

Enterprise Architecture: Twenty Years of the GERAM Framework

Peter Bernus*, Ovidiu Noran*, Arturo Molina**

* *IIS Centre for Enterprise Architecture Research and Management,
Griffith University, Australia, [P.Bernus, O.Noran]@griffith.edu.au*

** *Tecnológico de Monterrey, Mexico, armolina@itesm.mx*

Abstract: Apart from the 20-year anniversary in 2014 of the first publication of the GERAM ('Generalised Enterprise Reference Architecture and Methodology') Enterprise Architecture Framework, the timeliness of this paper lies in the new interest in the use of systems theory in Enterprise Architecture (EA), and consequently, 'light-weight' architecture frameworks (AFs). Thus, this paper is about the use of systems thinking and systems theory in EA and about how it is possible to reconcile and understand, based on a single overarching framework, the interplay of two major enterprise change endeavours: on the one hand enterprise engineering (i.e. *deliberate* change) and on the other hand *evolutionary, organic* change. The paper also attempts to show how such change processes can be illustrated by employing systems thinking to construct dynamic business models: the evolution of these concepts is exemplified with some past applications in networked enterprise building and more recent proposals in environmental, disaster and healthcare management. Finally, the paper attempts to plot the way GERAM will continue to contribute to society in the context of future challenges and emerging opportunities.

Keywords: Enterprise Integration, Enterprise Modelling, Reference Architecture, Systems Engineering, Complex Systems.

1. INTRODUCTION: THE HISTORY OF GERAM

1.1 The Origins

The late 1970s and early 1980s saw a great effort in the manufacturing industry to design and build highly automated, computer-controlled systems of unprecedented complexity. The scope spanned from numerically-controlled machine tools and computer-aided design of parts and process planning, to complete factory automation featuring workshop and factory level control, scheduling and planning (including production planning and material requirements planning). As a result, the computer-integrated manufacturing (CIM, or 'enterprise integration' (EI)) movement has steadily evolved, increasingly demanding methods and tools for integrating the information and material flow throughout the enterprise.

In 1990, on recommendation from Jim Nevins (a former architect of the Apollo Guidance Computer, and Deputy Director of MIT Charles Stark Draper Lab), the International Federation for Information Processing (IFIP) and International Federation for Automation and Control (IFAC) established a joint Task Force (TF) with the mandate to review existing architecting approaches to EI and to make recommendations to the industrial and research community. At that time, the IFIP constituent of the Task Force was operating under IFIP TC5 (Technical Committee on Computer Applications in Technology), and the IFAC constituent under TC-MI (Technical Coordinating Committee for Manufacturing and Instrumentation').

The TF included representatives from the industrial and research communities, with researchers coming from industry management and consultancy backgrounds. The TF was first chaired by Prof. Emeritus Ted Williams (Director, Purdue Laboratory for Applied Industrial Control) from 1992 to 1996 and later by Assoc. Prof. Peter Bernus (Griffith University) from 1996 until 2002 when it completed its mandate. After 2002, the TF's legacy was inherited by IFIP WG5.12 ('Architectures for Enterprise Integration') and IFAC TC5.3 ('Enterprise Integration and Networking'). The TF reviewed several approaches developed for designing CIM systems and classified them into two categories as outlined below (Williams *et al*, 1994; Bernus *et al*, 1996).

1.2 From Type I (snapshot) to Type II (life cycle) architectures and their generalisation

A first approach was based on generic models, or designs, (called 'architectures') that could subsequently be implemented as information systems (IS) products (or product families) incorporating most or all information-processing tasks in the enterprise (especially referring to its management). At the time, these were called 'Reference Architectures of type I'. Such architectures appealed to vendors because the models produced allowed the definition of a stable set of software products. The resulting implementations were called Enterprise Resource Planning (ERP) systems. In addition, a number of CIM system-specific Reference Models were developed, attempting to systematise the CIM systems' functional building blocks' (such as planning, scheduling and control

system modules), thus giving rise to the set of products called 'Manufacturing Execution Systems' (MES). Unfortunately, the number of competing models was in the order of several dozens, and neither of these achieved an industry-wide acceptance, nor standard status. It is only recently that international standards were developed for the interface between ERP and MES systems, such as the IEC 62264 set of standards for enterprise-control system integration (IEC, 2013).

The second approach developed was based on the recognition that similarly to many engineering disciplines (chemical, manufacturing, software, civil, and systems-), *enterprise* engineering should also be based on a so-called 'life-cycle' approach. Accordingly, to design an integrated enterprise, the enterprise creation activities (and thus methodologies) should extend over the entire life of the enterprise – i.e., from its inception to its decommissioning. Several such architectures were developed by groups with manufacturing systems and Information Systems (IS) background.

Therefore, the TF recommendation was that methodologies for designing and creating CIM systems should be based on architectures with a *life cycle* approach. These architectures were called at the time 'Reference Architectures of type II'.

The TF identified three such architectures: CIMOSA (CIM Open Systems Architecture) (CIMOSA Association, 1996), GRAI-GIM (Graphs with Results and Activities Interrelated) (Doumeingts *et al*, 1998) and PERA (Purdue Enterprise Reference Architecture) (Williams, 1994), and recognised that these three had mutually complementary and useful characteristics. After deliberations, it was proposed that instead of trying to select and recommend a 'single best' reference architecture, the TF would serve the EI community better if it *generalised* its findings and codified these as *requirements that any reference architecture should satisfy*. Thus, authors of various proposed approaches could use these requirements constructively to evolve their own reference architectures.

Based on this idea, Bernus and Nemes (1994, 1996) developed the first such generalisation, which was adopted by the Task Force and called the 'Generalised Enterprise Reference Architecture and Methodology' (GERAM). The first comprehensive report was published in several articles and a monograph (Williams *et al*, 1994; Bernus, Nemes and Williams, 1996). The TF decided to base its further work on this proposal and bring the specification of GERAM to its completion.

1.3 From GERAM to ISO15704: 'Requirements for enterprise-reference architectures and methodologies'

The TF realised that it must form links with appropriate standardisation bodies, and thus became a category 'A' liaison to ISO TC184/SC5/WG1 ('Industrial Automation Systems & Integration / Architecture, Communications, and Integration Frameworks / Modelling and Architecture'). Workgroup (WG)1 was keen to develop standards in the area and endeavoured to formulate the GERAM requirements as an ISO standard.

Between 1995 and 1999 the TF held the majority of its meetings jointly with ISOTC184/ SC5/WG1, chaired by Jim Nell

of the US National Institute of Standards and Technology (NIST). The task standing before the TF was twofold: (1) complete the definition of GERAM, and (2) develop an ISO standard specifying the requirements that an Enterprise Reference Architecture must satisfy. The result was GERAM 1.6.3, released in 1999 and ISO15704, "Requirements for Generalised Enterprise Reference Architectures and Methodologies" (ISO15704, 2000; IFIP/IFAC Taskforce, 2003) developed on this basis. The mandatory section of ISO15704 lists requirements that any enterprise reference architecture must satisfy, while the GERAM document is a public appendix to the standard, exemplifying how to fulfil them.

In subsequent years, the same committee extended ISO 15704 with additional views (decisional and economic), and also released ISO19439:2006, a 'standard' version of the GERA modelling framework (MF).

1.4 Other Relevant Developments in the field of EA

During the same period, the IS, software engineering (SE), and systems engineering (SE) communities saw a number of similar efforts. Thus, for example the IS community widely publicised the Zachman Framework (Zachman, 1987) that initially targeted the enterprise IS, but was subsequently extended to model the entire enterprise's architecture. The evolved Zachman framework had a similar aim to the enterprise reference architectures of type II – although the stakeholder role-related 'rows' in the framework only corresponded to life cycle activities because life cycle abstraction levels were typically of concern to different stakeholders. Later there were several adaptations, including the Federal Enterprise Architecture Framework FEAF (1999).

The Defence community saw the development of the DoDAF (DoD, 2010) Framework, with three simple life cycle abstractions, and a complex MF. DoDAF has several offshoots in the defence community (MODaF, DONDAF, NATOAF, etc).

The IT industry also developed its own framework called The Open Group Architecture Framework (TOGAF) (2006; 2011), which gained popularity with IT consultancies due to its detailed Architecture Development Method (ADM).

Consulting companies, individual researchers, and some vendors also developed their own Architecture Framework (AF) versions (not listed here due to space constraints). The variety of proposals appears to owe to AFs being a conceptual structure suitable for organising content for a domain.

The American Heritage Dictionary of the English Language (2009) defines a 'framework' as "*A set of assumptions, concepts, values, and practices that constitutes a way of viewing reality.*" An EA framework is therefore a means of sense-making in the complex world of change, in the domain of EA. Similarly, as individual learning is a fundamentally constructivist process, organisations must construct their shared meanings of change-related concepts by building the new concepts of EA into the fabric of their own pre-existing, path-dependent concepts of organisational change.

The original terminology used by the manufacturing community called such constructs 'reference architectures'. However, the internationally accepted terminology has evolved;

thus, in line with the above discussion, what ISO15704 / GERAM and its predecessors called a 'Reference Architecture type II', is today called an 'Architecture Framework' (see for example ISO42010:2011). In today's terminology, GERAM is an AF (thus should perhaps more suitably called Generalised Enterprise Architecture Framework, 'GEAF'); the origin of the acronym remains only of historical interest.

Notably, current efforts are underway in the Systems and Software engineering community to develop ISO42030 for 'Architecture Evaluation' (as of today in Committee Draft), which is complementary to ISO15704, defining concepts for the evaluation of an Architecture exhibited by a system.

It would be relevant for respective ISO communities (ISO TC184 and ISO/IEC JTC1) to investigate the benefits in collaborating to fully harmonise these standards; as the underlying philosophy behind these communities is similar, the user community could benefit from such harmonisation.

2. A SUMMARY OF THE CONTRIBUTIONS OF GERAM TO THEORY AND PRACTICE

2.1 GERAM's contribution to the scientific community

The central problem of EA is how to harmonise and synthesise all the knowledge necessary for understanding, sense-making, leading, managing and executing change in enterprises as complex socio-technical systems. In other words, the role of EA is to create coherency across management levels and activity domains in the enterprise (Doucet *et al*, 2008); for an AF to qualify as an *enterprise* AF, its scope must encompass any type of enterprise and constituent entity.

Certainly, a company is an enterprise, but so is a network of companies, or a consortium, a government department, etc. – essentially, any undertaking with a shared goal to produce some services, or products, is an enterprise. An enterprise may live for a longer period of time, or have limited life span, like a project or a program; it can be a single organisation (e.g. a company) or a virtual organisation (virtual enterprise).

GERAM uses the concept of *enterprise entity*, defined as an enterprise or any system of interest in an enterprise (this may include enterprise entities in the environment); thus, even the product of the enterprise is seen as an enterprise entity. For an AF to qualify as an EA framework, the scope of the framework must encompass all enterprise entity types.

The power of this generalisation is that the choice between what is the current system of interest is *orthogonal* to all other choices. E.g., we can talk about the life cycle of an enterprise or the life cycle of a product in exactly the same way.

GERAM as an AF allows a number of elementary differentiations to be made, which can be combined into more powerful concepts as needed. In the GERA MF (which is part of GERAM), the underlying elementary concepts are derived from pairwise differentiations, mandating the ability to describe:

- human & automated constituents,
- service delivery & management constituents, and
- software & hardware constituents

of any enterprise entity, and to do this on any life cycle level. Furthermore, any combination of these can be described from the point of view of the *function*, (and associated objects processed, such as *information and material*), the *resource* performing the function, and of course the mapping between function and resource (called *organisation*). These differentiations define the scope of *potential* enterprise models, aiming for it to be as complete as possible and the framework to remain holistic – considering enterprise entities in their totality.

According to Ted Williams, when the Purdue Consortium set out to write a methodology for CIM systems development, it transpired that separately designing the automated parts of 'material and information processing systems' was too complex a task, and an approach was needed that simultaneously considered the human and automated elements of the enterprise. If a framework, due to its limited scope, insists on subdividing a complex system-of-systems along a border where there are too many interconnections (such as the human-machine interface) then the designer will have limited ability to reduce apparent complexity through decomposition and modelling of interactions among system constituents. Thus, the advice is: '*don't cut it where it is the thickest*' – the first impetus behind the development of the Purdue Enterprise Reference Architecture (PERA) (Williams, 1994).

Finally, GERA states that descriptions (or *enterprise models*) may be created for an individual entity (called *particular models*), for a type of entity (called *reference- or partial models*), as well as the *models of the languages* used for expressing these models. GERAM calls these 'Generic Enterprise Modelling Concepts' as they are essentially *ontologies*, which, in increasing level of formality, may take the form of terminology definitions in natural language, meta-models, or formal ontological theories. The term 'partial model' is preferred over the term 'reference model', because a reference model is often associated with the situation where a particular system of interest is a parametric specialisation (instantiation) of the reference. Partial models on the other hand can be model fragments, building blocks, or patterns.

The above pairwise differentiations can be combined as needed due to their orthogonally to form a large number of model types that *may* be of interest to answer relevant stakeholder concerns about a given enterprise entity.

Thus, instead of creating and prescribing a very long list of model types for every usual type of enterprise entity and typical problem area, and codifying them in the AF, GERAM as an 'AF proper' allows its user to independently contemplate and if necessary specify the problem at hand (as a set of stakeholder concerns), together with model types needed for a solution, and model views for the stakeholders.

GERA is like a 'bookshelf', with labels like the Dewey system used in libraries: 'like' models go to the same shelf. However, even if we pinpoint a model type, many kinds of models can still be produced. For example, given a factory workshop as our target enterprise entity (or 'system of interest'), there exist many options to describe the functional requirements of the manufacturing equipment (machine tool hardware). The model may take the simple form of a list of functions, or a set of static process models, or a set of dynam-

ic process models, etc. The nature of the model is determined by the task at hand, and who needs to peruse the model for what purpose; models are created to answer concerns of stakeholders (ISO42010, 2011).

GERAM's *orthogonality of modelling scope* definitions may be applied in the development of architecture metamodels, to keep them as simple as possible, or possibly discover hitherto hidden aspects of modeling. For example this could be relevant in the current efforts to introduce human view elements into the NATO AF (Handley and Smillie 2008; 2010).

Note that an enterprise model's pragmatic information content (its interpretation by the stakeholders) is also dependent on the model's *status* (Hysom, 2003) in the context, which is the *life history* of the entity. For example, a model may be interpreted as representing the entity 'as perceived by the analyst', or the 'agreed view of AS-IS by all stakeholders', or 'as proposed to be', 'as agreed to be', and so on.

The concept of *life cycle* is fundamental to EA and systems engineering; however, the concept is not well-defined in the majority of literature. At a meeting of two Globeman 21 consortium participants at Sanctuary Cove, in 1997, the Japanese and Australian teams were discussing the life cycle of chemical plants, and the life cycle of the project that designs and builds these plants. Each team had its life cycle representation. The Australians used the GERAM life cycle model, while the Japanese used a representation familiar to the project managers in their engineering firm. It soon became apparent that the two teams did not agree. The advocates of the GERAM representation argued that life cycle does not (and should not) represent time, while the Japanese team's model was clearly temporal. During this crucial meeting it became clear that using the same name for such dissimilar concepts is untenable, and the term 'life history' was proposed to refer to how the life of the respective entities unfolds *in time*.

Shortly after the above events, the IFIP IFAC TF met, and a participant proposed at the end of the meeting that the GERAM life cycle was to be extended with a 're-engineering phase'. However, when at the next TF meeting the life history concept was introduced, it became evident that 're-engineering' is a mere repetition of some life cycle activities: the proposal was dropped, and the life history concept was adopted to be incorporated in the specification of GERAM.

The ISO15704 / GERAM concepts of *life cycle* and *life history* deserve discussion, because no other EA framework has made a differentiation between these concepts, with resulting confusion by architecture practitioners. Even ISO 15288:2008, ('Systems and software engineering -- System life cycle processes'), makes no such differentiation.

The GERAM life cycle consists of life cycle *activity types* (or 'phases'), each of which consider the enterprise entity on a different level of abstraction. These 'phases' are not temporal in nature – if they were, then the term 'cycle' would not be appropriate. There is a two-way information flow among such life cycle activities. The lower the level of abstraction of the phase, the more concrete detail of the entity is known to the corresponding activity. For example, considering the enti-

ty on the 'detailed design' level needs more information than considering the same entity on the 'requirements level' level.

As opposed to the above, the *life history* of an entity describes how changes to the entity happen in time. A life history consists of life cycle *activity instances*, which in turn may form sequences of events (with some parallelism). Also, the timeline of the evolution of an entity may be subdivided into life history *stages* and milestones may also be defined.

Finally, a crucial observation: if, as stated, life cycle is a-temporal, we suddenly realise that the GERAM life-cycle concept can be used in two ways: (1) models used in deliberate enterprise engineering activities (where models on various life cycle levels are utilised for design, to direct change), and (2) models can be produced, on any appropriate life cycle level, to assess and make sense of *emergent* change, i.e. *a posteriori*, as a decision making tool. The difference between these two types of change would be visible as various patterns of life cycle activity instances in the life history of a system. Thus, the life cycle / life history dichotomy allows both researchers and practitioners to discuss deliberate and emergent / organic change within the same framework.

2.2 The contribution of GERAM to the EA community

Given that GERAM is a sense-making instrument rather than a prescriptive framework, it can be used by those who work on the development of their own respective frameworks. ISO15704 was developed to list *requirements* that AFs need to satisfy and not be prescriptive about *how* to achieve this. The logic behind this approach was that the role of the standard was to facilitate development, rather than being an attempt to override the work done by significant groups in the EA community. Saha (2006, p16) writes: "GERAM/ISO15704: 2000 is an excellent baseline to map and assess candidate architecture frameworks..." and refers to GERAM as a meta-framework (a language to talk about frameworks), or 'framework of frameworks'.

A number of EA frameworks were analysed and mapped against the ISO15704 requirements (Noran, 2003; Saha, 2004). The conclusion is that, what is often published under the name 'Architecture Framework', is a collection of artefacts that includes an 'AF proper' (as defined in Section 1.4) *plus* a set of artefacts described by that AF. For example, an AF proper may define the concept 'EA methodology', but any such methodology is not part of the AF proper. For a defined type of stakeholder community and typical application, AFs may come *bundled* with a collection of artefacts, such as architecture description languages (modelling languages), an architecting methodology, a set of reference models, or even a suggested modelling tool. A taxonomy could be built for any of these; for example, a factory design methodology may have specialised versions for typical factory types.

The content of these artefacts (methodologies, principles, reference models, etc.) is normally not developed by the EA discipline. Thus, a methodology to develop the business model of an enterprise is likely to come from management science, reference models for projects and programmes from the project management discipline, process reference models for IT management from the IS community, and so on.

Prescribing a methodology even for a typical EA problem can quickly become complicated, as the author must prepare advice for *all* types of situations, contexts, and user skill levels. An alternative is to prescribe a *meta-methodology*, which is a more economical approach (see Section 3.1).

In summary, the EA discipline is (or should be) an interdisciplinary *synthesis* of the contributions of underlying disciplines. The EA terminology and a theory of EA must be able to describe the interactions among theories, methods, models, and tools developed by these disciplines and *how new insights and practices can emerge* through such interaction. Thus, EA can be seen as *the systems science of change*.

For EA researchers, GERAM can also be used as a research framework, with the view of circumscribing the scope of research and formalising the research problem at hand – in other words, to translate the initial informal research question into a formal one (Vesterager *et al*, 2000; Noran, 2007; Magoulas *et al*, 2010; Chaharsooghi *et al*, 2011).

2.3 Manufacturing and Engineering Applications

The first industrial application of GERAM was in the Globeman 21 Consortium of approximately 40 industry partners that demonstrated the use of EA principles in a number of applications in the manufacturing industry. The final report of the consortium (Brown and Syntera, 1999) outlined the most important outcomes as follows:

- 1) The Virtual Enterprise Workbench for World-wide Integration & Development (VIEWBID), providing a company with an integrated set of tools and methods for co-ordinating its global business process for bidding / tender preparation;
- 2) Virtual and Real Information Technologies driven Global Engineering/ Enterprise (VRIDGE) - a virtual enterprise (VE) that carries out the design, procurement, construction, and manufacturing of a chemical plant. This VE facilitated the understanding of global business processes and the investigation of requirements for global product information access and control, as well as of IT infrastructure for VEs.
- 3) GlobOS, who demonstrated a concept of networking among shipyards where pre-qualified partners in the network deliver different generic steel structure assemblies for the final ship. GlobOS illustrates how facilities, methods and assemblies can be modelled and shared within a network.

The subsequent Globemen consortium of 16 industry and 3 academic partners focused on inter-enterprise collaboration based using global information exchange and control (Karvonen *et al*, 2003). Globemen demonstrated the applicability of the GERAM framework (specialised by the consortium under the name VERAM) to the creation of global enterprise networks that are capable, on demand, of creating *service* virtual enterprises (Vesterager *et al*, 2000; 2001) that provide after-sales service (troubleshooting, turnaround, factory optimisation, etc.) in a transparent way to chemical factories, although each service typically requires the competencies of multiple globally distributed companies.

GERAM has also been applied to support the development of Small and Medium Enterprises (IMMPAC) (Molina and Car-

rasco, 2003), to design and create the concept of Virtual Industry Cluster (Molina and Flores, 1999), and technological platforms such as PyME CREATIVA (Molina *et al*, 2006; Giraldo *et al*, 2007; Nogueira *et al*, 2013), and to build an EI engineering reference framework and toolbox to improve the design and operations of manufacturing companies (Vallejo, Romero and Molina, 2011).

GERAM was the methodological basis for the development of a reference model for Integrated Product and Process development (IPPMD) and which was subsequently applied in multiple industry cases (Pereda and Molina, 2013). Three case studies have been published; the first is related to a product transfer for an aeronautical firm to a Mexican manufacturing company. The second case study was implemented in a manufacturing firm to automate a planning process for design and manufacture of an automotive part.

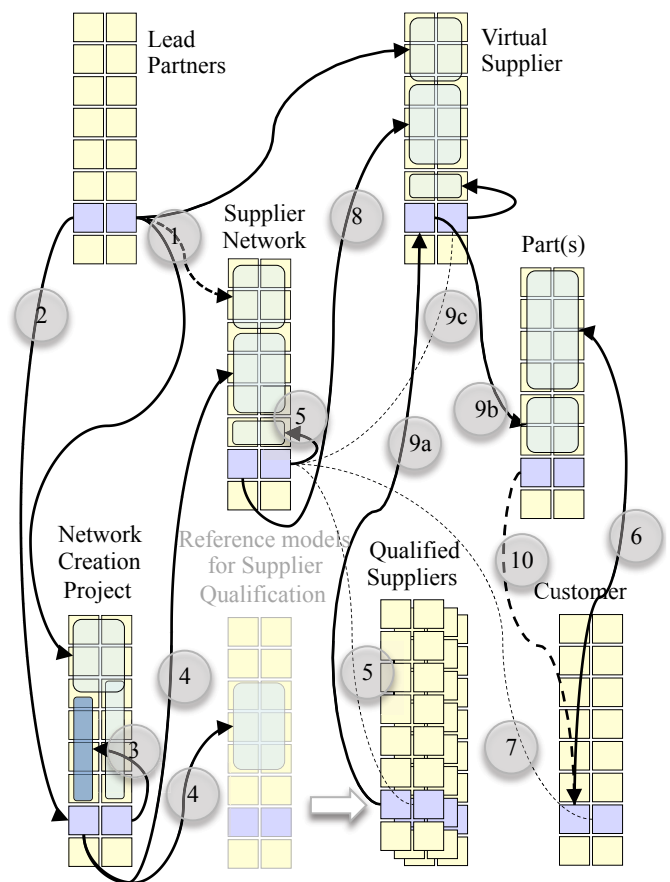


Fig. 1. A Dynamic Business Model of the creation of an Enterprise Network and its Virtual Enterprises (VEs)

Finally, the last case (illustrated in Fig.1) describes how a Small or Medium Enterprise (SME) created its development process for a new product. Thus, IECOS is a Virtual Supplier integrating qualified suppliers into a network, that according to their website is “an engineering and manufacturing firm that focuses on developing high added value solutions for [...] clients and allies through three different business units: IECOS Supply-, Engineering- and Technology Services”. ‘IECOS Supply’ integrates the capabilities of more than 20 small and medium-sized metal-mechanical and plastics manufacturing companies (Molina *et al*, 2007).

2.4 Applications to the Standards Community, Environmental and Emergency Management, and Collaborative Healthcare

The wide applicability of GERAM as an overarching, non-prescriptive ('light-weight', see Section 3) AF has allowed its involvement in conjunction with systems thinking and system-of-systems paradigms to address several major challenges faced by society today.

For example, in the standards community, currently there is limited, mainly ad-hoc collaboration between custodian work groups within each technical and scientific committee, causing low awareness of other groups' work and resulting in gaps and overlaps in standards' scope and inconsistencies in standards' glossaries. This makes standards difficult to use together – which is typically expected for any sizeable project. GERAM has been proposed as an essential tool in discovering gaps and overlaps in the areas of relevance and building an intelligent repository underlying an expert system that allows achieving an improved and *sustainable* interoperability of standards (Noran, 2012).

Environmental management (EM), an increasingly essential aspect of any enterprise or network, is often not properly integrated in the business, resulting in sub-optimal environmental performance and merely minimal compliance with standards. GERAM was used to support the discovery of the areas and level of EM integration and to ensure that environmental information of adequate quality and detail is promptly delivered to the management to properly support the long and short-term decisional processes (Noran, 2009).

Emergency management is another field of application, in the context of climate change and increasing rate of natural and man-made disasters. Here, GERAM has been used to provide a checklist of participants and aspects required in modelling the present (AS-IS) and desired future (TO-BE) states of emergency management services (Noran and Bernus, 2011). This was done with a view to improve the effectiveness and resilience of the command and control and response teams.

In addition, Noran and Panetto (2013) proposed the use of GERAM and systems thinking principles in the lifecycle-centric analysis and design of improvements to healthcare interoperability and collaboration, to meet long term challenges (aging population, patient safety, complexity of services) and improve response to short term, acute challenges (large scale medical emergencies such as pandemics).

Further applications of GERAM continue to emerge, such as enterprise sustainability (Alves *et al.*, 2013; Kandjani, Bernus and Nielsen, 2013), and in preparedness building for Mergers and Acquisitions (Vaniya, Bernus and Noran, 2013).

3. 'HEAVY WEIGHT' VS. 'LIGHT WEIGHT' ARCHITECTURE FRAMEWORKS

3.1 The complexity conundrum

Readers following the development of AFs presented above would have noticed the following contradiction:

i) AFs are intended to help simplify the complex and thus sometimes unpredictable and sensitive enterprise change processes ('change' may mean incremental or major change, or the engineering and construction of a green field installation);

ii) Change, including deliberate (engineered and managed), has multiple aspects and numerous stakeholders, with many concerns and relationships; an AF must address all of these to achieve the desired coherency of decision-making and action.

The conundrum lies in the fact that according to the above logic, the 'AF that can address everything' would likely be complex, thus creating *unpredictability*, whose avoidance was the primary objective in the first place (Kandjani, Bernus and Nielsen, 2013; Kandjani and Bernus, 2013).

If the AF is too complex for the stakeholders to fully understand, then how can we use it to solve the problem, when it creates another problem of similar nature? The question is: *is it possible to have a simple AF as a basis for shared understanding and the development of an approach that can be used to adequately manage the complexity of change?*

The objective of leading and managing change cannot be deterministic control, because enterprises operate in an environment capable of producing chaotic, non-deterministic behaviour. An EA methodology should help management navigate this partially determined, partially controllable world, steering change into a future where some important characteristics will hold, without having to predict the *exact* future trajectory and *exact* future state. With a vision of the future, management must *channel* change in a desirable direction, and an EA methodology should help achieve this. Part of this channelling is the establishment of change governance (with principles and processes that work in parallel with the change) without *a priori* prescribed sequence of actions.

When members of the Globemen Consortium requested a step-by-step methodology for developing an enterprise network that creates 'Service Virtual Enterprises', we realised that a pragmatically useful generic methodology is too hard to produce, due to the complexity of preparing for all possible cases. We saw, however, that for a fixed set of involved entities (a leading engineering firm, some representative factory owners, OEM vendors, and typical local service providers) whose life cycle relationships were determined (i.e., which entity should contribute to which life cycle activity of the another), a relatively simple, customised methodology could be defined (Bernus *et. al.*, 2002; Noran, 2004).

Thus, the answer to the dilemma seems to be not to bundle prescribed generic methodologies with AFs, but to use a simple framework and a *meta-methodology* to adaptively create up-to-date change processes, combining proven methods, techniques and tools of underlying disciplines as necessary.

Some frameworks have a built-in methodology (e.g. TOGAF ADM), but methodology should be tailored to the specific case. Thus, the ADM is a *repository* of techniques and methods, and a meta-methodology should be used to identify the parts needed for a particular case.

The elements of the simple framework may be described using a (meta) meta-model (Bernus and Noran, 2010), but care

must be taken not to confuse (or mix) with meta-models expressing the semantics of enterprise modelling languages.

According to the same argument we must avoid creating a 'heavy weight' AF by prescribing model deliverables. However, for typical problems one can define a methodology, with associated models and languages.

3.2 Systems Science and Systems thinking in EA

It is worth illustrating how systems thinking can be applied to locate the need for change and establish the structure of a model that is underlying a meta-methodology.

A typical case is continuous improvement, whereupon someone in the operations notices a problem (on the shop floor or in management) and is searching for a solution by analysing the observed problem and its causes. One can use a systems thinking diagram, or one of the many types of root cause analysis tools, for this to identify the entities involved, those affected, and the location of needed change.

Figure 2 shows a systems thinking diagram that a lead partner may have used to represent the problem of decreasing orders. The systems thinking diagram identifies the root cause: the SME's limited capability set to fulfil complex orders.

The solution is to create a virtual entity that *looks* like a medium sized supplier, able to perform complex operations. One way to achieve this is to create a network, in which processes, information flow, planning and scheduling, performance and resource management are coordinated. This needs tactical, and possibly strategic information sharing beyond operational interactions, which is only viable if partners are pre-qualified and trusted and they subscribe to the network's operating policies and principles (finances, contracts, technological compatibility, quality and process). Thus the need for a tightly regulated network, which in turn can create the desired virtual manufacturing company that looks like a single well-managed medium sized supplier. This thinking is useful to understand the need for transformation (change, or innovation); subsequently, a *dynamic* business model can be used to represent transformation from the present (AS-IS) to the chosen future (TO-BE).

The change process can be modelled as a 'dynamic business model', a special type of systems thinking diagram of action and outcome used to consider what *may* have to be performed as part of a change. In this model, each entity is represented by its life cycle, and each arrow connects the entity that performs some change activity to the life cycle activity(ies) of the entity being changed. We used such diagrams in practical projects to define complex engineering and change processes, and found that despite a seemingly complicated structure, the model became a collaborative thinking tool, to define the scope and location of change.

Figure 1 previously shown is a model of action and outcome leading to the creation of a Virtual Supplier by a network of qualified suppliers; continuous arrows represent 'generative relationships', while dotted arrows represent operational interactions.

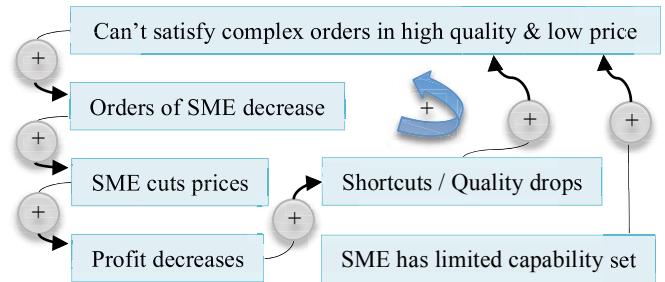


Fig. 2. Systems thinking diagram illustrating why is it hard for SMEs to compete

The Systems Story unfolds like this: the lead partner(s) identify the entities to be part of the business model, and the concept and mandate of each, and sets up a Network Creation Project (1). Lead partner(s) contribute to this project (2). Project management designs the details of the project and oversees project execution (3). The tasks are to design the supplier network, and a set of reference models (functions, processes, standards, policies and principles etc.) that a qualified member of the network will have to follow (4). Network management builds the network from qualified partners (5).

When a potential customer designs a part (6), eaving some or all detailed design decisions to the supplier, it contacts the network for a quote (7). If successful, the Network Office creates a 'Virtual Supplier' (8), assembled from qualified suppliers compliant with the reference models (4). The selected suppliers contribute to the Virtual Supplier's processes (9a) to perform the detailed design, production and delivery, so as the parts eventually reach the Customer (10). Note that the VE does have its own management function, but its operations are also overseen by the Network Office (9c).

4. WORK FOR THE FUTURE

4.1 Why is it necessary to work on the future of EA?

Today's enterprises struggle with the number of fragmented models, tools and methods proposed to them by multiple disciplines, and the resulting adoption is less than coherent (Doucet *et al*, 2008). We believe that the future of EA is in its ability to develop an interdisciplinary language and theory enabling a concerted and synergistic application of contributions of underlying disciplines (results from management science, systems, industrial, manufacturing and software engineering, IS, Artificial Intelligence, and so on).

GERAM is a valuable *baseline meta-framework* - to discuss the above, to create new theories, schools of thoughts, integration of engineering practices and tools, explanations of how technologies can be part of EA, methodologies for EA implementation, and Partial and Particular models for re-use.

4.2 The evolution of the Science of EA

EA is an evolving discipline, including and evolving Theory, Framework, Standards, Methodologies, Tools and Models. First and foremost the EA community needs to *continuously improve and enrich the theory of EA*, for it to remain relevant and useful. As expressed in Section 2.2: *EA should be the systems science of change*. As a consequence, we must:

- Develop a theory and harmonise the contributions of the above disciplines, so that all stakeholders (not only architects) in enterprise change and evolution can use EA as a common language (Kandjani and Bernus, 2013).
- Develop recommendations of who and how should *learn this harmonised language*, contribute to *harmonise EA-related international and industry standards*, and continue interpretation and *mapping* of particular frameworks as they develop, in keeping with the common language,
- Extend the theory to cover the *interplay* of *evolution* and *enterprise engineering*, where complexity management plays a crucial role, and extend the scope to include social and ecological systems.

4.3 Emerging Technologies and EA

- Keep demonstrating the usefulness of the EA language and theory by developing commonly understandable *interpretations of existing and new technologies*, so practitioners' use, as a tool for industry analysts to moderate the 'hype cycle' of new technologies. E.g., one could start with explaining the role in EA in 'big data analytics', 'cloud computing', 'sensing enterprise', etc.
- *Re-interpret* notable and important *architectural styles, schools of thought, and reference models* using EA's common language, to make these accessible, comparable and synergistic. This is a very large agenda, which needs the help of underlying disciplines. However, the outcome can be a 'LEGO' of components (models, tools, languages, methodologies) enabling innovation.
- Notable reference models / Reference Architectures exist for service delivery, management & control, or both. The following is an example list of well known ones (with bibliographic references omitted): Service orientation on the IT level and on the Business level, Fractal organisations, Model Driven Architectures to instantiate process oriented organizations, governance models (of IT and of EA), process improvement (Kaizen, CMMI, SPICE, Six Sigma), Reference Architectures for Programme and Project management, etc.

4.4 Applications, Enterprise Integration Engineering and Enterprise Architecture

- Continue research on Architecture principles, and conduct and publish architecture evaluations (*cf* ISO 42030), and comparative studies to demonstrate value.
- Publish case studies and solutions to contribute to knowledge of the EA practice, e.g. challenges to introduce EA practice (soft factors, human interactions, change management, interoperability, etc.)
- Enrich the relevance of the discipline by publishing case studies of EA in government defence, petroleum, cement, retail, global companies, , health care, etc.
- Develop problem-specific methodologies, arising from *scale* (micro-enterprises, SMEs, global companies) and *type of change* (for implementing new business models, work practices, automation, M&As, new management style, reorganisation, productivity improvement), so as to

reflect a relevant sample of typical stakeholder concerns and disseminate solutions to address them.

4.5 EA Community Building:

The community is fragmented, by *industry* (IT/Systems Engineering, Manufacturing, Public Sector, Defence, Service industries), *nature of enquiry* (scientific/applied,) and by *schools of thought*. Communities include academia (IFAC, IFIP, IEEE), Research Groups, Industry Associations & standards bodies (Open Group, INCOSE, ISO, NIST, DIN), consortia, consulting firms, CIO councils, etc. These communities have their own conferences or other fora, but there is no forum to bring these together. We believe that a revival of the ICEIMT-style conference and workshop series could play a role in creating coherence among the existing 'EA silos'; perhaps EA practice could be used to provide an answer.

DEDICATION

This article is dedicated to the memory of Jim Nevins and Ted J. Williams.

REFERENCES

- Bernus, P., Nemes, L. (1994) A Framework to Define a Generic Enterprise Reference Architecture and Methodology. In Proc. ICARV'96 Vol3/3. Nanyang TU. pp88-92. Reprinted in (1996) in *Comp Int Mfg Sys* 9(3):179-191.
- Bernus, P., Nemes, L., Williams, T.J. (Eds) (1996) *Architectures for Enterprise Integration*. Chapman & Hall.
- Bernus, P., Noran, O. (2010) A Metamodel for Enterprise Architecture. in *IFIP AICT*, 326. Berlin:Springer :56-65
- Bernus, P., Noran, O., Riedlinger, J. (2002) Using the Globemen Reference Model for Virtual Enterprise Design in After Sales Service. In Karvoinen, I. *et al* (Eds). *Global Engng and Mfg in Enterprise Networks* (Globemen), *Symposium Series 224*. Helsinki : VTT. :71-90.
- Brown, R.H., Syntera, H. (Eds) (1999) *Globeman 21* (Global Mfg. in the 21st Century) Final Report. IMS Programme. http://www.ims.org/wp-content/uploads/2011/11/2.4.10_2-Final-Report-GLOBEMAN-21.pdf
- Chaharsooghi, K., Achachlouei, M.A. (2011) Developing life-cycle phases for the DoDAF using ISO15704 Annex A (GERAM). *Computers in Industry*. 62(3):253–259
- DoD (2010) DoD Architecture Framework Version 2.02. US Department of Defense.
- Doucet, G., Götze, J., Saha, P., Bernard, S (2008) Coherency Management: Using Enterprise Architecture for Alignment, Agility, and Assurance. *Journal of EA* :1-12
- Doumeings, G., Vallespir, B., Chen, D. (1998). GRAI Grid Decisional Modelling. In P. Bernus, K. Mertins and G. Schmidt (Eds.) *Handbook on Architectures of Information Systems*. Berlin : Springer. pp313-339.
- FEAF (1999) Federal Chief Information Officers Council: Federal Enterprise Architecture Framework. Version 1.1.
- Franco dos Reis Alves, D., Campos, R., Bernardi de Souza, F. (2013) GERAM: Building Sustainable Enterprises. 6th IFAC Conference on Management and Control of Production and Logistics (MCPL13), Fortaleza, Brazil.
- Giraldo, J., Galeano, N., Molina, A. (2007) Virtual Organization Breeding Environment: Experiences from its Implementation in Mexico, in Proc. 8th IFAC Symposium

- on Cost Oriented Automation Affordable Automation Systems, La Habana, Cuba (CD Memories)
- Handley, H.A.H., Smillie, R.J. (2008) Architecture framework human view: the NATO Approach. *Systems Engineering*. 11(2):156-164.
- Handley, H.A.H., Smillie, R.J. (2010) Human view dynamics - the NATO approach. *Systems Engineering*. 13(1):72-79
- Hysom, R. (2003) Enterprise Modelling – The Readiness of the Organisation. In Handbook on Enterprise Architecture. Berlin : Springer. pp373-416
- IEC62264 (2013) Enterprise-control system integration. ISO.
- ISO 19439(2006) Enterprise integration – Framework for enterprise modelling
- ISO/IEC/IEEE 42010 (2011) Systems and software engineering – Architecture description. Geneva : ISO
- ISO15704 (2000) Industrial automation systems – Requirements for enterprise-reference architectures and methodologies; (Amd1 2005) Additional views for user concerns
- Kandjani, H., Bernus, P. (2013) The Enterprise Architecture Body of Knowledge as an Evolving Discipline In Maciaszek, L.A., Cuzzocrea, A. and Cordeiro, J. (Eds.) LNBP 141, Berlin, Heidelberg : Springer. pp452-471
- Kandjani, H., Bernus, P. and Nielsen, S., (2013). Enterprise Architecture Cybernetics and the Edge of Chaos: Sustaining Enterprises as Complex Systems in Complex Business Environments. In Ralph H and Sprague Jr. (Ed). Proc HICSS2013. IEEE Xplore. pp3858-3867.
- Karvonen, I. et al (Eds) (2003) Global Engineering and Manufacturing in Enterprise Networks GLOBEMEN. VTT Symposium Vol.224. Helsinki : VTT Technical Research Centre of Finland <http://www.vtt.fi/inf/pdf/symposiums/2003/S224.pdf>
- Magoulas, Th., Hadzic, A., Saarikko, T., Pessi, K. (2010) Alignment in Enterprise Architecture: investigating the Aspects of Alignment in Architecture Approaches. in Proc 5th European Conf. on Information Management and Eval. Reading : Acad. Publ. Ltd. pp321-331
- Molina, A., Carrasco, R. (2003) The use of GERAM to support SMEs development in Mexico, in Handbook on Enterprise Architecture. Berlin : Springer. pp757-778.
- Molina, A., Mejía, R., Galeano, N., Najera, T., Velandia, M. (2006) The HUB as an enabling IT strategy to achieve Smart Organizations. In Mezgar, I. (Ed.) Integration of ICT in Smart Organizations. Idea Group. pp64-95
- Molina, A.; Velandia, M. and Galeano, N. (2007) Virtual Enterprise Brokerage: A Structure Driven Strategy to Achieve Build to Order Supply Chains. *International Journal of Production Research*. 45(7): 3853-3880
- Nogueira, J.M., Romero, D., Espadas, J., Molina, A. (2013) Leveraging the Zachman Framework Implementation Using Action-Research Methodology – A Case Study: Aligning the Enterprise Architecture and the Business Goals. *Enterprise Information Systems*. 7(1):100-132
- Noran, O. (2003). Mapping of Individual Architecture Frameworks. in Handbook on Enterprise Architecture. Berlin : Springer. pp65-210
- Noran, O. (2004) Towards a Meta-methodology for Collaborative Networked Organisations. In L. Camarinha-Matos (Ed.) IFIP AICT Vol 149:71-78.
- Noran, O. (2007) Using Reference Models in Enterprise Architecture. in Fettke, P. and Loos, P. (Eds). Reference Modelling for Business Systems Analysis. Hershey : Idea Group. pp141-165
- Noran, O. (2009) Engineering the Sustainable Business: An Enterprise Architecture Approach. In Doucet, G., Gotze, J., Saha, P. (Eds.), Coherency Management: Architecting the Enterprise for Alignment, Agility, and Assurance. Int. Enterprise Architecture Institute. pp179-210
- Noran, O. (2012) Achieving a Sustainable Interoperability of Standards. *Annual Reviews in Control*. 36(2).
- Noran, O., Bernus, P. (2011) Effective Disaster Management: An interoperability Perspective. *LNCS*. 7046:112-121.
- Noran, O., Panetto, H. (2013) Modelling a Sustainable Cooperative Healthcare: An Interoperability-driven Approach *LNCS*. 8186:238-249.
- Pereda, F.J., Molina, A. (2013) Model driven architecture for engineering design and manufacturing. *IFAC-PapersOnline*, Vol.6. Part 1. pp400 – 407.
- Saha, P. (2004) Analyzing TOGAF from the GERAM Perspective. http://www.opengroup.org/architecture/wp/saha/TOGAF_GERAM_Mapping.htm
- Saha, P. (Ed) (2006) A synergistic assessment of the Federal Enterprise Architecture Framework against GERAM (ISO15704:2000). in Handbook of Enterprise Systems Architecture in Practice. IGI Global.
- Taskforce, IFIP-IFAC (2003) The Generalised Enterprise Reference Architecture and Methodology. in Handbook on Enterprise Architecture. Berlin : Springer. pp22-64.
- The American Heritage Dictionary of the English Language, 4th Edition (2000, Updated 2009) Houghton Mifflin.
- The Open Group. (2006; 2011) The Open Group Architecture Framework. v8.1.1; v9.1. The Open Group.
- Vallejo C., Romero D., Molina A. (2011) Enterprise Integration Engineering Reference Framework & Toolbox. *Int J. of Production Research*.50(5):1489-1551
- Vaniya, N., Bernus, P., Noran, O. (2013) Examining Potentials of Building M&A Preparedness. in Proc. 15th Int. Conf. on Enterprise Information Systems. Vol3. Setubal : SCITEPRESS. pp201-212.
- Vesterager, J., Bernus, P., Larsen, L.B., Pedersen, J.D., Tølle, M. (2000) Use of GERAM as Basis for a Virtual Enterprise Framework Model. In Mo, J., Nemes, L. (Eds) Global Engineering, Manufacturing and Enterprise Networks (DIISM2000) Dordrecht : Kluwer. pp75-82
- Vesterager, J., Bernus, P., Pedersen, J. D., Tølle, M. (2001) The what and why of a Virtual Enterprise Reference Architecture. in E-work and E-commerce, Novel solutions and practices for a global networked economy. Amsterdam : IOS Press. pp846-852.
- Williams, T.J. (1994) The Purdue enterprise reference architecture. *Computers in Industry*. 24 (2):141-158.
- Williams, T.J., Bernus, P., Brosvic, J., Chen, D., Doumeingts, G., Nemes, L., Nevins, J.L., Vallespir, B., Vlietstra, J., Zoetekouw, D. (1994) Architectures for Integrating Manufacturing Activities and Enterprises. *Computers in Industry. Special Issue on CIM*. 24(2-3):111-140
- Zachman, J. (1987) A Framework for Information Systems Architecture. *IBM Systems Journal*, 26(3):276-292.