

Enterprise Architecture: Looking Back and Looking Ahead

by

John A. Zachman

Ó Copyright 1999 Zachman International

Looking at the Past First

For those readers who have heard me talk about the subject of Enterprise Architecture, you will probably remember the frog ... I have a lot of years of my professional life invested in Architecture and related subjects!! These are NOT new ideas!!! However, historically, as most information professionals would also recognize, in spite of the overwhelming logic of Architectural concepts, those of us who care about these issues have not exactly been doing a “land office business” putting these ideas into practice.

I can think of four reasons why the reality of the deplorable lack of Architecture defies the logic of Architecture itself, the Information Age Enterprise IMPERATIVE!

Architecture Is Counter-Cultural

First, Architecture is counter-cultural. For the first 50 years of experience in the discipline of Information Systems, it has been challenging enough to simply get the systems to run. Our whole focus has been on producing running code. In fact, all of the typical performance measures are directed to that end including lines of code, function points, up time, down time, response time, mean time, expenses, budget, etc., etc. Whatever you measure, that is what you get ... running code.

Although we pay verbal homage to standards, reuse, interchangeable parts, integration, design for change, administering change, alignment, assets, investments, and so on, the practical fact of the matter is, we are not DOING any of them. Our deficiency in this regard is complicated by the fact that the accountants, I believe, are really not that adept at putting values on things that are going to be used more than one time, with the exception maybe, of used cars. There is not a ready secondary market to help put residual values on bridges, roads, armies, aircraft carriers, information, models, repositories, architecture, business rules, many assets in general. The problem with many of these kinds of assets is, when you need them for survival, you might not have them unless someone has had sufficient foresight to acquire them beforehand, whether they can be “cost-justified” in a classic I/T manner, or not.

So, Architecture is counter-cultural. We don’t know how to measure it so we are not getting it, and no one yet sees it as a sufficiently critical survival issue to acquire it,

whether they can measure it or not ... which brings me to the second reason we don't have Enterprise Architecture in place.

Architecture is Not Perceived a Survival Issue

The Enterprise has not associated Enterprise Architecture with survival.

Making money during the Industrial Age has been relatively easy. This is clearly a gross over-simplification but basically, what was needed in the Industrial Age was a good product and somebody to sell it. The marketplace was big enough to absorb whatever products or services could be produced. You could spill more than you made and still make a profit. The advent of the computer contributed immensely by removing labor constraints. An Enterprise could reduce costs and lower prices by replacing labor with technology. There was a great deal of room for productivity improvements through the use of automation.

As the Industrial Age has matured, the marketplace has become saturated. Products have become commoditized. It is pretty hard to tell the difference between Southern California Edison's electricity and Cleen 'n Green's electricity. You can shop for Guess Jeans in Johannesburg just the same as in Los Angeles. Benneton Colors stores look the same in Stockholm as they do in Freeport, Maine. The concept of a local market is now defunct. The game has changed. Competition is global and intense. The "system" no longer is a productivity aid. The "system" is the price of entry into the game. You cannot even play in the game without extensive automation. The systems are no longer discretionary support to the Enterprise. They are mandatory. Systems have replaced the people of the Enterprise and are doing the work of the Enterprise. In fact, many of the folks that have been in my seminars or at my presentations have heard me say time and again, "the systems ARE the Enterprise."

In any case, the second reason for lack of Architecture is the Enterprise has not yet discovered that the design of the system IS the design of the Enterprise and worse, if the system can't change, THE ENTERPRISE CAN'T CHANGE! However, be that as it may, Architecture still is not yet perceived by the Enterprise to be a survival issue.

We Haven't Known How to Actually DO Architecture

The third reason we don't see a lot of Architecture in reality is, we simply haven't known how to DO Architecture.

We owe Tom DeMarco(1) a huge debt of gratitude for teaching us how describe the functional aspects of Enterprise Architecture, certainly at rows 3 and 4 of the Zachman Framework(2). In fact, we know how to design those cells and I might add that the laws of coupling and cohesion are still as valid today as they ever were. Actually, Mike Hammer and Jim Champy (3) have more recently brought Row 2 of the Process column into focus with the concentration on Business Process Reengineering.

Peter Chen(4) and Bob Brown(5), Clive Finkelstein(6) and Charlie Bachman(7) have taught us how to describe and how to design the Column 1 models, the Data Models.

And, the laws of semantic integrity have not gone away any more than coupling and cohesion have gone away.

But, beyond Process and Data, the state of the art has been a little thin. We don't know a lot about describing or designing the Column 3 models, that is, the distribution models, the technology architecture, the "infrastructure," the hardware/systems software models, etc.

We are also a little thin in the area of work flow, presentation architecture, etc., the Column 4 models. Peter Drucker(8) has made major contributions in organizational concepts in Row 2 of Column 4, but below that, there is a lot to be learned.

In Column 5, the time column, Peter Senge(9) has made major contributions with the systems dynamics models, but we have not explicitly tied these to the "systems" Row 3 and below of Column 5.

Ron Ross(10) and Gladys Lam are making major progress in putting definition around Business Rules and their relationship to the Objectives and Strategies of the Enterprise which are made manifest in the Row 2 models of Column 6, the Motivation Column, but there are still limitations to our understanding and the work is certainly not yet universally accepted.

All-in-all, there is still a LOT to learn about Enterprise Architecture. We have clearly not known how to do it all and there were still some gaping holes in our understanding, but worst of all, even though we know quite a bit, still we are not practicing it! Our cultural mentality (as reflected in the discussion of reason number one above) still tends to be, "you start writing the code and I will go find out what the users have in mind!! ... Forget Architecture!!"

This brings me to the fourth and last reason why Architectural concepts are not prevalent in the practice of the Enterprise.

Architecture Requires Actual Work

Architecture requires actual work.

We keep looking for the "quick fix," a technological solution, a tool, a package, a new processor, the perennial "silver bullet." We wish we could simply throw money at the problem and have the pain go away.

This is simply not realistic! We have been looking for the "Holy Grail" of technology solutions to Enterprise problems for forty years that I personally know of and are no closer today than when we first started. In fact, I might argue, we are even farther away.

At the point in time when the Enterprise recognizes that the computer is not simply a productivity enhancement tool but that the "system" IS the Enterprise, the "Owner," "Designer" and "Builder"(11) will have to sit together, have a meeting of the minds and

decide what the Enterprise is and is to be, how to “architect” it in that context and then how to transform that architecture into reality, that is, to “construct” it (the Enterprise) according to the architectural specifications ... and then, maybe the key to the future, establish the Enterprise process for changing the Enterprise.

You probably wouldn't be able to do this without technology to aid in its creation, specification and construction as well as to be integral with its ongoing operation and change, however, technology, in and of itself, is not going to do any of this automatically. It will require actual collaborative work on the part of the Owner, Designer and Builder ... and further, it will require time. Enterprises are simply too complex to expect simplistic solutions.

We send Engineers to college for four years and on to graduate school for several more to basically learn the standard conventions for describing complex physical objects like buildings and airplanes, etc. Then, there is a decade or two of apprenticeship to gain experience before they are certified to carry the responsibilities for actually designing complex things. In contrast, we send people to THREE DAYS of COBOL programming school or THREE DAYS of data modeling school and expect them to be a competent COBOL programmer or a competent data modeler. These are the people we typically expect, with little more training or apprenticeship, to architect and construct our enterprises (that is, “systems,” remembering “the systems ARE the Enterprise”) ... which are orders of magnitude more complex than airplanes or buildings.

In Summary of the Past

In summary, the four reasons that Enterprise Architecture historically has not been a vital part of the Enterprise agenda are:

1. Architecture is counter-cultural.
2. It is not perceived to be a survival issue by the Enterprise.
3. We don't know how to actually do all of it.
4. It takes time and actual work.

Looking Toward the Future

Looking toward the future, there are four reasons why an Architectural revolution is imminent for every Enterprise that intends to be a player in the Information Age.

Architecture IS the New Culture

First, the game has changed. We are no longer playing an Industrial Age game. We are playing an Information Age game. No longer is all the information (and therefore all the power) concentrated in a very few people at the very top of an Enterprise. Now everybody has access to the same information at the same time. A “Powershift(12)” has taken place. The power shifts outboard within the Enterprise ... even beyond the enterprise as the customers have access to the same information as the Enterprise. No longer is the Enterprise challenge simply to produce a product (or service) and then look

for customers to sell it to. Now the challenge is to find a good customer and produce and integrate whatever products are necessary to meet that customer's requirements and keep them a good customer. This adds orders of magnitude of complexity to the Enterprise as the burden of integration now falls on the Enterprise, not on the customer. No longer is it possible to remember how the Enterprise works without formal conventions and mechanisms for describing it and retaining knowledge about it ... that is, without Architecture.

At the same time, the rate of change is escalating. "Knowledge is change and the ever-increasing body of knowledge feeding the 'great engine of technology' is creating ever-increasing change.(13)" Without architectural representations that describe the Enterprise as it exists at a given point in time, there is no way to understand what can be changed or to understand the impact of potential changes or to actually change the Enterprise, short of starting with a blank sheet of paper and attempting to re-design it from scratch every time you want to change.

Complexity and high rates of change are the challenge of the Information Age Enterprise and neither can be accommodated without Enterprise Architecture.

Architecture Is a Survival Issue

Second, Architecture IS a survival issue because of complexity and high rates of change and many Enterprises are failing for lack of it. Take IBM for example: Why are they no longer the dominant factor in the I/T marketplace? Is it because they were stupid, or because they didn't understand the changes in the marketplace? I worked there for 26 years and I can assure you that they were neither stupid nor were they insensitive to the market changes. I would suggest that even though they wanted to change, since there was no architecture, they couldn't change the systems, that is, couldn't change the *Enterprise*, didn't have understanding what and how to change and basically ran out of time.

What about General Motors? Why are they continuing to lose major portions of market share, take out tens of thousands of people and close down hundreds of sites? I have had very limited contact with General Motors, but I suspect it is not because they are stupid or because they aren't sensitive to the marketplace. In fact, I suspect that they can't react to the market pressures of mass-customization (the Information Age, automobile industry, "Powershift") and higher quality at lower prices because there is no architecture, they don't know how the Enterprise works, they do not know what and how to change in response to market pressures and they are running out of time.

I don't believe that these are isolated instances. I think they represent general trends and I submit that virtually every enterprise is confronted by the very same challenges. In fact, I have confidence that it won't be long before the academic community will validate the relationship between the lack of architecture and the inability to deal with complexity and high rates of change through their research and development of case studies for instructional purposes.

Peter Drucker makes a strong case in “The Next Information Revolution”(14) that the revolution is well underway, but not where information professionals are looking for it because it is not a revolution in technology, machinery, techniques, software or speed. It is a revolution in CONCEPTS. I would argue that the revolutionary concept is ARCHITECTURE! No longer is it adequate to simply get the system to run. Now, we must see the Enterprise holistically as the implementation itself so it can be dynamically re-shaped and re-defined as the environmental change and complexity escalates.

Now We Know How to DO Architecture

Third, we are rapidly filling in gaping holes in the understanding of how to actually do architecture.

In the last year, I have received two recently written books that make major contributions to the Architecture body of knowledge. Bernard H. Boar, recently retired from AT&T, sent me a manuscript of his book “Constructing Blueprints for Enterprise IT Architectures”(15). In this book, Bernie is proposing a formal architectural notation (Row 3) for application function and technology distribution (Column 2 and Column 3). I would observe that this is a breakthrough. It is equivalent to DeMarco’s 1978 publication(1) on Structured Analysis (Column 2, Row 3) or Chen’s 1977 publication(4) on logical data models (Column 1, Row 3) in that it is the first attempt, at least that I have seen, to establish a standard notation for describing the technology environment.

The second book I received is “High Performance Client Server: A Guide to Building and Managing Robust Distributed Systems”(16) by Chris Loosley and Frank Douglas. This book is also a major breakthrough in the nature of the early books on coupling and cohesion, and semantic integrity. Chris is describing how to decide where to put the Data (Column 1), where to put the Process (Column 2), and where to put the Presentation (Column 4) in relation to the Location (Column 3). Further, Chris argues that these decisions must be made at Row 2, NOT at Row 4 or Row 5.

Ron Ross and Gladys Lam are presently blazing new trails in describing and designing Business Rules (Column 6, Rows 2 and 3) in the Data To Knowledge Newsletter(17). Ron has written the definitive book on Business Rules (Column 6, Row 3), “The Business Rule Book: Classifying, Defining and Modeling Rules”(10). In fact, the Business Rules Group, of which Ron and I are a part, is on the verge of publishing the meta-model for Column 6, Row 2. (See www.businessrulesgroup.com).

Tom Bruce wrote the seminal book on Column 1, data models, "Designing Quality Databases with IDEF1X Information Models."(18)

Melissa Cook wrote “Building Enterprise Information Architectures”(19) and Steve Spewak wrote “Enterprise Architecture Planning”(20), both of which address Columns 1, 2 and 3, Rows 1 and at least a high level of Row 2.

The only places where we still are thin on theory are in Work Flow (Column 4), Dynamics (Column 5) in Rows 3 and below and Motivation (Column 6) Row 4 and

below, and I suspect that even these issues are well known and thoroughly documented in disciplines other than the information discipline.

There are few breakthroughs left to be made. However, I would argue that in essence, we KNOW how to DO Enterprise Architecture.

Tools Are Available for Actual Architecture Work

Fourth and last, the tools for supporting Enterprise Architecture are rapidly improving and removing all the remaining obstacles to doing actual Architecture work. The tools will never be a substitute for the intellectual work of producing Architecture, but they are reducing the time it takes to produce architectures by making the process more productive.

In the last year, I have seen substantive tool activity that is directly deriving from or in support of the Framework, including:

- a. A new release of Structure from Framework Software, Inc. which is an artifact classification and access tool,
- b. Framework from PTECH which is a graphic Enterprise modeling tool that has grown out of the process modeling discipline of the pre-IDEF0 days,
- c. METIS, an enterprise modeling tool out of Norway that has been acquired by NCR and used in support of their consulting practice,
- d. Design Bank from Metadata Management Corp. which is a Repository built to Framework specifications and which has evolved from Infospan origins,
- e. ystem Architect, a CASE tool from Popkin Software, which has evolved from the LBMS tool set,
- f. Visible Advantage, a CASE tool from Visible Systems whose lineage goes back to Clive Finkelstein's USER: Expert Systems, and
- g. Architect, from ZTI which is a Repository that can be traced back to Atkinson Tremblay's Assyst Developer.
- h. RuleTrack, a Business Rules analysis and planning tool from Framework Software that was designed in support of Business Rule Solutions' (Ron Ross/Gladys Lam) business rule methodology.

I am sure that I am not doing these tools justice with such a brief characterization but the thing that is impressive in every case is the fact that these are not just "johnny-come-lately" "silver bullet" candidates. In every case, they can be traced back to substantive, in-depth, theoretical foundations and maturity over decades of experience. Also in each case, the presently available product has been recently re-built with current technology and/or re-engineered specifically to accommodate Enterprise Architecture principles. In the words of a friend of mine from many years ago, Steve McMinamin, "the second time you design something, it is always better!" ... and all of these tools have many iterations of maturity.

Also, I am sure there are many other tools that are available and many other books that have been written. Life is too short. I am simply unable to see every tool or to read

every book. I don't even know all the books and tools specifically discussing or supporting my own Framework. The principle point to be observed is, there is no longer any major obstacle, theoretical or technical, for doing Enterprise Architecture. The only thing remaining to be done is doing the actual work of Architecture.

In summary, there are four reasons why the Enterprise Architecture revolution will occur quickly:

1. Architecture IS the Information Age Culture (and we are, like it or not, in the Information Age),
2. Architecture IS an Enterprise survival issue,
3. There are few theoretical breakthroughs left to be made, and
4. There are no technical obstacles for doing Architecture work.

Truly, Enterprise Architecture is the issue of the 21st Century. The playing field is fairly level at the moment, but the Enterprise that acquires the capability to deal with complexity and to accommodate high rates of change is going to be very difficult to compete with as we progress into maturity in the Information Age. Enterprise Architecture will quickly become the admission price to play in the Information Age game.

Note: For an extensive bibliography that supports Enterprise Architecture see the Zachman Institute for Framework Advancement (ZIFA) website, www.zifa.com.

References

1. DeMarco, T. *Structured Analysis and System Specification*. New York. Yourdon Press, 1978
2. Sowa, J. and Zachman, J. *Extending and Formalizing the Framework for Information Systems Architecture*. IBM Systems Journal, vol. 31, no. 3, 1992 Armonk, NY.
2. Hammer, M., Champy, J. *Reengineering the Corporation*. New York. Harper Business 1993
3. Chen, P.P. *The Entity Relationship Model - Toward a Unified View of Data*. ACM Transactions on Database Systems, Vol. 1 No. 1, March 1976
4. Brown, R.G., and the Data Base Design Group. *Logical Database Design Techniques*. Mountain View, CA. Data Base Design Group 1984
6. Martin, J. and Finkelstein, C. *Information Engineering*. London. Savant 1981
7. Bachman, C. *Data Structure Diagrams*. Communications of the ACM. SIGBDP Database Vol. 1, No. 2, 1969
8. Drucker, P. F. *The Practice of Management*. New York. Harper & Row 1954
9. Senge, P.M. *The Fifth Discipline: The Art and Practice of the Learning Organization*. New York. Doubleday, 1990
10. Ross, R.G. *The Business Rules Book: Classifying, Defining and Modeling Rules*. Boston. Database Newsletter, 1997

11. Zachman, John A. *A Framework of Information Systems Architecture*. IBM Systems Journal, Vol. 26, no. 3, 1987, Armonk, N.Y.
12. Toffler, A. *Powershift*. New York. Bantam, 1990
13. Toffler, A. *Future Shock*. New York. Bantam, 1970
14. Drucker, Peter F. *The Next Information Revolution*. Forbes ASAP, August 24, 1998.
15. Boar, Bernard H. *Constructing Blueprints for Enterprise IT Architectures*. Wiley, 1998
16. Looseley, Chris and Douglas, Frank. *High Performance Client Server: A Guide to Building and Managing Robust Distributed Systems*. Wiley 1998.
17. Ross, Ronald G., Lam, Gladys S.W. *Data To Knowledge Newsletter* (formerly published as the Data Base Newsletter). Business Rule Solutions, Inc., since 1973.
18. Bruce, Thomas A. *Designing Quality Databases with IDEFIX Information Models*. Dorset House 1992.
19. Cook, Melissa. *Building Enterprise Information Architectures*. Prentice Hall, 1996
20. Spewak, Steven H. *Enterprise Architecture Planning*. QED 1993.

*John A. Zachman
Zachman International
2222 Foothill Blvd. Suite 337
La Cañada, Ca. 91011
818-244-3763 (Phone and Fax)
johnzachman@compuserve.com*