CEADA), that will enable CDM during enterprise architecture creation. The CEADA method will assist enterprise architects and organizational stakeholders to (effectively and efficiently) make joint (or collaborative) decisions when creating enterprise architecture. Hence, the method attempts to resolve challenges in architecture creation that are caused by ineffective collaboration between enterprise architects and stakeholders.

Drawing upon the causality analysis theory,[12] the generic decision-making process,[13] enterprise architecture approaches (and literature), and the CDM theory,[59] a theory for CEADA was formulated. The (evolving) theory for CEADA defines the requirements for CDM to be realized in enterprise architecture creation, whereas the (evolving) CEADA method aims to validate and address these requirements.

In other words, the requirements for CDM to be realized in enterprise architecture creation can be perceived as the requirements for the CEADA method. This method mainly constitutes a collaboration process that was designed using collaboration engineering. Herein, four models (collectively referred to as CEADA models) have been used to illustrate the theory for CEADA, the requirements for CDM in enterprise architecture creation, and the CEADA method. The CEADA models include the following:

(1) *Cause–effect analysis*, a model describing cause–effect and conditional relations of key variables in achieving CDM in enterprise architecture creation.
(2) *Requirements and steps*, a model describing the requirements for (and steps in) a collaborative decision-making process for enterprise architecture creation.
(3) *Decomposition of requirements*, a model showing the requirements (in (2) above) decomposed into explicit or SMART requirements.
(4) *Activities, patterns, and ThinkLets*, a tabular description of the agenda for the collaboration process that addresses the requirements for CEADA (definition based on Ref. 14).

In Design Science, evolving artifacts are evaluated using methods that are observational, analytical, experimental, descriptive, or testing-oriented.[10,11] In this research, the earlier versions of the CEADA models were evaluated using analytical, experimental, and observational methods. This paper discusses the implications of findings from these three evaluation phases of the CEADA models, and also presents the validated requirements for CDM to be achieved in enterprise architecture creation. Therefore, this paper specifically answers two questions: (1) How should a collaborative decision-making process for enterprise architecture creation look like? (2) In which activities during architecture creation does the enterprise architect need to deeply involve organizational stakeholders and how? Note that this research does not attempt to present another enterprise architecture framework, but instead focuses on strengthening enterprise architecting guidelines (defined by existing architecture approaches) with collaborative activities. Consequently, this will enable effective execution of collaboration-dependent architecture guidelines, i.e. those whose successful execution depends on effective collaboration between stakeholders and architects during enterprise architecture creation.

The remainder of this paper is structured as follows. Section 2 presents the research framework, while Sec. 3 presents the theory and the requirements, for CDM in enterprise architecture creation. Section 4 discusses how the CEADA method addresses those requirements, while Sec. 5 discusses the implications of findings from the analytical, experimental, and observational evaluation of the CEADA models. Section 6 concludes the paper.

2. Research Framework

This section gives the problem definition, research motivation, existing related work, gap analysis, and the research methodology.

2.1. *Problem definition*

When an organization has to create an enterprise architecture, there is no standard and explicit way of doing so in a collaborative setting (where enterprise architects involve organizational stakeholders). The need for effective collaboration between enterprise architects and key organizational stakeholders during enterprise architecture development has been emphasized by several researchers and practitioners,[1–3,6,8,9,15–17] but a standard and explicit approach to address it is lacking. Although current enterprise architecture approaches define guidelines for (and expected outcomes of) enterprise architecting, they lack detailed support for activities that require collaboration among actors during architecture creation. Actually many architecture projects have yielded abstract and complex models that are hardly usable in practice, implying that architecture development was not well integrated in the organization, and one of the causes for this is that collaboration between architects and stakeholders is often difficult.[6]

Collaboration among actors is mainly affected by their conflicting goals and concerns. For example, in an enterprise transformation, several stakeholders often have conflicting concerns regarding the transformation, yet they should agree on the direction of the transformation.[3] This requires the architect to identify all stakeholders' concerns, and then develop architecture views that reflect how all concerns will be addressed and the intended tradeoffs.[18] This places a demand on the methodology for designing architectures,[4] since it requires the architect to build relationships (with stakeholders) so as to create products (or views/models in this case) that appropriately address stakeholders' goals.[9] Such relationships are built through effective and efficient collaboration between enterprise architects and stakeholders. Thus, efficient collaboration is one of the main critical factors for enterprise architecting.[6]

Effective and efficient collaboration enhances a shared understanding of key organizational aspects between stakeholders and architects. This shared understanding (of processes, systems, and concerns) facilitates the negotiation,[19] which occurs during tradeoff analysis and evaluation of possible alternative ways in which the enterprise architecture can be designed. On the other hand, a shared understanding is the basis for achieving efficient collaboration between architects and stakeholders.[6] This two-way (i.e. conditional, and then causal) relation between collaboration (of actors) and shared understanding (among actors) of aspects pertaining to the organization, implies the need to deploy techniques that enable collaborative problem solving and decision-making during enterprise architecture creation.

Therefore, the motivation for this research is the need to support existing enterprise architecture approaches with CDM. This is possible if requirements for CDM in enterprise architecture creation are first explicitly defined and validated, and thereafter a method (i.e. CEADA) to address those requirements is developed. The method can be visualized as a plug-in for existing enterprise architecture approaches, to be used during architecture creation to enable effective execution of collaboration-dependent activities. Examples of existing architecture approaches in which CEADA could be used are The Open Group Architecture Framework (TOGAF), ArchiMate, Zachman, Integrated Architecture Framework (IAF), Dynamic Architecture Method (DYA), etc. In general, the definition of requirements for CDM in architecture creation, the development of CEADA to address those requirements, and the application of CEADA in existing architecture approaches, are all positive steps toward improving architecture creation and filling the gap, highlighted in Ref. 3, of the lack of scientific research on success factors for enterprise architecting.

2.2. *Related work*

In literature, existing efforts toward improving enterprise architecture creation through encouraging and enabling effective collaboration between enterprise architects and organizational stakeholders can be categorized into mainly three forms as follows: (A) Some efforts report drawbacks that should be avoided, or challenges that are encountered and need to be solved, during enterprise architecture development. (B) Other efforts report guidelines for improving enterprise architecture development. (C) Other efforts appear as approaches toward overcoming challenges in (or drawbacks of, or fulfilling guidelines for improving) enterprise architecture development. A mixture of these efforts is discussed below.

Under category A, issues confronting enterprise architecture development are mainly associated with political issues and project management and organizational weaknesses.[8] Examples of challenges (and drawbacks) that are associated with project management and organizational weaknesses, and arise due to ineffective collaboration between enterprise architects and stakeholders during enterprise architecture creation, include the following. First, the two main drawbacks in enterprise architecture development, i.e.: (1) choosing an ineffective leader as the lead enterprise architect; and (2) not involving business (or organizational) stakeholders in the architecture program.[54] Moreover, in Ref. 8 it is reported that during architecture development, it is often difficult to make stakeholders understand enterprise architecture models; and to make executives of organizations (used to making decisions in a reactive and proactive way) understand the role of an enterprise architect. This lack of stakeholder understanding and support arises when business stakeholders are not involved in developing the enterprise architecture; the architecture content is not being used in other projects in the organization; and management is not understanding the value of enterprise architecture.[54] What triggers these issues is the failure to explain the architecture (i.e. the process and content or products) in a simple business language (not technical) that stakeholders understand; and the failure to communicate enterprise architecture content with organizational stakeholders early and frequently.[61]

Under category B, the following are the guidelines given to improve enterprise architecture creation (development). Gartner,[54] gives the following as guidelines that architects can consider so as to avoid the drawbacks mentioned in category A above: (1) highly considering educating stakeholders and communicating the enterprise architecture so as to secure sponsorship from executives, and developing and executing an enterprise architecture communications plan that consolidates all stakeholder audiences in the organization; (2) getting involved with stakeholders to develop the business context so as to properly align IT and business goals; and (3) collaborating with stakeholders and forming virtual teams that will develop (create and agree) on enterprise architecture content so as to secure buy-in for the enterprise architecture. Moreover, in Ref. 6, in order to enable architects to understand and meet stakeholders' expectations, the following are reported as aspects that stakeholders consider to be essential in the architecture creation process: (1) having their (stakeholders') roles in the architecture function explicitly defined; (2) the willingness of architects to visualize with stakeholders and understand their goals and problems; (3) architects ensuring effective communication with stakeholders; and (4) architects having a long-term and realistic view about the realization of the organization's business and IT strategy. Note that attribute (4) is made possible through effective collaboration between stakeholders and architects, so that architects can conceptualize the "*as-is*" and "*to-be*" situations of the organization.

Common to categories A and B above is the need for effective communication, effective collaboration, and (shared) understanding (of aspects pertaining to architecture development) among organizational stakeholders and architects during enterprise architecture creation. On addressing these aspects, the following solutions (i.e. constituents of category C) have emerged over time. The Open Group developed a detailed method (i.e. TOGAF Architecture Development Method — ADM) that offers step-by-step guidelines for executing the architecture development process.[18] In addition, ArchiMate was developed to enable expression of business processes and their IT support in an easily understandable way (without low-level implementation details); visualization; analysis; communication; realization; and management of architectures.[4] ArchiMate also complements TOGAF by offering generic concepts that enable creation of consistent and integrated models that appropriately communicate TOGAF architecture views and enable communication and decision making across organization domains.[20]

Furthermore, attempts have been made to improve the way architecture principles are defined, since they have a significant role in enterprise architecture development. Architecture principles represent general requirements for a class of systems (in this case an enterprise)[5]; they guide the enterprise architecting process; and justify decisions made on architecture components.[21] In Ref. 23, an Enterprise Engineering Framework is presented that supports definition of principles in a specific and measurable way; effective and efficient assessment of the impact of principle(s); detection of possible contradictions in principles so that they can be adequately prioritized/clarified; and traceability in cause–effect analysis of aspects. Moreover, in Refs. 21, 23 and 24, approaches are presented for enabling formulation of architecture principles in a collaborative context, involving all key actors.

Although these approaches have been developed, they do not consider how other aspects or products of the architecture process can be realized in a collaborative context (involving architects and organizational stakeholders). Examples of such aspects include: creating shared understanding (of the organization's problem and intended solution) among stakeholders and architects; building consensus on the concerns, requirements, solution scenarios, and quality criteria that the enterprise architecture must address; collaboratively evaluating design alternatives for the enterprise architecture; and selecting the most appropriate design alternative for realizing the planned transformation.

Still under category C, attempts that are very close to addressing these issues (and those in categories A and B above) include the following. In Ref. 62, Business Scenarios are discussed as a technique for defining business requirements in the architecture development process, and it is advised that when gathering business requirements, approaches such as workshops, basic research, qualitative and quantitative analysis, surveys, calls for information can be used. In Ref. 60, it is discussed how communication between stakeholders and enterprise architects during architecture development can be perceived as a conversation that aims at achieving certain goals, but is affected by several parameters, and can be implemented using workshops, interviews, and e-mailing among others. Even efforts of architecture modeling using discrete event simulation to create a shared understanding among actors during architecture creation are reported in Ref. 15. Spewak[53] also gives a detailed description on how interviews can be used during enterprise architecture creation, and discourages the use of workshops because it is difficult to stick to the agenda when using them.

In category C, it is evident that aspects related to realizing CDM in enterprise architecture creation have only been superficially addressed. This is because there is still lack of a clear and standard (but flexible) operational perspective on how to realize CDM in enterprise architecture creation. As noted in categories A and B, most practitioners and researchers mainly give prescriptions of what should be done to improve enterprise architecture creation through encouraging and enabling CDM aspects, but details of how to implement or achieve the prescriptions are seldom given. Moreover, closer attempts to implement the existing prescriptions for realizing CDM in architecture creation (as noted in category C) appear in a generic form — remaining somewhat silent on some essential or operational details (e.g. an in-depth standard but flexible sequential or procedural specification) for realizing CDM in architecture creation. However, these attempts define useful concepts that can be adapted in this research to define the requirements for CDM to be realized in architecture creation, and to devise an approach that can address those requirements.

2.3. *Assessment of possible approaches*

Several approaches exist as remedies to collaborative problem solving and decision making in complex organization situations. This section gives an account of the relevant approaches to this research, thus giving a justification for approaches in the theoretical knowledge base of Fig. 1 (see Sec. 2.4).

2.3.1. *Causality analysis theory and the generic decision-making process*

In the causal analysis theory, explaining an event usually involves explaining its cause, and the analysis of the relationship between the cause and effect of events is essential to several formations of theory (i.e. processes, conjectures, models, frameworks, or body of knowledge).[12] Therefore, in this research causal analysis can be done so as to identify, examine, and understand the relationship between the core concepts underlying the requirements for CDM to be achieved in enterprise architecture creation. In addition, existing literature on enterprise architecture creation (e.g. in categories A, B, and C in Sec. 2.2) can be used as a starting point for analyzing the cause–effect and conditional relations in enterprise architecture creation.

Moreover, Simon[13] defined the following as the three key phases that constitute all types of decision making (and problem solving).

(1) Intelligence phase: This entails examining a situation or (problem) environment in order to identify conditions or scenarios that call for decision making or intervention or problem solving action.
(2) Design phase: This entails devising or formulating possible courses of action or possible decision alternatives for solving the identified problem or for improving the examined environment.
(3) Choice phase: This entails choosing a particular course of action or decision alternative from those identified or formulated.

Furthermore, according to Ref. 3, enterprise architecture creation generally involves the following activities: understanding the purpose of creating the architecture and the organization context; determining the (essential) deliverables of the architecture effort; monitoring organizational context and stakeholders; creating a shared understanding of organizational aspects pertaining to architecture development (among stakeholders and architects); designing the actual process of creating the architecture; determining impacts of various decisions made during architecture creation; and communicating architecture creation results. These activities essentially characterize enterprise architecture creation as a task that involves problem solving and decision making. Thus, Simon's generic decision-making process can be adapted along with the cause–effect analysis concept, in order to rationally structure the activities required for CDM to be achieved during enterprise architecture creation.

2.3.2. *Collaborative decision making*

According to Ref. 59, there are four broad approaches to decision making i.e.: (1) decision analysis, a prescription of how an analytically inclined individual should and could make wise decisions; (2) behavioral decision making, a description of the psychology of how ordinary individuals do make decisions; (3) game theory, a normative approach of how groups of ultrasmart individuals should make separate interactive decisions; and (4) negotiation analysis, an approach of how groups of reasonably bright individuals should and could make joint, collaborative decisions. The suitable approach in this research is negotiation analysis (also referred to as Joint Decision Making or CDM theory). This is because the CDM theory aims at yielding joint decisions, joint payoffs for all actors involved, reciprocal communication, and creativity.[59] Moreover, since game theory aims at maximizing individual pay offs for each player involved,[59] it is not relevant in this research.

2.3.3. *Soft systems approaches*

Systems engineering successfully solves well-defined technical problems, however the intricacies and confusions in fuzzy and ill-defined management situations (that involve human and cultural factors) dispel the application of systems engineering (i.e. hard systems thinking), and instead demand for soft systems thinking.[22] Enterprise architecture being a valuable asset (or means) for managing organizational transformations,[3,4,6,7,51,52] its creation process demands more of soft systems thinking than hard systems thinking. This is also justified by the nature of activities involved in architecture creation (see Sec. 2.3.1) and the nature of decisions made during enterprise architecture creation. According to the classification of types of decisions (i.e. structured, semistructured, and unstructured) defined in Ref. 55 the nature of decisions made during enterprise architecture creation are unstructured and (at times) semi-structured decisions (hence the need for soft systems thinking).

Moreover, in hard systems thinking, "system" is conceived as something outside ourselves that exists in the world, where different parts of the world are also conceived as systems that can be engineered.[22] On the other hand, in soft systems thinking, "system" is conceived as the process of how we deal with the world to solve a given problem, i.e. "*the process of inquiry into real-world complexity is itself a system for learning*".[22] Given this comparison and the summary of activities involved in architecture creation (see Sec. 2.3.1), it can be further asserted that the process of inquiry during enterprise architecture creation is a system in itself. This claim can as well be justified using the fundamental definition of a system. A system is a collection of interrelated objects (i.e. people, resources — which can be inputs or outputs) and processes that interact to achieve a given goal; it is surrounded by an environment and includes a feedback mechanism.[55] Likewise, during architecture creation, inputs are stakeholders' concerns and requirements; processes involved were listed in Sec. 2.3.1; outputs are the architecture creation results; the environment comprises of the organization's social and political aspects; and the feedback is obtained after implementing the architecture.

Thus, it is justifiable that the process of enterprise architecture creation involves soft systems thinking, or is systemic in nature. This implies the need for assessing the use or adaptation of soft systems approaches in realizing CDM during enterprise architecture creation. Soft systems approaches include: Strategies Assumption Surfacing and Testing (SAST)[32]; Social Systems Design (SSD)[33,34]; Social System Sciences (SSS)[35]; and Soft Systems Methodology (SSM)[22,65] among others. Most soft systems approaches (except for SSM) seem to focus more on strategy formulation and less on strategy execution (which requires having an explicit strategy definition). Yet enterprise architecture creation concentrates more on strategy implementation (particularly on devising an explicit or SMART strategy definition — see Sec. 1) and offers insights into future strategy formulation. This implies the need for assessing the use or adaptation of SSM in this research.

SSM initially comprised of seven stages.[65] Since its seven-stage model was unable to support its flexible use in practice, a four-stage model (with an amendment of analyzing cultural factors) was developed.[22] Discussions herein are based on the four-stage SSM model, which according to Ref. 22 includes the following activities:

(1) Investigating all aspects (including cultural and political) in a problem situation, and then representing them in a "Rich Picture" (so as to encourage a holistic and exploratory thinking about the situation); or performing "Analysis One Two Three" (so as to explicitly show the actual problem owners and the social and political factors in the situation).
(2) Formulating purposeful activity models that describe the desired situation. This is done by: (a) using "Root Definitions", i.e. short phrases that define the required transformation processes for realizing the desired situation; (b) performing "CATWOE" analysis, i.e. assessing **C**ustomers or stakeholders that will be affected by the transformation, **A**ctors who will perform activities in the transformation, **T**ransformation process(es) that are to be changed, **W**orld perspective on the transformation, **O**wner(s) controlling the transformation, and **E**nvironmental and external issues affecting the desired transformation; and (c) assembling transformation process.
(3) Debating the problem situation using the models in order to: (a) define desirable and (culturally) feasible changes that would improve the situation; and (b) seek and find accommodations between conflicting interests so as to take action.
(4) Taking action so as to realize the desired improvement.

Activities (1)–(3) in the SSM model reflect the three phases of Simon's generic decision-making process (see Sec. 2.3.1). Where activity (1) of SSM relates to the intelligence phase, activities (2) and (3(a)) relate to the design phase, and activity (3(b)) relates to the choice phase. Moreover, SSM clearly recommends the use of interviews in activity (1) to gather information that is vital to formulating the rich picture and performing Analysis One Two Three, thereby giving a high-level operational outlook on how to achieve activity (1).

In activity (2), SSM further clearly describes the use of root definitions, CATWOE analysis, and multilevel (or hierarchical) thinking. However, it is rather silent on how the problem solvers interact with stakeholders (or problem owners) to formulate, e.g. the root definitions that make up the purposeful activity models. In activity (3), it is still implicit how the debate is to be successfully conducted. The methods are not given that are to be used to successfully facilitate the debate when identifying desirable and culturally feasible changes in the activity models, and seeking accommodations between conflicting interests. Thus, SSM is somewhat silent on *how* to acquire a shared understanding among stakeholders (or problem owners), and how the evaluation of decision alternatives or alternative courses of action is done. These issues being key in the architecture creation process, there is need to deploy additional techniques if SSM is to be used in the pursue of realizing CDM during architecture creation. Despite the implicit issues, SSM concepts, i.e. Rich Picture, Analysis One Two Three, Root Definitions, CATWOE analysis, and multilevel thinking, can be adapted in this research (see Secs. 3 and 4).

2.3.4. *Collaboration engineering*

A team that uses a Group Support System (GSS) is often more productive than a team that does not use it (or a conventional team), e.g. the former is characterized by even and full participation during interactions, compared to what happens (e.g. fearing to speak, domination, poor grasp of the problem) in the latter.[42,72] A GSS is a computer-based environment that supports interactive, concerted, and coordinated efforts in a team in order to enable it to collaboratively complete a given task.[72] GSSs help to improve the quality of group decision making in task-oriented group processes, i.e. processes that are complex and yet involve multi-actor, multicriteria, ill-structured, and evolving dynamic problems that require actors to cooperate and conflict in order to define and solve them.[70] GSSs includes (1) Problem Structuring Methods (PSMs), also referred to as model-based or model-driven approaches; and (2) Electronic Meeting Systems (EMSs), also referred to as technology-based or technology-driven or workstation approaches.[73]

On the one hand, a PSM enables one to represent a given situation using a model(s) in order to enable stakeholders in that situation to explicitly discuss their (complex) impasse; jointly define their feasible problem or matters associated with it; and agree on ways of (partly) resolving it.[49] On the other hand, an EMS supports task-oriented collaborative efforts in (face-to-face) meeting processes that involve planning, problem solving, decision making, deliberation, generating and evaluating alternative courses of action, negotiation, and building consensus.[37]

When using a PSM, the presence of a professional facilitator is compulsory,[36] and it is difficult to evaluate its performance because its support depends on the uniqueness of the situation at hand.[73] However, collaborative tasks may consist of a combination of several unique (but interrelated) meeting processes, which require flexible, quick, and efficient facilitation support.[37] An example is the nature of collaborative problem solving and decision making that is required during enterprise architecture creation — it varies across organizations and yet involves various types of meeting processes.[58] Hence, the need for flexible facilitation. Flexible facilitation support can be offered by EMSs (e.g. MeetingWorks, GroupSystems, Facilitate.com, etc.) since they are equipped with capabilities for enhancing effectiveness and efficiency of (and user satisfaction with) group meetings.[37]

Despite the numerous benefits of EMSs (or GSSs in general), they have not been widely adopted by organizations.[27,37] Moreover, lab and field results from GSS-related research are ambiguous and at times conflicting.[42] This is (among other factors) due to the following two factors: (1) GSSs have a high conceptual load that requires one to first understand the intended effect of GSS functionalities for the user, and so organizations resort to hiring (or training) professional facilitators in order to be able to successfully use the technology.[27,42] (2) GSSs are facilitator-driven, and yet maintaining professional facilitators is not easy due to economic and political issues faced by organizations.[27] These adoption issues call for GSS researchers to ensure that GSS results are more replicable and predictable by transferring their research experiences into organizations so as to provide clear practical support for GSS.[42] This can be realized through diverting (some) research attention to ThinkLets — a concept from collaboration engineering,[14,27,38,40,42,43] or to Group Model Building (GMB) scripts — a concept from GMB.[28–30,36]

Collaboration engineering is an approach used to design reusable collaboration processes that yield predictable success for recurring mission-critical tasks, and the deployment of such processes for execution by practitioners rather than professional facilitators.[38–40] Since professional facilitators are an additional cost to organizations, collaboration engineering helps to transfer relevant facilitation skills, knowledge of GSSs, and group dynamics to practitioners.[27] On the other hand, GMB is an approach used in strategic decision making to create new insights into strategic issues of a problem and enable stakeholders to acquire a shared reasoning about a problem, improve communication among the stakeholders, reduce conflicts, and reach a consensual agreement.[41] GMB is "*a system dynamics model-building process in which a client group is deeply involved in the process of model construction*".[30] Although GMB is a PSM[36] and therefore is a facilitator-driven approach,[29,36] it can be implemented using "*scripts*".[28] Scripts are useful for structuring the design of specific GMB sessions.[30] Scripts are pieces of small group processes that can be suitably sequenced for successful execution of a collaborative task.[28]

Thus, collaboration engineering can be used to design collaboration processes,[27] and GMB scripts can be designed,[28,29] that can be effectively executed in the absence of a professional facilitator.[27–29] In this research, this implies that: (1) a collaboration process can be developed to address some requirements for realizing CDM during enterprise architecture creation; and/or (2) a GMB script can also be developed to address some requirements for realizing CDM during enterprise architecture creation. However, the latter in this research is limited due to the flexible facilitation aspect (see third paragraph of this section), while the former is discussed in Sec. 4.

2.3.5. *VPEC-T and ASE*

In Ref. 56, the VPEC-T (i.e. Values Policies Events Content Trust) framework is described to enable effective communication between business stakeholders and IT professionals during Information Systems (IS) development. Although VPEC-T framework is generic to IS development, its concepts can be adapted to improve communication in enterprise architecture creation, by providing a standard vocabulary between stakeholders and architects. Moreover, in Refs. 47 and 48, the Accelerated Solutions Environment (ASE) is presented as a method used in practice to create commitment among critical stakeholders, and to gain buy-in at the start of a business transformation initiative. Although ASE covers several business transformation initiatives, its concepts can also be used in enterprise architecture creation to secure stakeholders' commitment and approval.

2.4. *Design Science research methodology*

This research is guided by Design Science, a problem-solving paradigm that facilitates the creation and evaluation of artifacts for solving identified organizational problems[10]; or for offering opportunities through which practice can be improved, even before practitioners identify any problem with their way of working.[25] Design Science artifacts are created using existing theories, frameworks, instruments, constructs, models, methods, and instantiations; and evaluated using observational, analytical, experimental, descriptive, or testing methods.[10,26] Section 2.1 defines the organizational problem or challenge this research is addressing, i.e. the realization of CDM in architecture creation (see left part of Fig. 1). As shown in the middle upper part of Fig. 1, the evolving artifact in this research (to address the problem) is the CEADA method.

The right part of Fig. 1 shows the contents of the knowledge base that have been relevant in developing CEADA. The knowledge base in Design Science refers to the collection of scientific foundations and methodologies (i.e. theories, frameworks, instruments, constructs, models, methods, and instantiations) that are relevant in the creation and evaluation of an artifact.[10] The right part of Fig. 1 shows the "kernel" theories for this research and other approaches that are relevant for the flexible application of CEADA. The justification for the contents in the knowledge base has been given in Secs. 2.2 and 2.3. Sections 3 and 4 discuss how the contents of the knowledge base were adapted to formulate the CEADA models. The middle lower part of Fig. 1 shows the relevant evaluation methods in this research, however this paper discusses findings from only the analytical, experimental, and observational evaluation of CEADA (see Sec. 5).

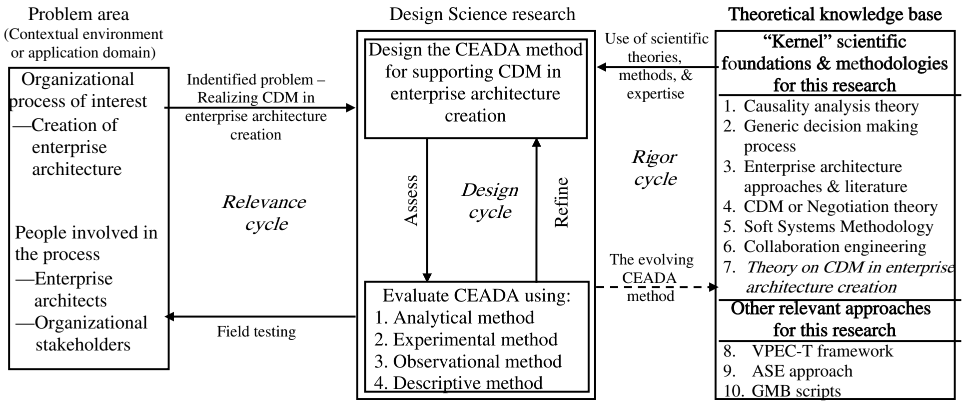

Fig. 1. Adapted methodology — Design science (based on Refs. 10 and 26).

3. CDM in Enterprise Architecture Creation

This section discusses the theory and the requirements for realizing CDM in enterprise architecture creation.

In Ref. 60, guidelines are presented for (1) formulating conversation strategies; and (2) selecting architectural knowledge goals and conversation techniques for communicating architectural models. Figure 2 is a conceptual representation of the adaptation of these guidelines into this research. Communication in system development involves conversations in which knowledge about the intended system (and its development) is created and shared among actors.[60] These conversations are affected by several situational factors,[60] which are denoted as SFj in Fig. 2, where $j = \{1, 2, \ldots, n\}$. Prior to the conversation, it is vital to determine the architectural knowledge goals or states that the conversations will aim to achieve, i.e. creating new knowledge, agreeing to it, or committing to it.[60] As shown in Fig. 2, in this research these knowledge goals are pointers to the requirements for realizing CDM in conversations on enterprise architecture creation. To achieve the architectural knowledge goals, a conversation should follow a strategy.[60] As shown in Fig. 2, in this research such a strategy can be conceived as the method that can address the requirements of CDM in enterprise architecture creation (i.e. CEADA). Furthermore, according to Ref. 60, a conversation strategy should articulate the following:

(1) An execution plan showing the execution order of subconversations that will achieve subgoals that contribute to achieving the main goal. Figure 2 shows that in this research the execution plan was designed using collaboration engineering.
(2) The languages that will be used when describing aspects in the conversation. Figure 2 shows VPEC-T as one of the adapted languages in this research.
(3) The type of media that will be used during the conversations. Figure 2 shows the adapted media in this research.

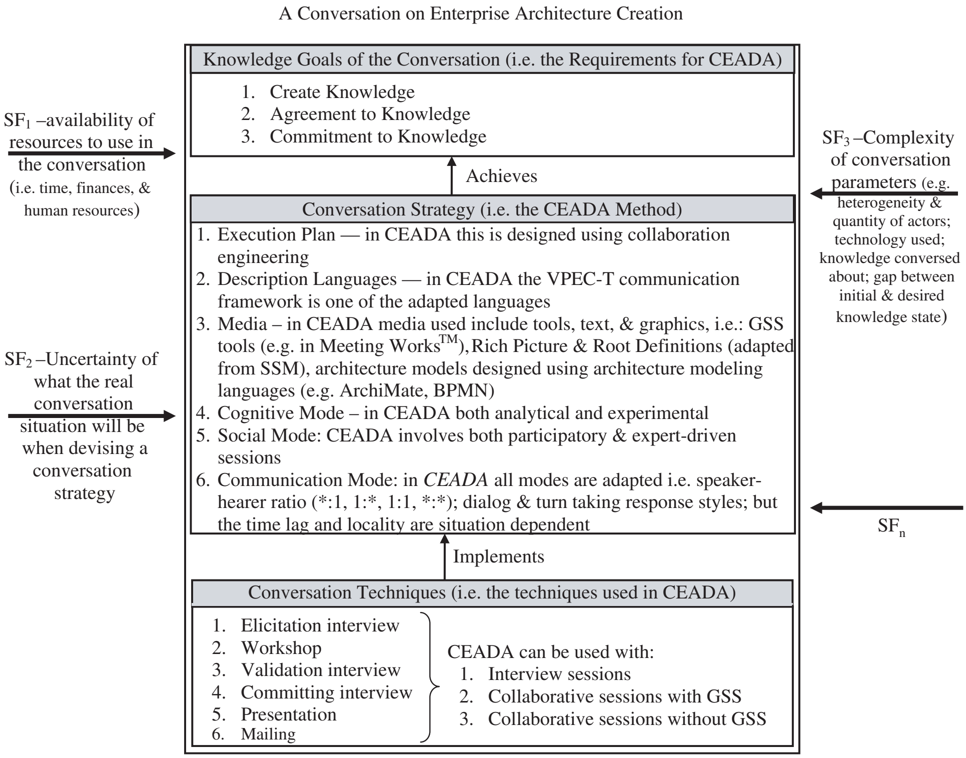

Fig. 2. Conversations on enterprise architecture creation (based on guidelines in Ref. 60).

(4) The cognitive mode showing how actors will gather and process knowledge during the conversation — it may be analytical (i.e. abstracting information so as to reach a deeper shared understanding); or experimental (i.e. experimenting ideas using prototypes or other techniques so as to reduce uncertainties). Figure 2 shows that in this research both the analytical and experimental knowledge gathering and processing approaches are adapted.

(5) The social mode showing how system development actors will collaborate with business actors during the conversation — it may be expert-driven (where the development team uses their own expertise and interviews with business actors to produce descriptions, and then delivers them to business actors for approval); or participatory (where the development team produces descriptions in close cooperation with business actors using workshops). Figure 2 shows that in this research both the participatory and expert-driven are adapted.

(6) The communication mode showing the basic patterns of communication that will be used in the conversation, i.e. speaker–hearer ratio (which may be *:1, 1:*, 1:1, *:*), the response style (which may be dialog and turn taking if an answer is expected from the hearer), communication time lag between speaking and

hearing, and the locality or physical and cultural distance between actors. The modes considered in this research are shown in Fig. 2.

In this research, as Fig. 2 shows, the above guidelines were adapted by making use of existing approaches in the knowledge base of Fig. 1 (Sec. 2.4). However, prior to adapting these guidelines, there was a need to first understand the orchestration of the underlying variables or aspects (and their relationships) in realizing CDM in enterprise architecture creation. This is because the knowledge of causal relations enables predictions to be made from theory.[12] These predictions consequently enable, e.g. rational structuring of subconversations in a conversation execution plan. Thus, Sec. 3.1 discusses the theory of CDM in enterprise architecture creation that was used as a basis for defining SMART architectural knowledge goals; developing an execution plan for the architecture creation conversation; choosing the appropriate media, cognitive modes, social modes, and communication modes in architecture creation conversation; and developing a process that supports the participatory social mode in the architecture creation conversation.

3.1. *The theory for CEADA*

Sections 2.3.1 and 2.3.2 list the activities involved in enterprise architecture creation, define the generic decision-making process, explain the essence of cause–effect analysis in this research, and define CDM. Basing on those definitions, the following conclusion can be drawn, i.e. "realizing CDM in enterprise architecture creation requires enterprise architects and organizational stakeholders to cooperate, with the aim of: gaining shared understanding of the "as-is" and "to-be" situations of the organization; identifying and devising possible design alternatives for realizing the desired or "to-be" (or target) organization situation; evaluating the possible impacts of these design alternatives; and finally selecting (and agreeing on) the design alternative that is feasible, effective, and efficient".[57] The underlying factors in this declaration, the associated (causal and conditional)[a] relations, and the resulting sequential relations, constitute the theory shown in Fig. 3. Earlier versions of this theory were discussed in Refs. 57 and 66. Findings from the evaluation of CEADA models (see Sec. 5.3.3) were used to validate the notions in this theory.

Relation 1 means that successful enterprise architecture creation[b] involves a successful negotiation process. This relation is derived from the following line of thought. It has been reported in Ref. 15, that during the development of enterprise (or reference) architecture, a negotiated solution may be more appropriate than an optimal solution. Negotiations among stakeholders and enterprise architects enable stakeholders to understand why all their concerns and requirements

[a]A causal relation between x and y means that more of x leads to more of y, yet a conditional relation between x and y means that x is vital for the success of y (with no ample x there is no y).
[b]Herein successful architecture creation is perceived as gaining stakeholders' acceptance of the designed architecture and being able to implement it with their support and commitment.[3]

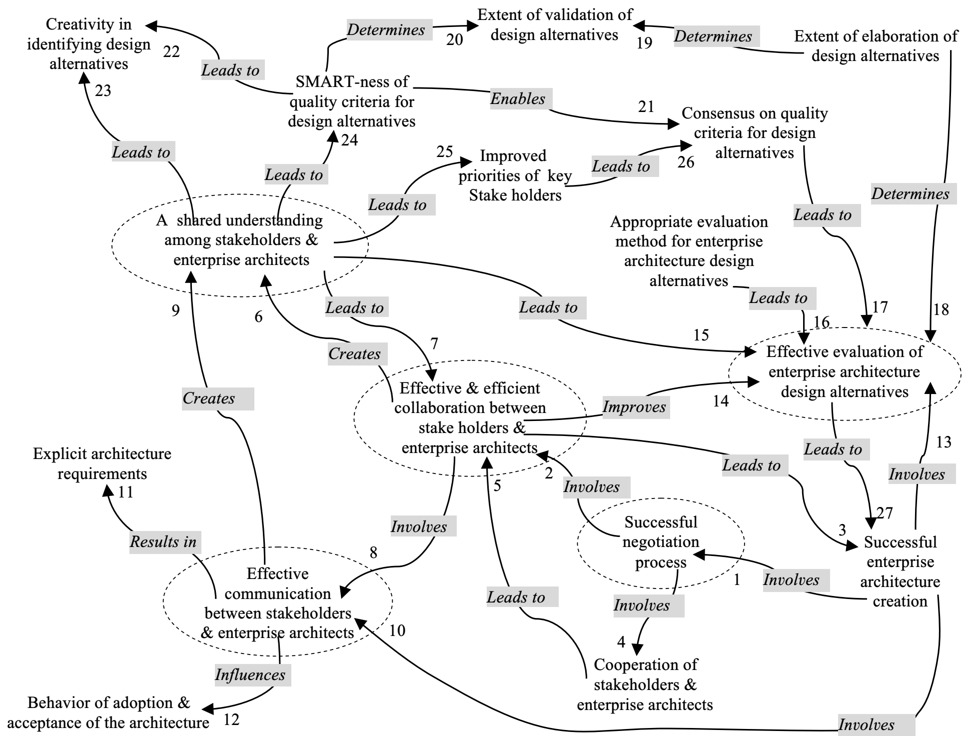

Fig. 3. Theory of CDM in enterprise architecture creation.

cannot be satisfied by the enterprise architecture.[57] This implies the need for the negotiation theory in enterprise architecture creation. In the negotiation theory, the joint decision is not only the final decision in a given task, since a negotiation involves several emerging opportunities for joint decisions that eventually lead to the final joint decision.[59] Likewise, an enterprise architecture can be perceived as a collection of joint decisions that are made throughout the phases of architecture creation. For example, according to Ref. 18, output from the preliminary phase of the architecture development method is used as input in the architecture vision phase, and output from the architecture vision phase is used as input in the phases of developing business, data, applications, and technology architectures, and output from all these phases yields the ultimate enterprise architecture. Relation 1 is therefore a conditional relation, where successful negotiation is required for the success of enterprise architecture creation.

Negotiations involve multiple individuals cooperating to arrive at a joint (or collaborative) decision which results in joint consequences for each individual.[59] Hence, relations 2 and 4. Relation 2 means that successful negotiation involves effective and efficient collaboration among stakeholders and enterprise architects, while relation 4 means that a successful negotiation process involves cooperation of stakeholders and enterprise architects. Moreover, collaboration is when multiple individuals join their efforts to achieve a given goal.[31] This implies that relations 2 and 4 are conditional relations, in the sense that successful negotiations cannot occur unless individuals are willing to collaborate and cooperate. From relations 1 and 2, it can be deduced that effective and efficient collaboration (among stakeholders and enterprise architects) will lead to successful architecture creation (as indicated by relation 3). Relation 3 is also based on existing literature on enterprise architecture,[1–3,6,9,15–18,53,54] which recommends that during enterprise architecture creation, enterprise architects need to work collaboratively with organizational stakeholders. The resulting sequence from relations 1, 2, and 3 is denoted as 1–2–3.

Furthermore, since cooperation is when an individual renders his or her (expected) effort to a group result without intentionally frustrating the efforts of other individuals, this implies that cooperativeness is an individual's trait.[31] Therefore, as indicated by relation 5, the cooperation of individual stakeholders and enterprise architects leads to effective and efficient collaboration among stakeholders and enterprise architects. Moreover, stakeholders' cooperation is vital for the success of the architecture project, since they provide the organizational resources that are required in architecture development; determine the requirements and constraints of the architecture; influence others; and make decisions.[3] This implies that relation 5 is conditional. Stakeholders' cooperation, therefore, indirectly leads to successful architecture creation (as indicated by sequences 4–5–3 or 1–4–5–3).

Relation 6 means that effective and efficient collaboration creates a shared understanding of the "as-is" and "to-be" aspects of the organization among stakeholders and enterprise architects. This relation is derived from the definition of a collaborative environment, i.e. a collaborative environment people purposely "*spend as much time understanding what they are doing as actually doing it*", with the aim of "*creating a shared understanding that didn't exist before*".[68] Moreover, since a shared understanding is a basis for effective collaboration,[6] relation 7 means that shared understanding among stakeholders and architects will result in effective and efficient collaboration. This is because stakeholders' commitment increases as they gain shared understanding of the "as-is" and "to-be" aspects.[15] Thus, relation 6 is a conditional relation, while relation 7 is a causal relation. In addition, in Ref. 18, it is noted that the lack of shared goals and expectations between stakeholders and architects is the source of project failures in many cases. Hence, sequence 6–7–3.

Relations 8 to 12 are derived from the *calls* for effective communication in enterprise architecture development[15,18,19,54,60,61,69] (see Sec. 2.2 in categories (A) and (B)). Relation 8 means that effective and efficient collaboration between stakeholders and architects involves effective communication among stakeholders and enterprise architects. This effective communication in turn helps to create a shared understanding among stakeholders (as indicated by relation 9). Relation 10 further indicates that successful architecture creation involves effective communication between stakeholders and enterprise architects. Moreover, from Ref. 15, effective communication eliminates ambiguities and this results in explicit requirements for the architecture (as indicated by relation 11) as well as positively influencing the acceptance and adoption of the architecture (as indicated by relation 12). In general, it is vital for architects to communicate critical and actionable information, and to collaborate with willing stakeholders (who value the architecture concept), this will help to build relationships that will yield long term and mutually beneficial results that encourage adoption of architecture throughout the organization.[9,61] Hence, relations 8 and 10 are conditional in the sense that effective communication is vital for effective collaboration and successful architecture creation (respectively), while relations 9, 11, and 12 are causal.

Relation 13 means that successful architecture creation involves effective evaluation of architecture design alternatives, while relation 27 means that effective evaluation of alternatives leads to successful architecture creation (and hence sequence 13–27). This is because in addressing stakeholders' concerns and requirements, the architect develops architecture views that show the trade-offs required to resolve conflicting concerns.[18] Such trade-offs are clarified through evaluation of (solution and design) alternatives.[3] Moreover, satisfactory solutions are obtained through evaluating possible (design) alternatives or courses of action.[11,13] Relation 16 means that using an appropriate evaluation method for enterprise architecture design alternatives leads to effective evaluation of those alternatives. Relation 14 means that effective and efficient collaboration between stakeholders and enterprise architects improves the evaluation of architecture design alternatives. This is because for (complex) organizational problems it can be difficult for one individual to understand and foresee all implications of a given decision or course of action, and therefore the best decision requires combining expertise of people from different disciplines or domains.[31] Note that sequence 14–27 confirms relation 3.

Furthermore, it has been noted that the commitment of stakeholders increases as they acquire a shared understanding[15] or a shared goal.[31] This implies that achieving a shared understanding directly improves the priorities of stakeholders, and this consequently results in consensus on quality criteria for design alternatives.[66] Therefore, relation 25 means that a shared understanding among stakeholders leads to an improvement in their priorities, while relation 26 means that an improvement in stakeholders' priorities leads to consensus on quality criteria for design alternatives. Consequently, if stakeholders have reached consensus on quality criteria for evaluating alternatives, this will lead to effective evaluation of alternatives (as indicated by relation 17). Hence, sequence 25–26–17. Both sequences 25–26–17 and 7–14 imply relation 15 which means that if stakeholders have acquired a shared understanding of the "as-is" and "to-be" contexts of the organization, then they can effectively evaluate architecture design alternatives.

Relation 24 means that if stakeholders have acquired a shared understanding of the "as-is" and "to-be" organization contexts, then they can unambiguously define quality criteria for design alternatives, and this leads to SMART (i.e. Specific, Measurable, Achievable, Realistic, and Time bound) quality criteria.

If quality criteria for architecture design alternatives are SMART, then it is quicker for stakeholders to reach consensus on these criteria (as indicated by relation 21). Hence sequence 24–21. Sequence 24–21–17 also implies relation 15. Moreover, if quality criteria for design alternatives are SMART, then this leads to creativity in identifying (possible and relevant) architecture design alternatives (as indicated by relation 22). Relation 23 (which is implied by sequence 24–22) means that if stakeholders have acquired a shared understanding (of the "as-is" and "to-be" organization contexts), then this leads to creativity in identifying architecture design alternatives. Besides, creativity in formulating solution strategies and new alternatives is one of the core components of the negotiation theory.[59] Relation 20 means that the SMARTness of quality criteria for design alternatives determines the extent of validation of design alternatives. Relations 18 and 19 mean that the extent of elaboration of design alternatives determines the effectiveness of the evaluation of design alternatives and the extent of validation of design alternatives.

It is vital to note that all the relations discussed in this theory hold as long as other relations are considered. Major conclusions from this theory include the following sequential relations, i.e. 1–2–3; 1–2–14–27; 1–2–6–25–26–17; 1–4–5–8–9–23; 6–7–8–9; and 1–4–5–8–9–25–26–17. From these sequences, the core notion in CEADA is drawn, i.e. *the core parameters for effective and efficient* collaborative evaluation of enterprise architecture design alternatives (*which is the vision of CEADA*) *are effective communication*, *negotiation*, *and shared understanding among enterprise architects and organizational stakeholders*.

3.2. *The requirements for CEADA*

This section presents the requirements for CDM in enterprise architecture creation that were extracted from the theory discussed in Sec. 3.1. The extraction was done by making predictions (that are based on relations and sequences in the theory) of the conditions (or requirements or activities) that have to be fulfilled before other conditions (or requirements or activities) can be fulfilled. All the requirements are then sequenced and structured into steps and sessions, by adapting the intelligence, design, and choice phases of the generic decision-making process (defined by Simon[13]); and the multilevel thinking concept of SSM (defined by Checkland[22]).

Therefore, Figs. 4 and 5 define the requirements for CEADA (or the explicit knowledge goals that any architecture creation conversation ought to endeavor to achieve). In other words, these requirements define *what* ought to be done in order to achieve CDM in conversations on enterprise architecture creation. Since human conversations are significantly affected by the confusion that sprouts from the failure to properly organize thoughts and expressions,[22] there is need to have an explicit (flexible) structure of conversations on creating enterprise architecture. Thus, the requirements in Figs. 4 and 5 depict the structure of the architecture creation conversation. Earlier versions of Figs. 4 and 5 were discussed in Refs. 58 and 66.

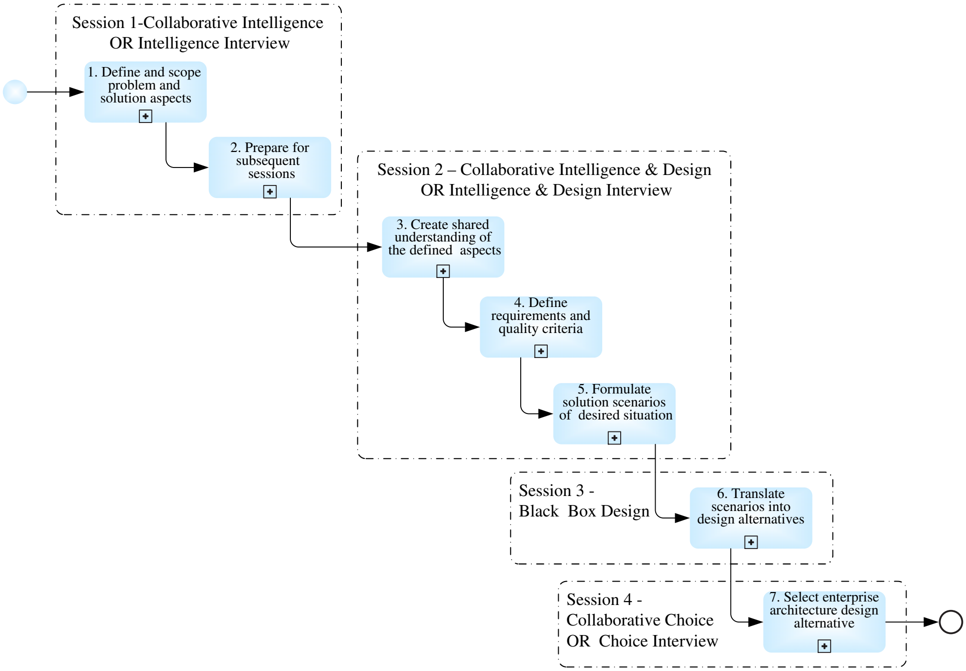

Fig. 4. Requirements for CEADA.

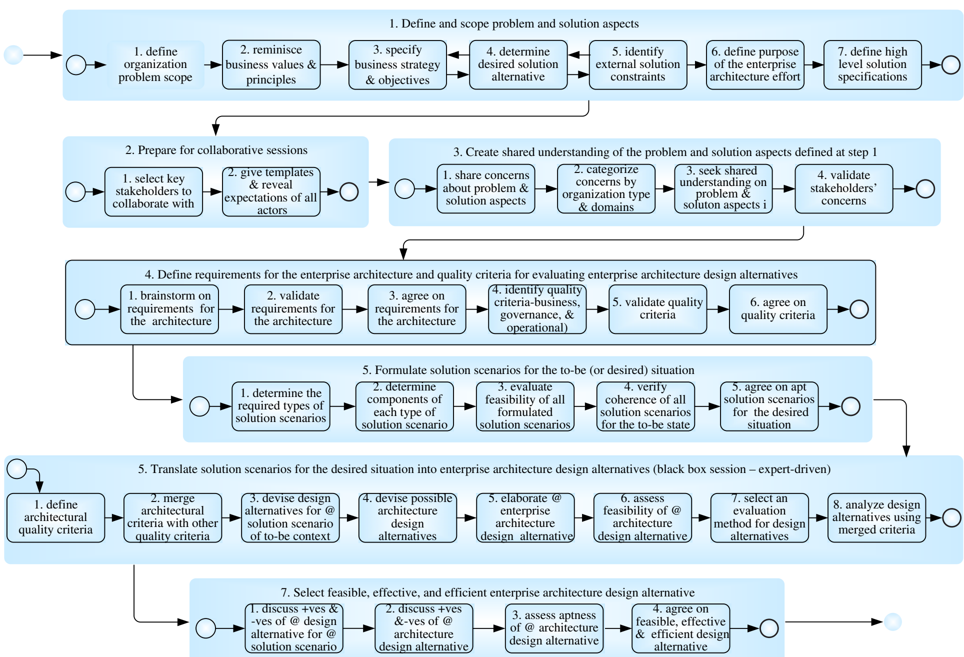

Fig. 5. Decomposition of requirements for CEADA.

Figure 4 is principally an adaptation of the phases of the generic decision-making process in Ref. 13, which have been summarized in Sec. 2.3.1. Session 1 is derived from the intelligence phase, session 2 is derived from both the intelligence phase and design phase, session 3 is derived from the design phase, and session 4 is derived from the choice phase. The grouping of steps in each session is also based on the concept of multilevel thinking defined by Checkland[22] in SSM. Thinking in levels or layers or hierarchies is vital because it enables actors or decision makers to purposely distinguish the "*whether*", the "*what*", and the "*how*".[22] Thus, in Fig. 4, session 1 addresses the *whether* level (or layer) of thinking and also addresses (to an abstract extent) the *what* level of thinking in the enterprise architecture creation conversation. In session 2, steps 3 and 4 address details of the *what* level of thinking, while step 4 addresses (to an abstract extent) the *how* level of thinking in the architecture creation conversation. Session 3 addresses the details of the *how* level, while session 4 completes the *how* level of thinking in the architecture creation conversation. Moreover, session 4 also involves, to some extent, the *whether* level of thinking, since there is need to choose an enterprise architecture design alternative that will suit the dynamic nature of the business environment.

This structuring is made more explicit in Fig. 5, which decomposes the requirements in Fig. 4, into explicit requirements. The decomposition of requirements was also based on the relations and sequences in the theory for CEADA (see Sec. 3.1). Earlier versions of models showing the requirements in Figs. 4 and 5 were evaluated using the analytical, experimental, and observational evaluation methods (as discussed in Sec. 5). Findings from the evaluation (see Secs. 5.3.1–5.3.3) were then used to refine and validate the earlier models so as to yield Figs. 4 and 5. The production life cycle of Figs. 4 and 5 is thoroughly discussed in Refs. 58 and 66. Section 4 discusses the design of a collaboration process that defines *how* the requirements in Figs. 4 and 5 can be addressed.

4. Using Collaboration Engineering to Address the Requirements

In collaboration engineering (which was introduced in Sec. 2.3.4), the requirements for realizing CDM in enterprise architecture creation (presented in Sec. 3.2) can be conceived as the basic activities that a collaboration process should support during enterprise architecture creation. This section therefore presents an operational outlook on realizing CDM in architecture creation, by discussing how the requirements in Sec. 3.2 can be addressed using collaboration engineering, and other concepts adapted from SSM and ASE.

4.1. *Patterns of collaboration and ThinkLets*

Effective collaboration requires participants to undergo a reasoning process that comprises of a series of activities referred to as basic patterns of collaboration or thinking.[27,42] In collaboration engineering, six general patterns of collaboration are defined in Refs. 27, 38 and 42 as follows:

(1) *Generate*: This involves participants moving from a state of having fewer concepts to a state of having more concepts that are shared by the group.
(2) *Reduce*: This involves participants moving from a state of having many concepts to having fewer concepts that the group considers worthy of further attention.
(3) *Clarify*: This involves participants moving from a state of having less to more shared understanding of concepts and phrases used to express those concepts.
(4) *Organize*: This involves participants moving from a state of having less to more understanding of the relationships among concepts considered by the group.
(5) *Evaluate*: This involves participants moving from a state of having less to more understanding of the relative value of the concepts under consideration.
(6) *Build consensus*: This involves participants moving from a state of having fewer to more group members willing to agree and commit to a proposal.

Each of the above patterns of collaboration (or some variation on each) is created by a unit known as a ThinkLet.[27,42] A ThinkLet (which was initially tagged as a recipe, or technique, or reasoning module) "*is the smallest unit of intellectual capital required to create one repeatable, predictable pattern of thinking among people working toward a goal*".[42] According to Refs. 27 and 42, a ThinkLet consists of three components, i.e. (1) a tool — the specific hardware and software technology that should be used to create a given pattern of thinking; (2) tool configuration — specification of how the hardware and software have to be configured so as to create a given pattern of thinking and participant interaction; and (3) a script — the sequence of events and instructions that have to be given to the participants so as to create a given pattern of thinking. Thus, ThinkLets are building blocks for designing collaboration processes.[14,39] The criteria used to select ThinkLets that make up a collaboration process formulated for supporting a given task are given below.

In Refs. 42, 27 and 71, the following details (or selection criteria) about several ThinkLets are discussed: (1) a (memorable) name that is associated with the (group) dynamics that a given ThinkLet creates; (2) criteria for selecting or deciding when to use or when not to use a given ThinkLet; (3) the composition of a given ThinkLet; (4) advantages (based on field experience) of using a given ThinkLet over other ThinkLets that create a given pattern of collaboration; (5) real life experiences that clarify circumstances under which a given ThinkLet might be useful; and (6) an explanation of the origin of the name of the ThinkLet.

4.2. *The design approach for collaboration processes*

To address the requirements for CEADA in Sec. 3.2, a collaboration process was designed. According to Refs. 14 and 40, the following are the steps involved in designing a collaboration process.

(1) *Task diagnosis*: Determine the goal and deliverable(s) of the collaboration process. In this research, the goal of the collaboration process is to support the realization of CDM during enterprise architecture creation.
(2) *Task decomposition*: Determine the basic activities for achieving the goal and the deliverable(s). In this research, the requirements, or basic activities, that the collaboration process should support are shown in Figs. 4 and 5 (see Sec. 3.2). The decomposed requirements in Fig. 5 are described as activities under the column "activity description" in Tables 1–3. In addition, each of these activities is assigned a "pattern of collaboration" that will enable actors (i.e. enterprise architects and stakeholders) to undergo a level of thinking or reasoning that is required for them to accomplish a given activity. In the CEADA method, the assigned patterns of collaboration are shown in column "pattern of collaboration" in Tables 1–3.
(3) *ThinkLet choice*: Assign each basic activity a suitable ThinkLet that will support its realization. The matching of ThinkLets and the basic activities is based on some criteria, i.e. the purpose of a given ThinkLet or conditions for using it (see Sec. 4.1). In Refs. 14, 27, 42, 43 and 71, the criteria for matching basic activities with suitable ThinkLets are discussed. Examples of ThinkLets, as defined in Refs. 27, 42 and 43, include *LeafHopper*, *FreeBrainstorm* (are some of the ThinkLets used to create the "Generate" pattern of collaboration); *MoodRing* (is one of the ThinkLets used to create the "Build consensus" pattern of collaboration); *StrawPoll* (is one of the ThinkLets used to create the "Evaluate" pattern of collaboration), etc. In CEADA, the "ThinkLet(s)" column in Tables 1–3 shows the ThinkLets that have been selected to address each requirement in Fig. 5. An extract from Ref. 71, of some of the selected ThinkLets in CEADA, is given in the appendix.
(4) *Agenda building*: Prepare all information required for validating the process, and (graphically) represent it in a facilitation process model. This is essentially "*the logic of the flow of the collaboration process from activity to activity*".[14]
(5) *Design evaluation and validation*: Use walkthroughs, pilot testing, simulation, and expert evaluation to validate and evaluate the collaboration process. In this research, the evaluation of the design of CEADA in Tables 1–3 is discussed in Sec. 5.
(6) *Documentation*. The design of the collaboration process of CEADA is shown in Tables 1–3. In other words, noting from Fig. 2 in Sec. 3, Tables 1–3 show the execution plan of sessions that constitute a conversation on enterprise architecture creation. These tables are discussed in Secs. 4.3–4.6.

4.3. *Session 1 — Collaborative intelligence session*

Table 1 shows the design of the collaboration process (or the execution plan) for session 1 of CEADA. Session 1 originates from Figs. 4 and 5, where the decomposed requirements of steps 1 and 2 in Fig. 5 relate to the activities in the rows of Table 1.

Table 1. Design of session 1 of CEADA — collaborative intelligence.

Session 1: Collaborative Intelligence Session (OR Intelligence Interview Session)			
#	Activity Description	Pattern of Collaboration	ThinkLet(s)
1.0	Communicate purpose of the session and kind of information required from session	—	—
1.1	Define organization's problem scope (or existing situation)		
	1.1.1. Identify aspects on the problem & its scope 1.1.2. Agree on aspects of the problem & its scope	Generate, Clarify, Organize Build consensus	OnePage, Could-Be-Should-Be StrawPoll, CrowBar
1.2	Reminisce (& precisely define) the key business or organization values & principles or policies (assumption is that these are already in existence)	Generate, Clarify, Organize	LeafHopper, ReviewReflect
1.3	Define basic information on business strategy, business objectives, and business requirements	Generate, Clarify, Organize, Evaluate, Build consensus	DealersChoice, FastFocus, StrawPoll, CrowBar
1.4	Determine suitable (business) solution alternative for achieving the business strategy & business goals (or the desired organization situation)	Generate, Clarify, Evaluate, Build consensus	OnePage, Could-Be-Should-Be MoodRing
1.5	Identify external constraints (from e.g. regulatory authorities) to the solution aspects chosen in 1.3 and 1.4	Generate, Clarify, Organize	DealersChoice, FastFocus
1.6	Define purpose of the enterprise architecture effort		
	1.6.1. Determine what the architecture results will be used for 1.6.2. Agree on the purpose of the architecture results	Generate, Clarify, Organize Build consensus	OnePage, ReviewReflect MoodRing

Table 1. (*Continued*)

Session 1: Collaborative Intelligence Session (OR Intelligence Interview Session)			
#	Activity Description	Pattern of Collaboration	ThinkLet(s)
1.7	Define high level solution specifications of the solution alternative in 1.4		
	1.7.1. Generate ideas on high level solution specifications. This gives the scope of the solution alternative chosen in activity 1.4.	Generate Additional concept to use: formulate the high level solution specifications in form of "Root Definitions"	FreeBrainstorm
	1.7.2. Filter generated high level solution specifications. The filtering is based on the output from activities 1.1, 1.2, 1.3, & 1.5	Reduce, Clarify	FastFocus
	1.7.3. Agree on high level solution specifications	Evaluate, Build Consensus	StrawPoll, CrowBar
1.8	Seek shared understanding on the scope of the problem and its solution, and seek consensus on whether the scope of these aspects is worth a collaboration effort of organization key stakeholders	Build Consensus	MoodRing
2.1	Select key stakeholders to participate in the subsequent architecture creation conversation with enterprise architects, and define their roles in the architecture conversation	Generate, Clarify Additional concept to use: "Analysis One Two Three" using deliverables of activity 1.1, 1.2, & 1.7	OnePage, ReviewReflect
2.2	Give information templates for session 2; and communicate the calendar of upcoming events in the architecture effort, the expectations of architect team, & find out stakeholders' expectations in the subsequent collaborative efforts during enterprise architecture creation		

Table 1 also shows the patterns of collaboration assigned to each activity in session 1, and the ThinkLets chosen to realize each activity. Although some activities may have the same set of patterns of collaboration assigned to achieve them, the set of ThinkLets assigned to achieve them may be different. For example, see rows 1.1.1, 1.2, 1.5, and 1.6.1 in Table 1. This is because, as discussed in Sec. 4.1, each pattern of collaboration or a variation of it is realized using a number of ThinkLets.[27,42,43] So the appropriate ThinkLet for a given activity depends on the aim of that activity, or number of stakeholders required to accomplish that activity, or input required to accomplish that activity, or desired output of that activity, or a combination of all these factors.

Moreover, noting from Fig. 2 in Sec. 3, the social mode in this session is both participatory and expert-driven. Moreover, in the participatory social mode of this session, the nature of participation of stakeholders is referred to as "*governance-driven-division*". Governance-driven-division means that the type of stakeholders that enterprise architects are collaborating with to accomplish a given task (or the type of stakeholders required to accomplish a given task) are senior management or line of business managers, or key decision makers in the organization. Governance-driven-division is required when executing activities 1.1–2.1, while activity 2.2 requires participation of all stakeholders in the enterprise architecture initiative.

In CEADA, the concept of determining key stakeholders for accomplishing a given task, or dividing stakeholders to accomplish a given task, is adapted from: (1) the architecture development guidelines in the TOGAF ADM[18] and (2) the "*take-a-panel*" and "*share-a-panel*" techniques of the ASE approach. From Refs. 47 and 48, and the walkthroughs, it was explained that in group sessions of ASE, take-a-panel means dividing participants into small groups, so that they solve a given problem and learn new skills (within a short session), whereas share-a-panel means giving each participant an opportunity or turn to explain his or her own concepts to members in his or her group, after a take-a-panel session has ended. In CEADA, these two concepts are adapted but instead used in four flavors, i.e. (1) governance-driven-division, (2) "*specialization-driven-division*", (3) "*task-driven-division*", (4) "*interest-driven-division*". Section 4.4 explains specialization, task, and interest driven.

The focus of this session is to define and determine the scope of key aspects of the problem (or existing situation) and the intended solution (or desired situation) in the organization; and to make preparations for sessions 2–4 (in which measures for materializing these aspects are to be devised). Output from this session is used to formulate the as-is (or baseline) architecture models for the organization. In CEADA, the "Rich Picture" and "Analysis One Two Three" concepts of SSM have been adapted to be used in the formulation of the as-is situation in the organization. According to Ref. 22, a Rich Picture is a representation of aspects in a problem situation (that are gathered from e.g. semistructured interviews) which encourages a holistic and exploratory thinking about the situation. In CEADA, output from activities 1.1 and 1.2 (see Table 1) is used to formulate the Rich Picture that can be translated into baseline architecture models.

In addition, the "Root Definition" concept of SSM is adapted when executing activity 1.7 of Table 1. According to Ref. 22, Root Definitions are short phrases formulated in the form of, "Do P by Q in order to contribute to achieving R", in order to answer the questions of (or influence thinking in terms of): what to do (i.e. P), how to do it (i.e. Q), and why do it (i.e. R). In CEADA, this concept is used when defining high-level solution specifications (in activity 1.7), and when formulating solution scenarios in activities 5.1–5.4 (see Table 2).

Output from activities 1.1–2.1 is used to perform Analysis One Two Three, and gives a high-level sketch of the to-be or desired situation of the organization. Analysis One Two Three involves assessing the problem situation to identify the correct list of possible problem owners (which is a pointer to the relevant systems for improving the situation) and to find out the social (i.e. roles or norms or values) and political factors in the situation.[22] In CEADA, this concept is adapted in activity 2.1 so as to successfully complete activity 2.2 (see Table 1). Moreover, in CEADA the list of problem owners consists of stakeholders who are invited in activity 2.2, and is also a pointer to the type of solution scenarios that have to be formulated in activity 5.1 (see Table 2 in Sec. 4.4). Thus, noting from Fig. 2 in Sec. 3, the media used in this session includes graphics showing the as-is Rich Picture and baseline architecture models, and text describing the Analysis One Two Three and Root Definitions.

In addition, this session is referred to as intelligence interview session, if it is not possible to conduct group sessions with the stakeholders in the governance-driven-division category. In the field study evaluation of CEADA models, this session was executed as an intelligence interview session in cases 1, 2, and 4, while in case 3 it was executed as both a collaborative intelligence and an intelligence interview session (see Sec. 5.3).

4.4. *Session 2 — Collaborative intelligence and design session*

Table 2 shows the design of the collaboration process (or the execution plan) for session 2 of CEADA. Session 2 originates from Figs. 4 and 5, where the decomposed requirements of steps 3, 4, and 5 in Fig. 5 relate to the activities in the rows of Table 2. Table 2 also shows the patterns of collaboration assigned to each activity in session 2 and the ThinkLets chosen to realize each activity. Like in session 1, although some activities may have the same set of patterns of collaboration assigned to achieve them, the set of ThinkLets assigned to achieve them may be different. For example, see rows 3.4 and 5.5 in Table 2. Section 4.3 discusses the cause of this difference. In addition, noting from Fig. 2 in Sec. 3, the social mode in this session is both participatory and expert-driven. Moreover, in the participatory social mode of this session, the nature of participation of stakeholders can be in four alternative ways, i.e. specialization-driven-division, task-driven-division, interest-driven-division, and governance-driven-division (which is explained in Sec. 4.3).

Specialization-driven-division means that when executing a given task in the collaborative session, stakeholders need to be divided into small groups based on

Table 2. Design of session 2 of CEADA — Collaborative intelligence and design.

Session 2: Collaborative Intelligence & Design Session (OR Intelligence & Design Interview Session)				
#	Activity Description	Stakeholder Participation	Pattern of Collaboration	ThinkLet(s)
3.0	Communicate the purpose of the session and kind of information required	All		
3.1	Stakeholders share their concerns about the organization's problem & solution aspects (that were represented in the rich picture of the baseline situation & the sketch picture of the target situation formulated in session 1)	Specialization-driven or task-driven or interest-driven	Generate	LeafHopper
3.2	Categorize concerns by type and organization domains/units/departments		Reduce, Clarify	PopcornSort
3.3	Analyze and discuss concerns while seeking a shared understanding of all problem and solution aspects	All	Organize	BucketWalk, BucketBriefing
3.4	Validate stakeholders' concerns basing on the output from activities 1.2, 1.3, 1.4, 1.5, & 1.7 in session 1	All	Evaluate, Build Consensus	StrawPoll, Red-Light-Green-Light
4.1	Brainstorm on the (business) requirements that the enterprise architecture must address	Specialization-driven or task-driven or interest-driven	Generate	LeafHopper
4.2	Categorize and discuss requirements for the architecture		Reduce, Clarify, Organize	PopcornSort, BucketWalk, BucketBriefing
4.3	Validate and agree on requirements for the architecture, basing on the valid stakeholders' concerns from 3.4 and output from activities 1.3 & 1.7 in session 1	All	Evaluate, Build Consensus	StrawPoll, Red-Light-Green-Light

Table 2. (*Continued*)

Session 2: Collaborative Intelligence & Design Session (OR Intelligence & Design Interview Session)				
#	Activity Description	Stakeholder Participation	Pattern of Collaboration	ThinkLet(s)
4.4	Brainstorm on business, governance, & operational quality criteria for evaluating design alternatives. This is based on output from activities 4.3 & 1.2 in session 1	Specialization-driven or task-driven or interest-driven	Generate	LeafHopper
4.5	Categorize and discuss quality criteria		Reduce, Clarify, Organize	PopcornSort
4.6	Validate and agree on (business, governance, & operational) quality criteria	All	Evaluate, Build Consensus	StrawPoll, Red-Light-Green-Light
5.1	Identify the required type of solution scenarios. This is based on high level solution specifications from activity 1.7 in session 1	All	Generate, Clarify	Comparative Brainstorm, ReviewReflect
5.2	Determine components of each type of solution scenario (i.e. identify, detail, & assemble components of each identified solution scenario)	Specialization-driven or task-driven or interest-driven	Generate Additional concept to use: formulate the solution scenarios in form of "root definitions"	FreeBrainstorm
5.3	Evaluate feasibility of all formulated solution scenarios & their components	All	Clarify, Organize	Could-Be-Should-Be
5.4	Verify coherence of all solution scenarios for the to-be state (or desired situation of the organization)	All	Organize	ChauffuerSort
5.5	Agree on apt solution scenarios for the desired situation	All	Evaluate, Build Consensus	MultiCriteria, Red-Light-Green-Light

their specialization units or departments in the organization. Task-driven-division means that when executing a given task in the collaborative session, stakeholders need to be (randomly) divided into small groups where each small group is assigned a subtask that contributes to a main task in the session. Interest-driven-division means that when executing a given task in the collaborative session and completing that task requires stakeholders to be divided into small groups (where each small group is assigned a subtask that contributes to a main task in the session), the stakeholder has a free will of deciding which small group he or she would like to join and work with. Task-driven and interest-driven are used as substitutes for the specialization-driven-division in situations where stakeholders in an organization are few in number, or when some departments are represented by one person in session 2 of CEADA. This can happen due to the factors that range from time, culture, expertise, governance, to even political factors. The required division of stakeholders when executing each task in this session is indicated in the column "stakeholder participation" in Table 2.

The focus of this session is to create a shared understanding of the problem and solution aspects (that were defined in session 1); to define (business) requirements and quality criteria that the enterprise architecture must address; and to formulate solution scenarios that cater for those requirements. Activity 3.1 (see Table 2) highlights the reference to the rich picture formulated in session 1. From Ref. 22, a rich picture can be used as a starting point of an exploratory discussion with people in a problem situation. This concept was adapted in CEADA to represent information from session 1 and to start off discussions (that will create a shared understanding among stakeholders) in session 2. Moreover, the rich picture can be used to solicit comments and views (from the problem owners) on what the main issues in the situation are; give a holistic view of the situation; and contribute to the understanding of the social and cultural aspects of the situation.[22] Thus, since the main focus of this session is to create shared understanding (among organizational stakeholders and enterprise architects) of the problem and solution aspects pertaining to architecture development, the rich picture was adapted to be used in both sessions 1 and 2 of CEADA. This session, in activities 5.1–5.4, also adapts the Root Definition concept of SSM, to be used during the formulation of solution scenarios for the desired or to-be situation of the organization. This concept has been defined in session 1 of CEADA (see Sec. 4.3).

In addition, this session can be perceived as an intelligence and design interview (which is executed by using interviews), if it is not possible to conduct collaborative sessions. As explained in Sec. 4.3, this mainly arises due to organization politics, or when stakeholders have failed to get time to attend the collaborative sessions.

4.5. *Session 3 — Black box design session*

This is essentially expert-driven (involving enterprise architects only). Thus, the social mode in this session is expert-driven. This session originates from step 6 (i.e. translate scenarios into enterprise architecture design alternatives) in Figs. 4 and 5.

Table 3. Design of session 4 of CEADA — collaborative choice.

Session 3: Black Box Design Session (Translate Solution Scenarios into Architecture Design Alternatives)				
Session 4: Collaborative Choice Session (OR Choice Interview Session)				
#	Activity Description	Stakeholder Participation	Pattern of Collaboration	ThinkLet(s)
7.0	Communicate purpose of session and kind of information required	All	—	—
7.1	Discuss the positive and negative implications of each possible design alternative for each type of solution scenario (that was chosen in session 2), to each group of stakeholders who are affected by a given type of solution scenario	Specialization-driven or task-driven-division	Organize	—
7.2	Discuss the positive and negative implications of each enterprise architecture design alternative (i.e. compositions of design alternatives for solution scenarios) to all stakeholders	All	Organize	—
7.3	Evaluate enterprise architecture design alternatives		Evaluate	MultiCriteria
7.4	Agree on feasible, effective, and efficient enterprise architecture design alternative		Evaluate, Build Consensus	MultiCriteria, Red-Light-Green-Light

CEADA does not delve into the details of the activities in this session, since the tasks that enterprise architects do to accomplish this step are technical and are defined by enterprise architecture approaches and architecture modeling languages.

4.6. *Session 4 — Collaborative choice session*

Session 4 originates from Figs. 4 and 5, where the decomposed requirements of step 7 in Fig. 5 relate to the activities in the rows of Table 3. The focus of this session is to select a suitable (i.e. feasible, effective, and efficient) enterprise architecture design alternative. We consider an enterprise architecture (or its design alternative) to be effective if it is capable of addressing its planned purpose and realizing organization objectives; efficient if it addresses all stakeholders' concerns[3]; and feasible if it is achievable given the organization's resources. This session can be treated as a choice interview, which is executed by using interviews if it is not possible to conduct collaborative sessions.

5. Evaluation of CEADA Models

Design Science artifacts are evaluated using observational, analytical, experimental, descriptive, or testing methods.[10,26] This section discusses the implications of findings from the analytical, experimental, and observational evaluation of the CEADA models.

In Design Science, research is considered to be relevant to IS and IT practitioners, if the resultant artifact will address the problems they face and maximize opportunities from the interaction of people, organizations, and IT.[10] Likewise, in this research, it was vital to find out early in time the practical relevance of (developing) the CEADA method. This was done by exposing the CEADA models to practitioners using the analytical evaluation method, as explained in Sec. 5.1. Since evaluation reveals weaknesses in a theory or artifact and the need for its refinement and reassessment,[10] feedback obtained from evaluating artifacts is used to progressively refine them.[10,26] Similarly, in this research, findings from the analytical evaluation were used to refine the earlier versions of CEADA models (see Appendix). The refined models were then evaluated using an experiment. Experiment findings were used to refine the models further. The models were then further evaluated using a field study. Findings from the field study were used to achieve the models discussed in Secs. 3 and 4. These phases of evaluation and refinement of CEADA models are discussed in Secs. 5.1–5.3.

5.1. *Analytical evaluation of CEADA models*

According to Ref. 10, analytical evaluation can be done in four ways, i.e. (1) static analysis — inspecting static qualities in the structure of an artifact; (2) architecture analysis — inspecting how an artifact fits into a technical Information Systems architecture; (3) optimization — inspecting and revealing the in-built optimal properties of an artifact; and (4) dynamic analysis — investigating dynamic qualities of an artifact while it is being used. Additional ways in which (domain) experts can do analytical evaluation include: Usability inspection,[44] heuristic evaluation, and (cognitive) walkthroughs.[45] In this research, walkthroughs were used to analytically evaluate the initial versions of CEADA models.

A walkthrough is a step-by-step review and discussion with practitioner(s), of activities that make up a process to reveal errors that are likely to hinder the effectiveness and efficiency of the process (or method) in realizing its intended plan.[14,31] Walkthroughs generally involve one or more evaluators (or experts) performing a stepwise review of a scenario (or representation of the design of an artifact) so as to note possible problems.[45] Several variations of walkthroughs are commonly used in software development to find errors in software code and functionality, verify software requirements, validate software against predefined standards, reduce risks of discontinuity, and generally improve software quality.[46] Similarly in collaboration engineering, walkthroughs are used as one of the methods for evaluating and validating the design of a collaboration process.[14] Therefore, in this research, prior to using other evaluation methods, conducting walkthroughs with enterprise architects and professional facilitators was the appropriate way of first validating the requirements for CEADA and the design of the collaboration process in CEADA.

Analytical evaluation of CEADA models was done using six bilateral structured walkthroughs (with experienced enterprise architects and professional facilitators) and an expert review (which was done by a collaboration engineer). The walkthroughs were set up and conducted as follows:

(1) The aim of the walkthroughs was to obtain practical insights on collaborative aspects in enterprise architecture creation and to identify and eliminate faults and ambiguities in the requirements and activities described in the initial versions of the CEADA models.
(2) Participants: bilateral structured walkthrough sessions were used; therefore, each session involved two actors, i.e. the researcher and an experienced enterprise architect or professional facilitator.
(3) Duration: each session lasted for at least 1 h and at most 2 h.
(4) Inputs to the analytical evaluation phase were the initial CEADA models (see Appendix).
(5) Agenda: the researcher started a session by explaining the aim of the research (as stated in Sec. 2), the aim of the walkthrough (given in (1) above), and the roles of the practitioner (i.e. architect or professional facilitator) during the walkthrough. The roles of the practitioner were to (1) comment on the relevance of CEADA in practice; (2) review the requirements (and activities) described in the models with the focus of identifying faults and ambiguities, and giving practical insights into eliminating them; and (3) verify (based on his or her experience) the relevance of the defined requirements and activities in achieving the general aim of the research. These were done through a stepwise discussion of the inputs to the walkthrough.

(6) Output: during the walkthroughs the researcher took notes which are discussed in Sec. 5.3. Findings from the analytical evaluation were used to refine the CEADA models, which were further evaluated as discussed in Sec. 5.2.

5.2. *Experimental and observational evaluation of CEADA*

In Ref. 25, it is recommended that it is vital to first evaluate evolving artifacts (in Design Science) in laboratory or experimental settings before evaluating them in field studies. Hence the reason for first evaluating the design and performance of CEADA using an experiment prior to using observational evaluation. Findings from this evaluation (see Sec. 5.3) were used to further validate and refine the intermediate versions of CEADA models, so as to obtain the CEADA models that have been discussed in Secs. 3 and 4.

5.2.1. *Evaluation criteria and performance measurement for CEADA*

The criteria for evaluating the CEADA method are derived from the models that describe the requirements of CDM in enterprise architecture creation (see Figs. 4 and 5 in Sec. 3.2), and are classified into effectiveness and efficiency. In assessing efficiency, the goal is to find out the shortest possible time in which CEADA satisfies criteria 1–4 below. Criteria 1–3 were earlier reported in Ref. 58, and criterion 4 has been added to assess whether the CEADA method achieves its ultimate purpose. In assessing effectiveness, the goal is to find out whether CEADA really supports the following:

(1) Creating a shared understanding of the organization's problem and solution aspects (as well as the scope of those aspects) among stakeholders and enterprise architects. This specifically involves assessing whether CEADA supports explicit definition, and agreement on, the requirements, quality criteria, and solution scenarios that the enterprise architecture must address.
(2) Creating or enhancing commitment among stakeholders toward the success of the architecture creation effort.
(3) Evaluating, selecting, and agreeing on, a suitable enterprise architecture design alternative.
(4) Providing a clear operational structure and facilitation support for all activities, so as to enable effective communication and realization of CDM in enterprise architecture creation.

The following are the indicators that were used to measure the performance of CEADA under criteria 1–4 above (i.e. its effectiveness). Indicators 1–3 were earlier reported in Ref. 58, and indicator 4 has been specified to measure CEADA's performance under criterion 4 above:

(1) Support for creating a shared understanding of the organization's problem and solution aspects (among stakeholders) was measured by the level of consensus

(or variation) among stakeholders on the concerns, requirements, quality criteria, solution scenarios that the architecture must address.

(2) Support for creating or enhancing stakeholders' commitment was measured by stakeholders' dedication to accomplishing the activities in the CEADA method. Stakeholders' dedication takes into account their attendance of the CEADA sessions and willingness to participate and make contributions toward the success of the enterprise architecture effort.
(3) Support for evaluation of enterprise architecture design alternatives, and selection of, and agreement on, the suitable architecture design alternative was measured by the level of consensus (or variation) among stakeholders on a given enterprise architecture design alternative.
(4) Support for providing a clear operational structure and facilitation support for all activities was measured by the level of consensus among stakeholders on their satisfaction with the activities done in each session of CEADA, and how they were done.

Furthermore, in performance indicators 1 and 3 above, the level of consensus (or degree of variation) among stakeholders was measured by the standard deviation of the weights or scores that stakeholders assigned to the concerns, requirements, quality criteria, formulated solution scenarios, and enterprise architecture design alternatives during the CEADA sessions. Data of these scores and their standard deviation was captured in the GSS data logs or reports. The GSS technology that was used to support execution of activities in CEADA was Meetingworks™. Moreover, in performance indicator 4, the level of consensus (or degree of variation) among stakeholders was measured by the mean score and standard deviation of scores that stakeholders gave in questionnaires that they filled after the CEADA method was used. Questionnaires were filled by all participants or stakeholders who attended the sessions of CEADA. This approach of using a questionnaire survey to measure participants' satisfaction with a collaboration process and its outcome was introduced by Briggs *et al.*[67] In the questionnaires used to evaluate CEADA, the 5-point Likert scale questions was used, with responses ranging from strongly disagree (point 1) to strongly agree (point 5). Results of CEADA's performance evaluation are given in Table 6 and discussed in Sec. 5.3.

In indicator 2, stakeholders' dedication above was measured by their attendance and willingness to participate (or their enthusiasm) in the sessions of CEADA. To achieve this, the researchers observed the "*atmosphere*" in the rooms in which the CEADA sessions were conducted. The performance of CEADA under efficiency criterion is measured by the duration of each collaborative session and the time spent on preliminary activities that prepare for the collaborative sessions.

5.2.2. *Research method used in the experimental and observational evaluation*

According to Ref. 10, observational design evaluation methods are case study (i.e. studying the artifact in a given business environment) and field study (i.e. studying and monitoring the use of an artifact in several projects or contexts). In this research, field study was used. A field study can be done using action research method.[26] Thus, in the experiment and the field study (which involved four cases), the action research approach was used. In action research, researchers steadily work with subjects in the investigation and transformation experience.[64] Thus, action research was used because there was a necessity to first evaluate CEADA in a setting where researchers facilitate the CEADA sessions before evaluating it in a setting where real architects (practitioners) facilitate the CEADA sessions.

According to Ref. 63, action research involves five steps, i.e. (1) Diagnosing step, where the root cause of the desire for change in an organization is identified; (2) Action planning step, where the organizational actions that will address the main problem are determined and specified; (3) Action taking, which involves researchers collaborating with practitioners (and stakeholders) to implement the planned action so as to realize the desired changes in the organization; (4) Evaluating step, which involves researchers and practitioners (and stakeholders) determining whether the theoretical effects of the action taken were realized; and (5) Specifying learning, which involves using knowledge gained from the research (whether it was successful or not) to improve a theoretical framework or the organization's situation. Table 4 summarizes how the steps in action research were undertaken in the experiment and the field study.

5.3. *Discussion of findings*

Key findings from the analytical, experimental, and field study evaluation of CEADA are summarized in Table 5, and discussed in Secs. 5.3.1–5.3.3.

5.3.1. *Discussion of main findings from the analytical evaluation*

Below is a discussion of items 1–6 that are listed in the second cell of the last column of Table 5.

(1) The need to add new requirements or decompose some requirements for CEADA (that existed in the initial models) implied defining new requirements by incorporating the recommendations that practitioners gave during the walkthroughs. Examples of the key added requirements in session 1 of CEADA (after the walkthroughs) include definition of the organization problem scope, business goals and strategic drivers, external constraints from regulatory bodies, purpose of architecture effort, high-level solution specifications, and specification of key stakeholders (and decision makers) in the architecture effort. Sections 3 and 4 show the CEADA models that incorporate these insights (from practitioners). Moreover, in the field study, performing these activities in session 1 helped to gather sufficient information that was used to prepare for sessions 2, 3, and 4.

(2) It was also advised that session 1 of CEADA can be executed using a collaborative session, or by using both interviews and a collaborative session. This implied designing the facilitation for session 1 of CEADA (see Sec. 4.3). Actually

Table 4. Action research in the experiment and the field study.

#	Steps in Action Research	Experiment (About a Fictitious National University in NL)	Case 1 — Nsambya Home Care (NHC)	Case 2 — Makerere University Guest House (MUKGH)	Case 3 — Wakiso District Local Government (WDLG)	Case 4 — Bugema University (BU)
1	Problem Diagnosis	Lack of insight on how to implement the strategy of expanding from a National University to a Networked European University	The hectic and time consuming process of capturing data, retrieving records, & compiling reports for donors funding the organization	Lack of the basic infrastructure & service management for delivering the quality services desired by clients	The desire to achieve coordinated service deliver to residents in Wakiso district & to effectively communicate with other districts local governments in Uganda	Lack of a centralized Information System for enabling quick & accurate information storage, dissemination & retrieval
2	Action Planning	Develop an enterprise architecture (vision) for the desired Networked European University, so as to guide & inform the transformation from a National University to a Networked European University	Automate some operational processes in NHC so as to ensure effective & efficient data capturing, storage, sharing, retrieval & reporting	Upgrade from a guest house to a 3-star hotel, provide quality services & increase customer base Diversify products & services	Automate some of the operational processes in each department, & foster IT-enabled information capturing, storage, & sharing across all departments in WDLG	Devise new ways of working in all departments in order to enable development of a centralized information system for managing students' information
			Create an enterprise architecture (vision) that will guide & inform the desired transformation in each of the 4 organizations			
3	Action Taking	1. The CEADA method was used to support the enterprise architecture creation conversation among the researchers and the stakeholders in each of the 4 organizations (or participants in the case of the experiment). 2. In each organization TOGAF was used as the guiding enterprise architecture framework; and BPMN as the modeling language.				
4	Evaluate	1. The design and performance of CEADA were evaluated by the participants and stakeholders using questionnaires. 2. The CEADA execution environment was evaluated by the researchers.				
5	Specify learning	Lessons learned from the experiment and the 4 cases were used to refine the CEADA models.				

Table 5. Main findings from three evaluation phases of CEADA models.

Type of Evaluation Phase		Total Number of Participant(s)	The Aim of the Evaluation Phase	Summary of Main Findings from the Evaluation Phase
Phase 1: Analytical evaluation method using bi-lateral structured walkthrough sessions	Session 1	1 enterprise architect	1. To find out the practical relevance of (developing) the theory for CEADA & the CEADA method 2. To validate & refine requirements for CDM in enterprise architecture creation	1. There was need to add new requirements or decompose & elaborate some requirements for CEADA that existed in the initial models. 2. Session 1 of CEADA can be executed using collaborative sessions and not only by using interviews, or by using both. 3. Session 2 of CEADA is successful if stakeholders prepare their concerns & requirements for the architecture before the collaborative sessions. 4. Some requirements in the sessions of CEADA can be realized through adapting the concepts of take-a-panel & share-a-panel techniques of ASE. 5. Enterprise architecture design alternatives are at 2 levels i.e. high-level solution alternatives & low-level solution design alternatives. 6. Need to distinguish between causal & conditional relations in the theory for CEADA; & to assign new ThinkLets to support some activities in CEADA.
	Session 2	2 enterprise architects		
	Session 3	1 enterprise architect		
	Session 4	1 enterprise architect		
	Session 5	1 professional facilitator		
	Session 6	1 professional facilitator & enterprise architect		
	Expert review	1 collaboration engineer		
Phase 2: Experimental evaluation using action research approach		26 students (participants)	1. Evaluate the performance of CEADA 2. Evaluate & validate the design of CEADA	1. The relevance of adapting the take-a-panel & share-a-panel techniques of ASE during sessions 2 and 4 of CEADA. 2. Need to allocate more time in CEADA sessions for: negotiating on validity of brainstormed items; & formulating & evaluating solution scenarios. 3. The core parameters in CEADA are: effective communication, negotiation, and shared understanding among enterprise architects & stakeholders.

Table 5. (*Continued*)

Type of Evaluation Phase		Total Number of Participant(s)	The Aim of the Evaluation Phase	Summary of Main Findings from the Evaluation Phase
Phase 3: Observational evaluation using action research approach	Case 1 Case 2 Case 3 Case 4	13 stakeholders 1 stakeholder 20 stakeholders 8 stakeholders	1. Evaluate the performance of CEADA 2. Evaluate and validate the design of CEADA	1. Executing the collaborative sessions in CEADA without using GSS had a negative impact on CEADA's performance in case 3, compared to the better performance of CEADA in case 1 where sessions were supported by GSS. 2. The need to use other quicker techniques during the formulation of solution scenarios because the manila chart approach that was earlier used (when formulating solution scenarios) was very time consuming & did not motivate some stakeholders to be deeply involved in the formulation of the solution scenarios. Hence the adaptation of *rich picture & root definitions* of SSM. 3. Although CEADA's performance improves when take-a-panel & share-a-panel techniques of ASE are adapted, the *way* of adapting & using these two techniques in the CEADA collaborative sessions is a situational factor which varies depending on a given situation in the organization. 4. The need to devise ways of enhancing negotiations during the CEADA sessions when seeking consensus on requirements, quality criteria, solution scenarios, and enterprise architecture design alternatives. 5. The need to make explicit the distinction between concerns & requirements, and the type of concerns that stakeholders need to share during session 2 of CEADA. Hence the adaptation of the VPEC-T framework in CEADA.

in the field study, session 1 of CEADA in cases 1, 2, and 4 was executed using interviews. Yet in case 3, session 1 of CEADA was executed using both interviews and a collaborative session. This is because in case 3, many key stakeholders had to be contacted in order to accomplish the activities defined in session 1 of CEADA, and yet some stakeholders had time for interviews but did not have time for collaborative sessions.

(3) It was also advised that there was a need to ensure that stakeholders prepare their concerns and requirements for the architecture prior to the commencement of session 2 of CEADA. This implied that in session 1 of CEADA, there was a need to devise a way of specifying to stakeholders two aspects: (1) the type of concerns and requirements that they are required to share in session 2; and (2) that their concerns and requirements are to be evaluated (in session 2) using the organization's policies, principles, high-level solution specifications, and external constraints. Thus, from the walkthroughs it was advised that at the end of session 1 of CEADA, it could be useful if an informal gathering is organized where the architectural team communicates its expectations from the organizational stakeholders, and stakeholders also communicate their expectations from the architectural team and the architecture effort. It was also advised that in session 2 of CEADA, in order to define explicit evaluation criteria for the enterprise architecture design alternatives, there was need to decompose the evaluation criteria into business criteria, architectural criteria, governance criteria, and operational criteria.

(4) Some requirements in the sessions of CEADA could be realized through adapting the concepts of take-a-panel and share-a-panel in the ASE approach. Sections 4.3–4.6 discuss the adaptation of these two techniques. The effects of not adapting, or of adapting, the take-a-panel and share-a-panel concepts in CEADA are explained in Secs. 5.3.2 and 5.3.3.

(5) The need to consider that enterprise architecture design alternatives are at two levels, i.e. high-level solution (or business) alternatives and low-level solution design alternatives. From the walkthroughs it was advised that during the formulation and evaluation of the high-level solution alternatives, stakeholders' participation is very important. However, the translation of a given solution alternative into low-level solution design alternatives involves several technical aspects that are difficult for most stakeholders to understand. Thus, only enterprise architects (in sync with domain experts) are involved in that translation. Moreover, to enable selection of an appropriate enterprise architecture design alternative in session 4 of CEADA, architects need to indicate (in a language that the organizational stakeholders understand) the positive and negative implications of the low-level solution design alternatives. Practitioners also highlighted that it saves time if some possible solution alternatives are formulated (by architects) before the collaborative sessions, as this would give stakeholders a clue of what is required during the formulation of solution scenarios.

It was also advised that during evaluation of enterprise architecture design alternatives, there was need to consider the "relevance of opinion" of each stakeholder by assigning weights to stakeholders during the voting activities. Moreover, from the walkthroughs a professional facilitator advised that in Meetingworks^{TM} GSS, this aspect (of stakeholders' relevance of opinion) is done by filtering weights assigned by stakeholders from a given unit or position in the organization.

(6) There was need to distinguish between causal and conditional relations in the theory for CEADA (as this was not explicit in the initial CEADA models). This implied specifying which relations in the theory were causal (i.e. where more of x leads to more of y), and which relations were conditional (i.e. where x is required for the success of y — in other words, without sufficient x there is no y). This has been specified in the revised theory for CDM in enterprise architecture creation (see Sec. 3). Moreover, some activities in the collaboration process were assigned new ThinkLets based on insights from the practitioners. For example, in session 2 of CEADA, although categorize concerns (by type and organization domains) involves reducing and clarifying brainstormed concerns (and therefore requires "reduce and clarify" patterns of collaboration), Fast focus ThinkLet was not the appropriate ThinkLet for this activity (as was indicated in the initial CEADA models). Instead, Popcorn sort ThinkLet is the appropriate for supporting this task. Also, in session 2 of CEADA, another ThinkLet had to be chosen to support the task of analyzing and discussing stakeholders' concerns. This is because it was advised that the Crowbar ThinkLet (which had been chosen in the initial CEADA models) cannot be used before voting — as one of its inputs is an indication of standard deviation of voting results.

5.3.2. *Discussion of main findings from the experimental evaluation*

This section discusses main findings from the experiment (that are summarized in the third cell of the last column of Table 5) and CEADA performance results from the experiment (that are presented in Table 6).

Using the "performance evaluation questionnaire" technique that has been explained in Sec. 5.2.1, CEADA's performance was evaluated by participants in the experiment and stakeholders of cases 1 and 3 (who participated in the collaborative sessions). In cases 2 and 4, CEADA was executed using interviews (see Sec. 5.3.3 for details). Table 6 therefore, presents results of CEADA's performance in the experiment and in only cases 1 and 3.

From Table 6, evaluation results from the experiment are discussed below along with items 1–3 (that are listed in the third cell of the last column of Table 5).

(1) The experiment was run without adapting the take-a-panel and share-a-panel techniques in the CEADA sessions. Thus, in all collaborative sessions of CEADA, all participants worked in one group to accomplish activities in each session (following the design in the initial CEADA models). This way of working

Table 6. Performance evaluation of CEADA in the experiment and the field study.

#	Evaluation Criteria for CEADA	Type of Evaluation	Performance Indicator	
			Mean Score	Standard Deviation of Scores
1	Satisfaction with the activities done in the collaborative sessions	Experiment (GSS was used)	2.00	0.88
		Case 1 (GSS was used)	4.20	0.42
		Case 3 (No GSS was used)	3.71	0.76
2	Satisfaction with the outcome(s) of the collaborative sessions	Experiment	2.05	0.91
		Case 1	4.20	0.42
		Case 3	3.43	0.79
3	Collaborative sessions helped to increase understanding of the concerns and requirements of all units in the organisation	Experiment	3.89	0.94
		Case 1	4.50	0.53
		Case 3	4.12	0.54
4	Collaborative sessions helped stakeholders to freely express their views about the current operations in the organisation	Experiment	3.53	1.22
		Case 1	4.50	0.53
		Case 3	3.71	1.38
5	Collaborative sessions helped stakeholders to understand why some of their concerns/views were not voted by others during the sessions	Experiment	3.11	1.05
		Case 1	3.30	1.25
		Case 3	3.86	0.90

Brief summary on how CEADA was implemented	In the experiment, Meetingworks™ GSS was used in executing CEADA
	In the case 1, Meetingworks™ GSS was used in executing CEADA
	In case 3, CEADA was executed without GSS, but by using the alternative manual techniques used in traditional group meetings or tasks (i.e. brainstorming in terms of turn taking, use of stickers, manila paper, makers etc.)
	In cases 2 and 4, CEADA was executed using interview sessions

hindered participants from rigorously negotiating on the requirements and solution scenarios for the enterprise architecture, and on the enterprise architecture design alternatives. As a result, the level of consensus (among participants) on these aspects was seriously impaired (as indicated in Table 6). This implied that before the field study, there was need to specify activities in sessions 2 and 4 of CEADA that required use of take-a-panel and share-a-panel techniques. There was also need to determine how these two techniques were to be used along with the ThinkLets. Moreover, adapting take-a-panel and share-a-panel in session 4 of CEADA implied decomposition of activities in that session. This was done to enable proper use of the two techniques, and also to enable stakeholders to understand how their particular concerns and requirements are addressed in the possible enterprise architecture design alternatives. This would (hopefully) enable them to rationally evaluate the design alternatives. The CEADA models in Sec. 4 explicitly show these refinements and the adaptation of take-a-panel and share-a-panel techniques. However, in the field study, margins of using the take-a-panel and share-a-panel techniques in CEADA were identified (as discussed in Sec. 5.3.3).

(2) There was need to allocate more time to CEADA sessions for negotiating on the validity of the brainstormed concerns and requirements of participants and formulating and evaluating solution scenarios that the architecture must address. This is because in the experiment participants were allowed to submit as many concerns and requirements (and ideas of solution scenarios) as they had, but all their concerns and requirements could not be cleaned, evaluated, and validated during session 2 of CEADA which was allocated a duration of 2 h. However, in the field study, the use of take-a-panel and share-a-panel techniques helped to make use of the 2 h more productively. Moreover, stakeholders were not willing (due to their busy schedules) to sit in a collaborative session for more than 2 h.

(3) A reflection on the experiment results and their implications gives a clue to the validity of the core notion in CEADA that was drawn from the theory for CDM in enterprise architecture creation (see Sec. 3.1). The implied notion is "The core parameters for effective and efficient *collaborative evaluation of enterprise architecture design alternatives* are: effective communication, negotiation, and shared understanding among enterprise architects and organizational stakeholders". The validity of this notion can be explained using the following reasons (from the experiment results) and other reasons from the field study (see Sec. 5.3.3).

First, since all participants worked in one group during the collaborative sessions in the experiment, there was insufficient individual-to-individual interactions or communication among participants who played the same role in the experiment, thereby leading to heterogeneity among a group of participants who played the same role, let alone the heterogeneity between groups of participants playing different roles in the experiment. This heterogeneity was indicated by the fact that participants who played the same role would assign high priorities to concerns and requirements that pertain to their role, and then assign low priorities to concerns and requirements from participants playing other roles.

Moreover, from the theory for CEADA (see Sec. 3.1), it was discussed how effective communication (directly and indirectly) helps to create a shared understanding among stakeholders and architects. Hence, due to poor communication in the experiment (that was caused by the way CEADA was executed), a shared understanding among participants was not achieved. The validity of this relation is indicated in the experiment results and confirmed in the field study results (see Sec. 5.3.3). Second, participants working in one group during the collaborative sessions hindered fruitful negotiation as explained in item (1) above. As a result of poor communication and almost no negotiation, shared understanding of the problem and solution aspects in the experiment could not be achieved.

5.3.3. *Discussion of main findings from the field study*

This section discusses field study findings (that are summarized as items 1–6 in the last column of Table 5) and CEADA performance results from cases 1 and 3 (that are presented in Table 6), as well as insights from cases 2 and 4 (in which CEADA was executed using interview sessions).

(1) CEADA's performance in case 1 was much better compared to its performance in case 3. Thus, executing the collaborative sessions in CEADA without using GSS had a negative impact on its performance in case 3 (see Table 6). In case 3, GSS was not used due to factors that will not be discussed for confidentiality reasons. On the other hand, this resulted in evaluating CEADA's effectiveness and efficiency in situations when it is not possible to use GSS to execute CEADA collaborative sessions. In such situations, alternative manual techniques are used, i.e. dialog and turn taking communication mode to perform tasks that require brainstorming (this substitutes the anonymous posting of items and commenting on items by participants when GSS is used); sketching solution scenarios by using manila papers, makers, white boards; commenting on solution scenarios by making use of stickers; and manual voting (which substitutes the quick anonymous voting when GSS is used). Using these manual techniques was very time consuming, thereby limiting discussions and negotiations on all stakeholders' concerns and requirements. Thus, it is more effective and efficient to use GSS to support CEADA activities that require brainstorming, categorizing brainstormed items, discussion of critical and sensitive aspects, and voting or ranking of items. Moreover, the anonymity provided when using GSS was very much appreciated by stakeholders in case 1 (who participated in the sessions supported by GSS).

(2) The manual alternative approach of devising sketches of the desired situation did not motivate some stakeholders to be deeply involved in the formulation of the solution scenarios that had to be addressed in the enterprise architecture vision. Hence the need to use other quicker and creative techniques during the formulation of solution scenarios. This is why SSM techniques (i.e. Rich Picture and Root Definitions) have been adapted in CEADA (see Secs. 4.3 and 4.4).

(3) The performance of CEADA improves if take-a-panel and share-a-panel techniques of ASE are adapted in the sessions. This is reflected in the improved performance of CEADA in case 1 compared to its performance in the experiment (see Table 6). However, the way take-a-panel and share-a-panel techniques can be adapted depends on the situation in a given organization. For example, in cases 1 and 3, these two techniques were adapted by dividing stakeholders according to their specialization units or departments in the

organization (i.e. specialization-driven-division). However, in case 3 it was noted that specialization-driven-division was not the appropriate way of using take-a-panel and share-a-panel, since some departments in case 3 were represented by one or two persons in the CEADA sessions. This implied the need to devise alternative ways of adapting and using the take-a-panel and share-a-panel techniques in the CEADA collaborative sessions. These alternative ways, as discussed in Sec. 4, include specialization-driven-division, task-driven-division, governance-driven-division, and interest-driven-division. Thus, this implies that the *way* of adapting take-a-panel and share-a-panel techniques during the execution of CEADA collaborative sessions is a situational factor, since it depends on a given situation in the organization. Adaptation of these techniques varies, as discussed in Sec. 4.4.

(4) Table 6 shows that stakeholders did not understand why some of their concerns, requirements, or views were not chosen or voted by other stakeholders. This implies that there is need to devise ways of enhancing negotiations during the CEADA sessions when seeking consensus on requirements, quality criteria, solution scenarios, and enterprise architecture design alternatives. This is because, based on the notion discussed in Sec. 5.3.2 (item 3), improving negotiations will help stakeholders to understand why some of their concerns or requirements or view are not chosen by others during the collaborative sessions.

(5) In session 1 of CEADA, interviews effectively supported the gathering of preliminary information that was used as a starting point in session 2. Stakeholders in session 2 however preferred to be given the output from session 1 early in time so as to enable them to prepare for the session. This concern was raised by most of the participants in cases 1 and 3 (who attended session 2 of CEADA), since majority of them had not attended session 1 because it was conducted using interviews with (senior) management. Moreover, stakeholders in session 2 felt that concerns and requirements were synonymous, and preferred to be given a template before session 2 that clarifies aspects that will be discussed in the session. These issues implied the need to improve communication in the CEADA sessions. Thus, the need to adapt the VPEC-T framework in CEADA, to offer a communication structure and vocabulary among organizational stakeholders and enterprise architects.

6. Conclusions

In this research, the Design Science methodology was adapted to design CEADA, a method that offers operational guidelines for realizing CDM in enterprise architecture creation. The theory of CDM in enterprise architecture creation was first formulated to serve as a basis for developing CEADA. CEADA has been designed by adapting a number of existing theories and approaches such as causal analysis, the generic decision-making process, SSM, CDM, collaboration engineering, and ASE among others. The requirements and activities represented in the CEADA models have been evaluated using analytical, experimental, and observational (i.e. field study) evaluation methods. The models were first evaluated using analytical and experimental methods, so as to (economically) arrive at models that could be evaluated using a field study. Moreover, this helped to achieve better results in the field study (which involved four real cases). Findings from these three evaluation phases were used to validate and refine the initial models, so as to yield the CEADA models that have been discussed herein. Currently, the CEADA models are being operationalized into the TOGAF ADM and Zachman framework, since CEADA is visualized as a potential plug-in for enterprise architecture frameworks.

Acknowledgments

The authors are grateful to Richard Bredero, Karin Blum, Arnold van Overeem, Claudia Steghuis, Hans Mulder, Raymond Slot, Tommes Snels, Mark van der Waals, and Gwendolyn Kolfschoten for their valuable contributions and practical insights into this research. They are also grateful to Daniel Kalibbala (the data manager of NHC), Joseph Kayizzi (the manager of MUKGH), the administration and staff members of NHC, Stephen Kasumba (the district planner of Wakiso district), the staff members of Wakiso district local government, and Francis Lowu, and the staff members of Bugema University, for their participation in the collaborative and interview sessions that validated the evolving models in this research.

Appendices

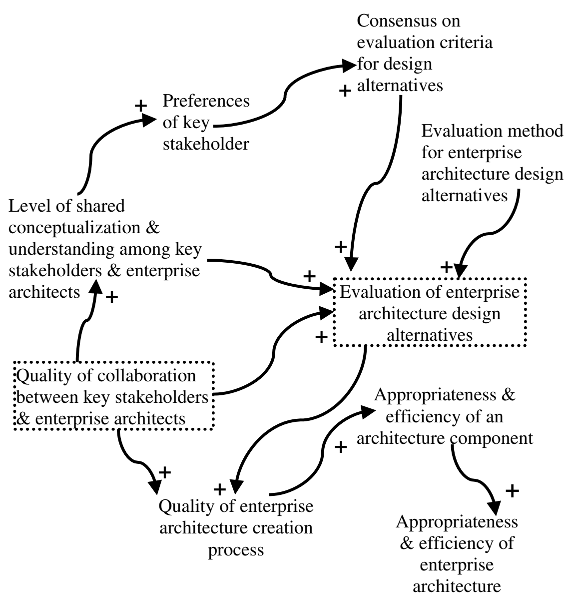

Fig. A.1. Input to analytical evaluation phase — Initial cause–effect analysis.

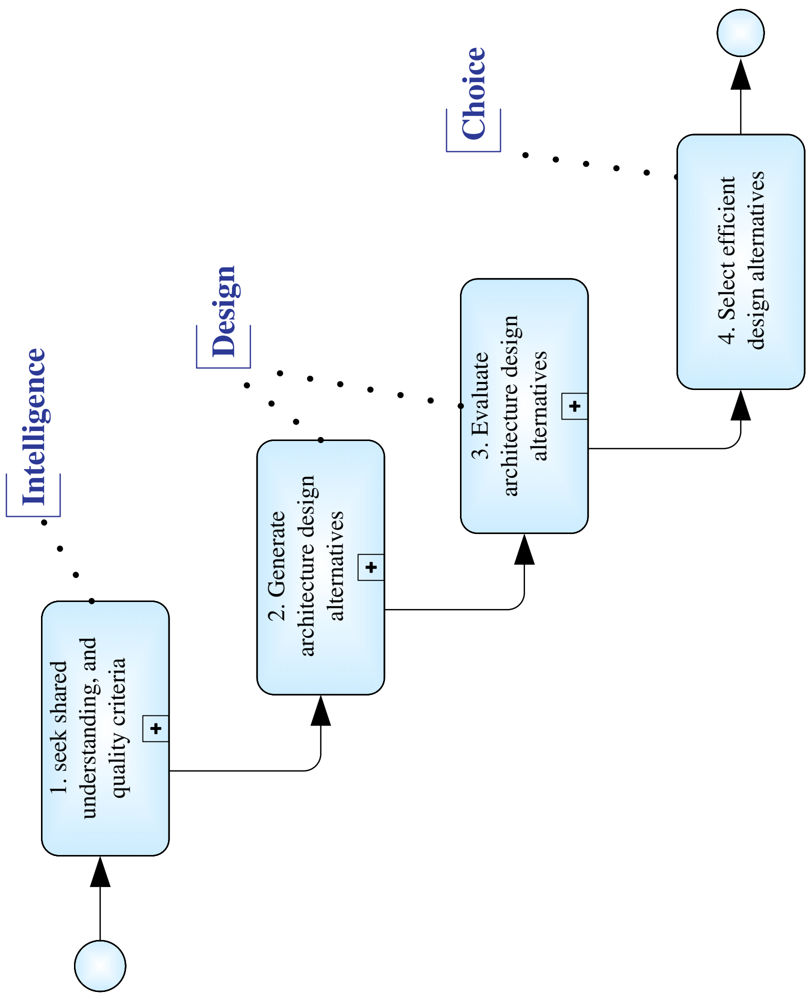

Fig. A.2. Input to analytical evaluation phase — Initial requirements and steps.

Table A.1. Input to analytical evaluation phase — Initial activities, patterns, and ThinkLets.

#	Activity Description	Deliverable	Pattern of Collaboration	ThinkLet
0	Prepare for architecture development sessions	Architecture Development information & sensitization	—	—
	Session 1 — Shared conceptualization & common evaluation criteria			
1A	Introduction/Briefing	Guiding information	—	—
1B	Share concerns	Concerns	Generate	LeafHopper
1C	Categorize concerns	Categories of concerns	Reduce & Clarify	FastFocus
1D	Discuss concerns while seeking shared conceptualization & understanding of enterprise aspects	Shared understanding of aspects & a common view of the enterprise	Build Consensus	CrowBar
1E	Identify criteria & methods for evaluating design alternatives	Evaluation criteria & methods	Generate	Free Brainstorm
1F	Categorize criteria & methods	Categories of criteria & methods	Reduce & Clarify	FastFocus
1G	Evaluate criteria & methods	Evaluated criteria & methods	Evaluate	StrawPoll
1H	Agree on evaluation criteria & method	Common evaluation criteria & evaluation method	Build Consensus	MoodRing
	Session 2 — Generation of enterprise architecture design alternatives			
2A	Identify design alternatives	Design alternatives	Generate	Comparative Brainstorm
2B	Elaborate alternatives	Elaborated alternatives	Generate	TheLobbyist
2C	Validate alternatives	Validate alternatives	Evaluate	StrawPoll
	Session 3 — Evaluation and selection of design alternatives			
3A	Evaluate alternatives	Evaluated alternatives	Evaluate	MultiCriteria
4A	Select appropriate & efficient alternative(s)	Appropriate & efficient design	Build Consensus	MoodRing

Table A.2. Extract from Ref. 71 of some of the selected ThinkLets.

OnePage (Extract from Briggs and Vreede 2000, Pages 16–17)
Choose this ThinkLet: To generate a few (<80) comments on one topic; when 5 or fewer people will brainstorm together; when 6 or more people will brainstorm for fewer than 10 minutes; when there aren't likely to be very many comments generated on the topic under discussion; to support back channel communication among distributed team members.

Do not choose this ThinkLet: When you expect more than 80 or so comments because it may cause information overload (consider FreeBrainstorm or ComparativeBrainstorm instead); when more 6/more people will brainstorm until they run out of ideas (consider FreeBrainstorm or ComparativeBrainstorm instead); when the team must address more than one topic at a time (consider LeafHopper or Dealer's choice instead).

Overview: Team members will contribute comments simultaneously to the same electronic page or list at the same time. Inputs — the brainstorming question or prompt. Output: A set of comments in response to a brainstorming question or prompt.

How to Use OnePage:
Setup: Open a single list or comment window in Topic Commenter, Vote, group Outliner, or Categorizer; Match views with participants to open the same list or card on their screens.

Steps: (1) Make sure the participants understand the brainstorming question or prompt (by encouraging them to ask where they don't understand); (2) if necessary, facilitate a verbal discussion to address any understanding difficulties, or else reformulate the question or prompt; (3) Inform participants of time limits (if any); and (4) let the participants contribute comments until they run out of ideas or until you call time.

DealersChoice (Extract from Briggs and Vreede 2000, Page 27)
Choose this ThinkLet: When the group must brainstorm on multiple topics; when you want to assure that certain participants address particular topics.

Do not choose this ThinkLet: If the order in which topics are addressed by the group does not matter;

If the background, organizational position, or expertise of the participants is of no essence to the discussion of topics.

Overview: The facilitator decided which participant is brainstorming on which topic. The facilitator offers the topics to participants identifying who is going to work on what. Often you use this to make sure that the team focuses on the same topics when you want it. You can make sure that each topic receives sufficient attention. Inputs — a set of topics. Outputs — a set of comments organized by topic.

How to Use DealersChoice:
Setup: Post the set of topics in Topic Commenter.

Steps: (1) Explain the topics to the group; (2) explain what kinds of responses will be useful; (3) determine which participants have to contribute to which topic, and assign participants to work on their topics; (4) Monitor progress among participants and if necessary, stimulate participants to move on.

ReviewReflect (Extract from Briggs and Vreede 2000, Pages 82–83)
Choose this ThinkLet: When you must review, validate, and modify the content of an existing outline or other information structure.

Do not choose this ThinkLet: When you need to generate an information structure from scratch. Consider BranchBuilder ThinkLet instead.

Overview: Inputs — pre-existing content in the form of a list, outline, or other document. Outputs — a revised document that more closely meets the needs of the task at hand.

How to Use ReviewReflect:
Setup: Post the existing contents in Group Outliner; Configure Group Outliner so that double-clicking any node of the outline opens a discussion window to collect anonymous comments on that note.

Table A.2. (*Continued*)

Steps: (1) Encourage participants to read each item in the list and reflect on its usefulness, and encourage them to open an item that they feel is not useful and to explain why; (2) allow all users to review, reflect, and comment on the outline simultaneously; (3) find an item that has comments under it, double click it open and read the comments; (4) use a match-views capability to open the same content window on the screens of other users; (5) encourage participants to propose a change to the outline list; (6) moderate an oral discussion, revise the outline as directed by the group; (7) repeat steps 3 through 6 until all comments have been addressed.

References

1. J. Schekkerman, *How to Survive in the Jungle of Enterprise Architecture Frameworks, Creating or Choosing an Enterprise Architecture Framework* (Trafford Publishing, Canada, 2004).
2. J. Schekkerman, *The Economic Benefits of Enterprise Architecture, How to Quantify and Manage the Economic Value of Enterprise Architecture* (Trafford Publishing, Canada, 2005).
3. M. Op't Land, H. A. Proper, M. Waage, J. Cloo and C. Steghuis, *Enterprise Architecture-Creating Value by Informed Governance* (Springer, Germany, EU, 2008).
4. M. Lankhorst *et al.*, *Enterprise Architecture at Work: Modelling, Communication, and Analysis* (Springer Verlag Berlin, Heidelberg, 2005).
5. Netherlands Architecture Forum, Extensible Architecture Framework version 1.1 (format edition) (2006), http://www.naf.nl/content/bestanden/xaf-1.1-fe.pdf.
6. B. van der Raadt, S. Schouten and H. van Vliet, Stakeholder perspective of enterprise architecture, in *ECSA*, eds. R. Morrison, D. Balasubramaniam and K. Falkner, LNCS, Vol. 5292 (Springer, Heidelberg, 2008), pp. 19–34.
7. J. Dietz, *Architecture Building Strategy into Design* (Netherlands Architecture Forum, Academic Service SDU, The Hague, The Netherlands, 2008), ISBN-13: 9789012580861, http://www.naf.nl.
8. S. H. Kaisler, F. Armour and M. Valivullah, Enterprise architecting: Critical problems, in *HICSS* (IEEE Press, 2005).
9. G. Muller, How to relate design decisions to stakeholder satisfaction. Bridging the broad stakeholder universe and the detailed technology world (2007), http://www.via-nova-architetura.org.
10. A. R. Hevner, S. T. March, J. Park and S. Ram, Design Science in Information Systems Research, *MIS Quarterly* **28**(1) (2004) 75–105.
11. H. A. Simon, *The Sciences of Artificial* (MIT Press, Cambridge, MA, 1996).
12. S. Gregor, The nature of theory in information systems, *MIS Quaterly* **30**(3) (2006) 611–642.
13. H. A. Simon, *The New Science of Management Decision* (Harper & Row, New York, 1960).
14. G. L. Kolfschoten and G. J. de Vreede, The collaboration engineering approach for designing collaboration processes, in *CRIWG*, eds. J. M. Haake, S. F. Ochoa and A. Cechich, LNCS, Vol. 4715 (Springer, Heidelberg, 2007), pp. 95–110.
15. M. Janssen and A. Cresswell, The development of a reference architecture for local government, in *HICSS* (IEEE Press, 2005).
16. F. J. Armour, S. H. Kaisler and S. Y. Liu, A big picture look at enterprise architectures, *IT Professional IEEE* **1**(1) (1999) 35–42.
17. F. J. Armour, S. H. Kaisler and S. Y. Liu, Building an enterprise architectures step by step, *IT Professional IEEE* **1**(4) (1999) 31–39.

18. The Open Group, *TOGAF — The Open Group Architecture Framework Version 9*, Personal PDF Edition, http://www.togaf.org (2009).
19. V. G. E. van Zanten, S. J. B. A. Hoppenbrouwers and H. A. Proper, System development as a rational communicative process, *Journal of Systemics, Cybernetics, and Informatics* **2**(4) (2004) 47–51.
20. M. Lankhorst and H. van Drunen, Enterprise architecture development and modelling (2007), http://www.via-nova-architectura.org.
21. M. Op 't Land, Principles and architecture frameworks, *Educational Material of University based Master Architecture in the Digital World* (Radboud University Nijmegen, The Netherlands, 2005).
22. P. Checkland, *Systems Thinking, Systems Practice — Includes a 30-year Retrospective* (John Wiley & Sons Ltd, Chichester, 1999), ISBN-10: 0-471-98606-2.
23. M. Op 't Land and H. A. Proper, Impact of principles on enterprise engineering, in *ECIS*, eds. H. sterle, J. Schelp and R. Winter (St. Gallen, Switzerland, 2007), pp. 1965–1976.
24. J. Nabukenya, P. van Bommel and H. A. Proper, Collaborative policy-making processes, *Technical Report: ICIS-R6036* (Radboud University Nijmegen, The Netherlands, December, 2006).
25. J. Iivari, A paradigmatic analysis of IS as a design science, *Scandinavian Journal of Information Systems* **19**(2) (2007) 39–64.
26. A. R. Hevner, A three cycle view of design science research, *Scandinavian Journal of Information Systems* **19**(2) (2007) 87–92.
27. R. O. Briggs, G. J. de Vreede and Jr. F. Nunamaker, Collaboration engineering with ThinkLets to pursue sustained success with group support systems, *Journal of Management Information Systems* **19** (2003) 31–64.
28. D. F. Anderson and G. P. Richardson, Scripts for group model building, *System Dynamics Review* **13**(2) (1997) 107–129.
29. E. A. J. A. Rouwette and J. A. M. Vennix, System dynamcis and organisational interventions, *Systems Research and Behavioral Science* **23** (2006) 451–466.
30. J. A. M. Vennix, Group model building: Tackling messy problems, *System Dynamics Review* **15**(4) (1999) 379–401.
31. G. L. Kolfschoten, *Theoretical Foundations for Collaboration Engineering* (Delft University of Technology, The Netherlands, 2007).
32. R. O. Mason and I. I. Mitroff, *Challenging Strategic Planning Assumptions: Theory, Cases, and Techniques* (Wiley, New York, 1981).
33. C. W. Churchman, *The Design of Inquiring Systems, Basic Concepts of Systems and Organisation* (Basic Books, New York, 1971).
34. C. W. Churchman, *The Systems Approach and Its Enemies* (Basic Books, New York, 1979).
35. R. L. Ackoff, *Redesigning the Future* (John Wiley and Sons, New York, 1974).
36. D. F. Andersen, J. A. M. Vennix, G. P. Richardson and E. A. J. A. Rouwette, Group model building: Problem structuring, policy simulation and decision support, *The Journal of the Operational Research Society* **58**(5) (2007) 691–694.
37. G. Pervan, L. F. Lewis and D. S. Bajwa, Adoption and use of electronic meeting systems in large Australian and New Zealand organizations, *Group Decision and Negotiation* **13** (2004) 403–414.
38. R. O. Briggs, G. L. Kolfschoten, G. J. de Vreede and D. L. Dean, Defining key concepts for collaboration engineering, in *AMCIS*, Vol. 12 (Acapulco, Mexico, 2006).
39. G. L. Kolfschoten, R. O. Briggs, J. H. Appelman and G. J. de Vreede, ThinkLets as building blocks for collaboration processes: A further conceptualization, in *CRIWG*, eds. G. J. de Vreede, L. A. Querrero and G. M. Raventos, LNCS, Vol. 3198 (Springer, Heidelberg, 2004), pp. 137–152.

40. G. J. de Vreede and R. O. Briggs, Collaboration engineering: Designing repeatable processes for high-value collaborative tasks, in *HICSS* (IEEE Press, Waikoloa, 2005).
41. J. A. M. Vennix, Building consensus in strategic decision making: System dynamic as a GSS, *Group Decision and Negotiation* **4** (1995) 335–355.
42. R. O. Briggs, G. J. de Vreede, Jr. J. F. Nunamaker and D. Tobey, ThinkLets: Achieving predictable, repeatable, patterns of group interaction with group support systems (GSS), in *HICSS* (IEEE Press, 2001).
43. G. J. de Vreede, A. Fruhling and A. Chakrapani, A repeatable collaboration process for usability testing, in *HICCS* (IEEE Press, 2005).
44. J. Somervell, Introduction to human computer interaction, http://www.mcs.uvawise.edu/jps5a/courses/326/notes/ch4.pdf (accessed September 2009).
45. J. J. Preece, Y. Rogers and H. Sharp, *Interaction Design: Beyond Human-Computer Interaction*, 2nd edn. (John Wiley & Sons, Chichester, 2007), ISBN: 978-0-470-01866-8.
46. P. Jody, Structured walkthroughs and formal technical reviews, http://www.jodypaul.com/SWE/WT/walkthroughs.html (accessed 5 August 2009).
47. Capgemini, *Accelerated Solutions Environment — The Art of Collaboration: Rapid Solutions Workshop* (Utrecht, The Netherlands, 2005).
48. Holland ASE Team, *The Power of Group Dialog: Accelerated Solutions Environment* (Utrecht, The Netherlands, 2005).
49. J. Mingers and J. Rosenhead, Problem structuring methods in action, *European Journal of Operational Research* **152** (2004) 530–554.
50. J. A. P. Hoogervorst and J. L. G. Dietz, Enterprise architecture in enterprise engineering, *Enterprise Modelling and Information Systems Architecture* **3**(1) (2008) 3–13.
51. J. Zachman, A framework for information systems architecture, *IBM Systems Journal* **26**(3) (1987) 276–292.
52. J. Ross, P. Weill and D. Robertson, *Enterprise Architecture as Strategy: Creating a Foundation for Business Execution* (Harvard Business School Press, Boston, 2006).
53. S. H. Spewak, *Enterprise Architecture Planning: Developing a Blue Print for Data, Applications, and Technology* (John Wiley & Sons Inc., New York, 1992).
54. Gartner, Inc., Gartner identifies ten enterprise architecture pitfalls, Gartner Enterprise Architecture Summit 2009, http://www.gartner.com/it/page.jsp?id=1159617 (accessed 15 August 2010).
55. E. Turban and E. A. Jay, *Decision Support Systems and Intelligent Systems* (New Jersey, Prentice Hall, 1998).
56. N. Green and C. Bate, *Lost in Translation: A Handbook for Information Systems in the 21st Century* (Evolved Technologist Press, New York, 2007).
57. A. Nakakawa, P. van Bommel and E. Proper, Towards a theory on collaborative decision making in enterprise architecture, in *Global Perspectives on Design Science Research* (*DESRIST*), eds. R. Winter, J. L. Zhao and S. Aier, LNCS, Vol. 6105 (Springer, Heidelberg, 2010), pp. 538–541.
58. A. Nakakawa, P. van Bommel and E. Proper, On supporting collaborative problem solving in enterprise architecture creation, in *Practice Driven Research on Enterprise Transformations* (*PRET*), eds. F. Harmsen *et al.*, LNBIP, Vol. 69 (Springer, Berlin, 2010), pp. 156–181.
59. H. Raiffa, J. Richardson and D. Metcalfe, *Negotiation Analysis: The Science and Art of Collaborative Decision Making* (Harvard University Press, Cambridge, 2002), ISBN 0-674-00890-1.
60. H. A. Proper, S. J. B. A. Hoppenbrouwers and G. E. Veldhuijzen van Zanten, Communication of enterprise architectures, in *Enterprise Architecture at Work: Modeling, Communication and Analysis*, ed. M. Lankhorst (Springer, Berlin, 2005), pp. 67–82.

61. T. W. Rehkopf and N. Wybolt, Top 10 architecture land mines, *IT Professional* **5**(6) (2003) 36–43, doi:10.1109/MITP.2003.1254967.
62. T. Blevins and J. Spencer, The open group architecture forum, *Manager's Guide to Business Scenarios* (The Open Group), ISBN: 1-931624-15-1.
63. G. Susman and R. Evered, An assessment of the scientific merits of action research, *Administrative Science Quarterly* **23**(4) (1978) 582–603.
64. R. Baskerville, Investigating information systems with action research, *Communications of the Association for Information Systems* **2** (1999) 19.
65. P. Checkland, *Systems Thinking, Systems Practice — Includes a 30-year Retrospective* (John Wiley & Sons Ltd., New York, 1981).
66. A. Nakakawa, P. van Bommel and E. Proper, Quality enhancement in creating enterprise architecture: Relevance of academic models, in practice, *Practice Driven Research on Enterprise Transformations* (*PRET*), eds. E. Proper, F. Harmsen and J. L. G. Dietz, LNBIP, Vol. 28 (Springer, Heidelberg, 2009), pp. 109–133.
67. R. O. Briggs, B. A. Reinig and G. J. de Vreede, Meeting satisfaction for technology-supported groups: An empirical validation of a goal-attainment model, *Small Group Research* **37** (2006) 585–611.
68. M. Schrage, *Shared Minds: The New Technologies of Collaboration* (Random House, New York, 1990).
69. B. van der Raadt, J. Soetendal, M. Perdeck and H. van Vliet, Polyphony in architecture, *International Conference on Software Engineering* (*ICSE*) (IEEE Computer Society, Edinburgh, May 2004), pp. 533–542.
70. L. F. Lewis and M. F. Shakun, Using a group support system to implement evolutionary systems design, *Group Decision and Negotiation* **5** (1996) 319–337.
71. R. O. Briggs and G. J. de Vreede, *ThinkLets: Building Blocks for Concerted Collaboration* (Delft University of Technology, Delft, 2001).
72. J. F. Jr. Nunamaker, R. O. Briggs, D. D. Mittleman, D. R. Vogel and P. A. Balthazard, Lessons from a dozen years of group support systems research: A discussion of lab and field findings, *MIS* **13**(3) (1996) 163–207.
73. E. A. J. A. Rouwette, J. A. M. Vennix and A. J. A. Felling, On evaluating the performance of problem structuring methods: An attempt at formulating a conceptual model, *Group Decision and Negotiation* **18**(6) (2007) 567–587.