Avoiding Unintended Consequences of Change

John A. Zachman
Zachman International

© 2018 John A. Zachman, Zachman International®
AGENDA

- Historical Perspective
- The New Paradigm
- Origin of Enterprise Architecture
- A Zachman Framework Story
- The Profession Service Cycle
- Conclusions
ORIGINS OF ENTERPRISE ARCHITECTURE

- Frederick Taylor "Principles of Scientific Management" 1911
- Walter A. Shewhart "The Economic Control of Quality of Manufactured Product" 1931 (Dr. Edward Demming's Mgr.)
- Peter Drucker "The Practice of Management" 1954
- Jay Forrester "Industrial Dynamics" 1961
- Peter Senge "The Fifth Discipline" 1990
- Sherman Blumenthal "Management Information Systems: A Framework for Planning and Development" 1969
- Alvin Toffler "Future Shock" 1970
- George Steiner "Comprehensive Managerial Planning" 1972
- Etc., etc., etc.
“Computers!”

Better!
Quality!!

Faster!
Time!!

Cheaper!
Money!!

Every minute it’s not implemented, it is costing: quality, time and money!!

Intense motivation to get to implementation ASAP!
The Information Age

"Future Shock" (1970) - The rate of change.
"The Third Wave" (1980) - The structure of change.
"Powershift" (1990) - The culture of change.

-Alvin Toffler

Agricultural Age → Industrial Age → Information Age

Complexity and Change

© 2018 John A. Zachman, Zachman International®
The Information Age

"Future Shock" (1970) - The rate of change.
"The Third Wave" (1980) - The structure of change.
"Powershift" (1990) - The culture of change.
-Alvin Toffler

Agricultural Age

Industrial Age

Information Age

Extreme Complexity! and
Extreme Change!
Complexity and Change

The Challenge
What is your strategy (Chief) for addressing
Orders of magnitude increases in complexity
and
Orders of magnitude increases in the rate of change?

Seven thousand years of history would suggest that the only known strategy for addressing complexity and change is:

ARCHITECTURE

If it (whatever it is) gets so complex that you can’t remember how it works (at the level of definition required to create it)
you have to write it down,
that is: ARCHITECTURE

If you ever want to change what you have created,
you have to retain the descriptive representations you created to identify and avoid the unintended consequences of change
that is: ARCHITECTURE

The KEY to accommodating complexity and change is:

ARCHITECTURE
New Paradigm

Enterprise Architecture is NOT about “Building and Running Systems”
(Building and Running Systems is a MANUFACTURING idea.)

OLD (Industrial Age) Paradigm

Enterprise Architecture is an ENGINEERING idea.
(Which is a DIFFERENT idea.)
NEW (Knowledge Age) Paradigm

The end object is NOT to get the code to run.
The end object is to create and manage the Knowledgebase required to design and change the Enterprise …

… DYNAMICALLY!!
Information Technology (IT) is a manufacturing business.

A “system” literally IS your business.

Therefore, IT has actually “manufactured” YOUR business.

BUT … your business was never “engineered” (i.e. never “designed.”)

Therefore, IT isn’t “manufacturing” your business …

They are only “manufacturing” PARTS of your business.
But the parts don’t fit together!!
(That is, in the IT vernacular, the systems are not “Enterprise-wide INTEGRATED??”)

What do you do with PARTS that don’t fit together??

SCRAP AND REWORK!!
(Nobody wants to hear this!!)

If you want PARTS to fit together,
you have to ENGINEER them to fit together
BEFORE you manufacture them!

(“Programming is manufacturing, NOT engineering.”)

Fred Brooks
ENGINEERING derived characteristics:

**Enterprise (i.e. Business):**
- Integration
- Interoperability
- Flexibility
- Alignment
- Reusability
- etc., etc.

You will never realize these engineering derived characteristics without doing any engineering!
MANUFACTURING vs ENGINEERING

Manufacturing work requires multi-variable, holistic descriptions of parts of the object.

(Analysis - Decomposition) (Composite)

(This is the current OLD paradigm)

IN CONTRAST

Engineering work requires single-variable, ontologically-defined descriptions of the whole of the object.

(Normalization - Synthesis) (Primitive)

(This is the NEW paradigm)
Complexity

Reduce the sample size through Classification

One Dimensional Decomposition (Hierarchy, “Taxonomy”)
The deeper the tree, the smaller the parts (faster and cheaper).
The same content can occur in multiple nodes.

ANALYSIS
Lends itself to implementation (Manufacturing)
The problem is, this not only disintegrates the Data, it disintegrates EVERYTHING, Data, Process, Location, Responsibility, Timing and Motivation.

Note: This slide is intended to illustrate the legacy disintegration problem which could be resolved by governance, requiring REUSE of the single-variable, ontologically-defined, “Primitive” components in the creation of every “Composite” implementation instantiation.
Unintended Consequences

You want to avoid the unintended consequences of change??

Don’t do this!!
(Old Paradigm)
The end object is
NOT to get the code to run!!!
(If you have to replicate something,
it has to be “controlled.”)

The end object is
to design the Enterprise to accommodate
EXTREME Complexity and DYNAMIC Change!!
INTRODUCTION TO ENTERPRISE ARCHITECTURE

ORIGIN OF ENTERPRISE ARCHITECTURE

JOHN A. ZACHMAN
ZACHMAN INTERNATIONAL

© 2018 John A. Zachman, Zachman International®
Complexity

Reduce the sample size through Classification

One Dimensional
Decomposition (Hierarchy, “Taxonomy”)
The deeper the tree, the smaller the parts (faster and cheaper).
The same content can occur in multiple nodes.

ANALYSIS
Lends itself to implementation (Manufacturing)

Multi Dimensional
Normalization (Matrix, Cube)
One (type of) fact in one place (set theory).
Identify and eliminate recurrences (redundancies)

SYNTHESIS
Lends itself to design (Engineering)
"Ontology"

<table>
<thead>
<tr>
<th>Classification</th>
<th>Names</th>
<th>Model</th>
<th>Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audience Perspectives</td>
<td>Executive Perspective (Business Context Planners)</td>
<td>Architect Perspective (Business Concept Owner)</td>
<td>Engineer Perspective (Business Logic Designer)</td>
</tr>
<tr>
<td>Inventory Identification</td>
<td>Process Identification</td>
<td>Distribution Identification</td>
<td>Responsibility Identification</td>
</tr>
<tr>
<td>What</td>
<td>How</td>
<td>Where</td>
<td>Who</td>
</tr>
</tbody>
</table>

This is simple ... remember 1st Grade?
Put the alligators in one box
Put the fire engines in a different box
Put the people in another box etc., etc., etc.
(Organizing your Knowledgebase for Engineering design)
Ordered structure for accumulating normalized increments of Business Knowledge, iteratively/incrementally, little by little, issue by issue, CEO’s problem by CEO’s problem from here to eternity.

Key to the Knowledge Age

© 2018 John A. Zachman, Zachman International®
ALL business knowledge and ALL Principles need not be completed or in constant practice to continue operations or do new work. This can be defined/designed little by little
If you do not have these ontological descriptions, I submit, nobody knows how the Enterprise works, certainly not at the level of definition required to create it or change it.
**Business Knowledge Engineering**

**Traceability, Impact Assessment & Compliance**

**Governing Rules**


What is happening here???

I LOVE this slide!!
This is EXACTLY the issue that set me on the path to discovering the pattern that constitutes ENTERPRISE ARCHITECTURE

**Automated Rules**

Code Tables, Parameter Settings, Procedural Code, Implementation Rule Statements, Help Messages

Ron Ross
Business Agility Manifesto
The "Enterprise Ontology"

The Framework for PRODUCT Architecture

Scope - Future Limits

"Customer Requirements"

"Engineering Design"

"Mfg. Eng. Specification"

Tooling Configuration

The Airplane or Whatever

Manufacturing Terminology
The "Enterprise Ontology"

The Framework for PRODUCT Architecture

Bills of Material  
Functional Specs  
Drawings/Geometry  
Operating Instructions  
Timing diagrams  
Design Objectives

The Airplane or Whatever

Framework
Origin

Manufacturing
Terminology
The "Enterprise Ontology"

The Zachman Framework for Enterprise Architecture

The Enterprise Ontology

Version 3.0

Scope Lists (Contexts)

Business Models (Concepts)

Systems Models (Logic)

Technology Models (Physics)

Tool Models (Configuration)
The "Enterprise Ontology"

The Zachman Framework for Enterprise Architecture

**Framework Origin**

**Enterprise Terminology**

**THE ENTERPRISE INSTANTIATION**

© 2018 John A. Zachman, Zachman International®
Governing Rules

Automated Rules
Code Tables, Parameter Settings, Procedural Code, Implementation Rule Statements, Help Messages
The “Enterprise Ontology”

The “Business Knowledge-base”

For Engineering & Manufacturing Enterprises

Prerequisite for Agility

This IS Enterprise Architecture
(The Enterprise Architecture “Periodic Table”)
Framework Graphic


(For a publication release of the Framework Graphic send requests to the Contact Us link on zachman.com)

You may be interested in several articles by John A. Zachman at Zachman.com

“Architecture Is Architecture Is Architecture”
“John Zachman’s Concise Definition of the Zachman Framework” and
“The Zachman Framework Evolution” by John P. Zachman
Engineering Versus Manufacturing

**Engineering View**
One Variable - Total Product

**Manufacturing View**
One Part - Multiple Variables

© 1990-2011 John A. Zachman, Zachman International®
Engineering Versus Manufacturing

Engineering View
One Variable - Total Product

Manufacturing View
One Part - Multiple Variables
REMEMBER!!!

ALL business knowledge and ALL Principles need not be completed or in constant practice to continue operations or do new work. This can be defined/designed little by little.
The "Enterprise Ontology"

The "Business Knowledge-base"

For Engineering & Manufacturing Enterprises

Prerequisite for Agility
The "Enterprise Ontology"

The "Business Knowledge-base"

For Engineering & Manufacturing Enterprises

Prerequisite for Agility
The “Business Knowledge-base”
For Engineering & Manufacturing Enterprises

Prerequisite for Agility
The “Enterprise Ontology”

The “Business Knowledge-base”

For Engineering & Manufacturing Enterprises

Prerequisite for Agility

The Zachman Framework for Enterprise Architecture

© 2018 John A. Zachman, Zachman International®
Introduction to Enterprise Architecture

A Zachman Framework Story

John A. Zachman
Zachman International

© 2015 John A. Zachman, Zachman International®
In two weeks - identified new terrorist Cell and linkages with all existing cells.
Generated - Not Built
Key to Diagnosis
This is NOT a “System”
The Key

1. Single-variable, precisely unique, relevant (not arbitrary), ontologically-defined components.

2. Binary Relationships (only two components at a time).

The Key to diagnosing the CEO’s Problems and prescribing alternative solutions

This IS an (incomplete) Enterprise Architecture (Not Enterprise-wide, No Relationship Entities)

A “system” REUSES these Architecture components.

© 2015 John A. Zachman, Zachman International®
Introduction to Enterprise Architecture

Profession
Service Cycle

John A. Zachman
Zachman International

© 2015 John A. Zachman, Zachman International®
Roger Greer:
Dean
School of Library and Information Management
University of Southern California
(My notes from a 1991, IBM GUIDE Conference presentation)

Here is the metaphor:

The Enterprise is the PATIENT and Management is the Brain.
The Enterprise Architect **ought to be** the DOCTOR and IT is the Technician.
Conclusions

John A. Zachman
Zachman International
1. The ontological structure is defined quite independently from the subject domain.

2. Single variable independent “Primitives,” the equivalent of “elements.”

3. Populate with “Binary” Models from which ANY complex, composite structure can be created.
**Designed for Change**

Separation of Independent Variables  
(change one thing without changing everything)

Normalization  
(eliminate redundancy unless specifically controlled)

Reusability  
(changes to a “primitive” changes it for every deployment)

Mass Customization  
(custom enterprises, mass-produced in quantities of one for immediate delivery - dynamic reconfiguration)

Enterprise Architecture  
(Knowledgebase - precisely represents the “as is,” “real time” description of the Enterprise)

Enterprise Impact Analysis  
(Problem diagnosis)
Challenge to Enterprise Architects

Reframe the concept of Enterprise Architecture:

It is not about implementation (running code)!

It is about solving Enterprise problems (while iteratively and incrementally building out the inventory of complete, reusable, Primitive Models), the descriptive representations (that is, the Knowledgebase), that constitutes Enterprise Architecture.

(Accommodating extreme complexity and extreme change.)