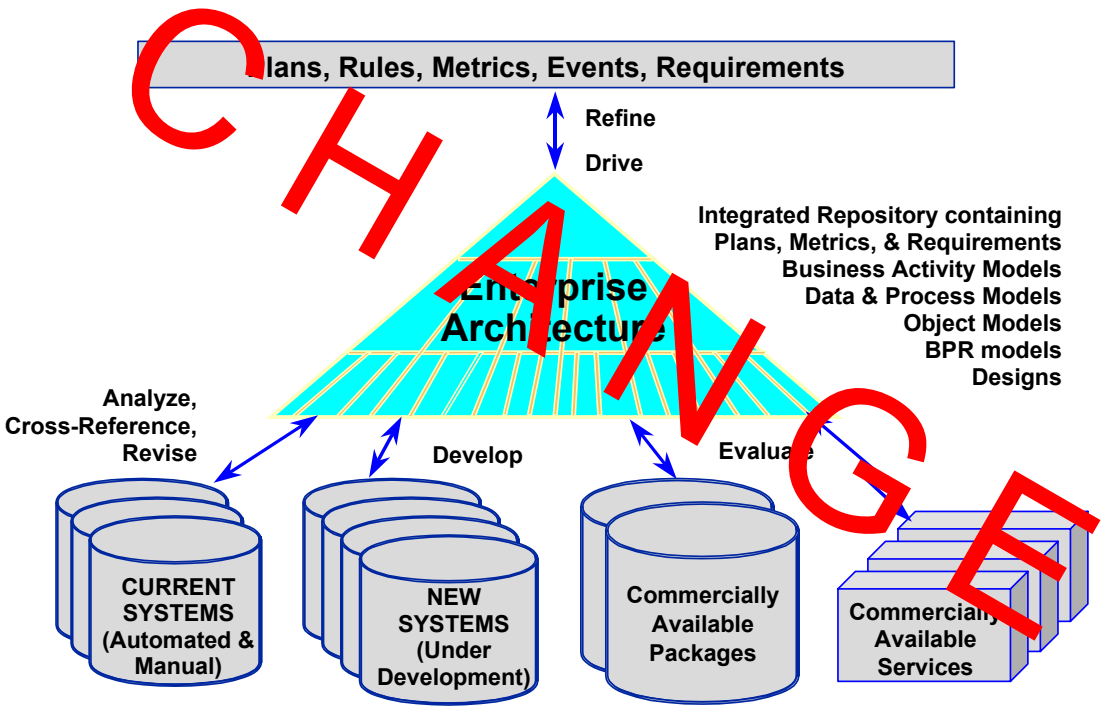
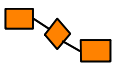
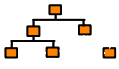
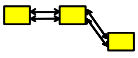
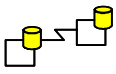
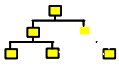
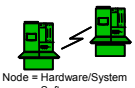
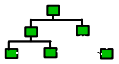
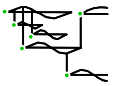
Creating a "High Performance" Enterprise in a Changing Environment



Introduction (Who & What)

John Zachman wrote, “To keep the business from *dis*integrating, the concept of an [enterprise] architecture is becoming less of an option and more of a necessity.” From that assertion over a decade ago, the Zachman Framework for Enterprise Architecture has evolved and become the standard around which many of the most successful major enterprises view and communicate their enterprise architecture.

The Zachman Framework draws upon the discipline of classical architecture to establish a common vocabulary and set perspectives -- a framework -- for defining and describing today’s complex enterprise systems. The Zachman Framework provides a comprehensive structure for classifying and managing Enterprise Architecture artifacts. The dimensions of the framework (see illustration below) include multiple perspectives of all the basic interrogatories.

abstractions / perspectives	DATA *What*	FUNCTION *How*	NETWORK *Where*	PEOPLE *Who*	TIME *When*	MOTIVATION *Why*
SCOPE *Planner* contextual	List of Things - *Important to the Business*  Entity = Class of Business Thing	List of Processes - *the Business Performs*  Function = Class of Business Process	List of Locations - *in which the Business Operates*  Node = Major Business Location	List of Organizations - *Important to the Business*  People = Class of People and Major Organizations	List of Events - *Significant to the Business*  Time = Major Business Event	List of Business Goals and Strategies  Ends/Means=Major Business Goal/Critical Success Factor
ENTERPRISE MODEL *Owner* conceptual	e.g., Semantic Model  Entity = Business Entity Rel. = Business Relationship	e.g., Business Process Model  Process = Business Process I/O = Business Resources	e.g., Logistics Network  Node = Business Location Link = Business Linkage	e.g., Work Flow Model  People = Organization Unit Work = Work Product	e.g., Master Schedule  Time = Business Event Cycle = Business Cycle	e.g., Business Plan  End = Business Objective Means = Business Strategy
SYSTEM MODEL *Designer* logical	e.g., Logical Data Model  Entity = Data Entity Rel. = Data Relationship	e.g., Application Architecture  Process.= Application Function I/O = User Views	e.g., Distributed System Architecture  Node = IS Function Link = Line Characteristics	e.g., Human Interface Architecture  People = Role Work = Deliverable	e.g., Processing Structure  Time = System Event Cycle = Processing Cycle	e.g., Business Rule Model  End = Structural Assertion Means = Action Assertion
TECHNOLOGY CONSTRAINED MODEL *Builder* physical	e.g., Physical Data Model  Entity = Tables/Segments/etc. Rel. = Key/Pointer/etc.	e.g., System Design  Process= Computer Function I/O =Data Elements/Sets	e.g., Technical Architecture  Node = Hardware/System Software Link = Line Specifications	e.g., Presentation Architecture  People = User Work = Screen/Device Format	e.g., Control Structure  Time = Execute Cycle = Component Cycle	e.g., Rule Design  End = Condition Means = Action
DETAILED REPRESEN-TATIONS *Subcontractor* out-of-context	e.g. Data Definition  Entity = Field Rel. = Address	e.g. Program  Process= Language Statement I/O = Control Block	e.g. Network Architecture  Node = Addresses Link = Protocols	e.g. Security Architecture  People = Identity Work = Job	e.g. Timing Definition  Time = Interrupt Cycle = Machine Cycle	e.g. Rule Specification  End = Sub-condition Means = Step
FUNCTIONING ENTERPRISE	**DATA** Implementation	**FUNCTION** Implementation	**NETWORK** Implementation	**ORGANIZATION** Implementation	**SCHEDULE** Implementation	**STRATEGY** Implementation

John A. Zachman, Zachman International

John’s belief in the need for a business-driven Enterprise Architecture which would provide an organization with an enterprise-scale blueprint -- or architecture --for their information infrastructure. Or, to put it in John’s words, ”If you are going to build a log cabin, go cut logs! However, if you want to build a skyscraper, you had better have an architecture!”

The author adds that if you want to build several skyscrapers, your architecture should include a plan for the city.

John says, “Without a comprehensive architecture you may find yourself paving over the cow paths of yesteryear or enabling the enterprise to make some of the same mistakes of the past...only faster!”

In today’s world, change is the only constant, and the ability to manage that change, is the ***only*** competitive advantage. Success and even survival is dependent upon knowing your enterprise, your stakeholders, your world, and how change affects them all. Enterprise Architecture, properly documented and deployed, that is shared throughout the enterprise and reflects the “business of the business,” becomes a guide for providing the information necessary to manage a successful business and to manage change.

EA Uses and Benefits (Why)

Enterprise Architecture serves as an "enterprise blueprint." It is a repository for designs and specifications of technologies, physical data structures and applications, as well as business plans, data models, and process models. Furthermore, it serves as a map of all the linkages among business initiatives, data required to support those initiatives, business processes that use the data, and physical information systems that support data requirements and processes.

It is these links that make the architecture a powerful vehicle for enterprise, information resource and technology management. For example, by providing a complete picture of data and processes – across information systems and functions – the enterprise architecture enables an enterprise to identify and control redundancy in data and functionality. By incorporating links between business initiatives and information structures and systems, it provides a business goal-driven framework for reengineering and integrating existing information systems. The comprehensive logical description of data enables the enterprise to maximize reusability and portability of data structures and processes. Enterprise Architecture also enables impact analyses both prior to and during implementation of business or technology changes, to examine potential and actual effects of new business requirements (e.g., Legislative initiatives) on information resources, and impact of proposed or actual system changes on business plans and requirements.

Such an architecture is critical for developing and implementing Data Warehouses, Data Marts, Enterprise Portals, Decision Support Systems, and Executive Information Systems that provide timely, trusted strategic information to enterprise decision makers. It is also critical for successful e-business and e-commerce.

EA Engineering (How, When & Where)

The author recommends an engineering approach to developing enterprise architecture (see the author’s paper “Enterprise Architecture Engineering). This means performing rigorous and repeatable activities and tasks with well-defined, consistent deliverables and milestones. It also means continually coordinating with affected or interested business experts, system users, as well as enterprise management.

The author’s EA Engineering life cycle involves a multi-phased, iterative approach that coordinates strategic, operational, and organizational demands. The focus of EA Engineering is the Motivation column of the Zachman Framework. This ensures that the enterprise architecture fully supports enterprise goals and strategies – all artifacts in the

other columns are linked to and driven by their performance elements. EA Engineering results in an architecture that is both actionable and adaptable.

EA Engineering involves a combination of forward and reverse engineering to establish the enterprise architecture. Forward engineering activities include business planning, performance planning and data and process modeling. Reverse engineering activities include analysis and documentation of all existing systems, applications and structures of the enterprise. The result is a set of models that represents an integrated view of the enterprise, with redundancies and discrepancies resolved and documented.

To be most useful, models representing all conceptual, logical and physical architecture components should be maintained in a computer-based Enterprise Architecture repository. The table below illustrates examples of what John calls "primitive" architecture artifacts that can be documented and managed in such a repository.

	DATA What	**FUNCTION How**	**NETWORK Where**	**PEOPLE Who**	**TIME When**	**MOTIVATION Why**
SCOPE Contextual (Planner)	Enterprise Data Dictionary	Functional Hierarchy Breakdown	Locations List	Organization & Stakeholder View list	Business Event	Strategic Plan Goals, CSF etc
ENTERPRISE MODEL Conceptual (Owner)	Enterprise Meta Model	Activities within Function & Costing	Location Hierarchy & Geography	Organization & Stakeholder Hierarchy	Business Event Hierarchy	Business & Operational Plan
SYSTEM MODEL Logical (Designer)	Logical Data Model	Process Model or Detailed Activities	Logical Network Model	Organization & Stakeholder Roles	Logical Process Model or Sequence diagram	Business Rules & Sys Requirements w/ Logical links
TECHNICAL MODEL Physical (Builder)	Physical Database Design	Object model/ Components or structured	Network Design*[3]	Security design for App/DB/Net*[4]	Job Scheduler*[5] & App Modules for Events	Rules for DB/App Components
DETAILS Implemented (Sub-Contractor)	SQL DDL/ DBMS Tables*[1]	Source Code/ Executable Applications	Network Cabling & protocols*[3]	Net/App/ DBMS Security	Scheduled Batch & Online apps	App modules & DB tables, Data and Function Details

The repository becomes the single source for every "primitive" model, composite model and report regardless of the tool that is used to create it. For that reason the underlying "repository engine" should be able to import and export models from a wide variety of tools.

Effective EA Engineering means addressing the following critical success factors:

- Sponsorship and Involvement
- Business Requirements
- Enterprise Architecture Models
- Development Environment

These CSF are described in more detail in another of the author's papers.

Summary

Every enterprise has an architecture. Most enterprises simply let their architecture grow and evolve uncontrolled. It is always undocumented and is most often characterized by high maintenance costs, long development cycles, poor quality software, non-interoperable applications, lack of data sharing, limited strategic information, and difficult change management.

Enterprises that *engineer* their architecture are able to avoid and/or overcome the problems resulting from out-of-control architectures. An engineered Enterprise Architecture provides the blueprint for the enterprise's infrastructure and provides a means for managing complexity and managing change.

The Zachman Framework is the most comprehensive structure for classifying the basic building blocks of enterprise architecture.

The author's high performance EA Engineering approach, supported by effective, interoperable modeling and repository management tools provides the means to implement the Zachman Framework and develop enterprise architectures that are actionable and adaptable and that exactly meet enterprise goals and measures.

For more information please contact:

Visible Systems Corporation

24 School Street, 2nd floor
Boston, MA 02108
https://www.visiblesystemscorp.com/Strategy.html

Alan Perkins has been a Systems Analyst on the White House staff, Director of the US Army Data Processing School in Germany, Vice President of R&D for a virtual corporation, Vice President of Consulting for Visible Systems Corporation, and General Manager of a high-tech consulting firm. He has provided information and enterprise management consulting to numerous companies, associations and government agencies.

Mr. Perkins specializes in Enterprise Architecture Engineering. He helps clients quickly engineer enterprise architectures that are actionable and adaptable. His approach results in architectures that enable and facilitate enterprise initiatives such as Corporate Portals, Enterprise Data Warehouses, Enterprise Application Integration, Software Component Engineering, etc.

The following are papers are available at www.visible.com:

"Enterprise Architecture Engineering"

"Enterprise Architecture Engineering Critical Success Factors"

"*High-Performance* Enterprise Architecture Engineering – Implementing the Zachman Framework for Enterprise Architecture"

"Enterprise Change Management – An Architected Approach"

"Getting Your Acts Together – An Architected Solution for Government Transformation"

"A Strategic Approach to Data Warehouse Engineering"

"Data Warehouse Architecture – A Blueprint For Success"

"Critical Success Factors for Data Warehouse Engineering"

"How to Succeed in the 21st Century – Critical Information Management Success Factors"

"XML Metadata Management – Controlling XML Chaos"

"Business Rules Are Meta-Data"

"Enterprise Application Modernization – Solving IT's Biggest Problem"

"Strategic Enterprise Application Integration"

"e-Engineering – A Unified Method"

"Enterprise Portal Engineering"

"Quality Software [Component] Engineering"

"Software Engineering Process Improvement"