THE ZACHMAN FRAMEWORK: AN INTRODUCTION

David C. Hay, Essential Strategies, Inc.

This column is concerned with system development methodologies, and how they are related to each other. For the past twenty years or so, the industry has been deluged with methods, methodologies and techniques, each of which is supposed to radically improve productivity allowing us to deliver more, better systems with less energy. The great hopes have not been entirely met, but there have been some significant gains. Each new approach, while it doesn't represent the millenium (or a way of dealing with it), does contribute to the body of knowledge.

Enough has happened that the time seems to be right to examine that body of knowledge as a whole, and see what sorts of general conclusions we can draw from it. As it happens, John Zachman has developed a "framework" for doing just that. This column is to be about that framework.

Introduction – The System Development Life Cycle and John Zachman

Many methodologies are organized around the "system development life cycle", an organization of the steps required to develop systems. This is expressed as consisting of:

Strategy – The planning of an organization's overall systems development effort.

Analysis – The detailed definition of requirements for a particular area of the business.

Design – The specific application of technology to the requirements defined during analysis.

Construction – The actual construction of the system.

Documentation – Preparation of the user manuals, reference manuals, etc. to describe the system.

Transition – The implementation of the system, so as to make it part of the infrastructure of the organization.

Production – The ongoing monitoring of the system to ensure that it continues to meet the needs of the organization.

Notice that each of these steps addresses issues of data and function. Data and functions are typically addressed as separate topics, although ideally, they are addressed cooperatively.

Most methodologies portray the system development life cycle in terms approximating these. Some go so far as to give it the acronym "SDLC."

In 1987, John Zachman published a different approach to the elements of system development. Instead of representing the process as a series of steps, he organized it around the points of view taken by the various players. These players included (1) someone who has undertaken to do business in a particular industry, (2) the business people who run the organization, (3) the systems analyst who wants to represent the business in a disciplined form, (4) the designer, who applies specific technologies to solve the problems of the business, (5) the builder of the system, and finally (6) the system itself. Mr. Zachman represents each of these perspectives as a row in his matrix.

He then acknowledged that each of these participants were looking at the same categories of information. Just as a journalist asks, about a situation, what, how, who, where, when and why, he identified similar things that each participant should look at. If the roles are represented by rows in a matrix, the things examined can be represented by columns. Specifically, the columns represent the data manipulated by an organization (what), its functions and processes (how), locations where business is conducted (where), events that trigger business activities (when), the people and organizations involved (who), and the motivations and constraints which determine how the business behaves (why).

The set of cells thus defined turn out to house all the various modeling techniques we use in the information processing industry. Moreover, they identify some places where we could use some new models. In addition, the translations required to go from row to row reveal interesting things about both how we do and how we should do our business.

John Zachman's "Framework" is diagrammed in Figure 1. The rows represent the points of view of different players in the systems development process, while columns represent different aspects of the process. The players are:

1. ***Scope (Ballpark view):*** Definition of the enterprise's direction and business purpose. This is an industry view, concerned with the things that define the nature and purpose of the business. This is necessary to establish the context for any system development effort.
2. ***Model of the business (Owner's view):*** This defines — in business terms — the nature of the business, including its structure, functions, organization, and so forth.
3. ***Model of the information system (Architect's view):*** This defines the business described in step 2, but in more rigorous information terms. Where row two described business functions, for example, as perceived by the people performing them, row three describes them specifically as transformations of data. Where row two described all the things of interest to the enterprise, row three describes those things about which the organization wishes to collect and maintain information, and begins to describe that information.

4. ***Technology model (Designer's view):*** This describes how technology may be used to address the information processing needs identified in the previous rows. Here relational databases are chosen over network ones (or vice versa), kinds of languages are selected and program structures are defined, user interfaces are described, and so forth.
5. ***Detailed representations (Builder's view):*** This is a view of the program listings, database specifications, networks, and so forth that constitute a particular system. These are all expressed in terms of particular languages.
6. ***Functioning system:*** Finally, a system is implemented and made part of an organization.
The columns in the Zachman framework represent different areas of interest for each perspective. The columns describe the dimensions of the systems development effort. These are:
1. ***Data:*** Each of the rows in this column address understanding of and dealing with any enterprise's data. This begins in ***row one*** with an list of the things that concern any company in this industry, affecting its direction and purpose. As you pass down through the rows, you move to progressively more rigorous descriptions of the data (***row two*** is the business person's view, and ***row three*** is a disciplined translation of this), until you get to ***row four***, where a specific design approach (and a specific database management system) is specified. ***Row five*** is the detailed representation of the data on the computer (tablespaces and the like), and ***row six*** is the working database.
2. ***Function:*** The rows in the function column describe the process of translating the mission of the enterprise into successively more detailed definitions of its operations. Where ***row one*** is a list of the kinds of activities the enterprise conducts, ***row two*** describes these activities in a contiguous model. ***Row three*** portrays them in terms of data transforming processes, described exclusively in terms of the conversion of input data into output data. The technology model in ***row four*** then converts these data conversion processes into the definition of program modules and how they interact with each other. Pseudo-code is produced here. ***Row five*** then converts these into source and object code. ***Row six*** is where the code is linked and converted to executable programs.

	Andmed (Mis)	**Funktsioon (Kuidas)**	**Arvutivõrk (Kus)**	**Inimesed (Kes)**	**Aeg (Millal)**	**Motivatsioon (Miks)**
Eesmärk / Skoop	Ettevõttele oluliste objektide loetelu	Ettevõtte protsesside loetelu	Ettevõtte osakondade asukohtade loetelu	Organisatsiooni ühikute loetelu	Sündmuste ja äritsüklite loetelu	Ärieesmärkide loetelu
Äri-mudel	Objekt-suhte diagramm (m:m, n-suhted)	Äri-mudel (physical data flow diagram)	logistikavõrk (nodes and links)	Organisat-siooni skeem, rollid; kvalifikatsioon; andmetele juurdepääs	Ajagraafik	Äriplaan
Info-süsteemi mudel	Andme-mudel (1:m suhted ja normaliseeritud)	Tegelik andme-voogude diagramm application architecture	Hajutatud süsteemi-arhitektuur	Väline juurdepääs (roles, data, access)	Sõltuvus diagram objekti elu-tsükkel	Ärireeglite mudel
Tehnoloogia mudel	Andmete arhitektuur	Süsteemi Projekteerimine: struktuuri-diagramm	Süsteemi arhitektuur (hardware, software types)	Kasutaja liides (how the system will behave); security design	Juhtimis voogude diagram (juht-struktuur)	Ärireeglite projekteerimine
Detailne kavanda-mine	Füüsiline andme-mudel	Detailne programmi projekteerimine	Võrgu arhitektuur protokollid	Ekraani-vormid, andme-turbe arhitektuur (who can see what?)	Ajastus-mudel	Ärireeglid programmi-loogikas
	(Töötav süsteem)					
Töötav süsteem	Reaalsed andmed	Täitev programm	Kommunikatsiooni-vahendid	Välja-õpetatud inimesed	Reaalne äri-sündmus	Kehtestatud ärireeglid

Joonis 1: Zachmanni tugiraamistik

3. ***Network:*** This column is concerned with the geographical distribution of the enterprise's activities. At the strategic level (***row one***), this is simply a listing of the places where the enterprise does business. At ***row two***, this becomes a more detailed communications chart, describing how the various locations interact with each other. ***Row three*** produces the architecture for data distribution, itemizing what information is created where and where it is to be used. In ***row four***, this distribution is translated into the kinds of computer facilities that are required in each location, and in ***row five***, these facilities requirements are translated into specification of particular computers, protocols, communications facilities, and the like. ***Row six*** describes the implemented communications facilities.
4. ***People:*** The fourth column describes who is involved in the business and in the introduction of new technology. The ***row one*** model of people is a simple list of the organizational units and each unit's mission. In ***row two***, this list is fleshed out into a full organization chart, linked to the function column. Here also, requirements for security are described in general terms. In ***row three***, the potential interaction between people and technology begins to be specified, specifically in terms of who needs what information to do his job. In ***row four***, the actual interface between each person and the technology is designed, including issues of interface graphics, navigation paths, security rules and presentation style. In ***row five,*** this design is converted into the outward appearance of each program, as well as the definitions of access permissions in terms of specific tables and/or columns each user can have access to. In ***row six,*** you have trained people, using the new system.
5. ***Time:*** The fifth column describes the effects of time on the enterprise. It is difficult to describe or address this column in isolation from the others, especially column two. At the strategic (***row one***) level, this is a description of the business cycle and overall business events. In the detailed model of the business (***row two***), the time column defines when functions are to happen and under what circumstances. ***Row three*** defines the business events which cause specific data transformations and entity state changes to take place. In the technology model (***row four***), the events become program triggers and messages, and the information processing responses are designed in detail. In ***row five***, these designs become specific programs. In ***row six*** business events are correctly responded to by the system.
6. ***Motivation:*** As Mr. Zachman originally described this column, it concerned the translation of business goals and strategies into specific ends and means. This can be expanded to include the entire set of constraints that apply to an enterprise's efforts. In ***row one***, the enterprise identifies its goals and strategies in general, common language terms. In ***row two,*** these are translated into the specific rules and constraints that apply to an enterprise's operation. In ***row three***, business rules may be expressed in terms of information that is and is not permitted to exist. This includes constraints on the creation of rows in a database as well as on the updating of specific values. In ***row four***, these business rules will be converted to program design elements, and in ***row five*** they will become specific programs. In ***row six***, business rules are enforced.

Implications

This approach has several immediate effects on our understanding of the SDLC:
First of all, the analysis phase typically takes on two different perspectives: one is to describe the situation in purely business terms, while the second, without yet addressing technology, describes the situation in information processing terms.
Second, Mr. Zachman specifically addresses more than the data and functions we usually concern ourselves with. His matrix encompasses, for each phase, data, function, location, people, time, and motivation.
Third, he doesn't call them "phases" or "steps." Each row in his matrix represents the perspective of one of the set of players in the development process. It is more important, he asserts, to recognize that systems are developed by distinct groups with different points of view, than that it is to see the movement of systems from one step to another.
Finally, he does not address either documentation or transition explicitly. The matrix itself provides an organization for system documentation. And transition is the process of moving from the "as is" matrix to the "to be" matrix.
What does it mean to view the development process in these terms?
The major contribution of the Framework is its explicit recognition that there is more at work here than functions and data. From the beginning, we should be recognizing the organizational issues; from the beginning, we should be dealing with multiple locations; from the beginning we should be explicitly concerned with timing – triggers, schedules, and so forth.
We do not have models, or even well developed methods for dealing with many of the cells. Mr. Zachman does not advocate the use of any particular modeling style for those cells where multiple techniques are

available, and he is the first to recognize that in some cells no good techniques exist. It is difficult, for example, to model the logic (row three) of a distributed information network – at least in a way that links to our models for functions and data.

This represents an assignment for us all. He has pointed out things that we should be capturing and accounting for in our systems. It is for us to figure out how to do so.

This column will be dedicated to the analysis of the various parts of Mr. Zachman's framework, and their implications on how we go about our business. Each issue I will describe one or more cells in detail, the techniques currently available for addressing those cells, what new ones could be invented, and the relationship of those cells to others.

Dave Hay is Founder and President of Essential Strategies, Inc., a firm specializing in computer-aided systems and information engineering, business process re-engineering, modeling, and the management of systems implementation efforts. He has over twenty-five years experience in the information processing industry, with the last ten of that focused on using CASE for strategic information planning and systems analysis. He is the author of the book, Data Model Patterns: Conventions of Thought, from Dorset House. Dave can be reached at (713) 464-8316 or davehay@essentialstrategies.com.