

Builder-Architected Systems &
The Zachman Framework

Information Systems Architecture

Builder-Architected Systems

Form-First Architectures

Form-First Architecture

- Form-First Architecture
 - The builder bases a construct's architecture on existing architectures or innovative technologies, rather than on client needs
 - Predesigned homes
 - Innovative computers or cars
 - This approach may be based on familiarity with architectures and reuse of modules
 - Form-first allows a builder to be an innovator

Form-First Architecture

- Competitive Strategies for Builders:
 - Disrupt & Dominate
 - Disrupt existing operational patterns and build barriers to prevent others from taking advantage
 - Agile Response
 - Quick and effective reaction to market fluctuations
 - Attrition
 - Obtain advantage from being largest builder with extensive low-cost operations

Form-First Architecture

- The form-first approach can be risky for the builder
 - Will the clients move up to new designs? Will they consider a need for the new designs?
 - Is the timing right for an innovation or is the existing architecture outdated?
 - Innovative new architectures often require substantial replacements/adjustments in the client's entire system.
 - E.g. Just-In-Time manufacturing requiring warehouse and ordering automation

Form-First Architecture

- Reducing form-first risks:
 - Build in options to allow architectural flexibility and/or modularity
 - Use open architectures to ensure market fit
 - Examine the assumptions underlying the new architecture: are they still valid?
 - Prototype the new architecture for customers
 - Precursor or partial configurations
 - Ensure the prototype can be realized as demonstrated so that expectations can be met

Form-First Follow-Up

- If a Form-First project is a success, sustaining it can be difficult:
 - Companies may find little support for a follow-up product with an evolutionary or all-new architecture
 - E.g. IBM's PC -> OS/2
 - Lack of a successful follow-up may then drain the organization's resources, not allowing it to compete effectively

Form-First Follow-Up

- So how can Form-First success be sustained?
 - Control the architecture
 - Use competitors for add-ons to extend the architecture

Successful architectures are proprietary, but open.

 - The book cites IBM's PC as a good example – do you agree? What would be another example?

Form-First Follow-Up

- Sometimes successful architectures become obsolete and need to be abandoned
 - Original architecture may no longer be effective
 - Competitors have bypassed the architecture
 - Original architecture may be too costly or risky
 - Apple II -> Macintosh
 - Saturn V -> Space Shuttle
- This requires strategic, sometimes drastic action by the organization

Form-First Teams

- Architecture team members greatly influence the outcome of the architecting process
 - Start-up companies need to recruit talent
 - Established companies need to allow for innovative talent to enact change
- The best architectures are the product of...
 - A single mind?
 - A team of single mind?
 - A single vision?

Form-First Teams

- Architecture teams must be:
 - Creative, communicative, respected, and oriented towards a singular goal
- But also:
 - Cultural in form, diverse in nature, obsessive in dedication

The team that created and built a presently successful product is often the best one for its evolution – but seldom for creating its replacement.

Form-First & Basic Research

- Form-First architecture, basic research, and corporate R&D are closely linked
 - Research into new technologies, architectures, products, and applications is both costly and risky
 - As a result, there is an increasing amount of communication among research centers and between academia and business

IS Architecture Frameworks

The Zachman Framework

Introduction to Zachman's ISA

- Purpose of Information Systems Architecture:
 - To develop, maintain, and document large and complex information systems
 - "The purpose of the ISA framework is to show how everything fits together." (Zachman, 1992)
- Foundation of this framework
 - Based on concepts from classical architecture, engineering, and manufacturing

Introduction to Zachman's ISA

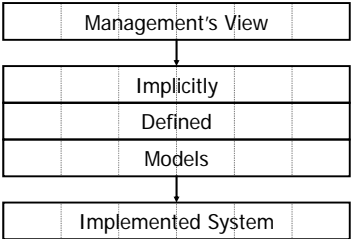


CEOs declare that the biggest challenge facing the modern enterprise is "change."
There's no way to change any complex thing quickly (or safely) unless you start with the descriptive representations of the thing itself.

J.A. Zachman, 1997
Enterprise Architecture: The Issue of the Century

Introduction to Zachman's ISA

- The Great Disconnect:



Introduction to Zachman's ISA



It likely will be necessary to develop some kind of framework for rationalizing the various architectural concepts and specifications in order to provide for clarity of professional communication, to allow for improving and integrating development methodologies and tools, and to establish credibility and confidence in the investment of systems resources.

J.A. Zachman, 1987
A Framework for Information Systems Architecture

Zachman's Definition of Architecture



That set of design artifacts or descriptive representations that are relevant for describing an object, such that it can be produced to requirements (quality) as well as maintained over the period of its useful life (change).

J.A. Zachman, 1997
Enterprise Architecture: The Issue of the Century

Artifact: An object or descriptive or architectural representation produced by human workmanship.

The Zachman Framework - ISA



The Zachman Framework is a classification scheme for design artifacts, that is, descriptive representations, of any complex object.

Inmon, Zachman & Geiger, 1997
Data Stores, Data Warehousing, and the Zachman Framework

The Zachman Framework - ISA



The Zachman Framework is a chart-of-accounts for keeping track of our architectural deliverables.

Von Halle, 1996
Architecting in a Virtual World

The Zachman Framework - ISA



The Zachman framework is a logical structure intended to provide a comprehensive representation of an information technology enterprise.

Whatis.com, 2004

The Zachman Framework - ISA

- One way to deal with increasing complexity, increasing uncertainty, and reduced time-frames:

Use
Precise
Communication!

Zachman develops his framework

- 1987: "A Framework for Information Systems Architecture"
 - Introduces ISA foundations and concepts
 - Includes 5 perspectives & 3 characteristics
- 1992: "Extending and Formalizing the Framework for Information Systems Architecture"
 - Extends with 3 additional characteristics
 - Defines ISA framework rules
 - Introduces "conceptual graphs" for notation

The ISA framework catches on...

- ISA rapidly becomes the default framework for IS architecture discussion
 - E.g., Barbara von Halle's article on "Architecting in a Virtual World"
- 1990: Katz reports how IBM applies ISA to its business & enterprise modeling activities
- 1993: Framework Software Inc. develops "Structure" which supports ISA usage.
- 1995: Schoch & Laplante adapt the ISA for a real-time systems context.

The ISA framework catches on...

- 1996: Cook's book "Building Enterprise Information Architectures" applies ISA from a business reengineering perspective
- 1996: Evernden spins off the Information Framework (IFW)
- 1997: Zachman teams up with Data Warehousing guru William Inmon.
- 2000: US Dept. of Treasury integrates the framework into its Treasury Enterprise Architecture Framework (TEAF).
- 2001: Rational Software integrates the framework with RUP and UML.
- 2003: IBM purchases Rational Software for \$2.1 billion.

Impact of the Framework

- IS includes more than data & function
- Importance of different players and their perspectives
- Search for new forms of description and better integration
 - Including exciting new developments to be covered in future programs

ISA and Granularity

- Granularity
 - Granular: consisting of or appearing to consist of granules
 - Granule: a small particle; especially: one of numerous particles forming a larger unit
- Within an IS architecture:
 - The level of detail of architectural representations

ISA and Granularity



Granularity is of paramount importance to today's enterprise.

For the entire history of data processing, we have been operating at the lowest levels of architectural granularity.

J.A. Zachman , 1997
Enterprise Architecture: The Issue of the Century

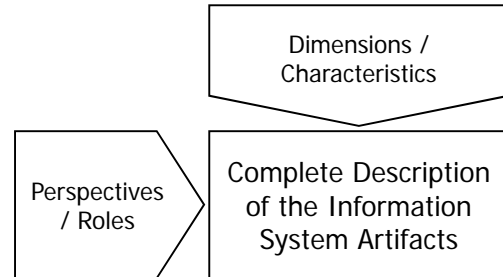
ISA and Granularity



Our implementations at the program, system, application, and departmental levels have inadvertently introduced discontinuity and redundancies into the enterprise data, the network, and the business rules.

J.A. Zachman , 1997
Enterprise Architecture: The Issue of the Century

Zachman's Information Systems Architecture Framework



ISA Perspectives



Complex products don't have a single architectural representation. Instead, they have a set of representations depicting different constraints implicit in different perspectives - or roles - being played in the process of producing the product.

J.A. Zachman , 1997
Enterprise Architecture: The Issue of the Century

ISA Perspectives

- Each perspective is a "layer" of the total picture and represents a total view of the solution from a particular perspective.
- Each perspective adds value to the development of the structure.
- Perspectives are not levels of understanding, complexity, or detail.

ISA Perspectives

- Each perspective adds constraints to the development of the structure. Constraints are additive in that they often affect all other perspectives.
- If a constraint in a lower perspective is inconsistent with a model in a higher perspective, the respective designers - and possibly the owner - need to negotiate a solution.

ISA Perspectives

- Zachman's complete ISA framework contains 6 perspectives:
 - Planner / Scope
 - Owner / Enterprise Model
 - Architect / System Model
 - Builder / Technology Model
 - Subcontractor / Components
 - User / Functioning System

Perspectives Example

- Building a home:
 - **Planner** - Real estate developer with property
 - **Owner** - Home owner who purchases the home
 - **Builder** - Contractor who needs sequence of construction activities
 - **Subcontractor** - Electrician, plumber, etc.
 - **User** - Owner's family who'll live in the home
 - **Architect** - Architect needs to comprehend, design, coordinate, and satisfy all perspectives!

ISA Perspectives

- The planner is concerned with positioning the product in the context of its environment, including specifying its scope
- The owner is interested in the business deliverable and how it will be used
- The designer works with the specifications for the product to ensure that it will, in fact, fulfill the owner's expectations

ISA Perspectives

- The builder manages the process of assembling and fabricating the components in production of the product
- The subcontractor fabricates out-of-context (and hence reusable) components which meet the builder's specifications.
- The user receives the product of the planning, design, and construction activities.

ISA Perspectives in IT

Zachman Perspective	IT Perspective	Deliverable
Planner	Business Executive	The scope of the investigation for the software product
Owner	Business Representative	The vision or business view of the software product to be developed along with the appropriate funding
Designer	Systems Analyst	The detailed physical representation of the software product to be built
Builder	Developer/ Programmer	The detailed specifications of how the software is to be built
Subcontractor	Quality Analyst	The actual software code and file definitions
User	Business End User	The software product

ISA Perspectives

	Data	Function	Network	People	Time	Motivation
Scope	The content of these cells defines the scope of the enterprise, identifying what should possibly be modeled.					
Enterprise	These cells comprise the Business Model – the Owner's expectations from a business perspective for the operating enterprise.					
System	These cell models comprise the technology neutral System Model – the Designers plan for enabling the Business Model					
Technology	These cell models comprise the Technology Model – the Builder's plan for applying technology to the System Model					
Components	These cells are listings, identifying the actual solutions that have been implemented.					
Functional	The Functioning enterprise.					

ISA Dimensions



The architect's representations of the intersecting characteristics - the "material," "function," and "geometry" - were actually descriptions of *what* the product was made of, *how* the product worked, and *where* the components were located relative to one another.

J.A. Zachman , 1997
Enterprise Architecture: The Issue of the Century

ISA Dimensions



The reason for isolating one variable (abstraction) while suppressing all others is to contain the complexity of the design problem.

For example, it is complicated enough to design the process-to-process relationships of an enterprise without attempting to address the entity-to-entity and location-to-location design issues at the same time.

J.F. Sowa & J.A. Zachman , 1992
Extending and formalizing the framework for ISA

ISA Dimensions

- **Starting Point: fundamental questions regarding the structure:**

- What is it made of?
- How does it function?
- Where are things located?
- Who is involved and what do they do?
- When do events happen?
- Why are things done?

ISA Dimensions

- **ISA answers to these questions:**

- What is it made of? - Data, Things
- How does it function? - Functions, Transformations
- Where are things located? - Network, Transportation
- Who is involved? - Job Descriptions, Work Organization
- When do events happen? - Time, Work Cycles
- Why are things done? - Motivation, Rationale

The Tangible ISA Dimensions

- The data column contains business items, entities, relationships, object classes, databases, and data
- The function column represents the system's interrelated activities, functions, and processes
- The network column specifies where the systems components (processors, servers, printers, terminals, etc) are located or used

The Intangible ISA Dimensions

- The people column describes the organizations, divisions, departments, and functions of the people for whom the IS is intended
- The time column describes the events and cycles of the enterprise, along with their dependencies and interrelations
- The motivation column describes the reasons why the enterprise exists and why the IS is required to do certain things.

Information Systems Architecture

ORIGINAL ARCHITECTURE DIMENSIONS - 1987

	DATA	FUNCTION	NETWORK
SCOPE			
ENTERPRISE MODEL			
SYSTEM MODEL			
TECHNOLOGY MODEL			
COMPONENTS			
FUNCTIONING SYSTEM			

Information Systems Architecture

ADDITIONAL ARCHITECTURE DIMENSIONS - 1992

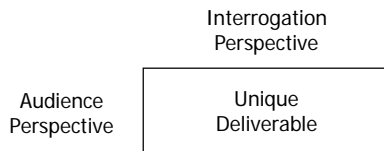
	PEOPLE	TIME	MOTIVATION
SCOPE			
ENTERPRISE MODEL			
SYSTEM MODEL			
TECHNOLOGY MODEL			
COMPONENTS			
FUNCTIONING SYSTEM			

ISA Framework Rules

- Dimension Importance
 - The columns have no order of priority or sequence
- Dimension Simplicity
 - Each column has a simple, basic model to describe its portion of the enterprise and the IS architecture. These models are interdependent.
- Dimension Uniqueness
 - The basic model of each column must be unique so that each artifact can be unambiguously classified and retrieved
- Perspective Uniqueness
 - Each row represents a distinct, unique perspective
- Cell Uniqueness
 - Cell content cannot be found in more than one cell.

ISA Framework Rules

- Each cell is a discrete deliverable



ISA Framework Rules

- Dimension Necessity
 - All 6 dimensions are needed to represent a complete model for each perspective.

DATA Employee	FUNCTION Hire	NETWORK HR Office
PEOPLE Recruiters	TIME end of new candidate evals	MOTIVATION Fill open job

ISA Framework Rules

- Logic Recursiveness
 - Version Recursiveness: alternative systems versions (such as existing and one or more planned) may be kept
 - Decomposition Recursiveness: alternative levels of detail or granularity may be kept

So What Goes Into Those Cells?

Metadata!

- “Metadata is data about data.”
- But it’s not just for databases anymore!
- Metadata can be about hardware configurations, network topologies, program functionality, job responsibilities, process phases, strategy statements, etc.

ISA in English

	DATA (What)	FUNCTION (How)	NETWORK (Where)
OBJECTIVE/SCOPE (Contextual)	List of Things Important in the Business	List of Core Business Processes	List of Business Locations
ENTERPRISE MODEL (Conceptual)	Conceptual Data / Object Model	Business Process Model	Business Logistics System
SYSTEM MODEL (Logical)	Logical Data / Class Model	System Architecture Model	Distributed Systems Architecture
TECHNOLOGY MODEL (Physical)	Physical Data / Class Model	Technology Design Model	Technology Architecture
COMPONENT MODEL (Out of Context)	Data Definitions	Program	Network Architecture
FUNCTIONING SYSTEM	Usable Data	Working Function	Usable Network

ISA in English

	PEOPLE (Who)	TIME (When)	MOTIVATION (Why)
OBJECTIVE/SCOPE (Contextual)	List of Important Organizations	List of Significant Events	List of Business Goals
ENTERPRISE MODEL (Conceptual)	Work Flow Model	Master Schedule	Business Plan
SYSTEM MODEL (Logical)	Human Interface Architecture	Processing Structure	Business Rule Model
TECHNOLOGY MODEL (Physical)	Presentation Architecture	Control Structure	Rule Design
COMPONENT MODEL (Out of Context)	Security Architecture	Timing Definition	Rule Specification
FUNCTIONING SYSTEM	Functioning Organization	Implemented Schedule	Working Strategy

Do we really need all perspectives?

Suppose that you are in charge of a large organization's data model, and you have developed an Entity Relationship Diagram that covers all functions and processes of the organization and tracks all entities, attributes, and relationships that are relevant...

How clear or informative would such a detailed model be to the organization's CEO?

Do we really need all perspectives?

How would you explain the data tracked by the organization's information systems to the CEO?

What would you leave out of the ERD?

What terms would you change to make them easier to understand?

What additional information would need to be added?

Do we really need all that detail?

/// John Zachman advocates developing enterprise-wide architecture at an enormous level of detail, in the same way as an aerospace company needs blueprints for everything down to nuts and bolts.

Some regard this as an extreme position in terms of vertical scope, but it can certainly be justified when compared with the lifetime costs of alternatives.

Zachman's argument is that information systems are not special. In other industries where very expensive, complex things are built, and where there is an expectation of repair or change, models are kept at an enormous level of detail, with concurrent expense.

TOGAF – The Open Group Architectural Framework, 2007.

Information Systems Architecture Subsets

	Data	Function	Network	People	Time	Motivation
Scope	Business Architecture					
Enterprise						
System	Data Architecture					
Technology						
Components						
Functional						

Information Systems Architecture Subsets

	Data	Function	Network	People	Time	Motivation
Scope	Business Architecture					
Enterprise	Business Architecture					
System	Business Architecture					
Technology	Application Architecture					
Components						
Functional						

Information Systems Architecture Subsets

	Data	Function	Network	People	Time	Motivation
Scope	Business Architecture					
Enterprise	Business Architecture					
System	Business Architecture					
Technology			Technology Architecture			
Components						
Functional						

ISA and the UML

Source: Rational Software

	Motivation (Why)	People (Who)	Function (How)	Data (What)	Time (When)	Network (Where)
Scope View	Vision Needs	Vision Stakeholders	Vision Features	Business entities	Business workflow	
Owner's View	Business rules	Actors	Use cases	Business object model	Use case flow of events	Network configurations
Designer's View	Constraints, multiplicity, workflow attributes	Boundary classes	Use case realizations	Business classes	Interaction diagrams	Deployment model
Builder's View		End-user support material	Components	Data model	Process model	Process-to-side mapping
Detailed View		UI design classes	Design classes	Columns, types, keys, indexes	State machines	

The Relevance of the Framework

- The organization is a complex entity comprised of many components which are inter-connected in many ways.
 - The best way to manage and control an organization is through integration and optimization.
 - Ignorance of some components may allow for inappropriate, outdated, inefficient, or poorly designed components to act as weak links.

The Relevance of the Framework

- The Framework provides an organization with a complete inventory of all aspects of its existing, de-facto information systems architecture.
 - Lacking such an inventory makes it difficult to move the organization forward.
 - Organizations may maintain a current and a target framework.
 - Gap analysis results in planned development and maintenance projects.

The Relevance of the Framework

- The Framework captures systems development as an interactive process involving various participants.
 - Participants have simultaneous perspectives with specific requirements and constraints
 - Systems development should therefore be a collaborative process.
 - Systems Planning ↔ Planner
 - Analysis ↔ Owner
 - Logical Design ↔ Architect
 - Technical Design ↔ Builder
 - Construction ↔ Subcontractor
 - Implementation/Use ↔ User

The Relevance of the Framework

4. The Framework stresses that both the organization and its information systems should be engineered.
 - Methodical design and development.
 - Engineering design objectives:
 - Alignment, integration, reusability, interoperability, flexibility, reduced-time-to-market, etc.
 - True integration of strategy and architecture.

The Payoff of ISA



Enterprise architecture initiatives prove that doing it right (that is, architecting for the enterprise as a whole) is faster and cheaper.

Every day that you don't build something that's redundant you are starting to generate benefit; every time a piece of data is reused, the enterprise gets a 100-percent return on its investment.

J.A. Zachman , 1997
Enterprise Architecture: The Issue of the Century

The Payoff of ISA

- Resolving the Great Disconnect:

	Management's View	
	Implicitly	
	Defined	
	Models	
	Implemented System	

Zachman's Conclusion



Someday you're going to wish you had every one of these models, both horizontally and vertically integrated, at an excruciating level of detail. And the best time to start is now.

J.A. Zachman
Traditional conclusion to his seminars

ENTERPRISE ARCHITECTURE - A FRAMEWORK™

	DATA	Wks	FUNCTION	Wks	NETWORK	Wks	PEOPLE	Wks	TIME	Wks	MOTIVATION	Wks	
SCOPE (CONTEXTUAL)	List of Things Important to the Business		List of Processes for Business Processes		List of Locations in which the Business Operates		List of Organizations Important to the Business		List of Events Significant to the Business		List of Business Goals/Strat. Critical Success Factors		SCOPE (CONTEXTUAL)
Planner	Planner		Planner		Planner		Planner		Planner		Planner		Planner
ENTERPRISE MODEL (CONCEPTUAL)	e.g. Segment Model		e.g. Business Process Model		e.g. Logistics Network		e.g. Work Flow Model		e.g. Master Schedule		e.g. Business Plan		ENTERPRISE MODEL (CONCEPTUAL)
Owner	Owner		Owner		Owner		Owner		Owner		Owner		Owner
SYSTEM MODEL (LOGICAL)	e.g. Logical Data Model		e.g. Application Architecture		e.g. Distributed System Architecture		e.g. Human Interface Architecture		e.g. Processing Structure		e.g. Business Rule Model		SYSTEM MODEL (LOGICAL)
Designer	Designer		Designer		Designer		Designer		Designer		Designer		Designer
TECHNOLOGY MODEL (PHYSICAL)	e.g. Physical Data Model		e.g. System Design		e.g. System Architecture		e.g. Presentation Architecture		e.g. Control Structure		e.g. Risk Design		TECHNOLOGY MODEL (PHYSICAL)
Builder	Builder		Builder		Builder		Builder		Builder		Builder		Builder
DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)	e.g. Data Dictionary		e.g. Program		e.g. Network Architecture		e.g. Security Architecture		e.g. Timing Definition		e.g. Risk Definition		DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)
Sub-Constructor	Sub-Constructor		Sub-Constructor		Sub-Constructor		Sub-Constructor		Sub-Constructor		Sub-Constructor		Sub-Constructor
FUNCTIONING ENTERPRISE	e.g. DATA		e.g. FUNCTION		e.g. NETWORK		e.g. ORGANIZATION		e.g. SCHEDULE		e.g. STRATEGY		FUNCTIONING ENTERPRISE

Zachman Institute for Framework Advancement - (810) 231-0531

Copyright - John A. Zachman, Zachman International