Zachman Framework

From Wikipedia, the free encyclopedia

The **Zachman Framework** is an Enterprise Architecture framework for enterprise architecture, which provides a formal and highly structured way of viewing and defining an enterprise. It consists of a two dimensional classification matrix based on the intersection of six communication questions (What, Where, When, Why, Who and How) with six rows according to reification transformations.[1]

	Why	How	What	Who	Where	When
Contextual	Goal List	Process List	Material List	Organizational Unit & Role List	Geographical Locations List	Event List
Conceptual	Goal Relationship	Process Model	Entity Relationship Model	Organizational Unit & Role Rel. Model	Locations Model	Event Model
Logical	Rules Diagram	Process Diagram	Data Model Diagram	Role relationship Diagram	Locations Diagram	Event Diagram
Physical	Rules Specification	Process Function Specification	Data Entity Specification	Role Specification	Location Specification	Event Specification
Detailed	Rules Details	Process Details	Data Details	Role Details	Location details	Event Details

Current view of the Zachman Framework.

The Zachman Framework is not a methodology in that it does not imply any specific method or process for collecting, managing, or using the information that it describes.[2] The Framework is named after its creator John Zachman, who first developed the concept in the 1980s at IBM. It has been updated several times since.[3]

The Zachman "Framework" is a schema for organizing architectural artifacts (in other words, design documents, specifications, and models) that takes into account both whom the artifact targets (for example, business owner and builder) and what particular issue (for example, data and functionality) is being addressed.[4]

Contents

- 1 Overview
- 2 History
  - 2.1 Information Systems Architecture Framework
  - 2.2 Extension and formalization
  - 2.3 Framework for enterprise architecture
  - 2.4 Extended and modified frameworks
- 3 Zachman Framework topics
  - 3.1 Concept
  - 3.2 Views of Rows
  - 3.3 Focus of Columns
  - 3.4 Models of Cells
  - 3.5 Framework set of rules
  - 3.6 Flexibility in level of detail
- 4 Applications and influences
  - 4.1 Customization
  - 4.2 Standards based on the Zachman Framework
  - 4.3 Mapping other frameworks
  - 4.4 Base for other enterprise architecture frameworks
    - 4.4.1 Example: One-VA Enterprise Architecture
- 5 See also
- 6 References
- 7 External links

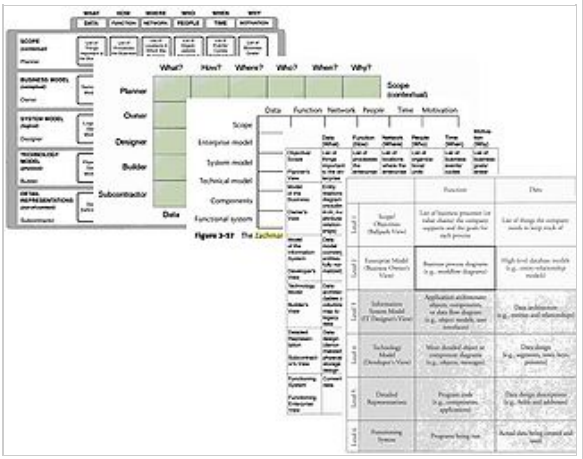
Overview

The term "Zachman Framework" has multiple meanings. It can refer to any of the frameworks proposed by John Zachman:

- The initial framework, named *A Framework for Information Systems Architecture*, by John Zachman published in an 1987 article in the IBM Systems journal.[5]
- The *Zachman Framework for Enterprise Architecture*, an update of the 1987 original in the 1990s extended and renamed .[6]
- One of the later versions of the Zachman Framework, offered by Zachman International as industry standard.

In other sources the Zachman Framework is introduced as a framework, originated by and named after John Zachman, represented in numerous ways, see image. This framework is explained as, for example:

- a framework to organize and analyze data,[7]
- a framework for enterprise architecture.[8]
- a classification system, or classification scheme[9]
- a matrix, often in a 6x6 matrix format
- a two-dimensional model[10] or an analytic model.
- a two-dimensional schema, used to organize the detailed representations of the enterprise.[11]



Collage of Zachman Frameworks as presented in several books on Enterprise Architecture from 1997 to 2005.

Beside the frameworks developed by John Zachman numerous extensions and or applications have been developed, which are also sometimes called Zachman Frameworks.

The Zachman Framework summarizes a collection of perspectives involved in enterprise architecture. These perspectives are represented in a two-dimensional matrix that defines along the rows the type of stakeholders and with the columns the aspects of the architecture. The framework does not define a methodology for an architecture. Rather, the matrix is a template that must be filled in by the goals/rules, processes, material, roles, locations, and events specifically required by the organization. Further modeling by mapping between columns in the framework identifies gaps in the documented state of the organization.[12]

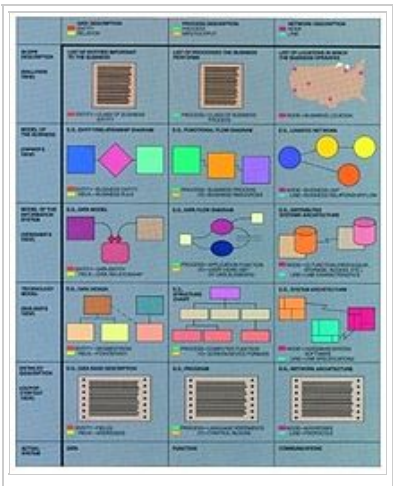
The framework is a simple and logical structure for classifying and organizing the descriptive representations of an enterprise. It is significant to both the management of the enterprise, and the actors involved in the development of enterprise systems.[13] While there is no order of priority for the columns of the Framework, the top-down order of the rows is significant to the alignment of business concepts and the actual physical enterprise. The level of detail in the Framework is a function of each cell (and not the rows). When done by IT the lower level of focus is on information technology, however it can apply equally to physical material (ball valves, piping, transformers, fuse boxes for example) and the associated physical processes, roles, locations etc. related to those items.

History

In the 1980s John Zachman had been involved at IBM in the development of Business System Planning (BSP), a method for analyzing, defining and designing an information architecture of organizations. In 1982 Zachman[14] had already concluded that these analyses could reach far beyond automating systems design and managing data into the realms of strategic business planning and management science in general. It may be employed in the (in that time considered more esoteric) areas of enterprise architecture, data-driven systems design, data classification criteria, and more.[14]

Information Systems Architecture Framework

In the 1987 article "A Framework for Information Systems Architecture"[15] Zachman noted that the term "architecture" was used loosely by information systems professionals, and meant different things to planners, designers, programmers, communication specialists, and others.[16] In searching for an objective, independent basis upon which to develop a framework for information systems architecture, Zachman looked at the field of classical architecture, and a variety of complex engineering projects in industry. He saw a similar approach and concluded that architectures exist on many levels and involves at least three perspectives: raw material or data, function of processes, and location or networks.[16]



The Information Systems Architecture is designed to be a classification schema for organizing architecture models. It provides a synoptic view of the models needed for enterprise architecture. Information Systems Architecture does not define in detail what the models should contain, it does not enforce the modeling language used for each model, and it does not propose a method for creating these models.[17]

Extension and formalization

The original 1987 "Information Systems Architecture Framework".

In the 1992 article "Extending and Formalizing the Framework for Information Systems Architecture" John F. Sowa and John Zachman present the framework and its recent extensions and show how it can be formalized in the notation of conceptual graphs.[18] Also in 1992:

> John Zachman's co-author John Sowa proposed the additions of the Scope perspective of the 'planner' (bounding lists common to the enterprise and its environment) and the Detailed Representation perspective of the 'sub-contractor' (being the out of context vendor solution components). The Who, When and Why columns were brought into public view, the notion of the four levels of metaframeworks and a depiction of integration associations across the perspectives were all outlined in the paper. Keri Anderson Healey assisted by creating a model of the models (the framework metamodel) which was also included in the article.
>
> —Stan Locke, *Enterprise Convergence in Our Lifetime, from THE ENTERPRISE NEWSLETTER*[19]

	DATA *What*	FUNCTION *How*	NETWORK *Where*	PEOPLE *Who*	TIME *When*	MOTIVATION *Why*
Objective/Scope (contextual) *Role: Planner*	List of things important in the business	List of Business Processes	List of Business Locations	List of important Organizations	List of Events	List of Business Goal & Strategies
Enterprise Model (conceptual) *Role: Owner*	Conceptual Data/ Object Model	Business Process Model	Business Logistics System	Work Flow Model	Master Schedule	Business Plan
System Model (logical) *Role:Designer*	Logical Data Model	System Architecture Model	Distributed Systems Architecture	Human Interface Architecture	Processing Structure	Business Rule Model
Technology Model (physical) *Role Builder*	Physical Data/Class Model	Technology Design Model	Technology Architecture	Presentation Architecture	Control Structure	Rule Design
Detailed Representation (out of context) *Role: Programmer*	Data Definition	Program	Network Architecture	Security Architecture	Timing Definition	Rule Speculation
Functioning Enterprise *Role: User*	Usable Data	Working Function	Usable Network	Functioning Organization	Implemented Schedule	Working Strategy

Simple example of the 1992 Framework.

Later during the 1990s[19]

- Methodologists like Clive Finkelstein refocused on the top two framework rows which he labeled Enterprise Engineering and has one of the most successful methods for converging the business needs with information engineering implementation, and determining a logical build sequence of the pieces.

Framework for enterprise architecture

In the 1997 paper "Concepts of the Framework for Enterprise Architecture" Zachman explained that the framework should be referred to as a "Framework for Enterprise Architecture", and should have from the beginning. In the early 1980s however, according to Zachman, there was "little interest in the idea of Enterprise Reengineering or Enterprise Modeling and the use of formalisms and models was generally limited to some aspects of application development within the Information Systems community".[20]

In 2008 Zachman Enterprise introduced the Zachman Framework: The Official Concise Definition as a new Zachman Framework standard.

Extended and modified frameworks

Since the 1990s several extended frameworks have been proposed, such as:

- Matthew & McGee (1990)[21] extended the three initial perspectives "what", "how" and "where", to event (the "when"), reason (the "why") and organization (the "who").[16]
- Evernden (1996) presented an alternative Information FrameWork.
- The Integrated Architecture Framework developed by Capgemini since 1996.[22]
- Vladan Jovanovic et all (2006) presents a Zachman Cube, an extended of the Zachman Framework into a multidimensional Zachman's Cube.[23]

Zachman Framework topics

Concept

The basic idea behind the Zachman Framework is that the same complex thing or item can be described for different purposes in different ways using different types of descriptions (e.g., textual, graphical). The Zachman Framework provides the thirty-six necessary categories for completely describing anything; especially complex things like manufactured goods (e.g., appliances), constructed structures (e.g., buildings), and enterprises (e.g., the organization and all of its goals, people, and technologies). The framework provides six different transformations of an abstract idea (not increasing in detail, but transforming) from six different perspectives.[24]

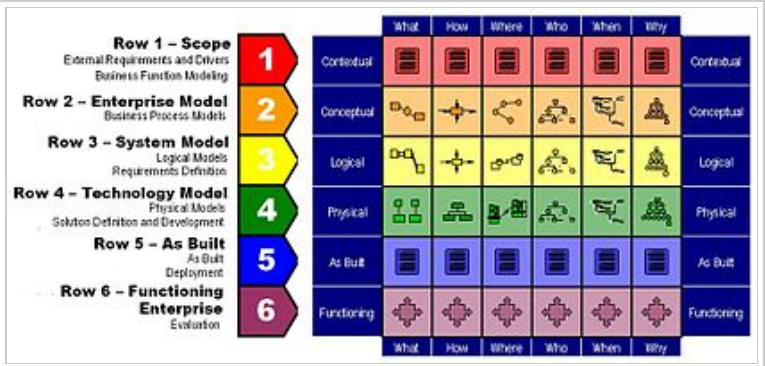
It allows different people to look at the same thing from different perspectives. This creates a holistic view of the environment, an important capability illustrated in the figure.[25]

Views of Rows

Each row represents a total view of the solution from a particular perspective. An upper row or perspective does not necessarily have a more comprehensive understanding of the whole than a lower perspective. Each row represents a distinct, unique perspective; however, the deliverables from each perspective must provide sufficient detail to define the solution at the level of perspective and must translate to the next lower row explicitly.[26]

Each perspective must take into account the requirements of the other perspectives and the restraint those perspectives impose. The constraints of each perspective are additive. For example, the constraints of higher rows affect the rows below. The constraints of lower rows can, but do not necessarily affect the higher rows. Understanding the requirements and constraints necessitates communication of knowledge and understanding from perspective to perspective. The Framework points the vertical direction for that communication between perspectives.[26]

In the 1997 Zachman Framework the rows are described as follows:[26]



The Veterans Affairs Zachman Framework with an explanation of its rows.[27][28]

- *Planner's View* (Scope) - The first architectural sketch is a "bubble chart" or Venn diagram, which depicts in gross terms the size, shape, partial relationships, and basic purpose of the final structure. It corresponds to an executive summary for a planner or investor who wants an overview or estimate of the scope of the system, what it would cost, and how it would relate to the general environment in which it will operate.
- *Owner's View* (Enterprise or Business Model) - Next are the architect's drawings that depict the final building from the perspective of the owner, who will have to live with it in the daily routines of business. They correspond to the enterprise (business) models, which constitute the designs of the business and show the business entities and processes and how they relate.
- *Designer's View* (Information Systems Model) - The architect's plans are the translation of the drawings into detail requirements representations from the designer's perspective. They correspond to the system model designed by a systems analyst who must determine the data elements, logical process flows, and functions that represent business entities and processes.
- *Builder's View* (Technology Model) - The contractor must redraw the architect's plans to represent the builder's perspective, with sufficient detail to understand the constraints of tools, technology, and materials. The builder's plans correspond to the technology models, which must adapt the information systems model to the details of the programming languages, input/output (I/O) devices, or other required supporting technology.
- *Subcontractor View* (Detailed Specifications) - Subcontractors work from shop plans that specify the details of parts or subsections. These correspond to the detailed specifications that are given to programmers who code individual modules without being concerned with the overall context or structure of the system. Alternatively, they could represent the detailed requirements for various commercial-off-the-shelf (COTS), government off-the-shelf (GOTS), or components of modular systems software being procured and implemented rather than built.
- *Actual System View* or The Functioning Enterprise

Focus of Columns

In summary, each perspective focuses attention on the same fundamental questions, then answers those questions from that viewpoint, creating different descriptive representations (i.e., models), which translate from higher to lower perspectives. The basic model for the focus (or product abstraction) remains constant. The basic model of each column is uniquely defined, yet related across and down the matrix.[26] In addition, the six categories of enterprise architecture components, and the underlying interrogatives that they answer, form the columns of the Zachman Framework and these are:[24]

1. The data description — What
2. The function description — How
3. The Network description — Where

4. The people description — Who
5. The time description — When
6. The motivation description — Why

In Zachman's opinion, the single factor that makes his framework unique is that each element on either axis of the matrix is explicitly distinguishable from all the other elements on that axis. The representations in each cell of the matrix are not merely successive levels of increasing detail, but actually are different representations — different in context, meaning, motivation, and use. Because each of the elements on either axis is explicitly different from the others, it is possible to define precisely what belongs in each cell.[24]

Models of Cells

The kinds of models or architectural descriptive representations are made explicit at the intersections of the rows and columns. An intersection is referred to as a cell. Because a cell is created by the intersection of a perspective and a focus, each is distinctive and unique. Since each cell is distinctive and unique, the contents of the cell are normalized and explicit per the perspective's focus.[26]

The cell descriptions in the table itself uses general language for a specific set of targets. Below the focus of each cell in this particular Zachman Framework is explained:

	Why	How	What	Who	Where	When
Contextual	Goal List	Process List	Material List	Organizational Unit & Role List	Geographical Locations List	Event List
Conceptual	Goal Relationship	Process Model	Entity Relationship Model	Organizational Unit & Role Rel. Model	Locations Model	Event Model
Logical	Rules Diagram	Process Diagram	Data Model Diagram	Role relationship Diagram	Locations Diagram	Event Diagram
Physical	Rules Specification	Process Function Specification	Data Entity Specification	Role Specification	Location Specification	Event Specification
Detailed	Rules Details	Process Details	Data Details	Role Details	Location details	Event Details

Current view of the Zachman Framework.

Contextual

1. (Why) Goal List – primary high level organization goals
2. (How) Process List – list of all known processes
3. (What) Material List – list of all known organizational entities
4. (Who) Organizational Unit & Role List – list of all organization units, sub-units, and identified roles
5. (Where) Geographical Locations List – locations important to organization; can be large and small
6. (When) Event List – list of triggers and cycles important to organization

Conceptual

1. (Why) Goal Relationship Model – identifies hierarchy of goals that support primary goals
2. (How) Process Model – provides process descriptions, input processes, output processes
3. (What) Entity Relationship Model – identifies and describes the organizational materials and their relationships
4. (Who) Organizational Unit & Role Relationship Model – identifies enterprise roles and units and the relationships between them
5. (Where) Locations Model – identifies enterprise locations and the relationships between them
6. (When) Event Model – identifies and describes events and cycles related by time

Logical

1. (Why) Rules Diagram – identifies and describes rules that apply constraints to processes and entities without regard to physical or technical implementation
2. (How) Process Diagram – identifies and describes process transitions expressed as verb-noun phrases without regard to physical or technical implementation
3. (What) Data Model Diagram – identifies and describes entities and their relationships without regard to physical or technical implementation
4. (Who) Role Relationship Diagram – identifies and describes roles and their relations to other roles by types of deliverables without regard to physical or technical implementation
5. (Where) Locations Diagram – identifies and describes locations used to access, manipulate, and transfer entities and processes without regard to physical or technical implementation
6. (When) Event Diagram – identifies and describes events related to each other in sequence, cycles occur within and between

events, without regard to physical or technical implementation

Physical

1. (Why) Rules Specification – expressed in a formal language; consists of rule name and structured logic to specify and test rule state
2. (How) Process Function Specification – expressed in a technology specific language, hierarchical process elements are related by process calls
3. (What) Data Entity Specification – expressed in a technology specific format; each entity is defined by name, description, and attributes; shows relationships
4. (Who) Role Specification – expresses roles performing work and workflow components at the work product detailed specification level
5. (Where) Location Specification – expresses the physical infrastructure components and their connections
6. (When) Event Specification – expresses transformations of event states of interest to the enterprise

Detailed Representation

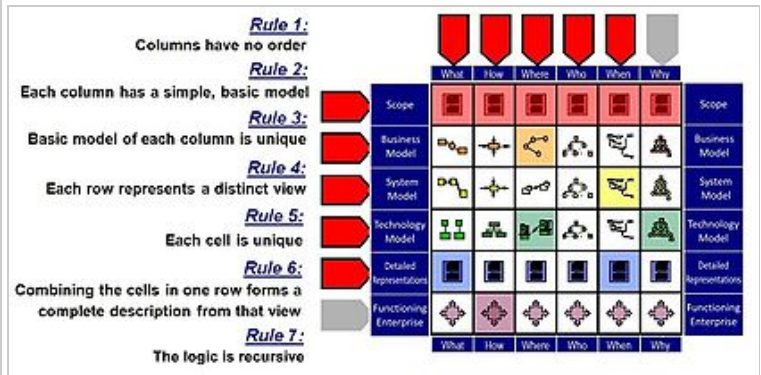
Eventually the cells with the detailed representation give Rules detail for (Why); Process detail for (How); Data detail for (What); Role detail for (Who); Location detail for (Where); and Event detail for (When).

There is a sixth row in the current Zachman framework, but it is not used for enterprise architecture — while the enterprise is described by rows one to six, enterprise architecture uses only rows one to five, thus only five rows are shown here.[3]

Since the product development (i.e., architectural artifact) in each cell or the problem solution embodied by the cell is the answer to a question from a perspective, typically, the models or descriptions are higher-level depictions or the surface answers of the cell. The refined models or designs supporting that answer are the detailed descriptions within the cell. Decomposition (i.e., drill down to greater levels of detail) takes place within each cell. If a cell is not made explicit (defined), it is implicit (undefined). If it is implicit, the risk of making assumptions about these cells exists. If the assumptions are valid, then time and money are saved. If, however, the assumptions are invalid, it is likely to increase costs and exceed the schedule for implementation.[26]

Framework set of rules

The framework comes with a set of rules:[29]



Example of Zachman Framework Rules.

- *Rule 1 The columns have no order* : The columns are interchangeable but cannot be reduced or created
- *Rule 2 Each column has a simple generic model* : Every column can have its own meta-model
- *Rule 3 The basic model of each column must be unique* : The basic model of each column, the relationship objects and the structure of it is unique. Each relationship object is interdependent but the representation objective is unique.
- *Rule 4 Each row describes a distinct, unique perspective* : Each row describes the view of a particular business group and is unique to it. All rows are usually present in most hierarchical organizations.
- *Rule 5 Each cell is unique* : The combination of 2,3 & 4 must produce unique cells where each cell represents a particular case. Example: A2 represents business outputs as they represent what are to be eventually constructed.
- *Rule 6 The composite or integration of all cell models in one row constitutes a complete model from the perspective of that row* : For the same reason as for not adding rows and columns, changing the names may change the fundamental logical structure of the Framework.
- *Rule 7 The logic is recursive* : The logic is relational between two instances of the same entity.

The framework is generic in that it can be used to classify the descriptive representations of any physical object as well as conceptual objects such as enterprises. It is also recursive in that it can be used to analyze the architectural composition of itself. Although the framework will carry the relation from one column to the other, it is still a fundamentally structural representation of the enterprise and not a flow representation.

Flexibility in level of detail

One of the strengths of the Zachman Framework is that it explicitly shows a comprehensive set of views that can be addressed by enterprise architecture.[12] Some feel that following this model completely can lead to too much emphasis on documentation, as artifacts would be needed for every one of the thirty cells in the framework.

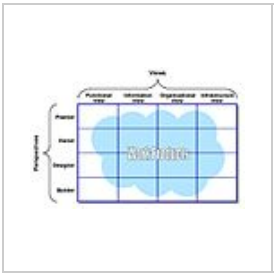
John Zachman clearly states in his documentation, presentations, and seminars that, as framework, there is flexibility in what depth and breadth of detail is required for each cell of the matrix based upon the importance to a given organization. An automaker, whose business goals may necessitate an inventory and process-driven focus, could find it beneficial to focus their documentation efforts on **What** and **How** columns. Whereas a travel agent company, whose business is more concerned with people and event-timing, could find it beneficial to focus their documentation efforts on **Who** and **When** columns. However, there is no escaping the **Why** column's importance as it provides the business drivers for all the other columns.

Applications and influences

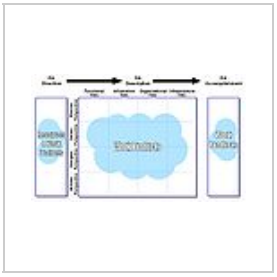
Since the 1990s the Zachman Framework has been widely used as a means of providing structure for Information Engineering-style enterprise modeling.[30] The Zachman Framework can be applied both in commercial companies and in government agencies. Within a government organization the framework can be applied to an entire agency at an abstract level, or it can be applied to various departments, offices, programs, subunits and even to basic operational entities.[31]

Customization

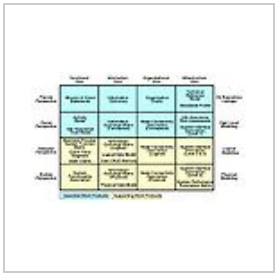
Zachman Framework is applied in customized frameworks such as the TEAF, built around the similar frameworks, the TEAF matrix.



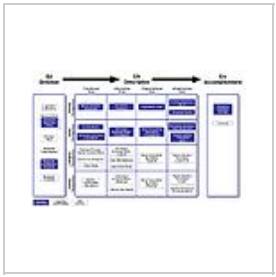
TEAF Matrix of Views and Perspectives.



Framework for EA Direction, Description, and Accomplishment Overview.



TEAF Products.



TEAF Work Products for EA Direction, Description, and Accomplishment.

Other sources:

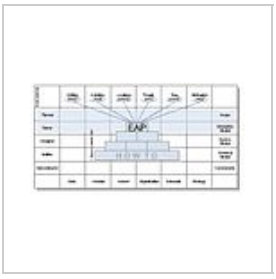
- The TEAF matrix is called a customization sample, see *here* (http://www.mega.com/wp/active/document/company/wp_mega_zachman_en.pdf) , p. 22

Standards based on the Zachman Framework

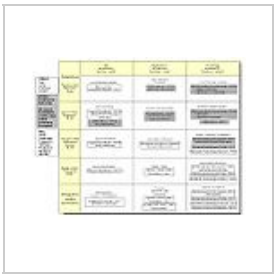
Zachman Framework is also used as a framework to describe standards, for example standards for healthcare and healthcare information system. Each cell of the framework contains such a series of standards for healthcare and healthcare information system.[32]

Mapping other frameworks

Another application of the Zachman Framework is as reference model for other enterprise architectures, see for example these four:

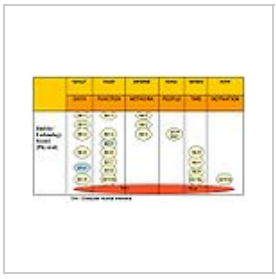


EAP mapped to the Zachman Framework, 1999



Mapping the C4ISR, 1999

DoD Products Map to the Zachman Framework Cells, 2003.



Mapping a part of the DoDAF, 2007.

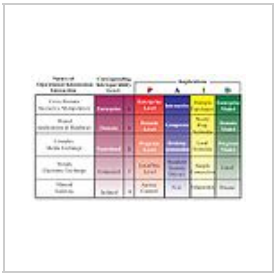
Other examples:

- Analysis of the Rational Unified Process as a Process,[33]
- How the Model-driven architecture (MDA) models used in software development map to the Zachman Framework.[34]
- Mapping the IEC 62264 models onto the Zachman framework for analysing products information traceability.[35]
- Mapping the TOGAF Architecture Development Method (e.g. the methodology) to the Zachman Framework.[6]

Base for other enterprise architecture frameworks

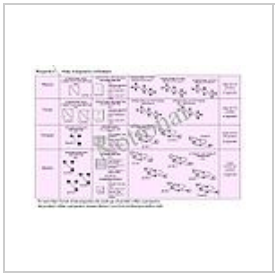
Less obvious are the ways the original Zachman framework has stimulated the development of other enterprise architecture frameworks, such as in the NIST Enterprise Architecture Model, the C4ISR AE, the DOE AE, and the DoDAF:

NIST Enterprise Architecture Model.[26]



C4ISR AE, 1997.

DOE AE, 1998.

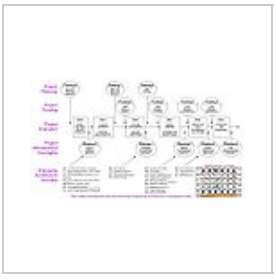


DODAF, 2003.

- The Federal Enterprise Architecture Framework (FEAF) is based on the Zachman Framework but only addresses the first three columns of Zachman, using slightly different names, and focuses in the top of the three rows.[36] (see *here* (http://books.google.nl/books?id=QjB5c_v-uMwC&pg=PA51&dq=%22Zachman+Framework%22+updated&lr=lang_en&as_brr=0&as_pt=ALLTYPES))

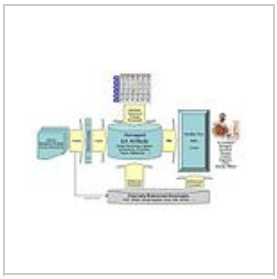
Example: One-VA Enterprise Architecture

The Zachman Framework methodology has for example been used by the United States Department of Veterans Affairs (VA) to develop and maintain its One-VA Enterprise Architecture in 2001. This methodology required defining all aspects of the VA enterprise from a business process, data, technical, location, personnel, and requirements perspective. The next step in implementing the methodology has been to define all functions related to each business process and identify associated data elements. Once identified, duplication of function and inconsistency in data definition can be identified and resolved, .[37]



Integrated Process Flow for VA IT Projects (2001)

VA Zachman Framework Portal



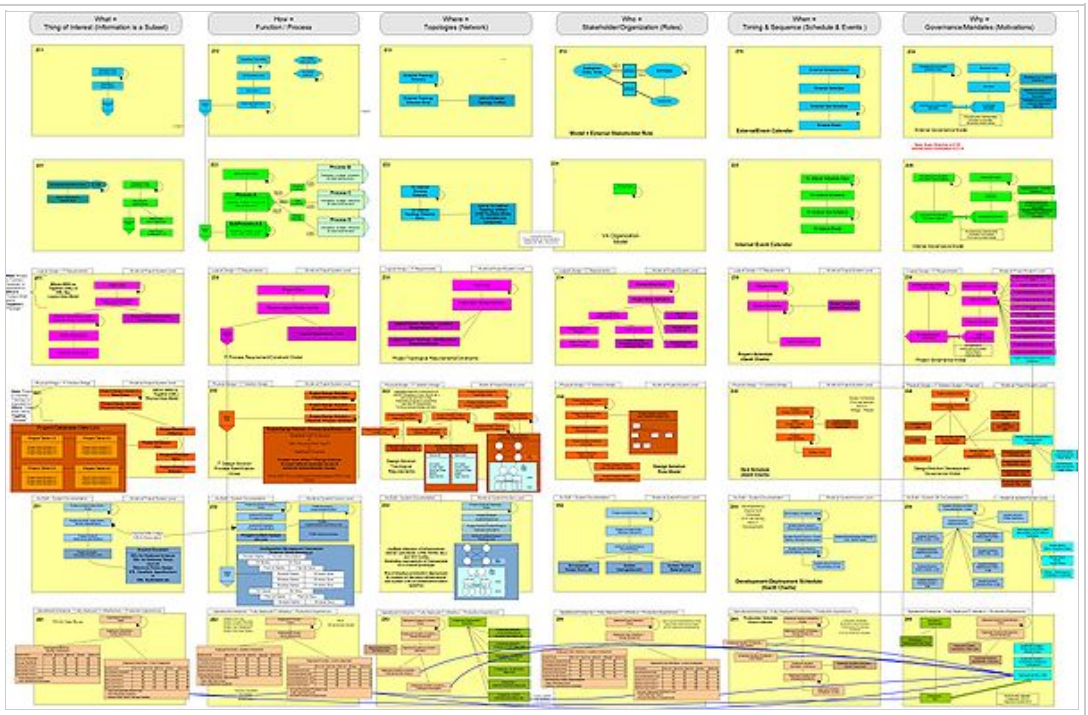
VA EA Repository Introduction (2008)

A Tutorial on the Zachman Architecture Framework

The Department of Veterans Affairs at the beginning of the 21st century planned to implement an enterprise architecture fully based on the Zachman Framework.

- The Zachman Framework was used as a reference model to initiate enterprise architecture planning in 2001.
- Somewhere in between the VA Zachman Framework Portal was constructed.
- This VA Zachman Framework Portal is still in use as a reference model for example in the determination of EA information collected from various business and project source documents.
- Now somewhere in the past this "A Tutorial on the Zachman Architecture Framework".

Eventually an enterprise architecture repository was created at the macro level by the Zachman framework and at a cell level by the meta-model outlined below.[38]



VA EA Meta-Model Cell Details Enlarged.

This diagram[39] has been incorporated within the VA-EA to provide a symbolic representation of the metamodel it used, to describe the One-VA Enterprise Architecture and to build an EA Repository without the use of Commercial EA Repository Software. It was developed using an object oriented database within the Caliber-RM Software Product. Caliber-RM is intended to be used as a software configuration management tool; not as an EA repository.

However, this tool permitted defining entities and relationships and for defining properties upon both entities and relationships, which made it sufficient for building an EA repository, considering the technology available in early 2003. The personal motivation in selecting this tool was that none of the commercial repository tools then available provided a true Zachman Framework representation, and were highly proprietary, making it difficult to incorporate components from other vendors or from

open source.

This diagram emphasizes several important interpretations of the Zachman Framework and its adaptation to information technology investment management.

1. Progressing through the rows from top to bottom, one can trace-out the Systems Development Life Cycle (SDLC) which is a de facto standard across the Information Industry;
2. The diagram emphasizes the importance of the often-neglected Zachman Row-Six (the Integrated, Operational Enterprise View). Representations in Mr. Zuech's interpretation of Zachman row-six consist, largely, of measurable service improvements and cost savings/avoidance that result from the business process and technology innovations that were developed across rows two through five.

Row-six provides measured return on investment for Individual Projects and, potentially, for the entire investment portfolio. Without row-six the Framework only identifies sunk-cost, but the row-six ROI permits it to measure benefits and to be used in a continuous improvement process, capturing best practices and applying them back through row-two.

See also

- Data model
- Enterprise Architecture framework
- Enterprise Architecture Planning
- FDIC Enterprise Architecture Framework
- View model

References

1. ^ "John Zachman's Concise Definition of the The Zachman Framework" (http://zachman.com/about-the-zachman-framework) . Zachman International. 2008. http://zachman.com/about-the-zachman-framework.
2. ^ "The Zachman Framework: The Official Concise Definition" (http://zachman.com/about-the-zachman-framework) . Zachman International. 2008. http://zachman.com/about-the-zachman-framework.
3. ^ ***a*** ***b*** "The Zachman Framework Evolution" (http://zachman.com/ea-articles-reference/54-the-zachman-framework-evolution) . Zachman International. April, 2009. http://zachman.com/ea-articles-reference/54-the-zachman-framework-evolution.
4. ^ *A Comparison of the Top Four Enterprise Architecture Methodologies* (http://msdn2.microsoft.com/en-us/library/bb466232.aspx) , Roger Sessions, Microsoft Developer Network Architecture Center,
5. ^ "A framework for information systems architecture" (http://zachman.com/images/ZI_PIcs/ibmsj2603e.pdf) . IBM SYSTEMS JOURNAL, VOL 26. NO 3,. 1987. http://zachman.com/images/ZI_PIcs/ibmsj2603e.pdf.
6. ^ ***a*** ***b*** The Open Group (1999–2006). "ADM and the Zachman Framework" (http://www.theopengroup.org/architecture/togaf8-doc/arch/chap39.html) in: *TOGAF 8.1.1 Online*. Accessed 25 Jan 2009.
7. ^ William H. Inmon, John A. Zachman, Jonathan G. Geiger (1997). *Data Stores, Data Warehousing, and the Zachman Framework: Managing Enterprise Knowledge*. McGraw-Hill, 1997. ISBN 0-07-031429-2.
8. ^ Pete Sawyer, Barbara Paech, Patrick Heymans (2007). *Requirements Engineering: Foundation for Software Quality*. page 191.
9. ^ Kathleen B. Hass (2007). *The Business Analyst as Strategist: Translating Business Strategies Into Valuable Solutions*. page 58.
10. ^ Harold F. Tipton, Micki Krause (2008). *Information Security Management Handbook, Sixth Edition, Volume 2*. page 263.
11. ^ O'Rourke, Fishman, Selkow (2003). *Enterprise Architecture Using the Zachman Framework*. page 9.
12. ^ ***a*** ***b*** James McGovern et al. (2003). *A Practical Guide to Enterprise Architecture*. p. 127-129.
13. ^ Marc Lankhorst et al. (2005). *Enterprise Architecture at Work*. p. 24.
14. ^ ***a*** ***b*** "Business Systems Planning and Business Information Control Study: A comparisment (http://www.research.ibm.com/journal/sj/211/ibmsj2101D.pdf) . In: *IBM Systems Journal*, vol 21, no 3, 1982. p. 31-53.
15. ^ John A. Zachman (1987). " A Framework for Information Systems Architecture" (http://www.research.ibm.com/journal/50th/applications/zachman.html) . In: *IBM Systems Journal*, vol 26, no 3. IBM Publication G321-5298.
16. ^ ***a*** ***b*** ***c*** Durward P. Jackson (1992). "Process-Based Planning in Information Resource Management". In: *Emerging Information Technologies for Competitive Advantage and Economic Development*. Proceedings of 1992 Information Resources Management Association International Conference. *Mehdi Khosrowpour (ed). ISBN 1-878289-17-9*.
17. ^ Alain Wegmann et al. (2008). "Augmenting the Zachman Enterprise Architecture Framework with a Systemic Conceptualization" (http://infoscience.epfl.ch/record/129325/files/Wegmann_et_al-SEAM_%26_Zachman-EDOC2008.pdf) . Presented at the 12th IEEE International EDOC Conference (EDOC 2008), München, Germany, September 15–19, 2008.
18. ^ John F. Sowa and John Zachman (1992). "Extending and Formalizing the Framework for Information Systems Architecture" (http://www.research.ibm.com/journal/sj/313/sowa.pdf) In: *IBM Systems Journal*, Vol 31, no.3, 1992. p. 590-616.
19. ^ ***a*** ***b*** Stan Locke (2008). "Enterprise Convergence in Our Lifetime" (http://www.ies.aust.com/ten/TEN42.htm#Enterprise_Convergence) In: THE ENTERPRISE NEWSLETTER, TEN42 September 16, 2008

20. ^ John A. Zachman (1997). "Concepts of the Framework for Enterprise Architecture: Background, Description and Utility (http://www.ies.aust.com/PDF-papers/zachman3.pdf) ". Zachman International. Accessed 19 Jan 2009.
21. ^ R.W. Matthews. &. W.C. McGee (1990). "Data Modeling for Software Development" (http://www.research.ibm.com/journal/sj/292/ibmsj2902F.pdf) . in: *IBM Systems Journal" 29(2). pp. 228–234*
22. ^ Jaap Schekkerman (2003). *How to Survive in the Jungle of Enterprise Architecture Frameworks*. page 139-144.
23. ^ Vladan Jovanovic, Stevan Mrdalj & Adrian Gardiner (2006). A Zachman Cube (http://www.iacis.org/iis/2006_iis/PDFs/Jovanovic_Mrdalj_Gardiner.pdf) . In: *Issues in Information Systems*. Vol VII, No. 2, 2006 p. 257-262.
24. ^ *a b c* VA Enterprise Architecture Innovation Team (2001). *Enterprise Architecture: Strategy, Governance, & Implementation* (http://www.va.gov/oirm/architecture/ea/2002/VAEAVersion-10-01.pdf) report Department of Veterans Affairs, August, 2001.
25. ^ The government information factory and the Zachman Framework (http://www.inmongif.com/_fileCabinet/gifzach.pdf) by W. H. Inmon, 2003. p. 4. Accessed July 14, 2009.
26. ^ *a b c d e f g* The Chief Information Officers Council (1999). Federal Enterprise Architecture Framework Version 1.1 (http://www.cio.gov/documents/fedarch1.pdf) . September 1999
27. ^ US Department of Veterans Affairs (2002) A Tutorial on the Zachman Architecture Framework (http://www.va.gov/oirm/architecture/EA/theory/tutorial.ppt) . Accessed 06 Dec 2008.
28. ^ Bill Inmon called this image "A simple example of The Zachman Framework" in the article John Zachman - One of the Best Architects I Know (http://www.b-eye-network.in/print/1962) Originally published 17 November 2005.
29. ^ Adapted from: Sowa, J.F. & J.A. Zachman, 1992, and Inmon, W.H, J.A. Zachman, & J.G. Geiger, 1997. University of Omaha (http://www.isqa.unomaha.edu/vanvliet/arch/ISA/isa.htm)
30. ^ Ian Graham (1995). *Migrating to Object Technology: the semantic object modelling approach*. Addison-Wesley, ISBN 0-201-59389-0. p. 322.
31. ^ Jay D. White (2007). *Managing Information in the Public Sector*. p. 254.
32. ^ ZACHMAN ISA FRAMEWORK FOR HEALTHCARE INFORMATICS STANDARDS (http://apps.adcom.uci.edu/EnterpriseArch/Zachman/Resources/ExampleHealthCareZachman.pdf) , 1997.
33. ^ DJ de Villiers (2001). "Using the Zachman Framework to Assess the Rational Unified Process" (http://www.ibm.com/developerworks/rational/library/content/RationalEdge/mar01/UsingtheZachmanFrameworktoAssesstheRUPMar01.pdf) , In: *The Rational Edge* Rational Software 2001.
34. ^ David S. Frankel et al. (2003) *The Zachman Framework and the OMG's Model Driven Architecture* (http://www.bptrends.com/publicationfiles/09-03%20WP%20Mapping%20MDA%20to%20Zachman%20Framework.pdf) White paper. Business Process Trends.
35. ^ Hervé Panetto, Salah Baïna, Gérard Morel (2007). Mapping the models onto the Zachman framework for analysing products information traceability : A case Study (http://hal.archives-ouvertes.fr/docs/00/11/91/96/PDF/Panetto_et_al_JIM.pdf) .
36. ^ Roland Traunmüller (2004). *Electronic Government* p. 51
37. ^ Statement of Dr. John A. Gauss, Assistant Secretary for Information and Technology, Department of Veterans Affairs (http://www.va.gov/oca/testimony/hvac/soi/13mr02it.asp) , before the Subcommittee on Oversight and Investigations Committee on Veterans' Affairs U.S. House of Representatives. March 13, 2002.
38. ^ Meta-Model Cell Details (http://www.va.gov/oit/ea/4_3/process/modeling/metamodel.html) Accessed 25 Dec 2009
39. ^ This diagram is the exclusive work of Albin Martin Zuech of Annapolis Maryland, who placed it in the public domain in 2001. Al Zuech maintains the original visio diagram in numerous stages of its development between 2000 and present. Al Zuech was the Director, Enterprise Architecture Service at the Department of Veterans Affairs from 2001 until 2007.

External links

- The Zachman Framework: The Official Concise Definition (http://zachman.com/about-the-zachman-framework) by John A. Zachman at Zachman International, 2009.
- The Zachman Framework Evolution (http://zachman.com/ea-articles-reference/54-the-zachman-framework-evolution) : overview of the evolution of the Zachman Framework by John P. Zachman at Zachman International, April 2009.
- UML, RUP, and the Zachman Framework: Better together (http://www.ibm.com/developerworks/rational/library/nov06/temnenco/) , by Vitalie Temnenco, IBM, 15 Nov 2006.