

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

Chapter 1 : Lean Enterprise Architecture Framework (LEAF) - Enterprise Development

The book then shows how to document and build an Enterprise Architecture: architecture principles, design method, best practices, delivery checklists, EA patterns, development process, tooling and repository, integration to the mundane solution architectures, typical use cases, a simple EA development example, how to measure the EA maturity and value delivered, what are the typical EA triggers and roadblocks and how to build an EA practice, its governance, organization, funding and site.

Free preview of Gartner research Already have a Gartner account? Sign in to view more Gartner research. One of our first orders of business was to unify our research positions. We convened a working group of analysts with a mandate to identify and resolve positional differences, with an eye toward maintaining legacy continuity for both sets of customers. However, we recognized that this was an excellent opportunity to advance our thinking in the enterprise architecture space, and we expected new dimensions and insights to be developed as part of the unification efforts. Gartner and Meta always agreed, in broad strokes, on the principles of enterprise architecture. We agreed that it is a "top-down" discipline that derives the optimum constellation of business, information and technology to support the business strategy. We agreed that any "solution" required business, information and technology components to interoperate in support of business capabilities. We agreed that the future state architecture should be developed before the current state is documented, and that no architecture was useful unless it included a gap analysis and an actionable road map for getting from where you are to where you want to be. We agreed that good enterprise architecture does not happen in a vacuum, but that it is one of several strategic planning disciplines that organizations should practice to align their technology with their business strategy. However, there were also some significant differences. Reconciling these secondary products carried some challenges, because we had to flesh them out in more detail to understand where the similarities and differences were. We believe that successful enterprise architecture programs are process-focused, so we started with the Enterprise Architecture Process Model. Our objective was to accurately reflect the activities associated with the initiation and implementation of an enterprise architecture program. The result is a multiphase, iterative, nonlinear model focused on enterprise architecture process development, evolution and migration, along with governance, organizational and management subprocesses. It represents a synthesis of best practices of how the most successful organizations have developed and maintained their enterprise architectures, and it is described in detail in "Gartner Enterprise Architecture Process: Enterprise architecture is a complex subject with abstract components â€” frameworks are important because they provide a context within which the organizational thinking can be structured, and consistent use of a framework in all components of an enterprise architecture program is a best practice. A good framework will define the components of an enterprise architecture and the relationships between them, providing the architecture team and the organization a set of shared semantics and concepts with which to describe their architecture. The Gartner Enterprise Architecture Framework defines a "business context" consisting of the business strategy and external trends that provide the overall context for the enterprise architecture. We advocate the development of a minimum of three interdependent viewpoints: The aspect-oriented approach allows for the articulation of additional viewpoints, should the organization require them. An important aspect of the Gartner enterprise architecture framework is the recognition that the "solution architecture," where the systems that support the enterprise are actually specified and designed, takes place at the intersection of the viewpoints. One such component is the legacy Gartner concept of "bricks" â€” those elemental technology building blocks. We felt that the life cycle management concepts that were embedded in the brick were worth retaining. Another construct that we retained is the architectural pattern â€” a concept that both organizations had written about extensively. Patterns are the best vehicle that we know of for communicating the guidelines of the architecture to developers and implementers. We explore the concept of patterns and how it fits into the new framework and process in "Enterprise Architecture Patterns: No

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

enterprise architecture is effective unless it is implemented, and no enterprise architecture is implemented unless the vision of the future, guidelines and standards are embedded into the decision-making processes of the organization. We explore the relationship between enterprise architecture, IT strategic planning, IT portfolio management and enterprise program management in "Enterprise Architecture Improves IT Planning Synergies. Enterprise architecture is a wide topic area with much opportunity for additional research and commentary. Our new view on enterprise architecture provides us with a solid base for expanded research into all of the aspects of the discipline" and it provides our clients with valuable insight into the best practices for building a well-defined, well-aligned enterprise architecture in a mature, productive enterprise architecture program. Reproduction and distribution of this publication in any form without prior written permission is forbidden. The information contained herein has been obtained from sources believed to be reliable. Gartner disclaims all warranties as to the accuracy, completeness or adequacy of such information. Although Gartner's research may discuss legal issues related to the information technology business, Gartner does not provide legal advice or services and its research should not be construed or used as such. Gartner shall have no liability for errors, omissions or inadequacies in the information contained herein or for interpretations thereof. The opinions expressed herein are subject to change without notice.

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

Chapter 2 : The Open Group Architecture Framework - Wikipedia

The framework looks like a content page showing the chapters of a book or, in this case, the components of the Enterprise Architecture without actually describing them but showing how they fit into the whole.

Document Results in Accordance with Decision-Maker Needs The high-level, 6-step architecture development process provides guidance to the architect and Architectural Description development team and emphasizes the guiding principles. The process is data-centric rather than product-centric e. This data-centric approach ensures concordance between views in the Architectural Description while ensuring that all essential data relationships are captured to support a wide variety of analysis tasks. The views created as a result of the architecture development process provide visual renderings of the underlying architectural data and convey information of interest from the Architectural Description needed by specific user communities or decision makers. The figure above depicts this 6-step process. It is important to note that the development of Architectural Description is an iterative process and a unique one, in that every Architectural Description is: Different in that architecture creation serves a specific purpose, and is created from a particular viewpoint. Serving differing requirements, necessitating different types of views to represent the collected data. Changeable over time as requirements become more focused or additional knowledge about a process or requirement becomes known. The methodology described below is designed to cover the broadest possible set of circumstances, and also to focus on the most commonly used steps by the architecture community.

Determine Intended Use of Architecture. Defines the purpose and intended use of the architecture "Fit-for-Purpose" ; how the Architectural Description effort will be conducted; the methods to be used in architecture development; the data categories needed; the potential impact on others; and the process by which success of the effort will be measured in terms of performance and customer satisfaction. This information is generally provided by the process owner to support architecture development describing some aspect of their area of responsibility process, activity, etc. Determine Scope of Architecture. While many architecture development efforts are similar in their approach, each effort is also unique in that the desired results or effect may be quite different. As an example, system development efforts generally focus first on process change, and then concentrate on those automated functions supporting work processes or activities. Information collected for Architectural Descriptions describing services is similar to information collected for Architectural Descriptions describing systems. Similar situations occur with Architectural Description development for joint operations. Joint capabilities are defined processes with expected results, and expected execution capability dates. The Architectural Descriptions supporting the development of these types of capabilities usually require the reuse of data already established by the military services and agencies, analyzed, and configured into a new or updated process that provides the desired capability. These types of data are presented in models. The important concept for this step is the clarity of scope of effort defined for the project that enables an expected result. Broad scoping or unclear definition of the problem can delay or prevent success. The process owner has the primary responsibility for ensuring that the scoping is correct, and that the project can be successfully completed. Clarity of scope can better be determined by defining and describing the data to be used in the proposed Architectural Description in advance of the creation of views that present desired data in a format useful to managers. Early identification of needed data, particularly data about the Architectural Description itself, the subject-matter of the proposed Architectural Description, and a review of existing data from COIs, can provide a rich source for ensuring that Architectural Descriptions, when developed, are consistent with other existing Architectural Descriptions. It also ensures conformance with any data-sharing requirements within the Department or individual COIs, and conformant with the DM2. An important consideration beginning with this and each subsequent step of the architecture development process is the continual collection and recording of a consistent, harmonized, and common vocabulary. The collection of terms should continue throughout the architecture development process. As architectural data is

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

identified to help clarify the appropriate scope of the architecture effort, vocabulary terms and definitions should be disambiguated, harmonized, and recorded in a consistent AV-2 process documented in the "DoDAF V2. The required level of detail to be captured for each of the data entities and attributes is determined through the analysis of the process undergoing review conducted during the scoping in Step 2. This includes the data identified as needed for execution of the process, and other data required to effect change in the current process, e. These considerations establish the type of data collected in Step 4, which relate to the architectural structure, and the depth of detail required. The initial type of architectural data content to be collected is determined by the established scope of the Architectural Description, and recorded as attributes, associations, and concepts as described in the DM2. A mapping from DM2 concepts, associations, and attributes to architecture models suggests relevant architectural views the architect may develop using associated architecture techniques during the more comprehensive and coherent data collection of Step 4. This step is normally completed in conjunction with Step 4, a bottom-up approach to organized data collection, and Architectural Description development typically iterates over these two steps. As initial data content is scoped, additional data scope may be suggested by the more comprehensive content of Architectural Views desired for presentation or decision-making purposes. This step can often be simplified through reuse of data previously collected by others, but relevant to the current effort. Work is presently underway within the Department to ensure uniform representation for the same semantic content within architecture modeling, called Architecture Modeling Primitives. The Architecture Modeling Primitives, hereafter referred to as Primitives, will be a standard set of modeling elements, and associated symbols mapped to DM2 concepts and applied to modeling techniques. Using the Primitives to support the collection of architecture content and, in concert with the PES, will aid in generating common understanding and communication among architects in regard to architectural views. The full range of Primitives for views, as with the current BPMN Primitives, will be coordinated for adoption by architecture tool vendors. Architects typically collect and organize data through the use of architecture techniques designed to use views e. The architectural data should be stored in a recognized commercial or government architecture tool. Terms and definitions recorded are related to elements of the DM2. Designation of a data structure for the Architectural Description effort involves creation of a taxonomy to organize the collected data. This effort can be made considerably simpler by leveraging existing, registered artifacts to include data taxonomies and data sets. Each COI maintains its registered data either directly or through a federated approach. In addition, some organizations have developed templates, which provide the basis of a customizable solution to common problems, or requirements, which includes datasets already described and registered in the DMR. Conduct Analyses in Support of Architecture Objectives. Architectural data analysis determines the level of adherence to process owner requirements. This step may also identify additional process steps and data collection requirements needed to complete the Architectural Description and better facilitate its intended use. Validation applies the guiding principles, goals, and objectives to the process requirement, as defined by the process owner, along with the published performance measures metrics , to determine the achieved level of success in the Architectural Description effort. Completion of this step prepares the Architectural Description for approval by the process owner. Changes required from the validation process, result in iteration of the architecture process repeat steps 3 through 5 as necessary. The final step in the architecture development process involves creation of architectural views based on queries of the underlying data. Presenting the architectural data to varied audiences requires transforming the architectural data into meaningful presentations for decision-makers. This is facilitated by the data requirements determined in Step 3, and the data collection methods employed during Step 4. DoDAF-described Models are those models that enable an architect and development team whose data has already been defined and described consistent with the DM2. The models become views when they are populated with architectural data. Fit-for-Purpose Views are user-defined views that an architect and development team can create to provide information necessary for decision-making in a format customarily used in an agency. These views should be developed consistent with the DM2, but can be in formats e.

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

DoDAF does not require specific models or views, but suggests that local organizational presentation types that can utilize DoDAF-created data are preferred for management presentation. A number of available architecture tools support the creation of views described in this step. The PES provides the format for data sharing. Scoping Architectures to be "Fit-for-Purpose" Establishing the scope of an architecture is critical to ensuring that its purpose and use are consistent with specific project goals and objectives. Meeting intended objectives means those actions that either directly support customer needs or improve the overall process undergoing change. At each tier of the DoD, goals and objectives, along with corresponding issues that may exist should be addressed according to the established scope and purpose, e. Establishing the Scope for Architecture Development Establishing a scope for an architecture effort at any tier is similarly critical in determining the architecture boundaries Purpose and Use expected , along with establishing the data categories needed for analysis and management decision-making. Scope also defines the key players whose input, advice, and consensus is needed to successfully architect and implement change i. Importantly, scope also determines the goals and objectives of the effort, consistent with both boundaries and stakeholders; since goals and objectives define both the purpose for architecture creation and the level of the architecture. Establishing the scope of an effort also determines the level of complexity for data collection and information presentation. Architecture development also requires an understanding of external requirements that may influence architecture creation. An architecture developed for an internal agency purpose still needs to be mappable, and consistent with, higher level architectures, and mappable to the DoD EA. For some architecture developments, consideration must be given in data collection and graphical presentation to satisfaction of other external requirements, such as upward reporting and submission of architectural data and models for program review, funding approval, or budget review due to the sensitivity or dollar value of the proposed solution. This site contains guidance on data collection for specific views required by instruction, regulation, or other regulatory guidance i. Architecture scoping must facilitate alignment with, and support the decision-making process and ultimately mission outcomes and objectives as shown in the figure below. The figure below shows how the development of architectures supports the management decision process. Having a disciplined process for architecture development in support of analytics will produce quality results, not be prone to misinterpretations, and therefore, be of high value to decision makers and mission outcomes. A similar convergence addresses the kinds of techniques, pattern, and designs that are independent of specific application domains, and that enable effective production of responsive, scalable, flexible, and unifiable enterprise applications. Since DoDAF provides the conceptual, logical, and PES but does not otherwise prescribe the configuration of the product composition, architects and stakeholders are free to create their views of data that best serve their needs. Architectural Descriptions define a strategy for managing change, along with transitional processes needed to evolve the state of a business or mission to one that is more efficient, effective, current, and capable of providing those actions needed to fulfill its goals and objectives. Architectural Descriptions may illustrate an organization, or a part of it, as it presently exists; any changes desired whether operational or technology-driven ; and the strategies and projects employed to achieve the desired transformation. An Architectural Description also defines principles and goals and sets direction on issues, such as the promotion of interoperability, intra-, and interagency information sharing, and improved processes, that facilitate key DoD program decisions. Such support extends beyond details or summaries of operational and systems solutions, and includes program plans, programmatic status reporting, financial and budget relationships, and risk management. In addition to detailed views of individual solutions, the framework supports the communication of enterprise-wide views and goals that illustrate the context for those solutions, and the interdependencies among the components. Beyond the solution space, standard mechanisms for communicating program plans, financial information, and project status are established so that executives and managers can evaluate and direct their programs. The DoD EA is an Architectural Description that is an enterprise asset used to assess alignment with the missions of the DoD enterprise, to strengthen customer support, to support capability portfolio management PfM , and to ensure that operational goals and strategies

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

are met. The DoD EA is shown below. Its purposes are to guide investment portfolio strategies and decisions, define capability and interoperability requirements, provide access to Segment architecture information, to establish and enforce standards, guide security and information assurance requirements across the Department of Defense, and provide a sound basis for transition from the existing DoD environment to the future. Its content includes but is not limited to rules, standards, services and systems lifecycle information needed to optimize and maintain a process, or part of a process that a self-sufficient organization wants to create and maintain by managing its IT portfolio. The DoD EA provides a strategy that enables the organization to support its current operations while serving as the roadmap for transitioning to its target environment. Components of the DoD EA The JCA portfolios describe future, required operational, warfighting, business, and Defense intelligence capabilities, together with the systems and services required. They provide the organizing construct for aligning and federating DoD EA content to support the Department portfolio management structure. The description of the future DoD operating environment and associated capability requirements represent the target architecture of the DoD EA. These are time-phased as determined by functional owners and JCA developers. Transition Planning As discussed above, one major impetus for creating and using Architectural Descriptions is to guide acquisition and development of new enterprises, capabilities and systems or improvements to existing ones. It is expected that the To-Be Architectural Descriptions will change over time as Departmental priorities shift and realign. Federated Approach to DoD Architecture Management The Department has adopted a federated approach to distributed architectural data collection, organization, and management among the Services, Agencies and COIs as its means of developing the DoD Enterprise Architecture, with a virtual rather than physical data set described through supporting documentation and architectural views.

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

Chapter 3 : Enterprise architecture framework - Wikipedia

The Lean Enterprise Architecture Framework (LEAF) is a tool- and modelling supported part of the Lean Enterprise Architecture Development approach. The LEAD approach consists of operating model, framework, and Service-Driven Approach as a method. The LEAD Operating Model organizes enterprise's capabilities around a value delivery chain.

A practical guide to developing enterprise architecture Franki Schafrik Published on October 18, Enterprise architecture is a logical organization of a business and its supporting data, applications, and IT infrastructure, with clearly defined goals and objectives for the future success of the business. A typical architecture consists of diagrams, or models, that show how aspects of your business relate. For example an organizational chart is a model of how business units relate to each other. Businesses should have an "as-is" architecture that represents its current state, and a planned architecture to show the direction of the business over the next one to five years. Enterprise architecture aligns the following key areas. Note the examples in each area: Processes, strategies, organization charts, and functions Information: Conceptual, logical and physical data models to show what information is needed and how it relates to other information For example, a customer and an order Application: Portfolios, interfaces, and services Infrastructure: Network concept diagrams, technology reference models To achieve alignment, you model each key area from its own perspective, and then link the models from each perspective. For example, model business processes from a business perspective. Do not include things like applications. Then, link the business processes to the applications that support them, which helps you achieve alignment. We do this to ensure that every decision is based on a business need; therefore, an application is not dictating the way a business process is designed. Throughout this article, we assume that you have a modeling tool to create your architecture. The implementation specific information in this article is based on Rational System Architect. If you do not have a purpose, your project will fail The simplest way to ensure that your architecture fails is to not have a purpose for doing it. When projects are not successful, I ask why they are creating an enterprise architecture. They respond, "Because we want an architecture! Identify the purpose of your architecture You can define the purpose of your architecture by asking the following questions: What information is important for the architecture? How much detail is needed to support analysis and decision making? Who will produce or use the architecture? What is the expected ROI of the architecture? What are the maintenance considerations? If you cannot answer these questions, your architecture project will likely fail. Without a purpose you can waste months drawing business process diagrams that no one cares about. Or you may draw complex diagrams of application interfaces that cannot be presented to senior management because it will make their heads explode. As an example, for a hotel chain, hotel managers were identified as the audience for EA. By knowing the purpose of the architecture, you can scope the necessary models and data that are needed to ensure people use your architecture for analysis and business decisions. Do not go overboard on your first foray into architecture. Even if you have a very large and experienced team, you will not be able to capture all of the information about your organization. It is also important to remember that comprehensive architecture can obfuscate the important things. Identify your critical business questions, and use those as the focus of your first architecture project. Architecture provides a route for answering questions Step 2. Identify your business questions The first thing I do with a client is discuss the questions that are critical to their business; and then I help them identify the ones that are hard for them to answer. The following questions are ones that many clients need answered: What is the impact of retiring an application? What is the impact of moving a location? What applications are needed to support a business process? What is the impact of replacing servers? What processes need to be developed to support a new strategy? Where are the gaps or redundancies in our application portfolio? Your questions drive the content of your architecture. If most questions concern your application portfolio, then focus on defining the application area. If you need to understand how your processes support a new strategy, then focus on the business area. Then you can begin to expand the scope of your architecture with new business questions. Identify assumptions and business rules

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

Now that you have identified the audience, purpose, and questions, you should identify the business rules that constrain or explain the area of interest. Every business has rules. For example, if you are capturing information about critical business processes, you must also capture any regulations or corporate standards for the process. A corporate regulation would then be created to show that the company is meeting the requirements of HIPAA. You should capture assumptions about your architecture, such as "New application information will be uploaded on Friday" or "Every business unit is responsible for documenting business processes. Identify your framework The following industry standard frameworks can help you create an enterprise architecture: A framework also provides guidance on what information you need to capture based on the stakeholders who will use the architecture. It provides guidance on organizing information but does not suggest a specific implementation for your architecture. There is plenty of information on the Internet about each of these frameworks. The framework you choose depends on the goal of your architecture, the experience of your team, and whether you want to follow a defined process like ToGAF, or just need help identifying which model to use for what purpose as in Zachman. You can also combine frameworks. ToGAF and Zachman are often used together. Your choice should be based on your purpose; do not make a random selection How does a framework fit in to your architecture? A framework provides guidance on what to model. Methodologies are then used to create models. A methodology is a rule set that explains how to model something. A framework helps with methodology selection A framework helps organize the key areas of the architecture and identifies the views you need to model, such as the perspective and the data needed to answer business questions. The hotel chain decided to use the Zachman framework. Whenever possible, use an industry standard methodology rather than something "home grown". Industry standard methodologies have rule sets and standard ways of modeling. Most homegrown methodologies fail to capture information in a useful way because the rule set is not clearly defined, which allows people to model the same information in multiple ways. This also affects analysis because the information is not captured according to a standard. Multiple models are produced to support the framework based on the type of information you need. Framework with supported methodologies View image at full size Step 5. Create a metamodel A metamodel is an abstract view of your architecture. It shows the data you are trying to capture, and the relationships among the data. This is where you realize alignment, which is based on answers to your business questions. For example, if you need to know the application that supports a certain business process, there must be a relationship between those two things in your metamodel. Otherwise, there is no connection between the data, you cannot answer your business question, and the architecture is not functional. Note that you do not want a direct relationship between everything in your metamodel, and you should only link things together that have logical relationships. For example, linking an organizational department to a technology does not make sense, but linking a technology to an application does. A good modeling tool such as Rational System Architect supports traversing the metamodel to create complex reports. So, in this metamodel example you can report on the hardware that supports a business function even though there is not a direct relationship in the metamodel. In a metamodel you can potentially traverse from a business function, to a business process owned by that function, to a location of the business process, to a supporting application the process needs, and finally to the technologies that support that application. Example of relationships metamodel Your metamodel should include the following features: Relationships between the architecture elements. For example, a business process to an application. Definitions of the elements. For example, the meaning of the term "application" and what properties you will capture. Traceability to business questions. For example, if your question is "What applications support what business processes? Identify the models needed in the architecture Now that you have identified your business questions, your framework, and the metamodel you need to answer your questions, you need to figure out what models to draw. Using a business process as an example, there are many industry standards that support modeling business processes, such as BPMN and flow charts. Choose your modeling methodology based on the following criteria: The audience for the information. The elements of the metamodel. If in your metamodel you need to understand data as it relates to business processes,

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

consider using BPMN to model that. If instead you are just worried about the sequence of process steps consider creating a flow chart. After knowing the audience and the content you want to model you can then identify the diagrams you need to create. In the above example, since you needed information about business processes and system interfaces, you could select the following models: BPMN captures business processes System architecture captures applications Using the hotel example, they needed to answer the business question "What applications support what business processes? Further, separation of the architectural views, such as the application view from business view, is a best practice.

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

Chapter 4 : How to Create an Enterprise Architecture Framework Diagram - Dragon1

Oracle White Paper in Enterprise Architecture – “The Oracle Enterprise Architecture Framework 3 Enterprise Architecture Frameworks Creating an Enterprise Architecture from scratch can be a daunting task, so EA frameworks were created to simplify the process and guide an architect through all areas of architecture development.

The idea behind this LEAF is to manage end to end value delivery chains of any kind of development targets, such as services. The LEAF consists of three layers as follows: See the reference implementation created with Sparx EA via this link. Management Part The topmost Management layer of the Framework consists of management elements, from which all the other views and elements are to be derived from. It is important to visualize all the top-level aspects that widely influence to overall development and operations of the organization. Other management level elements that guide the overall development are e. Management can also identify required capabilities that are required to support strategies. The value delivery chain based approach is the fundamentally crucial idea behind this part of the LEAF. The outcome of the process can be e. Capabilities are composed from organization resources: This Service Delivery Chain provides overall visibility to life-cycle of each and every development target such as service, digital transformation plan or any domain specific development case. The service delivery chain can be used to manage the overall development, whether a service is in its early ideation or design phase, in development phase, or whether the service is released into production. Accordingly, the service delivery chain is divided into phases that are related to service life-cycle. These value streams are as follows: In addition, portfolios can be related to those phases accordingly: The value delivery chain can be defined according to process that is the most appropriate and fits for purpose. Architecture Landscape Part Each development target, -domain or -service on the middle layer is associated with relevant architectural views or diagrams, which are composed from architectural building blocks. This part of the framework can be organized into layers, each of which consists of architectural elements such as business services and -processes, applications etc. Alternatively, this Architecture Landscape can be organized e. The Architecture Landscape provides a view to both current state and future state of the organization and its parts. The elements of the Architecture Landscape are modeled together with those stakeholders that are involved in design, development or operation of the services. Elements and diagrams are created when needed: The framework can be created by any modeling tool that supports standard modeling notation, ArchiMate , and some other optional standard notations such as BPMN and UML. This framework can be implemented with any of the available modelling tools, that can be used for Enterprise Architecture Management EAM purposes. Some of the modeling tools can be integrated with other tools such as mentioned above. Architecture part is based on ArchiMate Framework, which consists of the most relevant concepts of an enterprise. Every placeholder on the landscape figure above can contain more specialized views diagram types “ according to what is appropriate for the purpose. Business Processes can contain e. Process Maps and Process Co-operation Views. It is easier to most of the stakeholders from diverse backgrounds to understand the big picture and overall context when it is visualized with layers. The middle layer of the Framework can be modified according to the operating model and value stream approach that are used in the organization. The middle layer Value Delivery Chain contains the portfolios e. Roadmaps can be located in this part. The bottom layer Architecture Landscape contains the actual EA content. For example, the Architecture Landscape can be organized based on following aspects: Business Domains, each of which contains the actual EA content on the 2nd level when opened on the 1st level. The overall view of the enterprise is on the top of the underlying domains figure 1 , Architectural layers according to ArchiMate: Motivation aspect on the left side and layered views on the right side figure 3 , Conventional EA domains: Business-, Data-, Application and Technology Architectures figure 4. LEAF “ variant 2: Architectural Layers according to ArchiMate framework Figure 3: LEAF “ variant 3: In this usage scenario, the life-cycle of a service can be managed with the help of the framework. For example, in the service design phase, the

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

framework can be used for designing service blueprints or customer journeys. All the new and existing application services etc. In the service development phase, all the related elements can be linked to the service in development: When the Service is realized and deployed to production, all the operational aspects can be added into those same diagrams or completely new diagrams, all of which can help operational tasks of the services on production. Those methods can be used as step-by-step development approaches " for any development targets of any size e. Examples of practical methods are as follows: SDA is all about services: This is how the LEAF concretizes the service life cycle management. As such, LEAF as a framework " implemented on modeling tool " provides a pragmatic, concrete tool for service development in practice. The SDA combines terminologies, principles, tools and methods from two complementary disciplines: Business, Application and Technology. In addition, ArchiMate Strategy elements can be used for modelling strategy aspects. As typical to Sparx EA models, visual elements can be clicked to drill-down into the child diagram. More example content is to be added continuously into this model. Some tools, such as Sparx EA, provides powerful scripting- and data import capabilities. With scripting, it is possible e. For more advanced purposes, even the whole model can be created based on the existing data. There is couple of alternative ways to do that. The main advantages of using a graph database for end user visualization, are as follows: For documentation purposes, an architecture model maintained in a repository, can be used for dynamic document creation. Most of the modelling tools provide document generation capabilities, with which a set or subset of the model repository data can be used for generating a document e. Word or pdf format. For example, a document can be generated based on diagrams and description texts that are maintained in the modelling tool. No extra effort is needed for documentation. This is aligned e. Conclusions The Lean Enterprise Architecture Framework LEAF can be used with an appropriate development method in an organization to support design, development and operation of services. The terminology is simplified and only the most relevant elements are introduced for the sake of simplicity. The LEAF can be applied and modified to fit for the purpose and what is appropriate. The main advantage of this kind of framework is to provide a easier approach to support Lean and agile overall, holistic enterprise development ion an organization, which is not architecture-driven but development-driven: You may also like.

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

Chapter 5 : An Enterprise Architecture Development Framework, 4th edition by Adrian Grigoriu

An enterprise architecture framework (EA framework) defines how to create and use an enterprise architecture. An architecture framework provides principles and practices for creating and using the architecture description of a system.

September 30, 1. Introduction In this digital age, Enterprise Architecture EA planning is more important than ever before. More practical and easier approach with simpler framework is needed. The LEAD consists of a value chain based operating model and b framework. The operating model organizes enterprise and its capabilities around value delivery chain. The Framework, with revised architecture practice, support the organization to fulfill the design, development and operational tasks. The LEAD can be adapted to target area of any size: In LEAD, all the relevant organizational capabilities related from management to operations are organized around the value delivery chain based operating model. Those capabilities cover management, design, development and operations. This means a paradigm shift from conventional EA practice to new, modernized, lean and agile EA practice. Also, the LEAD can be adapted to target area of any size: Questions to which the LEAD concept tries to answer: How to set up an integrated, Lean and agile operating model that: How to renew a conventional Enterprise Architecture practice to: Operating Model LEAD integrates and mobilizes the organization and its capabilities into operating model, that is capable of dealing with customer demands and producing services efficiently. All the organizational functions and roles are adjusted seamlessly into the same production line, to the value delivery chain based integrated LEAD production machine. This encourages enterprise to get organized around value delivery chain. Service is the primary unit of value creation, development and operation. LEAD value delivery is focusing on services instead of projects. As such, LEAD combines customer centric service design thinking with enterprise architecture. The value delivery chain can be expressed in the form of Service Delivery Chain, as illustrated in the figure 1 below. The layers are as follows: Contains views that are related to governance, strategic planning and business planning of the organization. Contains the Idea to Production value stream, which consists of portfolio views Conceptual, Logical and Realized Services. This layer visualizes the overall view of development targets services from ideation to production. That is a development method, in which the service -concept is crucial. The phases design, development and operations can be mapped to as plan, build and run phases. Contains views of elements related to overall development. These elements are foundational enterprise architecture content building blocks, that are re-used in composed views on the middle layer. This layer consists of business domains, each of which is divided into three layers according to ArchiMate core framework as follows: Business, Application and Technology. The idea behind this framework is to provide simple and easy navigation view into diagrams for different stakeholder groups. It is not necessarily crystal clear what are the required or existent capabilities and how they interact “ nor how the information flows between them. It is not explicitly defined which capabilities, skills and resources are providing what value exactly, what is their reason for being. Should they be internal to organization, should they be externalized, or should they exist at all? There are many unclear aspects and functions inside the organization, as there is no explicitly defined value chain to which all the capabilities can be assigned to. Working conditions may suffer the conditions caused by poor managership, by the command and control management system. As result of this, the organization is not optimally adjusted for value creation nor customer centricity. Often these matters mentioned above are symptoms of problems in management. EA Crisis Enterprise decision making and -development should be supported by enterprise architecture, but only occasionally that is the case. Lots of frameworks since Zachman framework have been seen the dawn, but none of them never succeeded with full extent. Conventional enterprise architecture frameworks and methods are considered as too difficult, too academic and cumbersome to understand “ for others than architects. Some may argue that EA practice has been proven useless as a function, as it is not providing any direct business benefits. EA is too difficult, it is not understood, it is not interesting. The EA practice has to be changed. Finnish public sector organizations

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

have been practicing enterprise architecture according to Public Administration Recommendation latest version v 2. Although the recommendation is comprehensive, it is lacking of easy adoption guidance and the content framework is not very practical. EA domain based content framework is aligned with Togaf domains business, data, application and technology , but this approach is not the most optimal in the age of agile and high-speed development. Regardless of the disreputation of the EA, it has intrinsic value. EA supports management decision making, strategy execution and business development, as well as operational activities. Enterprise architecture management EAM is relatively important capability for every organization. Architecture competencies and architects of some kind are useful not especially enterprise architects, but architects of some professionalism such as solution architects anyhow. There should be architects of some level capable of exercising enterprise architecture management in the organization. Architect roles should be carefully considered and aligned with the needs of organization: It goes without saying that old-school enterprise architects have to orientate themselves to more agile, practical and collaborative ways of working. The most tedious part of the EA is the framework and method combination. That should be the most beneficial and valuable part of the puzzle, but that seems to be the hardest part. In spite of their meaningful purpose, EA frameworks and methods have been found somewhat awkward and boring from the business people point of view. Architects disagree, but accident has already happened. Thus, some more practical frameworks and methods could still renew, not only the EA, but also the whole enterprise development to be more efficient in its full extent. The Good, the Bad and the Ugly: Enterprise Architecture is valuable and EAM capability is necessary. Enterprise Architects are the ones to blame of bad reputation of EA function , they have to disrupt the EA function or die. EA Frameworks and methods are too difficult, complex, cumbersome and clumsy. According to practices of Agile methods, too much planning in early phases of development is unnecessary waste. Development should be left open to changing conditions. It cannot be seen precisely beforehand, what is the nature of the development target, what are the requirements exactly and how much planning can be done. As typical to agile methods, only high level epics or user stories are enough to get started experimenting and learning by doing. What is the right size of architecture work? Are we doing development with or without architecture? The overall complexity increases in accelerating speed. New cloud services or APIs need to be managed in controlled way. Agile efforts are delivering new services fast. Without any overall planning the enterprise services landscape is shifting to chaos. It has been recognized that the truth lies somewhere in between both extremes: Some long-term architecture planning, continuous architecture repository refactoring and updating keeps the overall architecture landscape coherent. But architecture team cannot anymore create architecture artifacts lists, maps, diagrams etc. However, this is not black or white, but the main idea gets clear: Pragmatic Approach Needed Efficient enterprise is balanced production machine, in which all the capabilities are integrated together enabling seamless information flow. Customer demands are flowing through the value chain-based production line in controlled manner. Architecture is co-operating with other functions by providing overall architecture landscape of existing and planned services of the enterprise. Challenges can be solved with better co-operation and interaction. This can be enabled with the LEAD concept, which is a practical approach for enterprise operational development. The key principles are collaboration and visibility. LEAD concept is concrete development approach, which: Enables customer centric development and more efficient value creation Encourages to establish cohesive operating model with clearly defined capabilities Increases collaboration between multidisciplinary roles in teams, Motivates to cultural changes in the organization, to modern, open, innovative, co-operative approaches based on trust, openness, transparency etc. The operating model consists of capabilities and their interactions. The LEAD operating model defines how the information flows between the identified capabilities “ and to be more precise “ what information flows between which capabilities. Some of the capabilities are related to management layer, some capabilities are assigned into the Service Delivery Chain, and some of them are supporting capabilities. Operating model is to be defined: This capability handles all the incoming demands from different channels. Some business-driven demands can raise when customer relationship management is

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

in contact with the customers. Demand Management is illustrated in the Idea to Production value stream diagram below. In practice, LEAD operates in collaborative way that requires strong personnel involvement and commitment, on both management and operational levels. All the ideas and other incoming demands are managed in the single backlog and single Kanban board physical and electronic. Idea to Production Value Stream. Customer Perspective And Holistic Development Approach One of the most important observations is that when taking the customer perspective, the organization is forced to consider the enterprise as whole, to take a holistic, systemic development approach. This means that the behavior and structure of the enterprise have to be designed and optimized as a one system “ based on the customer viewpoint. No partial optimization, but overall approach instead.

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

Chapter 6 : Gartner's Enterprise Architecture Process and Framework Help Meet 21st Century Challenges

defining architectural frameworks to guide in the development of large, complex systems development. A Comparison of Enterprise Architecture Frameworks.

Overview[edit] Enterprise architecture regards the enterprise as a large and complex system or system of systems. The components of an architecture framework provide structured guidance that is divided into three main areas: Each view describes one slice of the architecture; it includes those entities and relationships that address particular concerns of interest to particular stakeholders; it may take the form of a list, a table, a diagram, or a higher level of composite of such. Methods for designing architecture: Usually, an overarching enterprise architecture process, composed of phases, broken into lower-level processes composed of finer grained activities. A process is defined by its objectives, inputs, phases steps or activities and outputs. It may be supported by approaches, techniques, tools, principles, rules, and practices. Overview of Enterprise Architecture Frameworks evolution “ Evans and Lou R. Many of the aims, principles, concepts and methods now employed in EA frameworks were established in the s, and can be found in IS and IT architecture frameworks published in that decade and the next. Then and in later papers, Zachman used the word enterprise as a synonym for business. The paper did not mention enterprise architecture. It was promoted within the U. It was not an EA framework as we see it now, but it helped to establish the notion of dividing EA into architecture domains or layers. The business mission is the primary driver. Then the data required to satisfy the mission. Then the applications built to store and provide that data. Finally the technology to implement the applications. Enterprise Architecture Planning is a data-centric approach to architecture planning. An aim is to improve data quality, access to data, adaptability to changing requirements, data interoperability and sharing, and cost containment. TOGAF started out taking a strategic and enterprise-wide, but technology-oriented, view. It emerged from the desire to rationalize a messy IT estate. Right up to version 7, TOGAF was still focused on defining and using a Technical Reference Model or foundation architecture to define the platform services required from the technologies that an entire enterprise uses to support business applications. It introduced structured analysis, after information technology engineering , which features, for example, mappings of organization units to business functions and data entities to business functions. Today, business functions are often called business capabilities. A research project for the development of professional certificates in enterprise and solution architecture by the British Computer Society BCS showed that enterprise architecture has always been inseparable from information system architecture, which is natural, since business people need information to make decisions and carry out business processes. Enterprise Architecture strives to align business information systems technology with given business strategy, goals and drivers. EA framework topics[edit] Architecture domain[edit] Layers of the enterprise architecture.

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

Chapter 7 : A practical guide to developing enterprise architecture

TOGAF is an enterprise architecture framework that helps define business goals and align them with architecture objectives around enterprise software development.

Two Dragon1 web applications were used to create this interactive and dynamic visualization of the EA Framework: An Enterprise Architecture Framework in its smallest form is nothing more than a classification scheme of types of architectures and their most important parts in a single schematic overview. An Enterprise Architecture Framework often can be visualized in a reasonably small but still comprehensive diagram. The AS-IS is of impact today, even if you did not create the enterprise architecture framework yet. Therefore it is always useful to visualize the AS-IS framework and reuse the current old strategy aspects to create the diagram. It is just sitting there, waiting for you to dust it off! Enterprise Architecture Frameworks show the types of architectures used at a certain moment in time and it shows the concepts, principles, standards norms , building blocks, metamodels and user models that are part of the underlying architectures. You, as an enterprise architect, can decide yourself which parts of the architectures you think should be part of the framework. A diagram is only an architecture diagram when it contains information about concepts that together create an architecture or are part of an architecture. An architecture diagram is a conceptual structure diagram. If a diagram omits the conceptual layer and only show elements of the logical layer it is not considered anymore an architecture diagram but a logical structure diagram. If a diagram only shows the components and objects of the physical layer the diagram is considered to be a physical structure diagram. Often the conceptual, logical and physical layers in diagrams are mixed up. Dragon1 advises always to focus on a layer in a diagram or at least to complete one layer in one diagram the diagram shows the complete logical layer, etc.. Make a list of candidate architectures and describe the enterprise architecture function: Look at the structures, systems, domains and functions which are present in your organization. Are there already architectures defined for them? Check if there is an architecture for a structure, system or domain, check if there are concepts and principles applied and if there are policies, rules, standards applied for that structure, system or domain. Try to find a reference model for the framework that is common for your industry. Review your list with others to verify the actual status of the architectures. Try to get a common definition for the architecture to define its content, context, scope , who is the owner, who is the manager and who is the user? List 7 issues for the architecture. List 7 products of each architecture to create or have. With these products, the owner and enterprise architect of an architecture can direct developments and realizations. Create the enterprise architecture framework model. Create a management overview of the Enterprise Architecture Framework model. Be sure you are not visualizing the perfect world figure 1 , but the current actual situation figure 2. It could well be that your organization holds two different business architectures and technical architectures. That all depends of course of your definition of architecture. Visualize in a diagram the management overview of the Enterprise Architecture Framework model. Write down in a document why, when and how you created the diagram and how it should be maintained. Make the diagram part of the maintenance update cycle for architecture products. That example you could create without an assignment and you can prepare a case or situation to depict so you can really show the benefits of having such an EAF diagram available when does it come in handy. What are questions people now have, what are issues and problems regarding consisting architectures and how does this diagram help to solve them? An AS-IS Enterprise Architecture Framework is one of the first products to create as part of your Enterprise Architecture Baseline Usage of this Enterprise Architecture Framework Click to enlarge Architecture Framework Management Report View With an Enterprise Architecture Framework you can do a lot of reporting, such as using colors how well the architectures are designed and implemented yet, or how well they comply to certain standards. If created correctly it can act perfectly as auditing framework. The color orange you can use for the parts which are partially compliant and the color green that it is fully compliant. And black means you could

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

not get the status and white means getting it status has not been done or planned.

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

Chapter 8 : TOGAF or not TOGAF: Extending Enterprise Architecture beyond RUP

The Open Group Architecture Framework (TOGAF) is a framework for enterprise architecture that provides an approach for designing, planning, implementing, and governing an enterprise information technology architecture.

These work products and the activities that produce them are well understood within both the technical and business communities. However, the ways in which we conceptualize, prioritize, and select which business problems and user needs to implement in software remains a highly variable process throughout our industry. I begin by contrasting the discipline of enterprise architecture with the solution architecture and business architecture disciplines, while relating them to RUP. Contrasting different architectural frameworks In terms of their scope, there is a certain amount of overlap between enterprise, solution, and business architectural frameworks as they are commonly understood. So, how do they relate? Solution architecture Solution architecture frameworks take a variety of forms. IT professionals are already accustomed to dealing with Application, Data, Technology and other solution architecture forms often called domains during information systems development and maintenance projects. New and overall more specialized solution architecture forms, such as Security and Testing, are quickly becoming mainstream also. The most widely recognized solution architecture domains, their major subjects, and the dependencies among them are shown in the Figure 1. View image at full size Figure 1: Domains and subjects of the solution architecture discipline All the solution architecture domains shown in Figure 1 are considered "technical," as their scope includes various elements of technology, such as software, data, and IT infrastructure. These domains are typically dealt with by technologists -- i. Business architecture Business architecture emerged as a separate domain in the s, when many organizations embraced the business architect role as they attempted to optimize their business processes. The business architecture discipline is concerned with the "terms of reference" of the business and with describing how it operates. Although not everyone agrees about what components should be included within business architecture frameworks, there is a general consensus that Process and Information, Organization, and Performance aspects are relevant. Each of these components is, in itself, fairly significant and incorporates multiple subject areas, as shown in Figure 2. View image at full size Figure 2: Components and subjects of the business architecture domain The Process and Information component is, arguably, the focal point for business architecture activities because among other things it defines, describes, and classifies business processes and supporting structures that comprise the organizational business model. This component also comprises a host of related subjects, such as usability and accessibility. Every organization will, of course, have different business processes, structures, workflows, etc. Likewise, one organization might have that "extra something" in its business process model that gives it an edge over the competition while another organization struggles in this area. The Organizational component is concerned with the structure and design of work practices, and with the operational style of an organization. The subjects dealt with by this component include the organizational structure, the products and services the business produces, its business units and their locations, and so forth. Business Performance is a management-oriented component that encompasses subjects that define and quantify organizational efficiency and effectiveness. Concerns about productivity, business risks, and other related subjects belong here. The enterprise architecture domain Before I discuss enterprise architecture, I want to sketch the problem this discipline attempts to address. The majority of existing implementation methodologies include techniques for creating the solution to a well-defined business need. These methodologies, however, are not directly concerned with how and why a business need has emerged. Instead, they focus on the relative importance of responding to a given need versus whatever other needs the organization might have. The RUP delivery process takes the decision to begin an implementation largely for granted, and no RUP artifact is directly concerned with evaluating the relative urgency of the business need being addressed by the solution implementation. In contrast with RUP and other disciplines that focus primarily on implementation, the principle concern of the enterprise architecture domain is the enterprise

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

-wide identification, specification, and prioritization of business needs. The viewpoints discussed and the models drawn in the context of an enterprise architecture framework address a range of problems both current and potential. An EA roadmap is likely to include more than a single proposed solution as illustrated in Figure 3 , which may result in multiple, simultaneous implementations. A sample enterprise architecture roadmap

Envisioning the enterprise of tomorrow The Institute for Enterprise Architecture Development IFEAD sums up the key guiding principle of the enterprise architecture discipline this way: An important aspect of this assertion is that enterprise architecture is a holistic discipline that unites business and technology elements based on a single, strategic enterprise vision see Figure 4. Elements of the enterprise architecture domain

Now is the time Although enterprise architecture concepts have been around for more than two decades, the EA discipline began gaining momentum more recently. This can be attributed to the acceleration of environmental changes in organizations of all sizes across most industries. Business agility and, in particular, the ability of the technology infrastructure to respond to change in a timely manner have reached critical importance. Another factor contributing to a growing appreciation for the enterprise architecture discipline has been the more stringent regulatory climate of recent years, both in the US and elsewhere, which is driving organizations not only to improve their accountability and reporting practices, but also to make compliance organic to every business process. In response to a sharper focus on EA principles, several robust enterprise architecture frameworks have recently emerged, such as TOGAF. Similar subjects, different perspectives

The enterprise architecture discipline touches on virtually all the same subjects as the solution architecture discipline, but from a different perspective and in a distinctly different context. The EA context is holistic and its perspective is organizational, while the solution architecture is implementation-specific. By the same token, the enterprise architecture discipline is more than a superset of solution architecture domains. While the subjects of solutions architecture see Figure 1 are meant to be applied to the solution implementation, the EA subjects shown in Figure 5 are largely used for enterprise analysis, planning, and architecture governance. For example, to a solution architect, the "Applications" subject area can refer to a group of linked components and classes, whereas for an enterprise architect it can entail a node running a group of business processes or providing a range of services. Note that some typically low-level subjects from the solution architecture discipline are not included in the scope of EA, while several additional mostly higher-level subjects are added. Also note that the key business architecture subjects are subsumed intact within the EA discipline. Components and subjects of the enterprise architecture discipline

Scope of the enterprise architecture Enterprise architecture activities start long before projects driven by solution architecture commence, and are ongoing continuously throughout the lifespan of the enterprise. As a continuously running process, the enterprise architecture lifecycle has a single entry point that coincides with the creation of a practice, along with multiple process inputs that are distributed along the enterprise architecture timeline. In these frameworks a cycle consists of multiple phases with multiple feeds into EA activities, especially in the initial phases of a cycle. In some cases an idea is a by-product of an elevator talk between a CTO and an architect. In others, it is a coordinated response to requests escalated by the business planners, end users, or other stakeholders. In the case of the most mature organizations, it is an output of the strategic architecture analysis and planning process that also includes input from sources like those just mentioned. View image at full size Figure 6: Inputs to the enterprise architecture process

Enterprise architecture activities last as long as the enterprise is operational and has a vision. Yet, there is always a need for an EA perspective in the organization and EA activities must never cease. The emerging role of enterprise architect An enterprise architect is a thought leader, visionary, and industry expert. In most companies this is a new role that combines the skills of project manager, solution architect, and business analyst with the intuition of an executive. A common limitation in perspective among many IT architects is that they are grown-up programmers and tend to be very inward-looking. While this is not altogether a hindrance for architecting and designing "boxed" solutions, it is a less desirable trait in the context of "architecting" the enterprise. An enterprise architect is likely to be more extroverted and well able to use professional, working, and even personal relationships with business owners, business leaders, colleagues, and

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

customers to interpret, architecturally describe, and help to execute the enterprise vision see Figure 7. View image at full size Figure 7: The enterprise architect role The function of an enterprise architect is often compared to that of a city planner. In contrast, the function of a building architect is more readily associated with the IT architect role. The enterprise architect role often emphasizes the inductive skills of a detective over the deductive skills of a builder. However, the high-level perspective of the enterprise architect does not mean that this role is disengaged from the user community. On the contrary, an enterprise architect must be involved in helping customers understand their real needs as opposed to wants and to work with them throughout the implementation of a solution. At the same time, an enterprise architect should be able to view his or her domain at a level of abstraction that forestalls direct involvement in the practical aspects of implementations. As David Jackson of IBM put it, an enterprise architect should "be able to understand the business problem and the business domain and explain it to the technical people and to be able to understand the technology domains and explain the technical possibilities to business people. Architecture governance is the glue that provides both a context and a framework for all enterprise and project architecture activities. This is demonstrated in Figure 8, which highlights the serial nature of solutions implementation projects. As shown, a typical organization has a single instance of continuously running enterprise architecture process along with an arbitrary number of consecutive solution implementations. View image at full size Figure 8: RUP versus enterprise architecture lifecycles It is worth mentioning that several attempts have been made to extend the scope of RUP toward the enterprise as a whole. EUP introduces as many as seven new disciplines, including the Enterprise Architecture discipline, and more than twenty-five new roles, and also provides guidance for their tailoring. Meanwhile, RUP itself has evolved considerably since its inception and currently includes several enterprise processes, including Business Modeling and Change Management, in its core disciplines. The reality is that many organizations are simply not capable of implementing and maintaining an enterprise architecture program all at once, and would be better served to focus on less far-reaching process improvement techniques that emphasize efficiencies rather than effectiveness. The first approach projects organizational artifacts and processes onto the framework meta-structure. This approach works well for organizations that are very proficient at doing modeling. Organizations that favor this approach usually choose Zachman or an equivalent framework. One danger of this approach is that the framework structure may constrain creativity and add bureaucracy to the EA implementation process. The other problem with this type of framework stems from the serious shortage of implementation guidance. The second approach is based of a belief that an enterprise architecture program has to be process-driven. Since this approach primarily focuses on activities rather than artifacts, it may be easier to understand and link with the existing enterprise and solution methodologies and techniques. While both approaches have their pros and cons, a compromise could be to use an activity-driven process overall, while applying a meta-framework as a supporting structure or for analysis purposes. A sample enterprise architecture implementation checklist Here are some of the basic activities that must take place as part of any EA implementation. This list should give you an idea of what the enterprise architecture work is really all about: Study existing business practices. Understanding the business model of your organization and, at a minimum, its high-level business processes is a pre-condition for commencing the enterprise architecture implementation. Engage with senior management to understand strategic intent. Obviously, senior management holds a key to interpreting strategic vision. Understanding a vision is critical for the purposes of drawing the roadmap of the enterprise architecture, as this artifact will drive the "to-be" part of the architectural work to come more about this below. Connect with the business community to uncover urgent needs. Your goal is to extract those facts and level them with the expectations set by the strategic vision. Build a panoramic understanding of the existing technology environment. Draw the improvement roadmap. Having collected data from various sources, create a roadmap to advise the stakeholders -- including senior management, business, and technology leaders -- about how you intend to act upon the needs they have communicated to you. Keep enterprise architecture models up-to-date. Needless to say, after you create the roadmap for the enterprise architecture and receive stakeholder approval on it, you

DOWNLOAD PDF AN ENTERPRISE ARCHITECTURE DEVELOPMENT FRAMEWORK

ought to make the best effort to update it over time.