Roger Sessions

An Introduction to the Major EA Methodologies

ObjectWatch

Presentation Overview

- Introductions
- Why EA?
- Overview of Methodologies
- Zachman
- TOGAF
- FEA
- Agile EA
- VPEC-T
- SIP
- Putting Them Together
- Summary

Roger Sessions

- CEO of ObjectWatch, specializing in Enterprise Architectures
- Seven Books, including *Simple Architectures for Complex Enterprises* published by Microsoft Press
- Keynote speaker at more than 100 conferences
- Author of more than 100 articles
- Board of Directors of The International Association of Software Architects
- Editor-in-Chief of Perspectives of The International Association of Software Architects
- Writer and Creator of *The ObjectWatch Newsletter*
- Microsoft MVP

Legal Notices

Presentation Overview

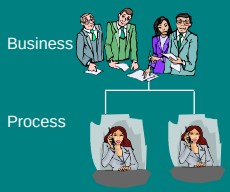
- Introductions
- Why EA?
- Overview of Methodologies
- Zachman
- TOGAF
- FEA
- Agile EA
- VPEC-T
- SIP
- Putting Them Together
- Summary

Architectural Levels: Process

Process 

Domain of Concern: How a business process works.
Example: Loans are taking too long to approve.

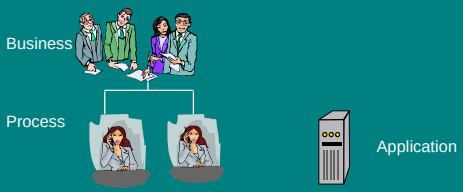
Architectural Levels: Business



Domain of Concern: How processes work together.

Example: Customer information is inconsistent between Loan Origination and Collections.

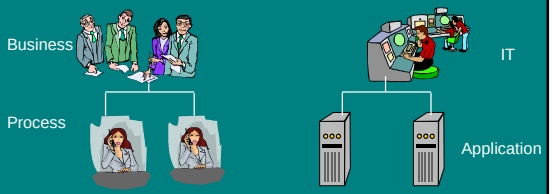
Architectural Levels: Application



Domain of Concern: How an application works.

Example: Loan Applications crashes when loan amount is less than zero.

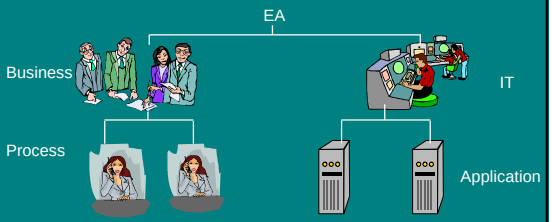
Architectural Levels: IT



Domain of Concern: How applications work together.

Example: Loan applications can't automatically move from the Origination application to the Approval application.

Architectural Levels: EA



Domain of Concern: How IT delivers business value.

Example: The last major IT initiative lacked critical capabilities the business needed.

Examples of EA Problems

- Up-to-date data is not available for making agile business decisions.
- We want to acquire a new company and we need to make sure that their processes can be handled by our information systems.
- We need to make sure that our IT systems can meet new regulatory requirements.
- Our competition spends half as much on IT maintenance than we do.
- We need to automate our partner relationships.
- We are beginning a new complex IT project and we can't afford to have it fail.
- We are trying to decide whether we should build a new system from scratch or purchase an existing off-the-shelf system.

Presentation Overview

- Introductions
- Why EA?
- Overview of Methodologies
- Zachman
- TOGAF
- FEA
- Agile EA
- VPEC-T
- SIP
- Putting Them Together
- Summary

Definitions

- **Taxonomy**: A way of categorizing things.
- **Process**: A recipe for making things.
- **Focused Process**: A recipe for making some aspect of things.
- **Framework**: A description for how things fit together.
- **Model**: A way of thinking about things.
- **Architecture**: A description of how something is organized.
- **Architect**: One who is responsible for an creating an architecture.
- **Architectural Artifact**: Some item that contributes to an architecture.

Acronym Soup

- **AEA**: Agile Enterprise Architecture
- **FEA**: Federal Enterprise Architecture
- **TOGAF**: The Open Group Architecture Framework
- **SIP**: Simple Iterative Partitions
- **VPEC-T**: Value, Policy, Events, Content, Trust
- **Zachman**: John Zachman's Framework

Comparison by Approach

- **Architectural Taxonomy**: Zachman.
- **Architectural Process**: TOGAF, AEA.
- **Architectural Framework**: FEA.
- **Focused Architectural Process**: SIP, VPEC-T
- **Architectural Model**: SIP

Comparison by Generation

First Generation Methodologies
Time Period: 1987-1995
Mantra: We need to align IT and business!
Methodologies: Zachman

Second Generation Methodologies
Time Period: 1995-2003
Mantra: We need a process to follow!
Methodologies: FEA, TOGAF

Third Generation Methodologies
Time Period: 2005-
Mantra: This is taking way too long!
Methodologies: VPEC-T, AEA, SIP

Presentation Overview

- Introductions
- Why EA?
- Overview of Methodologies
- Zachman
- TOGAF
- FEA
- Agile EA
- VPEC-T
- SIP
- Putting Them Together
- Summary

Zachman Essentials

- Introduced in 1987
- *A framework for information systems architecture*, by J.A. Zachman in IBM Systems Journal 1987
- Best Source of Information: *Enterprise Architecture Using the Zachman Framework* by Carol O'Rourke, Neal Fishman, and Warren Selkow.
- Web Site: www.zifa.com
- Main Proponent: John Zachman

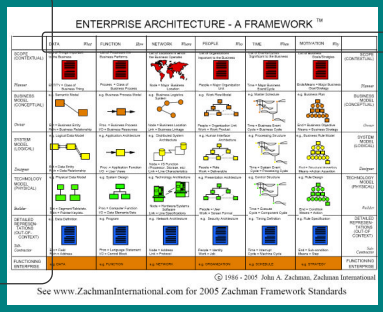
Zachman Ideas

- Introduced the idea of Enterprise Architecture
- An architecture needs to consider every important aspect from the perspective of every important stakeholder.
- IT perspectives should be driven by Business perspectives

Zachman Diagram

Perspective

System Attribute



Zachman Wrap-up

Drawbacks

- No process to follow.
- Not specific to enterprise architecture.
- Nobody fills in all cells... which are necessary?

Favorite Quotation

We are having difficulties communicating with one another about information systems architecture, because a set of architectural representations exists, instead of a single architecture. One is not right and another wrong. The architectures are different. - John Zachman

Presentation Overview

- Introductions
- Why EA?
- Overview of Methodologies
- Zachman
- TOGAF
- FEA
- Agile EA
- VPEC-T
- SIP
- Putting Them Together
- Summary

TOGAF Essentials

- Originally based on TAFIM (circa 1998, now defunct).
- Most recent version: 8.1.1
- Owner: The Open Group
- Definitive Reference*: Guide to Enterprise IT Architecture by* Col Perks and Tony Beveridge
- Web Site: www.opengroup.org/architecture/togaf8/downloads.htm
- Main Proponent: IBM

TOGAF Ideas

- There should be a standard process that everybody follows.
- That standard should be controlled by an industry consortium that is platform agnostic.
- All information on that standard should be available freely to anybody.

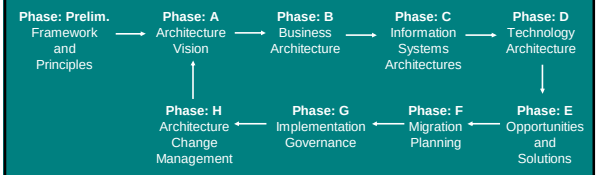
What TOGAF Includes

- An Enterprise Continuum
- A Technical Reference Model
- A Standards Information Base
- A reference model for an Enterprise Architecture.
- An Enterprise Architectural Process called ADM (Architecture Development Method.

Four Pieces of a TOGAF EA

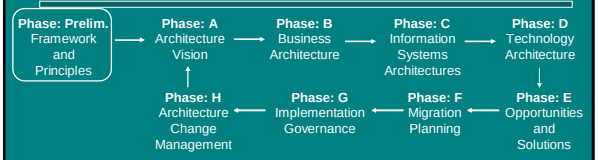
- Business Architecture that describes how the business meets its goals.
- Application Architecture that describes how applications are designed and interact with each other.
- Data Architecture that describes how the enterprise data stores are organized and accessed.
- Technical Architecture that describes the software and hardware infrastructure.

The TOGAF Process



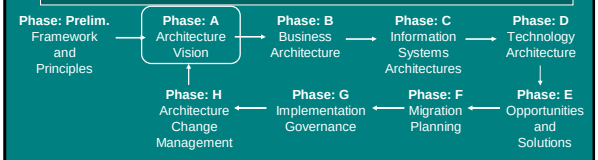
ADM: Architecture Development Method

The TOGAF Process



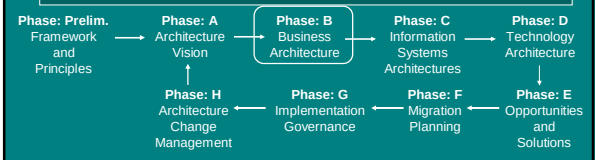
- Introduce TOGAF process
- Modify process as necessary
- Set up governance
- Putting Them Together

The TOGAF Process



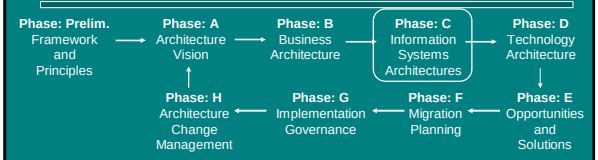
- Receive *Request For Architectural Work*.
- Define high level start and end goals.
- Create *Statement of Architectural Work*.
- Approve statement by all stakeholders.

The TOGAF Process



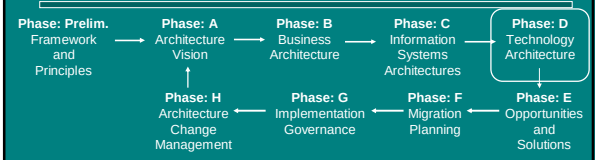
- Create detailed baseline and target business architecture.
- Create business models.
- Develop technical requirements.
- Describe migration plan.

The TOGAF Process



- Create baseline and target IS architecture
- Review and validate reference models, viewpoints, etc.
- Map business architecture to CRUD data operations.
- Deliver *Target Data and Applications Architecture*.

The TOGAF Process



- Determine infrastructure requirements.
- Determine baseline and target architectures.

Remaining phases continue like this.

TOGAF Wrap-up

Drawbacks

- Process is long and tortuous.
- Idea of data architecture is outmoded.
- Artifacts are poorly defined.
- "Standard" is so flexible as to be meaningless.

Favorite Quotation

TOGAF is not wholly specific with respect to generated documents; in fact, it provides very little in the way of prescriptive document templates – merely guidelines for inputs and outputs. (Col Perks and Tony Beveridge, *Guide to Enterprise IT Architecture*).

Presentation Overview

- Introductions
- Why EA?
- Overview of Methodologies
- Zachman
- TOGAF
- FEA
- Agile EA
- VPEC-T
- SIP
- Putting Them Together
- Summary

FEA Essentials

- Specific to the U.S. Federal Government
- Replaces FEAF (Federal Enterprise Architecture Framework)
- Owned by Office of Management and Budget (OMB - a branch of The White House)
- Use of this framework is required by the budgetary process.

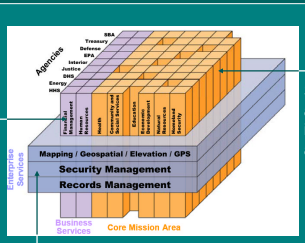
FEA Ideas

- The Federal Government needs a standard framework that all government bodies can follow.
- This standard framework will allow government bodies to share data, code, and processes.
- This sharing will result in a highly efficient government.

What FEA Includes

- A "Segment" Model
- A set of references models, including business, service, components, technical, and data.
- An ADM-like process.
- A transitional process for migrating from a pre-EA to post-EA paradigm.
- A taxonomy for describing EA assets.
- A maturity model to measure success in applying EA.

The FEA Segment Model



Services that can be implemented once and deployed in each organ-ization

Services that are specific to individual agencies.

Services that can be deployed centrally and shared across the government.

FEA Wrap-up

Drawbacks

- Process is even more long and tortuous than ADM.
- There is little evidence that this is helping.
- The process is largely tied to the Federal Government.

Favorite Quotation

[FEA is] a common language and framework to describe and analyze IT investments, enhance collaboration and ultimately transform the Federal government into a citizen-centered, results-oriented, and market-based organization as set forth in the President's Management Agenda. (OMB)

Presentation Overview

- Introductions
- Why EA?
- Overview of Methodologies
- Zachman
- TOGAF
- FEA
- Agile EA
- VPEC-T
- SIP
- Putting Them Together
- Summary

AEA Essentials

- Introduced: 2007
- Focal Point: Agile TOGAF
- Definitive Reference: www.AgileEA.com mostly by Charles Edwards (looking for help!)
- Main Proponent: Charles Edwards

AEA Ideas

- EA needs agile concepts such as adaptability, agility, continuous improvement, similar to what Agile Development provides.
- Take the EA one month at a time (called "Sprints".
- Keep list of hot work items within a bigger picture.
- Daily status meetings ("Scrum Meetings") to evaluate progress.
- Focus on time-boxing the AE process.

AEA Wrap-up

- AEA is still in Beta, ETA end of year.
- The goal of making TOGAF agile is good, but not yet clear how AEA will achieve this.
- Think of AEA as an evolving set of Best Practices and open source tools for making TOGAF usable.

Favorite Quotation: A lot of thought has gone into software development and software architecture disciplines over the last few decades, with the Agile community at the forefront and making excellent progress more recently. The same cannot be said for the discipline of Enterprise Architecture (EA)...
– Charles Edwards

Presentation Overview

- Introductions
- Why EA?
- Overview of Methodologies
- Zachman
- TOGAF
- FEA
- Agile EA
- VPEC-T
- SIP
- Putting Them Together
- Summary

VPEC-T Essentials

- Introduced: 2006
- Focal Point: Business/IT Communications
- Definitive Reference: *Lost in Translation* by Nigel Green and Carl Bate
- Main Proponent: Capgemini
- Web Site: www.LITHandbook.com

VPEC-T Ideas

- IT Failures are primarily due to poor communications.
- IT doesn't understand business speak.
- Business doesn't understand IT speak.
- We need a common language that both understand.
- This language should facilitate a clear understanding of what the business needs from IT.
- The issues that are least comfortable to discuss may be the most important to understand.

VPEC-T In Practice

Every major discussion, from business requirements to technical design needs to include a VPEC-T dialogue.

The PECs of VPEC-T

- P stands for Policies

Definition: The policies of an organization define the rules that govern activities within that organization.

Example: Any loan over $100,000 must be approved by a VP.

Note: Policies often describe workflow patterns.

The PECs of VPEC-T

- E stands for Events
  - Definition: Events are widely visible changes in the state of one or more business activities.
  - Example: Any loan that is overdue by more than 60 days triggers a loan-default event.
  - Note: The concept of events naturally lead to EDA (Event Driven Architectures)

The PECs of VPEC-T

- C stands for Content
  - Definition: Content is the information that is needed to tie an event to a particular business entity and to interpret that meaning of that event.
  - Example: The contents of loan defaults includes the name of the loan holder, the amount of the loan, the date of the last payment, the amount outstanding, the number of days past due.
  - Note: In an services-oriented architecture, content would be the definition of the message.

The Tough Parts of VPEC-T

- V stands for Values
  - Definition: Values are the agenda items (often unwritten) that drive patterns of behavior.
  - Example: Loan-Origination only cares about how many loans it originates each day (this is how it is measured) it doesn't care how many of those loans are eventually approved.
  - Note: This information is rarely captured in business requirements, yet can be critical determining the success of a project.

The Toughest Part of VPEC-T

- T stands for Trust
  - Definition: Trust is a measure of confidence that one party has in another.
  - Example: Loan-Approvals has very little trust in the data that Loan-Origination collects and neither group believes that IT can deliver a working system.
  - Note: This information is not only rarely captured, but is rarely verbalized outside of some trusted circle. Yet if trust issues are not captured, they fester and cannot be addressed.

VPEC-T Wrap-up

Strengths

- Focus on meaningful communications that can make or break projects.
- Can be applied at many different levels.

Drawbacks

- Gives little guidance as to how to get people to discuss the hard issues.
- Success will be determined by the skills of the facilitator.

Favorite Quotation

A Trust issue might be the single largest barrier to adoption of an IT solution. – *Lost in Translation* by Nigel Green and Carl Bate.

Presentation Overview

- Introductions
- Why EA?
- Overview of Methodologies
- Zachman
- TOGAF
- FEA
- Agile EA
- VPEC-T
- SIP
- Putting Them Together
- Summary

SIP Essentials

- Introduced: 2007
- Precedent: Software Fortress Methodology
- Focal Point: EA Complexity
- Definitive Reference: *Simple Architectures for Complex Enterprises* by Roger Sessions
- Web Site: www.objectwatch.com
- Main Proponent: ObjectWatch

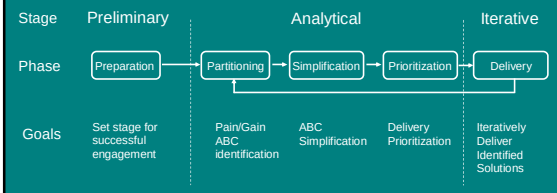
SIP Ideas

- EA needs to be based on rational, reproducible processes.
- The main inhibitor to delivering business value from IT is complexity.
- The most important goal of EA is to remove complexity.
- A model for EA complexity based on mathematics allows us to validate architectures before they are built.

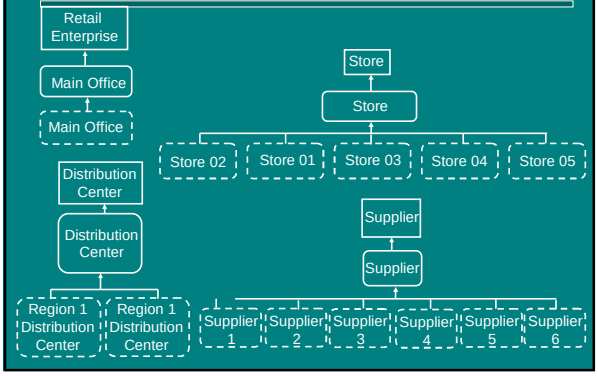
What SIP Includes

- A model for EA complexity based on probability theory, set theory, and equivalence relations.
- A process for partitioning an enterprise into autonomous units called Autonomous Business Capabilities (ABCs)
- A process for prioritizing the delivery of those ABCs based on risk, cost, and value.
- An iterative delivery model of those ABCs that is amenable to agile development.

The SIP Process



The SIP Enterprise Model



How SIP Reduces Complexity

- Partitioning of the enterprise into ABCs.
- Functional reduction within ABCs.
- Consolidation of similar ABCs.
- Elimination of unnecessary ABCs.

SIP Wrap-up

Strengths
- Focus on controlling complexity.

Drawbacks
- SIP focuses only on the issue of complexity. It can be augmented with other processes as needed.
- It is based on math which can feel intimidating.

Favorite Quotation

Complexity is the Enemy.
– *Simple Architectures for Complex Enterprises* by Roger Sessions.

Presentation Overview

- Introductions
- Why EA?
- Overview of Methodologies
- Zachman
- TOGAF
- FEA
- Agile EA
- VPEC-T
- SIP
- **Putting Them Together**
- Summary

None are Complete

- TOGAF is all process, no deliverables.
- Zachman is all deliverables, no process.
- FEA is too big.
- SIP is too small.
- VPEC-T is all talk, no action.
- AEA is all action, no talk.

The bottom line: you need a little of everything.

One Possibility

- Start with SIP to address the burning issue of complexity.
- Introduce VPEC-T to guide the discussions.
- Aim for a segmented architecture a la FEA.
- Once you have created SIP style subsets, use TOGAF to align the business and technical pieces within them.
- Keep TOGAF under control with AEA.
- Manage your artifacts with a Zachman approach.

But Whatever You Do

- Measure the business value your EA approach delivers.
- Track the time it takes to deliver that value.
- If you can't deliver measurable value quickly, change your process.
- EA is about value, not process.

Presentation Overview

- Introductions
- Why EA?
- Overview of Methodologies
- Zachman
- TOGAF
- FEA
- Agile EA
- VPEC-T
- SIP
- Putting Them Together
- **Summary**

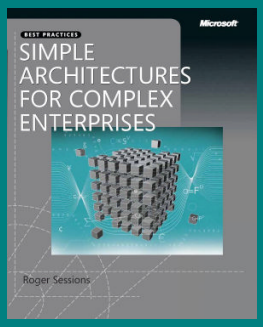
The High Points

- Zachman was the first. It helps organize artifacts and relate them to each other.
- TOGAF introduces formal processes and AEA may make those processes usable someday.
- FEA is a government monster but includes a useful model for enterprise segmentation.
- SIP breaks a complicated enterprise into manageable morsels and focuses business and IT on their common enemy: complexity.
- VPEC-T helps business and IT communicate in a positive, non-confrontational style.

The Take Home Message

- The methodologies are not mutually exclusive.
- Each has strengths and weaknesses.
- Your goal should be to use the best of each to meet your enterprise needs.
- Keep focused on why you care about EA : NOT spinning through processes, but delivering the highest possible business value from your IT investments.

For more on SIP



www.objectwatch.com
roger@objectwatch.com